```
    {
      "cell_type": "markdown",
      "source": [
        "# Introduction to Credit Card Approval Prediction Project:\n",
        "\n",
        "In the financial sector, particularly in the credit card industry, the ability to accurately assess an individual's creditworthiness is paramount. As banks strive to mitigate risk while acquiring new credit card customers, the integration of data science methodologies has become increasingly crucial.\n",
        "\n",
        "This project focuses on leveraging machine learning approaches to predict credit card approval based on customer information. Traditionally, banks have relied on application scorecards to assess creditworthiness, integrating factors such as Credit Bureau Scores (e.g., FICO Score in the US, CIBIL Score in India), and internal data. However, with the advent of advanced data analytics and the availability of vast amounts of external data, banks can now enhance their credit assessment processes significantly.\n",
        "\n",
        "In this project, we aim to develop a predictive model that utilizes both traditional and non-traditional data sources to assess credit card applicants' risk levels. By analyzing historical data and employing machine learning algorithms, we seek to create a robust credit scoring system that can accurately predict credit card approval decisions.\n",
        "\n",
        "## Crisp-DM Approach:\n",
        "\n",
        "The Cross-Industry Standard Process for Data Mining (CRISP-DM) is a widely-used methodology for conducting data mining projects. It consists of six phases: Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment.\n",
        "\n",
        "### Business Understanding:\n",
        "\n",
        "In this phase, we define the project objectives and requirements from a business perspective. We aim to understand the bank's credit card approval process and identify the key factors influencing creditworthiness.\n",
        "\n",
        "### Data Understanding:\n",
        "\n",
        "We explore the available data to gain insights into its structure, quality, and relationships. This involves data profiling, descriptive statistics, and visualization techniques to understand the characteristics of the dataset.\n",
        "\n",
        "### Data Preparation:\n",
        "\n",
        "Data preparation involves cleaning, transforming, and integrating the data to create a dataset suitable for modeling. This includes handling missing values, encoding categorical variables, and scaling numerical features.\n",
        "\n",
        "### Modeling:\n",
        "\n",
        "In the modeling phase, we build and train machine learning models using the
prepared dataset. We experiment with various algorithms and techniques to identify the
most suitable model for predicting credit card approval.\n",
        "\n",
        "### Evaluation:\n",
        "\n",
        "The evaluation phase involves assessing the performance of the trained models
using appropriate evaluation metrics. We analyze the model's accuracy, precision,
recall, and F1-score to determine its effectiveness in predicting credit card
approval.\n",
        "\n",
        "### Deployment:\n",
        "\n",
        "Once we have a satisfactory model, we deploy it into production, making it
available for use in the bank's credit card approval process. We monitor the model's
performance and periodically retrain it to adapt to changing trends and data
patterns.\n",
        "\n",
        "By following the CRISP-DM approach, we aim to develop a robust and effective
credit card approval prediction system that enhances the bank's decision-making
process and improves overall risk management.\n"
      ],
      "metadata": {
        "id": "wKjVRwtz4loC"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "# Business understanding"
      ],
      "metadata": {
        "id": "7qyLqllGv4bh"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "## Importance of Credit Card Approval Prediction in Today's World\n",
        "\n",
        "Predicting credit card approval is crucial in today's world for several
reasons:\n",
        "\n",
        "1. **Risk Management:** Banks need to mitigate the risk of default and fraud
associated with issuing credit cards. By predicting the creditworthiness of applicants,
banks can make informed decisions and reduce the likelihood of defaults.\n",
        "\n",
        "2. **Customer Satisfaction:** Approving credit cards for customers who are likely to use them responsibly can enhance customer satisfaction and loyalty. Predictive models can help tailor offerings to individual customers, providing them with suitable credit options.\n",
        "\n",
        "3. **Efficiency:** Automated credit card approval systems powered by machine learning models can streamline the application process, reduce manual effort, and speed up decision-making, leading to a more efficient banking operation.\n",
        "\n",
        "## Impact on the Banking Sector\n",
        "\n",
        "The implementation of credit card approval prediction models can significantly impact the banking sector:\n",
        "\n",
        "1. **Improved Decision-Making:** Banks can make more accurate and data-driven decisions regarding credit card approvals, leading to better risk management and increased profitability.\n",
        "\n",
        "2. **Cost Reduction:** Automated approval systems can reduce the time and resources required for manual credit assessment, resulting in cost savings for banks.\n",
        "\n",
        "3. **Competitive Advantage:** Banks that deploy advanced predictive models for credit card approval may gain a competitive edge by offering faster and more reliable services to customers.\n",
        "\n",
        "## Addressing Knowledge Gaps and Future Prospects in India\n",
        "\n",
        "In the Indian banking sector, there may be gaps in the adoption of advanced analytics and machine learning techniques for credit risk assessment. Here's how the proposed method can address these gaps and offer future prospects:\n",
        "\n",
        "1. **Data Accessibility:** Access to comprehensive and accurate data is essential for building robust credit card approval prediction models. Banks in India can leverage data analytics and technology to improve data collection, integration, and analysis processes.\n",
        "\n",
        "2. **Regulatory Compliance:** Compliance with regulatory requirements is critical in the banking sector. Machine learning models for credit card approval must comply with regulatory guidelines to ensure fairness, transparency, and accountability.\n",
        "\n",
        "3. **Capacity Building:** Banks need to invest in training and upskilling their workforce to understand and utilize advanced analytics tools effectively. Collaborations with academic institutions and industry experts can facilitate knowledge transfer and skill development in the field of credit risk management.\n",
        "\n",
        "By addressing these knowledge gaps and embracing innovative approaches to credit card approval prediction, banks in India can enhance their risk management
```

```
practices, improve customer experiences, and drive sustainable growth in the financial
sector.\n"
      ],
      "metadata": {
        "id": "fZbSdSz7MXhx"
      }
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "S7vq_ykIHs9T"
      },
      "source": [
        "# Loading Data"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "5Dl7siOAN6O7"
      },
      "outputs": [],
      "source": [
        "# importing necessary libraries\n",
        "\n",
        "import pandas as pd\n",
        "import matplotlib.pyplot as plt\n",
        "import seaborn as sns\n",
        "from sklearn.metrics import accuracy_score\n",
        "from sklearn.metrics import classification_report\n",
        "from sklearn.model_selection import cross_val_predict\n",
        "from sklearn.metrics import confusion_matrix\n",
        "from sklearn.metrics import precision_score, recall_score, roc_auc_score\n",
        "data = pd.read_csv(\"/content/Credit_card.csv\")\n",
        "labels = pd.read_csv(\"/content/Credit_card_label.csv\")"
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "As the data and the label is split into two datasets,first we have to join
the two datasets"
      ],
      "metadata": {
        "id": "sS1QfQFaAd6c"
      }
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "rwUylyobOVA4"
      },
      "outputs": [],
      "source": [
        "#merging data and label\n",
        "raw_merged_data = pd.merge(data,labels,on = \"Ind_ID\")"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "UDfF9OKGOX_l",
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 313
        },
        "outputId": "fe748648-b75f-4f7f-e517-228155542e5a"
      },
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "    Ind_ID GENDER Car_Owner Propert_Owner  CHILDREN  Annual_income
\\\n",
              "0  5008827      M         Y             Y         0       180000.0
\n",
              "1  5009744      F         Y             N         0       315000.0
\n",
              "2  5009746      F         Y             N         0       315000.0
\n",
              "3  5009749      F         Y             N         0            NaN
\n",
              "4  5009752      F         Y             N         0       315000.0
\n",
              "\n",
              "             Type_Income         EDUCATION Marital_status
Housing_type  \\\n",
              "0            Pensioner  Higher education        Married  House /
apartment   \n",
              "1  Commercial associate  Higher education        Married  House /
apartment   \n",
              "2  Commercial associate  Higher education        Married  House /
apartment   \n",
              "3  Commercial associate  Higher education        Married  House /
apartment   \n",
              "4  Commercial associate  Higher education        Married  House /
apartment   \n",
              "\n",
              "   Birthday_count  Employed_days  Mobile_phone  Work_Phone  Phone
EMAIL_ID  \\\n",
              "0        -18772.0         365243             1           0      0
0   \n",
              "1        -13557.0           -586             1           1      1
0   \n",
              "2             NaN           -586             1           1      1
0   \n",
              "3        -13557.0           -586             1           1      1
0   \n",
              "4        -13557.0           -586             1           1      1
0   \n",
              "\n",
              "  Type_Occupation  Family_Members  label  \n",
              "0             NaN               2      1  \n",
              "1             NaN               2      1  \n",
              "2             NaN               2      1  \n",
              "3             NaN               2      1  \n",
              "4             NaN               2      1  "
            ],
            "text/html": [
              "\n",
              "  <div id=\"df-04a5c404-9969-433d-99c4-7458bc0e79d9\" class=\"colab-df-
container\">\n",
              "    <div>\n",
              "<style scoped>\n",
              "    .dataframe tbody tr th:only-of-type {\n",
              "        vertical-align: middle;\n",
              "    }\n",
              "\n",
              "    .dataframe tbody tr th {\n",
              "        vertical-align: top;\n",
              "    }\n",
              "\n",
              "    .dataframe thead th {\n",
              "        text-align: right;\n",
              "    }\n",
              "</style>\n",
              "<table border=\"1\" class=\"dataframe\">\n",
              "  <thead>\n",
              "    <tr style=\"text-align: right;\">\n",
              "      <th></th>\n",
              "      <th>Ind_ID</th>\n",
              "      <th>GENDER</th>\n",
"      <th>Car_Owner</th>\n",
"      <th>Propert_Owner</th>\n",
"      <th>CHILDREN</th>\n",
"      <th>Annual_income</th>\n",
"      <th>Type_Income</th>\n",
"      <th>EDUCATION</th>\n",
"      <th>Marital_status</th>\n",
"      <th>Housing_type</th>\n",
"      <th>Birthday_count</th>\n",
"      <th>Employed_days</th>\n",
"      <th>Mobile_phone</th>\n",
"      <th>Work_Phone</th>\n",
"      <th>Phone</th>\n",
"      <th>EMAIL_ID</th>\n",
"      <th>Type_Occupation</th>\n",
"      <th>Family_Members</th>\n",
"      <th>label</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>0</th>\n",
"      <td>5008827</td>\n",
"      <td>M</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>0</td>\n",
"      <td>180000.0</td>\n",
"      <td>Pensioner</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>-18772.0</td>\n",
"      <td>365243</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>NaN</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>1</th>\n",
"      <td>5009744</td>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>N</td>\n",
"      <td>0</td>\n",
″      <td>315000.0</td>\n″,
″      <td>Commercial associate</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
″      <td>House / apartment</td>\n″,
″      <td>-13557.0</td>\n″,
″      <td>-586</td>\n″,
″      <td>1</td>\n″,
″      <td>1</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>NaN</td>\n″,
″      <td>2</td>\n″,
″      <td>1</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>2</th>\n″,
″      <td>5009746</td>\n″,
″      <td>F</td>\n″,
″      <td>Y</td>\n″,
″      <td>N</td>\n″,
″      <td>0</td>\n″,
″      <td>315000.0</td>\n″,
″      <td>Commercial associate</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
″      <td>House / apartment</td>\n″,
″      <td>NaN</td>\n″,
″      <td>-586</td>\n″,
″      <td>1</td>\n″,
″      <td>1</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>NaN</td>\n″,
″      <td>2</td>\n″,
″      <td>1</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>3</th>\n″,
″      <td>5009749</td>\n″,
″      <td>F</td>\n″,
″      <td>Y</td>\n″,
″      <td>N</td>\n″,
″      <td>0</td>\n″,
″      <td>NaN</td>\n″,
″      <td>Commercial associate</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
″      <td>House / apartment</td>\n″,
              "      <td>-13557.0</td>\n",
              "      <td>-586</td>\n",
              "      <td>1</td>\n",
              "      <td>1</td>\n",
              "      <td>1</td>\n",
              "      <td>0</td>\n",
              "      <td>NaN</td>\n",
              "      <td>2</td>\n",
              "      <td>1</td>\n",
              "    </tr>\n",
              "    <tr>\n",
              "      <th>4</th>\n",
              "      <td>5009752</td>\n",
              "      <td>F</td>\n",
              "      <td>Y</td>\n",
              "      <td>N</td>\n",
              "      <td>0</td>\n",
              "      <td>315000.0</td>\n",
              "      <td>Commercial associate</td>\n",
              "      <td>Higher education</td>\n",
              "      <td>Married</td>\n",
              "      <td>House / apartment</td>\n",
              "      <td>-13557.0</td>\n",
              "      <td>-586</td>\n",
              "      <td>1</td>\n",
              "      <td>1</td>\n",
              "      <td>1</td>\n",
              "      <td>0</td>\n",
              "      <td>NaN</td>\n",
              "      <td>2</td>\n",
              "      <td>1</td>\n",
              "    </tr>\n",
              "  </tbody>\n",
              "</table>\n",
              "</div>\n",
              "    <div class=\"colab-df-buttons\">\n",
              "\n",
              "  <div class=\"colab-df-container\">\n",
              "    <button class=\"colab-df-convert\"
onclick=\"convertToInteractive('df-04a5c404-9969-433d-99c4-7458bc0e79d9')\"\n",
              "            title=\"Convert this dataframe to an interactive
table.\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\" viewBox=\"0
-960 960 960\">\n",
              "    <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220
220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-
160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",
              "  </svg>\n",
              "    </button>\n",
              "\n",
              "  <style>\n",
              "    .colab-df-container {\n",
              "      display:flex;\n",
              "      gap: 12px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert {\n",
              "      background-color: #E8F0FE;\n",
              "      border: none;\n",
              "      border-radius: 50%;\n",
              "      cursor: pointer;\n",
              "      display: none;\n",
              "      fill: #1967D2;\n",
              "      height: 32px;\n",
              "      padding: 0 0 0 0;\n",
              "      width: 32px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert:hover {\n",
              "      background-color: #E2EBFA;\n",
              "      box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n",
              "      fill: #174EA6;\n",
              "    }\n",
              "\n",
              "    .colab-df-buttons div {\n",
              "      margin-bottom: 4px;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert {\n",
              "      background-color: #3B4455;\n",
              "      fill: #D2E3FC;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert:hover {\n",
              "      background-color: #434B5C;\n",
              "      box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",
              "      filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",
              "      fill: #FFFFFF;\n",
              "    }\n",
              "  </style>\n",
              "\n",
              "    <script>\n",
              "      const buttonEl =\n",
              "        document.querySelector('#df-04a5c404-9969-433d-99c4-
7458bc0e79d9 button.colab-df-convert');\n",
              ″      buttonEl.style.display =\n″,
              ″        google.colab.kernel.accessAllowed ? 'block' : 'none';\n″,
              ″\n″,
              ″      async function convertToInteractive(key) {\n″,
              ″        const element = document.querySelector('#df-04a5c404-9969-433d-
99c4-7458bc0e79d9');\n″,
              ″        const dataTable =\n″,
              ″          await
google.colab.kernel.invokeFunction('convertToInteractive',\n″,
              ″                                                    [key], {});\n″,
              ″        if (!dataTable) return;\n″,
              ″\n″,
              ″        const docLinkHtml = 'Like what you see? Visit the ' +\n″,
              ″          '<a target=\″_blank\″
href=https://colab.research.google.com/notebooks/data_table.ipynb>data table
notebook</a>'\n″,
              ″          + ' to learn more about interactive tables.';\n″,
              ″        element.innerHTML = '';\n″,
              ″        dataTable['output_type'] = 'display_data';\n″,
              ″        await google.colab.output.renderOutput(dataTable, element);\n″,
              ″        const docLink = document.createElement('div');\n″,
              ″        docLink.innerHTML = docLinkHtml;\n″,
              ″        element.appendChild(docLink);\n″,
              ″      }\n″,
              ″    </script>\n″,
              ″  </div>\n″,
              ″\n″,
              ″\n″,
              ″<div id=\″df-76ea1270-23ee-4639-8651-ac1286aa39ff\″>\n″,
              ″  <button class=\″colab-df-quickchart\″ onclick=\″quickchart('df-
76ea1270-23ee-4639-8651-ac1286aa39ff')\″\n″,
              ″            title=\″Suggest charts\″\n″,
              ″            style=\″display:none;\″>\n″,
              ″\n″,
              ″<svg xmlns=\″http://www.w3.org/2000/svg\″ height=\″24px\″viewBox=\″0 0
24 24\″\n″,
              ″     width=\″24px\″>\n″,
              ″    <g>\n″,
              ″        <path d=\″M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-
.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\″/>\n″,
              ″    </g>\n″,
              ″</svg>\n″,
              ″  </button>\n″,
              ″\n″,
              ″<style>\n″,
              ″  .colab-df-quickchart {\n″,
              ″      --bg-color: #E8F0FE;\n″,
              ″      --fill-color: #1967D2;\n″,
              ″      --hover-bg-color: #E2EBFA;\n″,
              "      --hover-fill-color: #174EA6;\n",
              "      --disabled-fill-color: #AAA;\n",
              "      --disabled-bg-color: #DDD;\n",
              "  }\n",
              "\n",
              "  [theme=dark] .colab-df-quickchart {\n",
              "      --bg-color: #3B4455;\n",
              "      --fill-color: #D2E3FC;\n",
              "      --hover-bg-color: #434B5C;\n",
              "      --hover-fill-color: #FFFFFF;\n",
              "      --disabled-bg-color: #3B4455;\n",
              "      --disabled-fill-color: #666;\n",
              "  }\n",
              "\n",
              "  .colab-df-quickchart {\n",
              "    background-color: var(--bg-color);\n",
              "    border: none;\n",
              "    border-radius: 50%;\n",
              "    cursor: pointer;\n",
              "    display: none;\n",
              "    fill: var(--fill-color);\n",
              "    height: 32px;\n",
              "    padding: 0;\n",
              "    width: 32px;\n",
              "  }\n",
              "\n",
              "  .colab-df-quickchart:hover {\n",
              "    background-color: var(--hover-bg-color);\n",
              "    box-shadow: 0 1px 2px rgba(60, 64, 67, 0.3), 0 1px 3px 1px rgba(60,
64, 67, 0.15);\n",
              "    fill: var(--button-hover-fill-color);\n",
              "  }\n",
              "\n",
              "  .colab-df-quickchart-complete:disabled,\n",
              "  .colab-df-quickchart-complete:disabled:hover {\n",
              "    background-color: var(--disabled-bg-color);\n",
              "    fill: var(--disabled-fill-color);\n",
              "    box-shadow: none;\n",
              "  }\n",
              "\n",
              "  .colab-df-spinner {\n",
              "    border: 2px solid var(--fill-color);\n",
              "    border-color: transparent;\n",
              "    border-bottom-color: var(--fill-color);\n",
              "    animation:\n",
              "      spin 1s steps(1) infinite;\n",
              "  }\n",
              "\n",
              "  @keyframes spin {\n",
              "    0% {\n",
              "      border-color: transparent;\n",
              "      border-bottom-color: var(--fill-color);\n",
              "      border-left-color: var(--fill-color);\n",
              "    }\n",
              "    20% {\n",
              "      border-color: transparent;\n",
              "      border-left-color: var(--fill-color);\n",
              "      border-top-color: var(--fill-color);\n",
              "    }\n",
              "    30% {\n",
              "      border-color: transparent;\n",
              "      border-left-color: var(--fill-color);\n",
              "      border-top-color: var(--fill-color);\n",
              "      border-right-color: var(--fill-color);\n",
              "    }\n",
              "    40% {\n",
              "      border-color: transparent;\n",
              "      border-right-color: var(--fill-color);\n",
              "      border-top-color: var(--fill-color);\n",
              "    }\n",
              "    60% {\n",
              "      border-color: transparent;\n",
              "      border-right-color: var(--fill-color);\n",
              "    }\n",
              "    80% {\n",
              "      border-color: transparent;\n",
              "      border-right-color: var(--fill-color);\n",
              "      border-bottom-color: var(--fill-color);\n",
              "    }\n",
              "    90% {\n",
              "      border-color: transparent;\n",
              "      border-bottom-color: var(--fill-color);\n",
              "    }\n",
              "  }\n",
              "</style>\n",
              "\n",
              "  <script>\n",
              "    async function quickchart(key) {\n",
              "      const quickchartButtonEl =\n",
              "        document.querySelector('#' + key + ' button');\n",
              "      quickchartButtonEl.disabled = true;  // To prevent multiple
clicks.\n",
              "      quickchartButtonEl.classList.add('colab-df-spinner');\n",
              "      try {\n",
              "        const charts = await google.colab.kernel.invokeFunction(\n",
              "            'suggestCharts', [key], {});\n",
              "      } catch (error) {\n",
              "        console.error('Error during call to suggestCharts:', error);\n",
              "      }\n",
              "      quickchartButtonEl.classList.remove('colab-df-spinner');\n",
              "      quickchartButtonEl.classList.add('colab-df-quickchart-
complete');\n",
              "    }\n",
              "    (() => {\n",
              "      let quickchartButtonEl =\n",
              "        document.querySelector('#df-76ea1270-23ee-4639-8651-
ac1286aa39ff button');\n",
              "      quickchartButtonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "    })();\n",
              "  </script>\n",
              "</div>\n",
              "\n",
              "    </div>\n",
              "  </div>\n"
            ]
          },
          "metadata": {},
          "execution_count": 1280
        }
      ],
      "source": [
        "raw_merged_data.head()"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "m6eOY7sAOYuO",
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 347
        },
        "outputId": "a1d070bb-16b5-4795-c748-1e470e4eb51c"
      },
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "       Ind_ID GENDER Car_Owner Propert_Owner  CHILDREN  Annual_income
\\\n",
              "1543  5028645      F         N             Y         0            NaN
\n",
              "1544  5023655      F         N             N         0       225000.0
\n",
              ″1545  5115992      M         Y             Y         2       180000.0   
\n″,
              ″1546  5118219      M         Y             N         0       270000.0   
\n″,
              ″1547  5053790      F         Y             Y         0       225000.0   
\n″,
              ″\n″,
              ″                Type_Income                      EDUCATION  \\\n″,
              ″1543  Commercial associate               Higher education   \n″,
              ″1544  Commercial associate              Incomplete higher   \n″,
              ″1545               Working               Higher education   \n″,
              ″1546               Working  Secondary / secondary special   \n″,
              ″1547               Working               Higher education   \n″,
              ″\n″,
              ″            Marital_status       Housing_type  Birthday_count  
Employed_days  \\\n″,
              ″1543               Married  House / apartment        -11957.0           
-2182   \n″,
              ″1544  Single / not married  House / apartment        -10229.0           
-1209   \n″,
              ″1545               Married  House / apartment        -13174.0           
-2477   \n″,
              ″1546        Civil marriage  House / apartment        -15292.0            
-645   \n″,
              ″1547               Married  House / apartment        -16601.0           
-2859   \n″,
              ″\n″,
              ″      Mobile_phone  Work_Phone  Phone  EMAIL_ID Type_Occupation  \\\n″,
              ″1543             1           0      0         0        Managers   \n″,
              ″1544             1           0      0         0     Accountants   \n″,
              ″1545             1           0      0         0        Managers   \n″,
              ″1546             1           1      1         0         Drivers   \n″,
              ″1547             1           0      0         0             NaN   \n″,
              ″\n″,
              ″      Family_Members  label  \n″,
              ″1543               2      0  \n″,
              ″1544               1      0  \n″,
              ″1545               4      0  \n″,
              ″1546               2      0  \n″,
              ″1547               2      0  ″
            ],
            ″text/html″: [
              ″\n″,
              ″  <div id=\″df-dc859d97-933f-4608-9409-3a8b4093909d\″ class=\″colab-df-
container\″>\n″,
              ″    <div>\n″,
              ″<style scoped>\n″,
              ″    .dataframe tbody tr th:only-of-type {\n″,
              ″        vertical-align: middle;\n″,
"    }\n",
"\n",
"    .dataframe tbody tr th {\n",
"        vertical-align: top;\n",
"    }\n",
"\n",
"    .dataframe thead th {\n",
"        text-align: right;\n",
"    }\n",
"</style>\n",
"<table border=\"1\" class=\"dataframe\">\n",
"  <thead>\n",
"    <tr style=\"text-align: right;\">\n",
"      <th></th>\n",
"      <th>Ind_ID</th>\n",
"      <th>GENDER</th>\n",
"      <th>Car_Owner</th>\n",
"      <th>Propert_Owner</th>\n",
"      <th>CHILDREN</th>\n",
"      <th>Annual_income</th>\n",
"      <th>Type_Income</th>\n",
"      <th>EDUCATION</th>\n",
"      <th>Marital_status</th>\n",
"      <th>Housing_type</th>\n",
"      <th>Birthday_count</th>\n",
"      <th>Employed_days</th>\n",
"      <th>Mobile_phone</th>\n",
"      <th>Work_Phone</th>\n",
"      <th>Phone</th>\n",
"      <th>EMAIL_ID</th>\n",
"      <th>Type_Occupation</th>\n",
"      <th>Family_Members</th>\n",
"      <th>label</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>1543</th>\n",
"      <td>5028645</td>\n",
"      <td>F</td>\n",
"      <td>N</td>\n",
"      <td>Y</td>\n",
"      <td>0</td>\n",
"      <td>NaN</td>\n",
"      <td>Commercial associate</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>-11957.0</td>\n",
″      <td>-2182</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Managers</td>\n″,
″      <td>2</td>\n″,
″      <td>0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>1544</th>\n″,
″      <td>5023655</td>\n″,
″      <td>F</td>\n″,
″      <td>N</td>\n″,
″      <td>N</td>\n″,
″      <td>0</td>\n″,
″      <td>225000.0</td>\n″,
″      <td>Commercial associate</td>\n″,
″      <td>Incomplete higher</td>\n″,
″      <td>Single / not married</td>\n″,
″      <td>House / apartment</td>\n″,
″      <td>-10229.0</td>\n″,
″      <td>-1209</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Accountants</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>1545</th>\n″,
″      <td>5115992</td>\n″,
″      <td>M</td>\n″,
″      <td>Y</td>\n″,
″      <td>Y</td>\n″,
″      <td>2</td>\n″,
″      <td>180000.0</td>\n″,
″      <td>Working</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
″      <td>House / apartment</td>\n″,
″      <td>-13174.0</td>\n″,
″      <td>-2477</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Managers</td>\n″,
″      <td>4</td>\n″,
″      <td>0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>1546</th>\n″,
″      <td>5118219</td>\n″,
″      <td>M</td>\n″,
″      <td>Y</td>\n″,
″      <td>N</td>\n″,
″      <td>0</td>\n″,
″      <td>270000.0</td>\n″,
″      <td>Working</td>\n″,
″      <td>Secondary / secondary special</td>\n″,
″      <td>Civil marriage</td>\n″,
″      <td>House / apartment</td>\n″,
″      <td>-15292.0</td>\n″,
″      <td>-645</td>\n″,
″      <td>1</td>\n″,
″      <td>1</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>Drivers</td>\n″,
″      <td>2</td>\n″,
″      <td>0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>1547</th>\n″,
″      <td>5053790</td>\n″,
″      <td>F</td>\n″,
″      <td>Y</td>\n″,
″      <td>Y</td>\n″,
″      <td>0</td>\n″,
″      <td>225000.0</td>\n″,
″      <td>Working</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
″      <td>House / apartment</td>\n″,
″      <td>-16601.0</td>\n″,
″      <td>-2859</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>NaN</td>\n″,
″      <td>2</td>\n″,
″      <td>0</td>\n″,
″    </tr>\n″,
″  </tbody>\n″,
              "</table>\n",
              "</div>\n",
              "    <div class=\"colab-df-buttons\">\n",
              "\n",
              "  <div class=\"colab-df-container\">\n",
              "    <button class=\"colab-df-convert\"
onclick=\"convertToInteractive('df-dc859d97-933f-4608-9409-3a8b4093909d')\"\n",
              "            title=\"Convert this dataframe to an interactive
table.\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\" viewBox=\"0
-960 960 960\">\n",
              "    <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220
220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-
160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",
              "  </svg>\n",
              "    </button>\n",
              "\n",
              "  <style>\n",
              "    .colab-df-container {\n",
              "      display:flex;\n",
              "      gap: 12px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert {\n",
              "      background-color: #E8F0FE;\n",
              "      border: none;\n",
              "      border-radius: 50%;\n",
              "      cursor: pointer;\n",
              "      display: none;\n",
              "      fill: #1967D2;\n",
              "      height: 32px;\n",
              "      padding: 0 0 0 0;\n",
              "      width: 32px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert:hover {\n",
              "      background-color: #E2EBFA;\n",
              "      box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n",
              "      fill: #174EA6;\n",
              "    }\n",
              "\n",
              "    .colab-df-buttons div {\n",
              "      margin-bottom: 4px;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert {\n",
              ″      background-color: #3B4455;\n″,
              ″      fill: #D2E3FC;\n″,
              ″    }\n″,
              ″\n″,
              ″    [theme=dark] .colab-df-convert:hover {\n″,
              ″      background-color: #434B5C;\n″,
              ″      box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n″,
              ″      filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n″,
              ″      fill: #FFFFFF;\n″,
              ″    }\n″,
              ″  </style>\n″,
              ″\n″,
              ″    <script>\n″,
              ″      const buttonEl =\n″,
              ″        document.querySelector('#df-dc859d97-933f-4608-9409-
3a8b4093909d button.colab-df-convert');\n″,
              ″      buttonEl.style.display =\n″,
              ″        google.colab.kernel.accessAllowed ? 'block' : 'none';\n″,
              ″\n″,
              ″      async function convertToInteractive(key) {\n″,
              ″        const element = document.querySelector('#df-dc859d97-933f-4608-
9409-3a8b4093909d');\n″,
              ″        const dataTable =\n″,
              ″          await
google.colab.kernel.invokeFunction('convertToInteractive',\n″,
              ″                                                   [key], {});\n″,
              ″        if (!dataTable) return;\n″,
              ″\n″,
              ″        const docLinkHtml = 'Like what you see? Visit the ' +\n″,
              ″          '<a target=\″_blank\″
href=https://colab.research.google.com/notebooks/data_table.ipynb>data table
notebook</a>'\n″,
              ″          + ' to learn more about interactive tables.';\n″,
              ″        element.innerHTML = '';\n″,
              ″        dataTable['output_type'] = 'display_data';\n″,
              ″        await google.colab.output.renderOutput(dataTable, element);\n″,
              ″        const docLink = document.createElement('div');\n″,
              ″        docLink.innerHTML = docLinkHtml;\n″,
              ″        element.appendChild(docLink);\n″,
              ″      }\n″,
              ″    </script>\n″,
              ″  </div>\n″,
              ″\n″,
              ″\n″,
              ″<div id=\″df-eae96b38-c38e-4165-be2a-41a4b2cd62c6\″>\n″,
              ″  <button class=\″colab-df-quickchart\″ onclick=\″quickchart('df-
eae96b38-c38e-4165-be2a-41a4b2cd62c6')\″\n″,
              ″            title=\″Suggest charts\″\n″,
              ″            style=\″display:none;\″>\n″,
              "\n",
              "<svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0 0
24 24\"\n",
              "     width=\"24px\">\n",
              "    <g>\n",
              "        <path d=\"M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-
.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\"/>\n",
              "    </g>\n",
              "</svg>\n",
              "  </button>\n",
              "\n",
              "<style>\n",
              "  .colab-df-quickchart {\n",
              "      --bg-color: #E8F0FE;\n",
              "      --fill-color: #1967D2;\n",
              "      --hover-bg-color: #E2EBFA;\n",
              "      --hover-fill-color: #174EA6;\n",
              "      --disabled-fill-color: #AAA;\n",
              "      --disabled-bg-color: #DDD;\n",
              "  }\n",
              "\n",
              "  [theme=dark] .colab-df-quickchart {\n",
              "      --bg-color: #3B4455;\n",
              "      --fill-color: #D2E3FC;\n",
              "      --hover-bg-color: #434B5C;\n",
              "      --hover-fill-color: #FFFFFF;\n",
              "      --disabled-bg-color: #3B4455;\n",
              "      --disabled-fill-color: #666;\n",
              "  }\n",
              "\n",
              "  .colab-df-quickchart {\n",
              "    background-color: var(--bg-color);\n",
              "    border: none;\n",
              "    border-radius: 50%;\n",
              "    cursor: pointer;\n",
              "    display: none;\n",
              "    fill: var(--fill-color);\n",
              "    height: 32px;\n",
              "    padding: 0;\n",
              "    width: 32px;\n",
              "  }\n",
              "\n",
              "  .colab-df-quickchart:hover {\n",
              "    background-color: var(--hover-bg-color);\n",
              "    box-shadow: 0 1px 2px rgba(60, 64, 67, 0.3), 0 1px 3px 1px rgba(60,
64, 67, 0.15);\n",
              "    fill: var(--button-hover-fill-color);\n",
              "  }\n",
              "\n",
"  .colab-df-quickchart-complete:disabled,\n",
"  .colab-df-quickchart-complete:disabled:hover {\n",
"    background-color: var(--disabled-bg-color);\n",
"    fill: var(--disabled-fill-color);\n",
"    box-shadow: none;\n",
"  }\n",
"\n",
"  .colab-df-spinner {\n",
"    border: 2px solid var(--fill-color);\n",
"    border-color: transparent;\n",
"    border-bottom-color: var(--fill-color);\n",
"    animation:\n",
"      spin 1s steps(1) infinite;\n",
"  }\n",
"\n",
"  @keyframes spin {\n",
"    0% {\n",
"      border-color: transparent;\n",
"      border-bottom-color: var(--fill-color);\n",
"      border-left-color: var(--fill-color);\n",
"    }\n",
"    20% {\n",
"      border-color: transparent;\n",
"      border-left-color: var(--fill-color);\n",
"      border-top-color: var(--fill-color);\n",
"    }\n",
"    30% {\n",
"      border-color: transparent;\n",
"      border-left-color: var(--fill-color);\n",
"      border-top-color: var(--fill-color);\n",
"      border-right-color: var(--fill-color);\n",
"    }\n",
"    40% {\n",
"      border-color: transparent;\n",
"      border-right-color: var(--fill-color);\n",
"      border-top-color: var(--fill-color);\n",
"    }\n",
"    60% {\n",
"      border-color: transparent;\n",
"      border-right-color: var(--fill-color);\n",
"    }\n",
"    80% {\n",
"      border-color: transparent;\n",
"      border-right-color: var(--fill-color);\n",
"      border-bottom-color: var(--fill-color);\n",
"    }\n",
"    90% {\n",
"      border-color: transparent;\n",
"      border-bottom-color: var(--fill-color);\n",
              "    }\n",
              "  }\n",
              "</style>\n",
              "\n",
              "  <script>\n",
              "    async function quickchart(key) {\n",
              "      const quickchartButtonEl =\n",
              "        document.querySelector('#' + key + ' button');\n",
              "      quickchartButtonEl.disabled = true;  // To prevent multiple
clicks.\n",
              "      quickchartButtonEl.classList.add('colab-df-spinner');\n",
              "      try {\n",
              "        const charts = await google.colab.kernel.invokeFunction(\n",
              "            'suggestCharts', [key], {});\n",
              "      } catch (error) {\n",
              "        console.error('Error during call to suggestCharts:', error);\n",
              "      }\n",
              "      quickchartButtonEl.classList.remove('colab-df-spinner');\n",
              "      quickchartButtonEl.classList.add('colab-df-quickchart-
complete');\n",
              "    }\n",
              "    (() => {\n",
              "      let quickchartButtonEl =\n",
              "        document.querySelector('#df-eae96b38-c38e-4165-be2a-
41a4b2cd62c6 button');\n",
              "      quickchartButtonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "    })();\n",
              "  </script>\n",
              "</div>\n",
              "\n",
              "    </div>\n",
              "  </div>\n"
            ]
          },
          "metadata": {},
          "execution_count": 1281
        }
      ],
      "source": [
        "raw_merged_data.tail()"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "K79vYdzaGoWE"
      },
      "outputs": [],
      "source": [
        "merged_data = raw_merged_data.copy()"
      ]
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "Zkawmcf6Qv2O"
      },
      "source": [
        "# Data Understanding & Data Preparation"
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "Below is the explanation of all the features in the dataset"
      ],
      "metadata": {
        "id": "U-I5klJyAozS"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "| Feature Name   | Description                                      |\n",
        "|----------------|--------------------------------------------------|\n",
        "| Ind_ID         | Client ID                                        |\n",
        "| Gender         | Gender information                               |\n",
        "| Car_owner      | Having car or not                                |\n",
        "| Propert_owner  | Having property or not                           |\n",
        "| Children       | Count of children                                |\n",
        "| Annual_income  | Annual income                                    |\n",
        "| Type_Income    | Income type                                      |\n",
        "| Education      | Education level                                  |\n",
        "| Marital_status | Marital status                                   |\n",
        "| Housing_type   | Living style                                     |\n",
        "| Birthday_count | Use backward count from current day (0), -1 means
yesterday |\n",
        "| Employed_days  | Start date of employment. Use backward count from current
day (0). Positive value means, individual is currently unemployed |\n",
        "| Mobile_phone   | Any mobile phone                                 |\n",
        "| Work_phone     | Any work phone                                   |\n",
        "| Phone          | Any phone number                                 |\n",
        "| EMAIL_ID       | Any email ID                                     |\n",
        "| Type_Occupation| Occupation                                       |\n",
        "| Family Members | Family size                                      |\n",
        "| Label          | 0 is application approved and 1 is application rejected
|\n"
      ],
      "metadata": {
        "id": "DTOvqNJL_weV"
      }
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "Qq332zQZTwu6",
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "outputId": "4ec8f3c8-66d5-41bb-a35f-7082d1ca29c6"
      },
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
            "<class 'pandas.core.frame.DataFrame'>\n",
            "Int64Index: 1548 entries, 0 to 1547\n",
            "Data columns (total 19 columns):\n",
            " #   Column           Non-Null Count  Dtype  \n",
            "---  ------           --------------  -----  \n",
            " 0   Ind_ID           1548 non-null   int64  \n",
            " 1   GENDER           1541 non-null   object \n",
            " 2   Car_Owner        1548 non-null   object \n",
            " 3   Propert_Owner    1548 non-null   object \n",
            " 4   CHILDREN         1548 non-null   int64  \n",
            " 5   Annual_income    1525 non-null   float64\n",
            " 6   Type_Income      1548 non-null   object \n",
            " 7   EDUCATION        1548 non-null   object \n",
            " 8   Marital_status   1548 non-null   object \n",
            " 9   Housing_type     1548 non-null   object \n",
            " 10  Birthday_count   1526 non-null   float64\n",
            " 11  Employed_days    1548 non-null   int64  \n",
            " 12  Mobile_phone     1548 non-null   int64  \n",
            " 13  Work_Phone       1548 non-null   int64  \n",
            " 14  Phone            1548 non-null   int64  \n",
            " 15  EMAIL_ID         1548 non-null   int64  \n",
            " 16  Type_Occupation  1060 non-null   object \n",
            " 17  Family_Members   1548 non-null   int64  \n",
            " 18  label            1548 non-null   int64  \n",
            "dtypes: float64(2), int64(9), object(8)\n",
            "memory usage: 241.9+ KB\n"
          ]
        }
      ],
      "source": [
        "# further Information about data\n",
        "merged_data.info()"
      ]
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "Rwj1x2rSUhSJ"
      },
      "source": [
        "Checking values in all columns"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "HSu8oAw7T2ZD",
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "outputId": "87122e7f-fdfb-4c1d-bf20-e75c1ac352dc"
      },
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
            "Ind_ID ------> Unique -> [5008827 5009744 5009746 ... 5115992 5118219
5053790] | N Unique -> 1548 | Nulls -> 0\n",
            "\n",
            "GENDER ------> Unique -> ['M' 'F' nan] | N Unique -> 2 | Nulls -> 7\n",
            "\n",
            "Car_Owner ------> Unique -> ['Y' 'N'] | N Unique -> 2 | Nulls -> 0\n",
            "\n",
            "Propert_Owner ------> Unique -> ['Y' 'N'] | N Unique -> 2 | Nulls -> 0\n",
            "\n",
            "CHILDREN ------> Unique -> [ 0  1  2  4  3 14] | N Unique -> 6 | Nulls ->
0\n",
            "\n",
            "Annual_income ------> Unique -> [ 180000.   315000.        nan  450000.
90000.   472500.   270000.\n",
            "  126000.   202500.   157500.   112500.   540000.   292500.   135000.\n",
            "   76500.   215100.   225000.    67500.   171000.   103500.    99000.\n",
            "  391500.    65250.    72900.   360000.   256500.   675000.   247500.\n",
            "   85500.   121500.   130500.   211500.    81000.    72000.   148500.\n",
            "  162000.   195750.   585000.   216000.   306000.   108000.    63000.\n",
            "   45000.   337500.   131400.   117000.   445500.   234000.  1575000.\n",
            "  144000.    67050.    73350.   193500.   900000.    94500.   198000.\n",
            "   54000.   166500.   167400.   153000.   423000.   243000.   283500.\n",
            "  252000.   495000.   612000.    36000.   139500.   133650.   427500.\n",
            "  261000.   231750.    90900.    45900.   119250.    58500.   328500.\n",
            "  787500.   594000.   119700.    69372.    37800.   387000.   207000.\n",
            "  189000.   333000.   105750.   382500.   141750.    40500.   405000.\n",
            "   44550.   301500.   351000.   175500.   121900.5  238500.    33750.\n",
            "  116100.   297000.   630000.   418500.    83250.   173250.   274500.\n",
            "  115200.    56250.    95850.   185400.   810000.   184500.   165600.\n",
            "  114750.    47250.    49500.    69750. ] | N Unique -> 115 | Nulls ->
23\n",
            "\n",
            "Type_Income ------> Unique -> ['Pensioner' 'Commercial associate'
'Working' 'State servant'] | N Unique -> 4 | Nulls -> 0\n",
            "\n",
            "EDUCATION ------> Unique -> ['Higher education' 'Secondary / secondary
special' 'Lower secondary'\n",
            " 'Incomplete higher' 'Academic degree'] | N Unique -> 5 | Nulls -> 0\n",
            "\n",
            "Marital_status ------> Unique -> ['Married' 'Single / not married' 'Civil
marriage' 'Separated' 'Widow'] | N Unique -> 5 | Nulls -> 0\n",
            "\n",
            "Housing_type ------> Unique -> ['House / apartment' 'With parents'
'Rented apartment'\n",
            " 'Municipal apartment' 'Co-op apartment' 'Office apartment'] | N Unique -
> 6 | Nulls -> 0\n",
            "\n",
            "Birthday_count ------> Unique -> [-18772. -13557.      nan ... -10229. -
15292. -16601.] | N Unique -> 1270 | Nulls -> 22\n",
            "\n",
            "Employed_days ------> Unique -> [365243   -586   -678  -1002   -913   -
248  -2470  -1644  -4327  -1674\n",
            "  -1086   -925   -854   -185  -3350   -691  -4770  -2394   -384  -3647\n",
            "  -1546   -808  -1285   -855  -7369  -2269  -4114  -1161  -7288   -166\n",
            "   -866   -564  -3366  -2150  -1458  -8369   -508  -1567   -137  -3974\n",
            "  -1570  -2878   -140  -2576  -1905  -1724  -1328  -4987  -3496  -1394\n",
            "  -1265  -2531  -2105   -860   -217   -535  -6866   -584  -2227  -8161\n",
            "   -613   -606  -6944   -346  -1808  -3420   -863  -7413  -7553  -3931\n",
            "  -1039   -134   -622  -1595  -1626  -1868   -555  -1935   -931   -900\n",
            "  -4305   -499  -2418   -189  -1132  -1770   -919  -1081   -502  -2420\n",
            "  -1325  -6367  -2484   -341  -9422  -3054  -2987  -2128   -820   -141\n",
            "  -1692  -4686   -693   -567   -885  -2769  -1547  -3179  -5204  -3072\n",
            "   -320  -2469   -583   -834  -1085  -1399  -7310  -1748  -2479   -875\n",
            "  -2805  -2477  -1477  -1782   -431  -1416   -263  -2349   -412  -7068\n",
            "  -1787  -1345  -3717  -1048  -2667  -2606  -2135  -1534  -2311  -1323\n",
            "  -5061  -2213  -2152  -4509  -1552  -1569  -1679  -8684  -6337    -97\n",
            "  -1222   -531  -7591  -5639  -1776  -5498  -5880   -460  -4532  -3309\n",
″  -3873   -344  -1923   -604  -1922  -1496   -708 -12332  -5209  -6273\n″,
″  -2722  -9363   -746  -1322  -3458   -158  -2457  -2811  -7018  -2026\n″,
″   -188  -2967  -3166  -5107  -1649  -3694  -3697  -4596  -5674  -1682\n″,
″   -196   -530  -1696  -2168 -13382  -1509  -1347  -1405   -227  -9975\n″,
″  -1505  -5084  -2905   -356  -1719  -3680   -962  -3000  -4275   -110\n″,
″  -2078  -8377  -4392   -126  -7640  -3153   -139  -4023  -4888 -14887\n″,
″  -3319  -1966   -200  -6123   -117  -2625  -1763  -3689  -3414  -2910\n″,
″  -3412  -1107  -8036  -3608   -101  -3231   -203  -1589  -3577   -735\n″,
″  -8737  -1617  -1591   -979   -422   -655  -3653  -1261  -7346  -5545\n″,
″  -3810  -1190  -3246  -7676  -1353  -3715  -1113   -992  -2289 -12621\n″,
″  -3643   -869  -2174  -5336  -3535  -4407   -453   -960  -4711  -1965\n″,
″  -2543  -1393  -2234  -1631  -1619  -5981  -2682  -3443  -2330   -413\n″,
″  -6645   -505   -201  -3273  -1462  -2609   -521  -1321  -2467   -716\n″,
″  -1632   -614  -1904   -351  -7260   -626  -6853   -458  -1628  -6094\n″,
″   -734   -657  -2700   -880   -157   -135  -1497  -1266   -515  -1023\n″,
″  -2129   -575  -3262   -430  -2207  -1866  -4568   -309   -578  -6908\n″,
″  -1931  -8033  -2257  -3776  -4583  -4404   -727  -3805  -2316  -1455\n″,
″  -5422   -581   -893  -1860  -6317  -1000  -4766  -2259  -1344  -4690\n″,
″  -8290  -2052  -5004   -923  -1193   -793  -5453    -88  -5495   -593\n″,
″  -4816  -1586  -2104  -1032  -9698  -1131  -1175  -8375  -2199  -1912\n″,
″   -926  -2499  -2654  -1174   -230  -2276  -2197  -2468 -11448   -932\n″,
″  -2956   -619  -3787  -1253  -3242   -298   -124  -4082  -1431  -5330\n″,
″  -1017  -1128  -3088  -1466  -1689  -3003  -4039   -588   -282  -4491\n″,
″  -1155  -1220  -1203   -712  -2844  -4031  -3322  -6693  -2760  -2532\n″,
″  -1646 -11542  -5408  -1374  -3092  -2993  -2379  -1281  -7804  -3002\n″,
″   -825  -2057  -1611   -652  -2537  -2331  -4710  -1191  -1162  -3900\n″,
″   -219   -223  -3034  -3329   -235  -3503  -1324  -3112  -2959   -234\n″,
″   -719  -2616  -4410  -1587  -1537  -1430  -1928  -2753   -867  -2134\n″,
″  -3232    -95  -3690  -9870  -4979  -3396   -461  -5206  -3173   -732\n″,
″   -432  -2102  -6583   -156  -5460  -2506  -3397  -3792  -3230  -2464\n″,
″  -3334  -3110  -2966  -2728  -4161  -7494  -4280  -5891  -1350   -609\n″,
″  -2381  -1613  -3533  -6218  -2874  -1654  -2433  -4534   -522   -798\n″,
″  -2620   -145   -663  -2299  -1812  -2589   -130  -5056   -796  -7341\n″,
″  -1130  -5436  -2368  -1889  -1315  -1678  -2939  -1616  -8405  -3476\n″,
″  -3453  -3769  -1359  -2990  -2321  -4267  -1386   -645   -886  -1823\n″,
″  -4965  -3716  -1557  -3761  -1981   -102   -288  -2147  -3929   -424\n″,
″  -1069  -2323  -3332   -553   -375  -6352  -1056  -1172  -4663  -5587\n″,
″  -1406  -8966  -6860  -1463   -561  -3661  -6764 -10600  -3999  -3960\n″,
″  -2065  -1205  -1135  -5190  -3720  -2633  -3272  -2500  -6033   -170\n″,
″   -278  -3062  -8157  -2653  -1953  -5457  -1997  -3435  -1504  -2696\n″,
″ -13735   -132  -6621  -2481  -2818 -11906  -1077  -9258  -2277  -2422\n″,
″   -826  -2969  -2086  -1070  -8172   -701  -1384  -9447  -1160  -1952\n″,
″  -1354   -469   -563  -1526  -2092  -1749   -168  -4029  -4967  -3282\n″,
″   -339  -7979  -9683   -577  -2622   -976  -1994  -1563  -4523   -340\n″,
″  -6614  -1639  -3392  -3261  -3928  -5082  -3630  -3591  -1199  -4304\n″,
″  -2514  -2187  -7140    -96  -5176  -6761  -2356  -2908  -3333  -3596\n″,
″  -4606  -4030  -5246  -2906  -1915  -1267  -1256  -1721  -2317  -1518\n″,
″   -864  -7364  -1101  -2231  -1327   -355  -1934  -5179  -2326   -980\n″,
″  -1224  -7048  -2266    -73  -2661   -286  -8044   -523   -978   -795\n″,
            ″  -2351  -1114  -3760  -5145  -4119  -3641   -481  -4603   -888   -393\n″,
            ″  -1007 -10688  -2228  -8479  -2180  -1362  -1876  -2265 -11389  -4146\n″,
            ″  -5500  -5521   -658  -3695   -901  -1751  -3663  -3149  -1375   -183\n″,
            ″   -500   -489  -2450  -1539  -1556  -7095   -674  -2605  -1999  -4599\n″,
            ″  -6488  -1054   -745  -3505  -1585  -6904  -2530  -1420   -314  -2441\n″,
            ″ -11884  -4520  -3289  -1495  -3504  -3093  -2417  -2554   -818  -9359\n″,
            ″  -1508 -10364  -4331  -2296  -2142  -3163  -3961  -2896  -1185  -1670\n″,
            ″  -3670  -1202  -4936   -596  -6648   -747  -1741  -1615  -1444  -2089\n″,
            ″   -397  -2592  -1618  -2822  -1600  -2558   -991  -2758  -2988  -2904\n″,
            ″  -3377  -3902  -3202   -671   -848   -552  -2559  -2361  -3854  -1182\n″,
            ″  -2384  -8230  -2493  -2362  -4034  -2143  -2393  -2565   -450  -5496\n″,
            ″  -2319   -191  -3506  -2865  -2339  -2072  -7013  -3008  -2932   -565\n″,
            ″  -1282  -3154  -1225  -2333  -1869 -11940  -3515  -3291   -407   -444\n″,
            ″   -226   -279  -1661   -896   -670  -2533  -7400  -3339  -1259   -265\n″,
            ″  -1870  -2540  -2978   -187   -477  -1436  -1978  -1133  -2985 -10758\n″,
            ″  -4288    -89  -6930  -1807  -3143  -3850  -4560   -785  -3570   -777\n″,
            ″ -11451   -742  -5950   -464   -385  -2475  -2051  -3607  -2454  -7614\n″,
            ″  -2796   -642  -4004  -4917  -5622   -313  -1753  -5830  -1677  -3078\n″,
            ″  -3061   -625  -1381   -308  -5437   -218  -3195  -1198  -1473  -5483\n″,
            ″   -704 -12870  -5373  -7128  -2255  -5345  -3574   -164  -6226  -6678\n″,
            ″  -2024   -890  -6075  -1774   -686  -3229  -3455  -5454 -13010  -5507\n″,
            ″  -6640  -5488   -221   -127  -8760  -7718  -3123  -1652  -6879  -2673\n″,
            ″   -420  -1072  -5760  -1871  -1707  -1962  -1598  -1583  -7264   -536\n″,
            ″  -1313  -2512  -5673  -3380  -3944  -3654   -171  -1801  -1138  -1897\n″,
            ″  -2980  -1581  -6558  -4434   -210  -3032  -2389  -1467   -377 -10762\n″,
            ″  -2788  -3668  -2340  -9925   -918  -4219   -194  -3010   -370   -232\n″,
            ″   -285  -1830  -3853  -2866  -1022  -2423  -2545  -6230   -104  -2631\n″,
            ″   -367   -857  -4887   -433   -803  -3361  -1520  -5155  -5006  -2191\n″,
            ″  -1064   -828  -4874  -1217  -7557  -5334   -611  -5215   -438   -958\n″,
            ″  -3095  -2827  -3693  -2262   -195   -109  -5862  -1200  -1891  -4662\n″,
            ″  -2924  -1648  -3536  -2182  -1209  -2859] | N Unique -> 956 | Nulls ->
0\n″,
            ″\n″,
            ″Mobile_phone ------> Unique -> [1] | N Unique -> 1 | Nulls -> 0\n″,
            ″\n″,
            ″Work_Phone ------> Unique -> [0 1] | N Unique -> 2 | Nulls -> 0\n″,
            ″\n″,
            ″Phone ------> Unique -> [0 1] | N Unique -> 2 | Nulls -> 0\n″,
            ″\n″,
            ″EMAIL_ID ------> Unique -> [0 1] | N Unique -> 2 | Nulls -> 0\n″,
            ″\n″,
            ″Type_Occupation ------> Unique -> [nan 'Core staff' 'Cooking staff'
'Laborers' 'Sales staff' 'Accountants'\n″,
            ″ 'High skill tech staff' 'Managers' 'Cleaning staff' 'Drivers'\n″,
            ″ 'Low-skill Laborers' 'IT staff' 'Waiters/barmen staff' 'Security
staff'\n″,
            ″ 'Medicine staff' 'Private service staff' 'HR staff' 'Secretaries'\n″,
            ″ 'Realty agents'] | N Unique -> 18 | Nulls -> 488\n″,
            ″\n″,
            "Family_Members ------> Unique -> [ 2  3  1  4  6  5 15] | N Unique -> 7 |
Nulls -> 0\n",
            "\n",
            "label ------> Unique -> [1 0] | N Unique -> 2 | Nulls -> 0\n",
            "\n"
          ]
        }
      ],
      "source": [
        "for column in merged_data.columns:\n",
        "  print(f\"{column} ------> Unique -> {merged_data[column].unique()} | N
Unique -> {merged_data[column].nunique()} | Nulls ->
{merged_data[column].isnull().sum()}\", end = \"\\n\\n\")"
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "## Data Cleaning"
      ],
      "metadata": {
        "id": "B6xNJf-fYyvK"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "A value of 365243 employed days means the person is currently unemployed"
      ],
      "metadata": {
        "id": "DsD71R-1x9KR"
      }
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "VQi5zI3APDV1",
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "outputId": "3a63bb31-11ec-403d-e20d-653c52ba5aa4"
      },
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "array([365243])"
            ]
          },
          "metadata": {},
          "execution_count": 1285
        }
      ],
      "source": [
        "merged_data[merged_data[\"Employed_days\"]>0][\"Employed_days\"].unique()\n",
        "# checking for positive values"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "NGY5TJWql8p4"
      },
      "outputs": [],
      "source": [
        "# 365243 means unemployed, so replacing this with 0 and converting days to
positive years\n",
        "merged_data[\"Employed_years\"] =
round((merged_data[\"Employed_days\"].replace({365243:0}))*-1/360,2)"
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "merged_data = merged_data.drop(\"Employed_days\",axis = 1)"
      ],
      "metadata": {
        "id": "aapoeHkNyNVG"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "STZVzJJhWttg"
      },
      "source": [
        "## Treating Nulls"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "8LErie287qDA"
      },
      "outputs": [],
      "source": [
        "# since gender,annual income and birthday count combined consists only 3.36 %
of the total number of rows, it can be safely dropped\n",
        "columns_to_drop = ['GENDER', 'Annual_income', 'Birthday_count']\n",
        "merged_data['Annual_income'] =
merged_data['Annual_income'].fillna(merged_data['Annual_income'].median()) # filling
annual income column null values with median value\n",
        "merged_data['Birthday_count'] =
merged_data['Birthday_count'].fillna(merged_data['Birthday_count'].median())\n",
        "merged_data['GENDER'] = merged_data['GENDER'].fillna(\"F\")"
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "Transforming Employed_days"
      ],
      "metadata": {
        "id": "KCQqKscTM-Il"
      }
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 313
        },
        "id": "KFaUxQOdqMIE",
        "outputId": "99a284a6-9691-4c93-8ec3-c6b733199205"
      },
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "    Ind_ID GENDER Car_Owner Propert_Owner  CHILDREN  Annual_income
\\\n",
              "0  5008827      M         Y             Y         0       180000.0
\n",
              "1  5009744      F         Y             N         0       315000.0
\n",
              "2  5009746      F         Y             N         0       315000.0
\n",
              ″3  5009749      F         Y             N         0       166500.0
\n″,
              ″4  5009752      F         Y             N         0       315000.0
\n″,
              ″\n″,
              ″            Type_Income         EDUCATION Marital_status
Housing_type  \\\n″,
              ″0             Pensioner  Higher education        Married  House /
apartment   \n″,
              ″1  Commercial associate  Higher education        Married  House /
apartment   \n″,
              ″2  Commercial associate  Higher education        Married  House /
apartment   \n″,
              ″3  Commercial associate  Higher education        Married  House /
apartment   \n″,
              ″4  Commercial associate  Higher education        Married  House /
apartment   \n″,
              ″\n″,
              ″   Mobile_phone  Work_Phone  Phone  EMAIL_ID Type_Occupation
Family_Members  \\\n″,
              ″0             1           0      0         0             NaN
2   \n″,
              ″1             1           1      1         0             NaN
2   \n″,
              ″2             1           1      1         0             NaN
2   \n″,
              ″3             1           1      1         0             NaN
2   \n″,
              ″4             1           1      1         0             NaN
2   \n″,
              ″\n″,
              ″   label  Employed_years   Age  \n″,
              ″0      1            0.00  52.0  \n″,
              ″1      1            1.63  38.0  \n″,
              ″2      1            1.63  44.0  \n″,
              ″3      1            1.63  38.0  \n″,
              ″4      1            1.63  38.0  ″
            ],
            ″text/html″: [
              ″\n″,
              ″  <div id=\″df-ecf44e7d-736a-450c-8d48-fb2d9cb0a78b\″ class=\″colab-df-
container\″>\n″,
              ″    <div>\n″,
              ″<style scoped>\n″,
              ″    .dataframe tbody tr th:only-of-type {\n″,
              ″        vertical-align: middle;\n″,
              ″    }\n″,
              ″\n″,
              ″    .dataframe tbody tr th {\n″,
"        vertical-align: top;\n",
"    }\n",
"\n",
"    .dataframe thead th {\n",
"        text-align: right;\n",
"    }\n",
"</style>\n",
"<table border=\"1\" class=\"dataframe\">\n",
"  <thead>\n",
"    <tr style=\"text-align: right;\">\n",
"      <th></th>\n",
"      <th>Ind_ID</th>\n",
"      <th>GENDER</th>\n",
"      <th>Car_Owner</th>\n",
"      <th>Propert_Owner</th>\n",
"      <th>CHILDREN</th>\n",
"      <th>Annual_income</th>\n",
"      <th>Type_Income</th>\n",
"      <th>EDUCATION</th>\n",
"      <th>Marital_status</th>\n",
"      <th>Housing_type</th>\n",
"      <th>Mobile_phone</th>\n",
"      <th>Work_Phone</th>\n",
"      <th>Phone</th>\n",
"      <th>EMAIL_ID</th>\n",
"      <th>Type_Occupation</th>\n",
"      <th>Family_Members</th>\n",
"      <th>label</th>\n",
"      <th>Employed_years</th>\n",
"      <th>Age</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>0</th>\n",
"      <td>5008827</td>\n",
"      <td>M</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>0</td>\n",
"      <td>180000.0</td>\n",
"      <td>Pensioner</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
″      <td>NaN</td>\n″,
″      <td>2</td>\n″,
″      <td>1</td>\n″,
″      <td>0.00</td>\n″,
″      <td>52.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>1</th>\n″,
″      <td>5009744</td>\n″,
″      <td>F</td>\n″,
″      <td>Y</td>\n″,
″      <td>N</td>\n″,
″      <td>0</td>\n″,
″      <td>315000.0</td>\n″,
″      <td>Commercial associate</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
″      <td>House / apartment</td>\n″,
″      <td>1</td>\n″,
″      <td>1</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>NaN</td>\n″,
″      <td>2</td>\n″,
″      <td>1</td>\n″,
″      <td>1.63</td>\n″,
″      <td>38.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>2</th>\n″,
″      <td>5009746</td>\n″,
″      <td>F</td>\n″,
″      <td>Y</td>\n″,
″      <td>N</td>\n″,
″      <td>0</td>\n″,
″      <td>315000.0</td>\n″,
″      <td>Commercial associate</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
″      <td>House / apartment</td>\n″,
″      <td>1</td>\n″,
″      <td>1</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>NaN</td>\n″,
″      <td>2</td>\n″,
″      <td>1</td>\n″,
″      <td>1.63</td>\n″,
″      <td>44.0</td>\n″,
"    </tr>\n",
"    <tr>\n",
"      <th>3</th>\n",
"      <td>5009749</td>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>N</td>\n",
"      <td>0</td>\n",
"      <td>166500.0</td>\n",
"      <td>Commercial associate</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>NaN</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>1.63</td>\n",
"      <td>38.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>4</th>\n",
"      <td>5009752</td>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>N</td>\n",
"      <td>0</td>\n",
"      <td>315000.0</td>\n",
"      <td>Commercial associate</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>NaN</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>1.63</td>\n",
"      <td>38.0</td>\n",
"    </tr>\n",
"  </tbody>\n",
"</table>\n",
"</div>\n",
"    <div class=\"colab-df-buttons\">\n",
              "\n",
              "  <div class=\"colab-df-container\">\n",
              "    <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-ecf44e7d-736a-450c-8d48-fb2d9cb0a78b')\"\n",
              "            title=\"Convert this dataframe to an interactive table.\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\" viewBox=\"0 -960 960 960\">\n",
              "    <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220 220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",
              "  </svg>\n",
              "    </button>\n",
              "\n",
              "  <style>\n",
              "    .colab-df-container {\n",
              "      display:flex;\n",
              "      gap: 12px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert {\n",
              "      background-color: #E8F0FE;\n",
              "      border: none;\n",
              "      border-radius: 50%;\n",
              "      cursor: pointer;\n",
              "      display: none;\n",
              "      fill: #1967D2;\n",
              "      height: 32px;\n",
              "      padding: 0 0 0 0;\n",
              "      width: 32px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert:hover {\n",
              "      background-color: #E2EBFA;\n",
              "      box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px rgba(60, 64, 67, 0.15);\n",
              "      fill: #174EA6;\n",
              "    }\n",
              "\n",
              "    .colab-df-buttons div {\n",
              "      margin-bottom: 4px;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert {\n",
              "      background-color: #3B4455;\n",
              "      fill: #D2E3FC;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert:hover {\n",
              "      background-color: #434B5C;\n",
              "      box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",
              "      filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",
              "      fill: #FFFFFF;\n",
              "    }\n",
              "  </style>\n",
              "\n",
              "    <script>\n",
              "      const buttonEl =\n",
              "        document.querySelector('#df-ecf44e7d-736a-450c-8d48-
fb2d9cb0a78b button.colab-df-convert');\n",
              "      buttonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "\n",
              "      async function convertToInteractive(key) {\n",
              "        const element = document.querySelector('#df-ecf44e7d-736a-450c-
8d48-fb2d9cb0a78b');\n",
              "        const dataTable =\n",
              "          await
google.colab.kernel.invokeFunction('convertToInteractive',\n",
              "                                                     [key], {});\n",
              "        if (!dataTable) return;\n",
              "\n",
              "        const docLinkHtml = 'Like what you see? Visit the ' +\n",
              "          '<a target=\"_blank\"
href=https://colab.research.google.com/notebooks/data_table.ipynb>data table
notebook</a>'\n",
              "          + ' to learn more about interactive tables.';\n",
              "        element.innerHTML = '';\n",
              "        dataTable['output_type'] = 'display_data';\n",
              "        await google.colab.output.renderOutput(dataTable, element);\n",
              "        const docLink = document.createElement('div');\n",
              "        docLink.innerHTML = docLinkHtml;\n",
              "        element.appendChild(docLink);\n",
              "      }\n",
              "    </script>\n",
              "  </div>\n",
              "\n",
              "\n",
              "<div id=\"df-0aa02851-9392-4270-9892-8dd3b4034586\">\n",
              "  <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-
0aa02851-9392-4270-9892-8dd3b4034586')\"\n",
              "            title=\"Suggest charts\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "<svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0 0
24 24\"\n",
"     width=\"24px\">\n",
"    <g>\n",
"        <path d=\"M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\"/>\n",
"    </g>\n",
"</svg>\n",
"  </button>\n",
"\n",
"<style>\n",
"  .colab-df-quickchart {\n",
"      --bg-color: #E8F0FE;\n",
"      --fill-color: #1967D2;\n",
"      --hover-bg-color: #E2EBFA;\n",
"      --hover-fill-color: #174EA6;\n",
"      --disabled-fill-color: #AAA;\n",
"      --disabled-bg-color: #DDD;\n",
"  }\n",
"\n",
"  [theme=dark] .colab-df-quickchart {\n",
"      --bg-color: #3B4455;\n",
"      --fill-color: #D2E3FC;\n",
"      --hover-bg-color: #434B5C;\n",
"      --hover-fill-color: #FFFFFF;\n",
"      --disabled-bg-color: #3B4455;\n",
"      --disabled-fill-color: #666;\n",
"  }\n",
"\n",
"  .colab-df-quickchart {\n",
"    background-color: var(--bg-color);\n",
"    border: none;\n",
"    border-radius: 50%;\n",
"    cursor: pointer;\n",
"    display: none;\n",
"    fill: var(--fill-color);\n",
"    height: 32px;\n",
"    padding: 0;\n",
"    width: 32px;\n",
"  }\n",
"\n",
"  .colab-df-quickchart:hover {\n",
"    background-color: var(--hover-bg-color);\n",
"    box-shadow: 0 1px 2px rgba(60, 64, 67, 0.3), 0 1px 3px 1px rgba(60, 64, 67, 0.15);\n",
"    fill: var(--button-hover-fill-color);\n",
"  }\n",
"\n",
"  .colab-df-quickchart-complete:disabled,\n",
"  .colab-df-quickchart-complete:disabled:hover {\n",
"    background-color: var(--disabled-bg-color);\n",
"    fill: var(--disabled-fill-color);\n",
"    box-shadow: none;\n",
"  }\n",
"\n",
"  .colab-df-spinner {\n",
"    border: 2px solid var(--fill-color);\n",
"    border-color: transparent;\n",
"    border-bottom-color: var(--fill-color);\n",
"    animation:\n",
"      spin 1s steps(1) infinite;\n",
"  }\n",
"\n",
"  @keyframes spin {\n",
"    0% {\n",
"      border-color: transparent;\n",
"      border-bottom-color: var(--fill-color);\n",
"      border-left-color: var(--fill-color);\n",
"    }\n",
"    20% {\n",
"      border-color: transparent;\n",
"      border-left-color: var(--fill-color);\n",
"      border-top-color: var(--fill-color);\n",
"    }\n",
"    30% {\n",
"      border-color: transparent;\n",
"      border-left-color: var(--fill-color);\n",
"      border-top-color: var(--fill-color);\n",
"      border-right-color: var(--fill-color);\n",
"    }\n",
"    40% {\n",
"      border-color: transparent;\n",
"      border-right-color: var(--fill-color);\n",
"      border-top-color: var(--fill-color);\n",
"    }\n",
"    60% {\n",
"      border-color: transparent;\n",
"      border-right-color: var(--fill-color);\n",
"    }\n",
"    80% {\n",
"      border-color: transparent;\n",
"      border-right-color: var(--fill-color);\n",
"      border-bottom-color: var(--fill-color);\n",
"    }\n",
"    90% {\n",
"      border-color: transparent;\n",
"      border-bottom-color: var(--fill-color);\n",
"    }\n",
"  }\n",
"</style>\n",
              "\n",
              "  <script>\n",
              "    async function quickchart(key) {\n",
              "      const quickchartButtonEl =\n",
              "        document.querySelector('#' + key + ' button');\n",
              "      quickchartButtonEl.disabled = true;  // To prevent multiple
clicks.\n",
              "      quickchartButtonEl.classList.add('colab-df-spinner');\n",
              "      try {\n",
              "        const charts = await google.colab.kernel.invokeFunction(\n",
              "            'suggestCharts', [key], {});\n",
              "      } catch (error) {\n",
              "        console.error('Error during call to suggestCharts:', error);\n",
              "      }\n",
              "      quickchartButtonEl.classList.remove('colab-df-spinner');\n",
              "      quickchartButtonEl.classList.add('colab-df-quickchart-
complete');\n",
              "    }\n",
              "    (() => {\n",
              "      let quickchartButtonEl =\n",
              "        document.querySelector('#df-0aa02851-9392-4270-9892-
8dd3b4034586 button');\n",
              "      quickchartButtonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "    })();\n",
              "  </script>\n",
              "</div>\n",
              "\n",
              "    </div>\n",
              "  </div>\n"
            ]
          },
          "metadata": {},
          "execution_count": 1289
        }
      ],
      "source": [
        "# Employed_days\n",
        "merged_data[\"Age\"] = round((merged_data[\"Birthday_count\"]*-1)/360)\n",
        "merged_data = merged_data.drop(columns = \"Birthday_count\")\n",
        "merged_data.head()"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "PWEyx4daQc-1"
      },
      "outputs": [],
      "source": [
        "# columns having missing values\n",
        "ser1 = merged_data.isnull().sum()[merged_data.isnull().sum()>0]\n",
        "nulls = round((ser1/merged_data.shape[0])*100,2)"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "b_VHsNYnW4y4",
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "outputId": "6d8bcdcd-ecb7-4bb7-a8d3-b3bc47631b63"
      },
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "Type_Occupation    31.52\n",
              "dtype: float64"
            ]
          },
          "metadata": {},
          "execution_count": 1291
        }
      ],
      "source": [
        "nulls"
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "Type_Occupation column has 31% null values hence its important to treat this
column"
      ],
      "metadata": {
        "id": "UDiW1xpcLVMu"
      }
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "ulFBdXV3nmnW"
      },
      "source": [
        "### Handling missing values in type occupation column"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "FooPurlrzhll"
      },
      "outputs": [],
      "source": [
        "#replacing values where occ is null but people are employed with Unknown\n",
        "merged_data.loc[(merged_data[\"Type_Occupation\"].isnull()) &
(merged_data[\"Employed_years\"] != 0),\"Type_Occupation\"] = \"Unknown\""
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "NP4DDt51mVuC"
      },
      "outputs": [],
      "source": [
        "merged_data.loc[merged_data[\"Employed_years\"]
==0][\"Type_Occupation\"].unique()\n",
        "# all occupation type values for unemployed people are nan,so nan can be
replaced with Unemployed\n",
        "merged_data.loc[(merged_data[\"Employed_years\"] ==0) &
(merged_data[\"Type_Occupation\"].isnull()),\"Type_Occupation\"] = \"Unemployed\""
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "YBFAkGzG1brR",
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 313
        },
        "outputId": "830a89fe-79c6-4604-e816-8b3ec49fe0e5"
      },
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              ″    Ind_ID GENDER Car_Owner Propert_Owner  CHILDREN  Annual_income
\\\n″,
              ″0  5008827      M         Y             Y         0       180000.0
\n″,
              ″1  5009744      F         Y             N         0       315000.0
\n″,
              ″2  5009746      F         Y             N         0       315000.0
\n″,
              ″3  5009749      F         Y             N         0       166500.0
\n″,
              ″4  5009752      F         Y             N         0       315000.0
\n″,
              ″\n″,
              ″            Type_Income         EDUCATION Marital_status
Housing_type  \\\n″,
              ″0            Pensioner  Higher education        Married  House /
apartment   \n″,
              ″1  Commercial associate  Higher education       Married  House /
apartment   \n″,
              ″2  Commercial associate  Higher education       Married  House /
apartment   \n″,
              ″3  Commercial associate  Higher education       Married  House /
apartment   \n″,
              ″4  Commercial associate  Higher education       Married  House /
apartment   \n″,
              ″\n″,
              ″   Mobile_phone  Work_Phone  Phone  EMAIL_ID Type_Occupation
Family_Members  \\\n″,
              ″0            1           0      0         0      Unemployed
2   \n″,
              ″1            1           1      1         0         Unknown
2   \n″,
              ″2            1           1      1         0         Unknown
2   \n″,
              ″3            1           1      1         0         Unknown
2   \n″,
              ″4            1           1      1         0         Unknown
2   \n″,
              ″\n″,
              ″   label  Employed_years   Age  \n″,
              ″0      1            0.00  52.0  \n″,
              ″1      1            1.63  38.0  \n″,
              ″2      1            1.63  44.0  \n″,
              ″3      1            1.63  38.0  \n″,
              ″4      1            1.63  38.0  ″
             ],
             ″text/html″: [
              ″\n″,
              "  <div id=\"df-f12d71b5-616a-4c24-9547-c07be1401b77\" class=\"colab-df-
container\">\n",
              "    <div>\n",
              "<style scoped>\n",
              "    .dataframe tbody tr th:only-of-type {\n",
              "        vertical-align: middle;\n",
              "    }\n",
              "\n",
              "    .dataframe tbody tr th {\n",
              "        vertical-align: top;\n",
              "    }\n",
              "\n",
              "    .dataframe thead th {\n",
              "        text-align: right;\n",
              "    }\n",
              "</style>\n",
              "<table border=\"1\" class=\"dataframe\">\n",
              "  <thead>\n",
              "    <tr style=\"text-align: right;\">\n",
              "      <th></th>\n",
              "      <th>Ind_ID</th>\n",
              "      <th>GENDER</th>\n",
              "      <th>Car_Owner</th>\n",
              "      <th>Propert_Owner</th>\n",
              "      <th>CHILDREN</th>\n",
              "      <th>Annual_income</th>\n",
              "      <th>Type_Income</th>\n",
              "      <th>EDUCATION</th>\n",
              "      <th>Marital_status</th>\n",
              "      <th>Housing_type</th>\n",
              "      <th>Mobile_phone</th>\n",
              "      <th>Work_Phone</th>\n",
              "      <th>Phone</th>\n",
              "      <th>EMAIL_ID</th>\n",
              "      <th>Type_Occupation</th>\n",
              "      <th>Family_Members</th>\n",
              "      <th>label</th>\n",
              "      <th>Employed_years</th>\n",
              "      <th>Age</th>\n",
              "    </tr>\n",
              "  </thead>\n",
              "  <tbody>\n",
              "    <tr>\n",
              "      <th>0</th>\n",
              "      <td>5008827</td>\n",
              "      <td>M</td>\n",
              "      <td>Y</td>\n",
              "      <td>Y</td>\n",
              "      <td>0</td>\n",
"      <td>180000.0</td>\n",
"      <td>Pensioner</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>Unemployed</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>0.00</td>\n",
"      <td>52.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>1</th>\n",
"      <td>5009744</td>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>N</td>\n",
"      <td>0</td>\n",
"      <td>315000.0</td>\n",
"      <td>Commercial associate</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>Unknown</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>1.63</td>\n",
"      <td>38.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>2</th>\n",
"      <td>5009746</td>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>N</td>\n",
"      <td>0</td>\n",
"      <td>315000.0</td>\n",
"      <td>Commercial associate</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>Unknown</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>1.63</td>\n",
"      <td>44.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>3</th>\n",
"      <td>5009749</td>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>N</td>\n",
"      <td>0</td>\n",
"      <td>166500.0</td>\n",
"      <td>Commercial associate</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>Unknown</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>1.63</td>\n",
"      <td>38.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>4</th>\n",
"      <td>5009752</td>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>N</td>\n",
"      <td>0</td>\n",
"      <td>315000.0</td>\n",
"      <td>Commercial associate</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>Unknown</td>\n",
              ″      <td>2</td>\n″,
              ″      <td>1</td>\n″,
              ″      <td>1.63</td>\n″,
              ″      <td>38.0</td>\n″,
              ″    </tr>\n″,
              ″  </tbody>\n″,
              ″</table>\n″,
              ″</div>\n″,
              ″    <div class=\″colab-df-buttons\″>\n″,
              ″\n″,
              ″  <div class=\″colab-df-container\″>\n″,
              ″    <button class=\″colab-df-convert\″
onclick=\″convertToInteractive(’df-f12d71b5-616a-4c24-9547-c07be1401b77’)\″\n″,
              ″            title=\″Convert this dataframe to an interactive
table.\″\n″,
              ″            style=\″display:none;\″>\n″,
              ″\n″,
              ″  <svg xmlns=\″http://www.w3.org/2000/svg\″ height=\″24px\″ viewBox=\″0
-960 960 960\″>\n″,
              ″    <path d=\″M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220
220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-
160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\″/>\n″,
              ″  </svg>\n″,
              ″    </button>\n″,
              ″\n″,
              ″  <style>\n″,
              ″    .colab-df-container {\n″,
              ″      display:flex;\n″,
              ″      gap: 12px;\n″,
              ″    }\n″,
              ″\n″,
              ″    .colab-df-convert {\n″,
              ″      background-color: #E8F0FE;\n″,
              ″      border: none;\n″,
              ″      border-radius: 50%;\n″,
              ″      cursor: pointer;\n″,
              ″      display: none;\n″,
              ″      fill: #1967D2;\n″,
              ″      height: 32px;\n″,
              ″      padding: 0 0 0 0;\n″,
              ″      width: 32px;\n″,
              ″    }\n″,
              ″\n″,
              ″    .colab-df-convert:hover {\n″,
              ″      background-color: #E2EBFA;\n″,
              ″      box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n″,
              ″      fill: #174EA6;\n″,
              ″    }\n″,
              "\n",
              "    .colab-df-buttons div {\n",
              "      margin-bottom: 4px;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert {\n",
              "      background-color: #3B4455;\n",
              "      fill: #D2E3FC;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert:hover {\n",
              "      background-color: #434B5C;\n",
              "      box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",
              "      filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",
              "      fill: #FFFFFF;\n",
              "    }\n",
              "  </style>\n",
              "\n",
              "    <script>\n",
              "      const buttonEl =\n",
              "        document.querySelector('#df-f12d71b5-616a-4c24-9547-
c07be1401b77 button.colab-df-convert');\n",
              "      buttonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "\n",
              "      async function convertToInteractive(key) {\n",
              "        const element = document.querySelector('#df-f12d71b5-616a-4c24-
9547-c07be1401b77');\n",
              "        const dataTable =\n",
              "          await
google.colab.kernel.invokeFunction('convertToInteractive',\n",
              "                                                    [key], {});\n",
              "        if (!dataTable) return;\n",
              "\n",
              "        const docLinkHtml = 'Like what you see? Visit the ' +\n",
              "          '<a target=\"_blank\"
href=https://colab.research.google.com/notebooks/data_table.ipynb>data table
notebook</a>'\n",
              "          + ' to learn more about interactive tables.';\n",
              "        element.innerHTML = '';\n",
              "        dataTable['output_type'] = 'display_data';\n",
              "        await google.colab.output.renderOutput(dataTable, element);\n",
              "        const docLink = document.createElement('div');\n",
              "        docLink.innerHTML = docLinkHtml;\n",
              "        element.appendChild(docLink);\n",
              "      }\n",
              "    </script>\n",
              "  </div>\n",
              "\n",
              "\n",
              "<div id=\"df-0a74083e-fb99-45c6-8323-34751a8f0d1c\">\n",
              "  <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-
0a74083e-fb99-45c6-8323-34751a8f0d1c')\"\n",
              "            title=\"Suggest charts\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "<svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0 0
24 24\"\n",
              "     width=\"24px\">\n",
              "    <g>\n",
              "        <path d=\"M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-
.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\"/>\n",
              "    </g>\n",
              "</svg>\n",
              "  </button>\n",
              "\n",
              "<style>\n",
              "  .colab-df-quickchart {\n",
              "      --bg-color: #E8F0FE;\n",
              "      --fill-color: #1967D2;\n",
              "      --hover-bg-color: #E2EBFA;\n",
              "      --hover-fill-color: #174EA6;\n",
              "      --disabled-fill-color: #AAA;\n",
              "      --disabled-bg-color: #DDD;\n",
              "  }\n",
              "\n",
              "  [theme=dark] .colab-df-quickchart {\n",
              "      --bg-color: #3B4455;\n",
              "      --fill-color: #D2E3FC;\n",
              "      --hover-bg-color: #434B5C;\n",
              "      --hover-fill-color: #FFFFFF;\n",
              "      --disabled-bg-color: #3B4455;\n",
              "      --disabled-fill-color: #666;\n",
              "  }\n",
              "\n",
              "  .colab-df-quickchart {\n",
              "    background-color: var(--bg-color);\n",
              "    border: none;\n",
              "    border-radius: 50%;\n",
              "    cursor: pointer;\n",
              "    display: none;\n",
              "    fill: var(--fill-color);\n",
              "    height: 32px;\n",
              "    padding: 0;\n",
              "    width: 32px;\n",
              "  }\n",
              "\n",
              "  .colab-df-quickchart:hover {\n",
       "    background-color: var(--hover-bg-color);\n",
       "    box-shadow: 0 1px 2px rgba(60, 64, 67, 0.3), 0 1px 3px 1px rgba(60, 
64, 67, 0.15);\n",
       "    fill: var(--button-hover-fill-color);\n",
       "  }\n",
       "\n",
       "  .colab-df-quickchart-complete:disabled,\n",
       "  .colab-df-quickchart-complete:disabled:hover {\n",
       "    background-color: var(--disabled-bg-color);\n",
       "    fill: var(--disabled-fill-color);\n",
       "    box-shadow: none;\n",
       "  }\n",
       "\n",
       "  .colab-df-spinner {\n",
       "    border: 2px solid var(--fill-color);\n",
       "    border-color: transparent;\n",
       "    border-bottom-color: var(--fill-color);\n",
       "    animation:\n",
       "      spin 1s steps(1) infinite;\n",
       "  }\n",
       "\n",
       "  @keyframes spin {\n",
       "    0% {\n",
       "      border-color: transparent;\n",
       "      border-bottom-color: var(--fill-color);\n",
       "      border-left-color: var(--fill-color);\n",
       "    }\n",
       "    20% {\n",
       "      border-color: transparent;\n",
       "      border-left-color: var(--fill-color);\n",
       "      border-top-color: var(--fill-color);\n",
       "    }\n",
       "    30% {\n",
       "      border-color: transparent;\n",
       "      border-left-color: var(--fill-color);\n",
       "      border-top-color: var(--fill-color);\n",
       "      border-right-color: var(--fill-color);\n",
       "    }\n",
       "    40% {\n",
       "      border-color: transparent;\n",
       "      border-right-color: var(--fill-color);\n",
       "      border-top-color: var(--fill-color);\n",
       "    }\n",
       "    60% {\n",
       "      border-color: transparent;\n",
       "      border-right-color: var(--fill-color);\n",
       "    }\n",
       "    80% {\n",
       "      border-color: transparent;\n",
              "      border-right-color: var(--fill-color);\n",
              "      border-bottom-color: var(--fill-color);\n",
              "    }\n",
              "    90% {\n",
              "      border-color: transparent;\n",
              "      border-bottom-color: var(--fill-color);\n",
              "    }\n",
              "  }\n",
              "</style>\n",
              "\n",
              "  <script>\n",
              "    async function quickchart(key) {\n",
              "      const quickchartButtonEl =\n",
              "        document.querySelector('#' + key + ' button');\n",
              "      quickchartButtonEl.disabled = true;  // To prevent multiple
clicks.\n",
              "      quickchartButtonEl.classList.add('colab-df-spinner');\n",
              "      try {\n",
              "        const charts = await google.colab.kernel.invokeFunction(\n",
              "            'suggestCharts', [key], {});\n",
              "      } catch (error) {\n",
              "        console.error('Error during call to suggestCharts:', error);\n",
              "      }\n",
              "      quickchartButtonEl.classList.remove('colab-df-spinner');\n",
              "      quickchartButtonEl.classList.add('colab-df-quickchart-
complete');\n",
              "    }\n",
              "    (() => {\n",
              "      let quickchartButtonEl =\n",
              "        document.querySelector('#df-0a74083e-fb99-45c6-8323-
34751a8f0d1c button');\n",
              "      quickchartButtonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "    })();\n",
              "  </script>\n",
              "</div>\n",
              "\n",
              "    </div>\n",
              "  </div>\n"
            ]
          },
          "metadata": {},
          "execution_count": 1294
        }
      ],
      "source": [
        "merged_data.head()"
      ]
    },
{
  "cell_type": "code",
  "source": [
    "df_sql = merged_data.copy()"
  ],
  "metadata": {
    "id": "u3Mu14EzTK73"
  },
  "execution_count": null,
  "outputs": []
},
{
  "cell_type": "code",
  "execution_count": null,
  "metadata": {
    "id": "SdzfG236TKmN",
    "colab": {
      "base_uri": "https://localhost:8080/"
    },
    "outputId": "dbb203bf-d9e9-4ff5-ce42-57c433a0dee1"
  },
  "outputs": [
    {
      "output_type": "execute_result",
      "data": {
        "text/plain": [
          "array([1])"
        ]
      },
      "metadata": {},
      "execution_count": 1296
    }
  ],
  "source": [
    "merged_data[\"Mobile_phone\"].unique()"
  ]
},
{
  "cell_type": "markdown",
  "metadata": {
    "id": "ZhHCJtBBTdRK"
  },
  "source": [
    "merged data only has one value hence can be dropped"
  ]
},
{
  "cell_type": "code",
  "execution_count": null,
      "metadata": {
        "id": "YkxVxVYbRhVx"
      },
      "outputs": [],
      "source": [
        "merged_data = merged_data.drop(\"Mobile_phone\",axis = 1)"
      ]
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "ZfcoF_jI3l_R"
      },
      "source": [
        "Ind_ID can also be dropped as it is an identifier and does not influence
credit card approval"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "SAkMNKKF3lqL"
      },
      "outputs": [],
      "source": [
        "merged_data = merged_data.drop(\"Ind_ID\",axis = 1)"
      ]
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "sdKWn8YHJrUa"
      },
      "source": [
        "## EDA"
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "# helper function to analyse categorical columns\n",
        "\n",
        "def analyse_categorical_column(column_name:str,rotation=0,fontsize=10):\n",
        "  data =
merged_data.groupby([column_name,\"label\"]).size().unstack().rename(columns =
{0:\"approved\",1:\"not approved\"}).sort_values(by = \"approved\",ascending =
False).reset_index()\n",
        "  data['total'] = data['approved'] + data['not approved']\n",
        "  data['approval_percentage'] = (data['approved'] / data['total']) * 100\n",
        "  fig,ax = plt.subplots(1,2,figsize = (14,5))\n",
        "  bar1 = ax[0].bar(data[column_name],data[\"approved\"],color='lightblue')\n",
        "  bar2 = ax[0].bar(data[column_name],data[\"not approved\"],bottom =
data[\"approved\"],color='lightgray')\n",
        "  ax[0].tick_params(axis='x', rotation=rotation)\n",
        "  ax[0].legend([\"Approved\", \"Not approved\"],loc = \"upper right\")\n",
        "  for i in range(len(data)):\n",
        "      ax[0].text(i, data['approved'].iloc[i]/2,
str(round(data['approval_percentage'].iloc[i], 2))+'%', ha = 'center', color =
'black')\n",
        "  data = merged_data[column_name].value_counts()\n",
        "  ax[1].pie(data, labels=data.index, autopct=\"%1.2f%%\",
shadow=True,textprops={'fontsize': fontsize})\n",
        "  plt.tight_layout()\n",
        "  plt.show()"
      ],
      "metadata": {
        "id": "lxox_yME23FR"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "# helper function to analyse numerical columns\n",
        "\n",
        "def analyse_numerical_column(column_name:str):\n",
        "  # Create a figure and a 2x1 subplot structure\n",
        "  fig, ax = plt.subplots(2, 1, figsize=(13, 6))\n",
        "\n",
        "  # On the first subplot, plot a histogram of the column data\n",
        "  sns.histplot(x=column_name, data=merged_data, ax=ax[0])\n",
        "\n",
        "  # On the second subplot, plot a boxplot of the column data\n",
        "  sns.boxplot(x=column_name, data=merged_data, ax=ax[1])\n",
        "\n",
        "  # Display the plots\n",
        "  plt.show()\n"
      ],
      "metadata": {
        "id": "pLREp9kmGv2a"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "markdown",
      "source": [
        "### Numerical Columns Analysis"
      ],
      "metadata": {
        "id": "e9oi74c3Gm7U"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "#### Annual_income"
      ],
      "metadata": {
        "id": "6CXk4Nc3GmQc"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "analyse_numerical_column(\"Annual_income\")"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 542
        },
        "id": "_hQ2KqbZHAjR",
        "outputId": "77f115be-92d7-420c-b051-afffe6b23dc9"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 1300x600 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAABDsAAAINCAYAAAAwQHSoAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAABZv0l
EQVR4nO3deVyU9f7//yfMAFoCRiaLIm65ZaZGi39PW45Zai39PhY+VWpC1Wx8xvHdIWUtsOrbQys
4xcymiRNLOjlmsagpUi4ALMzPv7Rz/m4yQgMwID14/77TY3net6z/V+XdebYeZ6ciOBxhgjAAAAAAAiwjOdwE
AAAAAAAAVibADAAAAAABYCmEHAAAAAACwFMIOAAAAAABgKYQdAAAAAADAUgg7AAAAAACApRB2AAAAAAAASyHsA
AAAAAAAlmL3dwHVgcvlUmZmpkJDQxUQEODvcgAAAAAAwF8YY5bl6eTMzEzFxsb6uwwAAAA
AAHAWBw4cUMOGDctsQ9ghKTQ0VNLpNywsLMzP1QAAAAAAgL/Kzc1VbGys9Tu8LIQdkuepK2FhYYQdAAAAAABcxMozBQYTlAIAAAAAAEsh7AAAAAAAAJZC2AEAAAAAACyFsAMAAAAAAFgKYQcAAAAAALAUwg4AAAAAAGAphB0AAAAAAMBSCDsAAAAAAIClEHYAAAAAAABLIewAAAAAAACWQtgBAAAAAAAshbADAAAAAABYCmEHAAAAAACwFMIOAAAAAABgKYQdAAAAAADAUgg7AAAAAACApRB2AAAAAAAASyHsA
AAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAlmL3dwHVgcvlUmZmpkJDQxUQEODvcgAAAAAwF8YY5bl6eTMzEzFxsb6uwwAAAA
AAHAWBw4cUMOGDctsQ9ghKTQ0VNKfGywsLMzP1QAAAAAAgL/Kzc1VbGysex++LIQdkvvUlbCwMMIOAAAAAACqs
fJcfoILlAIAAAAAAEsh7AAAAAAAAJZC2AEAAAAAACyFsAMAAAAAAFgKYQcAAAAAALAUwg4AAAAAAGAp3HoW1Yr
D4ZDT6SxXW5vNJ2FFFtOBwONYyNU3ZWZrnaR0bF6OCB/QQeAAAAAAAAP7CWi2nA6ncrOytSNc
9co0B5UZluXo0jv3ttLTqeTsAMAAAAAAAA4IG9RFQ7gfYg2YKC/V0GsusFGAAAAAAKCG8usFSmfNmqW
48GDt3LnTo82pUeUmJioiy6oiy++WHXq1FGwQIUOUnZ3t0eaoi0eaXX37tRgUmdYJVA0ANMmMlAAAAAABAN
```

```
eHXsGP9+vVKTEzUV199pTVr1qioqEi9e/fW8ePH3W3+9a9/6YMPPtCyZcu0fv16ZWZm6sYbb3TPdzqdGjBggAo
LC/Xll19q0aJFWrhwoR566CF/rBIAAAAAAPCzAGOM8XcRxX777TfVr19f69ev1zXXXKNjx47pkksu0Ztvvvqmbb
rpJkrRjxw61bt1a6enp+vvf/66PPvpIAwcOVGZmpiIjIyVJ8+fP1wMPPKDffvtNwcFnPx0iNzdX4eHhOnbsmML
Cwip1HVG6goIC1apVSzf9Z91ZT2NxFhXq7Qk9d0rUKYWEhFRRhQAAAAAf/Fm392vR3b81bFjxyRJERERkqSMj
AwVFRUpISHB3aZVq1Zq1KiR0tPTJUnp6elq27at0+iQpD59+ig3N1c//vhjif0UFBQoNzfX4wEAAAAAKyh2oQ
dLpdL9957r7p27arLL79ckpSVlaXg4GDVrVvXo21kZKSysrLcbU4POornF88ryaxZsxQeHu5+xMTESJK6euaa
AAAf6k2YUdiYqK2bdumJUuWVHpfU6ZMUqPDPfwlIb1nC5SLW890mDBBK1eu1IYNG9SwYUP39Ki
oKBUWFionJ8fj6I7s7GxFRUW523zzzTcerYunhf88NwcFnPx0iNzdX4eHhOnbsmML
dLEY/6VV16poKAgpaWluaft3LlTv/zyi+Lj4yVJ8fHx+uGHH3T48GF3mzVr1igsLEyXXXZZ1awIAAAAAACoNvx
6ZEdiYqLefPNNvffeewoNDXVfYyM8PFy1a9dWeHi4Ro8eraSkJEVERCgsLEwTJ05UfHy8/v73v0uSevfurcsuu
OwjRozQUO89paysLE2fPl2JiYkcvQEAAAAAwHnIr2HHvHnzJEk9evTwmL5gwQLdfvvtkqRnnnlGgYGBGjJkiAo
KCtSnTx89//zz7rrY2mOHrnHHhjFB8frwsvvvFCjRo3SzJkzq2o1AAAAAABANRJgjDH+LsLfvLLXypAo
tWqpZv+s062oAy2zqLCvX2hB46deoUR/AAAAAAwHnAm333anM3FgAAAAAAgIpA2AEAAAAACyFsAMAAAAAAFg
KYQcAAAAAALAUwg4AAAAAAGAphBOAAAAAAMBSCDsAAAAAAIClEHYAAAAAAAABLsfu7AJwfHA6HnE5nmW0OKCgqqq
BoAAAAAgJURdqDSORwONYyNU3ZWnaG2MquSIAAAAAgJURdqDSOZ1OZWdl6sa5axsBoAAAAAgFVxgVIAAAAAAGAphBOAAAAAAMBSCDsAAAAAAIClEHYAAAAAAAA
ABLIewAAAAAAACWQtgBAAAAAAAshbADAAAAAABYCmEHAAAAAACwFMIOAAAAAAABgKYQdAAAAAADAUgg7AAAAAAC
ApRB2AAAAAAAASyHsAAAAAAAAlkLYAQAAAAAALIWwAwAAAAAAWAphBwAAAAAAsBTCDgAAAAAAYCmEHQAAAAAAw
FIIOwAAAAAAgKUQdgAAAAAAAEsh7AAAAAAAAJZC2AEAAAAAACyFsAMAAAAAAFgKYQcAAAAAALAUwg4AAAAAAGA
pfg07NmzYoEGDBikmJkYBAQFasWKFx/zbb79dAQEBHo++fft6tDly5Ihuu+O2hYWFqW7duho9erTy8/OrcC3gT
wUFBWd9OBwOf5cJAAAAAKhCfg07jh8/rvbt2ysljJaXUNNn379tWhQ4fcj8WLF3vMv+222/Tjjz9qzZo1WrlypTZ
s2KCxY8dWdunwM5fTIQUGKTQ09KyPqNiGh00Nn7CQnNnhyHqWLgESF8vb8Rn
rW+/fZbXXXVVZKk5557Tv3799fs2bMVFExNT4TWjejAul+RyavCcT2QPDi61nctRpHfv7SWn0ym73a8/7gAAAAC
AKlLtr9mxbt061q9fXy1bttT48eP1xx9/uHPS09NVt267kqqWBDkhISEhQYYGKivv/61GUGEBBQoNzfX44GaKdAeJ
FtQcKmPQHuQv0sEAAAAAFSxah12903bV6+++Xr5r+s7+cTqckSSrS/Xr1/d4ehLK
yskpd7qxZsxQeHu5+xMbGVup6x6AAAAACAqlOtj+u/+eab3f9PjuI9J9eS+UVPlN3f37s6+j3jdBTTPSqOFhOXoO+CGtUKjtX
D+vM6ICEhIZVdLmoYh8OOhhrFxys7KPGvbyKgYHTywn8DAAAAAKqhGrWndvQvDgDQf3xxxx+KjjNcJ6OsnjLnYK
Rm68sorJUmfffaZXC6XqxxrlrlryrzWBxc9BQAAAIDqza97vnff7e+8dc/nnn+dq1a5f7d+d69e7Vl1yxZ
FREQoIiJCM2bMJAhQxQVFaXdu3dW3bXbl1/dddddmj9/voqKijRhwgTdfPPNio6OrqSlAgAAAAAAAQAAAAAAAAAA
iu+6CkAAAAAoGby6wVKN23apI46aDpa46AK6SkSkJHXs2fJMWhjWbtAXM7m6ZKWkP2uAGNkMWvWjnR
3OAXljTfeUKddepf//+uvrq/6a5UAAAAAICf+fXIjh49YxdhgAYU+r8jz/+AAAAAA/MwB2Og3Dfd
QEAAAAgBqsWt96FgAAAAAwFuEHQAAAAAAwFIIOwAAAAAAgKUQdgAAAAAAAEsh7AAAAAAAJZC2AEAAAAAACz
Fr7eeBc4HDodDTqfzrO1sNpvsdtIAu2sAhAXEHAA0EAAAAA54i9KqASO
Z1OZWdl6sa5axRoDyq1nctRpHfv7SWn0OnYAQAAAADniL0qoAoE2oNkCwr2dxkAAAAAcF7gAqUAAAAAAAMBSCDs
AAAAAAIClEHYAAAAAAAABLIewAAAAAAACWQtgBAAAAAAAsxaewo2nTpvvrjjz/0mJ6Tk60ZuecclEAAAAAAC+8
ins2Ldvn5x05xnTCwoK90uvvv55zUQAAAAAAL6ye9P4/fffd///448/Vnh4uP/Vnh4uPu50+1UWlqaGjduXGHFAQAAAAA
AeMursGPPw4MGSpICAAA0aNcpjnt1uV+PGjfXUU09VWHEAAAAAAAvlkiQ1ad3377rerVq1cpRQEAA
AAAAPjKq7Cj2N69eyu6DgAAAAAAgArhU9ghSWvXrtXatWt1+PBhuVwuj3mvvPLKORcGAAAAAADgC5/CjhkzZmjm
zJnq2LGjGqjAAAAAAAAAAE0cQkvG0/5EqC52N2ngLaAhkMJM2IHiAhJfqmtN/dzoBkSJB/uAAAAAAAAAAAAAAgZAgd2Jeoepzd
YMWPGDM2coVNXXXWVoqFRAQUNF1AdWWy+mQAm0OeF+g9G2NMJVYEAAAAADidT2HH/PnzlA4Ki46Gcnm
eNySS6nBs/5RPbg4DLbFp06GaXPg6JsBRmLJSFRsiq4TL6Rt74zmwYqLq2BLiZ2BkaSwrTtKqoGAAAAAFA
s0JcXjRkzRm++b5ezgYSkzoTGsQCvkYmWBU3Lz1FpyfWwYFqL0
25Vhw4dJEnbtm3zmtfT7b0zB3yzgNSfEpA
znwsCAAAAAA4Fz6FHcXX6yhWRkVu2aN2bRo1alPTZ555pT8fs/28PzGYT8VRpTpDz/8sPLz88++pIAAAA
AAAgHNRodfs+Ic/vJ5d9M4sFSzrVs3lz/JDhg5XCb9S29jEXwu0RpeI4Aq8
TJj344IMVUhiqP4fDqQfD4s+fQNzfDITeKDfBIsaRPcsMxmbWu
LKzMsv9GmNMJVYEAAAAAMCffAo7FixYUCGb9iwQfB+97+VkZGhQ4c0afny5ReJl8eLB7vjFGycnJeumll5STk6O
uXbtq3rx5uvTSS91tjw5ookTJ+qDDz5QqDDzv5QYGCghgwZv7f+/6vStSpUyE1omROp1PZWZm6ce4aBdqDymxbdOq43
ps0QGQdAAAAAIcqcE7X7MjIyNDrr7+u/l19/XZs3b/b69PrLHf1f79u2VkpJS4vynnnpKqqzm77rObPn6vv/5aF15
```

```
4ofr06aNTp06529x222368ccftWbNGq1cuVIbNmzQ2LFjfV4neCfQHiRbUHCZj0Bb2WEIAAAAAAAVyacjOw4fP
qybb75Z69atU926dSVJOTk5uvbaa7VkyRJdcskl5VpOv3791K9fvxLnGWM0d+5cTZ8+XTfccIMk6dVXX1VkZKR
WrFihm2++Wdu3b9fq1av17bff6qqrrpIkPffcc+rfv79mz56tmJgYX1YPAAAAAADUYD4d2TFx4kTl5eXpxx9/1
JEjR3TkyBFt27ZNubm5+j//5/9USGF79USGF79+5VVlaWEhIS3NPCw8PVuXNnpaenS/rz7i9169Z1Bx2SlJCQoMDAQH3
99delLrugoEC5ubkeDwAAAAAAYAO+hR2rV6/W888r9atW7unXXbZZUpJSdFHH31UIYVlZWVJkiIjIz2mROZGu
udlZWWpfv36HvPtdrsiIiLcbUoya9YshYz8xsbEVUjMAAAAAAPA/n8IOl8uloKAzr8MQFBQkl8t1lzkVVtil
TpujYsWPux4EDB/xdEgAAAAAAqCA+hRO9e/bUP//bUP//5T2Vm/u9tR3/99Vf961//0nXXXVchhUVFRUmSsrOzPaZnZ
2e750VFRenw4cMe8xOh44cOeJuU5KQkBCFhYV5PAAAAAAAgDX4FHb85z//UW5urho3bqxmzZqpWbNkNmatKkiXJ
zc/Xcc89VSGFNmjRRVFSUOtLS3NNyc3P19ddfKz4+XpIUHx+vnJwcZWRkuNt89tlncrlc6ty5c4XUAQAAAAAAa
haf7sYSGxur7777Tp9++ql27NghSWrdurXHxUTLI7ZZs+m1atRI9957rx599rx599FFdeum
latKkiR588EHFxMRo8ODB7j779u2ru+66S/Pnz1dRUZEmTJigm2++mTuxAAAAAABwnvIq7Pjss88880YcIEffXVV
woLC10vXr3Uq1cvSdKxY8fUpkObzZ8/X926dSvX8jZt2qRrr73W/TwpKUmSNNGrUKC1cuFD333+/jh8/rrFjxyo
nJ0dXX321Vq9erVq1arlf88Ybb2jChAm67rrrFBgYqCFDhujZZ5/1ZrUAAAAAAICFeBV2zJO7V3fddVeJ17gID
w/X3XffrTlz5pQ77OjRo4eMMaXOt9lsmvHHPWJsM8Zu+ve/oj2Zq+yGmVHv6B+RDV3Zim8KnV5oVg6KNrrqVBfjxjucCV4T
ZUpK0Y8cOpaSkyOl0atq0aZVSKAAAAAAAQHl4FXZERkbqyy+/1Lhx4zRlyVQCAgLUp08fpaSkDIyslIKB
QAAAAAAKA+vwg5JiouL06pVq3T06FHt2rVLxhhdeumlluuiiiyqjPgAAAAAAK94HXYUu+iii9RLK94VdK
AAAA4Z15doBQAAAAAAKC6I+wAAAAAAACWQtgBAAAAAAAshbADAAAAAABYCmEHAAAAAACwFMIOAAAAAABgKYQdA
AAAAADAUgg7AAAAAACApRB2AAAAAAAASyHsAAAAAAAAlkLYAQAAAAAALIWwAwAAAAAAWAphBwAAAAAAsBS7vws
AUHkcDoecTme52tpsNtnt/EoAAAAAUPOxZwNYlMPGVnZOGVnAZarfWRUjA4e2E/gAQAAAKDGY68GsCin06nsr
EzdOHeNAu1BZbZ1OotkrO7r295HQ6CTsAAAAA1Hjs1QAWF2gPks0e7O8yAAAAAKKVahxOPDx7ULbfcojAAAABVij0QAOXmdQVnZWpGki2gPki0o2N9lAAAAAECV4QKlAAAAAADAUgg7AAAAAACpRB2AAAAAAAASyHsAAAAAAAAlkLYAQAAAAAALIWwAwAAAAAAWAphBwAAAAAAsBS7vwsAULM4HGznHbLb+dcCAAAAwBtgu3ewSO7T8d9R1gDU6kEfPFiWc3tZZl7g6-AGhdmW9m2PmeScs5yxWc2xmiXSAHuZ7Up6ILSKYRZmXcngsRCRnuI8ca2AHHK2vvPmYx3G1MS6z
2LWvM6jzL0o8uZ6FvdgEkKOHxbvY4AZLjpaPsJsJaZHmsFXwCy0jSSRmQLQ4RkPsoK4AS5b3VKyAzMNhM2DXKe9Qr9g1yxJAJZIBMPuzZEVHa7khOKIMQH
```

```
GqJn2UAAM6NJT4V33rrLSUlJWn+/Pnq3Lmz5s6dqz59+mjnzp2qX7++v8sDzmtnCxMqI2yo7Fv5lqfmmvCXVm9
2pqTy7VBVxjK94XQ6lZ2VqRvnrlGgPajUdi5Hkd69t5dOnDihkJCQsy63JownrMXbn2Wn08nPKABAEkcGFrPEm
s2ZM0d33XWX7rjjDknS/Pnz9eGHH+qVV17R5MmT/VwdcH7yNnDwJmw467Iq6Va+3qxT/cho7dn981k/QLwNe8r
bvjwfXuXdmZLKv0NVGcv0xdnub0Ttz2d5x7MyvjT480iZ6qC862+MUUBAQLmWWZO2U3nu0gUAQDGODPxfNX6tC
gsLlZGRoSlTprinBQYGKiEhQenp6SW+pqCgwGOH4dixY5Kk3Nzcyi22AhXXX5CfoOBb2TsURQUn3G2dBWfZ8St
nW5bJMsu1TJdT/WYuLTNwKDx1Qh8/fKt05eXIfpYv9N7WWXTquIyzqEKW6W5bjnUqKjil1Q/fqjp16pS5vN0dy
jsqm73OZToKTOkBgeXeOa93SaS2fr+5zA+v4t8jRaeOn/X3iOv/346//fZbmUdCVMYypfLvyJb3d2N5x/LPtuU
fz/Js92LlWSeHw6F2HTrq98PZZ11eZfTvTbvKWKY36x9gs8s4HeWqs7zbyZ/rXt6fZW/eR5VRpxWX6e/+WSbLr
M7L9Hf/LLN8nx/ZWZnq/+gy2cr4w5PTUaRV04fq6NGj5fr8qC6K99nL84fSAF0Rf071g8zMTDVo0EBffvml4uP
j3dPvv/9+rV+/X19//fUZr3n44Yc1Y8aqiwTAAAAAABUgAMHDqhhw4ZltqnxR3b4YsqUKUpKSnI/d7lc0nLki
C6++OJyJ3Deys3NVWxsrA4cOKCwsLBK6QP+wdhaF2NrXYytdTG21sXYWhdja02Mq3X5c2yNMcrLy1NMTMxZ29b
4sKNevXqy2WzKzvY8xDU7O1tRUElviYkJOSMQ3Xq1q1bWSV6CAsL481uUYytdTG21sXYWhdja12MrXUxttbEu
FqXv8a2vKd1B1ZyHZUuODhYV155pdLSOtzTXC6XOtLSPE5rAQAAAAAA54caf2SHJCUlJWnUqFG66qqr1KlTJ82
dO1fHjx93350FAAAAAACcPywRdgwfPlxHjhzx93350FAAAAAACcPywRdgwfPly//fabHnroIWVlZalDhw5avXq1IiMj/V2aWOhIiJKTk2vUlW5RPoytd
TG21sXYWhdja12MrXUxttbEuFpXTRnbGn83FgAAAAAgNPV+Gt2AAAAAAAAnI6wAwAAAAAAWAphBwAAAAAAsBT
CDgAAAAAAYCmEHRUoJSVFjRs3Vq1atdS5c2d98803ZbZftmyZWrVqpVq1aqlt27ZatWpVFVUKb3kzzti+99JK6d
eumiy66SBdddJESEhLO+rMA//H2fVtsyZIlCggI0ODBgyu3QPjM27H9/fffNW7cOEVHRyssLEwtWrTgc7ma8nZ
s
nzp2rli1bqnbt2oqNjdW//vUvVVq1aqlt27ZatWpVFVUKb3kzzti+99JK6d
...
```

Cg4NNp06dzFdffeWe1717dzNq1CiP9kuXLjUtWrQwwcHBpk2bNubDDz+s4opRXt6MbVxcnJF0xiM50bnqC8dZe
fu+PR1hR/Xm7dh++eWXpnPnziYkJMQ0bdrUPPbYY8bhcFRx1SgPb8a2qKjIPPzww6ZZs2amVq1aJjY21owfP94
cPXq06gtHqdauXVviZ2fxWI4aNcp07d1ND6NDBBAcHm6ZNm5oFCxZUed0402/Htnv37mW2R/Xhy/v2dIQd1
ZcvY7t9+3aTkJBgateubRo2bGiSkpLMiRMnqr740wQYwzGcAAAAAADAOrhmBwAAAAAAsBTCDgAAAAAAYCmEHQA
AAAAAwFIIOwAAAAAAgKUQdgAAAAAAAEsh7AAAAAAAAAJZC2AEAAAAAACyFsAMAANRICxcuVN26dcvV9uGHH1aHD
h0qtR4AAM53GzZsOKBBgxQTE60AgACtWLHC62UYYzR79my1aNFCISEhatCggR577DGvl0OPYAQAAzpCeni6bzaY
BAwb4u5QKMWnSJKWlpfm7DAAAL0348eNq3769U1JSfF7GP//cuaPXu2duzYoffff1+dOnXyyejl2nysAA
ACWlZqaqokTJyo1NVZmZmKiYnxdkOnnpE6dOqpTp46/P/Ol3/ywAAwNL69eunfv/36/ywAAwNL69eunfv/36lTq/oKBA06ZN0+LFi5WTk6PLL79
cTz75pHr06OG6c7cOA0YMEAKU1pamq666iotLPy6u3+zNG7cO9bW7AbBt4h9Sfn6pHr06NBBBDz/8sPv
5nDlz1LZtLtLZtW1144YWKjKuR5WTk6PLL79
cTz75pHr06NBBBDz/8sPv
5nDlz1LZtW1144YWKjY3V+PHjlZ+f79M6/fU0lttvv12DBw/W7NmzFR0drYsvvliJiYkqKipytykqKNADDzyg2
NhYhYSEqHnz5kpNTXXPX79+vf/vTp16qSQkBBFR0dr8uTJcjgc7vk9evRQYmKiHnvsMQgKNzwgi8OAJCfSiZIEd7LLDsC5SSjb4iBVmiuo5DsCgNGK5c0Qjx1RCnrSOrvkjwSSiElrS1FICcdZLAnVz+JNbXkJyb2KuUJvNAxHqmRIjtB7
LvKv2fvL1qA9xOEx3FW2xjYy7lVFrGlW/klmWgLYqWNBmDXZr9q0MTbXdAgXn7Wq4j8PPXn8i/DmeLBDQ9hn4fLcbFy9Dmnt2flvsNl3NnCekOJ3JGbDzAaFwEYvTi7/SAHbKDNqSVmD8ybNk5nHdK6wVZOq0ksXk6gOnDgZx+dxP3H9zFcLRr68h3Kes35bEnDLYDadFV+zZk0oL9dLVYsltffSj7gSWNybaVgCyOlFhp5jvVyDVTGPG9CB0uc+QiZkuhBjhbfIhu8rp6eGLRUJ4dIW1PuatdN0G0UUdVtoFjCZ7lWQEnaYWSTsLNJ0cmUWwIm4hB87cj/vSd79cd1yvcFhMFIGLxmfoWaW1Ltl/M7pUqgq0TwVrZwuQ2VEU5EYbU2Z/UUrzO+PD2UxgtpyNr1/ChCTwoojtHf

AMAAAAAAFgKYQcAAAAAALAUwg4AAAAAAGAphBOAAAAAAMBSCDsAAAAAAIClEHYAAAAAAABLIewAAAAAAACWQtg
BAAAAAAAshbADAAAAAAABYCmEHAAAAAAACwFMIOAAAAAABgKYQdAAAAAADAUgg7AAAAAACApRB2AAAAAAAASyHsA
AAAAAAAlkLYAQAAAAAALIWwAwAAAAAAWAphBwAAAAAAsBTCDgAAAAAAYCmEHQAAAAAAwFIIO2qoU6dO6aefftK
pU6f8XQrgN7wPAAAAAJSEsKOG+uWXXzR27Fj98ssv/i4F8BveBwAAAABKQtgBAAAAAAAshbADAAAAAAABYCmEHA
AAAAACwFMIOAAAAAABgKYQdAAAAAADAUgg7AAAAAACApRB2AAAAAAAASyHsAAAAAAAAlkLYAQAAAAAALIWwAwA
AAAAAWAphBwAAAAAAsBTCDgAAAAAAYCmEHQAAAAAAwFIIOwAAAAAAgKUQdgAAAAAAAEsh7AAAAAAAAJZC2AEAA
AAAACyFsAMAAAAAAAFgKYQcAAAAAALAUwg4AAAAAAGAphBOAAAAAAMBSCDsAAAAAAIClEHYAAAAAAABLIewAAAA
AAACWQtgBAAAAAAAshbADAAAAAAABYCmEHAAAAAAACwFMIOAAAAAABgKYQdAAAAAADAUgg7AAAAAACApRB2AAAAA
AAASyHsAAAAAAAAlkLYAQAAAAAALIWwAwAAAAAAWAphBwAAAAAAsBTCDgAAAAAAYCl2fxcAAFXJ6XRq69atOnz
4sLZv3y5Jqlevnv773//qOKFDiomJOZQpU1SnTh33azIzMzV27FidPHlStWvX1osvvvqiYmJgzlnnkyBHVrVtXk
nTkyBHl5OSobt26qlevntq1ayebzVZiTYWFhXrvvvfeUmZmpvLw8ffrpp+55Dz3OkHr27FmudTpy5IgiIiLK7Ot
OJO+e1AsvvKCDBw+qYcOGuvvvuu1W7du2zvs5bVdVPaXOFBweXe/ucPhYxMTG64YYbFBwcXGJbX7e7L6qyr+rM1
+1gxe2Xn5+vWbNmuX9W//p7CwCA0njzfacmI+wAcN7YsGGDnn/evRo4cKBatWql+fPnqlevXio
qKnLPz8/P16233qqgoCCtWbOmXMuUpKioKIOfP17XXH0Nx/T58+dr2bJlcjqdJb5u5u5syZmjlzptatW1fudSqtr
9NNmzZNGzdudD/ftGmTVqxYoa5du+qxxx4rc128N2N3N09GYv58+dr6NChuueeeeezza+rrdfVG
VfVVnvm4HK26/e+65Rzt27HA//+vLQAASuPN952ajtNYAJwXAKw0aoR48e7qAjJJ7cU6ZMc+TNTk9Pd+9dn99+d+ZMb61TmkvSQHaWw9pa6a5g5WU4aRvzKYGQaInSbMzw8GeRbNnmMUZbgAoFo3N+FIVHw8HEYAOH7zaBz
nNAMrQYYd80k1wlIRnwXxNfxJDEVYCz6
0Jj0+e1AsvvKCDBw+qYcOGuvvvuu1W7du2zvs5bVdVPaXOFBweXe/ucPhYxMTG64YYbFBwcXGJbX7e7L6qyr+rM1
+1gxe2Xn5+vWbNmuX9W//p7CwCAOnjzfacmI+wAcN7YsGGDnn/+eWVlZZXaZu/evRo4cKBatWql+fPnqlevXio
qKnLPz8/P16233qqgoCCtWbOmXMuUpKioKIOfP17XXHONx/T58+dr2bJlcjqdJb5u5u5syZmjlzptatW1fudSqtr
9NNmzZNGzdudD/ftGmTVqxYoa5du+qxxx4rc128UVX9lNVXSEiICgoK3NO9GYv58+dr6NChuueeeeezza+rrdfVG
VfVVnvm4HK26/e+65Rzt27HA//+vvLQAASuPN952ajtNYAJwXNmzYoOTkZAUG/vlrr27duoqIHDPL/4rb+PGj
RUQEKAdO3aoR48e7qAjJIiJCU6ZMcb+mqKhI1113nZKTkkzlxtNMjX7Rq5sEtoR46qFKnKp8lv1MoR4Fkf0jLXYevt5RQd5AQsvDs3GWXZKoN60vG3AGjn7v+FWAJWWfyGMRpVOOhA0Mg==

```
5+z7YsGGDnn/+eWVlZZ21batWrTR//nz16tXLffvZ0wUFBWnNmjXlXmZ0dLTGjRuna665xmN6Sfc6L8m6detKn
F5S/6X1dbriW7X+VdeuXSv0dLKq6qesvkJCQjwuSu3NWNhsthLv0+/rdvdFVfZVnfm6Hay4/YpvP/tXxb+3AAA
ojTffd6ojb/bdCTtE2FFdEHZYYxyrki/vA6fTqa1bt+rw4cPavn27JKlevXr673//q0OHDikmJkZTpkzx+MtoZ
mamxo4dq5MnT6p27dp68cUXPc5nLF7mkSNHVLduXUl/nuqSk50junXrql69emrXrl2pf0EuLCzUe++9p8zMT0X
l5bkPRZc8bzd7tnUqPt2mrL50d/LkSb3wwgs6ePCgGjZsqLvvvrtCjrTwVz+l9RUcHFzu7XP6WMTExOiGG24o9
S8cvm53X1RlX9WZr9vBitsvPz9fs2bNcv+s/vX3FgAApfHm+051Q9jhJcK06oGwwxrjWJWs+D4AAAAAUDJv9t2
5ZgcAAAAAALAUwg4AAAAAAAGAphBOAAAAAAMBSCDsAAAAAAIClEHYAAAAAAAABLIewAAAAAAACWQtgBAAAAAAAsh
bADAAAAAABYCmEHAAAAAACwFMIOAAAAAABgKYQdAAAAAADAUgg7AAAAAACApRB2AAAAAAAASyHsAAAAAAAAlkL
YAQAAAAAALIWwAwAAAAAAWAphBwAAAAAAsBTCDgAAAAAAYCmEHQAAAAAAwFIIOwAAAAAAgKUQdgAAAAAAAEsh7
AAAAAAAAJZC2AEAAAAAACyFsAMAAAAAAFgKYQcAAAAAALAUwg4AAAAAAGAphBOAAAAAAMBSCDsAAAAAAIClEHY
AAAAAAABLIewAAAAAAACWQtgBAAAAAAAshbADAAAAAABYCmEHAAAAAACwFMIOAAAAAABgKYQdAAAAAADAUgg7A
AAAAACApRB21FCNGjXSiy++qEaNGvm7FMBveB8AAAAAKInd3wXAN7V7VQbcsPRDWn5AAAAAAJSEIzsAAAAAAIClEHYAAAAAAABLIewAAAAAAACWQtgBAAAAAAAsh
bADAAAAAABYCmEHAAAAAACwFMIOAAAAAABgKYQdAAAAAADAUgg7AAAAAACApRB2AAAAAAAASyHsAAAAAAAAlkL
YAQAAAAAALIWwAwAAAAAAWAphBwAAAAAAsBTCDgAAAAAAYCmEHQAAAAAAwFIIOwAAAAAAgKUQdgAAAAAAAEsh7
AAAAAAAAJZC2AEAAAAAACyFsAMAAAAAAFgKYQcAAAAAALAUwg4AAAAAAGAphBOAAAAAAMBSCDsAAAAAAIClEHYAAAAAAABLIew
AAAAAAACWQthRgVJSUtS4cWPVqlVVLnTt31jfffFNm+2XLlqlZs2YKDgpqRatuWt4GTvVqqVu3MbSpqzWQ4LWnpqRqaK4WzrZtq1WrVlVRpfCWN2P7O2x4HRD0HmRakD0ROd/Zw54rp9rsCPFRK7+fMwgfUM9SO0SesN5VKf8ESFSOc0Rn8Fkin4GxvifRBKfmRyr6PY/GYzmXewBkhp1dL7nnEfi9PR3FbZjQqzRTS7Oj2jMHv6HPB+8Wow
IE7JaYNtXQ4dTeumujkZqlAL5YxCIRGDDvTJlzItxLHFNTSj3uOVMzdS+gNJiBGAIGsyMKkQd1jl1VNIRi5AuxlavsOIgOs4pRXrRUnYXvtaVEMhkbhrKROzQNy0JOEoKUlGTIsikMPBUYDDGw4Ut5tbThYfh2aTmNTE1JaTGlWEbUnmL54pSZd41PkssXOKulIgiQbQMiHMfHXP7COLXnGMq6CzoSySGVPfvSz4mOYSSVbUtsfOWPDxdpPa2SFMpaeFIFZ3gwqFnFJ2bbwtusRstZ2ZLBoEdPvmDr16eKLKmGtpbXBpM2L1PcRkZz9bs1Xo1lCO75C7fW+7WIAyOJPW75sdMDmHRl8hfCr35eb+S97d+XpeVfHbPddtaMTZdCl1DM9z3a4K0MpK8fxOdZOd4jA+7WeMVtX6PuExhKq7sVyNjpnnr32pV+2TpZ3+xqW9XFnU02Lh0YBsNGptGGAS7QptHyHpcFMvs7IoO4ry8ibD8eRtk0kIVwJf2F9XzavmXo1ho0ov/P2c7LTm1yh7uw+e1/YTuc3lJ/Tzm0Ap2FQF2SdynYc7fnIpjfFMwQHn4Dv8a+QaENaMEqhWnNmkSVNQZnLwcT1Mu6EzIYFVX9DnNNtA0DOeF5vTjqNQhV0qGSZLBEeVMCwVWIqXiaVP3PAnX1qFFR8DgnISW23ib52mJ8zi0YMJVawJ4B2q6G0EBrm3B6ZBa+gWG1u7OW4wKOyqy4eEaqpVqVD61JMGz7Udz8fSUPGrYYjjU9TmZyjOjM7s4PRoR2ka3uh22C8zeIAzWMyJTNi8kfOb1E1Sr29E3hMxZ3P1k1a7PaD0xhC8QnwmJ4LHyIPz7IZJnhWdZ/R3e4V73tTHCmIcqMIwvVUdlCItA2vXXb8w+cnzrXWXOXkmVb97NFddxGa7L5ch4eUMbp1C5a7mCgEf8c4dAYBq7CmdfYeAxfQzn4LxgZIZmO3dQ5Zpq5yHtgQGk8tCuqOI1Hq1tFNxh/2KoNDF5tb2UfCnMnwUG6Yt0FXUyDnjS7Mfy9fP4lF4RvaPGGaD8VwKPw0D8XdNN6CvOVP5oxfP6MhXBOKjGkRhCeqX3wM17eD4YM9MdnGaD7cHAGfMyOFCG3CnnNQCLDxP9OuNyY3dfNNtpphm5XR6u9LPUuDlYH0DOp3KVxmYtJyr7YhzHPdNnLGM3b+p1q3FLg99WlZMBf+qNt2Ow8yt2TBwbbHtFkL95mDpYKgp7MXhXvFdcgE+Gd5VUL3ErDLfC0htv3LeVQh2K0Cn4LkSHd9pq3V2xT6GC1rVOMkzvhO/2yDcMBTSq8KQ0UW5B/RlN1aJXdCCa9mk7U4cPJeUKzZWxZ2JfmYqsmKkZEmhBhnaG9QaW+W1SBsmcPGDtvGx+4r5vGu/dRgDuoVPgkIsJkZl7/f1Hj+rUK3eR5lKqF2WsX6Ho5hX2KKkT/j5+7lBvn2OlzpcRZfHcT81CoW+lzQM+Jq3SEbf5nVrLzKqA/N5F8WdT88k3mmYzUm4lnW3RhyJ6d7Y+2X8b9oDSK5A/lNZRjSV5LMpCLYm5hY3J5lyw7Ak2TWo7x7e2/VQNXyjb6ev9BA/W9UNNH7PY7a7Yx9mB2TOjnuN7tT5CT3QxsNp9ZHqg0P1HRr06N3GGYMqMwnR1Yw7p3YRyG1wMUhM/M0NGjtkKrm0NxjI5+aWKgg8WTT7GZ/qgzmpRvgsyq5BAOYM2+EEpe6DQ9MDuPOqA5OhEAM7C4T3lFwO/5dhNU4lrHy9Fk7I1RJUtbj7WbO5FgCTyN1zpZh6vfO7J+1tAdxGM6LtDdsLGSV+zFyM7n3pUh3IZQa0VH/7mzJNc1G1Aj2vrbqAAdeqs7r6w5IDGH8z2w5Th2wQhWfeN8AgHvlTiiHl4/R7j9Fip5HFwyzA/4lMlq0cB8Exi4Z4h1n3vpmYbLyRT2yO7Vx4RgPDBVBVqkKcCnPkOA8x4eFN/Wvc/UzrsrCYHy4W4E+8Ijw9B4TOqQmhUfXFQwj9uqyQtJdJvk6IZB8l4GcW5dJWb5wO9bSb6SDCeRq1WLnv1rDkl/Z1BNNyH+yBm4O2EEiRr7KNqXT
```

```
Zf3bXR0tIKCgmSz2dzTWrduraysLBUWFio4OLhSa0b5+DK2Dz74oEaMGKExY8ZIktq2bavjx49r7NixmjZtmgI
D+btPTVTa96iwsDCO6rCIJUuWaMyYMVq2bBlHx1pAXl6eNm3apM2bN2vChAmS/vweZYyR3W7XJ598op49e/q5S
vgq0jpaDRoOUHh4uHta69atZYzRwYMHdemll/qlLj7hKOBwcLCuvPJKpaWluae5XC6lpaUpPj6+xNfEx8d7tJe
kNWvWlNoe/uHL2ErSU089pUceeUSrV6/WVVddVRWlwkvejm2rVq3Ow8/aMuWLe7H9ddf774TQGxsbFWWjzL48
r7t2rWrdu3a5Q6wJOmn35SdHQ0QUc14svYnjhx4oxAozjUMsZUXrGoVHyPsrbFixfrjjvu0OLFizVgwAB/14M
KEBYWdsb3qHvuuUctW7bUli1b1LlzZ3+XiPQtWtXZWZmKj8/3z3tp59+UmBgoBo2b0i/wvx7fVTrWLJkiQkJC
TELFy40//3vf83YsWNN3bp1TVZWljHGmBEjRpjkye722/cuNHY7XYze/Zss337dpOcnGyCgoLMDz/84K9VQCm
8HdsnnnnjCBAcHm7ffftscOnTI/cjLy/PXKqAU3o7tX3E3lurL27H95ZdfTGhoqJkwYYYLZuXOnWblypalfv7559
NFH/bUKKIW3Y5ucnGxCQOPN4sWLzZ49e8wnn3ximjVrZoNG+avVUAJ8vLyzObNm83mzZuNJDNnzhyzefNms3/
/fmOMMZMnTzYjRoxwt9+zZ4+54IILzH333We2b99uUlJSjM1mM6tXr/bXKAU3o7tG2+8Yex2u0lJSfH4HpWTk
+OvVUApvB3bv+JuLNWXt2Ob15dnGjZsaG666Sbz448/m66SSe8kpOTq75wnJW37
9vTEXZUb9607Zdffmk6d+5sQkJCTNOmTc1jjz1mHA5HFVeN8vBmbIuKisGl+1cGQHAADwkJ+fr7feekvjxo3TgAEDtHDhQve8devWKSAgQ
Gk1SWlqav8sAAMDSjh8/rvbt2yslJcznzn/fzzzn//fJj+Kv5bXHD16RMxhsIk8C3CARGV2+rsVBiG/bvy1ysylJcXnZfzzn//Uyy+/rYH4U
GlpabrqqqtOwQUXqEuXLtq5c6c6e7TfEpI6+99poaN26s8PBw3XzzzcrLy303ady4seb0nevRb4cOHfTwww+7n8+
ZMOdt27bVhRdeqNjYWI0fP175+fk+rdNfT205/fbbNX/jwYM2ePVvR0dG6+OKL1lZiYqKKiInebgoICPfDAA4qNj
VVISIiaN2+u1NRU9/z1+KCgkJUXR0tCZPniyyHw+Ge36NHD03NHO02cOFH33nuvLrroIkVGRuql17S8ePHdcc
ddyg0NFTNmzfXRx995FHrtm3b1K9K9fP9WpU0eRkZEaMWKEo/8/9d5/WGwCA6mTChAlKTO0/XkiVLtHXVrVg0dOlR9+
/bVzz//LEn64IMPP1LRpU61cuVJNmjRB48aNTrbMGB05csTrvgg7AACh6VLl6pVq2bZkl/vGPf+iVV16RMca
jzbRp0/T0O9z6NNstvtuvPOWytWrLUFrhhh5o6IkSO1+3jdkLNuAwd1gDhAUMs
z7T/ffff87rV2zt2rXavXu31q5dq6xatVRXe5DuZ0mj/Xf+ehj0aJFqlevnr755+30evn+O7/OacgHWMvl68JYHD
v+v777zVv3jylpqbq0UcfhjdDLtnNnDh2Q2aNNjpbBtnQ0MlKnY/uQM8Yf+//IPZbpcTnewcSGzAVH+pC9vFxM97o+
jzbRp0/T0009r06ZNstvtuvPOOz3m9WytWrNDKlSu1cuVKrV+/XtOmTXVX+ed0FXTLUdwmG/PzwIzZjd4dKpO8/N8+lJ80eBJIaAa1j4pNKMn
tSzZO917NhRmzZt0urVq5Wdna1hwZ4aV2HoDAOAPv/zyixYsWKBly5apW5duyGnyj5h+atPmbNbJKCJf1FB1nfnw/XvuWxc/rU9U6J2QiEDCQtJPKL/T+X0dvoF0a8r7YXC9UJq+7A4nTvxvy6Ty+iu
tSzZ0917NhRmzZt0urVq5Wdna1hw4ZV2HoDA0APv/zyixYsWKBly5apW7duyGnyj5h+atPmbNbJKCJT1ZBb6/XPMMM3OOMnRZmwqlwRusH4eJZmQVw9b
H+/fv17Jly/Tqq69q4cKFysjIOE033eR9hwYAAOAOXbp0MXPnzjXGGFNUVGTq1atn1q5da4wxzu3atUaS+fTTT
93tP/zwQyPJnDx50hhjTHJysrngggvMBm6uu819991n0onfu7H4eFXdnnnnmGY9+27dvb5KTki9uta9myZebiiy9
2P1+wYIEJD8v/v1zolJyeb9u3bu5+PGjBDw3z4cG0MMTt37jSSzJo1pc3tSpU03Lli2Ny
+VyTOtJSTFl6tQxTqfTGGNM9+7dzdVXX+2e73AhGjBjhnnbo0CEjyaSnpxjjHnkkUdM7969PfocOC0
AkWR27txZrnUFAKA6kGSWL1/ufr5y5Uojy/Vx44YUeD7vdboYNG2aMMeauv+464zDYsLs+464zMvIyPDSDI7duzwqn+u2QEAA
Nx27typb775RsuXL5ck2e12Dp8vNYCdM6pmDK1N2MCNL9Ut27e7v9HR0dLkg4fw8fPqqWTz/
9VLNmzdKOHTuUm5sr8hU6d06cSJE7rgggt8XUW3Nm3ayGazeT4ww8/SJK2bNdu3Kz4+X
gEBAe5pXbt2VX5+vg4ePkPO0jeDqdvJ5vNposvvlht27Z1T4uMJjJQk97b5/vvvtbXbt2hKvL7J79261aNHC19UFAMC
v8vPzZbPZ1JGR4fH5H5K8K8n9uRcdHS6dWtWq/Jfx4ZUnwdj/Ig7AAAAG6pqalyOBweFyQ1xigkJET/+c9/3/3
N0CgoLc/y/e4Xe5XCXOL25z+vzAwMAzTo05/XoZ+/bt08CBAzVu3Dh99thjGvgijioiI0BdffKHRoersLCwQsK0smq
sXbv20S+/tD7K2nb5+fkaNGiQnnzySZUjkKj7D7ZlmRx+WTk5IJP5ZWsIWUBCOIEj76R5t4MMzJwvxUeLbNl2TrDvsUwzW3cziPgNjYfE2fY5ZC8yDQ+
fTTT5KkuLg4r/oj7AAAhV199VU8//bR69qVg99FyBLKj8hJ7w04MFavHixWrVqVSF9XXLJJTp06JD7eW5urvbu
+npGRIZfLpaeff1qBgX9eYmzpOqUV0nd5tG3bVi6dCQsIZ81u3bq133nlHHxxhh3YLFx40aFhoaqYcOGP
vd7xRVX6J133lHjxo1+utd6xBDsK1e1pq6mBxOZMOYjlYmT4yerl9gFKdTkKBDqRM54hRcTpbxp+Rr1tdzh11
Ulpamdu3aacCAAUpISNAVV1yhO++8U3PnzpXL5VmdV8dGnqGEjN+/HvtTpXyOkG69Ie+IKqpAnjMbiyHR0Jw9z7Ua
4YM8bgTybnq2bbOnXnvtNX3++ef64YcFNGrUVK9Dw5bj3bb6i1iI999xz2rNnjj157ZTXNnz+/wvo/m8aNG2vUqFG
6884ltWLFCu3du1fr1q7ByZ7x4yCtWMiRoveNbYVPVVfeekp0TlXSU5A5nfJGYmKgjR47470uO0lltuObffqvgdu
3fr448/1hh133GC0n01lRqwcAQKXYtGmTOnbsSSpKS1LfjRz300ESpAULFmjkyJH6n//5H7H7Vs2VKDBw/
Wt99+6z79MzAwUB988IRq1auZna665RgMGDFrDr1q21ZMKsr2vhTwYAAEDSn6ewJCQkYM4VMPPaWtW
7dWSF9TpkzR3r17NXdgQIWHh+uRRx7xz9TrpyPWSP/1MW/ePE2dO1X
```

```
jx4/XH3/8oUaNGmnq1KmSpAYNGmjVqlW677771L59e0VERGj06NGaPn36OfUZEx0jjRs36oEHHlDv3r1VUFCgu
Lg49e3b95xCFAAAqkKPHj3OOE31dEFBQZoxY4ZmzJhRapuYmBi9884751xLgCmrEgAAAAAAgBqGPxEAAAAAAAAB
LIewAAAA1Xps2bVSnTp0SH2+88Ya/ywMAAFWM01gAAECNt3//fo9b154uMjJSoaGhVVwRAADwJ8IOAAAAAABgK
ZzGAgAAAAAALIWwAwAAAAAAWAphBwAAAAAAsBTCDgAAAAAAYCmEHQAAAAAAwFIIOwAAAAAAgKUQdgAAAAAAAEs
h7AAAAAAAAJby/wA6y/nxHyQkJwAAAABJRU5ErkJggg==\n"
          },
          "metadata": {}
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "#### Children"
      ],
      "metadata": {
        "id": "AIC4BKjYHw1u"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "analyse_numerical_column(\"CHILDREN\")"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 542
        },
        "id": "qDH3-EkuHL-w",
        "outputId": "41de4920-903d-47b1-9ead-f643d711eb3f"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 1300x600 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAABEMAAAINCAYAAADV3rHEAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAABEckl
EQVR4nO39f5iVdYE//j9ngBkQGRCMGSZBOCwUMRSQkP24mbzFH7lptr7pTS6uvXUrNJEydRPH4iwpYQhrF619
UO3d9+blu4uLaJJbYgIkb9/baikDlggoxA/ZM73jy5mG0FFnZkzcj8e13Wua87rfp1znje9rvGeZ/d9n4pSqVQ
KAAAAQEFUljsAAAAAQHtShgAAAACFogwBAAAACkUZAgAAABSKMgQAAAAoFGUIAAAAUCjKEAAAAKBQlCEAAABAo
XQud4D3g6amprz44ovp0aNHKioqyh0OOHAAAA2IlSqZRXX3019fX1qax88/M/lCG74MUXX0z//v3LHQMAAADYBat
Wrco+++zzptuVuIbugR48eSf7Oj1lTU1PmNAAAAAMDONDY2pn///s1/x78ZZcgu2u2uV2hTE1NjTL0
6gCAAAAhaIMAQAAAApFGQIAAAAUijIEAAAAKBRlCAAAAFAoyhAAAACgUHy17m5Pr66uzh577NG
```

GiQAAAKC8lCG7sY0bN2bAwEH5w8trdvk1fT7QN88/u1IhAgAAwG5LGbIb27x5c/7w8pqcMO3/parbnm87f8sfX
8u/ff0z2bx5szIEAACA3ZYypACquu2Zqu415Y4BAAAAHYIbqAIAAACFogwBAAAACkUZAgAAABSKMgQAAAAoFGU
IAAAAUCjKEAAAAKBQlCEAAABAoShDAAAAgEJRhgAAAACFUtYyZNGiRTnxxBNTX1+fioqK3HHHHS22l0qlXHLJJ
enXr1+6deuWsWPH5umnn24xZ+3atZkwYUJqamrSq1evfP7zn89rr73WYs5DDz2U/+//+//StWvX90/fPzNnzmz
rXQMAAAA6qLKWIRs2bMhHP/rRzJkzZ6fbZ86cmdmzZ2fevHlZsmRJunfvnnHjxmXTpk3NcyZMmJBHH30OCxYsy
F133ZVFixblrLPOat7e2NiYY445Jvvuu2+WLVuWf/iHf8g3vvGN3HDDDW2+fwAAAEDHO7mcH37cccflu0OO2+m
2UqmUWbNm5eKLL86nPvWpJMD1NbW5o477sj48ePz+P3OOPPZ/78+Vm6dGlGjBiRJLnuuuty/PHH55vf/Gbq6
+tzyy23ZMuWLfne976XqqqqDBkyJCtWrMg1l1zToj6Tq/AAAAiqHD3jNk5cqVaWhoyNixY5vHevbsmVGjRmXx4sV
JksWLF6dXr17NRUiSjB07NpWVlVmyZEnzzo997/vzb17at2+lnb968OY2NjS0OeAAAAwO6hw
5YhDQ0NSZLa2toW47Wltc3bGhoa0rdv3xbb03funN69e7eYs7P3+PPPeKPp06enZ8+ezY/+/fu/9x0CAAAAOoQ
OW4aUO0UXXZT169c3P1atWlXuSAAAAEAr6bBlSF1dXZJk9erVLcZXr17dvK2psf3111/P2rV6r/P2rVrW8zZ2
Xv8+We8UXV1dWpqalo8AAAAgN1Dhy1DBg0alLq6uixcuLB5rLGxMUuWLMno0aOTJKNHj84rr7ySZcuWNc+5555
70tTUlFGjRjXPWbRoUbZu3do8Z8G8GCBfnIRz6Svfbaq532BgAAA0goylqGvPbaa1mxYkVWrFiR5E83TV2xYkWef
/75VFRUZPLkybnyivz05/+NA8//HD+5m/+JvX19TnppJOSJAceeGCOPPbYnHnmmXnggQfyX//1Xzn77LMzfvvz
41NfXJ0n+z//5P6mqqsrnP//5PProo7nttttttvy7W9/O10mTCnTXgMAAADlVNav1n3wwQdz1FFHNT/fXlBMnDgxN
998c772ta9lw4YNOeuss/LKK6/kL/7iL/7iLzJ//vz777dq1+TW33HJLzj777Bx99NGprKzMMKaecktmzZZdv79mzZ/7
zP/8zkyZNyvDhw7P33ZNyvDhw7P33nvnkksu8bW6AAAAUFAVpVKpVO4QHV1jY2N69uyZ9evXv6/uH7t07t07J18zP1Xd3
z73lg2NuX3KsVm7dq1LiAAAAHjf2dW/3zvsPUMAAAAA2oIyBAAAACgUZQgAAABQKMoQAAAAoFCUIQAAAAEChKEM
AAACAQlGGAAAAAAIWiDAEAAAAAKRRkCAAAAFIoyBAAAACgUZQgAAABQKMoQAAAAoFCUIQAAAAEChKEMAAACAQlGGA
AAAAIWiDAEAAAAKRRkCAAAAFIoyBAAAACgUZQgAAABQKMoQAAAAoFCUIQAAAAEChKEMAAACAQlGGAAAAAAIWiDAE

W2L19XW1jZva2hoSN++fVts79y5c3r37t08542mT5+eyy67rA32CAAAACi3d3VmSHtpamrKYYcdlquuuiqHHnp
ozjrrrJx55pmZN29em37uRRddlPXr1zc/Vq1a1aafBwAAALSfd3VmyFFHHZWKioo33X7PPfe860B/rl+/fjnoo
INajB144IH513/91yRJXV1dkmT16tXp169f85zVq1dn2LBhzXPWrFnT4j1ef/31rF27tvn1b1RdXZ3q6upW2Qc
AAACgY3lXZ4YMGzYsH/3oR5sfBx10ULZs2ZLly5dn6NChrRZuzJgxefLLJJ1uMPfXUU9l3332T/0lmqnV1dVm4c
GHz9sbGxixZsiSjR49OkowePTqvvPJKli1b1jznnnvuSVNTU0aNGtVqWQEAID3h3d1Zsi111670/FvfOMbee2
1195ToD933nnn5YgjjshVV12VU089NQ888EBuuOGG3HDDDUmSioqKTJ480VdeeWUOOOCADBoOKFOnTk19fX1OO
umkJH86k+TYY49tvrxm69at0fvsszN+/HjfJAMAAAAF1Kr3DPnc5z6X733ve632fiNHjsztt9+ef/qnf8rBBx+
cK664IrNmzcqECROa53ztal/LOeeck7POOisjR47Ma6+9lvnz56dr167I4//vj8xV/8R
XOhAgAAABTLuzoz5MOsXry4RQnRGj75yU/mk5/85Jtur6ioyOWX57LL7/8Tef07t07t956a6vmAgAAAN6f3lU
Z8ulPf7rF81KplJdeeikPPvhgpk6d2irBAAAAANrCuypDevbs2eJ5x2eJ5ZWVlPvKRj+Tyyy/PMcccOyrBAAAAANrCu
ypDbrrpptb0AQAAANAu3tM9Q5YtW5bHH388STJkyJAceuihrRIKAAAAoK28qzJkzZo1GT9+fH7+85nV69eSZJ
XXnklRx11VH70ox/lAx/4QGtmBAAAAGg17+qrdc8555y8uqrefTRR7N27dqsXbs2jzzySBobG/Pl3/PlL3+5tTMCA
AAAtJp3dWbI/Pnzc/fdd+fAAw9sHjvooINy7bXX55+bTf5zxl5zHwHi
Vd1WGfOITn8i5556bF198sXnshRdeyHnnnZejjjz661cIBAAAAtLZ3VYZ85zvfSWNjYwYOHj9998/+++HwYNG
pTGxsZcd911rZORAAAAoNW8q3uG90/fP8uXL8/dd9+dJ554Ikly4EZjJYjUThj5u8JlNBzvaMdZd+pt6c1vM+mT+mrUbvLL/6sH
BaWxsTEVFRf7X//pf0eecc3LOOedk5MiRGTJkSH7xi+0OVVYAAACA9+wdlSGzZs3KmWeemc/a9pt3ZXm5KqUFEBOemEc=
3+Xa665ptXCAQAAALS2d1SG/OY3v8mxxx77ptuPOeaaYLFu27D2HAgAAAGgr76gMWb169U6/Une7zp075+WXX37
PoQAAAADayjsqQz74wQ/mkUceedPtDz30UPr16/eeQwEAAAC0lXdUhhhx//PGZOnVqNm3atMO2P7xj7n0Okvzy
U9+stXCAQAAALS2d/TVuhdffF+/OMf58Mf/nDOPPvvf0QjH0mSPPHEE5kzZO62bduWr3/962620OSFAAAAKA1vKM
ypLa2Nr/61a/yxS9+Rdd2pgq4LnOVgekwCbj/+OWT+3w7TLbn2+
WdSKpVywAEHZK+99mqLfAAAAACt6h2XIdttmtAQAAAGhz7+gGqgAAAADvd8oQAAAAoFCUIQAAAEChKEMAAAACAQnlflSFXX311311KioqoMnny50axTZs2ZdkSenTp0/23HPs3//PM54YQTsscbe6vv/+f7e6Rv3
745//zz8/rrr7dzegAAAAKAjeN+UIUuXs0//uM/5pBpDDmkxft555555+555+X0O+/Mv/zLv+S+S+++mE9/+tPN27Pd
t25YTTjghW7Zsya9+9at8+rTfcrh8vEKHezWWBjd1+3w+ZvYx+5ZlgkTJuTG2/MXnv1Ty+fv36fPe73
80111yTT3ziExk++fHhuuumm/0pXv8r999+fJPnP//zqf2fqjJNwl7yZOVsbq9MyNnr0O/4nu1+H0+o+rEVM6VzyLIRHM
AAKBM3hdlyKRRJk3LCCSdXLVuWrVu3gfBTdhb7qrehZxq3j6q7bSuwD8k/dCD3ujWGn3e+e68KX9b8bbXeEt7lMLEn
31Op5+3efPmNDY2tgAAAAu4f05Q7wdn70ox9l+fLLlWbp06Q7bGqdKW1VlVlVV69erUYr62tUvZ/7z/wx/Mq6HV9Iv58+OPKxNC6cOL4/
fvm1npk+fnssuu6wV0dKwVgMAAAAdTYc+M2TVL09c0Xnqp2XvhfqPAJbJdxnVUOS6tTe67uXxnBKlAXxcoJQtfCCQGqLRBK
f80bV1dWpqalp8QAAAAB2Dx26bCiKqLR3pD/+PjFTT7ZerBTTsGmjx8uvb9fR1NbNO3uZ+RzLkmJS+7d7p1VG7lzOBAAK
pKMHj06/Dz/8cNasqB8EAawQHsxxwUL+2uo19D+xsMyzsqzoR0JRo2xkdHyNY9bUfHR9pj7rs9lUbrRvcSpXsgtBwMfkWRJZCC
yNwmwWuzBrFHTOzfe2kc3DJPMMFGqGY3ESQeGkLFtf3uqlPS4mxtzuQT/bTCnYtxjSLXR29/p7uvnO82zz6tq/1hxyeIXKuTqh
3U+N67LaZwmk4fUXssKjZVAuWUUV/l/6W4Gx8XFHtBrwxrIaGhUFudE1zf3GVt1m31+AwHrbVmH5Kf3e4ZcM0gHptyjvn
c9brrJXVKiVU3qi/uZ2mZw6a1JVVZ/vwyDUoNnVgc1C0BQD2tf6dtZvyYmiV4zn7/3YcCMF7dzsl7/3dPb/AQcHMzSRc
nzW3yZl

W2L19XW1jZva2hoSN++fVts79y5c3r37t08542mT5+eyy67rA32CAAAACi3d3VmSHtpamrKYYcdlquuuiqHHnp
ozjrrrJx55pmZN29em37uRRddlPXr1zc/Vq1a1aafBwAAALSfd3VmyFFHHZWKioo33X7PPfe860B/rl+/fjnoo
INajB144IH513/91yRJXV1dkmT16tXp169f85zVq1dn2LBhzXPWrFnT4j1ef/31rF27tvn1b1RdXZ3q6upW2Qc
AAACgY3lXZ4YMGzYsH/3oR5sfBx10ULZs2ZLly5dn6NChrRZuzJgxefLLJJ1uMPfXUU9l3332T/0lmqnV1dVm4c
GHz9sbGxixZsiSjR49OkowePTqvvPJKli1b1jznnnvuSVNTU0aNGtVqWQEAID3h3d1Zsi111670/FvfOMbee2
1195ToD933nnn5YgjjshVV12VU089NQ888EBuuOGG3HDDDUmSioqKTJ480VdeeWUOOOCADBoOKFOnTk19fX1OO
umkJH86k+TYY49tvrxm69at0fvsszN+/HjfJAMAAAF1Kr3DPnc5z6X733ve632fiNHjsztt9+ef/qnf8rBBx+
cK664IrNmzcqECROa53ztal/LOeeck7POOisjR47Ma6+9lvnz56dr167I4//vj8xV/8R
XOhAgAAABTLuzoz5MOsXry4RQnRGj75yU/mk5/85Jtur6ioyOWX57LL7/8Tef07t07t956a6vmAgAAAN6f3lU
Z8ulPf7rF81KplJdeeikPPvhgpk6d2irBAAAAANrCuypDevbs2eJ5x2eJ5ZWVlPvKRj+Tyyy/PMcccOyrBAAAAANrCu
ypDbrrpptb0AQAAANAu3tM9Q5YtW5bHH388STJkyJAceuihrRIKAAAAoK28qzJkzZo1GT9+fH7+85nV69eSZJ
XXnklRx11VH70ox/lAx/4QGtmBAAAAGg17+qrdc8555y8uqrefTRR7N27dqsXbs2jzzySBobG/Pl3/PlL3+5tTMCA
AAAtJp3dWbI/Pnzc/fdd+fAAw9sHjvooINy7bXX55+bTf5zxl5zHwHi

TU5Nzzjkno0ePzsc+9rEkyTHHHJODDjoop512Wmb0nJmGhoZcfPHFmTRpkrM/AAAAoIA6dBkyd+7cJMnHP/7xF
uM33XRTTj/99CTJtddem8rKypxyyinZvHlzxo0bl+uvv755bqd0nXLXXXfli1/8YkaPHp3u3btn4sSJufzyy9t
rNwAAAIAOpEOXIaVS6W3ndO3aNXPmzMmcOXPedM6+++6bf//3f2/NaAAAAMD7VIe+ZwgAAABAa1OGAAAAAIWiD
AEAAAAKRRkCAAAAFEqHvoEq8M5t3Lgxmzdv3qW51dXV2WOPPPdo4EQAAQMeiDIHdyMaNGzNg4KD84eU1uzS/zwf
65vlnVypEAACAQlGGwG5k8+bN+cPLa3LCtP+Xqm57vuXcLX98Lf/29c9k8+bNyhAAAKBQlCGwG6rqtmequteUO
wYAAECH5AaqAAAAQKEoQwAAAIBCUYYAAAAAhaIMAQAAAApFGQIAAAAUijIEAAAAKBRlCAAAAFAoyhAAAACgUJQ
hAAAAQKEoQwAAAIBCUYYAAAAAhaIMAQAAAApFGQIAAAAUijIEAAAAKBRlCAAAAFAoyhAAAACgUJQhAAAAQKEoQ
wAAAIBCKVQZMmf0nAwcODBdu3bNqFGj8sADD5Q7EtCGNm7cmHXr1u3yY+PGjeWODAAAtIPO5Q7QXm677bZMmTI
l8+bNy6hRozJr1qyMGzcuTz75Pr27VvueEAr27hxYwYMHJQ/vLxml1/T5wN98/yzK7PHHnu0YTIAAKDcCl0GX
HPNNTnzzDPzt3/7t0mSefPm5d/+7d3/yve99LxdeeGGGZ0wGtbfPmzfnDy2tywrT/l6pue77t/C1/fC3/9vXPZPP
mzcoQAADYzRWiDNmyZUuWLVuWiy66qHmssrIyY8ePzeLFi3eYL1y3eLFi3eYv3nz5mzevLn5+fr165MkjY2NbR+2FW3Pu2Hdm
mzdtOFt52/545/mrFq1apf2tVQqaKiYpeytPn5lVKg0sHOmXPN/ns8lvD/r1c3a8+NuM6r9QXwH+h6VPQzXlnqn/4fKdxQpt0CBVHqOKL7Xf+pX5d
bvrcc7TNXjo6Z4/2YWY60meP9mFm0jpnj/ZhZjo6Zoy0zd+3aNd26ddvl9+4Ith/Ll0qlt5xXXq7GbuBF198M
R/84Afzq1/9KqNHj24ev+CCC7JgwYIsWbJkh/lz587N9ddd/fd/e9rPDPW9yHUVwv9oJrNbaVkrV9foLXKSDbjbMG1yAkx2wtl2xb71nZgzhYvesPDl9X8x6wqBTMnAmXzW/gRPiTfQgsU41yS4ZEZ2LW8oHdXTf5Mc82ls+dYWYhg0WL9MVkOJrfl/XdGqRNaf2mYgSq1kAg0CtfM+mm1a+SSXhhlK1ZlkRhmdVpDVUcnC2wWGQ23U5+NCWU8TRN/gwW+WKHAf29TpdlO2BvrqP00D1btmxNlYRoHYszQ0Cuy9fBUNUmGv8lcVvGJ+wrnTVb8/AB0OVxsGTs3Y4DTqYdLoa7ZOm+Y9mC7p4+1nSqGtvP8WbzSWrzSZqh0iIQIsPSlYLw0NcKxG41pfY3ZcqbSYbgP+fcZy9GyIzahaAr+ARLDepqTLzCRIA2O68Uy1fsXq3qKkNkxtfxwBKnmfHcxF6
PnvWhQTdagcgQ+D2qR2mgJxR57Ue8SAkE5aSPRhaiJkuRNiq7pTzaz07tEoSJQzAzewxtPdmOPf3iqNK2eWSkCwBSSBlmzWKTYlFBnRllAsHoQVfExWwCvlIBkSmqU

```
NKAAQNKXbt2Le23336lr3/966XNmzeXOxpt7N57793pscHEiRNLpVKp1NTUVJo6dWqptra2VF1dXTr66KNLTz7
5ZHlDOybeai2sXLnyTY8j77333nJHp5W93e+FN9p3331L1157bbtmbAsVpVKp1E69CwAAAEDZuWcIAAAAUCjKE
AAAAKBQlCEAAABAoShDAAAAgEJRhgAAAACFogwBAAAACkUZAgAAABSKMgQAAAAoFGUIAFAWDQONOeecc7Lffvu
luro6/fv3z4knnpiFCxcmSQYOHJhZs2bt8LpvfOMbGTZs2C4/f6OPf/zjqaioSEVFRaqrq/PBD34wJ554Yn784
x/vMHf7vIqKitTU1GTkyJH5yU9+OmLOzTff3GLe9kfXrl2b55x++umpKjIlVdf3eK1d9xxRyoqKt7iXwkAaAv
KEACg3T377LMZPnx47rnnnnvzDP/xDHn744cyfPz9HHXVUJk2a1O0aff+aZZ+alll7Kf//3f+df//Vfc9BBB2X8+
PE566yzdph70OOf+UwefvjFnNqamry0ksvtXg899AtvE1lM2bMyLp169wC
At9e53AEAgOL5Ope+lIqKijzwwAPp3r178/iQIUNyxhlntPnn77HHHqmrq0uS7LPPPPvnYxz6WwYMH54wzzsipp
56asWPHNs/t1atX6urqUldXlyuuuCLf/va3c++992boOKHNcyoqKprf782MHTs2zzzzTKZPn56ZM2e2zY4BALv
EmSEAQLtau3Zt5s+fn0mTJrUoQ0qbrs1atX+4dKMnHix0y11147vVwmSV5//fV897vfTZJUVVW94/fv1KlTrrrq
lx33XX53e9+956yAgDvjTIEAGhXzzzzzTEqlUgYPHvy2cy+44ILLsueeeeLR5XXXVVm+SqrKzMhz/84Tz77LMtxj/
72c9mzz33THV1dc4777wMHDgwp556aos569869ev3yHnccccdt8NnnHzyRk2bFguvfTSNtkHAGDXuEwGAGhXpVJpl
+eef/75Of3001uMzZ49O0s4Ws569evWrlVH9SKpV2uKHptddem7Fjx+a3v/1tzjvvvMyePTu9e/duMadHjx75Zvnx5i7F
u3brt9DNmzJiRT3ziE/nqV7/auuEBgF2mDAEA2tUBBxyQioqKPPXUUxmwQgJaAE1UAVgOcpj65fHbKG2nKRAK
2n2e6yzdph7O035fjjj89jjz2Wvn37Ns+prKzcNKH3PPb8FLOxdxxRAl2ukPAGkgMbVp/Q7GcXE/rUBo
NEh/tIU1Vs/eP7l1G3B9eKcQWrqBFKVqt+ClHtTU3DMWzXVn5ypr7
EBtBu9fP9x6I12crhAcy1THIEv/RlSHM5mXcm0J+fKm27K0Zpr
2mMW8x2nUKfAFzrC27eXPCc6cVLSXLK5ZE9b3mYJ7JuxXXAwtU1ayw5Izdi
FzDyIN6Ae02pF6OoQr+p3nhrdO/6TEG+D7E4pHiUNLjMwB9X6n/g/yWK/+Hb
```

OAnkrnTp12i2+Bg0AgLZVhONG9wwBAAAACkUZAgAAABSKMgQAAAAoFGUIAAAAUCjKEAAAAKBQlCEAAABAoShDA
AAAgEJRhgAAAACFogwBAAAACkUZAgAAABSKMgQAAAAoFGUIAAAAUCjKEAAAAKBQlCEAAABAoShDAAAAgEJRhgA
AAACFogwBAAAACkUZAgAAABSKMgQAAAAoFGUIAAAAUCjKEAAAAKBQlCEAAABAoShDAAAAgEJRhgAAAACFogwBA
AAACkUZAgAAABSKMgQAAAAoFGUIAAAAUCjKEAAAAKBQlCEAAABAoShDAAAAgEJRhgAAAACFogwBAAAACkUZAgA
AABSKMgQAAAAoFGUIAAAAUCjKkN3Y3nvvnSQpVe1R5iSU24ABA3LDDTdkwIAB5Y4CAABQdp3LHYC2O6VLlz/9U
NGpvEEou65du+bDH/5wuWMAAAB0CM4MAQAAAApFGQIAAAAUijIEAAAAKBRlCAAAAFAoyhAAAACgUJQhAAAAQKE
oQwAAAIBCUYYAAAAAhaIMAQAAAApFGQIAAAAUijIEAAAAKBRlCAAAAFAoyhAAAACgUJQhAAAAQKEoQwAAAIBCU
YYAAAAAhaIMAQAAAApFGQIAAAAUijIEAAAAKBRlCAAAAFAoyhAAAACgUJQhAAAAQKEoQwAAAIBCUYYAAAAAhaI
MAQAAAApFGQIAAAAUijIEAAAAKBRlCAAAAFAoyhAAAACgUJQhAAAAQKEoQwAAAIBCUYYAAAAAhaIMAQAAAApFG
QIAAAAUijIEAAAAKBRlCAAAAFAoyhAAAACgUJQhAAAAQKFOLncA2l7FltdSueH35Y5RNpWb1pc7AgAAAB2IMmQ
31r9//1RWdkrXF5YnLywvd5yy6lJVnZ49e5Y7BgAAAB2AMmQ3Nnjw4PzTP92a9eudGdGzZ8/U1taWOWYAAAAdg
DJkN1dbW6sEAAAAgD/jBqoAAABAoShDAAAAgEJRhgAAAACFogwBAAAACkUZAgAAABSKMgQAAAAoFGUIAAAAUCj
KEAAAAKBQlCEAAABAoShDAAAAgEJRhgAAAACFogwBAAAACkUZAgAAABSKMgQAAAAoFGUIAAAAUCjKEAAAAKBQl
CEAAABAoXQud4D3g1KplCRpbGwscyIC1599dUkSf/+cucBAAAAHg7r776anr27Pm
m2ytKbleXkKamprz44ovp0aNHKioqyh1nlzU2Nqa2tra89/Rramb6aY/+HMrIW2M5aYDtrge2sBRLrgP9hLbDd+
3UtlEqlvPrqq6mvr09l5ZvfGcSZIbugsrIy++yzT7ljvGs1NTXvq8Vxbj66FYqxT7GmXyLIr5F6Cz7cZ6y3KZy1RLrgP9hLbDd+3E
tvNUZIdu5gSoAAABQKMoQAAAAoFCUIbux6urqXHrppamuri53FNrJa5t1uWnD3pF0a12BhmRHKbs1wR/tOcpJNqpOLqHP7SHyYyuz3KNh+wdOT17Gb3z4xlszO5aoUeqDvM7e8sVfkqdpdtrDkE+0UxFOqFvHGJvKFq9nbnyTXbeBzeq/1Zbu+4Zb2mgWLJmxnk3Gdn1wJBZ4wEK+4eAGnuPu23mc3QxjYM3z5lmrlSL+f+OpRK+Wnni8yG3U8Hd94uqbmm9d7jUI0TdhnX5MdJjbm+H99+N8Dh4f1UKc+7nBuyNHlqsOn9x6Pp4TrT73m4A/5nZvAY/yS3mPNzGGMhZ7P+4U2Lq6jVoBDM9d1cmmQZfwbc3Aw+vq55R3tkzVk8txmaVo0WYLmQGGSkwe0S8b4A6uDoTcUE9Egjo6YHw4N+3LVmWUPYLlK2I2AaWWkTA0IC//cWrnLAcS5dSMxvYN3g2dIN+FOIRU4XI5GfzEi0K1h8vfcOgoZmgF3jdV4lYIcnNgGRz7ub3Pg9Q4f+w7iPVyLcgYpKr1jwq8tUGkVH6mVWtsE8N/6FLS8mNWiSVgjMnovZ9XfnYg1HQc1f3Z0MLTeDLYvj2JQ/T5E+XgnrjLKt+1zL6b9nBnt8lXrVsJvL3TX0sbtkuaHsNh8XeKOF6M+ITGcuf6fr11IVXL93fQv5m5C3WWqDA3CnR4NfCvIY5P/FOmYkHbM5AFHNOXCW5a+qvmsrqF0N6M8AFVpjBSdnlUOJOIMINl5nF2JYlPyvd0YFbqDTtwNhB0zZn9S2rb2/Uh3n0jbfI/Xw1qIK7N+LSfW5nWpvjAoZmm6OXFVQUF7PuLrK4N3D5c57/rFU3GU+9p0lgtQ0lSc9ZV4U6z+n6t2X9KHnuypaInVnpqfwZzFBBE3TnXWeCMdafJtxkW+q4MxFG2PxBbuTdXXQ8uHb0VZqMyUGcbhRrCLmTpQS2QQrTnU3YA9KeVwiOhBDRgpx7HEmP0UsLrVrFbfdr3tyl9DU9+F75xbDvtdZg2F0jdWOxaBqcNrRL7yWUfZBuu9yHjMPUw4dubqC23bkxaf+SMAFjPx1H0vGeQHLaTTVsB8dJx5+yt2KU5eGWUtwR8tSYoIWoLGrVWxaBiRpYCi0tTUuVPnvRPNmffz2yxlKdWkjr/fk+nJ4kg2NuPfEeUmPUbF7W+x/pLtbCfb6zXPuH3sFH9nUPNrMLpwVl7QjfJ3PBFmIaJVIkYhDJHw9jbNQXCHNTfqqCuCmJzp3XRO/S5l8dN90Fr43DbmQ7PvS2XrpauVP/1HovtGlO2X3vqdbXNyaTfnfYmMxZ0IWnZ/4vqrVxmJfk95dT2VpO9XrtRaHvNS3N7JbLx1vDGvh5F2XT4MuhK/wDd5v4yNrj1Sk

```
rEtVJRKpVI79S4AAAAAZeeeIQAAAEChKEMAAACAQlGGAAAAAIWiDAEAAAAKRRkCAAAAFIoyBAAAACgUZQgAAAB
QKMoQAAAAoFCUIQBAWTQ0NOScc87Jfvvtl+rq6vTv3z8nnnhiFi5cmCQZOHBgZs2atcPrvvGNb2TYsGG7/PyNP
v7xj6eioiIVFRWprq7OBz/4wZx44on58Y9/vMPc7fMqKipSU1OTkSNH5ic/+UmLOTfffHOLedsfXbt2bZ5z+um
np6KiIldffXWL195xxxx2pqKh4i38lAKAtKEMAgHb37LPPZvjw4bnnnnnvyD//wD3n44Yczf/78HHXUUZk0aVKbf
/6ZZ56Zl156Kf/93/+df/3Xf81BBx2U8ePH56yzztph7k033ZSXXnopDz74YMaMGZPPfOYzefjhh1vMqampyUs
vvdTi8dxzz7WY07Vr18yYMSPr1q1r030DAN5e53IHAACK50tf+lIqKirywAMPpHv37s3jQ4YMyRlnnNHmn7/HH
nukrq4uSbLPPvvkYx/7WAYPHpwzzjgjp556aqKh
ofr83M3bs2DzzzDOZPn16Zs6c2TY7BgDsEmeGAADtau3atZk/f34mTZrUogjZrlevXu0fKsnEiROz11577fRym
SR5/fXX893vfjdJUlVV9Y7fv1OnTrnqqqty3XXX5Xe/+917ygoAvDfKEACgXT3zzDMplUoZPHhwuzxO0zTTLsue
ee7Z4XHXVVW2Sq7KyMh/+8Ifz7LPPthj/7Gc/mz333DPV1dU577zzMnDgwJx66qkt5qxfvz57771ni8dxxz/Hyy
Sdn2LBhufTSS9tkHwCAXeMyGQCgXZVKpV2ee/755+f0009vMTZ79uwsWrSolVP9Sa1U2uGGptdee23Gjh2b3/7
2tznvvvPMye/bs907du8WcHj16ZPny5S3GunXrttPmDFjRj7xiU/kq1/9auuGBwB2mTIEAGhXBxxwQCoqKvLEE
0+87dy99947H/rQh1qMvbGIaC3btm3bpm3L008/nZEjR7YYYYr6ury4c+9KF86EMfyk033ZTjjz8+jz32WPr27ds8p7K
ycoecb+bII4/MuHHjctFFF+1Q9AAA7cNlMgBAu+rdu3fGjRuXOXPmZMOGDTtM/9//sf4+WVV9o/VJLvf//7WbduXU455
ZQ3nXP44Ydn+PDhmTZt2nv6rKuvvjp33313li5d2iIbCPDc/PDDc/nll+eQQw7J66+
/ngULFmTu3Ll5/PHH39P7//GPf8yKFStajjPXo0SP7779/kmTjxoo1paGjI66+/nt/97ne5/fbbc+211+aLX/xij
jrqqLd878mTJ+fkk0/O1772tXzwgx9M8M8qfLaxoaGnaY27dv31RW7vj/PQ0d0jQTJkzI7Nmz3+UeAgDvhTIEAGh
3++23X5YvX55p06K1/5Sl566aV84AMfyPDhwzN37tz3/P5z4+kyYO33pgbb7wxV
VVV6dOnT4YPH57bbrstJ5988tu+97HHHptPpBgwZl2rRpuf7665MkjY2N6dev3w5zX3rppTf9yt3LL788t/9122zv
dNQCgFVSU3sldzAAAAADe59wzBAAAACgUZQgAAABQKMoQAAAAoFCUIQAAAEChKEMAAACAQlGGAAAAAIWiDAEAA
AAKRRkCAAAAFIoyBAAAACgUZQgAAABQKMoQAAAAoFCUIQAAAECh/P8BfSwCaWteM2MAAAAASUVORK5CYII=\n"
          },
          "metadata": {}
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "#### Employed_days"
      ],
      "metadata": {
        "id": "8I2xJaMeHOma"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "analyse_numerical_column(\"Employed_years\")"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 542
        },
        "id": "eDUyk-neHSJb",
        "outputId": "98c1670d-c81a-4345-ed74-2236679713e1"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 1300x600 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAABDsAAAINCAYAAAAwQHSoAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAABYSkl
EQVR4nO3deVyVdf7//+dhR+RgqGwBaZmJJZlLylROi4lmpWVm4162GWbGTIvfzKVsYpqZaprRdL6f1KZ0LD9Nm
pWUWS4lMohh7mVpWAqKCojKATnX7w9/5/pyBBUQzoGLx/1207e4tvf7dV3nvMHz7FpshmEYAgAAAAAAsAgfbxc
AAAAAAABQnwg7AAAAAACApRB2AAAAAAAASyHsAAAAAAAAlkLYAQAAAAAALIWwAwAAAAAAWAphBwAAAAAAsBTCD
gAAAAAAYCl+3i6gMXA6ndq/f79CQ0NCls9m8XQ4AAAAAADiDYRg6duyYYmJi50Nz7nM3CDsk7d+/X3+/X3Fxcd4uAw
AAAAAnMe+ffsUGxt7znUIOySFhoZKOi0yScn7l6sBAAAAABnKi4uVlxcnPkd/lwIOyTz0hW73U7YAQAAAAAAAEBAI
1aT209wg1IAAAAAAGAphBOAAAAAAAMBSCDsAAAAAAIClEHYAAAAAAABLaTRhR1pammw0tVUpammw2myZNmmTOKy0tVUpKilq
3bq2WLVtqyJAhys/Pd9suNzdXAwcOVIsWLRQREaEaGnnnnpKp06d8nD1AAAAAACgsWgUYUdWVpbmzp2rxMREEt/lPP
vmkli9friVLlmjXmjXav3+/7r77bn5vv/22FixYxYoKlq0fv+5Fv/22FixYoKlTp3p6FwAAAAAQCNhMwzD8GY
BJSUl6tatm2bPq2ZM2eqa9euev3v3111VVUKS2bdtq0aJFuueeVEJCgjIyMt7d2+twWCWRCt99+u/bv3
6/IyEhJ0pw5c/c+/oKDDggqFENxcXFCgsLU1FRUZN69GxmbDWeOjZRcHE/zdg/+wOQS8jF/Yk+wtWLFCt99+u/bv3
/D9VOVikpKRo4cKD69u2rmTNnmv0OzIK/bb6ZeQSXRt0eEFZL4+aFhT47fLdpEHtlsSXRYmB6yooZLyRdLY4
gRPdcsiXkD5DS6Mb0z1n51GBCLEdnnAYwK2KUfNhj7dpkrKByYuJIUvwBMiu3XHcSdfEnMEVUpq5Zd3pgbk
1xzTbV9Ohw00Rw0c7q4uLgB9qxh5ebmql0nBJ08ecKj/QYHt08ecKj/QYHt9D0nTsIPAAAAAAAjZZXw47FixdrO6ZNysrKqr
sLy9PAQEBatWq9dv8yMhI5eXl15KnmR1BzFZKkQVFVX16SyuaUUKzpvJhk2NUU69SkjpYvLGc6iwKdQkN
WnWrFmaNWvvWWbe55JJL9OmnzZwZkOGcQQjknmmnBfFjuidfHrbjlPVE
GcstHVWfJ+5MZl2KB9NgQ9Vx+TVu51UXpbtqJTRVPqEx5MeoxWBW4LDpg2ubjBdqgg94ZfwUX7uVvzehjwEVqJ3
YWYuxvcMFiKIRtNwCSUlg2w++ALsaakpiOgjyNyjaTwcAy0PBbhnhEv4acR5lKvsMyc1bPr+ILfhGAwCtpvu4Y6Yw
CKyuh3aRvzSgaOJkmGFJBg5SHeZt1pYWTWpkf6wrW5OEWqVRgKhRFsg+sdMeDL6lOcnL2vlIfjxvt1wRWMqR7
AAAAAAAqcbTSYdA4uyZGnbntfvbIggUGA9YdxPRQVTm2Q9KQDQwmHQnAFXvgICGQKMbFfgUbLUEe7YjwUEg2D2Sh
I5CmfOVpQ9t4sQ3l2hBwXBm5HqaICaWiVnYRjFAQKlGRqKbGJqlQBVCODBHYhIOMtZlhUXtd9SnEZ4tCC8IRcPJH
2qr9jYQXLb0sEwkgTN2+vzgxv98ZG0SIDKJwTN5SEVMCd1bDuNxlr3Nz3yhn8VYEwwNLBTSzgdpzf3gNAhpfbH
QGSBDB9Pf4bQDMQaBqtQCrJf38HxUUkA84ZXnfyiyO3TxkGYEfLxC0TKTrFgFmvBhHQFhHk7YWM2ZlKm7pQJGP
AzzuYdJ43BQYXyufaLCCBqw6tpV6HI0wR11LF1P+zrLHZGGxzn8QQnyv6SDLJIxJPShWwRt73BK+kzWEHATiMkv
Mr9JvFfqqADs95aXYudKJrj3UCg9WqEkBh8tWFXSkqIrw1KQgqTVwEyXnC+QC5ELKTo6E4Kxir0sK2ZDSqFxP7J
7yp9XUqDcugSJa5E7F9tKBqCB1BtukjuoG39NfxyEXuQLfUICa0K6H1xXZ1r/SvQ+PqgyhYgcPK5hRuTvCiI1HPV
4l+/QfjYGEsg+9M7xsdxttqjYyQdVulEKDJxeJb1hXGwAEJ/yQwDpt4HpQJpGEoIlcGA9NqyBG81qHgVpzBEZ
pHN7L9TtPAvVWQBBQrOJhVyKAl1LInwWlfzu/o9pC4VuPmkUjtGA2kbfeY4HtzoRdWUzOAjXVWZXUPfJ5EW
XYGIhwuMAKZSJ6BmZMkH5CVtO6DQ7eXDM/41t90F7M4WfPbhbiiWNzU9U6mOa4jBfM9B3JqaIsmjSQYQ57BLh
CHAZ8nJVPwCo/G55sxOX12xwJjDcQ3cBaidbdPpkJl18yz1PrDdbNRYDVgbTLBTVcvHCibxpC/x62mgW6Ks9iSk4Tvk
T9AGDUhdSBT53QeBxMTXJLsCbIYu1ZE9nBDDqfbMQz+F2LSNZJT1cmdkTMZ7pb2NjU99sTnHlHMXVIyVV1IHEp2O
yK8cIahT3oLAWQ6PBGHCEmoCdRsGSOAAAAAAWgBt5tYR7oSU/PNXRBOKgN+mn9HO2Ww1PrxScgO02fimrg
vMfEIqa1ltfLtmyJZB1QwHEWFiLyMDyW/lnpgxnRWFyG6VFHwq4BGHTnpg5HA1A+4lJAVZcdODJxpGcpc9FvO
7yVcPmOqNwwQaWjEkL+jZ2vVs5NxYRrbRNY39D42NU7iCqWD76kPyMArkGTsXrg9dx6YHJm0tdNFxY8vkm2jeB9a
YIPHttOcudERAgQBtfpfW7W+JYD4v1Ey99DIVxTYDGXG5VjnYg3dXuyLH0fLNSlhWgYRr4YJK5Wrk4bq4Mad3zv3W
/EEd37vN8JNM9FRJU0V4aOCeePpr6ymW1JvqrxUOtvnn5mhiyKe8dxwCf6KJP8pSeh2T/DYnShyc1h5Ecf1QGrY3g
u/OTGzN6ijG1U0NZ5RxH16klv5h29oIcJBPyRfeK/9wXP0RtVNJaJxZCNFmi8N9cVCeNu7I8AAJ/rFTEJhgPP/3
3JG3JbEQehVJ8Y4Ib76O0+dZhJqKeNv9KPSFddjbjIPUH/TXqDzvUqTO0jZYNsqk1jjUyxUQH7v4gJ/V0g0Bv
mtSyCUq/xFgc/ijSvjUQlmj62kmP2N6FAzQcrsnqFCfTLZJ+kE/iqeIfIZTy2YGw5Xu1+ToXIzwV/VMf0VEfD0
kvgsHWA1wuBIePqDdGzMxIwTzp/3WXRHcDpfIMFfIOL/rTv2Spxjm06ckCnswC/CFnWIBU7/zbUFdP+7zuWGtgjCD1
E1f0Z1Vw3WZbkGndnx4iI+WvE+7Dbg1IG/rzu7CbCfFBv4rT+6k/L8rJSjoerL49PMM/xwQAc4ymU0j1Tp8Ctu5P4
qMbFlc3WIP1sC+36exW4R3ZTKgeUT4vSMXtMQVYoSJuEbEQyeJ/dWMf9rSbC/RWyZV0gK1IWcEBHEQz4H27d
```

AAAlkLYAQAAAAAALIWwAwAAAAAAWAphBwAAAAAAsBTCDgAAAAAAYCmEHQAAAAAAwFIIOwAAAAAAgKUQdgAAAAA
AAEsh7AAAAAAAAJZC2AEAAAAAACyFsAMAAAAAAFgKYQcAAAAAALAUwg4AAAAAAGAphBOAAAAAAMBS6hR2XHrpp
Tp8+HCV+YWFhbrOOksvuCgAAAAAAIC6qlPYsXfvXlVUVFSZ73A49Ouvv15wUQAAAAAAAHXlV5uVP/roI/Pnzz7
7TGFhYeZORUWFVq1apXbt2tVbcQAAAAAAALVVq7Bj8ODBkiSbzaYxY8a4LfP391e7du3017/+td6KAwAAAAAq
K1ahR1Op1OS1L59e2VlZalNmzYNUhQAAAAAAEBd1SrscNmzZO991wEAAAAAAFAv6hR2SNKqVau0atUqHTx4ODz
jw2XevHkXXBgAAAAAAEBd1CnsmDFjhl544QX16NFDOdHRstls9VOXAAAAAAABAndQp7JgzZ44WLFigUaNG1Xc9A
AAAAAAAF8SnLhuVlZXpN7/5TX3XAgAAAAAAcMHqFHY8+OCDWrRoUX3XAgAAAAAAcMHqdBlLaWmp/vnPf+qLL75
QYmKi/P393Za/+uqr9VIcAAAAAABAbdUp7Pjuu+/UtWtXSdLWrVvdlnGzUgAAAAAA4E11Cju++uqr+q4DAAAAA
ACgXtTpnhOAAAAAAAACNVZ3O7LjpppvOebnKl19+WeeCAAAAAAAAALkSdwg7X/TpcysvLlZOTo61bt2rMmDH1URc
AAAAAAECd1CnseO2116qdP336dJWUlFxQQQAAAAAAAABeiXu/ZMXLkSM2bN68+mwQAAAAAAAKiVeg07MjIyFBQUV
OPlX375ZfXs2VOhoaGKiIjQ4MGDtWvXLrd1SktLLZKSotatW6tly5YaMmSI8vPz3dbJzc3VwIED1aJFCOVEROi
pp57SqVOn6mWfAAAAAABAO1Kny1juvvtut2nDMHTgwAFtCwGWVV2q6hDMHTgwAFt3LhRzz//fI3bWbNmjVVJSUtSzZO+dOnVK/+f//B/16
9dP27dvVOhIiCTpySef1CeffKIlS5YoLCxMEyZMMONN13cq1vvvvlGk1RRUaGBAwcqKipK69ev14EDBzR69Gj5+/v
rj3/8Y1l2DwAAAAAANGF1CjvCwsLcpn18fHTFFVfohRdeUL9+/WrcTnp6utv0ggULFBERoezcXPp0OdFRUV66
623tGjRItl8882SpPnz5ysh+IUEbNmxQ79699Xp2nnnn5yshIUEbNmxQ79699fnnn2v79+/6uv79uw9K7uPV6
AAAAAAAJqoOoUd8+fPr+86SJElFRUWSpPDwcElSdna2ysvL1bdv3+Xnn/G7vH+Hh8guBo+xw8pKq1hKpQ5ESfwFj/EuOpdfbW+k0H1PW2ysvL1bdvX3OdTp06KT4+XhkZGGerdu7cyMjLUpUsXRUZGm
uskJydr/Pjx2rZtm6655poaq/TgcDjkcDn06uLi4QfYHAAAAAAB4Xp3CDpfs7Gzt2LFDknTl1TAAAAAAAADn31lVdWGyzUlNPp1KR
Jk3TdddfpqquukiTl5eUpICBArVq1cls3MjJSeXl155jjqVgw7Xctey6rz88suaMWNGnWsFAAAAAACNV53CjoMHD
+q+++7T6tWrzSCisLBSBQN91OkxYvXqy2bdvWus2UlBRt3bpVX3/9dV1KqpXJycl2ZcXFyD9wsAAAA
AABpenZ7G8vjjj+vYsWPatm2bjhw5GPHHH+urr75SbGysOT8qKkplZWUqLCx0m5+fn6+oqChznTO/le9aPtc6oaGhCgjZ
CxOWz8/P19RUHmOmc+ncU17VrnTIGBgbLb7W4vAAAAAABgDXUKO9LTOzV79mwlJCSY8Spm2pKOkpNZcsWHTx4OFxn5cqVstvvt6
ty5c112DwAAAAAANGFluozF6XTK39+/ynx/f/f385nc4at5OSkqJFixZp2bJlCg0NNe+xYIWpuDgYIWFhWncuaHF
KTU1VeHi47Ha7n/8cSUlJal3796SpX+qlz584aNWqUXnnlFeXl5WnKlClKSUlRYGBgbXXPAAAAAAAOYXU6s
+Pmm2/WE088/of3795vzffffaff+fVz31Vz355J065ZZbatzOm2LZTBM2++qaKiIt14zg4eqzKt3AJ1OmiZ2GG8BZhxUL6t2++JlFhNvZ0xjgvnrXr1QeKAOZ8
+fRQVFaX//Oc/5nJfX199/PHH8vX1VVJSkkaOHKnRo0frhRde8OcWSAX8K2p7n3/dC/xG4htx
/bct2+frrrqKr377rs1bscwjPOuExQUpFmzZmnWrFlnXeeSy7Rp59+WuN+AQAAAACAddUp7IiLi9OmTZvOxRd
faOfOnZKkhIQE9e3bt16LAwAAAAAqK1aXcby5ZdfqnPnzioouLpbNZt0Ott96qxx9/XI8/XI8//rh69uypK6+8UuvWr
WuoWgEAAAAAAM6rVmHHHH66rnH76rrnneGDqqRQAAAAAACAmqhV2BEZGGan169dr/P/Pj
xmjx5sgzDkCTYbDYlJydrlqxZioyMbJBCCAQAAAAAAqJWYYckXLLJJJfr0009Ohr7d69W4Zh6PLLL9dF13UE
PUBAAAAAADUSq3DDpeLLrpIPXv2rM9aAAAAAAAALpiPPtwsAAAAAAACoT4QdAAAAAADAUgg7AAAAAAACApRB2AAA
AAAAASyHsAAAAAAAAlkLYAQAAAAAALIWwAwAAAAAAWIqftwsAGpvc3FwVFBR4tMooPj7AAAAAgFURd
gCV50ObmqlOnBJO8ecKj/QYHt9DOnTsIPAAAAACgHhB2AJUUFBFBTo5ohMT6ZAUQFBTo5MkT6vXANNmj2nh01JAJMkT6vXANNmj23mkz+IDe5U5b4YKCgoIOwA
AAACgHhB2ANWwR7XwxR7dz6QCgHhB2AANWwR7dTePwV3i4DAAAAAFAHhB1oEnbs2bs2GGpfgAAAAADYewA43ayaLDkmwaOXKkR/std5R5tD8AA
AAAQPOh7ECjVn7imCrOz/9wfe2ZGjrR//Uq7OnGrwvAAAAAEDDIOxAxAk9AyIt4j99AyIt4j99AoPrC3wfsAAAA
AADQsH28XAAAAAAAAUJ8IOwAAAAAAgKUQdgAAAAAAAAEsh7AAAAAAAAJZC2AEAAAAAACyFp7EAjcSOHTs+bN
mOUHx/vOT4BAAAAwBMIOwAvO1llJJNIO0e09Gi/wcEttHPnDgIPHnDgIPAAAAJZD2AF4WfmJY+4WfmJY5IMdR3+jNq27+SRPos
P7FXmvBkqKCgg7AAAAABgYQdQCQKBgg7AAAAAACApXg17Fi7dq1uv+N2xcTEyGaz6cMPP3RbbhiGpk6dqujoaAgiODlZ2drJTZ2PmkKCQlRcnKynn36ap06dctjn8uXL1a1bNwUGBqurq0sLFy5swMoAAAAAAMqvWMmZnVW1stuMbM5nwe6lu9aOw6lz2PQlpgKjVoLdlzaB6oNiuvD7bMLdDKqUlNPPVa5xfL70+lzf/HOrpyZ0tukuniNaM1BQcjWiq0+aw0GJrfSKfKuJqcSwT5VWnJzNkAqbtjcz9F2+Kvq2VCzXTvjNkaLD9Kll0DcCnKfa5o4U0RA0TVbfYwt4o7ZT0nsgC8KLyS+4a2l4aBfdhWc/WPJ8UMV4rdM+1lSc0M5v8r3ubXC18UNiM2gXsaedk8UPM8LAJ4n+qBaysL04qMtrCqJUjvkWs3Mj2azFfxAYqQcjYu60vdnRhnIjThsdRk+4i5eMDUzMvzKH
KTU1VeHi47Ha7Hn/8cSUlJal3796SpH79+qlz584aNWqUXnnlFeXl5WnKlClKSUlRYGBgbXXYPAAAAAAAOYXU6s
+Pmm2/WE088of3795vzfv31Vz355JO65ZZbatzOm2++qaKiIt14442Kjo42X++99565zmuvvaabbrpJQ4YM0a233qoXXnhBcrfdd1/NiHu+++q1atVKn376qeLi4vTVV1/p8ssv16OPPqru3bu37+++1b9O2pWVGxQeHt5gA0LYA3DlfFxcVqKiqSfn6+Pfffe9q4cWPd8P3338vX19f3X333Tp+/LjS0tK0YMGC+vmHBQAwj1IevbwOzNHPA5DC8+HF3g+BsHyNaJdSS1xU9XRn3V1l8pYa+Im7ZGc7O8ryBf/6Iz/3N4l8a+j/2R+wG9ADTaVJnezSQUuTzBNWWmZC1gjBQmhPf/nA3rGWUS8NeqbNN8RuE7KuhG6VrsJXw6uZk5Ks/BAADkiPkpF9JlNL80CqBmFbXKfpudtFpXvT4ThORHYoZgEBFEHMNxsNFZ2kdl4XaX+nC1O/7/LpR8wZlPkWTSfu6iB2A/3MD+3S4YG/kLZsNP2mm+nc0HjpbA3+qYz2cUGd4VnjL6NeX8nLS6jjWsuhRFUxeCfJ0dzoWp8J7ZmptO2qb5rYaAyKNVVnSbt9t9ICOAk2tGqPExNLwPjrLsP+ArfcWr+bf8urwq5MNYodP3iZw6b2X+TrKvyZ3MRg0MLKePs5KjOXpI2c8HLN9FmUtE+p8eajsR+4kkOenLeP8xzrNnufSWU7f43k3/qPKwDhWqOo2SXaZnLlpdz5Lfb9q7cWxVRQVmH2t+JmwL7mtyTzKu8U

```
ow46GNHnyZKWmpprTxcXFiouL82JFgHe0jIhXePwV3i6jweTm5qqgoMBj/XGpDgAAANA4NNqwIyoqSpKUn5+v6
Ohoc35+fr66du1qrnPw4EG37U6dOqUjR46Y21cnMDBQgYGB9V80gEYjNzdXnTol6OTJEx7rkOt1AAAAgMahOYY
d7du3V1RUlFatWmWGG8XFxcrMzNT48eMlSUlJSSosLFR2dra6d+8uSfryyy/ldDrVq1cvb5UOoBEoKCjQyZMn1
OuBabJHt2vw/rhUBwAAAGg8vBp2lJSUaPfu3eb0nj17/DwcMXHx2vSpEmaOXOmLr/8crVv317PP/+8YmJ
iNHjwYElSQkKC+vfvr4ceekhz5sxReXm5JkyYoPvuu48nsQCQJNmj21n6Uh0AAAAAVXk17Ni4caNuuukmc9p1H
40xY8ZowYIFevrpp3X8+HE9/PDDKiws1PXXX6/O9HQFBQWZ2yxcuFATJkzQLbfcIh8fHw0ZMkRvvPGGx/cFAAA
AAAAODl4NO2688UYZhnTabTS+88IJeeOGFs64THh6uRYsWNUR5AAAAAACgCfLxdgEAAAAAD1ibADAAAAA
ABYSqN9GgsA69mxY4cl+wIAAADQuBB2AGhwJ4sOS7Jp5MiRHu+73FHm8T4BAAAAeBdhB4AGV37imCRDXYc/o7b
tO3mkzwNbMrT1o3/q1KlTHukPAAAAQONB2AHAY1pGxCs8/gqP9FV8YK9H+gEAAADQ+HCDUgAAAAAAYCmEHQAAA
AAAwFIIOwAAAAAAgKUQdgAAAAAAAEsh7AAAAAAAAJZC2AEAAAAAACyFsAMAAAAAAFgKYQcAAAAAALAUwg4AAAA
AAGAphBOAAAAAAMBSCDsAAAAAAIClEHYAAAAAAABLIewAAAAAAACWQtgBAAAAAAAshbADAAAAAABYCmEHAAAAA
ACwFD9vFwAAVrJjxw6P9temTvFx8d7tE8AAACgsSPsAIB6cLLosCSbRo4c4c6dF+g4NbaOf0HZYOPHJzc1VQUOD
RPgmRAAAAmjbCDgCoB+Unjkky1HX4M2rbvpNH+iw+sFeZ82aooKDAsl/Mc3Nz1alalTgk6ePOHRfptDiAQAAGBlh
BOAUI9aRsQrPP4Kj/Zp5UtnCgoKdPLkCfV6YJrs0e080mdzCJEAAACsjrADAJqo5nTpjD26ncdCAHjA0kcdDJAAAADRdhBO
AOERx6QwAAABQPcIOAGjirHzpjKcv0QAtf0lDvKPNofAAAAmjbCDgBAjXn60pkSm+mdDWzK09aN/6
tSpUw3eFwAAAKyDsAMAUGueunmNxc2uk8qv0fV0B6Q4HAoMDPTY3i/mucAEGhcn55chAAAGhcCDsAAGhmcnNz1al
Tgk6ePOHZjmO2yTA81pO3nhwEAAAaB8IOAACq4embo3ryLISCggkdPHlCvR6YJnt004/06bokyVOXQPHkIAAAm
jfCDgAAKvHWTVi9cRkUAAAMOPYQcAAJV4+ias0v87C2HdunVKSEhQ9SfQMKp/AJ2lQvI8OpHlxqZvN57hD7y/2jzJ+IdjEdhz5N8sg
3CQAAVRF2AABQDU+egcAjfRu0J4Odawc06J57hqq09KTH+pS4WScAACCFsAMAACtxbb4SD3+X/ekQ/Ve+PT9wNwXZGPl05O9DsVN8+FD+t1mq/6g5R+fgSWcCGq
ajBkGtzajJWo3Y/v37dfHFF2v9+vVKSkoy5z/99/99NNas2aNNMjMzq2wzffgxw5NlAgAAAACAerBv3z7Fxp4
7xGzyZ3bUxeT5pn/E3OhiouLFRcXp3379slut3u2m7j72mlu3HKBZYAzQd4Me/ny6FtDa51EOtq
NGmTRv5+voqPz/vqPz/fbX5+fr6ioqKq3SsvIKHCc29XiYZSBIblN1uZ1AAHsa4AzQd4F
mOudgzD0LFjxxQTc/4zmZP82NGmTRv5+voqPz/fbX5+fr6ioqKq3SYwMLDKNV+tWrvqqBIblN1uZ1AAHsa4Azy
LMQd4HuMO8CzGXM2FhYXVaD2fBq6jwQUEBKh79+5atWqVqOc/pdGrVqlVul7UAAAAAAIDmocmf2SFJqampGjNmjj
Hr06KFrr71Wr7/+uo4fP24+nQUAAAAAADQflgg77r33Xh06dEiTJk1SXl6eunTpooyFC6u96ejZ7HYb0rJly2SxWKyyhSyGXbHUfdW8mT8SYnKPJmpoAnp0oSM7+a9wMTXY9xLCGNBmQP
AwAAAACWQtjRRM2aNYuLi4st9Ah0Xqpf/+97/eLgmwjLVr1+qO0+5QTEyMbMbDabli5dy6brNcVHlMjlV9Fnbd29FrxDr5NPV2jNbsm/SF2m7SBOD+eP5vA7EfMw1j3U6Xb+hyMIVa9Cg==
OVt++ffXDDz94p1jAAl5++WX17NlToaGhioiIOODBg7Vr1y63dUpLS5WSkqLWrVurZcuWGjJkiPLz871UMdC0v
fnmm0pMTJTdbpfdbldSUpJWrFhhLme8AQ0rLS1NNptNkyZNMucx7uofYUcT9N577577yk1NVXTpk3Tpk2bdPXVVys
5OVkHDx70dmmAJRw/flxXX321Zs2aVe3yzS2Fs2aVe3yV155RW+88YbmzJmjMxMhYSEKDk5WaWlpR6uFLCGNWvWKCUlRRs2b
NDKlStVXl6ufv366fjx4+Y6Tz75pJYvX66///3vWrNmjQ4cOKC7777bi1UDTVtsbKxeeeUV5eTkaOPPPPuuuWMGHJp
GDdK2bdskMd6AhpSVlaW5c+cqMTHRbT7jrv7xaNkmqFevXurZs6f+9Nf/lrt/aVhiLmcN51Prpd4O8uSAuraWgPgt
Gw2ffjhxo8eLCk02d1xMTE6Pe//SJKKioJpo+ecowcTdh3blxMTE6Pe//SJKKioUGRmpBQsW6L777777
vNitYAlHDpOSBEREvqzZo369u2roqIitW3bVosWLdLQoUMlSTt37lSnTp2UmZmp3r17e7lioGkLDw/Xa6+9pqFD
h6r/7uvXXX+fvXAPhzI4mpqysEsAIDTJ6ohdlxkx1t2ZrlWhs0U7ZZQlR2rWaqH3wp7HbkjbkVvv7mAxw3Mr7h+P2b9E5
s8Tk6tCgybgAHPkIUYZlAh9+zKuZmFsMozkUTQhqp
itW3bVosWLdI999zj1dtFi+ZUWD0uVP2C+IIN/vnsDj8dWSJCAWTHlj4DKUoupvh0YcqN2GxpNeh5kn+hqp8jrO
l276vXXX+fvXAPhzI4mpqysTNnZ2erbt685z8fHR6Y+Ti5ehEM7+tszrGiG2Qr9G3KQDDBiFZ28MZi/PPl88HrjAY
VSTr95UuSsrOzVV5e7jbuOnXqpHXr1ikzM9PbVULNpmt1XY2DnK3m6o4ktK4QWtc9vH3PtAP9P+FEWx+d3R97EhXN
YKCAlVUVCgyMtKtPTIyUnl5eV6qCmg+nE6nJk2apOuuu04dO3aUJOXl5ocgICBHV1eXWzvXJlB3QCrmn66kH8MFRAJC4eqX
3QN1t2bJFSUlJp6Ki0sOt0tVcuWLfXhx+sXatGmTsYyqizaKzu3aqvv
/7a26UAlnbFFVcoTio6ycb5mb2qFWFZ1vYiFaLTzl2l8cLiaNGBqrLUFVIaU7t3ydHYKp9djQ+vp
WuTNvfn6+oqKivFQVwOHy4xhljEKh/EyZM0Mcff/EyJN3vvmZWGJpkMOo8Ebxf5yBDV3/u3GDZTAvD+1Wqc8+Z
66XX35ZV199tf72t78x3oAGGkJ2drYNHD6pbt27y8/OTn5+f1q5dqzfeeEN+fn6KjIxk7AENJDo6WldeeaVbW0JC
qVavMeU6nU6tWrVJSUpKXqgKsTzDMDRhwgQtWbJEWVlZat++fZUr5nS6GMmQaPCT/xMt7GCzZGzA3+ijk6KHbj/eFTH8jn0iLjtnd2KwrrQH9cfNbf46/EAu6oMrYkPbCYwDaNGSgE6h+oP8+9K5wSHn8Yu8Dp1bbjMbRxn3HwEY9c3z3V0Oa85aTB3ZkXNx8mFiIo9p+Ix6ivHunWf+Lp+tvW3t6XyVF3Z+7s1xKjnl7hlF+N+HsT0GfrNGajYq//3y8qPzFmXPd2H9LU43zSflgS1WvWK7z/3p18/Kz56HAbKFQHvuQ7U4e21NZSybhO5aRaE7p
qVavMeU6nU6tWrVJSUpKXqgKsTzDMDRhwgQtWbJEWVlZat++fZUr5nS6GMmQaPCT/xMt7GCzZGzA3+ijk6KHbj/eFTH8jn0iLjtnd2KwrrQH9cfNbf46/EAu6oMrYkPbCYwDaNGSgE6h+oP8+9K5zPCk5BDL8QB1YTU3+dF8wkTEz++L0J8wDYcWPL3hKBaL5Qpz+/4cvZ34L6RsXTgGJ9NffFuUq37+bh9jeOQE8/FepI/eW+bzAOtrKJDlOVYE0iRDF4wsQe1zmvfZSgGbjUqUQ7r0uDvPd+P6WnOAaDlQ+T7yBwDzGdw3yJ7UQT2WCUm/O/CwOkjpiBYkGvXqnD9bcLvb4O3m/RHWVX/nQgCAVKmgnSCRGPBHaowX05mZzQIvfjsbN3fz/V7u+U5h22HjI39w9x4oZT95vXAxlbdlf6/UnJr3b3ClL6HrndI3NRB8PDnJp0Z/2Y8+61KR6+ziXKMPbm3mdRFjoTp7OekTQw/n0cl9p+2rQ8qd7V0k37v5h/m6DEDGVLBnsJcFulmHr22x8d8YSu4/M//7Izl6yqS+K8/bbGsmiw4ttJmfS+d3/mtFzm9Zq0R9hEy3ZcC7+JzM1+TP/nJdfXkHVoJFzz6Mo9+Bc+X8QBQ42UPdeMk2xVf29f+Wmvj88cm/rtZmFUYH0k/mz7WMtZ+uLxqzyHcfmOHtxxqVbS1q9Pv5n+eugVXpNGxaMWvqxWXkSbUbOLadKTpCqk+Wv7zGGanNefzsN/q83f4V7a6yT5U/UbVbxFeZVMnhmhXiKiWOpgH84jVdKlw32bL/Zz47adhMbKwGqvUqtTUutxy7K2dntUNQE8kbWaJHyr1P1Hq4BaqFIHyKA2TbOeDYUGaiMjcmdWsqzLfp+ufhjuqzDqPaUhAxnfC8iDfBqFj7pI/lTYKnDnwo2D/kHXKzoepVmvxA4EHMBvMbTDstEbXLYCKBJqs2hYRh1a4lvT6fp1Ep82W5UyQ5hK+pf5Ioo2LUs1A8a77mUEaMOB/RUEo66Lr4rlQtPYoyrp7fCEDEwLtUeqD/8SWEKYLsVgmvAA0bJR2rc9fSrzXc+jiK4rCDi62hUJr+tzdMoWr+dZ8zUm/bi4orlZ57G9t2I/YuL0X+6xuN13yXsf1a+1EOzqm8PHbtN97X7+e2YHXEEF0Fx9N1oDnUTzgffnqu9rUp8eixQIdWvuVn5hq8c3WtXFo2+QKXY0Hc3dV1oy6uhUKxxFRkdYYgD9NX6gHqPDGIhbnAK31TfGBGOKF2akqTX2aqtnQ3i1H58DV4lWGQx
```

```
9/fbdzt2rVLubm5jDugnjidTjkcDsYb0ABuueUWbdmyRTk50earR48eGjFihPkz467+cRlLE5SamqoxY8aoR48
euvbaa/X666/r+PHjuv/++71dGmAJJSUl2r17tzm9Z88e5eTkKDw8XPHx8Zo0aZJmzpypyy+/X03bt9fzzz+vm
JgY84ktAGonJSVFixYt0rJlyxQaGmpenxwWFqbg4GCFhYVp3LhxSk1NVXh4u0x2ux5//HElJSVxh3qgDiZPnqw
BAwYoPj5ex44d06JFi7R69Wp99tlnjDegAYSGhpr3oXIJCQlR69atzfmMu/pH2NEEDRs2TIc0HdLUqV0Vl5enr
l27Kj09vcoNEwHUzcaNG3XTTTeZ06mpqZKkMWPGaMGCBXr66ad1/PhxPfzwwyosLNT111+v9PR0rsEE6ujNN9+
UJN14441u8+fPn6+xY8dKkl577TX5+PhoyJAhcjgcSk501uzZsz1cKWANBw8e10jRo3XgwAGFhYUpMTFRn332m
W699VZJjDfAGxh39c9mGIbh7SIAAAAAAADqC/fsAAAAAAAAlkLYAQAAAAAALIWwAwAAAAAAWAphBwAAAAAAsBT
CDgAAAAAAYCmEHQAAAAAAwFII0wAAAAAAgKUQdgAAAI8b03asBg8e700yNH36dHXt2tXbZQAAgHpG2AEAQDM2d
uxY2Wy2Kq/+/ft7uzQAAIA68/N2AQAAwLv69++v+fPnu80LDAz0UjWozDAMVVRUyM+Pf7IBAFAbnNkBAEAzFxg
YqKioKLfXRRddJEmy2Wya03eubr/9drVo0UIJCQnKyMjQ7t27de0NNyokJES/+c1v900PP5rtuS4NmTt3ruLi4
tSiRQvde++9KioqOmsNDodDEyd0VEREhIKCgnT99dcrKytL0ukv/B06dNBf/vIXt21ycnJks9m0e/duSVJhYaE
efPBBtW3bVna7XTfffLM2b97stk1aWpoiIyMVGhqqcePGqbS0tEbHa03atfL391deXp7b/EmTJumGG24wp7/++
mvdcMMNCg40VlxcnCZ0nKjjx4+by9955x316NFDoaGhioqK0vDhw3Xw4EFz+erVq2Wz2bRixQp1795dgYGB+vr
rr7V582bddNNNCg0Nld1uV/fu3bVx48Ya1Q4AQHNE2AEAAM7pxRdf1OjRo5WTk6N0nTpp+PDheuSRRzR58mRt3
LhRhmFowoQJbtvs3r1b77//vpYvX6709HR9++23euyxx87ax9NPP60PPvhAb7/9tjZt2qQ0HTooOTlZR44ckc1
m0wMPPFDl7JP58+erT58+6tChgyRp6NChOnjwoFasWKHs7Gx169ZNt9xyi44c0SJJev/99zV9+nT98Y9/1MaNG
xUdHa3Zs2fX6Bj06dNHl156qd555x1zXnl5uRYuXKgHHnhAkvTjjz+qf//+Gj/jkiL777ju99957+vrrr92OTXl
5uV588UVt3rxZS5cu1d69ezV27Ngq/T377LNKS0vTjh07lJiYqBEjRig2NlZZWVnKzs7Ws88+K39//xrVDgBAs
2QAAIBma8yYMYavr68REhLi9nrppZcMwzAMSca UKVPM9TMyMgxJxltvvWX0+/e//20EBQWZ090mTTN8fX2NX37
5xZy3YsUKw8fHxzhw4IDZ76BBgwzDMIySkhLD39/fWLhwobl+WVmZERMTY7zyyiuGYRjGr7/+avj6+hqZmZnm8
jZt2hgLFiwwDMMw1q1bZ9jtdq00tNRt/y677DJj7ty5hmEYRlJSkvHYY4+5Le/Vq5dx9dVX1+hY/elPfzISEhL
M6Q8++MBo2bKlUVJSYhiGYYwbN854+0GH3bZZt26d4ePjY5w8ebLaNr0ysgxJxrFjxwzDMIyvvvrKkGQsXbrUb
b3Q0FBzXwEAwPlxZgcAAM3cTTfdpJycHLfXo48+ai5PTEw0f46MjJQQkdenSxW1eaWmpiouLzXnx8fG6+0KLzem
kpCQ5nU7t2rWrSv8//vijysvLdd1115nz/P39d/e2112rHjh2SpJiYGA0cOFDz5s2TJC1fvlwOhONDhw6VJG3ev
FklJSVq3bq1WrZsab727NljXmjXmKzY8c09erVy63vpKSkGh+nsWPHavvfu3dqwYYBBgwQK
3/p0Tk+V00rVnzx5JUnZ2tu644w7Fx8crNDRUv/3tbyVJubm5r1NVUPPvig+vbtq7S0NLfLhgAAQ
FXc7QoAgGYuJCTEvBSk0pUvl7DZbGed53Q6G6jjC0x588EGNGjVKr732mt732mubPn69hw4apRYsWkqSkhJFR0dr9er
VVbZr1apVvfQfERGh0+64Q/Pnz1f79u31YsWkqSSkhJFR0dr9er
```

shbADAAAAAABYCmEHAAAAAACwFMIOAAAAAABgKYQdAAAAAADAUgg7AAAAAACApRB2AAAAAAAASyHsAAAAAAAAl
kLYAQAAAAAALIWwAwAAAAAAWAphBwAAAAAAsBTCDgAAAAAAYCmEHQAAAAAAwFIIOwAAAAAAgKUQdgAAAAAAAEs
h7AAAAAAAAJZC2AEAAAAAACyFsAMAAAAAAFgKYQcAAAAAALAUwg4AAAAAAGAphBOAAAAAAMBSCDsAAAAAAIClE
HYAAAAAAABLIewAAAAAAACWQtgBAAAAAAAshbADAAAAAABYCmEHAAAAAACwFMKOJqqOtFTff/+9SktLvVOKAAA
ewd8+AABQU4QdTVRubq4efvhh5ebmersUAAA8gr99AACgpgg7AAAAAACApRB2AAAAAAAASyHsAAAAAAAAlkLYA
QAAAAAALIWwAwAAAAAAWAphBwAAAAAAsBTCDgAAAAAAYCmEHQAAAAAAwFIIOwAAAAAAgKUQdgAAAAAAAEsh7AA
AAAAAAJZC2AEAAAAAACyFsAMAAAAAAFgKYQcAAAAAALAUwg4AAAAAAGAphBOAAAAAAMBSCDsAAAAAAIClEHYAA
AAAABLIewAAAAAAACWQtgBAAAAAAAshbADAAAAAABYCmEHAAAAAACwFMIOAAAAAABgKYQdAAAAAADAUgg7AAA
AAACApRB2AAAAAAAASyHsAAAAAAAAlkLYAQAAAAAALIWwAwAAAAAAWAphBwAAAAAAsBTCDgAAAAAAYCmEHQAAA
AAAwFIIOwAAAAAAgKUQdgAAAAAAAEsh7AAAAAAAAJZC2AEAAAAAACzFz9sFAAAA1Ma7776rn376SX5+frryyiv
VsmVL7d69W4GBgWrdurVCQOPl6+urrl27qmvXrqqoqNCyZcu0f/9+RUREyGazKT8/X5GRkaqoqNND27dvvlcDjUs
WNHXXPNNfLx8dGRIOdUWFioVq1aqU2bNkpMTJSvr6/KysrMtmJiYjRoOCAFBARUqbGiokLfffedjjhw5ovDwcMX
Gxuqpp57S4cOH1bp1a7366qsKDw831920aZM+//wzTx5Ul26dNGdd96pnTt3mttteeWV2rZ+NGhZSDnKpXDj
9HKlStVXl6u+Ph49enTR1FRUVXllRTNqP6vKyYZhaPv27Qo0D1a/fv3Ur387Zx5j7XpN/atFNRUaGcnBz
l5ORIkq666irt3btXBw4ckCQoIiKixv2e2Z7r83K2bSvX1apVPKNbxZdSvX1apVKOkyPy0uny9kv89VZ12Pa3XbSnKb16lTJ
3300UfasmVLrd5vTzziz/ssvvvlz/8z//o19++UWxbF65JFHFBwcXG+fvTP7dn0+nE6n7Ha7tVPZSHVpB2n33XfdbPRKbX39XeOp2dnxj7p/f4j4c9wYOHCL
pSc31+Vk5c9uGOM4N2XZD1tvUNZdjYzMMw/B2Ed5WXFyssLAwfRUVyW63e7ucGn+/18MMMP65///Kc6duzok
b60d75TzpA2DdpXU+JzvEAh2z/iuDQCrvfCE+MBgPc88cQT2rx5WXNMGRV1IUKhz1uUP122WXasGG
DKioqzPm+vr4aOnSbFlpqD0pZWZZaO5Z2zzfDwcE2aNElpaWk6k6ceLEO5d1fdvmviTPXjYqKOmOPPPaY+f
fpUu/6cOXOOZMmSc7qxtVFLfi5vsvtvbE3qMftsdZPNBXS2+a/N1tm5tVfs7cPP67NSk9PvV2Fh4dVhT2hPmv
KtjV9T2vaf01dyHGtbtvKwcy5nO/NBTjbhMmdWXDyPxmJpQyb9mWE3EhTrQjuTRRDDAAOIDDCw8P
29TXG67t2T7ZpFc3t2HBmhzizo6Z9cQaD087saDw4swOwtpMnT2rAgEoWrXXs2DGVlZXJbreruLjVoAJbreruLjYXNarV
y99++23Ki8vl2EY8vPzk4+Pj8rKFG/jwoFz//OnZs6d2796toePqmfPnsrOzpbT6VRAQIACw8P
1zjvv6Pnnn9eGDRvk7++++v1q1ba+HChWYYc0rUKQ0d0lTFxcX6+OOPdf/99VSSy/VzjkzVVhYqLvvvuS9NFHH
yktLU179uzRu+++q5KSEg0aNeS1L1Jl5y1dulQ2m0OOjRoxQ+/bt5VpTGZjUKaddSTrMvLf2TSbQ/Ah59fgQpe7vdXQ9
c/Mza/j444/ldDrvVv39/c7969eql2IyDrgwQHa7XQsa7XXsXLtTttc99ubjtjxgwNG
zZMxcXFWr58uYPH67CwkJFRrc9m5vQFZvaXVgVGjBhh7nPlAMLpdFbbbb3XOlo5r/sGDB+Xn56cPPP/qQd955pP/xQd955p0JDQxU
YGKiysjKdOHFCF110kd5++23dd999K1Okkd5++23dd999XXXquff/65Sr+u9o4ePapu3brppZdeUkmVtyOp167rnntGnTJvOzI
smsa8aMGRo1apT588811VWVqagoCD961//OrRp07RnzX63n8+337U9HjU5rtVt65rXvn17SdJPP/2kQ4cOSZK
uvfZa2Ww27d27V1+++++64Mw9A999xT7fvtCWfW73A4NGDAAPn7+2v58uVl+84HfAH9Mwt
GLFCgUHB9foGJ2v76NHj+qaa67R6R3r17zTokmZ+PVq1a6dJLLzWPWV2OT+Xxt2TJErcQtvLvLvlRUrVpzzkpb6/qy
cue1PP/OkSRc8xuu7dk+2aRVWOTa1+e7eLO/Z4XA45HA4zOnK/K/an7/HH+Ubz6rgDUtWrTIbfrw4cPmz
/3799f7779f9vbGxvC2Clt+sNbHHgB3o1Oq/W+qmYyjmsV1fvRdxF6L6y+KRfvnLVk31UN16bTDJ7KrbcK
tW3XxxdSMbgtd+kIvvWmpJMIdRlGaUaV8JVXWaC9QOKM+u7+TF70t7+Th+rgRAcr+g
BihlJQUffddd277ExgYKD8+/P3377NfpyL9pLM/bj44ov15+dn1iFJv/3tbbJb8+CAwN14MABs9b
KfbtqX7ZsmSoqKjRu3Dh98/skn5r6r6PHDlSAQEB5n598sknAQEB5n598sknGjdunP76ML76ogH/qLf/
Pryo3vvv/Nw0pe3puAJfgOqDbhDv7eeuRlOfjd9eeWX1VvmeSlvaRT5HTT9FKbSRNWqQ37kWKPmIhiyzpxpHm8y8/+8zvDjG+
ObbrZ/++LrfRLPMrA1PcFzGUZ9/HmwUiU15WRkWHupyS
zrq1bt7r9nJ+fb/a/3t8ajjJca1u28rzJC+JklJcVcf9SoUeY8VZ1+l3xW9fSTJFzL5TIOvcTFDuSoneNNf4I3+S+1h/G
oKCzH1ftmyZDh48cnU+vevVUrV67U3LzlzNVnJEm1k+xV17dO0vTbsGGD23F5F6p7J7Gk1I7h2N0xe4p3+y4GJQbvPnp9jrNowZPBhGP4+y9Pz1Oke3GYiYffCnp8mqvBpLCt1
xPjNl35f2Z1Ups2+u2+/Mu4svvlh79+6Tsr7r9PSskpIq7R85csRMyHhQV9++aWOHDnitkZ5SUpIk6ZdfpEk8/+
eu0KZcePGuc0/c/vSOlK3+a71XMrKyqrs2ZXVXGGGGHWe2J0nHjh2rts3K6+7fv9+d3f+/s/9133zXnu9Z17df+/fvdL

q+p3MbZaj6zrepqr0xs7Zy5nesYu953V42udV3TrmN15vaVp6urufK8M9fNyMio8r0rv8p9n/lzXVzIca1u230
1V916Zx5XTzqzVtd7ftttt7nNd31+7733Xq1cudJcz6Wmn73q+pZOB4aV2znzZ9fyuh6fyu0v0pXH37nU92e1u
m3r2v751Nfvj4Zu0yqa47FplmHH5MmTzf/bIp0+syMuLs6LFdXdc889p0suuaRB+/j555/5EokmwRPjAYDnLVq
0SKtXr6522ZlfBFxfQFwKCgrMn3/99dcq4YjrDAjp9OUkZwoPDzfb3LFjhzmvMtcX39jYWG3cuFF79uzRlVdea
V5u89Zbb6l79+7as2dPtdsHBQW5zXet5+I6hb3yvu3atcutxjOFhoZW22bldV3HIiMjw+24u0p37VdMTIz585l
tVG7/yiuvrFLH2fb5TGdr58ztXMfY9b6fWZdr2nWszty+8nR1Vc+9meuW7lG18+u/sLDw932tab7fTYXclyr2
/Zsn6Oz5525XxeyD3V1Zv2u9/zTTz/Vww8/bNbr+sy6zuyKjY11a6cu7OH1dV2BWeXjWPlYuZbX9fhUHn+uS8g
qqzz+alJzfX1Wqtu2ru2fT339/mjoNq2iOR6bZhl2BAYGVvnHUFN1ySWXcI8C4P/HeACs6ZlnnnEL0yrfsyM9P
d1t3V9++UUBAQHmPTu2bNmigIAAlZWVaeXKlVq4cKFmz55tXvKRm5ur3bt3Szp9bxAfHx+99
J69atU1RUlPn4Tun0tfXz5s2Tr6+vHnnkEW3YsEELFy7UzJkz9eqrr+ruu+/W9u3bVVxcrIULFyo60lqJiYluZ
5E4HA6dOnVKiYmJioqK0rvvvviun02nW8uuvv+rUqVP69ddfzW3WrFlj/lvmqquuktPplM1mM/fL4XCYfTmdTre
+XQYNGqQ5c+borbfe0sKFC/Xmm29K0Ov1o3xkzZpj7NXDgQA0fPlySquy/q2bXPp95DXh1/Vbnb0245rvu2fHQQ
w9p+fLl+vzzzzxUZGal58+YpMDBQF110kTp37qwXXnhBPj4++vXXX6vvt19Xe0aNH9e6771a5Z8e7775rtufallX
XjBkz3H529R8dHa2rrrpK06ZNq/Lz+fa7tsejJse1um0rf7YkKTIy0rxnxzvvvvCObzWa2d+rUKb3111lvmvtd1H
+rqzPofeeQRLV26VEuWLNHo0aPNfR80aJCWLFmiL774QpL0yCOPmG3U5rnNXXd9Hjx5VZmamWx2SzM9Hq1at9N/
//veC3uPK469///5V7tnhGn+ue/ucr+b6W1UVJQkXfAYr+/aPdmmVTHBys6667zpwuL+0475bmZmZKisrk2EYCg401q1Tp1RWVqbLLrtMR48e1T333ON247WsrCwdO0
CwOL+O4875bmZmZKisrk2EYCg401qlTp1RWVqbLLrtMR48e1T333ON247WsrCwdPXpU0dHR5g1BpdOXjvTs2d0
8KehFF12k8vJyHTt2TJ9++qkKCgq0fPlyDR06VEePHtXQoUMVHBysxx57TBkZGZoyZYYoOHDhgPiHkzjvv1Pr16
9W3b19NmDDB7QtMSUmJ7rrrLi1dulQ333yzMjIylJmZKafTad43Y0jQocrMzDSfSnHq1CnzHmQPPPPCAbr/9djP
o8Pf3V2Zmph544AHt3LlTU6ZMUUZGhsaPH++27wEBAWb9I0aMUL12oWRkaEBAwbo6NGjSkxM1LBhw8xHc
D722GNubfj6+rrt87Zt23TixAlt27btrP1W52zt2DbJvP1W52zt7Ny5U2FhYXI6nSorK9Ozzzz6r66+/XoWFhTp06JCOHj0Oqh80
hDh06aNiwYTp69KiioqKUmZlZbb+ufsrKpSRkaEnnnnhC2dnZys7Ol hNPPKGMjAw5HA5zPyvXNW3aNKy
MhQamqqAgMDdfToUfn7+ys1NdX8P/TTpk2r8X7X9njU5LhWt63D4TBrz8s5LhWt63D4TBrz8jIOB133KG77rpLTqdTGzZsUEZGhm6
66SYtXbpUd91111nfb084s/6ffvpJvXv3Vn15uQYyMGK9l69dr6NChmj9/vg4fPijK1/vg4fPizDMHTRRRfpp59+qtNnr7q+y
8rKtGHDBgUEBCgjIOMTJkzQgw8ZH4+WrVqpQbNl zQe1x5/A0d01TLly+v9vfKuW50Wt3xutDPypnbPvbYY/U
yxuu7dk+2aRXN8djwNBbxNJaa9sVTR9zxNJbGg6exAM3DE088YT5+tqYCAgJ06tQpM8Soq+joaF166aXasGGDK
ioqzPm+vr4a0OnSoHn30UXPe2rVrNXv2bPNGh2cTHh6uSZMmKS0tTSdOnDjnur6+vm791mbd6vr7+vm791mbd60hojR8/Xn369Kl
2/Tlz5mjJkiXnbD8kJETPPPPMWduobPPPPMWduobp/P129t2vntb3+r9PR080v4udSk37Vr1+rVV1+t016rVq2UmppaZduav
qcl7b+mLuS4VretK3w733E83/vtCTU95p06dJhYeEFF/b07Lu6z0d9tV9ZdeOvut8r51Pfn5Uzt62vMV7ftXu
yTato6semNt/dCTtE2FHTvvhS746wo/Eg7ACaB9ffoz59+uinn36Sn5+frrzySrVs2VK7d+9WYGCgWrdurdDQU
Pn6+qpr167q2rWrKioqtGGZZMu3fv18RERGy2WzKz89XZGSkKioqtH37djkcDnXs2FHXXH0NfHx8d0TIERUWFqp
Vq1ZZq06aNEhMT5evrq7KyMrOtmJgYDRo0qNr/81pRUHvvvt0R44cUXh4uGJjY/XUU0/p8OHat26tV599VXzu
uiKigpt2rRJn3/+uU6ePKkuXbrozr7zySm3bts2c7tSpk/7zn/9/9o5cqVKi8vV3x8vPr06aNO/nVcr6tqKg
q67pqP5fK+xURESHDMLR9+3YFBwerX79+6tat23nbOHTB0Oofa9JvbdqqpKqKhQTk60cnJyY2Jj2+fGfv3r3mPVcSEhIUE
RFR437PbM9eM/1eTnbtpXrqhwaVP75Qvb7XHXW9bhWt60kt3mdOnXSXRx99+pC1bttTq/faEM+u//fbbS/
fFBsbq0ceeUTB9cH19tk7s2/8PNzt9OF9qenvlZrUXJfx4sfEx7kh227Iepu6pxxsCDtqi
bCjZn3xpd4dYUfjQdgBNA+EHQCaO8IOAE0JYQeA5o6wA0BTUt9hh/mxfsD1+FW0ZB2k/T8OC3oiSaLqq+gGTj8sS4pJbF2bhP56sdHk4L+y9uVbUkkU0v
bCjZn3xpd4dYUfjQdgBNA+EHQCaO8IOAE0JYQeA5o6wA0BTUt9hh/mxfsD1+FW0ZB2k/T8OC3oiSaLqq+gGTj8sS4pJbF2bhP56sdHk4L+y9uVbUkkU0v

+7zkar7CwMEVGRnq7DAAAAACNAGEHaiQyMpIvkgAAAACAJoEblAIAAAAAAEsh7AAAAAAAAJZC2AEAAAAAACyFs
AMAAAAAAFgKYQcAAAAAALAUwg4AAAAAAGAphBOAAAAAAMBSCDsAAAAAAIClEHYAAAAAAABLIewAAAAAAACWQtg
BAAAAAAAshbADAAAAAABYCmEHAAAAAACwFMIOAAAAAABgKYQdAAAAAADAUgg7AAAAAACApRB2AAAAAAAAS/Hzd
gGNgWEYkqTi4mIvVwIAAAAAAKrj+s7u+g5/LoQdko4dOyZJiouL83IlAAAAAADgXI4dO6awsLBzrmMzahKJWJz
T6dT+/fsVGhoqm83m7XJqpLi4WHFxcdq3b5/sdru3ywGaBcYd4FmMOcDzGHeAZzHmascwDBO7dkwxMTHy8Tn3X
Tk4sOOSj4+PYmNjvV1GndjtdgYF4GGMO8CzGHOA5zHuAM9izNXc+c7ocOEGpQAAAAAAwFIIOwAAAAAAgKUQdjR
RgYGBmjZtmgIDA71dCtBsMO4Az2LMAZ7HuAM8izHXcLhBKQAAAAAAsBTO7AAAAAAAAJZC2AEAAAAAACyFsAMAA
AAAAFgKYQcAAAAAALAUwo4matasWWrXrp2CgoLUq1cv/fe///V2SYBlrF27VnfccYdiYmJks9m0dOlSt+WGYWj
q1KmKjo5WcHCw+vbtqx9++ME7xQIW8PLLL6tnz54KDDQ1VRESEBg8erF27drmtU1paqpSUFLVu3VotW7bUkCFDl
J+f76WKgabtzTffVGJioux2u+x2u5KSkrRixQpzOeMNaFhpaWmy2WyaNGmS2WY9NYeaZFhpaWmy2WyaNGmSOY9xV/8IO5qg9957T6mpZo2bZo
2bdqkq6++WsnJySw4fjx47r66qs1a9asape/8soreuONNzRnzhx+lZmYqJCREcycnJKi0t9XClgDWsnW/bNGKSkp2rBqzR/v37dffdd3uxaqDpio2NVVpamj
bNGKSkp2rBhg1auXKny8fnL169dPx48fN9d58sntXsZci1ZskRr1qzR/v37dffdd3uxaqDpio2NVVpamrKzs7V
x4ObdfPPNGjRokLZt2yaJ8Q8QY0pKysLM2dO1eJiYlu8xl39Y9HzzZBvXr1Us+ePfWPf/xDkuR0OhUXF6fHH39cz
z77rJerA6zHZrPpww8/1ODBgyWdOqsiJiZGv/vd7/T000/LkYtRy1nXbZaBA/1ODBgyWdPqsjJiZGv//97/WHP/xBklRUVKTIyEgtWLBA9913nxerBazh0OKFDioiIOJo
la9SnTx8VFRWpbdu2WrRoke655x5J0nfdd6cBDjQKCMjAz17SWQkTzd+I17t3byxUDTV94eL4j+/Oc/65577Fj4+6tu3rFaPNmf8G
+zZszVz5kx17dpVr7/+On+fc15Pj4+6tu3rzIyMrxYGdA87Nmz1/3eW5jMCwsTL1
69WIMAvWkqKhIOukvX5KUnZ2t8vJyt3HHXqVMnxcfHM+6AemAYhhYvXqwTJ04oKSmJ8QY0kJSUFA1MDBRSglJUUDBw50G18Sf
+caip+3CODtFBQUqKKiQpGRkW7zb3IyIMjtXPnTi9VBTQfLU/eRGakW7zIyMjtXPnTi9VBTQfeXl5klTtGHQtA1B3TqdTkyZN0nXXXaerrrpKOulxFxA
QoFatWrmty7gi7Lli1KSkpSaWmpWrZsqQ8//FCdO3fWlAUXL9amTZuUlZVZR1/5xoGYQcAAGg0U
lJStHXrVn399dreLgWwvC5dusjJyVFRUZE++OADDRs2TGvWrPF2WYAl7du3T08++aRWrlypoKAgb5fTbKSnp2t2t/XTqufuEwzEEI5iqRg
amDZt2sjXx17fKnXz8z8/MVVFRXlaqA5sL1DDbvDhgnYbsbF8wCHvf8fd/s1/jpzg/wnXHRCGhbedr7v+PDNvaV2Y9Xirnpzqos
AhQhw4d1L17d7388su6+uqr9be//Y3xBjSA7OxsHTx4UN27d5eAfnL8fn5/8/PyvaSF9/Lz81NkZCTgrgEQdjQ
xAQEB6t69u6/q4/ljatWmXVqlZKSkrxYGdA8tG/fXlFRUW5j0OFRQVauXKlY4sbkcGxEvv46Gef6Vnme3g
u3dlnfv3l3+/v5u427Xrl3AH1Kzcl13AH1x0l0yuFwMN6ABnDLLbk5YTtysnJM89evTQiBEjlZOTo+6uW5VnWIzJMNT
UVIOZMOY9evTQtddeq9n9df13Hjx/X/fff7+3+3SAEso0KSnSnR7t27zek9e/YRNjs8Jk5kzK
t++vZ5//nnFxSlatGiBUPP65LCwMAUHGWPSlatGiRlilbptDQUPP65LCwMAUHByssLEzjxo1TamqqwqKiFVgaOHCgrw0bptrTYGu1lg3Y
71AN1MHnyZA0YMEDx8fE6e6duyYFi1apDQ01LwPlUtISIhat25t
KZ0naq8vDx17dpV6enpVW6YCKD2Nm3apJtuusmcTk1NlSQNGzZMb7zxhrZu3aoFCxbo8OHDiomJUb9+/fTii
zDSZQR2+++aYk6cYbb3SbP3/+fIf+fIOd01aS9Nprr8nHx0dDhgzRyZMnlZycrNmzZ3tsbvpyaQqOmZ9exyZGbfb9fjjjyspKYk
7AAAAACApRB2AAAAAAAASyHsAAAAAAAAlkLYAQAAAAAALIWwAwAAeNzq6eh6dOnq2vXrt4uAwAAAA1
DPCDgAAmrGxY8fq4osvrk7rENVefXv39/bpQEAAAAAGjAMQENSUuuFXuNQEEIAN7xe+ScBdO3cuhOvLeh9/0Pb
gzA4AAJq5wMBARUVuSZLNZtPHtPPcuXXN1++23+vXrr2ufeNF+xUEhIfrNb36++++/OW1WbdXkr9GdD
WnI3LlzFRcXpxWWlQYtCoStHpJzzhMOj+9HbeKGBHnsMzsXuSeXPbfeyU3Lmh1jtvXd/4Bh+nTdfnHMCyWKJqvQOtl9zbLdaKRfWjjLpYnhMGoVqtt1UZuMixm

```
aWluVO2BAAAquJuVwAANHMhISHmpSDVqXy5hM1mO+s8p9PZQBWe9uCDD2rUqFF67bXXNH/+fAObNkwtWrSQJJW
UlCg6OlqrV6+usl2rVq3qpf+IiAjdcccdmj9/vtq3b68VK1a49VdSUqJHHnlEEydOrLJtfHy8jh8/ruTkZCUnJ
2vhwoVq27atcnNzlZycrLKyMrf1XQGKy/TpOzV8+HB98sknWrFihaZNm6bFixfrrrvuqpd9AwDAagg7AABAvcv
NzdX+/fsVExMjSdqwYYN8fHxOxRVXVFn3sssuUOBAgL755htdcsklkk7f2yIrKOuTJkOy17vvttsUEhKiN998U
+np6Vq7dq25rFu3bsrLy5Ofn5/atWtXbUOJCQnKzMzU6NGjzXmuszRq6sEHH9Tvfvc7xcbG6rLLLnM7G6V0t27
avn37WYOjLVu2pLS1NcXJwk1eomox07dlTHjh3l5JNP6ne/+53mz59P2AEAwFlwGQsAAM2cw+FQX16e2
6ugoOCC2gwKCtKYMWOOefNmrVu3TvfddZ/+rqlDiIRo/Pjxeuqpp5Senq7t27froYce0okTJzRu3Dh
zPV9fX40dO1aTJO/W5Zdf7f7nYJSt++fZWUlKTBgwfr888/1969e7V+/Xo999xzZqDwxBNPaN68eZo/f76+//57T
Zs2Tdu2bavVfiUnJ8tut2vmzmzJm6//773ZY988wzWr9+vSZMmXmDOvj4eAUEBOjvf/+7fvr
pJ3300Ud68cUXz9vnyZXz9vnyZMnNWHCBK1evVo/////yzvvnmG2VlZSkhIaFWtQMAOJwQdgAAOMylp6crOjra7XX99ddfU
JsdOnTQ3Xffrdtuu039+vVTYmLiOZ98kpaWpiFDhmjUqFHq1q2bdu/erc8++++8x8BK7LuHHjVFZWViVosNls+vT
TT9WnTx/df//96tixo+677z79/PPPP5n1Ghg0bpueff15PP/20unfvrp9//lnjx4+v1X75+Pho7NixqiocDtDR
Dp9b5M1a9bo+++/1w033KBrrrlGU6dONc9uadu2rRYsWKAlS5aoc+fOSktLq/I43er4+vrq8OHDGj16tp27Kh
7771XAwYMOIwZM2pVOwAAzYnNMAzD2OUAAADrmD59upYuXep5pVeZr+oKBc4JQjZjw/xRUCkpNvGV8WZRdffLGCg4M9W
AOAALgQhBOAAAAAMBSuIwFAAAAAABYCmEHAAAAAACwFMIOAAAAAAAACwFMIOAAAAAAABgKYQdAAAAAADAUgg7AAAAAAACApRB2AAA
AAAAASyHsAAAAAAAAAlkLYAQAAAAAALOX/AzmVk38DHAClAAAAAElFTkSuQmCC\n"
          },
          "metadata": {}
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "#### Family_Members"
      ],
      "metadata": {
        "id": "glOEqF7DH3Hf"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "analyse_numerical_column(\"Family_Members\")"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 542
        },
        "id": "iRP9U1vnHUQf",
        "outputId": "d6c72e7a-bfe6-4323-b797-e158be3e1b54"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "display_data",
        "data": {
          "text/plain": [
            "<Figure size 1300x600 with 2 Axes>"
          ],
          "image/png":
"iVBORw0KGgoAAAANSUhEUgAABDsAAAINCAYAAAAwQHSoAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAABGWUl
EQVR4nO3de7hVdYE//vfhDkfuIwdQbpl4QRQcGkNmrICva0Oh8MS+DQ+hoZhmoiJo6ipdSSebrJQxFm0Kb90eUE
42jZhHeFZEg7+ZlQiSVy4RyRORAnP37o4c9nTAF5LBh8Xo9z34e1lqfvdZ7bRfHfd6sS1WpVCoFAAAAAoCCaVDo
AAAAAwNak7AAAAAAKRdkBAAAAFIqyAwAAACgUZQcAAABQKMoOAAAAoFCUHQAAAEChKDsAAACAQmlW6QDbg/r6+
rz55ptp27ZtqqqqKh0HAAAA+D0lUinvvvvtuunfvniZNPvzcDWVHkjjffDM9evSodAwAAADgIyxevDi77777h45
RdiRp27ZtkjV+YO3atatwwwatatwGgAAA0DP1dbWpkePHPuXf4T+rS+l07ZQcAAAABsxzbl9hNuUAoAAAAUirIDA
AAAKBRlBwAAAFAoFS071q9q9fn0mTJpVnz5p3bp19thjjjzzzm99MqVTVqVnz5o3bp19thjj3zzm99MqVQqjymVSrn44ovTrVu3tG7dOsOHD88rr7z
SYD0rVqzImDF1qzlmDFjQq5du3ToC0Ff+tKXsmrVqm290wAAAMB2oKJl1xVXXZUbb7wx3/3ud3ngxRdz0d/LiiiVTp07t27d0rp16wdwfPvT+tKsmrVqm29uzZ5eW1t
zeXx0yZMiVTp07N9N90nTM3fu3FRXV2fEiBFs2ZNecyYMWPr/pPnAGAAA+hrEsZx/6eEEL7i/bpvEf+ax3Jx+ytOhtH8ssn6vLe+4Fhb6XBbWdG0CwlB6a3+Qxx
XHBBJk6cWJ6ura1nJx49Gm17bJm6urr8fvmyjLzizRovcERNjjd8W6GmbbbCHtZBAT1NoonRzZ8yYMcsJdn4fxUXKIeUdWvZ9LrJ3gBpaiY5b4SGExJjnRz8D3nNg57Ru59ss3o5aLdb5OUZ5sEUdiv83ctNb5
/zzM3r06CRJ//79s2jRokyePDknnhiunbtmiRZunRpunXrXvn7f0qVLM2DAgCRJ165d2xZQ3dbz2xdGWs46ADAq5Ra0q3UxWFdh4oW6PXz0l1zwdjSaV3lcEKWeJMhFrtRvcnCLjbH24b79jeFVHG3
U3//nWrZsmZYtWzbCHtEYWrTeJS2q3UsFAACATVPRe3asXr06OZ00jNC0adPU19cnSfr06ZOuXbtm1qxZ5d+vra3NnP9P++ROc4VV69OHBdcY+6Zjf7I
bWZ03duBg8enCQZPHhw3nnnncyfP7885557pk+fPp0aVK6d++eUaNGdz222BfnHHbYfnyl7+c6dOnZ926
Rk/fnxGjx69SU9iAQAAAIqlomXXkGd/U8nEBGzKpUvnskbt39b54GsgbbaMHJ/iZxYZ4lMN3x8zvPbXZcw8de7Ev42a8p3+LFe+nZ/5ML7
33snf/u3f5r777kurVq38tgbl+xnZ/2LSPp2E1Y2D8JmzTXLWt0zlt0mWsmH9Ygc4kN8TWgsASKnFGDvAoqbDjxa22BaPGk6+Fpb
OAAAAoFCUHQAAAEChKDsAAACAQlF2AAAAAIWi7AAAAAKRdkBAAAAFIqyAwAAACgUZQcAAABQKMoOAAAAoFCUHQAAAEChKDsAAACAQlF2AAAAAIVS8bL
jjTfeyBe/+V5rrVu3tR/wPrX/frsrhfvPjyvPPJKg3WsWLEiYQ8aMSbt27Kt2Kd2rwzwPx6vvvPJKg3WsWLEiY8aMSbt27dKtRqyp9Ohd0k0o
4d86UtfyqpVq7b1rgAAAADbgYqWHWW+//WklpcOMGJEla9aUx4wZMyYrV65M/PPP56qrrsoPf/jDfPe73y2Pmz9/fjZ//vPjyvP+f
0lT/nOpVMp1112Xxy+7KP/3CD1NTU5U//3CD1NTU5U/M3CD1NTU5Kc//W8/W8/Mw99f/W1Nk3/t8/M9+fPW11v8z3rDnuE+H7c8zz+/
/+3//L927d9+2OwUAUEXP7LjrrsyaNCgsjVbc93Ny4nSQa9CvE2n/PvTk/ANPNz5p2n7mdfrzHn/pbt2+egg
w7KnDlzkiR5z5P8nPmPuHvmKxfHNxfPjxYmJ3LlzP3C7dXV1qa2tbfCAzZmTmWeOmo39/rv0nXvfvb82tbfACAAAAiqGiZcfvfvfe9+/zS0nsBrt7lkn3//aT6/PPP1f9/mPePj0/jPs6z/9cTv
/1mttycx3sW
/ayf5fr6WdbFGMhqb2tWzZn3ZbxvuvX0f4nZDUYwq7aX1CBoOzI44w+mgvyTabbJLR+vHJfvYqaZyzLMfHwI9NbQcRYfwOl6wfUPtOLWaf8mThaFezdI8Bt21VI
/6ChJKRsrNerw4Pyu9UX3fpkL6JT5ZmZuZ9tWJUDMePZUXxhSYvxWqkXbtyFYKvywmVV3g6lQuZU0Oynvxj1Gmxno5G72Wjg
4d86UtfyqpVq7b1rgAAADbgYqWHW+//XaGDBmS5s2b52c/+lees1leeOGFXH311enYszWN5zJQpUzJ16N+tRMnjsWcn
0TXV1dUaMGJE1a9aUx4wZMybGb2aHwsZMybGbbPP/98Z2bl22bj3lmXH3xSIjyVWy5x2+e9
01T/n0pVMpl112Xiy66KP3/...
```

AhaLsAAAAAApF2QEAAAAUirIDAAAAKBRlBwAAAFAoyg4AAACgULao7PjEJz6R3//+9xvNf+edd/KJT3ziY4cCA
AAA2FJbVHa89tprWb9+/Ubz6+rq8sYbb3zsUAAAAABbqtnmDL7rrrvKf/75z3+e9u3bl6fXr1+f2bNnp3fv3ls
tHAAAAMDm2qyyY9SoUUmSqqqqnHjiiQ2WNW/ePL17987VV1+91cIBAAAAbK7NKjvq6+uTJH369Mm8efPyV3/1V
40SCgAAAGBLbVbZscHChQu3dg4AAACArWKLyo4kmT17dmbPnp3l1ly5aVz/jY4Pvf//7HDgYAAACwJbao7Ljsssv
yjW98I4MGDUq3bt1SVVW1tXMBAAAAbJEtKjumT5+eW265JWPHjt3aeQAAAAA+liZb8qa1a9fm4IMP3tpZAAAAA
D62LSo7TjnllNx+++1bOwsAAADAx7ZFl7GsWbMmN998c/vHrZmN2Ssv+aaa7ZKOAAAAIDNtUVlxzP
PPJMBAwYkSZ577rkGy9sFAAAAKikLSo7Hnjgga2dAwAAAGCr2KJ7dgAAAABsr7bozI7Pfe5zH3q5yv3337/Fg
QAAAAA+ji0qOzbcr2ODdevW5amnnspzzzz2XE088cWvkAgAAANgiW1R2XHvttR84/9JLL82qVas+ViAAAACAj2O
r3rPji1/8Yr7//e9vzVUCAAAAbJatWnbMmTNMTMnrVq12qL3futb30pVVVUmTJhQnrdmzZqMGzcunTt3zi677JKjj
z46S5cubfC+119/PSNHjkybNm3SpUuXnHvuufnDH/7wcXYDAAAA2IFt0WUsX/jCFxpMl0qlLFixIk888UQZM
mbfb65s2bl5tuuin7779/g/lNsB+s2/1g33t/PmzkD/gH/lnnXVW7rnnnnvz4xz9RAvB3/bM/V1damrqytP19bWbrX9QAQAAAcpri8qQ0DebPn
58XX3wxSdKvX7+/WC78JXfccGiRTp06NBgfk1NTYsWVIe86dFx4b5G5b9JZMnT85
ll122WVkBAACAHcMWlR3Lli3L6NGj8+CDD5bLiHfeeSef+9zncscdd2TXXXf9yHUsXrw4Z555ZmbNmrXFNzdU
hdccEEmTpxYnq6trU2PHj22aQYAAACgcWzRPTtM/OP/30vPvuu3n+++8zYsWKrFixIs8991xqa2vzta2vpMNnbNI65s+
fn2XLluuXAAw9Ms2bZ0Ss2zZszz00OE02ZONvqmjVrlpqamjxbvvPNOg/ctW/Y6XVZBA4eTdYnDWkD7q/2v8Yl7ezbJjeM0aDt
GzZMu3atWvwAgIphi8qO++NFjKUCJsDFZ22Sxbtuoq/0BBHD22Wef8r9qr9QnsDdL69n5q06e2nf1rsrWjGbkpyh0MzNaBYmdlv+jOqjGD0mo/1cJ8SNsNZLr4T/j8XvnWTAePaBQoy7kHt2kN9YD9KiUMzE7Ao7ssJ79c50cbbNcnsDC8YvSdEcVdppvGfr4tPGuU/qapfFdSDp6W
jxpT/3Lx588yePbv8npdeeonHjn0BwI8oTTQv/M6fxdz+P6wzoFv1sG/SkV4tdVkoNEaaRs4trIVMzv15KlYcbpfwAiTDtdGkcDHN6kr1tTN8lmnDI6+TNPPC5m8aHFRVfGw76h3bW+FMAhS
NfXp3nz5hvNb968eerr6tzpHW3bts1+++DXU6Hd7fC76+cnv+1L30pEydOTKdOnKuXbucfvrpGTx4cD796U8nSQ4rzzIpRXmA8HFwlK0Tc+lH8rH+9yD+eQ
nSQ499NDsu+++GTt2bKZMmZZI1S5kkoosuyrhx49KyZcst2TUAAABgB7dFZ3YMHTo0Z555Zt58yvDfeeCNnn
XVWhg0btbttXCXXvttTniiCNy9NFH55BDDknXr13zk5/8pLy8adOmmTlzZpp2bTOADOMEMH+7j4VYPCrOGFRhUdk

1JXV6fsAAAAoNC2uOxg+9Ci9S5pUd2u0jEAAABgu7FZT2MBAAAA2N4pOwAAAIBCUXYAAAAAhaLsAAAAAApF2QE
AAAAUirIDAAAAKBRlBwAAAFAoyg4AAACgUJQdAAAAQKEoOwAAAIBCUXYAAAAAhaLsAAAAAApF2QEAAAAUSrNKB
wA23erVq1NXV7dJY1u2bJk2bdo0ciIAAIDtj7IDdhCrV69z598vvlyzZpf0Oddu+T11ZH2ZqPAAAgJ2OsgN2EHV
1dfn98mUZecWdadF6lw9T5f56kzNZp4mNLO5Ajg3cH8h9JmBuNuy7adLjvfNVq2JUWJkCsq/SyVcvd7/8ad5Arx8cZF2Pe3
Jk/OpT3Oqbdu2TZcuXTJq1Ki89NJL89iSdcasWbMm48Ds+sWp23dFr2rhyVAKk6ZOTUOEFyRWt/dQoTyncaaoXcqgXOVhU1wBJd0ybPaB0mvQ1zzwPAGMmqu4uYoXgU0ZGBV/4Awcvi9t9LcSMZqXwzfBS6/sN6jE71YMAAAMUdiSjVEVUrmUx7nsLYMKNkNQMpBvjyWmN/9fWqD9i1F2L0k3pHOvPF5jl71/UmK3FhIYVgmMCCLoFTrLU+QaHSY8RuwVIzpCZMsSbK

```
sqUKVmyZEkuuuiijBs3ztkbAAAAsBOqaNlx4403Jkk++9nPNpg/Y8aM/NM//VOS5Nprr02TJk1y9NFHp66uLiN
GjMgNN9xQHtu0adPcfffd0e200zJ480BUV1fnxBNPzDe+8Y1ttRsAAADAdqSiZUepVPrIMa1atcq0adMybdq0v
zimV69euffee7dmNAAAAGAHtd08jQUAAABga1B2AAAAAIWi7AAAAAAKRdkBAAAAFIqyAwAAACgUZQcAAABQKMo
OAAAAoFCUHQAAAEChKDsAAACAQlF2AAAAAIXSrNIBALbU6tWrU1dXt0ljW7ZsmTZt2jRyIgAAYHug7AB2SKtXr
O7P3n3y++XLNml851275PXXFio8AABgJ6DsAHZIdXV1+f3yZRl5xZ1pO0Xqrx279v1VuefCY1JXV6fsAACAnYC
yA9ihtWi9S1pUt6t0DAAAYDviBqUAAABAoSg7AAAAgEJRdgAAAACFouwAAAAACkXZAQAAABSKsgMAAAAoFGUHA
AAAUCjKDgAAAKBQlBOAAABAoSg7AAAAgEJRdgAAAACFouwAAAAACkXZAQAAABSKsgMAAAAoFGUHAAAAUCjKDgA
AAKBQlBOAAABAoSg7AAAAgEJRdgAAAACFouwAAAAACkXZAQAAABSKsgMAAAAoFGUHAAAAUCjKDgAAAKBQmlU6A
ABbx+rVq1NXV7dJY1u2bdo0ciIAAKiMwpzZMW3atPTu3TPTu3TutWrXKQQcdlCeffLLLSkQC2mdWrV6dn7z7p1Kn
TJr169u6T1atXVzo2AAA0ikKc2fHv//7//7vmThxYn56Df/nJNOil1lJMn369XyBRQc0NnW///ve/n/n/PPP7/C6QC
2nRatd0mL6naVjtEoXKYDAPDhfF/6Xzt82bF27drMnz8/F1xwQXlekyZNMnz48MyZM35NXV9fgAFi5cmWSp
La2tnHDbkUbsr739rKsW/PeR45f+4fN+/4fxyxevHiT9rNUKqWqqmqmmqTsmzO2MOZv+G/y6bs4+bu3+bkaMyxmzO+MT+
PxsrcmGN9Hg1tL59HY31277//fg4e8rd5e8XvN2m9HTt1zuOPPPZrWrVtv1RyNOVaOLR8rhxxxxba6wccmytsXLIs
bXGbs74zf6+1Pmv8sJzz+5QhceG766lUukjx1aVNmXUduzNN9/Mbzz/Mbr3/DBw/M89p47+1ck+rHQfMWKWzVzBMk
oPZdeemkuu+ybRkTAAAA2AoWL16IFjgcRxwP9yLLCBY4ENHAs7rlKplHfffTfdu3/yLE7fNnxV3/1V6X6pYP7
JHFixenEssCBbtinrrPpnEss6IFSIFgcRxwP9yLLCBY4ENHAs7rlKplHfffTfdu3/yLE7fNnxV3/1V2natGmWL13aYP7
SpUvTtWvXjvnW/sXD3xPy5YtO07/yjZ/jZcW06ixSnnFzt81zpAnv7sKRg0fE+dDnbstNZ7qbPqPbmZaAsSAqOeRt2pnZbZMR/SyQCMNr5lvSqvkR/1vt1x6pd0WWK0WQvXpdu3K7hIfGU7gnnPGnInO2MaCBNYnwfzpxLL8svYMJb1O1m
bAWqiVBHFyhKQrCVibjVYBWBnrPDbWnjDsVkXpWMe6oAWEnVXmf/T9Oef1iHZkrzbxT3TvEmxLhiBMWRoLovHYkdlbgdgU8XMDzwe
vSmrT8mDFqhcRPrDnRU0Ne4MVpRlGNv2/nPAwLmbrJLxrT1BzuUtuxlGQMNC3onn/QMbP8IRpwYaaf1i6G4YADI3n1PSNrnVNbh7BAGFVRhuNTNJZ8l
YLcTLaQNqlv0b8GCbLd66mdbNvkzMMhtf5ZHOomtzYjI1e5OCo3L8XJ8Q4a6Xp+SV3nt58D2G34pCP3Ve9+X65Cu7hPcw/TMH+MWMMwpQ5
f6Vrgx2yzTb42s3WNpHgY8dgDzBR3t5GNCpnPh0SN6X+b6sJMZb5hS4wGNLvCktD7vTjkfd26D1EtWWhnAJoWM1vDEtkHBQ55ieDo/K0Nb6
6MEo5TRXTNoz+Iut+f2GWwjm7NeOn2vsN1zo6dUJT94dSH6TuCXM4DHT68j5wQR0Nf3IYldw05jhrLDJOgX21bA4cAQXuHEXnxXNmoqp
2sB/hKm+YqZ2e9Sf+BJEJPqsP3HJSNyE/9J6enm1CPRM4cqfnbEKbPhsO1fYpLbJHYG7OJ05OTx3ldLVw/1ufgJHbP19fZ1QmiSmgK7ZbgUJ
Kr/ir9jHtJ2GvWQ8/ejJSZjVzmRQFVw2mUsyIfHLjyIdpyeWJwDlYbtB3YvTUqnNZAs5eHmtMMdg/OqfJvWO1bOllGNMX5iL4CTlqCeljA
```

pVNpGvQoAAABAo3ODUgAAAKBQlBOAAABAoSg7AAAAgEJRdgAAAACFouwAAAAACkXZAQAAABSKsgMAAAAoFGUHA
LBN9O7dO9ddd115uqqqKj/96U8rlqcx/Pk+AgCVoewAgJ3UP/3TP6Wqqmqj16uvvtoo25s3b15OPfXURln3Zz/
72VRVVeVb3/rWRstGjhyZqqqqXHrppY2ybQBg+6PsAICd2GGHHZa33nqrwatPnz6Nsq1dd901bdq0aZR1JOmPH
j1yyy23NJj3xhtvZPbs2enWrVujbbexrV27ttIRAGCHo+wAgJ1Yy5Yt07Vr1wavb3/72+nfv3+qq6vToePfO1
rX8uqVavK77nlllvSoUOH3H333dlrr73Spk2bHHPMMVm9enVuvfXW9O7dO7dOx07dswZZ5yR9evXl9/3YZd4DBO6N
OPHj28wb/ny5WnRokVmz569SftyxBFH5H/+53/y2GOPlefdeuutOfTQQ9Ol8S5cGY+vq6nLOOedkt912S3V1dQ4
66KA8+OCDH3sfk+Tdd9/N8ccfn+rq6uy2226ZNm1ag+XvvPNOTjnll0Oy6665p165dhg4dmqeffrq8/NJLL82AA
QPyr//6r+nTpO9atWqVJLnzzjvTv3//tG7dOp07d87w4cPz3nvvbdz//tG7d0p07d87w4cPz3nvvbdJnAwA7G2UHANBAkyZNMnXq1Dz//PO59dZ
bc//99+frX/96gzGrV6Y/O1KlTc8cdd+S+++7Lgw8+mKOOOir33ntv7r333vzbv/1bbrrpptx5552btM1TTjklt
99+e+rq6srzfvjDH2a33XbLOKFDN2kdLVq0yJgxYzJjxozyvFtuuSUnn3zyRmPHjx+fOXm5I477sgzzzzyTY48
9NocddlheeeWVj72P9P//Iv/5IDDjggv/71r3P++efzDPPzKxZs8rLjz322322Cxbtiw/+9nPMn/+/Bx44IEZNmxYV
qxYUR7z6quv5j/4/+4z/yk5/8JE899VTeeuutHH/88wXvvCFlEqlTfpsAGCnUwIAdkonnnh
iqWnTpqXq6ury65hjjtlo3E9/+tNS586dy/eP/T6enrj+pcrXnFbr/wl1LFjx9K///S23Nsnnumz4CePcAAw7InnvumTvvvDM
PPPBAxo4dm2bNGn7defbZZ7N+/fr07du7u2p/fr07ty5PL2l+zh48OCNpjdcvvPO09n1apVDbaTJO+//37++
7//uzzdq1ev7Lrrrg32a9iwYenv+/l7D26Nej6jNrQAmsQAx01BOAsBOrrq7OJz/5yfLO9a6+9li
OOCKnnXZarrjiinTq1CmPPvvpovvSlL2Xt2rXl2NVVfWB8+rr6zcYUN5+rr6zc5yymnnJIBAwbkd7/7XWbmJGhQ
4emV69em71PJ598cqZNm5YXXXnghTz71L6alaZNm2b+/Plp2rRpg2V/WmQ0xj6uWrUq3bp1a3B/kAO6dOh
Q/nN1dXWDZU2bNPy+OOP5xe/+EWuv/76XHjhZk7d26j3VAWAHZkyg4AoGz+/Pmpr6/nk2fPxK7Fmpr6/P1VdfnSZN/nhrr
x/96EfbZNv9+/fPoEGD8t3vfjfz58/PkPxIx8xz57IvdHHw8NRPpdg76WsFvn7LlaWSUvdzj/fvuTtpMVcj0ATJfI2zdZjnDUp8s+nqSYbDEd8H3oD1Vm1tJiR7IEOmxIDjV
vPm7sjTzxxBMbTe+zzz5JkgMPPDBLlixJs2bN7NOrt3781ab1VVVVYYMGZIhQ4bk4osvTq9evTJz5sxMnDhxaOUHg
MJQdgAAZZ/85Cezbt26XH/99TnyyCPz8+c/ff+l/tZrc18s/n5c3oL3vDOuPbDfdEy3g8Ebh3IzMDy3EdqIzqRyhdDdAQ8kdjaRMtOzrBRWGGbzaS7gF/fHbSoxDg
NnIz0ZFpM6TWkpWfk1r5u+V9b5Hkd8b9DtYeiPPsYvn+WWpTXn1MXfFMZ2smE7b5jZWTUrGTN2q/HPovz/7uLlNeedAzt33x23FUxDCrvQnPTEEzkFgy7YN2QX8Wo8fx+EUVv72p3+3Sd7t3L0sxejFgPaSGfT+4/5HaBf+F4eXL4P+5tpNPeXzzvLx9Xy22kfamZLbSQNPnm
4A2Sz9Vbt9UgGVBt4rb4/RBPG2cEXGYxWWaLpTlFs1MzrOd0aaq9ZnDwCH1i7Y1H3LZrFbOUqiKJEcL6jiFUZo7v0Q5VaMzkE5RJB9MqACL7jgDEmugIWZVXPtxMp4e5wAEOMN/5qSlTQLPbwYeyaQmJTDFZ+7xLbI+QUGzkDmb1bL9lNP4GgdbjnfaU2obNGo
ZyzUgOfM6U6cTGtd6ZsIdqbyX5tvVRxAn3bXqvD9r0n/RlcQLjPYhJrnp0fFMcJ2fcvNMNlR8+ixV4hpQMZGpATFNhqyGU4eJy+xvYNgkbe/n0J3gD3HuMrRs7TJ8sm4AP+ktpAAQ3M9yR2RSWSJCr0k/E1yrr4TbeOQ6hAkIvR6eOz2sP+z85tzKAxjl8wK4PdFMACfQ1Abv8zRchadpqtsnmI0pfu1ykuPVb4lLJ0cqnMHfrMDBbmNv0czmQkPC94hTuFeyGeG6axMBKN4NFjNqYQcCU

SV5+OGHKxORAKCidrbvS8oOgB3Uyy+/XP7zD37wg9x///356le/mvvvvz8/+MEPPnBcEa1fvz433HBDBg8enMs
vvzz9+vVLmzZtOq9fv1x++eUZPHhwbrzxxqxfv77SUQEAKmJn/L6OU17GUldXl7q6uvJObW1tBdN8PIsWLapOh
Ip66623Kh1hu7GzHws74/5vuFSlSZMm6dmzZ4NlPXv2TJMmTVJfX59TTz21OJezPPPMM1myZEkmTZrU4KyW5I+
fzZgxYzJu3Lg888wzGThwYIVSAgBUzs74fWmnLDsmT56cyy67rNIxtoorrriiOhHYTjgWdl7HHXfcB84/6qij8
h//8R/bOM22t2LFiiRJnz59PnD5hvkbxgEA7Gx2xu9LO2XZccEFF2TixInl6dra2vToOaOCibbchRdemF69elU
6RsU88cQT+f73v1/pGNuFnf1YWLRoOU5b+PzoRz/KV7/61Y3mz5w5swJptr1OnTolSRYuXJh+/fpttHzhwoUNx
gEA7Gx2xu9LO2XZObJly7Rs2bLSMbaKXr167ZSPl9xgZ7x04S/Z2Y+FndHNN9+cUO89NfX19Xn99dcbXMry+uu
vp76+vjyuyPbff/907do1t912Wy6//PIGp2bW19fnttuS7du3bL//vtXMCUAQOXsjN+X3KAUYAf1p+XWCSeck
KFDh+b666/POKFDc8IJJ3zguCJq2rRpvva1r2XOnDm56KKL8vzzz2f16tV5/vnnc9FFF2XOnDk57bTCvn8eAC
ATbEzfl/aKc/sACiKBx98sPx42fr6+o3u0VHkG5P+qUMOOSSXXXZZbrjhhowbN648v1u3brnssssK99x4AIDNt
bN9X1J2AOzgHnzwwbz88svlp7Mkf7xOpehndPy5Qw45JEOGDMkzzzzyTFStWpFOnTtl///0L9S8UAAAfx870fUn
ZAVAAffv23WnO4vgwTZs2LczjOgAAGsPO8n3JPTsAAACAQlF2AAAAAIWi7AAAAAAKRdkBAAAAFIqyAwAAACgUZ
QcAAABQKMoOAAAAoFCUHQAAAEChKDsAAACAQlF2AAAAAIWi7AAAAAAKRdkBAAAAFIqyAwAAACgUZQcAAABQKMo
OAAAAoFCUHQAAAEChKDsAAACAQlF2AAAAAIWi7AAAAAAKRdkBAAAAFIqyAwAAACgUZQcAAABQKMoOAAAAoFCUH
QAAAEChKDsAAACAQlF2AAAAAIWi7AAAAAAKRdkBAAAAFIqyAwAAACgUZQcAAABQKMoOAAAAoFCUHQAAAEChKDs
AAACAQlF2AAAAAIWi7AAAAAAKRdkBAAAAFIqyYwfVs2fP3HzzzenZs2elo8B2wd8JAABgg2aVDsCWadWqVfr27
VvpGLDd8HcCAADYwJkdAAAAQKEoOwAAAIBCUXYAAAAAhaLsAAAAAApF2QEAAAAUirIDAAAAKBRlBwAAAFAoyg4
AAACgUJQdAAAAQKEoOwAAAIBCUXYAAAAAhaLsAAAAAApF2QEAAAAUirIDAAAAKBRlBwAAAFAoyg4AAACgUJQdA
AAAQKEoOwAAAIBCUXYAAAAAhaLsAAAAAApF2QEAAAAUirIDAAAAKBRlBwAAAFAoyg4AAACgUJQdAAAAQKEoOwA
AAIBCUXYAAAAAhaLsAAAAAApF2QEAAAAUirIDAAAAKBRlBwAAAFAoyg4AAACgUJQdAAAAQKEoOwAAAIBCUXYAA
AAAhaLsAAAAAApF2QEAAAAUirIDAAAAKJRmlQ4AWOPV2lVp8t7/VDpGRTRZs7LSEQAAALYryg52aDl69EiTJk3
T6oOFyRsLKh2nYpq3aJn27dtXOgYAAMB2QdnBDm3vvff0//f/3Z6VK3fusxvat2+fmpqaSscAADYLig72OHV1
NT4RR8AAIAyNygFAAAACkXZAQAAABSKsgMAAAAoFGUHAAAAUCjKDgAAAKBQlBOAAABAoSg7AAAAgEJRdgAAAAC
FouwAAAAACkXZAQAAABSKsgMAAAAoFGUHAAAAUCjKDgAAAKBQlBOAAABAoSg7AAAAgEJRdgAAAACFouwAAAAAC
qVZpQNsDOqlUpKktra2wkkAAACAD7Lhd/YNv8N/GGVHknfffTdJOqNHjwonAQAAAD7Mu+++m/bt23/omKrSplQ
iBVdfX58333wzbdu2TVVVVaXjsBlqa2vToOePLF68O03atatOHCrIscAGjgUSxwH/y7HABo4FNnAs7LhKpVLef
ffddO/ePU2afPhdOZzZkaRJkybZfffdKx2Dj6Fdu3Z+UJHEscD/ciyQOA74X44FNnAssIFjYcfOUWdObOAGpQA
AAEChKDsAAACAQlF2sENr2bJlLrnkkkrRs2bLSUagwxwIbOBZIHAf8L8cCGzgW2MCxsHNwg1IAAACgUJzZAQAAA
BSKsgMAAAAoFGUHAAAAUCjKDgAAAKBQlB3scCZPnpxPfepTadu2bbp06ZJRoOblpZdeqnQStgPf+ta3UlVVlQk
TJlQ6ChXwxhtv5Itf/GI6d+6c1q1bp3///vnVr35fVhsY+vXr8+kSZPSp0+ftG7dOnvssUe++c1vxv3Yi+/hh
x/OkUceme7du6eqqio//elPGywvlUq5+OKL0KL061bt7Ru3TrDhw/PK6+8UpmwNKoPOxbWrVuX8847L/379091dXW
6d++eE044IW+++WblAtNoPurnwp/66le/mqqqqlx33XXbLB+NS9nBDuehhx7KuHHj8sQTT2TWrFlZt25dDj300
Lz33nuVjkYFzZs3LzfddFP233//SkehAt5+++OMGTIkzZs3z89+9rO88MILufrqq9OxY8dKR2Mbu+qqq3LjjTf
mO9/5Tl588cVcddVVmTJlSq6//vpKR6MRrF27NoMGDcrUqVM3uXzKlCn5yU9+kmnTpuVXv/pVqqurM2LEiKx
ds4KY3tw46F1atXZ8GCBZk0aVIWLFiQn/zkJ3nppZfy93//9xVISmP7qJ8LG8ycOTNPPPFEunfvvo2SsS149Cw
7vOXLl6dLly556KGHcsghh1Q6DhWwatWqHHjggbnhhhty+eWZ8CAAVr5ncz555+fxx57LI888kilo1BhRxxxR
GpqavK9732vPO/oo49069at88Mf/rCCydiWqqqqMnPmzIwaNSrJH8/qd6e84+++ycc845SZKVK1empqYmt9x
ySOaPHl3BtDSmPz8WPsi8efPyN3/zN3NN97IQQcdlJ///OcZOXJkJkyJkyY4CzhgnBmB
zu8lStXJkk6depU4SRUyrhx4zJy5MgMHz680lGokLvuuiuDBg3Ksccemy5dumTgwIH57ne/W+lYVMDBBx+c2bN
n5+WXX06SPP3O03nOOUdz+OGHVzgZlbRw4cIsWbKkwf8n2rdvn4MOOihz5sypYDK2BytXrkxVVVU6dOhQ6ShsY
/X19Rk7dmzOPffc9OvXr9Jx2MqaVToAfBz19fWZMGFChgwZkv3226/ScaiAO+64IwsWLMi8efMqHYUK+u1vf5s
bb7wxEydOzD//8z9n3rx5OeOMM9KiRYuceOKJlY5HI7rgggtSW1ubvffeO02bNs369etzxRVX5Nhjj610NCps0aJFSZKamJgG87W1NeVxd0arV6/O+e..." per zoom "bb7wxEydOzD//8z9n3rx5OeOMM9KiRYuceOKJlY7HNnT++eentrY2e++9d5o2bZr169fniiuuyJgxYyodjQpas
mRJkqSmpqbB/JqamvIydk5r1qzJeeedl+OPPz7t2rWrdBy2sauuuirNmjXLGWececUekoNAJlBzuOcePG5bnnnsu
jjz5a6ShUwOLFi3PmmWdm1qxZadWqVaXjUEH19fUZNGhQrrzyyiTJwIED89xzz2X69OnKjp3Mj37Oo9x22225/
fbbO69fvzz11FOZMGFCunfv7lgAGli3bl2OO+64lEql3ZjjZWOwzY2f/78fPvb386CBQtSVVVV6TgOApexsMM
aP3587r777jzwwAPZfffdKx2HCpg/f36WLVuWAw88MM2aNUuzZs3yOEMPZerUqWnWrFnWr19f6YhsI926dcu++
+7bYN4+++yT119/vUKJqJRzzz03559/fkaPHp3+/ftn7NixOeusszJ58uRKR6PCunbtmiRZvHhxg/nFixeXl7FzWbduXb7yla/kt7/9bWbNmuWsjp3Q7Nmzs2TJkuy5555p1qxZmjVrlt/+9reZOHFievfuXel4NKIuXbpkn332aTD3iU98Iq+99lqFElFJixZNVVNTk1GjRqW2tjYzZszI9OnTc/LJJ1c6GhXUp0+fdO3aNbNmzSrP1dbW5qmnnnIZ/E5oQ9Hxyiuv5Je//GU6d+5c6UhUwNixY/PMM880uG67W7duOffcc/OLX/yi0vFoREOGDNno0fMvv/xyevXqVaFEVNLatWvTpEnDX2ObNm2a+vr6CiWisaxZsyYLFy7MwoULk/zx/o4LFy7Ma6+9lqqqqkyYMCGXX3557r///jz77LM54YQT0q1bt4waNaqiudn6PuxYWLduXb785S/nV7/6VX7wgx9k/fr1WbRoURYtWpT353ffXV/efKoqP1xNpAQ6zqNpF9CQBNC6DQvdE3qkIaY8iA2Xmm0WmhMU7GtHh2aaIUrT8ayzBfVpHydjH/Z1+jJh3KTFYBmHpPfLMgS2E//x/6n+cRlpdNvE2bI=
fTp06fSkaiQYcOG5dnnnsu

```
/5TRt2jT19fUVSsT2oE+fPunatWtmz56dAQMGJElqa2szd+7cnHbaaZUNxza3oeh45ZVX8sADD6Rz586VjkQFj
B07dqP7vY0YMSJjx47NSSedVKFUbE3KDnY448aNy+23357//M//TNu2bcvX2rZv3z6tW7eucDq2pbZt2250r5b
q6up07tzZPVx2MmeddVY0PvjgXHnllTnuu0Py5JNP5uabb87NN99c6WhsY0ceeWSuu0KK90zZM/369cuvf/3rX
HPNNTn55JMrHY1GtmrVqrz66qvl6YULF+app55Kp06d0rNnz0yYMCGXX3559txz/Tp0yeTJk1K9+7dP/QpHey
YPuxY6NatW4455pgsWLAgd999d9avX1/+LtmpU6e0aNGiUrFpBB/1c+HPi67mzZuna9eu2WuvvvbZ1VBpDCXYwS
T7wNWPGjEpHYzvwmc98pnTmmWdW0gYV8F//9V+l/fbbr9SyZcvS3nvvvXbr55psrHYkKqK2tLZ155pmlnj17llq
1alX6xCc+UbrwwwgtLdXV1lY5GI3vggQc+8PvBiSeeWCqVSqX6+vrSpEmTSjU1NaWWLVuWhgObVnrppZcqG5pG8
WHHwsKFC//id8kHHnig0tHZyj7q58KfK69Wr+naa6/dphlPFWlUqm0jXoVAAAAgEbnBqUAAABAoSg7AAAAAgEJ
RdgAAAACFouwAAAAACkXZAQAAABSKsgMAAAAoFGUHAAAAUCjKDgBgm+jdu3euu+668nRVVVV++tOfVixPY/jzf
QQAKkPZAQA7qX/6p39KVVXVRq9XXX321UbY3b968nHrqqY2y7j7s9+rY2WjRw5MlVVVbn0OksbZds
AwPZH2QEAO7HDDjssb731Vsb73VoNXnz59GmVbu+66a9q0adu+66OQtJmUMmyr5n+9JZbbmkw74033s2bPTrVu3RttuYu7
m2lIwDADkfZAQA7sZYtW6Zr164NXm3b5ni7NXt/+9rfTv3//VFdXp0ePHvna176Wn+fzYm3F7ruz1157pU2
bNjnmmmGOyevXq3Hrrrendu3c6duyIuYM844I+vXry+/78Mu8Rg6dGjGjGx/fYN7y5cvTokWLzJJ49e5P25YgjjsjT//
M//5LHHHivPu/XWW3PooyW6Yh/Ve5+c4Uf/+S8UdcbW1dXlnHPOyW677ZbqZcb6Vrd12KP8vpqamvTu3btTuX56sn7I2Tk4F+aO0+7Ifff
dMm3atAbL33nnnZxyyinZDddd065duwduwd0jRPP/10efml116aAQMG5F//9V/TpO+ftGrVKkKly5513pn///mndu
nU6d+6c4c0H57333ukzwYAdjbKDgCggSZNmmzTq1K5l/vnnc+utt+b333//VlyYSm6//e+fbU1dWV5/3wbhz/MbrvtlrvQ0a
dlwcffDBHXVU7r333/7t9x6efdd8vzRowYUyYUfrKV75/S5Nuyf+fDlfnvXS7/eSmptvAwwbl
SimmTq1K15/vnnc+utt+b333//VlyYSm6//e+fbU1dWV5/3wbhz/MbrvtlqFDh27S0lq0a
JExY8ZkxowZ5Xm33HJTj755I3Gjh8/PnPmzLL8n9x7/Pk58MADM2zYsKxYsR6eee2woO+ywvPLKKx97H//nwwvF/l/4lBxxwQH7961/
n/PPPz5lnnplZs2aV1x977lFBUjQHvmqmTGqQX2ka1B4MDBAPyQABEf96KppVpGi3sRWkKFbZzhG95/Gd3
lX/b81p+TzpkPPQR/lgjeePo7qq2PfWqFSRY8vXlbHG7cvgq7fWJ35ZJjrfrh41i9/2e/1+2z7aYj/xI3+/7rMX+oubaRsG+iyqsv85ekudzwsCx9/PpvsyM5p3UEpBrHXQ2pt1Lbr8fmm4bwrDi8y8S/2M/QNkyBh2TzC4WJXgUtTfrOdDv0c8ef06tScb+fLW1u78n+NrdxWTH/KgjA77eaOg+mt0LGcFJVJQyXKJ
lmH7TzbptDtI0M1IVQ1llmNmQrY/vPyyd+f54dHnXhTPTpIGmIgQQpMw2Yyc/s8dK07qbu/LWXNXK0aQSUpP6C5fvv/6p+H/ou/F74pDfCp3kZ8zhw9y7mZk88U1fYZV+sX2/2LjfiThzNl4W2UOyOjrv3vRTJwjnfPvEfYKd5Dkt5MN7Kp8ytdHp2XssHlJFfn66Y3B7L/rQLI5HzsWE3/pw0ATDIvx/8nOYhaCBDxk8aMB8lCwrVkU0BVz7/r0Wrw0/qhLXg2MnvaG3p37Rq4+hZcBPvFwZb93pWDcyoB2v8NB8fCZ2D1dBMOj3kv6vbQmQc7yyGLyH3Z3r93dbQpJbaTy1r3Q9R5ml0H5qqQWAuKzHOc0aW5qQxUOkyvRR7e57OwhjZ0B+12wZF7Zj4dH9SEjXSMt5VMnnP3L79sQ2HiqXnP/jQj5/nmpm7k/6QV+cdbcvSr/0m7mV7NMb6hkuW8y26/KMb+AQE4ykfh0bY7/j5ArkSAWvNFWr9ZE5KYvTnnTA5E5tTKrXXMnRmT6qBHumZf90CFV4/dzIVh9G7mAncxp+3/O/P5g1bqN4q0lQYZJTsiHS9QKqnWRpXdXjXRMrnMYmhY7Y7Wfuv/oHwt5ihDVvSF9l+XbR6tdk1cxqr9GL6/3LTeB9bvHSh9Q0rjrA7rXqeDh6r2WyI8L+qNI2EMhKd9Lw5+2RI0n7mMK8p5sXnPRRCQpxXnQ8Nmv/S7VaPwBiE9r91q5cshh4q4ymP5Jx3ZP8j5vZ8dc3ljXbkUHmjxBZvVTPDVJkZ6zZdAhG1umVXH2X6xxXSYQcRRTZ1IeXfDIhBHS+FEZaDh6sbgi8nNfy+7tKdCHCfDtnHjvhAn7gv3Gpb/6nnx5Y+d0b0Pym0GcK4WAAxSdSaKCZGAgZ37mHnE61ab6y5Z9ntbnOF6LZTd2KNmLJ9e81CfXc8Mm1i8i6k6ZznR1MhgsahYKhMeJjqW1RDtUzlyKcIAfxsHkCQmFO1dtcdO2Z2vP9sCuqpBrVbvdE18Z/b8Bo/2ZcbN/iirnXSffZr+/rwT9wc3iEwYH/B/ghLlIP1ZYKxXf8cS/ao9UpwxKY+6OlbSJbCGdbMmZv8UD9VRcEbUbYAxBEf8a7sDEF1w1QdNzrS/vG6fW8z/Os5/tGnHkXIawHjWs/GvVWLjSXrn3Gah9zNUGx/ZgtSK0n/LlQ4S/MaWb7PXT/u6Gmrmhh6FDPnupt1m4qX2l3j2+2F4Ovj9UgAQ6xFHn5hVAJIGBwPtSqtIXT/nrpgWUMtG2jqE6wVwd6nyYP5TKjHsPHN/1CxXgpe68HzFmG8/kIRumBQeSdu4Yez
/9V+l/fbbr9SyZcvS3nvvXbr55psrHYkKqK2tLZ155pmlnj17llq
```

```
      "source": [
        "#### Age"
      ],
      "metadata": {
        "id": "KBVbNv66H4bU"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "analyse_numerical_column(\"Age\")"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 542
        },
        "id": "wLLhVMHaHVO5",
        "outputId": "5a408471-5be7-4b27-d274-9c9214f19971"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 1300x600 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAABDsAAAINCAYAAAAwQHSoAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAAA2k0l
EQVR4nO3de5TV9X3v/9fAwIDCDCIyQGQQI4KXYPASJOovMXBE4klDQxM10OsbEG8mKZSWKZMWQk9QsbEQU
tR4Uq9Z0WqaaBEj1lOCSpYXFPESFAy3og4DCAj0/v3R5T6deInCMHvm/PPd5d5N5D1/HWnuF+X6/s+e9XZ/1XTPPfPd3V5V
KpVIAAAAACqJLpQcAAAAAaE1iBwAAAFAoYgcAAABQKGIHAAAAUChiBwAAAFAoYgcAAABQKGIHAAAAUChiBwAAA
FAo1ZUeoD1obm70mjVr0rt371RVVV6HAAAAOCPlEqlbN6U997YbYkWDUqmSmjU9mjUZPHHhwpccAAAAA/oT
Vq1fnwAMPfM9jxI4kvXv3Tvf3TvJf/8Fqa2sr5POAAAADwx5qamjJ48ODy3/DvRexIym9dqK2tFTsAAACgHXs/t
1IAAACgUMQOAAAAAoFDEDgAAAKBQxA4AAACgUMQOAAAAAoFDEDgAAAKBQfPQsABWzatWqbNy4sdJjtKl+/fqloaG
h0mMAABSa2AFAARaxatSojRhyN4bN4odAAAB7kdgBQEEXVDjwofRuGV3oMAAAAKxA1KAQAAgEIROwAAAAIBCETsAAAAIBCETsAAAAIBCETsAAAAIBCETsAAACAQhE7AAAAgEIROwAAAAIBCETsAAACAQqlo7HjwwQfzmc98JTG98JoM
GDUpVVVXuvPOFvu/9KUvpaqqqsXj1n1FNPbXHMq6++mrPOiu1tbXp06dPmtPOiu1tbXp06dPzjnnnGzZsqV1mLzUsAUAAADQnlQ0dmzdujUnnHBCFr7mJnJqV
jVHHXVUrr322ix55zG2tRTT83TnF/rPOOitPOPfVUFixYVFFixYkPYkF/+8pd1Dxc9555UEHc9555+3tOQEAAIB2qrvqSP3z
ChAmZMGHCEx5TU1OTAQMGvO2+Jz/x5++x+2Tr/6/77/8/+ugQYNafWYAAACArmo1jW/qANX7AEHqB2qrqSP3z
+x44OEHOr9//wfPjwXXHBBBXnnllfK+K+yo5dsmTJkngd9zh07qSpqanAFAwAAACiGdh0
7Tj311NtkyKyYkDfffDNsm7duvTv37F/RXV6dv375Zt27Mt27duz7v7Nmz7U1dXV34MH
jx4r740AAAAoO1U9GOsf8oZZ5xxR/vdHPvKRjBjsv8IfzwAMPZOzYsbv9vDNmzV6vvep+P5L/uA1JbW9viAQAAAB
RDh4odL7/8cl555ZUMHDgwSYfM7SuQGnnHdbGv/vcWIjs6fERb+fOvnz3JnVrscrfwpfsb/Ey8kBbW9vi
AQAAABRDh4odL7/8cl555ZUMHDgwSYfM7SuQGnnHdbGv/vcWIjs6fERb+fOvnz3JnVrscrfwpfsb/Ey8kBbW9vi
AQAAABRDh4odL7/8cl555ZUMHDgwSYfM7SuQGnnHdbGv/vcWIjs6fERb+fOvnz3JnVrscrfwpfsb/Ey8kBbW9vi
```

3s2bNyqRJkzJgwIC88MIL+drXvpZDDjkk48ePT5IcdthhOfXUU/PlL385c+fOzc6dOzNt2rScccYZPokFAAAAO
qmKXtnx6KOPZtSoURk1alSSZPrO6Rk1alSuuOKKdO3aNU888UT+7M/+LIceemjOOeecHHPMMfn3f//31NTUlJ/
jn//5nzNixIiMHTs2n/70p3PiiSdm3rx5lXpJAAAAQIVV9MqOT37ykymVSu+6/9577/2Tz9G3b9/cdNNNrTkWA
AAAOIF1qHt2AAAAAPwpYgcAAABQKGIHAAAAUChiBwAAAFAoYgcAAABQKBX9NBYAAOioVq1alYObN1Z6jDbVr1+
/NDQ0VHoMgD9J7AAAgA9o1apVGTHisGzb9nqlR2lTPXvuk2eeWS54AO2e2AEAAB/Qxo0bs23b6xl99szUDjyoO
uOOiaa1L2bJdbOyceNGsQNo98QOAADYTbUDDDDOrfhuGVHgOAP+IGpQAAAEChiBOAAABAoYgdAAAAQKG4ZwdOAD7
aDgAA4POTO6Cd89F2ggcAAPDBiB3Qzvlo07EDAAD4YMQO6CB8tBOAAMD74walAAAAQKG4sgMA2tjy5csrPUKb2
rFjR2pqaio9Rptyk2UAqCyxAwDayLZNrySIPTGjRpUepW0VSpkkOnXq2TgPE+/Ex7k+x0sQ4rmfZ1vK2aZWmYvKc
rPlH/4Zx/W1fU8xV/ubGR5G4QztWsYfJczTVpqT5X8akVXbXo9vfGdSQ9AoPo8axuV6o2OFjtjy+X1GRJFRe8SsdU
BQOdUepw2sfbJxVl217xO9ZrdZBkAKk/sANqtznapv8veO49e/Rs6zQ2Hm9a+mKRzvWYAoPLEDqDd6ayX+rvsH
QAAWofYAbQ7nfFSf5e9AwBA6xE7gHarM1723pneutOZXisAG1L7ABoBzrrW3eSZOeONyo9AgAABSN2ALQDnfG
tO299SseuXbsqPQoAAAUjdgCOI53prTtvfUoHAAC0ti6VHgAAAACgNYkdAAAAQKGIHQAAAEChiBOAAABAoYgdA
AAAQKH4NBYAgL1g+fLlR6hTfXr1y8NDQ2VHgMAkogdAACtatumV5IUMmrUqEqP0qZ69NwnK55ZLngA7ZGQ8S
AtKKdr29OUspHv3hZDhg6otLjtImmtS9myXWzsnHjRrEDgHZB7AAA2At69W9W9W9I34bhlR4DADolNygFAAAAAA
AAAApF7AAAAAAKXewAAAAAAAKXsAAAAAApF7AAAAAAAKZbdix8EHH5xXXnnlbdsbtz8MEH7/FQAAAAALurene
+6cUXX8ybb775tu07duzIH/7whz0eCgFAAaJ+WL19e6RHaVL9+/dLQ0FDpMYAP6APFjrvuuqv873vvvTd1dXXlr
998880sXLgwBx1OUKsNBwAAtA/bNr2SpCqTJ0+u9ChtqmfPfPFffLMM8sFD+hgPlDsmDhxYpKkqqqO2bG2xe2
u0eigg3L11Ve32nAAAED7sPP1zUlK+KeepxSy5blY2btwodkAH84FiR3R9KvX9/rTGAAPCz3BTRWn3kBtwodkAH84FiR3Nzc5Jk6NCheeSRR
9KvX789+uEPPvhgve972Xp0qVZu3Zt7rjjjjnJQSZJSqZSM2fmJz/ZSM2fmJz/5SRobG3PCCSdkzpw5GTZsWPmYV199Rd
eeGHuvvvvud0nSJZMmTcoPf/jD90rVa49mAwAA3q5X/4b0aRnZd5H3w+5BNUyZ5XpOqiVjcu2ll
77j/u9+97h55pprMnfu3PIXZ0NL+W2MCxZsiT3339/zjzzzHz84x//0L9hfvsjceOF27dX/ju+97A6KfoHkVqFycQ4/Ix85DjIkBN6pA51/1rOp7iL33/hRFqkwQkWc+9OeMNHLm3tdYkAHDE3LGlrRYbmnHEXN9B1Fm1hubk5Ypw/L3tS0+4xsyJvUwQ+Zy+pGQjYqAg8HUmkzhM/t36rQvEs6+U
ka9sRuW5QzBEqRHz+Rd3UjgL/WhhBBSjxFRJKCxG6NwYWlFUdWDVTmhTzMX1r9/hL8KvRhPI2kIdmFTKlZR9/k9Qpl3bZB7kc
6WtWrIwJgCkrmSQANJ0k2XHX4KuWlxW/W3ZSRR4ecHgW7mmKNLpOWDTRT+BSgBYF8LvyLLL8v9gvyTB+9j3+Pxbh5O3bx7bnK0cPOR+QlEbs6M/z9gHUHjD8H/Uzw1gS1t6lpgRRPm/+/SYR3FMRAuYA8WH1lO9/IqOkB4sOPeQUvXO80E60b3zNtDMV7yI+ATJyKcXldJSexIrKE0BJdUUt85PqgAtbvdkHgzWvFtJxJLvI9dGDHY1QO22L15J5RjEgAB1PcvLvzR6Iw3UzlIyHEv8v2FmF0VUPBNRkkjA6N/y5vD3nuM6Y7wBfqhn4ALYOLEeWOh/ikt0GXyOD3s9PhkIiZ2V7uLFMkpOXy3UOsh4qZd2R6+L8vdWp69p7/Dqff5mUBnnF1/IqrLDvFVhbWkIAPpIKXPLXhx78dNIbDXw0B5yq9Q9RZFr2K/MfPXi2QSczb1Vq7QPLXiXtPd7LPhWkAe2wQnVDBHNdoPCyUbMDMIN5Qfy1fnXQ7gLZP7yvQSzyq3tnE/iQuQknhOvWtPObXNILGTMbdNPq63GjAgiUmfuHx7oSvrPPrFG/1q7vUKUVSSGn7AkiuHPX3R1AnrWjh0Tw3wnsMIwOXhXP4J7hHFdRsAq1/fkj05dpnjuvBmEp6H91xRN1HKKHXG/TdjrlUDdEnI/GuwUeeCdfLXHb8xn2FDEl9Ws8ogPxTfTsp3Zc7zIHcPEPjHMHfR/AGkRMSMqdK9tcRWnA8F3TWfXAtdXy7wpXCRxJcaW0hH0NSQ8MqhG8vVpb+Q5ZxJX+28aA/yx8BQ
pNdOjYVgQUZwBmVvHufMMl3yKWxiTFrtHuOH5QHDOfH6h
d4bnSmXfLNLMKdLvG0GGsa+l9qbmbQKBS3UrNBj6VDHwhvOGWSkVnq0+Vl5Ir1/tLiiMnMEoffqfMnP/DvgEF3Q3HCT8ptEA3FS6dqlj4jRwDN0KAw2hGpA17O+gOE1CvlR7QMfqPMuP/GrFgl/NLaQPgyrqJ1V+AfEdTg8tN/oVr6Da5B0mCmXEuhnQcuMAxYPOxlv6k+bLMZ9YgvoPrfN3HVgTuf9j9gT7Mj9sjxh9SHKqZMNRA4l0UJ5R8dZVsr+Jt5bi6xDUOcL+RfKX0hb75hfDYzx0FkpjAeoT3ybEGHZVePDEv9CPDbjnJ1+Ef48PeqEbd2Dw5yzNmM96tYx2PR2YvlWufvKbLq9hOk5kxHMEp21KNs0Ft5zH0Ds4pUp+9SQD0Y2wqjMNUMnEFUcS9vPLaPEGeUKNqzcXLHtWWHe3LP3mqnA4eLUxAuoVyi4n1H4QZC2yd5F8kkwHL4aJUNzg4Yxqs/5nlYDpb1pX03V1VbKOxNTwVwpnDUr4v0L4oJLXpbBKlvhnQWG0hp/NzuPN8fwYgsLcnwmVEXQ33N4u/UL3Zak+JEb0iGd2jIYj8aXQNDyyxkMO6aPWPCRUftMlyIuHVKoeQIBbPmyvPwi8xPCW8YRDP9CKUl+ZOeN0zh9o6VmVFRWqZw

wAAACgUsQMAAAAoFLEDAAAAKBQ3KKXDWbVqVTZu3FjpMdpMZ/uMcwAAgDOldtChrFq1KiNGHJZt216v9Chtbue
ONyo9AgAAQIcgdtChbNy4Mdu2vZ7RZ89M7cCDKj1Om1j75OIsu2tedu3aVelRAAAAOgSxgw6pduBBneZzv5vWv
ljpEQAAADoUNygFAAAACkXsAAAAAApF7AAAAAAKRewAAAAACkXsAAAAAApF7AAAAAAKRewAAAAACkXsAAAAAAp
F7AAAAAAKRewAAAAACkXsAAAAAAqlXceOK6+8MlVVVSOeIOaMKO/fvn17pk6dmv333z+9evXKpEmTsn79+gpOD
AAAAFRau44dSXLEEUdk7dq15cdDDz1U3nfppZfm7rvvvzu23355FixZlzZo1+dznPlfBaQEAAIBKq67OAH9KdXV
1BgwY8LbtmzZtyvz583PTTTflU5/6VJLk+uuvz2GHHZbf/va3Of7449t4t9tzVAAAAKAdaPdXdjz33HMZNGhQDj744
Jx111lZtWpVkmTp0qVZXXjTp0uqXVFyNnxoObVz52xIgRaWhoyOLFiys1LgAAAFBh7frKjtGjR+eGG27t2syaNSmR
nnXRSli1blnXr1qV79+tpuqXVWDPUdMHcTcOPjgHe1dShdMOFmT+dfJp6mdpZrmVeAuCMuWMyyRkvBBzVmJnW8nTqn
o8enSFDhuS2225Lz549d/t5Z8+enVmzZrXGiAAAAAEA7O+7fxvLf9enTJ4ceemief/75DBgwIG+88UaGxtbbHLN
+/fp3vMfHfzdjxoxs2rSp/Fi9evVenBoPBCBg4cmGOOOSbdunXLwoULy/tXrFiRVatWZsSUatWZ
cyYMe/5PDU1NamtrW3xAAAAAIqhXb+N5atf/Wo+85nPZMiQIVmzpkzmpyZrl275swzzOxdXV3OOeecTJ8+PX3
79k1tbWOuvPDCjBkzxiexAAAA8IEsX7680iO0qX79+qWhoaHSY3HSY+w17Tp2vPzzyzzxqiuv5IADDsiJJ56Y3
/72tznggw4AOSJH//93fL126ZNKkSdmxYx0cfGjx+ff/zHf6zw1ALCxs
82nXsuOWWW95zf48ePXLLttdfm2muvbaOJAAAAKJKdr29OUspHv3hZDhg6otLjtImmtS9myXWzsnHjRrEDAAAAi
qpX/4bObRhe6TFoJR3qBqUAAAAAf4rYAQAAABSK2AEAAAAUitgBAAAAFIrYAQAAABSK2AEAAAAUitgBAAAAFIr
YAQAAABSK2AEAAAAUitgBAAAAFIrYAQAAABSK2AEAAAAUitgBAAAAFIrYA
QAAABSK2AEAAAAUitgBAAAAFIrYAQAAABSK2AEAAAAUitgBAAAAFIrYAQA
AABSK2AEAAAAUitgBAAAAFIrYAQAAABSK2AEAAAAUitgBAAAAFIrYAQAAA
BRKdaUHYPetWrUqGzdurPQYbWr58uWVHgEAAIB2TuzooFatWpURIw7L7tm2vV3qUiti5441KjwAAAEA7JXZOUBs
3bsy2ba9n9NkzUzvwoEqPO2bWPrk4y+6al127dlV6FAAAANopsaODqx14UPo2DK/0GAAAAAAEFAAAAAAAFIrYAQAAABSK2AEAAAAUitgBAAAAFIrYAQAAA
6dB5++OFKjwQAAABUQCFix6233prp06dn5syZ+d3vfpejjjoqDx48ePz4YNGyo9GgAAANDGChE7vv/97+fLX
uqv/iqHH3545s6s6dm3322SfXXXddpUcDAAAA2lh1pQfYU2h+88aWLl2aGNn1dWL9+dWo961dN3ybM2aAbcC

ReV1+w1F1WnfM3rViX5r78rO9LfwW/NWiqV/uSxVaX3c1Q7tmbNmnzoQx/Kf/zHf2TMDmDH17V/72teyaNGiLFm
y5G3fc+WV2bWrFltOSYAADQClavXpODDzzPY/p8Fd27I4ZM2+vTp5a9zvZv0L/Zvc1Ud4reU1NT
Rk8eHBWr16d2travumavfZZoL2x2x90umsrH06K2ufzsrapzNqy3VfKpWyefPmDBo06Ba7Dm1wJ/9/6/fdwIAB7/g9NTU1atvqampT58+xfb
169dnwIAB7/g9NTU1atvqampT58+e2vEt6mtrXXyo1Oy9umsrH06K2ufzsrapzNqq3VfV1f3vo7r8Dco7d69e
4455pgsXLiwvK/K25uTkLFy5s8bYWAAAAAoHPo8Fd2JMn06dMzcqUqHvssfnfY2JMn0qGZWvMm4gnvvnb3Tv3
aUU07J1q1by8dceumlufvv3P77bdn0aJFWbNmTT73uc9VcFoAAACgUoGHd4E2LjokXl26Ft9aeaNAxaSXXEt9A9vbX7

HhxRWaE1vTmm2/m1ltuydatWzMmBhrnk5h6tSp0e2001qs88Q5n2J77rnnnnMMmjQoBx88ME566yyzcmVyzsmrVqiTWPcV
211135dhjj83nP//590/fP6NGjcpPfvKT8v6VK1dm3bp1LdZ/XV1dRo8ebf1TbbGG+88UZ+9r0f5eyzz05VVVVW7P
0+LHXtZc3NzLrnkkpxwggggk58sgjkyTr1q1L9+7dO6dPnxbH1tfXZ926daV3t6TdRWYE1rHkO8+mVo69eqWmpiz//ddnrsnt
uuCOHH364NU/h3XLLLfnd736XKl2gXmcXffl7hmR9J9Ujfcs7nDp2hPnUIOeheeedXiXy1a35cKoD9tzXgm5/iLRi/82
TMnw4YNy7333333psLLrggFrl0UX76ef8R2gJ0yt/P005PYfe3nuM7PFVMbRznc3SYO9hVvBnqcnnoeeHNnjZ5nm
yFu9hhaIaPnx4Hnw4HnwssWzatCk///nvVnPC2AKlCxqtKjSY8FetXr16xx8cVZsGBBevTEoUelxoM1MmDDcAh/0+RIQmd9
OjRGTJkSG677bO07NmzgpBP3XCC3Jxx7jJ0zZ3lt0J0Kl4jBNDPgmAwr7mt1h8Y0jDxL0iFilsXPbn7ZPk34mTjiBQYMmfVXq
Ud+XKjr1o2rRp+eUvf5nf/OY3OfnCONjY0tjl+/fn0GDBjQx1NC6+nevXsO0eSQHHPMmMMZk9e
3aOOuqo/PCHP7TmKbs1S5dmw4YNOfroo1NdXZ3q6uosWrQo11xzTaqrq1NfX2/9U2zx+w7G2WCvo+aL0zHAGMmjNlE0qN0M2oA5Hp2PD4a+zHuGVfhx5WUdH0uQ1i3lRQQZvl8L1ZHrmZyeG0uGxBeucLmWQBxbHj+V6mH8QuzWuMEXY2QE0DbN2Wr5fZ1s/w0O/mK1cZYO0Jrl3ZDQkaKXPqVuzwebsnyM//QlfCEKlSG06u9CAM5OHwdUUZYCzoUZaq6/EPDZm7DtSaq9Frq10sB9jkKpVdfB0aW3J9b6e3i5YL5T7tTYJ1fwIKTT/9OB0U0Fnc5g2NZ1PvDeYGqkZJM2SjEpPBM16XvqwZqQnqzaOkEzQ8Rmjmt4q75xHd89fl9n9sAE5G2v3s10z5YnJYYDZ2uBqUjyk8qVRvnj/KAzUA1uMcadpq5lnVqXoeRbdDq7k6/mGgMX0PZXYTF9sYznBIUSNgXuLJoqbtUqeB9fGc8xfaXcyv8OfDwPvDK3z9B6ZQ4zqPTDlg2ybY/VZNt+C38uaSJXVP2YV4eOCGu4C6bA3y1btRf4mVDR4s+ELXmkMtu58vSpnrvxpVM6SkmqfsO4/7iIlgG+1YY35+1OMsL6M9sOEgXzu/AuRkd6Nqw5WE3ODHY+WeUwsvPSbBbNWO7

t4MCBOfzww1tsO+yww8pv43prjf/xp1BY/xTFSy+9lPvuuy/nnntueVt7PO+LHXtBqVTKtGnTcscdd+T+++/PO
KFDW+w/5phjOq1btyxcuLC8bcWKFVm1alXGjBnT1uPCXtPc3JwdO3ZY8xTa2LFj8+STT+axxx4rP4499ticddZ
Z5X9b/3QGW7ZsyQsvvJCBAwc671NoJ5xwQlasWNFi27PPPpshQ4YkSYYOHZoBAwaOWP9NTU1ZsmSJ9U8hXH/99
enfv3900+208rb2eN73Npa9YOrUqbnpppvyL//yL+ndu3f5PXXp2bNn6urfqLcs4552T69Onp27dvamtrc+G
FF2bMmDE5/vjjjKzw97J4ZM2ZkwoQJaWhoy0ObNm3PTTTflgQceyL333mvNU2i9e/cu35PnFPpLfvuu2/233//8nbrn
yL66le/ms985jMZMmRI1qxZk5kkzZ6Zr164588wznfcptEsvvTQf//jH8+1vfztf+MIX8vDDD2fev2ekqS
qqiqXXHJJvvWtb2XYsGEZOnRoLr/88gwaNCgTJ06s7PCwh5qbm3P99ddnypQUpqa7+fzmhXZ73K/IZMAWX5BOf1
19/ffmYbdu2lf7mb/6mtN9++5X22WefOp//X2dW7mhYQ+dffbZpSFDpS6d+9eOuCAAOpjx44t/du//Vt
5vzVPZ/LfP3q2VLL+KabTz+9NHDgwFL37t1LH/rQh0qnnn3566fnnny/vt+4psrvvvvt05JFHlmpqakojRowoz
Zs3r8X+5ubmOuWXX16qr68qr68v1dTUlMaOHVtasWJFhaaF1nPvvfeWkrzem5v/2qUqlUqkxmAQAAAGh97tkBAAA
AFIrYAQAAABSK2AEAAAAUitgBAAAAFIrYAQAAABSK2AEAAAAUitgBAAAAFIrYAQAAABSK2AEAdCiLFy9O165dc
9ppp1V6FACgnaoqlUqlSg8BAPB+nXvuuenVq1fmz5+fFStWZNCgQZUeCQBoZ1zZAQBOGFu2bMmtt96a96aC4IKe
ddlpuuOGGFvvvvuuuuuDBs2LD169MjJJJ5+cn/7Op6mqqkpjY2P5mmIceeignnXRSevbsmcGDB+eiiy7Klq1b2/aFA
AB7ldgBAHQYt912WO0aMGJHhw4dn8uTJue666/LWRaorV67MX/zFX2X2i
pp2bSpEl54okncuutt+ahhx7KtGnTKvFyAIC9xNtYAIAO44QTsgXvvCFXHzxxxdm1a1cGDhyY22+/PZ/85Cfzt
3/7t/nXf/3XPPnk+Xj/GNb+Sqq67Ka6+9lj59+uTcc89N165d8+Mf/7h8zEMPPZS6/v+FGJlwtDJXdgAAHcKKFSvy8MMP58ehpg3tPh8sEqqs
sAEBliROAQLu3a9eu3Hjjjn66qtzyimnt9Ng3ceLE3HzzzzRk+fHh+9atftdj3yCOPtPj66KOPztNPPP51DDjlkr
88MAFSOe3YAAO3enXfemdNPPPz0bNmxIXV1di32XXXXXZZ7r///t220ZPnx4Lr300pxzzj157LHH8pWvfCCUvv/x
yGhsbU1dXyeeeCLHH398zj777rZ5Zd9998/9998/BggXv++oQAKD9c880AKDmzzmdmz7z5J//6L/3b/2hNPZ/ZNy
qOPPprNmzfn5z//eX7xi9l9k5MiRmTTNnTvnTWGpqapIkI0OezKJFi/Lss8/mpJNPNOyqhRo3LFFVeOeGsMANDxubI
DACisq666KnPnzs3q1asrPQoAOIbcsyNJc3Nz1qqxZ969e6e6eqqrS4wgAAu+knP/lJjj766PTt2zdLizJd7/73
Zx3nlpamqq9GgAwB4q1UrZvHHlzBg0a0C5C5d3vuNKq7sSPLyyy9n9n0n8ODBlR4aR4DAAAA+BNWr16ePvWVHkt6
9eyf5r/9gtbW1W4J4GAAA+GNNTU1ZsmSgAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA+GNNTU0ZPHhw5qAAAA

AAAUChiBwAAAFAoYgcAAABQKGIHAAAAUChiBwAAAFAoYgcAAABQKGIHAAAAUChiBwAAAFAoYgcAAABQKNWVHgC
AttFl+6ZKjwDvqcu2xhb/C52F8zNA6xM7AAqurq4u3brXJL9fVOlR4H3pufLBSo8Aba5b95rU1dVVegyAwhA7A
Aquvr4+P/s/N2bTJv/PIUB7VVdXl/r6+kqPAVAYYgdAJ1BfX++XaAAAOg03KAUAAAAKRewAAAAACkXsAAAAAAp
F7AAAAAAKRewAAAAACkXsAAAAAApF7AAAAAAKRewAAAAACkXsAAAAAApF7AAAAAAKRewAAAAACkXsAAAAAApF7
AAAAAAKRewAAAAACkXsAAAAAApF7AAAAAAKRewAAAAACqW60gOOB6VSKUnS1NRU4UkAAACAd/LW3+xv/Q3/XsS
OJJs3bO6SDB48uMKTAAAAAO9l8+bNqaure89jqkrvJ4kUXHNzc9asWZPevXunqqpqr/2cpqamDB48OKtXr05tb
e1e+znQ3lj7dFbWPp2VtU9nZe3TGbXlui+VStm8eXMGDRqUL3e+64cruxIOqVLlxx44IFt9vNqa2ud/OiUrHO
6K2ufzsrap7Oy9umM2mrd/6krOt7iBqUAAABAoYgdAAAAQKGIHW2opqYmM2fOTE1NTaVHgTZl7dNZWft0VtY+n
ZW1T2fUXte9G5QCAAAAheLKDgAAAKBQxA4AAACgUMQOAAAAoFDEDgAAAKBQxI69YPbs2TnuTnuuOPSu3fv90/fPxM
nTsyKFStaHLN9+/ZMnTo1+/f3r16pVJkyZl/fr1FZoY9ty0Mycu0TI1NbWpra2NmPGjMmvf/3r3r8n5rns7i0
9/5TqqqqnLJJZeUt1n/FNGVV16ZqqqoFo8RIOaU91v3FNkf/vCHTJ48Ofvvvvvvvvvvvvvv3969uyZj3zkI3nO0UfL+0ulUq6
44ooMHDgwPXv2zLhx4/2zLhx4/Lcc89VcGLYcwcddNbzvtVVVWZOnVqkvZZ33hc79oJFixZl6tSpe+e1vf5sFCxZk586dO
eWUU7J169769byMZdeemnuvvvv3H777nb6H0ctQ20+NXqzUU5/KZz/72TZ/Z35ZZZ/L34LO4+z1n369o24gtpSQ0lXY//8g
11FNJrHk6hOceeSQ//vGPM3LkyBbbrX+WtatWvLj4aGxxi8z7qnqF577bWcM7TNUTWWLLj4ceeqi8z7tqqnqF5+7W7Gc96Ma1W3796/n6aefz
tVXX5399tuvfMx3vv/vdXHPNNZk7d26WLFmSfffdN+PHj8/PHj8/27dsrODnsmUceeaTF0X/BggVJP
et2HDhlKSOqJFiOqlUqnU2NhY6tatW+n2228vH+vu4N/4j/U227Lh73/T/0T9Y8ncLmzZtLw4YNK
y1YsKD9iU98nreXSiXnfIpr5syZpSOPPPKd3dPfaaUyyyoZpaOOuod9n3FNllll1WOvHEE991f3Nzc2nAgAG1733ve+Vtjq
amtLNN9/cFiNCm7j44otLH/7whw0OaMSENDQxYvX
lyRGaE1vfnmm7nllluydevWpOO+2J577kMGjQoBx98cM4666y6ysWrUqiXVPsd1
111059thj8/nPfz79+/fPqFGj8pOf/KS8f+XKlVm3bl2L9V9XV5fRo0db/xTSc9Pp/xTGG2+8kZ/97GcG+3d8UVW7NgUVyVBPOwS7P+
2LHXtbc3JxLLrkkJ5xwQo488488488sgkybp169K9e/f06d0nxbH19fVZt25dBaeEPTdRnV/vHkkOt+mV69eqampyV//9V/njjj
uyOGHH27NU3i33nJL/vfvlSUPjR4/u732X/3VczR0dZNzI4onq9NU3i9zh5Jj7rm63v3bSaNUgcpMD2Y0gggzalepNTWd3Y7Nz9SHzS8CuV9EkXFZZ1LkJYGuuWQ69dFRxKmlXvx2i1efjZCmdJHbkaFCSbWvftyoBSWf1nOPYTrpm8aTkM3g+QQrzRo82JsoU90u9zBT7+tTEyrbu0pTs0AW4G6bGHIzmXQIb0PlaJexmKejCxz5OKzDIoGVjS8hlpBm60Hj6Nt2dzBVIHi6H6QaYA86TZIdFLkqJ8ekH6xhrJrHzFXyNxDGCk6G6HAI7r04a8eUnFYn+o9QDaWbT8/9nHHrdPR3Bp3zBePa/HeXtMexHgiGMVwDvuJX2PvO5HXu/PZMVv7tL2YAkZlCepIE7UeYZnX7lUz7nYMKjBR3zsVCa8j4EsUZRsf7Dx5+LyBa08YMS5W8/Lyv9gmfwP4Tq6x8xC8Jc1KdUOc/zP1IVxYvlvAR8Ve7Y9K44JkETHxOBGn9/3yfwF8WHGyWiWCMaWBbE3DLRC9EoNvHp1G4vKKtVjOAsAmz/eAD69Uy5sn1HAICxhL66gbnkcn3UnhcfyEJSecfv0nOGBM+Bf+k+mpvFp/5WhhNzzZyVXUZHFF7BhRJxzNjnLLnVaaaNgwK/YSpZeSfiZq6aaPAnOu3bO
AAAUChiBwAAAFAoYgcAAABQKGIHAAAAUChiBwAAAFAoYgcAAABQKGIHAAAAUChiBwAAAFAoYgcAAABQKNWVHgC

```
ACuuqq67K3Llzs3r16kqPAgCOIffsAAAK4x//8R9z3HHHZf/998///b//N9/73vcybdqOSo8FALQxsQMAKIznn
nsu3/rWt/Lqq6+moaEhX/nKVzJjxoxKjwUAtDFvYwEAAAAKxQ1KAQAAgEIROwAAAIBCETsAAACAQhE7AAAAgEI
ROwAAAIBCETsAAACAQhE7AAAAgEIROwAAAIBCETsAAACAQvn/AWrSkoycAYSkAAAAAElFTkSuQmCC\n"
          },
          "metadata": {}
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "### Categorical Column Analysis with approval percentage of categories"
      ],
      "metadata": {
        "id": "6J1e86BqH6PI"
      }
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "Or91emUB2Guj"
      },
      "source": [
        "#### Type_Occupation"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "QqwiGvyL2GGI",
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 612
        },
        "outputId": "9f32d394-56a6-4c61-c2d8-58257e5473bb"
      },
      "outputs": [
        {
          "output_type": "stream",
          "name": "stderr",
          "text": [
            "WARNING:matplotlib.text:posx and posy should be finite values\n",
            "WARNING:matplotlib.text:posx and posy should be finite values\n"
          ]
        },
        {
          "output_type": "display_data",
          "data": {
```

```
          "text/plain": [
            "<Figure size 1500x500 with 1 Axes>"
          ],
          "image/png":
"iVBORw0KGgoAAAANSUhEUgAABMIAAAIwCAYAAABk/ceIAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAAD/EEl
EQVR4nOzdd1yV5f/H8TdLEGUqwwFuceDWFHfkJrK03JrmyhmaA02UNAeapuLKPXHlKKxMc+ZKpRyYinvjBlRUE
M7vD3+dOl9H4AjtvJ6Px3nkua/r/tzXdctD6911X7eFwWAwCAAAAAAAAPiPs8zoAQAAAAAAAAD/BoIwAAAAAAA
AmAWCMAAAAAAAAJgFgjAAAAAAAACYBYIwAAAAAAAAmAWCMAAAAAAAAJgFgjAAAAAAAACYBeuMHsCzSE1N1cWLF
+Xg4CALC4uMHg4AAAAAAAykMFg0K1bt5QzZ05ZWj553ddrGYRdvHhRXl5eGT0MAAAAAAAAvEL0nTun3LlzP7H
9tQzCHBwcJD2cnKOjJwBwcJD2cnK0jYwaPBgAAAAAABkpISFBXl5exszoVSZ7LI0zPxyEdHR0JwgAAAAAACBJ/7iFFpvlAwAA
AAAwCwQhAEAAAAAAMAsEIQBAAAAAADALLyWe4QBAAAAAIDXV0pKipKTkzN6GHhN2NjYyMrK6rnrEIQBAAAAAIB
/hcFgUGxsrOLj4zN6KHjNODs7y9PT87le1EgQBgAAAAAA/hV/hnDu7u6yt7d/rlAD5sFgMOjOnTu6cuWKJClHjhzP
44cz1yLIAwAAAAALx0KSkpxhAsW7ZsGT7sSbpGdnsSbpy5Yrc3d3f+qMMBIZgFAAAAAAC/vXn53n21n2xsd0Zn
6+vNn53n2lyMIAwAAAAAA/xoeh8SzehE/NwRhAAAAAAAAMAsEYQAAAAAAADBRs2ZNBQUFZfQwXjiCMAAAAAAA
Blm5dFL/+r1Gvk82xsHd+7cqapVq6evXr6/vvvX/Co8G9hRRgAAAAAAMA/mDVrlnr06tr6n/16SU1l
JL/0a5oggDAAAAAAA4Clu376tpUqXqkuXLgoICDcuXONbs3b5aFhYW+//57lSpVSnZ2dqpcubIOHjxovOfNbLmh
StnZ2etXr1lahQoVkp2dnamxxbj3vVzTMbGkdk58nH5Fr3W5NxbfzfF1L/rzYlYJ3o+77ucxhWRYTSWpm/uJpZT
HDAaDbG1tNWnSpBd6rMtBzdSKaNbKuXeDKXnQpT/l2F2iZl+T61lQmR2JPQrW1XCvr6f3rMBnpcGEpCH8MJPkJ






































































































```

```
4w1UlJStHTpUrVs2VKSdOLECdWrVO+NGzfWgQMHtHTpUm3bts1YQ3oYwp07d06bNm3SN998oylTpujKlSv/8Dv
x/NIVhIOcOVIVKlSQg4OD3N3d9e677+roOaMmfWrWrGlMNv/8fPzxxyZ9zp49q4CAANnb28vd3V19+/bVgwcPn
n82AAAAAAAAL8F7772n0qVLa8iQIY9tHzdunN566y2FhISocOHCatu2rbp3764xY8ZIklxdXWVlZSUHBwd5enr
K09PzidcKCgrSm2++qbx588rf319ffPGFli1bZtInOTlZkyZNkp+fn8qVK6d58+Zpx44d2r17t7HPvXv3NH/+f
JUuXVrVq1dXeHi4lixZotjYWGOfLFmyaObevLiKFy+uiIgI43m+vr7y9/fXpEmTtGDBA12+fFlWVlZq1qy
ZIiIijDU2bNiguLg4NW7cWNLD/Khly5YKFgpSoUKFVLLlyZU2cOFHz58/XvXv3FBMTox9//FEzZsxQpUqVVK5cO
c2aNUt3795N/29MOqUrCNuyZYu6deumXbt2af369Up0TladOnVMlrZJUseOHTlSOfeOHY3p5qVLlzR69GhjWOpKigICApS
UlKQdO3Zo3rx5mjt3rgYPHvxiZZgQAAAAAAPAShIWFad68eTp8+PAjbYcPH1aVKlVMjlWpUkuq7z8
88/6623lKuXLnk4OCg1q1b6/r160pMTDT2sba2VkKJ
ECWXKlMlkHqVKlVKWLFlM5vH381q2bKnNmzfr4sWLkqRFixYpICBAzs7OkqT9+/dr7ty5ypo1q/FTt25dpaam6
tSpUzp8+LCsra1Vrly5R8b/sqUrCFu7dq3atm2r4sWLq1SpUpo7d67Onj2rqKgok3729vbGdNPTO1OOjo7GtnX
r1umPP/7QwoULVbp0adWvX1/Dhg3T5MmTzWJTNgAAAAAA8HqqXr266tatqwEDBry0a5w+fVPpvv/22SpYsqRUrV
igqKkqTJO+W9HI2s/974JVWFSpUUIECBbRkyRLdvXtXq1atMj4WKUm3b99W586WQVZ/fPuNn//79OnbsmAoUKPA
ih59uz7VH2J8brLm6upocX7RokbJnzy5fX18NGDDAJLHcuXOnSpQoIQ8PD+OxunXrKiEhQYcOHXrsde7fv6+Eh
ASTDwAAAAAAwL9t1KhRioyM1M6dO0O2OFy1aVNu3bzc5tn37dhUuXFhWVlaSpEyZMv3j6rCoqCilpqZq7NixqlS
pkgoXLmxcefV3Dx480N69e43fjx49qri7ffJLmwsZVdfU2nhAxgFz7qkOEF47YOZZsybn7tq1S5aWlsZN8R+naNGi2r9/v8nTf9u3b/kv
JYtW2rRokWKjIyUpaWlAgICjGly5bVH3/8oYIFCz7yyZQpk4oUKaIHDx6YLKz6c/wv2zMHYampqQoKClKVKlX
k6+trPN6iRQmZZt0oABA7RgwQK1atXK2B4bG2sSVmzzpsAAAAAACAZ
laiRAm1bNlSEydONDn+6aefasOGDRo2bJhiYmI0b948TZo0SX369DH2yZs3aoLFy7o2rVrj61fsGBBJSc
nKzw8XCdPntSCBQs0bdq0R/rZ2NioR48e+vXXXXxUVFaW2bduqUqVKeu0NN4x97Ozs9OGHH2r//v365ZdftLNnT
zVp0uSp+5O1bNnSeF50dLQ2bdqkHj16qHXr1iZ5TsuWLfXbb79p+PDhev/992vv2tLi2a5TsuWLfXbb79p+PDhev/992Vra2ts69+/v3bs2KHu3btr375
9OnbsmL799lvjZvk+Pj6qV6+eOnfubcx/hw4dlDz5n+4+8/P+1lP7Natm6Kjo7Vt2zaT4506dTL+ukSJEsqRI
4feeustnThx4pmXvvwOYMEC9e/c2fk9ISCAMAwAAAADgP+Dvi2teFOOHDtXSpUtNjpUtW1bLLi3T4MGDNWzYMOX
IkUNDhw5V27ZtTc7r3LmzChQooPooPv378tgMDxSu1SpUho3bpzCwsIOYMAAVa9eXSNHjlSbNm1M+tnb26tb26t///5q0
aKFLly4oGrVqmnWrFkQWLKhGjRqpQYMGunHjht5++21NmTLFeN5FnTooBo1asjDwDffPGFQ
jxj1S+4033tDu3bs1fvx4k7aSJUtqy5Yt+uyzz1StWjUZDAYVKFBATZs2NfaZM2eOOnTooBo1asjDw0NffPGFQ
kJCnjq2F8HC8Li7/g+6d++ub7/9Vlu3blW+fPme2e2vf0OnTvKmjWr1q5dq7p162rw4MH67rvvvtG/fPmOfU6dOKX/
+/Prtt99UpkyZf7x+QkKCnJycFB8fb7L/20OsWu0jo6o4eQbun5A+u/Pj8AAAAAwJPdu3dPp06dUr58wWRnZ5fRw
3mtzZ07VOFBQU99lDAONFSrV682yV9ed0o/7GUprVpSuRyMNBo06d++uVatWaePGjf8YrEhSdHS0JCl79uySJD8/PxeiGgoM
HD+rKlSvGPuvXr5ejo6OKFSv2LNLPGIPV3YwkVKW0yROY82ZG5F2SxiIMHQzAcsAnSmVK9+yjuHVe53VbJcBlKX4R4OhbGNo0zT4sNtBLlB8xp4+5ftpktwDUZdyh0Y+9GjVHIXiFc2NFfjtBj18
IMGDZQtWzYdOHBAvXr1UvXq1VWyZElJUp06dC9v+6GkXuUhUPnHFGsJMaONfFM0BxlI+Ak2hJU6PZ/H6MkKiUs48b6MdjHN4ZPx9jD7f9fAfPDyFk7c0kJS5HGQKDaxhRjZfGmIP3pN8JNKdc+tfbKxCrYLkSDrnxKHnbHKjE1pMaFBpnRuSxcOZZ8AdgzQzYp9Ya0Gk2VUd7fUqqYpHGg6
```

```
Iy8tL8fHxcnR0fNbhv3JWHr2U0UNIt0Y+0dLcl/m9etIzPwAAAAAAXlUJCQlycnL6x6zoufYIi4+PlyS5urpKk
qKiopScnKxatWoZ+xQpUkTe3t7auX0nJGnnzp0qUaKEMQSTpLp16yohIUGHDh167HVGjhwpJycn48fLy+t5hg0
AAAAAAAAz9MxBWGpqqoKCglSlShX5+vpKkmJjY5UpUyY5Ozub9PXw8FBsbKyxz99DsD/b/2x7nAEDBig+Pt740
Xfu3LMOGwAAAAAAAGbK+llP7Natm6Kjo7Vt27YXOZ7HsrW1la2t7Uu/DgAAAAAAP67nmlFWPfu3bVmzRpt2rR
JuXPnNh739PRUUlKS4uLiTPpfvnxZnp6exj7/+xbJP7//2QcAAAAAAAB40dIVhBkMBnXv3l2rVq3Sxo0blS9fP
pP2cuXKycbGRhs2bDAe03r0qM6ePSs/Pz9Jkp+fnw4ePKgrV64Y+6xfv160jo4qVqzY88wFAAAAAAAAeKJ0PRr
ZrVs3RURE6Ntvv5WDg4NxTy8nJydlzpxZTk50at++vXr37i1XV1c50jqqR48e8vPzU6VKlSRJderUUbFixdS6d
WuNHj1asbGxGjRokLp168bjjwAAAAAAAHhp0hWETZ06VZJUs2ZNk+Nz5sxR27ZtJUlfffWVLC0t1bhxY92/f19
169bVlClTjH2trKy0Zs0adenSRX5+fsqSJYs+/PBDDR069PlmAgAAAAAAADxFuoIwg8Hwj33s70w0efJkTZ48+
Yl98uTJox9++CE9lwYAAAAAAAACeyzNtlg8AAAAAAAC8bgjCAAAAAAAAYBYIwgAAAAAAAGAWCMIAAAAAAAABgFgj
CAAAAAAAAYBYIwgAAAAAAAGAWCMIAAAAAAAABgFgjCAAAAAAAAYBYIwgAAAAAAAGAWCMIAAAAAAAABgFgjCAAAAA
AAAYBYIwgAAAAAAAGAWCMIAAAAAAAABgFgjCAAAAAAAAYBYIwgAAAAAAAGAWCMIAAAAAAAABgFgjCAAAAAAAAYBY
IwgAAAAAAAGAWCMIAAAAAAAABgFgjCAAAAAAAAYBYIwgAAAAAAAGAWCMIAAAAAAAABgFgjCAAAAAAAAYBYIwgAAA
AAAAGAWCMIAAAAAAAABgFgjCAAAAAAAAYBYIwgAAAAAAAGAWCMIAAAAAAAABgFqwzegAAXh/R0dEZPYR08/X1zeg
hAAAAAABeEawIAwAAAAAAgFkgCAMAAAAAAAIBZIAgDAAAAAACAWSAIAwAAAAAAgFkgCAMAAAAAAAIBZIAgDAAAAA
ACAWSAIAwAAAAAAgFkgCAMAAAAAAAIBZIAgDAAAAAACAWSAIAwAAAAAAgFkgCAMAAAAAAAIBZIAgDAAAAAACAWSA
IAwAAAAAAgFkgCAMAAAAAAAIBZIAgDAAAAAACAWSAIAwAAAAAAgFkgCAMAAAAAAAIBZIAgDAAAAAACAWSAIAwAAA
AAAgFkgCAMAAAAAAAIBZIAgDAAAAAACAWSAIAwAAAAAAgFkgCAMAAAAAAAIBZIAgDAAAAAACAWSAIAwAAAAAAgFk
gCAMAAAAAAAIBZIAgDAAAAAACAWSAIAwAAAAAAgFlIdxC2detWBQYGKmf0nLKwsNDq1atN2tu2bSsLCwuTT7169
Uz63LhxQy1btpSjo60cnZ3Vvn173b59+7kmAgAAAAAADxNuoOw03fuqFSpUpo8efIT+9SrV0+XLl0yfhYvXmzXmz
S3rJlSx06dEjr16/XmjVrtHXrXrVnXq1Cn9owcAAAAAADSyDq9J9SvX1/169d/ah9t1t5eno+tu3w4cNau3at9
uzZo/Lly0uSwsPD1aBBA3355ZfKmTNNNneocE4F8SY5Mto4eQbr4ZPQAAAAAAwCvjpewRtnnzZrm7u8vHx0ddunT
R9evXjW07d+6Us70ozMQSTpFq1asnS01lK//vrrY+vdv39fCQkJh8AAAAAAAAgPV54EFavXj3Nnz9fGzZsUFhYm
LZs2aL69esrJSVFkhQbGyt3d3eTc6ytreXm5lhw5cqSnJjyMHy8vrxc9bAAAAAAAAPzHpfvRyH/SrFk
z469LlCihkiVLqkCBAtq8ebPeeuutZ6o5ZUMYAA9e7d2/g9ISGBMAwAAAAAADp8lIejfy7/PnzK3v27Dp+/Lgky
dPTU1euXDHp8+DBA924ce0J+4rZ2trK0dHR5AMAAAAAAACkx0sPws6fP6/r168rR44ckiQ/Pz/FxcUpkKirK2Gf
jxo1KTU1VxYoVX/ZwAAAAAAAAYKbS/Wjk7du3jau7J0nUqVPat2+fXF1dereqqs8//1yzYNfx8oX79+
qlgwYKqW7euJK1o0aKqV6+e0nbsqGnTpik50Vndu3dXs2bNeGMkAAAAAAAXpp0rwjbu3evypQPBqpataqmT5/+4
urTJlymjw4MGysrLSgQMH9M4776hw+hw4cJq3769ypUrp19++UW2trbGGGosWLVKRIkX0l1tvqUGDBqpataqmT5/+4
mYFAAAAAAAA/I90rwirWb0mDAbDE9t/+umnf6zh6uqqiIiI9F4aAAAAAAAAeGYvfY8wAAAAAAAA4FVAEAYAAAA
AAACzQBAGAAAAAAAAAs0AQBgAAAAAAALNAEAYAAAAAAACzQBAGAAAAAAAAAs0AQBgAAAAAAALNAEAYAAAAAAACzQ
BAGAAAAAAAAAs0AQBgAAAAAAALNAEAYAAAAAAACzQBAGAAAAAAAAAs2Cd0QMAgFdFdHR0Rg8h3Xx9fTN6CAAAAAD
w2mBFGAAAAAAAAMwCQRgAAAAAAADMAkEYAAAAAAAAzAJBGAAAAAAAAMwCQRgAAAAAAADMAkEYAAAAAAAAzAJBG
AAAAAAAAMwCQRgAAAAAAADMAkEYAAAAAAAAzAJBGAAAAAAAAMwCQRgAAAAAAADMAkEYAAAAAAAAzIJ1Rg/A3KW
kpCg0NFQLFy7UxUuxcnH30JvvNdH7XYJkYWEhSYq7dlULvhyu/du2kjPkv/du36M6teBUrX0ntB32hnHnzP7HuxpVLNXlgL
5NjNplsteTAKeP380AgbV69zKRP6ao1FTIzQpKUnHRfUwb10Z4NP8k5u7s6DhmhUpWrG/uunjVFm+7GKTw8/Ln
vAwAAAAAAwMtGEJbBwsLCNHXqVM2bN0/n7Nx0Inq/Jg3sJfusDhDgpo00EGg0Fh3T6SlY21gqfMUeYsWRU5d7o+/
6ipJqzZIjt7+yfWts/qoIk//mL8/mew9ndlqr2pbi0+Mn63yZTJ+0v1Sxfq5KEDGrEkUr9v3ajxfbpp9vYDsrC
w00XzZ/XzsggdOfD7C7oTAAAAAAAALxePRmawHTt2qGHDhgoICJB7l06yfVMz+KHUaM
koFS5RWrvwF1Sl0lJLu3d0271c9vbiFhVzc3I0f5+xuj3SxzpTJpE9WJ2dj2/mTx1Xev468C/moXsu2SrhxXQk
3b0iSpocGq3Wfz+To6PiibgUAAAAAAMBLRRCWwSpXrqwNGzbo6NGjKlKkiPbv369t27Zp/PjxkqRTp04pNjZWtWrVMp7j5OSkihUrKioqSs2aNVNUVJScnZ2NIZgk1apVS5aWloqOjtZ777334jcGAAAAAAAAAAOq4CQAqDBasDGbSbanpNBYqUKvhZn2ezHTgWaG/K6vKWZ4s3tcpxvWeb7GY+TVJr+tdpeBqA2lVuTlcBdNeGg6BHhj+rU8Xh6bp7K8Ekt7vBfRQ9iCUS5TkEH7xk2T1NFRQG8xfTT+ezyiMoNbrWFRbwcvBmbpQADOERR
WgnXGvtE+WnZ0H4aTsGGpFaYIV71FVUYZ4I2IG8YzgiNXDfKhJ13OvyO8V8QtPWXlytWgexx9dJKNcPXZ6JJ6gF
KDZs2crV65c/8o9AwAAAAAAeBYEYRls2bJlWrRokSIinTJ3l2njhzSnBFDjJvmW9vYqN/EWZoyqLc+rFhMllZ
WKulX7eGKMYPhiXV9ypSXT5nyJt8/CaihdUsXqvkn/SRJVQPeNbbn8SmqPD7F1K22nw7t3qGSftVkbW0jjoNHm
tSdNCBIDVq316nD0dq9Ya0079+v0aNHq2fPnlqxYsWLvTkAAAAAAAvEEFYBuvbt6+ut6+Cg4zPVrFkzt6+Cg4PVrFkzt6+Cg4PVrFkzt6+Cg4PVrFkzt6+Cg4PVrFkzt6+Cg4PVrFkzt6+Cg4PVr
```

q/aI+0Hd23XueMx6vLFWM0fPUxlq/srS5YsatKkiSZNmpT+yQOvmBibbBk9hHTzTUff6OjolzaOl8XXNzOzBAA
AAICnY4+wDJaYmChLS9PfBktLKxlSH13tlcXBUU6u2XTx9EmdiN6vCv5103ydlJQUnYk5LBc3jyf2uR57Ubfib
srF3f2RtqT79zRz2EB1/jxMVlZWSk1NUcqDB5Kk5ORkpaSkpHksAAAAAAAAGYEVYRksMDBQw4cPl7e3t67Yuen
U4WhFzv1a/o2bGfvsWBspR5dsyp4zl87GHNbs4YNV4a16Kl21prHPxP495eruqVafDpQkLZs8ToVLlZVnnnxKT
IjX6llTde3iBdX6oIUk6e6dOlo2eaz86gTIObu7Ys+d1oIxX8jTO59J3T8tnzJeZav7K3+xEpKkImUraP6YYTp
w4IAmTZqkKlWqvLybBAAAAAAA8AIQhGWw8PBwhYSEqGvXroq9fEUu7h6q3bS1Pujay9jn5pXLmjsqVPHXr8nZz
VOlG36g97sEmdS5dvGCLCz+Wll2JyFeUwf3VdzVq8rq5KT8xUtq+OJv5VWwsCTJ0spSZ44e1ubVy5V4KOEubh4
qVaWGmn/STzaZbE1qn405oh1rIzV21XrjMb+6b+vQ7p2qVq2afHx8FBER8dj5paSkKDQ0VAsXLtTFS7HGvc/e7
xIkCwsLY7/zJ45pwZdf6I89u5SS8kC5CxRW34kz5JYz92Prbly5VJMH9jI5ZpPJVks0/PXoZ9y1q1rw5XDt375
Fd27Fq1j5Smo/6AvlzJvf2GfOyFBtXr1Mtpkzq9Wnnxlf UiA9DCDn9IlUZGTkY8cAAAAAAABeLxYGw1N2XH9FJ
SQkyMnJSfHx8XJOdMzo4bwwK49eyughpFsjnxxPbR8xYoTGjRunefPm6Zydm05E79ekgb3UIqi/Atp0kCTFnj2
t/h8E6K33m6lqwLuyz+qgc8ePqnCpcnLKlv2xdTeuXKo5IwZr4o+/GI9ZWFjIObubJMlgMGhgs3dkZWOttv2HK
HOWrIqcO12/b9ukCWu2yM7eXns2rtOOwXO1YOp8XTpzUlM++1Rfb94jR5dsunMrQf3fb6BdWzfJ29s7Tffiv/j
797/+63P8r8+PPcIAAAAA/FelNStiRRheqh07dqhhw4YKCAjQyqOX5J7bS798v1rHD+4z9okYPOpla/irTd8Q4
zFP77z/XNzCQi5uj+5nJkmXTp9UzP4ofRW5Sd6FfCRJnUJHqX3VUtr2/SrV+qClLpw8Klw8puJv+KlgiVIqWKKU5ow
coivnz8nRJsWjPlCdZu3SXMIBgAAAAAAXn1slo+XqnLLytqwYYNiIkSaePHNKR33arTHV/SVJqaqqiNm9Qz
rz5NbR9c7WrXELBTQL0688//mPte4l31Nm/gjrVLKdRXdvq7LGjxrbkpCRJUibbvx7TtUe2mTDoctUeSlMe
nuE5EH9Dt+DidiD6gpHv35OmdV4ejftXJPw6qQev2L+w+AAAAAACAjEcQhpcqODhYYzZo1U5EiRdTE11t93qujt
9t0N07FFX/9mu4l3tGqGZNUptqbGjxrsd6oVU9jenTQod07n1g3V74C6jZ8nIInz9EoynoycpNVVnzV/R9djLz5
sz19Q2XPm0sJxI3U7Pk7JSUlaNWOSrsde0s2rlyVJZarVVPXARur/QQNNGhCkHqMvdazvaaHDlDn0DD9D9tHief
Hx8VKVKFRO6d0jl3ywAAAAAAPBS8WgkXqply5Zp0aJFioiIOCV7d5O6ckhzRgwxbbppvSE2VJFXwr6vAtp0kSfm
K+uro73v105L5Kv6G32Pr+pQpL58y5U2+fxJQQ+uWL1TzT/rJ2sZG/Sb9qD1ei/W1bv
KY9+qhpjz5/jXfSWJWsXE1WNtZaMW2CYv44pDVr1qhNmzaKiop6GbcIAAAAAAD8SwjC8FL17dvXuCps5dFLyuN
TVNcuntfKfK6eF6870mcnBxlZW1tftFtln/KRaCQDkftTvN1lSaAwH1fy87yaZU+aj0/CMgtYZm2X6O/uGN+W7+V
lq+O9Sl/wnZOqkrMClxg2JPEpuCEVs/xWb3r5XRiSx3YpFlWaeJkbkOdBmeAzPPxH4Pt7cgq8Yy5NerPFyb2lfCywnaXR3a0oRtLOHMj4U0XTeKOkptPC9
tiyLlfy79xM2Ofhu27alzvj1WsfCX5Vqys33/ZpL2b1mvo/G+MfSb27ylXd0+1+nSgJGnZ5HEqXKqsPPPkU2JCvFbPmqprF
y+o1gctjOfsWBspR5dsyp4zl87GHNbs4YNV4a16Kl215iPj/Hl5hBxds6mCfx1JUpGyFbRs0ljt2rVLP/74o4o
VKyZnZ+eXc5MAAAAAAMC/giAML1V4eLhCh3DhCQkLLUtWtXuV6+Ihd3DTVu2lofd0Ol17FOxdn11Ch2lldMnafbwMl
199J85Q0XIVjX2uXbwgC4u/VpDG1/TEXcV/l65raTSXf1XqzH/S8Zbr8rrjB2eiWl3Mk1d5R7+nLcmbbrLsZn85iD
mjeYOjWdW7rNtNk9nS9OlTQtHjpVGUqUa6nZjr/dS1c9B/8GRkJrz9J72N8p8Q5d6iZnGy2m7wsJlG2uW/8zFk2Y14AyBm
OWJIk4mbG9pAqPtDeXcPzcdKpuaAWCt4Nfh2DRnbDbJhK9I8KXs1v/+XNxlInJGhQCYyC5qaBxz4LNqFoHMaklU4f
yvRCXQhdN7YTA1GjSf0v7s1Sa7FjRkt/GqZZCOmjJ0Dh0iFRrwcwILGinybNuIKgW6GtLvB0/SjXPKPGcWXMzXEDQBCl
0lFTsHo+M0OObSKy6KjzvmMIP7DcfpyCv4cGaGI8/1DomsqTaX6rdNXvyHuMFHxOu/d+PmijP4AgqtMPl/2Jh/Rl+6k
9zIxB7CJxCjsoxNeRrMomz1BHJD6J6NmNXoIHh3TCtdnyT/5QAw7xWrx2sHfttiDAPJMPoj1r+OPE/tZrW7O9ZUs9uCl
PpZ3LxHJNr99PTs77zrCmZtnjQNTrqWkwnUs7Vsk7WGaRDC1Kv6jJdfeijAybVSNzoIrPaw8VeN5DkHfE8Hr1JEi
E5/jPkzIiZvOzLrOp2BKvenyz6jq9sAu/mI6mmUzE+xxKnQAAAAAB6yHDrgsvhYiG9iE+OAZYAAAAAAkdB9VC/gqvKy2d
C4OwVy+7Y+A2YvxAAAAAAAEDGn5Sk/FP0NkKdIz/qxqdsO+0nOMy3mbmcTIi4CXhm+sLa2U/E+g1gkZLwgKzAE7
dvp1zFbk6vWYfSExFcW2BfmMOvHtF2Hp5kSGfcQAAAA8tDWPj7vQPAaxjbJv0hmTZA/TDeVqI7VqYKCvIb1y48ZDoVD
/XddD8DaVKaPh6QCqbnEPIbvHIAAAAAAAAdt8zQW+mQ6ZaTz6ERz1RaqS1dqUKlcQr80fTQFvrb7NKuXKcY7pBzUAAAAAA
VWPl9cx3/nc9vxE2SVAy9/aQKAIJfHyDJeCBwovMaVGPcgOIkNYq+JJIkWLP94TqWK4IzyHCOqPBXnKt8QAdlDkYGKCAIJPJ
oS4/kAAJdYyDLxHdQeSO3Mh4k1Y2ZVgwRGmBo/v/CyQiYO9tu8t57RFcFtYP0VkCZjPAgTPJ6ivh4QBj3k4QH8a
ieZShMo9jBnkb7SedjSjYSaZrZTdHa12V3WYpQbyJAgHvpqMt3hTcVTZpdR1uoXr2T1ZJWlgYU0JMFiU0m5y
aBXpdm/a7vPVzU5aVFMaz+M7pJnmM8uqOtN9+9wRJRb6pKqKsn8UAlgE3dG9yg9wdhUHFfTjGCIEPTNW
OAEJgV8S5vT4LhO8+aLMFhXzJ7bcz7vTzpz3z6VgnpLsPU5TUATBQ1xOk0HeeqNqUXoJQYoSo8VeeJcqIqhw/IfL
mC4ZYFNnMoVqH+Vh6SWB+PvpEN0yuQJFmPNuNO3dXpeXmoK0s0k1mAf1YDBJ6Se+0rwJZI1mYjj0IkXhvTUfK
ysKbS+1w9Qvo63BZc8g2xfgMpvY1ZEPSvNLt2+XfRIt3qGJmLD2UfNI4VL4oGRfjXGHLRNRHBdjmBU9Ifz8tePBc
tIPd+6YkJ1+BxnLk9hPXW7EVzBJaTJUbKTzn7tHEZjSu5qLq8X+2Hn3hxrYdGe1YXLg56zCbb1fCYDXqlkHJvLnb
lXt5gbBnrCB8HZ3lU4FGU4hHR1h1eSnsW7m7Jn0jKUz7d/NnB9h+7/pwH3w1nqk7bxClzOZIFFq+SKgLG8qP8RfLs

```
0EnhOP/30kwwGgy7buCj2zCnNHzNMufIXlH+jprKwsNDbbTrom2kTlCNvPrnn8tbiiaPl4u6hN2rVM9YIbdtEb
9SqpwatPpIkLRw7QmWq+8stRy7dvXNbv6xZpU07dyhkZoTxnJtXryju2hXFnj0lSToTc0SZs2RR9hy550DsYjL
G5VPGq2x1f+UvVkKSVKRsBc0fM0wHDhzQpEmTVKVKlZd9mwAAAAAAyHDpXhG2detWBQYGKmf0nLKwsNDq1atN2
g0GgwYPHqwc0XIoc+bMqlWrlo4dO2bS58aNG2rZsqUcHR3l7Oys9u3b6/bt2881ESCjxMfHq1u3bupZv7omBn+
iImXfUMjMCFnb2EiS3u3QTfVbtdO0wf3U/4MGupd4RyEzFimT7V9vmow9e1q3bt74q+aNawrv31M96ldTaLsm0
h69TyEzI1SqSg1jn3VL5qvPe3U0NaSvJCmk1Xvq814d7dm4zmR8Z200aMfaSDXt0dd4zK/u2ypXo5aqVaumAwc
0aMKECS/l3gAAAAAA8CqxMBgMhvSc800PP2r79u0qV66cGjVqpFWrVundd981toeFhWnkyJGaN2+e8uXLp5CQE
B08eFB//PGH70Owe/od//fr1denSJX399ddKTk5Wu3btVERDzhqqYSEhLk50Sk+Ph40To6pmf4r7SVRy9
l9BDSrZFPjjT3ZX6vnvTMT/rvz5H5vXrS+zMKAAAAwDylNStK960R9evXV/369R/bZjAYNH78eA0aNEgNGzaUJ
M2fP18eHh5avXq1mjVrpsOHD2vt2rXas2as2ePypcvL0kKDw9XgwYN90WXXypnzpyP1L1//77u379vMjkAAAAAAAA
gPV7oZvmnTp1SbGysatWZTzm50SkihUraufOnZKkGnTt3ytnZ2RiCSVKtWrVkaWmpX3/99bF1R48eLSoKU2JHHy5/
8vrRQ4bAAAAAAAAAZuCFBmGxsbGSJA8PD5PjHh4exrbY2Fi5u5u7ubtFtbW8vV1dXY538NGDBA8fHxxs+5c+de5LA
BAAAAAABgBl6Lt0ba2trK1tY2o4cBAAAAAACA19gLXRHm6ekpSbp8+bLJ8cuXLxvbPDU9deXKFZP2Bw8e6e6MaNG
8Y+AAAAAAAAwIv2QoOwfPnyydPTUxs2bDAeS0hI0K+//qpKlSpJkipVqqS4uDjt3btX2gk0qK3o/MqWbfAtuP+ty3BLnJyl5kI9AHkj4hIkPBIrZydNv1EQGkXaFV1/QkoE4J6ZgdwiOR1fp4Pj7Hc56hSKKnhfFeKEdwbBzHnkA9Vdz26Jhdj5irNgBg4Pv
AAAAAAACAUbofjbx9+7a0Hz9u/H7q1Cnt27dPrq6u8vb2VlBQkL744gsVKlRI+fLLlU0hIiHLmzKl3331Xk1S0a
FHVq1dPHTt21LRp05ScnKzu3burWbNmj31jJAAAAAAAPAipDsI27t3r958803Q82d01f9+vX
TnTt31KlTJ8XFxa1q1apau3at70zsj0csWrRI3bt3l1tvvvSVLS0s1btxYEyd0fAHTAQAAAAAAB4v3UFYzZo1Z
TAYnthuYWGhoU0HaujQoU/s4+rqqoiIPReGgAAAAAAHhmL3SPMAAAAAAA0BVRRAGAAAAAAAAs0AQBgAAAAA
AALNAEAYAAAAAACzQBAGAAAAAAAs0AQBgAAAAAALNAEAYAAAAAACzQBAGAAAAAAAs0AQBgAAAAAALNAE
AYAAAAAACzQBAGAAAAAAAs2Cd0QMAAOBFiI60zughpJuvr29GDwEAAAAwK6wIAwAAAAAAgFkgCAMAAAAAAIB
ZIAgDAAAAAACAWSAIAwAAAAAAgFkgCAMAAAAAAIBZIAgD8FQpKSkKCQlRvnz51LxWft7afU76vSwAw6Xf+x
DGN7PKhWpf3UYsyBdTv/fq6evH8E+sObt1YjYvkfOQzvHNrY5+7d+5oxtCB6lqjXyqq9PAmropyXzTerMGRm
qDysWU6ea5bQ1cqVJ2461kQoMDHwBdwEAAAAA8F9gndEDAPBqCwsLCwsL09SpUzVv3jPTiej9mjzuUiigT
QdJUuzZ0/qsbt66/1matqjj+yz0ujc8aPKZGv3xfqQXKy8futuJv69NlatqjV2v3YW4eFl+2x5Gi3GUCn7pCvt1eyYamJO0apuMQ2d01f9+vX
3XF7at32LZgwdIFd3D1Xwr6s9G9doakpn32q01VryNElm+7cSlDEV2HatXXTy7s5AAAAAIDXC
ivCADzVjh07VX5vd9wLBhQwUEBMg27r2IuzUvKvjXlVO27E+s6+DsIhc
3d+PnwI6tsrXLrMr1/lrBdXTfXtV89wP5Vqws99xeqt00lfL6FN0xAw+vfeHkMRV/w08FS5RStbffU+asWXXl/
DlJ0oIxX6hu8zby9vZ+KfcFAAAAAPD6IQgD8FSVK1fWhg0bdHFBbFBMTIOk6feSQjvy2W2Wq+0uSUlNTFbV5g3Lmza+
h7ZurXeUSCm4SoF9//jFd19nwzWJVadBQvb2xmxM+pctrz8Z1un75kgwGgw7t2acbpeGlZpNiqeN0NPKUbk8
f3rCPS+rbyVwxwXqqNBYQeCzcXZf7fXuQKiwrV2Eaqc4aJqBzSSfKNW3Hcgd19z16Q3Q8bcsvKo1ldxZ8gQu3vWmVEZ5VnKvx9zqFhnSUQdjqWqUzbG2ayHvg9F48ip4tEd8ysl0g
SFH/9mu4l3tGqGZPU/JP+at3nM/3+yyaN6dFBn8/7RsXf8PvHaxw78Lv0HjuirsPHmhzvEPKFkvE5Kqzf0j8Nq6qUrHAfjEZo00CjxEMNcQCmXNsfK00bXEVuaYwL6eBDnWrIqzGNRUC0OwHOTJ6EgWEx4JpFoJCRkPyNVRwmMJL
S0LC0t1GTZGxStUkiSVqVZT1QMbqf8HDZTJ1k49vwyXbSZ7zTv0kzmVBhfsG9grNBL94TqwmLrVr95akZKmbzTn2FXJpNXEw0CacNoRGdjSZzQKnRWVZN7FYLlSCjFhUOQhDZYaHurHD+H3wSvE4G8lOmJIK1ck6eMpAV8AsVR
CAAAAALw2CMIAPNWyZcu0aNEiRURE6JK9u04d0aQ5I4bIxd1Db77yUrzP9JaDVKZMKU6ZTD7s6nmjqMJKb3SxXn7Kg
Z+mIGzDN4vlXbioCpUsY3L8hwWzFbM/SsFT5sFT5sotV279sWeXZgXzJWqQXF2S1LRHHzXtAeev8U4aq5KVq8na2kZR8n
Kxlorpk1QzB+HtGbNGrVp00ZRUEv6rYAAAAAF5DPBoJ4Kn69u2r40BgNWvWTH69u2r4OBgNWvWTHl8iqpmw/cPjC5Plyuu
LjKytpaXgULm5yXu0AhXbt04R/r30tM1PYfvft2M72tMf
W0n/ymLZErlSznv106NcdKlq+ktzc3NSkSRP99ttvunxXr1jPeBQAAAADAfwErwgA8VWJInk
iSbTJlU00LeULpw6YdLn4umTcsuZ+x/r71gbqeSkJNX4/0ct/yHKnCW
LUlNTlfLg4Vspk///7ZQpKSn/0B68vmJssmX0ENLNN6MHAAAAAJgZgjAATxUYGKjhw4fL29tbgAJ//L29tbV+zcd0pwtCLnf
i3/xs2MfRq276pxvT9WsfKV5Fuxsn7/ZZP2blqvof0/MfaZ2L+nXN091erTgSb1N65YrDdq1ZWDi6vJcfusDip
ewU/zxwxTJls7ueXKrUO7d2rLt9/ow+Ahj4zz5+URcnTNpgr+dSRJRcpW0OLJYY7Vr1y79+OOPKlasmJydnV88/QQ5cyXX30nz lDRchW
QEAAAAAvG4IwgA8VXh4uEJCQtS1a1dddnZO/3Ci0xH1jLQH7fKtRPD4CB9r/UE7H9l1LMCxRVs+wenKyu9M+i6Hwrmb7o2ss4UB5nEGIoAV4GQ9fYm58+lYiMVfD31F3kOP6YmsfEl3njdtTp5nwdb2kQS5J8Wxvi
Nfa5dvCALC9PVXRd0HtfhqN0aPGvxY6/da9xULRo3QhP6dtft+Dhlz5lLzPf6q26zAPv02m3KuYPAKTqjLi
4VKllFgu84KCAiQu7u75s2b9yJuBwwAAAADgNWZhMBgMGT2I9EpISWJPTk5piYvP5g0kGi3TX2uPLHzcX4SRyGP4i8FW6c6qR0TJTUfHKJfkWLu5qdLHT2LQdd
50tyX+b160jM/6b8/R+b36mF+AAAAAANIirVkRm+UDAAAAAAADALBCEAQAAAAAAAwCwQhAEAAAAAAAAs0AQBgAAAAAALBCEAQAAAAAAAwCwQhAEAAAAAAAs0AQBAAAAAAAMAsEIQBAAAAAAALBCEAQAAAAAAAwCwQhAEAAAAAAAs0AQBMGspKSkKCQl
Rvnz51LxUfnWt7af1U76SwWAw9om7dlXhwUHqUK2Mmpf0r2EdWuji6ZNPrTu4dWM1LpLzkc/wzq2Nf0at+0FDP
2qmDysWV+Mi0XXqcPQjdeaMDNWHFYupU81y2hq50qRtx9pIjBQYGPucdAAAAADzYZ3RAwCAjBQWFqapU6dq3rx
50mfnphPR+zVpYC/ZZ3VQQJsOMhgMCuv6J0kaxsrBU8ZY4yZ8mqyLnT9flHTTVhzRbZ2ds/tm7f8Jl6kJxs/H4r7
```

qY+fbeW/Oq+bTx2726iipR7Q5XrB2pqSN9HauzZuE7bvl+lkJmLdenMSU357FOVrlpDji7ZdOdWgiK+CtOurZt
e/EOBAAAAgP8ogjAAZm3Hjh1q2LChAgICtPLoJbnn9tIv36/W8YP7JEmXTp9UzP4ofRW5Sd6FfCRJnUJHqX3VU
tr2/SrV+qDlY+s6OLuYfN/+w7eytcusyvX+WsFVs+H7kqQr5889tsaFk8dU/AO/FSxRSgVLlNKckUNO5fw5Obp
kO4IxX6hu8zby9vZ+3lsAAAAAAGaDIAyAWatcubKmT5+umJgYSQ46feSQjvy2W22DQyVJyUlJkqRMtrbGcywtL
WWTKZMOR+15YhD2vzZ8s1hVGjR84gqyx8njU1zrly3S7fg4XT53Vkn37snTO68OR/2qk38cVMchI9NcC6+/6Oh
HH5191fn6+mbOEAAAAAATBGEAzFpwcLASEhJUpEgRWVpZKTUlRS2CglU9sJEkKVf+gsqeM5cWjhupjz8Pk21me
62ZN13XYy/p5tXLabrGsQO/6+yxI+o6fGy6xlamWk1VD2yk/h8OUCZbO/UYNUG2meO1PXSAuo8cr58Wz9OgZfO
VPXt2TZ8+XcWLFO/3/AEAAADAnBCEATBry5YtO6JFixQREaFL9u46deSQ5owYIhd3D735XhNZ29io38RZmjKot
z6sWEyWVlYq6VdNZar7S3/bUP9pNnyzWN6Fi6pQyTLpHl/THn3UtEefv8Y7aaxKVq4mKxtrrZg2QTF/HNKaNWv
UpkObRUVFpbs+AAAAAJgT3hoJwKz17dtXwcHBatasmfL4FFXNhu8rsG1HrZwebuxTwLekxq7+WfP3HNHMX/YpZ
GaEbsfdlIfXP+/PdS8xUdt/+FZvvd/8ucd6/uQxbYlcqWY9++nQrztUtHwlubm5qUmTJvrtt99069atNNW5e/u
2Zo8YrM7+FdS8VH4NbBZo3BNNStvbLB9nzbwZ6lGvqpqXyq9ONctpzsghSrp/z9i+NXKlOtUspzZvFNWckaEm5
145fO7d61ZV4u2OzQEAAAAAngVBGACzlpiYKEtLOz8KLS2tZEh9dLVXFgdHOblm08XTJ3Uier8q8q+Nf9x/o71kY
qOSlJNf7/UctnZTAY9PXg/mrbf4gyZ8mi1NRUpTx4+FbK5P9/O2VKSkqaak0J+VT7d2xVz7BwjftugOpVqaHP2
zXV9cuXJP31NsvWfQameXy/RK7UwrEj1KRbb034fou6fjFW23/4TovGjZIkJdy8rqmD+ujDfoM1eNZibY1cob2
b1hvPnz5OgFp9OlD2WR3SfE0AAAAASC8ejQRg1gIDAzV8+HB5e3vrip2bTh20Vu+XfuJmxz461kXJO0yabibunbnbpgm5cebjX2MVTJyRJztn
d5eLmbtL35+URcnTNpgr+dSRJRcpWOLJJY7Vr1y79+OOPKlasmJydf9vdvfv3dWudT8oePIcFa9QSdLDxy/3b
lqvnxbPV4ug/v/4NsvHOfL7XhUpWOHV/j/wc8/tpaoB7+rYgd8kSZfPnTx5T+Tdr65c
1q2Rtba1KdRqk+XoAAAAA8CwIwgCYtfDwcIWEhKhr166Kvv7XxFLu4eqt20oT7o2svY5+aVy5o7Kl1/J2c1dN
Rt+oPe7BJnUuXbxgiwsTFeWXTh5XIejdmvwrMWPvfaejes0OeeBf1xnXu4skqqUm33ib7gsVdu6oV0yZoxO0Lvc
KlSyjwHadFRAQIHd3d82bNy9N8019kKLU1BTZ/O0tmJKUycH5OR6J2p6nG4xQpU15bI1fq2IHfVahkGcWe06Pft
m5QjXcehmo58uTT/bt3dfKPg3LLmVvHD+6Xf6PmuhoftypzUl/x48d8/aPH6/x48d8
r5dFLj+OTOKaDAtp0eGqdoQtWPHIsV/6CWnHk4hPP8W/UVP6Nmv7jGJ2zu2naxkdemvJxDH/eP7fZc6aV
T6ly+mbKeOV038bvVytmX5Q8vfOmq9bfVQtspISbNzSo5bsyGAxKefDwrZTJ/4/4mS7t9TzYbvq0y1mpr8WW/Vb9l0l8+f06iubfXgwQM17qfap/cxjAQAAAIAnIQgDADPTCTAABWrUEl02bI/9IwAzHh659wBAAzHh659WwhSt1LzFRd2/fkou7h8b26iwPrzzPX
HPJxNGq/k5j1fqgpSQpj09R3R3bubqGmD+6rxx5888gkOkhLZ08rJeWBir/hJ0nKkTe/Yvb/ZtwPDQAAAABeFIIwADBTdvb2bM1JjE22jB5Cuvlm9AAAAAACA/0EQBgBm5vdfNkaeVtlvG+rvZvvr7DOs+c3UyeoTNU3lb94SZ3846BsI
7ozHwAAAACYH4IwADAzibcTG9hmcdDqcSF2PvaSszs6qVLuBWvQKlrWNjaS0vc3yf9+S+X6XIFlYWGjxhNG6cTl
Wjq6uKv9mbbUICja5tsFg0LTBTBfdU2eIjs7OdZQD5KS1CFHkC2XzyPFS7wEAAAAA8OQQBgBmp
kr9d1Sl/jtPbE/L2yz/9y2ZVtbWatL9UzXp/ulTz7OwsNDwiG8fOV7+zdoq/2btp54LAAAAAM/L8p+7AAAAAAAAAAK8/gjAAAAAAAACYBYIwAAAAAAAAmIUXHoSFhobKwsLC5FGiRAnjemJiorp16yZ3d3c5OTmpWbNmunLliskesbGxatiwobJkyaKcOXPq008/VXJy8oscEwAAAAAAAGbohd4RVrp0aW3YsMH42to4/ttvv9Xq1au1cuVKubq6qnv37mratKm2b98uSXrw4IEaNmwoT09P7dixQ5cuXVLbtm1lY2OjUaNG6ZUrUPKcEYQAAAAAAAMgXXqhnwiztraWp6en8ZM+eXZIUFxenuXPnasKECapVq5YqVqyoiIgI7dixQzt37pQkrVu3TocOHdLixYtVrlw51a9fXyNHjtS0adN0//79FzkmAAAAAAAAzNALDcKOHTum3Lhyq3DhwmrdurViY2MlSTExMUpKSpK/v7+xb4kSJZQvXz5FRUVJkqKiouTj46NcuXIZ+9StW1dxcXE6ePDgE4957949xcXFmTwAAAAAAACAv3lhQViVKlUUERGhNWvWaMaMGTp9+rSqV6+u+Ph4Xb58Wba2tnJzczPZJlewXLl8+bIk6fLlyyZPhD2j90ouPVWVIpUWpHC7x8dff/nFwtOdVVLVI+vDTzvXo0+Z+0xvLtEmjaflHfEuMd46AOm+qOnvfUPcVPqRPTWsBpWp1K5zVvEf3FqtEQm/ki9dmrHl6Y/k4e2jrz9YHPt5uqbK5DfvaN6gEZQbi//5+58pHbq36+yvu/P69rT9pPOdAAAAAAA=

```
p3Q45uWbT+qULdfLQAY1YEqnft27U+D7dNHv7AVlYWOjy+bP6eVmEjhz4/aXdGwAAAACPIggDALy2duzYoYYNG
yogIEArj16Se24v/fL9ah0/uE+SdOn0ScXsj9JXkZvkXchHktQpdJTaVy2lbd+vUq0PWj65uIWFXNzcn9hcLbC
Ryfe2waHa8M1inTn6h0r6VdP5k8dV3r+OvAv5yMPLW/PHDFPCzRtycs2m6aHBat3nM96CbGaio6Mzegjp5uvLL
mgAAOC/5aW/NRIAgJelcuXK2rBhg2JiYiRJp48c0pHfdqtMdX9JUnJSkiQpk62t8RxLSOvZZMqkw1F7nlr7XuI
ddfavoE41y2lU17Y6e+zoE/smJyVp/dKFsndwVN4ixSRJeX2K6UjUbt2/d1f7tm2Wi5uHHF1ctTVypWxsbVWxd
v3nmjsAAACA9GNFGADgtRUcHKyEhAQVKVJE1lZWSk1JUYugYFX//9VaufIXVPacubRw3Eh9/HmPba8286bo
eeOk3r15+Yt1c+Qqo2/BxyuNTVIm3bunb2VP1WfN3NH7NJmXzzGnst3fTen31aRfdv3tXLm4eGjJ7iRxdHr750
L9xM52J+UNBATXl40KqT8dP0+34OC2ZOEZD53+jiPFh6rd+jQoUKKDZs2crV65cL/dmAQAAACAIAwC8vpYtW6Z
FixYpIiJCl+zdderIIc0ZMcS4ab61jY36TZylerhKYN6680KxWRpZaWSftUerhgzGJ5Y16dMefmUKW/y/ZOAGlq3d
KGaf9LPeNy3YhV9uWq9bt28ofXLF2lsUGeNWva9nLJll7WNjToHml5SdI94a10rx/v0aPHq2
ePXtqxYoVL/4GAQAAADDBo5EAgNdW3759FRwcrGbNmmimPT1HVbPi+Att21Mrpf72JsYBvSY1d/bPm7zmimb/sU
8jMCN2OuykPL+8OX8faxkb5ivrq0plTJsft702VI08+FS5dTt2Gj5OltbU207tquc8djVL91O0X/ulN
lq/srS5YsatKkiTZv3vxM8wcAAACQPgRhAIDXVmJioiwtTf8qs7S0VGpqqiVH4dVeWRwc5eSaTRdn9SJ6P2q4F83z
ddJSUnRmZjDcnHzeGo/Q2qqkpPuP3I86f49zRw2UJ0/D5OVlZVSU10OU8uCBJCk50VkpKSlpHgsAAACAZ8ejkQC
A11ZgYKCGDx8ub29vZF06nD0Yqc+7X8Gzcz9tmxNlKOLtmUPWcunY05rNnDB6vCW/VUumpNY5+J/XvK1d1Tr
T4dKElaNnmcCpcqK888+ZSYEK/Vs6bq2sULqvVBC0nSvcRErZg2QRX868jZU03bt7Q2og5unE5Vn71Ah8Z5/I
p41W2ur/yFyshSSpStoLmjxmmAwc0aNKkSapSpcpLvEsAAAAA/kQQBgB4bYWHhyskJERdu3ZV7O0OUrcnH3U02mr
fVB117GPjevXNbcUaGKv35Nzm7uqtnwA73fJcikzrWLF2Rh8dfKsjsJ8Zo6uK/irl5VVicn5S9eUsMXfyyuvgoU
lSZZWlrpw6rg291yuhJs35ODsooIlSumLRavkXcjHpPbZmCPasTZSY1etNx7zqOu2u+lDEzO7ql/2vx5ZzuMe7mCPxvOzRw8QMDOE1N3hDuEcJkQw5I
e7gVRNjky2jh5Buvhk9AAAAgBfMwmB4ym7Br6iEhAQ5OTkpPj5eo6O0GT2cF2bl0UsZPYR+STI819md+rJz3
zk/77c2R+rx7m95f/+vyk//4c/+vzAwAAyEhpzYrYIwwAAAAAAABmgSAMAAAAAAAAAZoE9wgAAeIWlpKQoNDRUO
+fMU9y1q3Jx99Cb7zCb7zXR+12CZGFhIUmKu3ZZ1k9Ih/fHaf4sznvF8d2DMj+mSlf3FpX5/7ym4L4XpP1VeVnKXZ6K2r8ki2t
/8RJq2WuACpUsY+yzNPxLbfvhW12PvShrm0zKX7yEWgQFq3CpspKk5KT7mjjKoj/Zs+EnO2d3VccgIlapc3Xj+m
DFjdPbsWYWHh7+MWwMAAACkG0EYAACvsLCwME2cOFEzZ2w9X5MG9pJ9VgcFfTP1skPnDtdn3/UVBPWbJGdvf1j6x7avUNVA96VT5nysrGl1eoZkzzWOfXONX7NJ2Twe7guVM29+dQoaLpes3dx3o4UgbvmqQ/ZLg9xzyAQ2ZHTHzvKEtb1UqRUkssElixYTm8tB5Nyn0jTJN6eqgj0xbSXBE9XP8ZXdUHbPyx+0v2zglJqv
4NBIAgFfYjh07lLBhQ5WrWrWUUvuub3kV+9tlapSQ8cP7pMkXTp9UjH7o9RpyCgVLFFaufIXVKfQUUq6d0/bvl/1x
LpBXO5WvRZtla+or3LnL6QuX4yVITVVB3duM/apFthIpSpXl6dXHnkX8lHb4FAl3r6lM0f/kCSdP3lc5f3ryM0f+3ryLu
Qj+q1bKuEG9eVcP0GJG16aLDCwsL+Uy+1AQAwOuPIAwFdY5OcqbVItbv4IRR35bfbIcVaRbIIR35bbfKVPeXJCUnJ
UmSMtnaGs+xtLSUm7VY4lItiGfj3CEbUtw7Ivkjr/NkqgPuSa7bUUWtsiIVGHPNgEqAI0VBIKoHu20yptYS1vi0gcBNKt/JVqK3vumGyzmpOJ0hZSBNVPn8SvBfLnjGC9uy6q8B0hXLKX+v/VA8gMvYN0kdz6jJ1MDpsOlscz4MPGcEtAt4/1UDGhF5Dl/J1AFWtrEgDbfT2hNdG7KxNjAr6vuHR+0ENGUYvhjVVs4fVLn+1kJBwU3cLdfKzsMMfdZlHgdljr2QpWUmOEOZb/rMhTJ9hIIsQvFVtrpVmqhs3yqjp89ktBMQLOWjoQ6ZwZf7cmc8QcumR1jHdt3gEeu/ZWzmCGm8hwzmyjmJQmAfVrGKqNZSLJHO1HL6/sfyuRoY0qdGB6H0aL3bBH5LwvP1uDfl6k8IEF3tHZzvdH7zl+UP9cOUugaXfz/xU+9ExCfhRYAAPJnBgOkg6fdBWiKqwLSCm2lOoDDlHNhgLeQHUDSqmOD9IJX/BE6NLb8NDFyVWFMCXhAE7BoLCkBmSjuCU6kZt8tX6z7yU0tR/7n7DzQHK3mGQ/j+jRDImLC4l6jHDgKqLnvXXUttRqxi7
YoXdl8AAACAZ8GjkQAAvML69u2r40BgVQ14V3l8iqpmw/cV2Lajvk/602MBXxLauzqnzV/zxHN/GWfQmZG6Hb
cTXl4ef9j/W9nTdWqGZMVMnOx8voUe6Tdzt5eOfLkU+HS5dRt+DhZWltrwzeLH1vr4K7tOnc8RvVbtlPOrzvVo
EEDZcmSRU2aNNHmzZuf+R4AAAAALwpBGAAAr7DExERrSHqa/+SAUD6VN7dMrJNZsunj6pE9H/7lN
rr545Wd9MHa+QGYtUsESpNI3HkJqq5KT7jxxPun9PM4cNVOfPw2RlRw5UUlBQlJydLkmCAXpkWSkpKm+gAAAMDLR
BAGAMArLDAwUMOHD1fUfw5p915fw5/br+Rx0/VoVa9cz9tmxNlLRv+5Q7Lkz2r1hrYZ+1EwV3qqn0lVrGvtM7N9
TC8eOMH5fNWOSFk8Yo67Dx8ktl5duXr2im1ev6evXKrF2im1ev6OGdO5Kke4mJWjb2AhQ2O3LsfKrF/jIO
JdPGa+y1f2Vv1gJSVKRshW0cuVKHThwQJMmTVKVpMmTVKVK1XSPOe7t29r9r9ojB6uxfQc1L5dfAZoHGt2T+r6+H9FfjIjm
1Zt6Mf6z746I5+tj/DTUrmU/BTQJ07MDvJu1zRobqw4rF1KlmOW2NXGnStmNtpEZ83CbNcwAAAMCriT3CAAB4h
YWHhyskJETTThw5QwvXrcnH3UO2mrfVB117GPjevXNbcUaGKv35Nzm7uqtnwA73fJcikzrWLF2Rh8df///pp8Xw
9SE7Sl590NOnXpFtvNe3RR5ZWlrpw6rg291yuhJs350DsooIlSumLRavkXcjH5JyzMUe0Y22kxq5abzzmV/dt3
Y3Zr2rVqsnHx0cRERFpnvOUkE919thR9QwLl6u7h7Z+u0MCc2ukW28gIjqG2U+FLkv7h9z7qXpDsm4NRmDoEKQXoc2bBq7d9/dsjdEvmMj1oqpTvmqSh1E7Sl5Q/3Ui8+/WEdA8xtkgUmeAzIMR6NE5bRmH/nSwuPA1RtHZ3n4Ynd/yU0K+MA2zgJLGJgF9Aw0zxmRpNn9cQeNXM6FGn5VXfgFd+L1cz5j1G6qVz/ZV8Te1WV0HjDSbfwLxgg8Z5F9E3nl3WcnDb/mfjWKQDl4BqE3x2Ce3cGqtz+bEwrFHLfgWfqFddeEnU1NNZe91eoMNI0d5HUPdKkyNmIJ6P/BF3ARaqV7qOnBSkGeEdJoyHsXmbYWtDVZbS1fZyVJkwEGDEN93Gvp9VuzUX0ImfZ8ebDF0OUgekP4WyxaW1pwTfGM7TyTf/4FllW9Zr5cBUgW+JNmXX/hHPP5Uv8kkPjgkTzMvVxsTESd9nNlFiMdEMPVYk9tHuovjyKzjlDJ+OGRAt4jsUdkjLXm/0Y9xP1mKeWaeGzJv/Oh3d2AIoSOo2kzsl6tQLFOLk0QyJCVMRbu6zhhdVLmjkO7wd4cVwQ9cRR7PTUnkPcTaT0fcD9pyvfU4yo5VeU8k+JIbzTMM4L12c2fY4DnzYn4EL4pMpFEmH6b3pK4fiTRpRPZNfT0GBaO22uvLbaWFS6gsu8rRK2eTALhhdc35MHOAJ2LK9u9k3dR15bE9dgjcCNsMkHGChaB9wvaAyTRQLi9WvUdsUc8q3mDM4hxSyT7hvZgVVN1Cq3Apj58c7gNUx8zWSlNyT6d5egVV5qIKmNq8UW7ATOGzj2DUEmzUOA45w2/0nKRwRB17h0cB6Dyf9m5oT1iM+yRmB7+BwlThNTS5P0NyVD+4jqxDNDD8XoKpYZdU96jK/7pq/V2MdCjVhtyqthzWnO1h9ZYhRy+zOA0y1IB1Ocq2fKlr2i2BY9ZTiH4gBNpJ2vTCoOR9iHDzvOYAlBeSeBH0eF7J3pYfX4O57NB3vhNYVn0WqBnjBHD8rNlDDhvrXSnSmqlR+SCGP/lOGa0EqKb6eATXB46tl7aMFMCgEH7mRjuIUzBG0pLUm3DSLPJEp6YXy+Yl7xNkIjFCLnRPf3QEn7USUd8wnUwACNVMJqY0vKm4vOUN0WRjBBJLUl0CbeqMIntfZQ4U8TAWv6QM/MPqY6Lf1r5fSw0mUXDp/Ed6VvqY6+tnFdJwzE3QdJ/17fVHoiCeyoaFQ6lwO7VUTUHTFcDFZ/OxHLz7+57IgvTfyqmEOvDBPXB4QLKl2uGNB/eTkwSTFrFN5YZkFUnJLzNXTV1kaXMg6UbpGtO6cG0p0LJMJjGuBuSy2sqALfj+Lkv+jPx98oG+tbYhoLZnd6E/v7eMaMmVOu7KfSDqnmPWo7X6ZoWpOxBg3JRNWRGbNtJnCzeDo8hAjoj8sPzy0b/Lr1e9FdSk+X4ZDEfNpI7I/bhYFglJbxJxPwxQL8KEYbrALaqMTyzv6LR5s7+TfQwAfxZzSTsezb3M+wxxnBAX8lHw7yNKFFkLq0lf+zZCE3Vfdyh11TKRsLEEo/UB5H9Csp8qQnZ/VM8w9ZnYxyzoFhbdCAcIoR9ZSMebsL2bnZT7ZYCbMtB3p6bFvMDWSgWXnUxm2z6VA2V2ypXyF2T6klE6HY0L6mqthCDbXVeCTNwX0sFhMMH6/HnGThWc/WQ5Pl1P4rVW6TZ2fl+0ld2/mW9vg+e0DU2pXAylB4wkX0hmcEkHBSuvRq7S
```

SPnTletD1rIv3EzSVLnz8PO25YN2rBisRp16qELJ4+p+Bt+KliilAqWKKU5I4foyvlzcnTJpgVjvlDd5m3kljN
3uuYCAACAVw+PRgIAgFdG6oMUpaakyMbW1uR4Jjs7HYl6GN6lpqZqYr+eati+yyOrOx4nOSlJJw4dUMnK1YzHL
COtVdKvmmL2RUmS8vgU14noA7odH6cTOQeUdO+ePL3z6nDUrzr5xOE1aN3+Bc4SAAAAGYUVYQAA4JWROWtW+ZQ
up2+mjFfu/IXklN1N275frZh9UfLOzitJWj1jsqysrBSQxnDq1sObSk1JkXM2N5PjTtmz68Kp45KkMtVqqnpgI
/X/oIEy2dqpx6gJss1sr+mhA9R95Hj9tHieflw4Ww4urvp46JgOBXAAAAB49bAiDAAAvFJ6jg6XwWBQxxpl1ax
kXv2wYJaqBrwrCOtLnYg+oO8XzFT3keNlYWHxQq/btEcfTV63Q19FblTF2vW1anq4SlauJisba62YNkFfRHyrW
u+3UHj/nmmumZKSopCQEOXLlO/NS+VX19p+Wj7lKxkMf7318+6dO5oxdKA61iin5qXy65OAGvppyfx/rHOnIV4
zhg5Q+2ql1bR”EXnWvW1VRWzY8tu/K6eFqXCSnZo8YbHL8n14QEBj46IsRAAAAXmesCAMAAK8UT++8GrZwpe4lJ
uru7VtycffQ2F6d5eGVR4ejflX89Wvq7F/B2D81JUXzwj7XmnkzHrv3mYOLqyytrBR3/arJ8fhr1+Sc3e2R/pJ
O/uQxbYlcqS9XrtPGFYtVtHwlOblmU+X672jyZ7119/ZtZc6a9R/nEhYWpqlTp2revHk6Z+emE9H7NWlgL9lnd
TDu6zZ3VKiif92uTOaHyz2Xl/Zt36IZQwfI1d1DFfzrPrZuclKSPv+omZyyZVffCdPl6p5DVy+eVxZHxOf6Hj+
4T+uXLlQen2Imx9PygoBdWzf94xwBAABeJwRhAADglWRnby87e3vdjo/Tvm1b1LrPIPnVaaCSftVM+g3rOELVG
zaW/3tNH1vHJlMmFSheUgd3blPFWvUlPdxn7MCubarj/j/Q0Gg74e3F9t+w9R5ixZlJqaqpQHyZJk/Gdqakq
a5rBjxw41bNhQAQEBWnnOktxze+mX71fr+MF9xj5H9+1VzXc/kG/FypKkOk1baf3SBTp2YN8Tg7CNK5fodncycR
iz+TtY2NpIk99xej/S7e+eOxvfpro+HjdGKqRNM2tLyggBvb+8OzRP/DdHRORk9hHTx9fXN6CEAAF5DPBoJAAB
eKb//slm//7JJl8+f1f7tWzTkw/eV1yZ2d+XKX9BYI7RtE/2wcLbxe2DbTvp5e
YQ2rVqm8yeOaXposO7fTZR/o2aPXP/n5RFydM2mCv51JElFylZQ9K7titkXpci5O5W7YGFlcXRKO1wqV66sDRs
2KCYmRpJO+sghHfltt8pU9zf28SldXns2rtP1y5dkMBh0cNd2XTx9UqWq1Hhi3TObb18mndDnNGDpQH1UpqaDAN
7Vi2kSlpJgGdDOHDlS5mm+pVOXqj9TgBQEAAMAcsSIMAAC8UhJvJ2jRuJG6ceWScubOq9sdNVb91HyaPQRfZ7Xv6dsmOgFM0QB

gAAXilV6r+jKvXfSXP/x+OL9rhjDVp9pAatPnpqLefsbo89tOm33mrSrXeax/SnZcuWadGiRYqIiNAle3edOnJ
IcOYMkYu7h958r4kk6YcFsxWzPOrBU+bKLVdu/bFnl2YMHSgXd4/HruSSJEOqQU7ZsunjoWNkZWWlAr4ldf1yr
L6dPVVNun+qa5cuaPaIwRo8e4ky2do9cXxNe/RROx59/hrvpLEmLwiI+eOQ1qxZozZt2igqKird8wcAAHjVEIQ
BAAC8JH379lVwcLCaNWumlUcvKY9PUV27eF4rp4frfa/erp6tvfvsqS8G1Jybrm1ZsPY4/GRF9jxkkU9c5gDRTEahEkoedAT
AV9wQcZ5nZ9ZkgUA80RMPxj3m4wpZsMXqf2JCgsH1TZ6OnY71A4AnbfYLwMotPnSpQZwWWHW6Cv9JgX7/B4yOI8zfB39lCOdRmdPQJFs9+zqdPyGz1DaSzZh5f0lB1vn6iJ5f3/JfoCqrvPs5x/BBz0tHfRHv4qXDjdbDNcfFlNDRt/7lk9bbqY5FnQ0akIjvTHBZRfXBtLYRktUxpHhfxtQadRvsbCNx7/J1yHX65F2PgZBrDe1nVBH5/Q6DP4I7bRdpNydx6L5i70TX1oAvxCM3IbVs8kNbnL6n5TLwmgqg9kZbMRzI2xNX3hhEIkQm2srDgm5GRoBAv1vXL+Y9DaV0HW2k4gaKt3C6uLRAG4j8ChRpbAXNRXSkMuP1BcELJk4Js31AQAAXgeW/9wFAAAAAAAAePORhAEAAAAAAAMAs8GgkAAAAntmtW7cUEhKiRcu
/UcL168pXtLg++myYCpYoLenhGymXhI/Rz8sjlJiQIJ+y5dVpyCjlzJv/iTXv3r6txRNHZ69effff3xsTUnate4Hr
VsyXycOHdTt+JvEeCtU65StquuvXnJGh2rx6mWwzZ1arz4TYI6PCGo0DSdZGRSUEmVhEOkM4exLgsSvxbCe8bhL4UutYKNE0Sg69h8Dk+KUMPcbNl0uGwvtpV2cyeSz+QSf5fpMEHC/yDhLsmiFG9OtRuD3E9+hn6rWi9qfQvWQz4SpYH0Aq9Ua97G9n4vXNKUDPG2PABn+RtwROi29rL8EKkB5ZhvyB4/tYbxL7sz7Yp9/95xJSxHNzaUPr7BOU/Ld9hxcdSB9NakbcV9Xj0dDZIzmPAhIAVV+Ac0CmPPV5nhTCtbu4gtMfJ/3gFxbQr3ZyKgvFg1knr+ntrHTZDFpdOAsBFjI3tVDlSOlVgqEp4Mf8u2MsaPqhCrm9uoqPfcFqF6IcJUZx1bU/z3ZEOFDDGfQnRWr8nXXo7ctl90VLjXemdJzAhJrqk7i2W/yKVxdAm9ZKdfCkfNuzWn9wLcZ4RVKQXfLgU1sQVNFnDXjcwpKRS8R9K1GTvZRwl2Z47nZ6TEqW/YFj9DWYPtO9LAEVHQPu7ezbKMqGkvfmgwgUJmoFkqlAGlvHLfPN7V1XahW5zj+cR1Om+P2u14fW8E+gtRq0lwlDhrUwYqWvdHBt8Q0rSF

/dtZUtatcQuOql9B3s6eZtMXs/O19G9VVyoMH6ZpLSkqKQkJClC9fPjUvlV9da/tp+ZSvZDAYjHQMBoMWTxyt9
tVKq3mp/Apt1OQXT598at27t29r9ojB6uxfQc1L5dfAZoE6fnCfSZ/w4CA1LpLT5DOsQwtje3LSfU3o1OtyhV
W97pVtX/HVpPzV8+aoh49eqRrvgCAf8ajkQAAAHgmDx48UEpKiuzs7HT/b5b2dnpSNRuXT5/VnFXr6hk5WrGt
iwOjipUsoy07otS1YB3H6mZ+iBFqSkpsrG1NTn+Z820yuNTXOuXLdLt+DhdPndWSffuR7rw5H/aqTfzDyJQffCgZ/Wyg5S6XP20Mvjam8erojC9Q7Ybcz0ZgTPVxuJfv4BpQrsVnmJahSl6Gm5s3Y9
jOzvdf1VqaqrG9+mm9zr1UOmqNfXlJx21ftlCNWj1kSRp4dgRqtOstdxz5X6h17OdH6fPmjeUb8XKGjRjxds
+nS6ZPK6uRk7DN3VKiif92uTOaHyz2Xl/Zt36IZQwfI1d1DFfzrPrbukd/26KtPu6pl7wEqX7O2flmzSqO7f6Q

```
xK36Sd+EiOn30Dy0JH6OB0x4GqCM//lClqtRQHp+iSnnwQNND++vjoWNkZZ2+/3wJCwvT1KlTNW/ePJ2zc9OJ6
P2aNLCX7LM6KKBNB0l/rVrsMWq83HN7a8mEORrWoYUmfL9ZmWztHlt3SsinOnvsqHqGhcvV3UNbv1uhz9s1lfj
vNyubx1+PpJap9qa6jfjK+N0mUybjr9cvXaiThw5oxJJI/b51o8b36abZ2w/IwsJCl8+f1c/LInTkw0/pmi8A4
J+xIgwAAADPxMHBQX5+fho2bJhuXI5VSkqKtny3QjH7onTz6mXFXb0iSXLO5mZynlN2N8Vdu/LYmpmzZpVP6XL
6Zsr4x9ZMqzLVaqp6YCP1/6CBJg0IUo9RE2Sb2V7TQweoc2iYflo8Tz4+PqpSpYoOHTqU7rkPbt1Ys74YpPljh
unDisXUvmopLQ3/0tj+3Zyv1SvQXy3KFFCnmuU0/fMBunvnjrF948qlal2hiH7/ZbN6NqiulmULaliHFrp55eE
cb928oYSbN1SvxYfyLuSjCv51dOHkMUkPQ6UTB/cZg5wXadXMycqeI6e6jxyvQiXLyCO3t0pXrSlP77zGPkf37
VXNdz+Qb8XKcs/tpTpNWymvTzEdO7DviXW/XzBTZaq+qXfbd1XuAoXU/JN+yleshH5cNEeSdOHkceXxKaYSlaq
qpF815fEpqgunjkt6uFKsaPlKJnvEpdWOHTvUsGFDBQQEyD23l/zqva1SVWoYV2+9zFWLkmSdKZNc3NyNn6xOz
sa28yePq7x/HXkX8lG9lm2Vc0O6Em7ekCRNDw1W6z6fydHRMd1zBgA8HUEYAAAAntmCBQtkMBjUsUZZNSuZVz8
smKWqAe/KwvLZ/zWz5+jwF1KzaY8+mrxuh76K3KiKtetr1fRwlaxcTVY21loxbK2bdumDh06qE2bNs80zs2rl
8sus71GLl2j1n0GafmUr7R/+xZJkqWlpT76bJjGR25Wj1ETdHDXNi348guT85Pu3dV3c6aqZ1i4hi1Yq
WuXLmj
e6KGSJEfXbHJx89D+7Vt0/26iDu/9VXkKF9OD5GRN/3yaZN/3yAOn8+WlZWVs807qfZu3GdCviWOpefdFK7yiXU573aW
r9skUkfn9LltWfjOl2/fEkGgOEHd23XxdMnVapKjSfWjdkXZbIyUJJKV6mho/uiJEl5ChfVpdMndfXieV25
cF4
XT5+Ud6Eiij17WhtXLlWLT/o/03wqV66sDRs2KCYmRpJO+sghHflt8pU95ekf1y1+DjpWbV4aPdOtatcQj3qV
dXXocG69f9BlyTl9SmmI1G7df/eXe3btlkubh5ydHHV1siVsrG1VcXa9Z9pzgCAp+PRSAAAADyzAgUKaMuWLYr
4/YTu3r4lF3cPje3VWR5eeeTs5i5Jirt+VS7uuHsZz4q9dVd6ixZ9YO9M7r4YtXKl7iYmP1HxW508e05bIlfpy5
TptXLFYRctXkpubm5o0aaKPPvpvIt27dkoODQ7pq5vEpqibdP5Uk5cybXz8umqMDu7apvDjsZ+7rm91OK
T/vo6tL86/e2RzAfJyeocGmZcbVW/ZTstn/LwMToLCwt90n6a5owM1ezhg1W2hr/8GzfTqhmT5FuxsmxsbTWw+
Tu6df0G6rf6yPjI5P06f06sflo8X4Ft06lR5x46fnC/Zg8PkbWNjd58r4kkqqUPIF5oW0k+dapSTlbW1LCws1WX
YGBWvU0mJdeOuXZVTtuwmx5z/tjIwd4FCatErWEM/aiZJatl7gHIXKKKTQdk3Uuu9n2rdts5ZOHitra2u1Gzjsq
df6u+DgYCUkJKhIkSKytLJSakqKWgQFG98g+ryrFnPnLySn7G7a9v1qxeyLMlk5V6ZaTVWqU1/uubwVe/a0Fr
apS86tdKIJZGysrKSf+NmOhPzh4ICasrBxVWfjp+m2/FxWjJxjIbO/0YR48PUb/0aFShQQLNnz1auXLnSNGcAw
NMRhAEAAOC52dnby87eXndu3dT+7Rv1Qc1v5f3VM3TWbFrnDyuBJv31JqSoqS7t/T/
buJss1sLOmyzZzZJDhxcfNQ/P+fL0lFy1XU6G9+NH6/e0qENn/7jb5cuU4hrd9TQOsOKlvdX0GBb6pYhUrK61M
s3XP4XwZDqgoUL6mWvQdIkvIXK6Fzx45o3ZIFxiDshwWzqfM3sotV279sWeXZGgwdKBd3D5Wq/PiXF6RF3
WZtVLfZXz8Xm1YtU+U+YsWeVTurx61K+msOU/6HrsJX3Vu4umbtiVpprLli3TokWLFBEROv27jp15JDmjBgiF3c
P43yeRc/R4Zo8sLc61igrSysr5S9WQl1UD3tWJQWeMff6en6LK41NM33Wr76jDduHSrpVO3WNjbqONh0r7pJA
4LUoHV7nTocrd0b1urw/v0aPXq0evbsqRUrVjzzeAEf+HRSAAAADyzn376SStWrFDeaDAlW8XWDfnwXKX1D
+jZrKwsJCb7fpoG+mTdCejT/pzNHDmti/p1zcPfRGrXrGGqFtm+iH+iHHbON33//VbTlZ06HK1zJ
x4+8nbx1AmdOhytm1cfXcXu3atUtfffffWVihUrUjmdn53TP3craxvSAhYVSUw2
6cv6cRn78ofL4FFXfiTMOZsVadRg8XNLDVWBPPl8mbzP8X9OG9Ffb/oN1Nl/Ox0x+6d6
R7/4zi7uSt3wcImx3IVKKRrly5Ierg/VsT4UWobHKoK/nWU16eYGrT6SFUavPPImx5N6mZ3Mwn5pIerxJyzuz+
2f8LN61o2eZw6DPpCxw78ppx58ytn3vzKW64YUz6k6+Kpp7/V8U99+/ZVcHCwmjVrjw+RVWz4fsKbNtRK6eHG
+crPVy1+HfxTxmb9NeqxUW/HdfOTXsVtvwHPXiQ/NRVi55eeeTo4qrYM6cf235w1w7PjBz3V8
ZsmRRkyZNtHnz5jTN99+2fv16/FS5cWI6OjmrdurWS/vZM/j4eBUuXnzpzpzpzJwBECwKMIwgAAAPDM4kM4uPj1la1bN
/WsX1OTgz9RkbJvHoq/wG3Uu8o5AZi0zexhd79rTJ3kmJtxMOY+jAJ9aUpD0
b16nPe3UOonNrSdK43l3U5706WrfEdLPyuGtXtWLBLX/bJjxWKGSZRTYrrMCAgKObNkyzZkz54XekxHDshgS
NWH/YeocOlyypmvgHET/Gf18zcRyurkrAr+dZX6/6vXHjx48P//TFZqaupzj1uSipSpoInTpgxyPnw
sL+XBAz1ITn5kvzZLSysZnjKGwqXL6cDOXOyOHdixVT6lyz22/5yRoQr8sKOyeeZUaspfK/ikh6v3UlPTToIvM
TFRlo8d68PA8e+rFo3n/P+qxeN7e/s70314u5hXLX4d1xdU2Tu6pI9G6PBmxJ9+9p5rCB6r6vx5mKsysrJS
amqKU///9TU5Oef0qYViuVzbamqWrRooY8/8lg7d+GN7cNCwcHXNl1cWFhaPfLp16
6bTp08/ts3CwnbLLly9/YsFDs2ezJDs2FDZs2eXo9y4ocDAQGtVs2FQDs2eXo6Ojqlatk2bNhyocDAQGXNmlVlypTR77+
bvt2zW7duGjt27DPPG8DLxaORAAAAeGZNmjRRamqqihYtqpeP5XUVazLcmVHY/Ta0zbaLrBeJX676hK/Xeee
l3/Rk1NVog9iXN2t0fqS1KTbr21ZOKYx5zx/HLkyasHycn6lX+zto78tkc/LVnwPXir1/TiqkTNDziW01l
SVidn5S5QSN/Pm6FSVWro4M5tev/hT/K7lyxbq46+WOGK
ZOtndxy5dah3Tu15dtv9GHwEG0dif17ytXdU60+HShJfB/vZ3LLp0
+qR6jJkiSCpYoxpQsnT+i3rRt17dJFWVpaKme+++Ammb
```

/dc3lo8cfRjVy2+UauecS+233/ZLMmgnPkKKPbMKc0fM8xk1eLdO3e0bPJY+dUJkHN2d8WeO60FY76Qp3c+la5
a85FxLp8yXmWr+yt/sRKSHq5anD9mmA4cOKBJkyapSpUqaZrv39WsWVMlS5aUnZ2dZs6cqUyZMunjjz9WaGioJ
GncuHGaM2eOTp48KVdXVwUGBmrO6NHKmjWrJGnu3LkKCgrSOqVLFRQUpHPnzqlq1aqaM2eOcuTIoWvXrunatWv
q2rWr7Ozs9M477+jw4cOSHr6tc8+ePZo0aVK6x/13e/bsMQkBo6OjVbt2bX3wwQfy8vLSpUumf+5Mnz5dY8aMU
f36T37RwP+e8+OPP6p9+/Zq3Lix8djbb7+tQoUKaePGjcqcObPGjx+vt99+WydOnJCnp6eGDx+uW7du6bffftP
UqVPVsWNH7d27V5K0a9cu/frrr5o4eJzzR3AyOMQBgAAALxAeYsUV9vgUK2eMVmLxo1QsfKV1LL3AIX37/lM9
WYND1Fgu85y9fAOHus+crzCgz/R9wtmq2H7LipYovQLGXvBEqXVL3yWFo0bqeVTvpJ7bi+1GzDUuLm8JPUaN1W
Lxo3QhL7ddTs+Ttlz5lLzoP4m+3tdu3hBFhZ/rcQqUraCgr6crMXjw7Toq1HKkTef+k2aLe/CRUyuf//eXc0c9
pl6fzXNuJIrm2d0tR80TJMH9pJ1pkzqMWqCbOOyp2k+4eHhCgkJUdeuXRV7+Ypc3D1Uu2lrfdC1l7HPux266d7
dRE0b3E93EhJUpFyFNK1aXDRupK7HXlJWZ2dVqt1ALXoFG1ctWlpZ6szRw9q8erkSbyXIxc1DparUUPNP+skmk
+nbJs/GHNG0tZEau2q98Zhf3bd1aPd0VatWTT4+PoqIiPjHuf5vGH0tMUmz5sxVYNtOGro4UjH7ojRQJAM3kV
UqkoNRV+7rff7DJFHbm9dPn9G0z8foJNxicYX0uy9FKc7iYnqN3S42g37SpaWFprQr4eaduqmoC8ny2AwyMXNQ
5/PWaKSftX07boNqvluEy2LPqt+H3VUt+Hj903xx79w4E+NfHI8td3N7a+XGERHR2vevHny8vJStmzZjKHb3y1
evFi1a9fW6dOn//F+/WnBggV644L31JktLS92/f19t2rTRlClTFBkZKT8/P+3ev
VtvvvmmkpKS90abb+rrr79WdHS0kpOT1bZtW33++ec6fPiwfH190zyOP6WkpCg0NFQLFy5UbGyscubMqbZt22r
QoEGysLCQ9PAR6iFDhmjGjBmKi4tTlSpVNHXqVBUqVOiptSdPnqwxY8YeNjZWpUuX1Unh4uN544w1je+/evTV
lxlyZJFoOaNUsuWLY1ty5cv1/z58xUZGnuOQGvGgvD0zYieEUlJCTIyclJ8fHxcnR0ozOjhvDBP+j+pr7J/+sv
r75jfqyc985P++3Nkfq8e5veX//r8pP/+HJnfq4f5mXrd5vhfn5/0fD+jg1s3Vmpqir5YtNp4rP8HDeRbqYpaf
/rZI+fvXLtGX4f219xdhyRJG1cu1eSBvTR53Q7jSx3WRszV8ilfada/ir3YChWjV
jkm7F3VStD1pq2uC+T32zaXrm9/vvv8vf319t2rRRx44dH2k/dOiQmjVrpgULFqh06dJpqnnt2jXVrl1bX3zxh
QICAiQ9DJneeecdlStXTv369VOmTJm0cOFCzZkzR999952cnJw0fvx4nTt3TmFhYVq8eLWrl2rRYsWafr06bp
586b69+8vSc8UhIYOTKLx4ce3du1ft2rXT8OHD1bPnwzA9Gd+hxnWnwzA9LCxMIOe01Lx585QvXz6FhITo4MGD+
uOPP2RnZ/fYukuXLlXr1q01bdo0VaxYUWXKlFZu/OKYukXLlWbNm00bdo0VaxYUePHj9fy5ct19fy5ct19OhRubu7KzIyUh07dnTuq
u+Ph4VahQQT///LO8vb8enXr1k2xbFq2LChZs2apUyZMkcScV7r169/rkd48epIa1bEijAAAAAA+
A/6L73ZdOGW3Uq4dUslGrdVjE22R9rnfPujchcoJPsKbykmjTVXf79EdlmyKk/9Joqx+StAGjj3G4V1+OiVKlW
ShaWlnFyza+DMxbqcPb8uS6rxcV9NDw1W7YC35Z7TS51GTNmC/H65rvvNXLJd/r0izDt275FNfwqasaMGXJyc
krzPHfs2KGDRsag7m8efNq8eLF2r374SPeBoNB48eP1+DRo0ePp24g+p5vrebCOB9kE5nQvRq7Tvb6L3mtl+fPwWD3ehmmlI2qUoycOZxrEDAt4fj
mbPe0QAsTWazl7dehJ2OkNGTl12oJbg3+fQGA28Bgb9G8iRhGz2bK+XU8sbhDDh0pMD7
D0pQ+Jb0ikKaW5zuPBRQc43Ih4jYmzfPzTzmPY8GQjX7CmRvmhtAuBzrKQxaC+0Sgkm83LvrbkC5eHjXcSivNNAVbnxN3KhZkDdmrBNLTwgn36SDvTlsvekdHjz0WBmaVd5s46BWC1PlWK09RUGi+Ijb/9qVp5Rjqpbrq88+6VWWXhlv3lkUB1Mlqc1nXrV1uMx
7Q8/ItcHjIdlCpWJ2+LSzIMslPsMdAkqBxkUUA7aLB1aEP1IR5g3LcRZSHsHTCn4lMc9JZ2r1n5A52U3nLDN3j6VrEBbMd+kdrltC4DIdHGtdAsFaX5XQbzw3RUvyd7BUWvfAyYxjf75S4CgsjQ5c9IVvJHhd+Qge2BkFwkUkQhBsDpAWEmB+ZVJXWBWgKHJyHHdXwxOrn8eqBdVJwTGTIoR+mL2ZEvQDsebJ0YMhP2mc5PcwsDChMohM6kXh+tXeTFYgLLrvf3jDe+ZIxWnA+A0uXb4lgSfL0lk40P3+oiV6V8kc9WyvG52wLsz7PKj2G1YG4uZRYl8+7Um2Ko9x2CEWTh0v5RVo6eY2bMgdBi0GH0rWqsbPk+DIuF1qUGNlo8M0iPHOo5EXz/e3RzuX+6sWsqCHsOXCkDqg8edLgmE0cNTPSxEz7bo1PIdgJVxXXbEc/ZI4fD+zVaZqc2jdhmWoFiqSfW3IRGyEDcQhaHFr0brcsWJnslFQ4MzsR+YJg3QsCK16YN1o4AGdI

uHQCgcuWCYt/HRbDCr42MjIr9Nw0MDKCsrFzs9xT+e59KSEjAwYMHMW7cOFy7dg1NmjSBoaEhOnXqhHv37iEzM
/Objutz7O3t5b6uXr26rHB38uRJtG/fHjVr1oSOjg6GDh2KV69eISsrS/Z8TU1NWRHsO+8HgDZt2uDq1atITEz
EqlWrkJiYiODgYMydOxdDhw7FyJEjce7cOfj7+yM6Olohx/Tp8fn5+cHKygrDhg1DO6ZNcerUKQDAhAkT4OzsD
HNzc7i4uGDu3LnYuVN+1mFhoa9p06Zo3Lgxxo4dK/v+wkLfH3/8gfr168PBwaHYQp+1tTVWrlyp8GMTMpoRRgg
hhBBCCCEi4795d5Gx6as2yj53Hz4S7sNHyj3u1OP/ZgO5ePSHiOd/ucebu3bB7pjHRf7dSYtXFxmzsnfA8sNnv
zn3p06ePImUlBRM9Si+CXzY7u2oXK06GrZ2LPbxx4nxyHyXLjd2/tA+MMbQpmvPIs/XNaiM3//Ziq1LF8DPsx+
kebkwqWONaas2Fpk9duPcGTxNScL4hStkY1OGe+HB7WhM79cVrVoOl812+i+FGwBkvs/BC1UduQOB3qjq4ANTQ
qxqZTALQ+hXMcbBq7egZFdQZMvKeIfoW7fRbtCIYjcSgCpgWd8ex65GwaRzPwAFBcbzl6+iy+Cimw8wxjDrDx8
Mmu6Ph3omeAp1pOUX/PczJQUlhFglPTT/qiP7MlVV+T5OEokE+fn5SEpKQrdu3TB69GjMmzcPhoaGOH/+PEaMG
IGcnBxoamp+9vu/1Mdu1KhRWLRoEfLz83Hjxg307dsXmpqacHRORHh4eJHCXEn9V6Fv/vz5iImJQXp6OvLy8vD
+/XtkZWXJju9rC32FYmNjERwcjBs3bqBdu3bw9fVFly5dOKBBA7Rr1O7hxydUXAthq1atwl9//YWnT5+iYcOGW
LFiBZo1a8YzEiGEEEIIIYSQcqJjx45gjCHO/pNiHx88acYXZ7MVV7jr2H8IOvYf8tnvqWPXELPWb/vPbA5tneD
Q1klurJKGJqYsXQMA8LCu/p//xqea0nfA7sDlqFK9JkzqWCPx3m0cCFoDl94FhUCJRIJuw35ESOAyVDe3gHFNU
2xbvhAGxlXRzLWz7N+ZPbwfmrl2htsQbwAFhc8VOyegdoOGsLJ3wMFN/+BDdhZciikwnty1FbqGlfGDSOcABbP
cdq5chNib1xF5Ngy16tSFlu7X74RZ6NNz+DIrBw/eZMqNP854j7eSbKzafxLS/HyOHjkFj5WU8JgBJ6PvAwAOx
D2Flm42rj1JQ26+/M/G5Uevi/1vAQV97LJVNZFn/QP2xjwCAOy+mwotHV0kv3mHGO/eFPm+7zmHHxN7oU+ouBX
CduzYgUmTJiEwMBDNmzfHOqVL0alTJ9y/fx/Gxsa8YhFCCCGEEIIIVz8+PtcbFu+EGv9ZyD91SsYGFdFh/5DO
ffnibLn9PxxDN5nZyFw1i/ITE9HvSY/YOY/W+R2v3yakoR3b17Lvm7t1gNvX7/C9hV/Ie3FC1jY1Mfv/2yBvlE
Vuf9+2ssX2B24DAHb9svGr0wd4O41CvNGDYNe5coYt2BZKf4fKCCOPnZAxSzOCRW3QtjixYvh4+MDLy8vAEBgY
CAOHTqEDRs2YPr06bxiEUIIIYQQQgghXGhoa8P7V394/+r/2edIJBIMHP8LBo7/5bPPCQy7UmTMbMbYi3bIbY5+g
bVSn2e/uNmYR+YyZ98XsV6eM+dlsWB8C2aQsMnjQDK6aN/65/73N97FZM98WhzRtK1MfuewixOCcmXAphOTk5u
H79OmbM+L8prEpKSnB1dcXFixeLPP/Dhw/4/480GD7Ou3b98CANLTO4s8V8iyMt7xjvDN0tO1vvq5dHzlz7ccHyD
+Y6TjK3/o+P6P2I8PEP8x0vGVP3R88oR2jGI/PoB+Rj9Gx1f+lPR3cPrqoCLj4/9cLhtr32cg2vcZKPc9m6/dl
z3eoqMbWnR0k/t+uxZtsPna/SL/rZ/8Fxb5b9W0rIMFOw99Nh9QsnMolUqRl5sjNy7NyONeXi6Ma5li0MQZ2LN
2BbYsCoB146bo+/MErPGbhqzMDEiUlJDz4T3AmNz3f3j/vtj/1to5M9B5OHCoa2nJHvtxZgDWzp6GQ5vXw22oN
2pY1C7yfd96Dsu7whrRl5aPAoCE/dczSsHjx49Rs2ZNXLhwASlbtpSN//LLLzgdHY2IiOj/zPe7du34e/v
5d3nFIh9mOk4xM2Oj7hE/sx0vEJm9iPDxD/MdLxCRsdn/CJ/Rjp+ISNju/bMMbw7tO71KhR44vP41KbrK
yMp49eyeY3/uzZM1SrVg4MnQRKlSrJjenr65TGK6GL1MzJt1YvtP7gmD6M/lmKkUySTVdmRHQYBx4+5EYV6r4w1uLO8kqfcvP7u4TI4NTOmIc3FqZ75JdLsS1

IuQS5cuYdCgQUhOTi6yxl+If8Cys7OxadMmxMfHY9GiRfDx8ZFNR//UkiVLyjhd6RDri0glJSU8ffqU+3TmOib
WYgMg/nMo9uOrVq0ajh07hoYNG2Lr1q3w8/NDVFQUNm3ahLVr1+LGjRu8I5aYWM/hu3fvkJycDHt7e5w8efKzS
+saNmxYxskUS6zn7+jRo3jw4AHGjx8Pf39/6OjoFPs8X1/fMk6meAYGBrh06RKsra2xfPly7NixAxERETh+/Dh
++uknwb/O1tTURExMDExNTdGvXz/Ur18ffn5+SE1NhbW1tWD7T9E1RhwMDQ1x/vx52Nraok2bNhg2bBhGjhyJp
KQk2NraCvbnMyYmBg8ePED37t2xcePGz7bJ6dGjR9kEYkSUNDQ0WHJyMmOMsV9++YUNHTqUMcbY7du3mZGREc9
oCrFlyxbWunVrFhMTIxuLiYlhbdu2Zf/++y/HZIrRsGFD1rdvX3b37l325s0blpaWJvchNG/fvpV97uTkxN68e
cMvTBnR19eX/XwuW7aMtWrVijHG2LFjx5iFhQXPaN+sV69esnMYFBTEsrKyOCcqfVWrVmU3b95kjBVcb+rUqcM
yMzPZ//73P9aoUSPO6b6d2M+hg4MDe/36NWOMsdmzZ7PMzEzOiUqPuro6S0lJYYwxNnToUDZt2jTGGGPJyclMS
OuLZ7QSEfs5XLZsGcvOzmaMOe+gEEVFRTGpVMoYY2z48OEsPT2dc6LSpaWlxRITExljjLm7u7MFCxYwxgquM+r
q6hyTKYadnR1btmwZS0lJYbq6uuzChQuMMcauXbvGqlatyjnd96FrjHi4u7uzTp06MX9/f6aqqsoePnzIGCt4D
2FlZcU53ffZt28fy8nJYYyVn/NHhTCRqlKlCouMjGSMMdaoUSMWHBzMGGPswYMHgn6hXXMjS0lJ2fB+7du0aMzc
355BIsTQ1NVlcXBzvGAqjpKTEnj17xhhjzNnZuUIUwsT0IlJVVZU9fvyYMSZ/LsVMbMUGsZ9DdXV1lpqayhgT5
/F9zMrKiu3YsYNlZGSwKlWqsF0nTjHGGLt58yarXLky53TfT+znUFlZWXZMYjw+sZ+/j4/JwsKCvXz5knOiOtW
sWTM2bdo0dvbsWaauri67MXTx4kVWs2ZNzulKbteuXUxVVZUpKSmxDh06yMYDAgJY586dOSb7fnSNEY/k5GTWt
WtXZm9vz9atWycbnzBhAhs3bhzHZN9PSUmJPX/+XPZ5eTh/1CxfpDp06IAff/wRDg40iI2NhZubGwDgzp07MDc
35xtOAZ48eVLsGnipVIpnz55xSKRYzZs3x4MHD1CnTh3eURTi42b54eHhFaJZfv369REYGIiuXbvixIkT+OOPP
wAAjx8/FtxOU2JvtF4cExMTXLx4EYaGhjh69Ci2b980AHjz5g3U1dU5p/t2Yj+HFamJ7oQJEzB48GBoa2vDzMw
MTk5OAICzZ8/Czs6Ob7gSEPs5pEbW/0eI56+iNcv/888/0atXL/z111/w9PSULaXbv38/mjVrxjldyfXp0wdt2
rTBkydP5JYqOjc3R8+ePTkmKzmxn7PhiwyLiQ28dQs3xSZtLS0jBz5kykpKRg9OjR6Ny5MwD
Az88Pampq+O233zgnLB13d3c8evQI69atQ+PGjQEA169fx8iRI1GzZk3s37+eRr7/+ipiYGEH16btw4QImTZok2kbrxfnf//4HX19
fWbEhMjISSkpKWLFiBUJDQ3H69GneEb+J2M9hRWqiCxT8vUtUtJSUGHDh1kbwQOHToEAwAEHToEAwMDtGrVinO67yP2c0iNr
AsI9fxVtGb5QMHN5fTOdBgYGMjGkpKSZP14hSo3NxcaGhq4efMmHRowDuOwtA1poBQrzEfU1ZWxpMnT4r8nhV
OKhDi+aNm+aRM5OXlISAgAN7e3rId68TmxYsX8PTOxNGjR2VFory8ePHTq1AlBQUGC/gMNFDSB/JREIhHsH7CK1
iyfMYbU1FQYGBggLy9PVC8ixd6g9GOfKzbo6+ujdevN9P7GfQ7Efn7+/P6ZMmVLkGpqdnY2//vpL8HfCAfG
eQ2pkLWwVqVk+UPC6+syZM4iPj9egQYOgo6ODx+x48fo3bt2sjOzuaUrGTKW7N8KoSJ1MfbropZbGwsYmJiABQs/a
lbty7nRIqRnJz8Ts7Y8+ePZ+9+IlBfn4
+1NXVcefOHVhZWfGOo1DJyckwNTUtF1OaS0tubi7q1auDyckwNTUtF1OaS0tubi7q1auHgwcPwsbGhnchasI51DMxHinuKLZtGkTBgwYgEqVK
vGOQr6Dl5cXli9f/tlCmBgkJye+fOSElJwYcPHxAbGwtLSOv4+vriw4cPCAwM5B2xRNavX4/Q0FBs3rwZhoa
GvOMoHF1jhGn58uUAgIkTJ+KPP/6QKzhLpVKcPXsWSqEJgt8des6cOZg6dej8jepnJ0WUFSqEiVSPHj3g4eEBT09P3
lFKVU50DhITE1G7du1ip6KxZEydnh1KxZEydPnhRlIayQ2M/
h3bt3iz2+7t27c0qkGEpKSnj27Bmq.

aI5VKERQUhFOnTuH58+fIz8+XezwsLIxTMsWoV68e/Pz8MHDgQLn38bNmzcLr16+xcuXKMskhrvUepFjv37+Hu
ro67xgK9eLFi2J3CsnMzBRNA+isrCycP38eZ86cwenTp3Hjxg0OaNAATk5OvKOVyL1797Bt2zYAgIqKCrKzs6G
trQ1/f3/O6NFDNIUwoRa6/ktmZqbsd8/AwAAvXrxA3bp1YWdnJ6qeGmJ4kf85Yj+HUqlU1sTayMgIjx8/hrW1N
czMzHD//n3O6UrOOdER+fn5iI2NLfYFcrt27TglUxyxn8OnT5/Czs4OQMHmRm/fvgUAdOvWDTNnzuQZTSHEfv7
i4+PRtWtXAAV9ewpfe06cOBEuLi6iKIQtWrQInTp1gq2tLd6/f49BgwYhLi4ORkZGstdwQhYdHQ1XV1fo6ekhK
SkJPj4+MDQORGhoKFJSUhAcHMw7YonQNUbYfH19ERQUhK5du6JBgwaieW9bKCUlBa1atQIAaGho4N27dwCAoUO
HokWLFlQIIyUjlUoREBCAwMBAPHv2TNZDa+bMmTA3N8eIESN4RyyRpk2b4tChQxg3bhyA/1sTvm7dOrRs2ZJnN
IVo1aoVbty4ARsbGzg5OWH69Olo166dKNaKa2lpydaCV69eHfHx8ahfvz4A4OXLlzyjKVx8fDw2btyI+Ph4LFu
2DMbGxjhy5AhMTU1lxyw01tbWuH//PszNzdGwYUOsWbMG5ubmCAwMRPXq1XnHUyixbh0v9nPYoEED2QyN5s2bY
+HChVBTU8PatWthaWnJO16JXbp0CYMGDUJycjI+ndQvkUgEPWO4kNjPYa1atfDkyROYmpqidu3aOH78OBo3boy
rV6+KYpc3sZ8/AwMD2Ru3mjVr4vbt27CzsONaWhqysrI4p1OMWrVqISoqCtu3b0d0dDQyMjIwYsQIDB48GBoaG
rzjldikSZMwfPhwLFy4UG73Tzc3NwwaNIhjMsWga4ywbd++HTt37oSbmxvvKKWiWrVqeP36NczMzGBqaopLly6
hYcOGSExMLPK6plSV2SJMUqbmzJnDLC0t2b///ss0NDRka/u3b9/OWrRowTldyZ07d45pa2uzn376iamrqzNfX
1/WoUMHpqWlxa5du8Y7XokZGBiwypUrs4EDB7I1a9aw+/fv846kMD169GBr165ljDE2efJkVqdOHTZ37lzWuHF
j1r59e87pFOfMmTNMQOODubq6MjU1Ndnv4Pz581nv3r05p/t+mzdvZhs3bmSMMXbt2jVmZGQk64O2fft2vuEUK
CkpidWrV49pamoyZWVl2fkbP348GzVqFOd0JSP2c3jO6FG2e/duxhhjcXFxzNramkkEmZkZMROnjzJOV3JNWz
YkPXt25fdvXuXvXvXnzhqWlpcl9iIHYz+G0adPYvHnzGGMFr8tUVFRYnTp1mJqaGps2bRrndCUn9vM3cOBAtmjRI
sYYY/7+/qxKlSrsx9/ZGZmZoLvIVlR6OrqsgcPHjDG5HugJSUlsUqVKGMphB0jRG26tWri+q936dGjBjBZs+
ezRhjb0XKlbL3S/r6+szb27vMclCPMlJGqU6c01qxZg/bt28SMo39j3hFLLD4+HwsWBUVBQyMjJCgBMvLw6e3sjNTWVd0OSFomuMs
CxatAgJCQlYuXKlKI7nU/n5+cjPPtPSOuIPnDtPSOuI
ZjXyj/Px8dvXqVebp6clUVFREsetnRaClpcUSEhIYY/K/g4mJiYK+2zhnzhzyWmZlZZDwrK4vNmTOHQ6LSYWhoy
GJiYhhjRc+fhoYGz2glJvZz6OXlxdLTO4uMZ2RkMC8vLw6JFMvZ2ZkdOXKEd4xxSJfZzGB4eznJzc4uM5+bmsvD
wcA6JFEvs50+sJBIJe/bsmexzMe8+P2LECNazZ0+Wk5PDtLW1WUJCAktOTmYODg7M19eXd7wSo2uMsPXs2ZPp6
ekxCwsL1q1bN1HtXM4YY8nJySw/P7/IeH5+PktOTi6zHEplU24jZc3W1hbnzp0rMh4SEgIHBwcOiRTLOdERwcH
ByM7O5h2lVERGRmLx4sXo3r07KleujJYtWyI6Ohrjxo1DaGgo73glYmlpiVevXhUZTOtLE8W6/kL6+vp48uRJk
fEbN24IejeiOXPmICMjo8h4VlaWKBoEF/rc7qwPHz6U6U6U6yciRGI/h5s2bSr2bON58mQEBQX
h+vXriI6OlvsQA7GfQ2dnZ7x+/brI+Nu3b+Hs7MwhkWKJ/fwpKyvj+fPnRcZfvXoFZWV1DokUIz8/X7aRSuGMj
eI+xNCHcNGiRcjIyCxsTGys7Ph6OiIOnXqQEdHB/PmzeMdr8ToGiNs+vr66NWrFxwdHWFkZAQ9PT25D6GzsLD
Aixcvioy/fvOaFhYWZZaDmuWL1KxZs+Dp6YlHjx4hPz8foaGhuH//PoKDg3HMw4EHe8UrMwcEBU6ZMwbhx49CvX
z+MGDECLVq04B1LYZo1awYHBwc4OjrCx8cH7dq1E8WFw1e3ZWjx4dHJYLaCrF8u8R1Dx8EsWEVsLrNW59R/sm9b7
iQX5+PiIiIjBlyhQMGzaMd7zvxhgrdpp2VFQUDA0NOSQHRO7dsTSpUuxdu1aAAVNyDMyMuDn5yf45qVFPZp2yf45qViPYfp6
elgjIExhnfv3sntliyVSnH48OFidxsWmt69ZWMSiUR2XoX8JrWinMPP/Q6+evUKUWlpaHBIpRkU6f8X
58OFD2S3pKUW5ubno3LkzAgMDYWVlxTtOqdDT08OJEycQEREh12LF1dWVzSFoGuMsIl553Lg8z+fGRkZcue0t
FEhTKR69OiBAwcOwN3DrzjldjSpUvx999/Y//+/di0aRPatWuHOnXqwNvbGOO
HDkXVqlV5RyyR169fQ1dXl3cMhfq438SxY8fkCntSqRSnTp2Cubk5h2SKoIyAgAGPGjIGJiQmkUilsbWOhlUoxa
NAg/P7777zjfbPCvgoSiQT//OSP3ZmlUqRkZGGAKqiLR169aV+wMmlUqRkZGBn376iWNCxRLj1vFiP4f6+vpyx/cpiUQihlviYmJvCOUGrG
fQw8PDwAFxzF8+HC53dukUimio6PR6+/LHBoCFhgnaIcpvZEBoFkmcyldevU/Se3MCs1mtvtUYQXywaVfgqlPAt4omjnkPbMVkq1Mx21g8H
nNzoaGhgZs3b6J169Zo3bo70gKQ9cYYV9jPibGncsnTZoEoOA8zZw5E5qamrLHpFLpfIpl++jEaNGpVZHmqWL0J
5eXkICAiAt7c3atWqxTtOmXj+/DnWrl2LefPmQSqVws3NDePHj0etWrVbRZZGH5NGlYDuQk/x0BnU3Uux/+FBb0I/+ObMUG/+Xx01gh8mSEq82rRM4Ay54BNWdU
JQKVmIXzlz4mKqqKszNzbFo0SJ069aNR7xSk5KSgtu3byMjIwMODg6oVasWGgrU9jWcc3JbmdIaaCP4vuMXT0yLl5hAn6t7h1fgPp3oS3SfY5FBBdGGhfh+P6dx/2zSdtv4Hu7U8i8HWuAjIIY9DYEIq8WQeRj6LMA8HUzVlZlYMhUCqQj3QdlfywpfJzB8PfgW
uVUc3uxr5x+cY6NUzdtHPpWpc0fZFU/xTYiHU3D2/U3x2VqhIvR1fRqR8smXL8ZKqyjNTxhpr0XTBNhUnyjCoq
9ivXz+5a0nh76CPj49gN60Q+/krXLKTnJyMmKtgXTI/dpeSL8DLDx//v7+aN680a+ICjNx4kRUqlQJCxYs4B2lVFhaWmL
Pnj1o2LAh7ygKRdcYYV9jCiUnJ6Nz585ISUnBhw8fEBsbC0tLS0xLS/j6+uLDhw8IDAzkHfG7FC7LDQ8PR8uWVLeVmO
Bb+fE6ZMqXM3itRIUyktLW1cfv2bVHNsPmcK1euYOPGjdi+fTt0dXUxfPhwPHr0CFu2bJEsKBNbdUtxfPhwPHr0CFu3bsXPP/+Mv//+m3fEb/b
8+XPO798f4eHh0NfXB1DQQ8vZ2Rnbt28X1O6EALyfCvZRnbt29HlSpV+AYsAQsLC1y9elWwf4QruvDwcLLRu3Vq204tYvX//vkynZ5cls

Z/D5ORkmJqainKnpY/dvXsXKSkpyMnJkRsX+m5ugPjP4Zw5czBlyhRBL1H6ErGfP2dnZ4SGhgp2d8ivMW7cOAQ
HB8PKygpNmjQp8rO6ePFiTskUY/369QgNDcXmzZsF3RLgc+gaI2xi37ncy8sLy5Yt4776iQphItWjRw94eHjAO
9OTd5RS8fz5c2zevBkbN25EXFwc3N3d8eOPP6JTp06yi+L58+fRuXPnYptCl3f9+/dHQkICgoODYWNjA6DgTY+
npyfq1Kkj2KVZn5OWliYr+IlF4fTfTOkkEqirq6NOnTroOaOH4F6ARUZGQlVVFXZ2dgCAffv2YePGjbC1tcXs2
bNFOR8FAHR1ddGrVy8MGTIE7du3l81oFAOxn8OjR49CW1sbbdq0AQCsWrUK//zzD2xtbbFq1SrBv3lNSEhAr16
9cOvWLbkZtoV/+4TcI6yQ2M9hdnY2GGOyZSHJycnYs2cPbG1tObFjR87pSk7s5+9TUqkUt27dgpmZmWiO7UsN1
SUSCcLCwsowjeI5ODjgwYMHyM3NhZmZWZGCUWRkJKdkikHXGGH/HlauXBkXLlyAtbU1dHROZIWwpKQk2NraIis
ri3dEhUpPTOdYWBjq1atXtsvLy2BnSsLB6tWrWbVq1djkyZPZ1q1b2b59++Q+hE5VVZXVq1ePLVy4kD1//rzY5
7x9+5Y5OTmVcTLFONXVZVeuXCkyfvnyZaanp1f2gRRowYIFbPv27bKv+/TpwyQSCatRowa7efMmx2SK5eTkxHR
1dZmWlhZr3Lgxa9y4MdPW1mZ6enqsefPmTF9fnxkYGLA7d+7wjvpNmjZtykJCQhhjjMXHx7NKlSqxgQMHsjp16
ohiy/FCoaGhrE+fPkxDQ4NVq1aN+fr6sqtXr/KOpRBiP4cNGjRghw4dYowxFh0dzdTU1NiMGTNYixYt2PDhwzm
nK7lu3bqxHj16sBcvXJBtbW129+5ddu7cOdasWTN29uxZ3vEUQuznsEOHDmz16tWMMcbevHnDjI2NWa1atZi6u
jr73//+xzldyYn9/Pn6+rJ169YxxhjLy8tjrVq1YhKJhGlpabHTpO/zDUe+yuzZs7/4IXR0jRE2fX192fsDbW1
tFh8fzxhj7Ny5c8zY2JhnNIXo27cvW7FiBWOMsaysLGZZlZcVUVVWZioqK7PVpWaBCmEhJJJLPfigpKfGOV2Jie
bH/Odra2uzGjRtFxiMjI5m0jk7ZB1Igc3NzFhERwRhj7Pjx4OxfX58dO3aMjRgxgnXoOIFzOsVZsmQJ8/DwYG/
fvpWNpaWlsT59+rClS5eyzMxM1qNHD9axYOeOKb+drq4ue/DgAWOsoKhZmP/8+fOsVqlaPKOVivTOdLZhwwbWo
UMHpqyszKysrNicOXN4xyoRsZ9DLSOtlpiYyBhjzM/Pj/Xu3Zsxxtj169dZ1apVOSZTjMqVK7OoqCjGWMG5jIm
JYYwxdurUKdaoUSOeORSmIpzD27dvM8YY++eff5i9vT2TSqVs586drF69epzTlZzYz1+NGjVkN0ZPhl+NGjVkNOb27NnDatSow
e7fv89+//131qpVK87pCKFrjND169eP+fj4MMYK3hMmJCSwd+/eMRcxXF1EU+qpWrSqb/LBlyxZWp04dlpmZyf7
3v/+V6esY8az1IHLy8/M/+yGGZRNt27YFALx48QLnz5/H+fPn8eLFC86pFMfFxW+vr54/PixbOzRoOeYOHEi2
rdvzzzFZyT19+hQmJiYAgIMHD6Jfv37o2LEjfvnlly9y9epVzOsX566+/8Mcff8itf9fT08Ps2b0Oxc0FCaGpqYta
sWbh+/TrHlN+OMYb8/HwAwMmTJ+/+xn3nk8h2aMyb8/HwAwMmTJ+Hm5gYAMDExwcuXL3lGKxU6O
jrw8vLC8ePHERO0dDS0tLcHvSCT2c6impiZbN
nDy5EnZMhBDQOOkp6fzjKYQUqkUOjo6AAAjIyPZ3wkzMzPcv3+fZzSFEfs5zMrKkp3D48ePw8PDAOpKSmjRogW
Sk5M5pys5sZ+/V69eoVq1agCAw4cPo2/fvqhbty68vbl1x69YtzukU59q1a/jl1l8wYMAAeHh4yH2IxfXr1/Hvv
//i33//xYObN3jHURi6xgjbokWLEBERIbdzubm5OR49eoQ///yTd7wSe/v2rawlzNGjR9G7d29oamqia9euZdr
/jAphRJCysrLg7e2NGjVqoF27tmjXrh1q1KiBESNGLd9MqVK5Geng5zc3PUrl0btWvXhoWFBdLTO7FixQre8
UrEwMAAAqampAAoufq6urgAK3pyLoUhb6O3bt3j+/HmR8RcvXsj3aFHPnzsXmzZsRHh6Orl2
7AgASExNRtWpVzukU7/3799i5cyd69uyJxo0b4/Xr15g6dSrvWCUi9nPPYpk0bTJo0CX/88QeuXLki077Y2FhR7
KTcoEEDREVFAQCaN2+OhQsXIIiIAv7+/rCOt0ScTjHEfg7r1KmKKmDvDn35sGKIPbzV7V
qVdy9exdSqRRHjx5Fhw4dABS8Nv14JO8h275901q1aoV79+5hz549549yM3NxZ07dzdxAWFia347BQPX/+HC4uLvjhh
x8wfvx4jB8/HK4dr7FcWOdrjHCVqtWLURFReHX3/FxIkT4eDggAULFuDGjRswsnwbUoTtuGjRswNjbmHa/
mZuLoOaOyn883b96U7ZVZTb3jJS5M2fOsG7durHatSbrHatWuz2rVrM3d3d9EsKRw5ciSztLRkw8fZm/fvmVv375lh
w4dYrVr12Y//fQT73gkKkZ+fz44fP86WL1/Oli9fzk6cOME7kKKMGMGOmZmZMVdXV1a5cmX27tXX27t07xhhj27ZtYw4
ODpzTKc6gQYOYhYUFCwONZampqSw1NZWFhoYYsSOtLNmTIEMZYwTE3adKEc9JvExZaGhzkyJEsPDycyyyFEPs5TE5OZ127dmX29vayPj2vayPj6MMTZhwgQ2btw4jskU4+j
Ro2z37t2MMcbi4uKYtbU1k0gkzMjIiJO8eZJzOsUQ+znctWsXU1VVZUpKSnItAQICAljnzp05JlMMsZ8/Pz8/p
qenx+rVq8dMTU3Z+/fvGWOMrV+/nrVoOYJzOsWws7NjK1euZIz9X4+i/Px85uPjw2bNmsU5Xcn169ePNW3alN2
9e1c2dufOHdaOaVM2YMAAjskUg64xpDxbtWoVU1FRYfr6+qxhw4ZMKpUyhbvnx5mfb3p1OjRerff/+Fl5cXP
Dw8OLp1awBAREQE9uzZg6CgIAwaNIhzwpIxMjIhzwpIxMjJCSEgInJyc5MZPnz6Nfv36ieiJujljl5uZi2bJlSE1NxfDhw+H
g4AAAAWLJkCXROdPDjjz9yTqgYGRkZmDhxIoKDg5GX1wcAUFFRgaenJ5YsWQItLS3cvHkTANCoUSN+QRXk/fv3U
FZWhqqqKu8oCqGpYlu3bph8ODBcHNzE81xfYnYzmFF8vr1axgYGIh2K3kxevrOKZ48eYKGDRvKdqW9cuUKdHV
1y3bXLPJdQkJCkJqair59+pmoGzatAn6+vro0aMH53Qlp6WlhTt37sDc3ByVK1fGmTNnYGdnh3v37sHFxQVPn
jzhHbFE9PTOcPLkSfzwww9y41euXEHHjh2RlpbGJ5gCOTVGuPbv31/s+Mc7z1tYWJ2XvbxnCeWtO3rUvU/rdrbf
rawMADh06BH19fVntorRRIUykbGxsMHLkSEycOFFufPHixixfjnn39w7949TskUQ1NTE9evX4eNjY3c+J07d9CsW
TNkZmZySqY4p06dwlqTp/D8+XNZP59CGzZzZs4i4kWVFdc/00RXqsWH1gc2D6xbuVj10jh6ZvjRL5Bk/3Rqxws
y85mZmYlx48bR3wdCSInVqlULR44cgZ2dHezt7TF9+nQMHz4cU1raH27Tvbr1kNVzYGUFbP06dMCQ1RyNL3IGmgMCz3ydyyd/x
3BOdBRFnykiXEpKSpBIJPiOTFM4JpFIOKZNG+zduxcGBgawtTXhMtHDgQFy9eRO9f7uxcZGXTuj9c9+G3B8Tz7IDZwwO6+ITEzkkUqyqWLVv
Cz88P79+/1411Z2djzpw5aNmyJcdkijnFzhx07NgRp06XL/HmzRu5DyIc2cPvyAuK7lDt4uqALsE+nB729PRBxrOWPX/gyx
pCenv7ZDOJ42bRpE7Kzs4uMZ2dnIzg4mEMiQojYtGjTtoedaDbtcm0YtFZK22AIWpD

xAj/88ANOnDiBt2/f4u3btzhx4gSaN2+OgwcP4uzZs3j16hWmTJnCO6qgqfAOQEqHiYkJTpO6hTp16siNnzx5U
rZjn5AtW7YMnTp1Qq1atdCwYUMAQFRUFNTV1XHs2DHO6UouMDAQQUFBGDpOKO8opASuXbuGnTt3IiUlpUhT/ND
QUE6pyOcYGBjgyZMnMDY2hr6+frHLzArvxIlpYwciDOnp6WCMgTGGd+/eyTWUlUqlOHz4sCia6BJC+Fu5cqXsZ
vNvv/OGVVVVXLhwAb1798bvv//OOV3JrVy5Et27d4e5ubnsfVFqaioaNGiAf//9l3M6UtH5+vpi7dq1aNWqlWy
sffv2UFdXx8iRI3HnzhOsXboU3t7eHFMKHxXCRGry5MkYP348bt68KfslioiIQFBQEJYtW8Y5Xck1aNAAcXFx2
LJlC2JiYgAAAwcOxODBg6GhocE5Xcnl5OTIXfyI8Gzfvh3Dhg1DpO6dcPz4cXTs2BGxsbF49uwZevXqxTseKUZ
YWJhsO+fTpO9zTkOIvMLirEQiQd26dYs8LpFIMGfOHA7JCCFiU/i3EChYpjV9+nSOaRTPxMQEkZGROHnypOx9h
I2NjWwncOJ4io+PL3Z3T11dXVm7FSsrK7x8+bKso4kK9QgTsT179mDRokWyfmA2NjaYOnWqKJp4it2OadOgra2
NmTNn8o5CvpO9vT1GjRqFMWPGQEdHB1FRUbCwsMCoUaNQvXp1ObxhlUqluHXrFszMzETTpyAvLw8BAQHw9vYWx
TbcFdWDBw8QHx+Pdu3aQUNDQzabT6jCw8PBGIOLiwt2794t9OZVTUONZmZmqFGjBseEhBTYuHEj+vfvDO1NTd5
RyHdSVlaWzZD+2KtXr2BsbEyzokm5FRISgj59+vCOUSJt2rSBjo4OgoODUaVKFQDAixcvMGzYMGRmZuLs2bM4e
fIkxowZg/v373NO+/X8/f0xZcqUcvO3gQphRDA+t4NGcbp3716KSUqfr68vgoODZb2lPt3FbfHiXyZySKUZaWhp
CQkIQHx+PqVOnwtDQEJGRkahatSpq1qzJO55CiHXHpQkTJsDOzg4jRoyAVCqFo6MjLly4AE1NTRw8eLDITq5Cp
aOjg1u3bsHc3JxFIWTSqUICgr67GYcYWFhnJIpxqtXr9C/f3+EhYV
BIpEgLi4OlpaW8Pb2hoGBARYtWsQ7Yok
kJyfDxMREtguYGPXq1avYouXHO2YNGjQI1tbWHNIpxubNmxEYGIjExERcvHgRZmZmWLpOKSwsLAR/w7Jq1arIz
s5G3759MWLECNHNcP9cnOiJRIJKlSpBTU2tjBMpnpKSEp4+fVqkEPb48WPUrl272D6FQnPq1CksWbJEbsLAhAk
TRDMrLC4uDqdPny727/ysWbM4pSq5vLw8xMTEQE1NTW529L59+zBr1izExMgw4cPHBOW3P3799GjRw8kJibKL
d21tLTEvn37ULduXezduxfv3r0TVBudzxXYeaGlkSJ37do12QXe1tYWTZo04Zo+/Xs2fOrnieG/j3ROdGynWx
u377NN4yCRUdHw9XVFXp6ekhKSoKPjw8MDQ0RGhqKlJQU0TR7NjAwwLt37wAANWvWxO3bt2FnZ4e0tDRkZWVxT
vf9QkJCMGTIEADAgQMHkJiYiJiYGGzevBm//fYbIiIiOCdUDBcXF4SHh4uyEObr64ugoCB07doVDRoOEPQsqeJ
MnDgRKioqSElJkdtZuH///pg0aZLgC2FmZmIS0vDlStXin2DM2zYME7JFEdPTw979+6Fvr6+7HVLZGQk0tLS0
LFjR+zYsQN//vknTp06Wbbr
CvS6tWrMWvWLEyYMAHz5s2TvWbR19VBV8Ie/ToEQ4cOICgoCA40TnBOtI
SXl5e8PTORLVq1XjHK7HP9ZAsVKtWLQwfPhx+fn6CK1gvX74cQMFr6XXr1slt8iOVSnH27FnUq1ePVzyF+d///
gdfX1/O6dMHvr6+AIBLly7BzcONS5YswZgxYzgnLJl//vkHoOePhpGREapVqyb38yRSARbCLt9+za6deuG1NR
UAECPHj2wevVq9OvXD7dv34aPjw8OHTrEOWXJWVtb4+7duzh+/DhiY2NlYx06zdXqZ+fXF5Uu7mXzEiSqmpq
axNmzZMIpEwAwMDZmBgwCQSCWvdujVLTU3lHY9UY03bt2dTp05ljDGmra3N4uPjGWOMRUREMzM47JFGvgwIF
s0aJFjDHG/P39WZUqVdiPP/7IzMWK9evTin+36VKlWSXUN8fHyYr68vY4yhIQEpq0jwzGZYq1evZpVq1aNT
Z48mW3dupXt27dP7kPIKleuzA4d0Osq7RqmpWrUqu3nzJmNM/hoTHx/PtLS0eEZTiP379zMdHR0mkUiYnp4e09f
Xl3OYGBjwjqcQ06ZNY6NHj2ZSqVQ2JpVK2bi2dnvv34aPjw8OHTrEOWXJWVtb4+7duzh+/DhiY2NlYx06zdXqZ+fXF5Uu7mXzEiSqmpq

siNr1+/XtbY+tWrV4LeSTk7Oxtjx47Fpk2bABTcrLS0tMS4ceNQs2ZNQf69L+x/BhTOTZ43bx7OnDmDli1bAih
4HxEREYHJkyfziqgwderUwcyZM3HpOiXY2dkV+Tsv1EKKRCLBu3fvoK6uLtsJOjs7u8gNdl1dXU4JFWPixIlwd
3dHYGAg9PTOcOnSJaiqqmLIkCGy971CNWDAgHLTLJ92jRSpffv2ISAgAKtWrULTpk0BFDTOHzduHKZNmybIBns
fq1WrFqZPn46xY8fKja9atQoBAQF49OgRp2SKs3DhQixfvhznz5/Hjh07EBAQgMOHDwuyMfDH3r59iz59+uDat
Wt49+4datSogadPn6Jly5Y4fPgwtLS0eEdUiEuXLmHQoEFITk4u0hxSDBs6AMD79++hrq7OO4ZCMcbg7u6Ow4c
Po2HDhqhXrx4YY7h37x5u3bqF7t27Y+/evbxjfjMvL6+vfu7GjRtLMUnZePv2LVauXImoqChkZGSgcePGDNmD
KpXr847WomtX78e/v7+8PLyKvYNjtB3TQYKbtotWLAAK1euxLNnzwAU7ERY+BpGWVkZKSkpUFJSQq1atTin/T5
btmzB7NmzER8fDwCoUaMG5syZgxEjRnBOVnIjRozAjz/+KCswFIcxhpSUlCKz4oRg//796nu3L+rVqycr1F67d
g0xMTEICQlBt27dsHr1asTFxQl2l29XX19ERERg6dKjo2U71s2ePRs3btzgHfGbWVhYfNbXVhIhI
SSjlN6frSsQr5+JSUlORuRhYWwz79WuivsfX19XH58mVYW1tDX18fFy9ehI2NDS5fvgxPT0/ExMTwjvhdytuuk
VQIExEDAwO5iOFmZiby8vJkU5kLP9fSOsLr1695xVQIbW1t3Lx5E3Xq1JEbj4uLg40DAzIyMjglU6xp06Zh/fr
1kEql0OHLkiKim/OZERMi9SRXDktaPNWrUCHXr1sWcOXNkywg/Vl6mBX8rqVSKgIAABYG4tmzZ7K7xDNnzoS5u
bng38Rt3LgRvr6+2LdvX5FldGFhYejZsydWrlwpip35iDB9aRc6MbwB+FThnX6h3+H/nKysLGRkZJSbNwaKEBw
cjP79+8tmDBfKycnB9u3bRXH9Tv7ExMxJo1a+R2dBs1apRodho2MzPDjh070KJFC+jo6CAqKgqW1pZ48OABGjduX
GyLCOJKW3h4+Fc9z9HRsZSTlK4qVargwoULsLKyQt26tQJMTExaNKkCTIzM3lH/C5KSkp4+vRpufl
7R4UwESmcvvw1PDO9SzFJ6Rs0aBAcHByKrO3/+++/ce3aNWzfvp1Tsu/38ZTtj/39999o164dmjVrJhsT6pTm3
NxcaGho4ObNm2jQoAHvOKVKS0sLUVFRRYq1Qufv749NmzbB398fPj4+uH37NiwtLbF8/X+h9FqDp27Ag
XF5fPLvsICAhAeHg4jhO7VsbJFCcxMRF5eXmwsrKSG4+Li4Oqqqoo3silpaXhypUrxS5LFsObcCJsYv8d/Nxd/
1evXsHY2Fh0xVoxTU1P19/7gQFhUVhXbt2uHt27e8I5KvINYeb2LXsWNHDB8+HIMGDYKPjw+io6Mxfvx4bN6
8GW/evMHly5d5RxQF+oQ0QEaEXt77Ff63t/7ioJJSiOZIlS4odV1ZWRkREhKy3jUQiEcwxfUpVVRWmpqYV4kVw8
+bN8eDBA9EVwoKDg7F27Vq0b99erglw4YNBTtV+2PROdFYuHDhZx/v0qXLZ4vWQjF8+HB4e3sXeRN++fJlrFu
3DmfOnOETTTTEEOHDiAwYMHIyMjA7q6unKzMSUSCRXCB0DZs2eYMmUKTp06hefPnxdZXCg3JwuR2vdxdEHaVv
6jrEGFY38rIu37otLXoXk4cOH2L9/P1JSUpCfvTkyrCuHHjRNXjDcB
Xz0QU+gzigIAAvHv3DgAwb948DBs2DKNHj4aVlRU2bNjA0d33+7Rf7eeUVa9amhFWAbx//77IBV7oF4iKtM5fb
NavX4/Q0FBs3rxZrnmu2OzZswe///23ebeNaaBuqpUZmZqGAtiKlhTYTTBVDQyp2v9ntsWnKyrZJlX2Vzp4jJq11
cnPzZXlKPHz+GhYUFPnz4UMbJFEdXVxeRkZFFirQPHjxA06ZNkZaWxieYgtStWxdubm4ICAgVnK6GpaTXTkYQGm
qxZZZSk9HTp0gUpKSkY03ZssUvLe/TowSmZYoj1d9DBwQESiQRRUVGoX7++AwUqVSKxMREd07cWbZZjlD9V7F
d601HA0D8+fPo0qULhgwZgqCgIIwaNQp3797FhQsXEB4eXmRc6E5deoUunfvDktLS8TExBkBBgwZISkoCYwyNG
zdGWFgY74glIsYeb0DRHmGfEkuPMLH62n61ZdWrlmaEiVRmZiamTZ34tWrV0UeF/oFIF/0d1V1ZWRkREhKy3jUQiEcwxfUpVVRWmpqYV4kVw8
JBw8eoEaNGjAzMyvSHD8yMpQ3AHT29tbNiaRSAT/R9rW1hbnzp0OrOuA4JCQEDg40nFIpjlQq/eLyAWVlZ
eTl5ZVhIsUr3HXpU2/fvhXsz+XHj16hPHjx4uyCq/lpubi8TERKioqKB27dqIjIzE2D2IFGz59ZKBDZW7Vl9vLqR3eJpIwtNMRV8QAVOs...

eXmjx4sWySQFz5sxBRkYGduzYASsrK8HvGAkU9Hi7efMmFixYIMoebx8T8lLPz0lISKDiXhmgGWEitWTJEigrK
2P8+PE4efIk3N3dwRhDbm4uFi9eDF9fX94RS8TCwgJHjhxBvXr1eEcpFYsWLUJCQgJWrlwpugt8//79kZCQgOD
gYNjY2AAomDXl6emJ0nXqyHo2CJ27uzuUlZWxbt06WFhY4MqVK3j16hUmT56Mv//+W7YcRgj+6+7+x4R+p7+iO
HfuHNasWYOEhATs2rULNWvWxObNm2FhYSHoHkUVwadLlZSUlFClShW4uLhgxowZsmIEKV+io6PRoEEDKCkpITo
6+ovPFequwgCQl5eHrVu3ol0nTqhatSrv0OQ7fWlJpJh3ZM/Ly8Pz589Ro0YN3lFIMTw8POS+PnDgAFxcXIoUa
oXeIkdJSQm0jo4YMWIE+vTpI9p+i7xRIayCSE5OxvXr12FkZIR///0Xa9eu5R2pRDZt2oSjR49iw4YN0NDQ4B1
H4Xr16oXTp0/D0NAQ9evXLzLtXsgXeD09PZw8eRI//PCD3PiVK1fQsWNHwW4b/ykjIyOEhYXB3t4eenp6uHLlC
qytrREWFobJkycLauvqTZs2ffVzPT09SzEJUYTdu3dj6NChGDx4MDZv3oy7d+/C0tISK1euxOHDhwXZQ3L58uU
YOXIk1NXVi8yY+pTQZzKIVXp6OnR1dWWff0nh84RESUkJT58+hbGxMZSUlORmCn9MDD0KNDU1ce/evSK7CxNS3
kVFRaFx48aC/x0s9Pz582J7KQu12F4R+rgCBTvvbty4Edu2bUNOTg769++PESNGoFmzZryjiQoVwioYsVzgs70
z0atXLORERMDc3LxIoSgyMpJTMsX4rwu9kC/w0jo6OHfuHBo1aiQ3fuPGDTg6Ov7nGyChMDAwQGRkJCwsLFC7d
m2sW7c0zs70iI+Ph52dHbKysnhHJBWUg4MDJk6ciGHDhsn117hx4wa6d0mCp0+f8o74zSwsLHDt2jVUrly5Qs1
kePjwIQCgVq1anJ0UnLKyMp48eSJXKPoUY0ywhaLk5GSYmppCIpEg0Tn5i88VegHJyckJEyZMENUu0VRsrxjE8
j7p+vXr8PT0xL1794oU3IV6Da2I8vLysH//fgQFBeHo0a0oW7cuvL29MXToUNG0kuGJCmEVjFgu8P369cPp06f
Rp0+fYpvlF27hTcqfHj16IC0tDdu2bZNNPX/06BEGDx4MAwMD7Nmzh3NCxWjbti0mT56Mnj17YtCgQXjz5g1+/
/13rF27FtevX8ft27d5Ryyx9+/fF9kEQIgzNSoaTU1N3L17F+bm5nKFsISEBNja2uL9+/e8I5IvyM/Px9y5c7F
o0SJZjxsdHR1MnjwZv/322xc3CyjPwsPDZ0bp1a6ioqCA8PPyLz3V0dCyjVKXj/fv3ol7qsnPnTsyYMQMTJ04st
r+UEGejVLRie+/evdGsWTNMmzZNbnzhwoW4evUqdu3axSlZ6RLL+6SGDRuidu3amDHvk4RebK9oPnz4gP/
973+YMWMGcnJyoKamhNhnzpwW4evUqdu3axSlZ6RLL+6SGDRuidu3amDZtWrHvk4RebK9oPnz4gP/
973+YMWMGcnJyoKamhn79+uHPP/9E9erVeccTLCqEVTBiucBraWnh2LFjouttIC
EimOKmpqejevTvu3LkDExMT2ViDBg2wf/9+UcxsAIBjx44hMzMTHh4eiIuLg7u7O2JjY1G5cmVs374d7du35x3
xu2RmZmLatGnYuXMnXr16VeRzIQ9NfpEoVXQ3lOr7hB8xQIagvBgKJhoiJuBhDQg4UOkwdOfm6K44Yjv/Olc5x
7cGULBD2OzZs+Hj44N58+ZxTkj+i66uLnr16oUhQxoTFH4r6t8CnvDQ1hMYvqmLFqG0JZ+L7e8zM4HfhgGRmd5wzl
q6opnz55xSla6xPI+SUdHBzdu3BDtxlsVxbVr17BhwwZs374d2dnZ6NWrF9q2bYtZs2Zh6tSpiI+P5x2xRESxZv
kW7RhJBMjExEeXMk8/ttJSWlobr16+jW7duCAkJgbu7e9kGUyATExNERkbi5MmTiImLgAQDY2NjA1dWVcLF6tS
pk+xzKysrxMTPXr19/UeL48+uWXX3D69GmsXr0aQ4cwOxapVq/Do0STSWbMGCxYs4B2PfAUfHx/4+vpiw4YN
EgkePz4MS5evIgpU6Zg5sxyvZ0OVmNhnMmzatAnr1q1D9+7dZWP29vaoWbMmxo4dK+qCgdhZWFjgypUrWLt2
TBSQ8+w0HhYnw03D+2Sh/h5pYmZDNwsQT9NyQOtF1NVmZgN0V+eg8ICH1aCmcvw66Q4HQ3PjnYOQDFWW5c9o
q6qiroFhb/tVHF/fv3yyhJ6Wrfvj2ioaKoECZQixcvxsaNG3H//n10794d6667dvz48WjSpAn+97/XkXD
YGo3pn8Rst7kBW9GVSvhuRtBkMrWcwxmRkj1bbbYcVNsZBvjNYXy0bw7ed8vG3Wa0a7iq7tNJg1TqgXFRpu42e/
gAkczwkTm0900PpWWlobw8HDB3+k4d0gQVqxYgcDAwCFCQkJw4cIF3lHIZ3h7e3/x8/Xr15dSktKyhdYw4VHqsZTvTbiUfGhzPhBm3fMJ4Mjz6cS+
akpgoOD4eTkBF1dXURGRqJ0nTrYvHkzm3bJshG6xUNYwwBAQGYP3++rFddpUqVMGXKFPzxxx+c05Wc2GcyqKu
rIzo6GnXr1pUbv3//Pho1aiT4s3ANDQ3ExMTA3Nxc7uiGu3fv4scffxT8EbETJ0587ecTJ06U+Jan4pK5ym1AdyzIkZHM1pM1QtfFxdXeZObG7l
TFkyBDMmjWLdzTyBTVr1sT58+eLLuLHcvXs3hg0bhszMTE7JvtT58+eLLJHcvXs3hg0bhszMTE7JFKdZs2bolq1bkZ/F2bNn48CBA7h+/TqnZ
ViGXW4suXL+Hp6YlmzZqhQYMGRXopV3wjhZQYMGRXopf3wjhZQ/VlZW8Pb2xvDhwz+79DEnJwwfbtm2jTapKgAphIlNRdtMwMDBAV
lYW8vLyoKmpWeQCL5YXyZ+KjY1FFixYtBHl8YWFhGDt2LC5dulkNt/bt2/RqlUrBAYGom3btpwSKoaSkhLMzMz
g4OBQ7AutQKLthaatrY27d+/C1NQUmhoKJo1a4bExETY2dnJehaR8i8i8nJwcPHjxARkYGbG1toa2tzTuSQ
mhoa0DmzZuwtraWG4+JiGiYGDg4PgCOXNmzdH8+bNizPLy5xSqY4devWhZubGwICAqCpqck7Tpm
4e/cuBg8ej0joaMG/CQ8ODv7i40Kf0efn54d//vmnSDM5mDyt09d8Pb2Rl7QEPr27cs5YckBBEFrk9K3chjgzuyQunoz5dorQUrIzU
DRoEFxcxXAAAp06dwrZt27Br1y7y7Bznr7r40qCgm9h9aBAwcw0jQYmfviaHQJ3ZJSUkwNTUtssycMYbU1FSYmpp
ySiYutDRSZIRe4Ppan1tCKHYfPnwodqq6ECxduhQ+Pj7FLmnV09PDqFGjsHXrYsXwkaPHo1t27YhMTERXl5eG
DJkCAwNDXnHUhhLS0skJibC2Hny+0u8YNr5S7BxnYOxkcSOHAA+vr6vO0Rr+Dt7Y1ly5ZBR0cHtra2svHMMz
OsLMVC9nZ2WHHjh1FZjJs375diFJzJs375d7niFauHchejatStOnjyJli1bAgAuXryI1NRUOczIfPToEcaPHy/6Itj79++nf
/9+bN26FUePHkXVq1UxdepU3rFKzNfXV+7Qv1UNDZqamoIvhM2ZMwewvX7+Gq6sszp49i49G+t2Pdj8zNkx2xtzHiaPnJ5+kb/uklDEh1LtcaNG7cg4ChQaqZQF5syZqFa1Ku845BvvWJg
l1btovyx16/fg0LCwsqZCoIzQgjREAmTJiAmJgYHD16lHeUb2ZmoajR4/CXsam2MdJYmLQsWNHpKSklHeVyxfv
w4QNCQ0OxYcMGXLhwwAV27dsWIESPQsWNHQfcHA4AlS5ZZWVkZ48ePx8ePX8mTJ+Hu7g7FF48eIib4BI+aOsr
FzsC6yXLl+iWrVqyMvL45RMcQ6k+Fjjx49wv/+9z5+5Pos///yzbCdeofPw8MC8MCAAQPQr18/3FKxbFjx7B161b
s3bsXKioq6NOnDwYPHox27drdrxjlZq4uLiMHr0aEyd0lWuh6aQDR48GFevXsWvXsjR49kPd8IKQ90dHRw8+ZN1K5dm
3cU8h2UlJTw9nTIq/TkpOTYWtrK4rl+UVRy0ArWlVnHe+7TJo
0qdjxt2/fIjIyErIyErGxsTh79nnz1WlSxsTl1dHbdv3/5s484HDx7Azs5O8MBwcjLy8PCPd+7cE

```
c0SNKBg+nZhnzB7e3veccgXpKengzEGAwMDxMXFoUqVKrLHpFIpDhw4gOnTp+Px48ccUyrGoUOHEBAQgJs3b8p
mMvj5+Ql6JkNFsn79evj7+8PLywt2dnai62+jqakJd3d3DBoOCG5ubkWOT6yuXbuGIUOGyAq4QrJ///4iY7m5u
Zg4cSI6duwo9zMp9J/PQmlpaQgJCUFCQgKmTJkCQONDREZGomrVqqhZsybveOQLPD090bZtW/z444+8o5BvUPg
+cNmyZfDx8ZGbFS2VSnH58mUoKysjIiKCVORRoaWRRJDCw8PRpUsXtG7dGmfPnsW8efNgbGyMqKgorF+/HiEhI
bwjfpcbN24UO66rq4sOHTogNDSOSHNWoahZs+YXC2HR0dGfbQgpZB83ZhXjVGZzc/MKtWGFkOnr60MikUAikRR
ptA4U9A2ZM2cOh2SK17VrV3Tt2pV3jFKxceNGaGtrF+lDtGvXLmRlZYmica6Pjw8AwN/fv8hjQu9vk5eXhz///
BN9+/aV9ZeqKFRUVARbaP/STNINGzbIlpQL/eezUHR0NFxdXaGnp4ekpCT8+OOPMDQ0RGhoKFJSUv6zDxzhq27
dupgxYwb0nz9f7M2E8ePHc0pGvqTwfSBjDLdu3ZJrh6OmpoaVqvOKJDs0II4LUsmVL9O3bF5MmTYK0j
g6ioqJgaWmJK1euwMPDAw8fPuQdkXxi3LhxOHPmDK5evQp1dXW5x7Kzs9GsWTM4OzsXaQAtRB8vjTx//GsWTM4OzsXaQAtRB8vjTx//jy6des
GLy8vdO7cuUjjS6G4ePEiXr16hW7dusnGgoOD4efnh8zMTPTs2RMrVqxApUqVOKYkXxIeHg7GGFxcXLB792653
nVqamowMzMTzdI6Matbty7WrFkDZ2dnufHw8HCMHDkS9+/f55SMfC1NTU3cu3dPtP2KPp09xRjDkydPsHLlSpi
YmODIkSOckpGv5erqisaNG2PhwoVyr7MvXLiAQYMGISkpiXdE8gVfumkukUiQkJBQhmnIt/Ly8sKyZcuK7atMF
IcKYUSQtLW1cevWLVhYWMj9gU5KSkK9evXw/v173hHJJ549e4bGjRtDWVkZY8eOle3oFhMTg1WrVkEqlcqm3Av
Zzz//jO3bt8PExATe3t4YPHgwjIyMeMcqsS5dusDJyQnTpk0DANy6dQuNGzfG8OHDYWNjg7/++gujRo3C7Nmz+
QYl/yk5ORmmpqaC71f3MUNDQ8TGxsLIyAgGBgZfPDYh7rr7MXV1dcTExBSZiZmUlAQbGxvRLS8XIycnJ0yYMEE
U/eqK8+kNH4lEgipVqsDFxQWLFi0S5exvdHT00NkZCRq164t9zo70TkZ1tbW9Dq7HGOMISUlBcbGxtDQ0Adh
5Byi5ZGEkHS19fHkydPitzxuHHjBvUtKKeqVq2KCxcuYPTo0ZgxYwYKa/ASiQSdOnXCqlWrBF8EA4DAwECYmpr
C0tIS4eHhCA8PL/Z5oaGhZZysZG7evIk//vhD9vX27dvRvvHlz/PPPPPwAAExMT+Pn5USFMAMQ4C2XJkiXQ0dEBI
P5dhY2NjREdHV2kEBYVFYXKlSvzCVUKMjMzER4ejpSUF0Tk5Mg9JvRlPT///DMmT56Mhw8fokmTJtDS0pJ7X0j
9FvPz83lHKHXh4eH4+++/ce/ePQCAra0tpk6dKvidrwtVqlQJ6enpRcZjY2Pl+ksKiYODw1ffAIqMjCzlNKWHM
QYrKyvcuXMHVlZWvOOQr+Th4fHVzxXae4jyigphRJAGDBiAad0mYdeuXZBIJMjPzOdERASmTJki+G25xczMy
HDx/Gmzdv8ODBA9kfawMDA97RFGbYsGGimmlT6M2bN3KN3g8ohGiFxfLDHOyPqSgQMHYvz48
dDR0ZHtMhgeHg5fX18MGDCAczrzFuuHHjBtzc3JCVlYXMzEwYGhri5cuX0NTUhLGxseAU2L
277//wsvLCx4eHrJzGBERgfbt2yMoKAiDBg3inLDkunfvDn9/f+zcuRNAwc9nSkoKpk2bhktkn69e3N0933E0gPzU
0pKSrCyssKrV6+oECYgenp6vCNU0LQ0kghSTk40xowZg6CgIEilUqioqEAqlWLQoEEEqlCgqCsrIy74iEiIqZmRk
2b96Mdu3aIScnB/r6+jhw4ADat28PoGCppK0jo+CXnRFhKm7mwucIvedGTk40hg4dil27dkFFpeB+Zn5+PoYNG
4bAwEC55rpC5eTkhLp16yIwMBB6enqIioqCqqohgwZAl9f32+6c14eJScnf/Fxoc/a7N27N5o1ayZbSl9o4cK
FuHr1Knbt2sUpmWLY2Nhg5MiRmDhxotz44sXLaG3+rDjRoIB3HELKJSqEEUFLSUnB7du3kZGRgDdatk7Y27v3r1DjRol80TJE7Rs2RJHj
hwpMouRlC8HDhzAwoULsXr1aj0IB3HELKJSqEEUFLSUnB7du3kZGRAQcHB7rzQUgpGTl6NKKiovDnn39i796
92LRpEx4/fix707l1yxYsXboUV69e5ZyUVESFu7N+DbHMtomLi8Pv/LPy/D2toa+
vr6uHjxImxsbHD58mV4enoiJiaGd0TyBVWqVEFYWBjs7Ozkm/dugVXV1c8e/aMUzLFqFSpEu7cuVNkF++wHDx6
gQYMGouqfdf78eRHRyMjIwNNmjSR3fwi5ZuBgQQGysrKQL5cHNTW1Ir3Hn+xTbU+NeccgRPT++
0MPeHh4wNHREdra2ti0aZPcSiuz06d0yz5OSkjB9+nQMHz+nQMHz4cLVu2BFCw6+mmTZswf/58XhE
VzsrKSrQ3f1RVVWUN142NjZGSkgIbGxvo6emJokcAKcDMKD8BBjyQsYlIbQ0kgicqakpgvEbVqJEzcEcP1pppTE1NeccgRPT++
fvz4cHBw4Jyq5qlWrIjExESYmJsjJyUFkZKTcjOJ3795BVVWVY8LvZ2BggCdPnsDY2Bj6+vrFtgtgjEEikQi+U
PTLL7/g9OnTWL16NYYOHYpVq1bh0aNHWLNmDRYsWMA7HvkP1apVw+vXr2FmZgZTU1NcunQJDRs2RGJiImgOk+J
QIYwIOuTJk7Fs2TKsXLnyq/veEEJIRVWRmgS7ubkhNjYWq1evlu0u6O7ujp9++knwjawBYObMmfDw8EB8fLzcs
r0tW7ciJCSEczrFSE5ORuf0nZGSkoIPHz6gQ4cOONHRwZ9//okPHz4gMDCQd8QSmTRpEsaMGYP379+DMYYrV65
g27ZtmD9/PtatW8c7nkKkpaUhJCQECQkJmDJ1CgwNDREZGYmqVauiZs2avOOVyPPnzzzFgwACcOXMG+vr6AAq01
9nZGdu3b0eVKlX4BiwBNzc3TJ8+XbZDtKamJtq2bSt7PDo6GrVr1+aY8PuFhYXB0NBQ9rmY3z8cOHAAwcHBcHJ
ygpeXF9q2bYs6derAzMwMW7ZsweDBg3lHJF/g4uKC/fv3w8HBAV5eXpg4cSJCQkJw7do1eHh48I4nGrQ0kghSr
169cPr0aRgaGqJ+/fpF7k6FhoZySkYIIeXPpk2bvvq51GS3/Pt02VnDhg3h5+cHQ0NDUTSTL+ydtX79elSuXFm
2dPDMmTPw8fFBXFwc74gltmXLFsyePRvx8fEAgBo1amD0nDkYMWIE52Q1Fx0dDVdXV+jp6SEpKQn379+HpaUlf
v/9d6SkpCA40Jh3xBLp378/EhISEBwcLJvBd/fuXXh6eqJOnTrYtm0b54Tf7+XLl/Dw8MD58+dlOOT36tVL9nj
79u3RokULzJs3j2PKksvNzf3szLaXL1/CyMiojBMplra2Nu7evQtTU1PUqlULoaGhaNasGRITE2FnZ4eMjAzeE
ckX5OfnIz8/HyoqBXOWtm/fLuuTOWrUqGJ35SXfjgphRJC8vLy++PjGjRvLKAkhhJDy5ty5c1izZg0SEhKwa9c
u1KxZE5s3b4aFhQXatGnDO55CpaenY9u2bVi/fj2uX78u+CU9AFC5cmVcuHAB1tbWcj2OkpKSYGtrK/gNDz6Wl
ZWFjIwMGBsb846ia846iMK6urmjcuDEWL1wod/4uXLiAQYMGISkpiXfEEtHTO8PJkyfxww8/yI1fuXIFHTt2RFpaGp9
gCvS5HaJfv34NbW1927NOJCQorMCnv27Bnat2+P27dvcOqmGPb29lixYgUcHR3h6uqKRo0a4e+//8by
cuxcOFCPHz4kHdEQrijpZFEkKjQRQghX68iNQnevXs3hg4disGDByMyMhIfPnwAUDPDGLiAgAIcPH+acUDHOnj2
L9evXY/fu3Y8PDywatUq3rEUIj8/hkWJlZ2eDMQZNTU1oamrixsXWLpOKWxtbdGxY
Ofe8Urs6tWrWLNmTZHxmjVr4unTpxwSKVZ+fn6xs4lUVVWRn5/PIZHi6enpFTteuLRQ6FJSUvDjjz9i/fr1srE
nT57AxcUF9evX55hMMby8vBAVFQVHR0dMnz4JXJzc7F48WLe8chXKLyhFx8fj5CQEFHfOONFiXcAQ
r6FgYEBDA0Ni3xYWFigU6dO0HHiBO+IhBBS7hgYGOD58+cAAH19fRgYGYBT5KBwXurlz5yIwMBD//POP3JvV1q1
bIzIykm0Oyknv69CkWLFgAKysr9O3bF7q6uvjw4QP27t2LBQsWFJmhIlQdO3bE0qVLZZ+lJCSkoGGlBw+eG4
+jRQ7Y8MC0tDc2aNc0iRYvQo0cPrF69mnO6kqtUqVKxBffY2FhB988q50LiAl9fXzx+/Fg29ujRI0OycOBHt27f
nmIx8rcOHD+PChQuYNGkSAODx48dwcnKCnZ0ddu7cyTldyU2cOBHjx48HUDBD8969e9i6dStu3LgBX19fzunIf
9m9ezc6deoEDQ0N3Lhxo8gNPaIYtDSSCMpff/2l0OS5bqLCLTY0H0L3G0y0Q1JQzdyN8MSgVa2FaavFaVZHI2dIWA
zJWBhu9B3xVh+vSIgNtfX+BOnQBwmzsJIntC8bA0EEXA8+zvnK7vvq2ZWvjr57wbTVoPyCFe8wlGmgbRNLeJWlE
3bq3+UyBv/3oabhgxLNfGwiR0Vhe+WnE7ybqXgOvTVmbsnhMU4KWZmgJfYKTwJXzcBnGw+LaEwQ2PMmpzVLrsxQXM
nVsqyDfHNNWLnRQ0ShlQsqQ4GmDkXZoQKAUpj4V7ipnUoYd0JoXcLq1Pc+8bFDzUzPqY1CPOxJ8PXMAwNj3zXfB
W4jpyRWSOC+O2J7SvpJ6TpDzwMG4J7IIRUyP+qtoFhlb7mbu4VxdwR7s3xIv/Rk/KTwZd7F7yjgmXpZ6+6+YmxyI
uLixc2BJ06PL0YIOgXTEVG0rSDkkS8T1QFxKtqfQprKzzlbU1r68W4GA1NZdhhVgz/fZspWk/R35SYNvzzp4XrlNP
8jq2PNYcQKEc0+kBt8zjS4PGbM7ni6CGjWlaGqcXzGRmvrEcWVSLsO0WpbR1HFY2ZdzBq1aV8YZV6Tz9lFc1kRQ0
z9CuYY1+/Zdd2zAoF8j/uaqqjW5DHuGJ5yYJhvHzfo0FDZK0XE6bWk3V7m1SLAiCNqf4ZvAkWAzKhZk7B5D8lIvfSGvkoHrJpl
dnHF8Y3fNgSiqgwKe0z5v1ZXOhULAThRKzM5jr8AYwjfBqGCi5e3tfbbu+r8rczJuUZMJ9ZaG2cAvGrYhnyfvkrPpkx
FS4j0z9Ho2Et2cHnr/bRxwtL+Z2nD2/Dw2IdLJ3JxELJ8ys2ypqgaYHZ8VdGXMpDxYGi93JJDRWgypYx/Ys7ztyLa
tqPRfOaWtsLzFdnMvytMrsvKvqMBbQGBJ5hy8+g4Y+XgiD5ihq3LShXvyjFRaUHBsDqN7v1sDctGyPrYUGcs2zxzLxL
GZL9/Mzr5Szaj/zgYThzIHaM4L9AWYZ/8AAe2dc9vZZoS5GGcx8v4OmgOA4l3ZWWa6CbV9a1J8KAM63HPFydR+0f8zx
Qs8GyOqJ3jp7CUjg7WHnnLUJL0W3v2uKhBl5HTxmGYCYw4wdCvyJmzGJdcn0hm3BvLK4UkhYW/v7EJ3k9rpaSp7Hk8Mr
DA6VkpwfZB3PrNRY/rmvw/nhfHi7xf0JHgx6kfUxjPrqx0U4vyJMOy5evaXjI5LBWTGDFuDHM4rx1ls0dvxxJx6Fv
bHeuj8FA5GSc2pt3JAZLEN82bbqUBchmc/CnbC42Lx6t/uhdkJXCbpYOY8GOZwPAZ6YX4Dz3C6dYHhRPrKgDSNW+pUp
```

YIrVaokNxNMqE2ClZSUYGZmBgcHB3zppQ0tWyq/vqYH0+rVq4vdTYvwp66ujri4OJiYmAAA2rRpgy5duuC3334
DACQlJcHOzg7v3r3jGfO71a5dG4sWLULPnj2LfTw0NBRTpkxBQkJC2QYj30xPTw+RkZGoXbs2/vzzT4SFheHYs
WOIiIjAgAEDkJqayjviN/vaiQASiYR+RsupDx8+YPbs2Thx4gTU1NTwyy+/oGfPnti4cSN+++O3KCsrY+zYsbJ
Zt6RkaNdIQgghROSEeHf7W4OePRrbtm1DYmIivLy8MGTIEBgaGvKORb7BOqVL4ePjU+zOiXp6ehg1ahSWLFlCh
bByqmrVqkhMTISJiQlycnIQGRmJOXPmyB5/9+4dVFVVOSYsGTc3N8ycOROdO3eGurq63GPZ2dnw8/OT649Gyi/
GGPLz8wEAJO+elJO3ExMTvHz5kme075aYmMg7AimhWbNmYc2aNXB1dcWFCxfQt29feHl54dKlS1i8eDH69u1LL
SwUiGaEEUIIIUQUPnz4gNDQUGzYsAEXLlxA165dMWLECHTs2FHwSz8rAurBJGyjR49GVFQU/vzzT+zduxebNm3
C48ePZf2WtmzZgqVLl+Lq1auck36fZ8+eoXHjxrJZGdbW1gAKfi5XrVoFqVSKyMhIVK1alXNS8l9cXFxgYmICV
1dXjBgxAnfv3kWdOnUQHh4OTO9PJCUl8Y74Xdq1a4cePXqge/fusLKy4h2HfCNLSOssXboU3bt3x+3bt2Fvb4/
hw4dj/fr19BqmFIhjayhCCCGEfJXIyEjcunVL9vW+ffvQs2dP/Prrr8jJyeGYrOQqVaqEgQMH4sSJE7h79y7q1
6+Pn3/+Gebm5rRLlgA8e/bsizOGVFRU8OLFizJMVHo2b96M1q1boOaNGkhOTgZQMCNu3759nJN9vz/++AMqKip
wdHTEP//8g3/++Ueu6fiGDRvQsWNHjglLpmrVqrhw4QIaNGiAGTNmoFevXujVqxd+/fVXNGjQAOfPn6cimEAsX
boUkZGRGDt2LH777TdZ37eQkBCOatWKc7rvN2LECFy4cAGNGzeGjYONpk2bhoiIiC+2CyDlx8OHD2UbvjVoOAC
VKlXCxIkTqQhWSmhpJCGEEFKBjBo1CtOnT4ednROSEhLQv39/eHh4YNeuXcjKyhLNMkolJSVIJBIwxiCVSnnHI
V+hZs2aX2xGHh0djerVq5dxKsVbvXo1Zs2ahQkTJmDevHmyn099fX0sXboUPXr04Jzw+xgZGeHs2bN4+/YttLW
1iyzh2bVrF7S1tTmlUwwzMzMcPnwYb968wYMHD8AYg5WVFQwMDHhHI9/A3t5e7oZQob/++kvQS888PT3h6emJD
x8+4NSpU9i3bx/69u0LqVSKrl27onv37ujUqRPt0lhOSaVSuZsHKioqgr9mlme0NJIQQQgipQMTYJLjQx0sjz58
/j27dusHLywudO3eGkhJNgi/vxo0bhzNnzuDq1avF9mBq1qwqZnJ2d5XZHEyJbW1sEBASgZ8+e0NHRQVRUFCwtL
XH79m04OTkJtkcRIUL0888/w9/fHOZGRryjlJrLly9j//792L9/P+Lj4+Hi4oIZM2agdevWvKORjygpKaFLly6
oVKkSAODAgQNwcXGBlpaW3PNCQ0N5xBMdKoQRQgghFYiuri6uX78OKysrdOjQAd26dYOvry9SUlJgbW2N7Oxs3
hG/y88//4zt27fDxMQE3t7eGDx4sKjf2IhRRenBpKGhgZiYGJiZmckVwuLi4mBvby/Y30FChEhXVxc3b96EpaU
l7yhlIj4+Hvv374eJiQn69OnDOw75iJeX11c9b+PGjaWcpGKgpZGEEEJIBdK0aVPMnTsXrq6uCA8Px+rVqwEU7
Dgl5AJDYGAgTE1NYWlpifDwcISHhxf7PLqTWn4V9mAaPXoOZsyYIetrI5FIOKlTJ6xatUrQP6OFLCwscPPmTZi
ZmcmNf2mjAEJI6RDjnJDU1FRIJBLUqlULAHDlyhVs3boVtra2GDlyJCZOnMg5ISkOFbjKFhXCCCGEkApk6dKlG
Dx4MPbu3SuqJsHDhg2jhrIiUBF6ME2aNAljxozB+/fvwRjDlStXsG3bNsyfPx/r1q3jHY8QInCDBg3CyJEjMXT
oUDx9+hSurq5oOKABtmzZgqdPn2LWrFm8IxLCHS2NJIQQQgjev38PZWXlL+7aRwhRjC1btmD27NmIj48HANSoU
QNz5szBiBEjOCcjhAidgYEBLl26BGtrayxfvhw7duxAREQEjh8/jp9++gkJCQm8IxLCHcOII4QQQkiR5uSEkNI
zePBgDB48GFlZWcjIyICxsTHvSIRUGMOGDYOzszPatWuH2rVr846jcLm5ubKG6ydPnkT37t0BAPXq1cOTJO94R
iOk3KAtlAghhBBCCCkjLi4uSEtLAwBoamrKimDp6elwcXHhmIyQikFNTQ3z58+HlZUVTExMMGTIEKxbtw5xcXG
8oylE/fr1ERgYiHPnzuHEiRPo3LkzAODx48eoXLky53SElA+0NJIQQgghhJAyoqSkhKdPnxaZBfb8+XPUrFkTu
bm5nJIRUrE8evQIZ8+elW2wEhsbi+rVq+Phw4e8o5XImTNn0KtXL6Snp8PTOxMbNmwAAPz666+IiYmhTWMIAS2
NJIQQQgghpNRFR0fLPr979y6ePn0q+1oqleLo0aOoWbMmj2iEVEgGBgaoXLkyDAwMoK+vDxUVFVSpUoV3rBJhj
MHS0hIpKSnIy8uT22hk5MiR0NTU5JiOkPKDZoQRQgghhBBSypSUlGQ7mxb38ltDQwMrVqyAt7d3WUcjpEL59dd
fcebMGdy4cQM2NjZwdHSEk5MT2rVrJ/gdavPz86Guro47d+7AysqKdxxCyi0qhBFCCCEViFQqRVBQEE6dOoXnz
58jPz9f7vGwsDBOyQgRt+TkZNlsjStXrsjNPFFTU4OxsTGUlZU5JiSkYlBSUkVKlUwceJEeHh4oG7durwjKVT
9+vWxfv16tGjRgncUQsotKoQRQgghFcjYsWMRFBSErl27onr16rIZKoWWLFnCKRkhhBBS+qKiohAeHo4zZ87g3
LlzUFNTk80Kc3JyEnxh7MCBA1i4cCFWr16NBg0a8I5DSLlEhTBCCCGkAjEyMkJwcDDc3Nx4RyGkQrt79y5SUlK
Qk5MjN969e3dOiQipmKKiorBkyRJs2bIF+fn5kEqlvCOViIGBAbKyspCXlwc1NTVoaGjIPf769Wt0yQgpP6hZP
iGEEFKBqKmpoU6dOrxjEFJhJSQkoFevXrh16xYkEomsX1jh7EyhvwknpLxjjOHGjRs4c+YMzpw5g/PnzyM9PR3
29vZwdHTkHa/Eli5dyjsjCIeUezQgjhBBCKpBFixYhISEBK1euLLIskhBS+tzd3aGsrIx169bBwsICV65cwatXr
zB58mT8/fffaNu2Le+IhIiagYEBMjIy0LBhQ9mSyLZt20JfX593NEJIGaFCGCGEECJyHh4ecl+HhYXB0NAQ9ev
Xh6qqqtxjoaGhZRmNkArHyMgIYWFhsLe3h56eHq5cuQJra2uEhYVh8uTJuHHjBu+IhIjaoUOHOLZtW+jq6vKOU
mri4+OxceNGxMfHY9myZTA2NsaRIOdgamqK+vXr845HCHdKvAMQQgghpHTp6enJffTq1QuOjo4wMjIq8hghpHR
JpVLo6OgAKCiKPX78GABgZmaG+/fv84xGSIXQtWtXWRHs4cOHePjwIedEihUeHg47OztcvnwZoaGhyMjIAFDQC
83Pz49zOkLKB+oRRgghhIjcxoObeUcghPx/DRoOQFRUFCwsLNC8eXMsXLgQampqWLt2LSwtLXnHIOT08vPzMXf
uXCxatEhWJNLROcHkyZPx22+/QUlJ2HNFpk+fjrlz52LSpEmyojsAuLi4YOXKlRyTEVJ+UCGMEEIIIYSQMvL77
78jMzMTAODv749u3bqhbdu2qFy5Mnbs2ME5HSHi99tvv2H9+vVYsGABWrduDQA4f/48Zs+ejffv32PevHmcE5b
MrVu3sHXr1iLjxsbGePnyJYdEhJQ/VAgjhBBCKhAHB4dim+RLJBKoq6ujTp06GD580OJydnTmkIOT8OnXqJPu8T
p06iImJwevXr2FgYEAbWBBSBjZt2oR169ahe/fusjF7e3vUrFkTP//8s+ALYfr6+njy5AksLCzkxm/cuIGaNWt

```
ySkVI+SLseZ+EEEII+SadO3dGQkICtLSO4OzsDGdnZ2hrayM+Ph4//PADnjx5AldXV+zbt493VEJE6d9//5XNC
CtkaGhIRTBCysjr169Rr169IuP16tXD69evOSRSrAEDBmDatGl4+vQpJBIJ8vPzERERgSlTpmDYsGG84xFSLtC
ukYQQQkgF4uPjA1NTU8ycOVNufO7cuUhOTsY///wDPz8/HDpOCNeuXeOUkhDxqlKlCrKzs9G9e3cMGTIEnTp1g
rKyMu9YhFQYzZs3R/PmzbF8+XK58XHjxuHq1au4dOkSp2SKkZOTgzFjxiAoKAhSqRQqKiqQSqUYNGgQgoKC6Hp
DCKgQRgghhFQoenp6uH79OurUqSM3/uDBAzRp0gRv375FTEwMfvjhB7x7945TSkLEKy8vDOePHsW2bduwb98+a
Gpqom/fvhg8eDBatWrFOx4hohceHo6uXbvC1NQULVu2BABcvHgRqampOHz4MNq2bcs5oWKkpqbi1q1byMjIgIO
DA6ysrHhHIqTcoKWRhBBCSAWirq6OCxcuFBm/cOEC1NXVARTsqFX4OSFEsVRUVNCtWzds2bIFz58/x5IlS5CUl
ARnZ2fUrl2bdzxCRM/RORGxsbHo1asXOtLSkJaWBg8PD9y/f18URTB/f39kZWXBxMQEbm5u6NevH6ysrJCdnQ1
/f3/e8QgpF2hGGCGEEFKBzJO7FwEBAfDx8cEPP/wAALh69SrWrVuHX3/9Fb/99huWLFmCw4cP48SJE5zTEiJ+L
1++xPbt2xEYGIh79+5BKpXyjkRIhfTw4UP4+/tj7dq1vKOUiLKyMp48eQJjY2O58VevXsHY2JiuMYSACmGEEEJ
IhbNlyxasXLkS9+/fBwBYW1tj3LhxGDRoEAAgOztbtoskIUTxsrKysGfPHmzZsgWnTp2CiYkJBg4ciMGDBxfbx
JsQUvqioqLQuHFjwReKlJSU8OzZM1SpUkVuPCwsDP3798eLFy84JSOk/KBCGCGEEEIIIWVkwIABOHjwIDQ1NdG
vXz8MHjxY1qeIEMKPOAthBgYGkEgkePv2LXR1deV2opVKpcjIyMBPP/2EVatWcUxJSPmgwjsAIYQQQgghFYWys
jJ27txJuOUSQhRq6dKlYIzB29sbc+bMgZ6enuwxNTU1mJubU9GdkP+PZoQRQgghImdoaIjY2FgYGRnJ7hh/zuv
Xr8swGSGEEFI+CH1GWKHw8HCOatUKqqqqvKMQUm7RjDBCCCFE5JYsWQIdHROABXeMCSFla/ny5Rg5ciTU1dWxf
PnyLz53/PjxZZSKkIrFw8Pji4+npaWVTZBSkJ6eDl1dXQCAg4MDsrOzkZ2dXexzC59HSEVGM8IIIYQQQggpRRY
WFrh27RoqV64Mc3Pzz87KlEgkSEhIKONOhFQMXl5eX/W8jRs3lnISxft4pOglJaVirzGMMUgkEsHPeCNEEagQR
gghhFQA6enpX/U8ulNMCCGECEt4eDhat24NFRUVhIeHf/G5jo6OZZSKkPKLCmGEEEJIBfC5O8SF6E4xIaUvNzc
X9erVw8GDB2FjY8M7DiGEEFIhUY8wQgghpAI4ffqO7HPGGNzc3LBu3TrUrFmTYypCKhZVVVVW8f/+edwxCSAWQl
ZWFlJQU5OTkyI3b29tzSkRI+UEzwgghhJAKSEdHB1FRUbCOtOQdhZAKJSAgALGxsVi3bh1UVOieNCFEsV68eAE
vLy8cOXKk2Mdp5jchNCOMEEIIIYSQMnP16lWcOnUKx48fh52dHbSOtOQeDwON5ZSMECIGEyZMQFpaGi5fvgwnJ
yfs2bMHz549w9y5c7FoOSLe8QgpF6gQRgghhBBCSBnR19dH7969eccghIhUWFgY9u3bh6ZNmOJJSQlmZmboOKE
DdHV1MX/+fHTt2pV3REK4oOIYIYIYQQUkF9qXk+IaRObNy4kXcEQoiIZWZmwtjYGABgYGCAFy9eoG7durCzsONkZ
CTndISUD1QII4QQQioADw8Pua/fv3+Pn376iZZlEVJG8vPz8ddff2H//v3IyclB+/bt4efnBwONDd7RCCEiYm1
tjfv378Pc3BwNGzbEmjVrYG5ujsDAQFSvXp13PELKBSqEEUIIIRWAnp6e3NdDhgzhlISQimnevHmYPXs2XF1do
aGhgWXLluH58+fYsGED72iEEBHx9fXFkydPAAB+fn7o3LkztmzZAjU1NQQFBfENROg5QbtGEkIIIYQQUsqsrKw
wZcoUjBo1CgBw8uRJdO3aFdnZ2VBSUuKcjhAidImJibCwsCgynpWVhZiYGJiamsLIyIhDMkLKHyqEEUIIIIYQQU
soqVaqEBw8ewMTERDamrq6OBw8eoFatWhyTEULEoLAxvrOzM1xcXODk5ETXFkI+g5ZGEkIIIYQQUsry8vKgrq4
uN6aqqorc3FxOiQghYhIWFoYzZ87gzJkz2LZtG3JycmBpaQkXFxc4OzvD2dkZVatW5R2TkHKBZoQRQgghhBBSy
pSUlNClSxdUqlRJNnbgwAG4uLjIbVpBG1YQQkrq/fv3uHDhgqwwduXKFeTm5qJevXq4c+cO73iEcEeFMEIIIIYQ
QQkqZl5fXVz1v48aNpZyEEFJR5OTkICIiAkeOHMGaNWuQkZEBqVTKOxYh3FEhjBBCCCGEEEIIEbicnBxcunQJp
O+fxpkzZ3D58mWYmJigXbt2aNeuHRwdHWFqaso7JiHcUSGMEEIIIYQQQggRMBcXF1y+fBkWFhZwdHRE27Zt4ej
oiOrVq/OORki5Q4UwQgghhBBCCCFEwFRVVVG9enXO7NkTTk5OcHROROXKlXnHIqRcokIYIYQQQgghhBAiYJmZm
Th37hzOnDmDO6dP4+bNm6hbty4cHR1lhbEqVarwjklIuUCFMEIIIYQQQgghRETevXuH8+fPy/qFRUVFwcrKCrd
v3+YdjRDulHgHIIQQQgghhBBCiOJoaWnBONAQhoaGMDAwgIqKCu7du8c7FiHlAsOII4QQQgghhBBCBCw/Px/Xr
l2TLY2MiIhAZmYmatasCWdnZ9mHmZkZ76iEcEeFMEIIIYQQQgghRMBOdXWRmZmJatWqyYpeTk5OqF27Nu9ohJQ
7VAgjhBBCCCGEEEIEbM2aNXB2dkbdunV5RyGk3KNCGCGEEEIIIYQQQgipEKhZPiGEEEIIIYQQQgipEKgQRgghh
BBCCCGEEEIqBCqEEUIIIYQQQgghhJAKgQphhBBCCCGEEEIIIaRCoEIYIYQQQgghhBBCCKkQqBBGCCGEEEIIIYQ
QQioEKoQRQgghhBBCCCGEkArh/wGlKyfgJqU+sQAAAABJRU5ErkJggg==\n"
          },
          "metadata": {}
        }
      ],
      "source": [
        "data =
merged_data.groupby([\"Type_Occupation\",\"label\"]).size().unstack().rename(columns =
{0:\"approved\",1:\"not approved\"}).sort_values(by = \"approved\",ascending =
False).reset_index()\n",
        "data['total'] = data['approved'] + data['not approved']\n",
        "data['approval_percentage'] = (data['approved'] / data['total']) * 100\n",
        "fig,ax = plt.subplots(1,1,figsize=(15, 5))\n",
        "bar1 =
ax.bar(data[\"Type_Occupation\"],data[\"approved\"],color='lightblue')\n",
        "bar2 = ax.bar(data[\"Type_Occupation\"],data[\"not approved\"],bottom =
data[\"approved\"],color='lightgray')\n",
        "plt.xticks(rotation = 90)\n",
        "plt.legend([\"Approved\", \"Not approved\"],loc = \"upper right\")\n",
        "for i in range(len(data)):\n",
        "    ax.text(i, data['approved'].iloc[i]/2,
str(round(data['approval_percentage'].iloc[i], 2))+'%', ha = 'center', color =
'black')\n",
        "ax.tick_params(axis='x', rotation=90)\n",
        "plt.show()"
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "1. Most of the applicants are are laborers (89.55%), then unemployed
(86.59%)\n",
        "2. The profession of a significant number of people is still unknown"
      ],
      "metadata": {
        "id": "-4DHstYDFeVs"
      }
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "E2h60XofKCjh"
      },
      "source": [
        "#### Car_Owner"
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "analyse_categorical_column(\"Car_Owner\")"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 507
        },
        "id": "KyuN2sFk3Gt7",
        "outputId": "a0e38e74-0597-4e55-fc6a-2ff1bbc015a6"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 1400x500 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAABPUAAAHqCAYAAABsn3sdAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAACD1kl
EQVR4nOzdd3hUZcLG4edMSWbSC6kQeu8dEQUUBARZURR7HVtu7hrWfdby9rL2hDLqiig2MUuKCKgIE16Cb2Fk
AKk92nfH6yjkdCTnEzyu68rl3LOOzPPRAzJw1sMn8/nEwAAAAAAAICAYTE7AAAAAAAAIATQ6kHAAAAAAAABBh
KPQAAAAAAACDAUOoBAAAAAAAAAAYZSDwAAAAAAAAgwlHoAAAAAAAABAgKHUAwAAAAAAAAIMpR4AAAAAAAAAQYGxmB
zgZXq9X+/btU3h4uAzDMDsOAAAAcEQ+n0O+FhYVKTk6WxcLfqQMAgOoRkKXXevn371JKKSYnYMAAAA4LilpaWpSZM
mZscAAAD1RECWeuHh4ZIOfWMUERFhchoAAAADgyAoKCpSSkuL/HhYAAKA6BGSp9+uS24iICEo9AAAAAAAAAAAAA
NWJTT0AAAAAAACAAEOpBwAAAAAAAAQYSj0AAAAAAAAAgwATknnoAAAA1xePxxyOVymRODAcRut8tqtZodAwAAAANDC
UegAAAJJ8Pp8yMzOVl5dndhQEoKioKCUmJnIyBgAAqDWUegAAAJK/0EtPT1cIuIIIYBgJlAjlDI6Lz+dTUmkpdpVQ
UkmJwIAAA0FpR4AAGjwPB6Pv9CLjY01Ow4Cjs1ms0hPSUklo1eRmEYaQAAAAAAAAAAAAAAAAAAAAA
eYT9GAABQWyj1AAAA/ocltzhZ/N4BAAC1jVIPAAAAAAAAAgAAAAAAAAgogwcP1sJJE82OAAQAAYCo0ygAAADiCm
ZszavX1Lmx3ciemLL68WGeccYZGj8ihr7/+uppTAQAAoC5yygAAgDCUSDAGGAAAEOpBwAAAAAAAABBo8H7dQy8kJ
MTkJ8Pp8yMzOVl5dndhQEoKioKCUmJnIYBgAAqDWUegAAAJK/0IuPj9+JKILLcJKIAAAAAAHISU6LRfK36O3Tl0+v
V+eefr5ycHC1YsCaxTsTm8mbEEs5BA6FEr6OwkiUSvdmXAiE9+KWZyIu+Cn8Bqk22pzHfpc2CISQV+Ps7kdLS1a
9OHs8dzzKRQ/UJLA9eQsmhFm9CRkFAkKdOfFQHAcjIX1xU0UONaQHk7KF+z0ADa1pxjkNmpx2b24SBpE2VKoy
0kfK1kMlFd/Ly/Fpx6QCPYHrvVkbnKmfg57ll2sggYPFqk3MzAwKdmF0jP8cggbTNhUD1Io8bNZqPBS0p5P8xLM
rPZjy3BMUg48vRKDJHFlVsbYBNkSAGs2CXFFhG5lPIzH5gVKxkNj44FwjCZ1Ld++1Y7Hd29Hv7dx4hNVaZQt
ZC8tKnuw9PJrrH4tUcDDjaMNh4YgT7fu+zQ49Jm7S3t+9c+0NfTEAlbNzGRw6AK/WDR1bkfDkfYjWqvD+zl/hzzmK/G/
xzW0fjnEA3vJFnG/LzMTw5X+SlEe+MuBzK1b1Fi+DaNVVjvYyY12ZjLuF0P3U/Lrr9PDA/BP+kQN4kXZn22rtcDr
XLQYO9N6+gDSYY6zd9SHPiKhfL8v50Xq0jG6XI2Q3aTI8X2qbjkFZt5gZkO+xg1oHCnXsJRrg10BZ8OfyeLPtg
fzbuMQ/eXnwzP4MDrw9Pzn9PFxuOrwU3JzzqIiHudh5PL++Kro57n8sx7IkTmLoLhuh6V0B5gxbwT7aQo+pDeEnLI
UKC9fPK3nDsxZpXuBu7XSmi3Ilt2+sHF3j+hEvvdYmL3nBKNvIxzDFjFJ6crRvU6Ra/2OOD8xqw5N4qOMbvDZ2hd
Ta2CKHlV8LWaQ2xCKo1HSa3S+9pgLx4Imdq/rX0YWg1d6zWmf+hdmyxWUfmPYNP9ozdBHYV+AHoRsPXqZ+gD/2ne
qaUHo14KHOxjPN/Yan6RBwLVUkE5R4Kj9PqgJtLk+Cb0QgdsBYGtC2tkNRzvV0ETC+7BCNJT4JmNvWLbUGZfzcC
1/iRi9XpCaawMa5mG9QKZX8k7BfJ5vqtFj6LP1zPfhtNKTqX0xRn9S8Bj5qSZ4JbdiSi+cyaywGjHsa7MUzFAJKK+z
1FZp3rBADCkOQDdIA6gLBdiD8L5x2HcNjq7mWnnwWzEGqXgvmWY8uafGgy/UPbsL8Pos9bgIOd8Tgg58wbiTOZ+LTj
cAqlXAL6FqUAGSKrdYtxOUSqfL0acRpnk7tUwxOwS8MdlaPzlxYmLMUpZOQZTqrvHOLl1evqPagtNtiTs7+i6X/ZGx
UIsT/0rNIpLKvS66Cpou3rpUcFhRAhqjAXVNe6Wo3FKPDjWzPb2Ee8ZYZKpSwMHPRaD/bk0f32kjA/sAGXHjOhybd
mPVSRd8X0ctUNn09e0oBSSfeqWbPEeuoyqfbQqG1fZTmeUb3SNmdRuCpzs3OZvLXYV+qnv4dTLVYijo0FYn1LpGd0G
J/L7FKn1f3tLbb1+bkM1EIPKm8awfbm/J6u2QFSIBUwbzLCBRPlpvk59jNC+tMxYIObuJuTHHbV8xL+HzIpmCSd+5
NLMVHZoKF0yQ6s7puVPnLb2Z3RKzfrlg2o5Zl0sT1MmBi0jFLRwO2NNsDeXFh5kFqDhqvNUdy+Usd9QHf+y3UOuW90ma
yTgo1/HTU6J6JiqmVk70dIt1WMLDD3zfYSXuplx0MQbNrf/6Jqf4sOFVSKNpkWdMxzHaxjDz4N4RZHmS5h4FS9lCl
AIb64ZiOOZi6C+EVyLUO1THBcFHa8+sI0ng0Qr12zySs8AfH3FKQr5jdJUs3cPLC6usz4aYcOIDpZgnv2Z2z1S8IC
rrU2wa5QTlUXyG2XHP+lhkEQO+aIdcxfU2GkldE6EC7WfnMWRsJ8s1QOYN7m/XTpaDiI27k5XIIPdRtuDJ13TIzwtBx
ekK+ylK1dVdkhpEsDzkvj2RQNoOZ0EZmJIRV6J1B3SzMVqUuhZ5S5YB2Amt2wV+e95Q7LeNnSyF09trbf2aU+vf9rr
O9JpV6jFYfdzd7mfr/6t2Ep9Ma4fRTGiPKDbnMCNvHHdXkx+PcUsV2x0zIv/uC9PiIdUXPlwy33l1RvY+KCD7pEbSS
Ph3g0TsRrfOPHnP/NH0fVi5G3jjOFvKDpeWhuY+qHjk1rfHRZi5/bUXa+ksuP8+NSt96IwkWJ7DCFmBJN06TydTy
XQMvNpZUhnrW+ec/BVGZQ+x3lX5FHvlJCBBojqzsEQ+gEpk4Y4ymGhTpgcxlb5hULUaPfsiaecf5SkGYG2Nk6IKFoFt
bZzaAcuCiH+6EqsYuapnp2+LiJqkfdHvr5ntuRVN4b9Vx4yDpvy9PzDHgVD9/jz4I6rrvNjFXkS1+Gt5fNmxPNeFy
HrCNEPz6UsfBfbR3iSIJC8+LZVlObb3tF9pGO81P0QL1Oh1+Kf2Il6bu9qK18GObsGaxkm0nRUdqVNhPdbs1QGqw76QZ
b/EhlOtbDSWUvxpy/ZqTHRWXAvBJUhZLxcoxQrnqwtd2WAm6Qd0QlkmRUDMxbBvJJjhE3Aqx8d5ePy12IR8Ehgro
ZlnfcIXc6P6dpAACDkl6LXnHK/CZfVJf/nz+8//4aV8hrwmO91bbaHhz94mvCjPPqYLKYgkzvJjQxgzVxXXmJ7OodeGBH
/DcGTncWzS7r89Vx9EqVT/ruJz4QUUjAzqc/5GmHMqRBuTEOjI6zFakbsl+Vmzgt33b/Z6AsJwaU0cPqn26GjCtAF
nNetCjcXLmAstBMDNfkUuoZeoBNN7LuQcwwO/zsCWcf9gXPdEZsyqQ9YKNUeX8eBy/dVIqwyL+LLLPLHpjSs2wHGMKh
r0bb/W7yL9Nu9SrOcDOuQPXeRNh6/R3XppbGtSZsvbAjQKS8dZIAmavpMwQIDEFjNyXdCE7VzMUyXmfxpiEU4ZDHG
```

HDMM46iy/Z599Vl26dFFoaKhSUlJ0yy23qKioyH9/6tSpioqK0meffaY2bdrI4XBo+PDhSktL84958MEH1b17d
/33v/9VSkqKQkJCNG7cOOXn5/vHXH311RozZoweffRRJScnq127dpKkdevW6eyzz5bT6VRsbKxuvPFG/+t/991
3cjgcysvLq5T5r3/9q84++2z/rxcuXKgzzzzxTTqdTKSkp+stf/qLi4mL//ezsbI0ePVp0p1MtWrTQjBkzjvq5B
VB/DRw4UMOHD9e9995rdhQAOAylHgAAQAA7nn2fVqxYoXHjxmn8+PFat26dHnzwQd13332aOnWqJJGnmzJlqOSq
JHnroIWVkZCgjI+OIr2exWDDRpiRt2LBB06ZN0w8//KC777670piSkhI9+uijmj59fqVCY8vWDqRhRHkhW7t3rrZzZv3qw5c+boq6++UnFxsYYPH67o6Ggt
7bpww8/1JdffqnZs2dr1apVuuWWWyqNmTt3rjZv3qw5c+boq6++UnFxsYYPH67o6GgtX75cH33+kb7//nvddtt
tkqQhQ4YoKipKn3zyif85PB6PPvjgA02YMEGStH37do0YMUJjx47V2rVr9cEHH2jhwoX+55A0FYppaWmaN2+eP
v74Y7388svKzs4+xn8JAPXVE088oS+//FKLFy82OwoAVMLyWwAAgAD3+32fpkyZct j9Z599VkO0GDNF9990nSWr
btq02btyop59+W1dffbViYmJktVoVHh6uxMTE077WxIkT/f/evHlzPfLII/rzn/+sl19+2X/d5XLp5Zdf1v79
yRp2rRp6tChg5YtW6a+fftKksrKyjR9+nQ1btxYkvT9cYbb/iX3b7++uv+x4WGhkq
SJk+erNGjR+vJJ59UQkKCxo8fr3fffVfXXXedpEPFYF5ensa0HStJevzxxxzVhwgT/+2jTpo0mTZqkQYMG6ZVXX
tGePXs0a9YsLVu2TH369jTZkyRR06dDj0/xoA6psuXbpowQJmjRpktlRAKASZuoBAAADUA08++aSmTZum1NT
Uw+6lpqZqwIABla4NGDBAW7du9lcfjOaHX+f777VVXbt38hd27hN8xQAAAAAAAAAAAAAAAPnAo4hAgAAAAAAAAAAAB27
GKSoqSpK0Zs0aTZ06VWFhYf6P4c0/+v1aufOnUpNTZXNZlOvXr0y5Jyw+gg4XrooYfk9XrNjgEAlVDqAQAAAAA1A01s
e/Trl27dN5556lr16rtF4JNPtGLFCr300kqaS2qqSjLwDvkUd1Zg3CprY6NoaAqkTiiLGjSUGmDpEqL/+c3Fa
NN2n16tX+jzVr1mjr1q1apVdcYHEKCmTp2qzz77rNK15s2bq7y8XD6f35xQAFAFlt8CAADUE0888YR6d+/uP
1TiVx06dNCiRYsqXVuzXzZd0aTHatm0rq9UqSQoKCjWrLOVK6bI6/Xv/6eJJkwjNATB/IZ1Ulv/XmdIcKjMj1tps
1TiVx06dNCiRYsqXVuzaJHatm0rq9UqSQoKCjrmrL0VK1bI6/Xqm6Eew0Yx6H5+fwPP8wPGudlu/fLLL/6ltps

pVlO3POn283a9muHLOjoIEqLHdr8rxtmvrzLl1+WjPdOLClYkIp91C9KPUAAAAAAPXCur35evq7zfpxy36zowC
SpKJyt15dsF3TF+/SNQOa6+bBrRXGgRqoJvxOAgAAAAAEtK1ZhXrmuy2avSHT7ChAlUoqPHpp3nZ9sHyv7hzWV
uN6p8hiMcyOhQBHqQcAAAAACEgHisr11OxN+njFXnk5zRYB4EBRuf4xc52mLd6t+87roNNbNTI7EgIYpR4AAAA
AIKB4vD69vXiXnp2zRQVlbrPjACcsNaNAl72+VOdOTNA/R3ZQiOahZkdCAKLUAwAAAAAEjOW7cnT/5xuUmlFgd
hTglM3ZmKX5m7N1Zf/m+suQNop02s2OhABCqQcAAAAAqPOyC8vOxDebNHNVutlRgGrl8vgOZeFOzVy5V3cOb6f
L+jaVYbDfHo6NUu8kbLHHmhOBQD3T2ewAAAAAdZTb49WOxbv13JwtKipnqS3qr9wSl/7vO/X6fPU+PTm2KOtyc
UwWswMAAAAAAFCVX3bl6LwXF+rhrzZS6KHBWLYzRyOe/1GvzN8ut8drdhzUYZR6AAAAAIA6pdztOaNfb9S4/y7
WpsxCs+MAta7c7dWTszdpzMuLtGFfvtlxUEedUKnn8Xh03333qUWLFnI6nWrVqpUefvhh+Xy/nR3u8/lO//33K
ykpSU6nUOOHDtXWrVsrPU9OTo4mTJigiIgIRUVF6brrrlNRUVH1vCMAAAAAQMBatzdf501aqNd/2imv79jjgfp
sfXqBzp+8SE/O3qQyl8fsOKhjTqjUe/LJJ/XKK69o8uTJSk1N1ZNPPqmnnnpKL774YsU1Tz31lF544QW9+uqrWrp
qVLFRoaquHDh6usrMw/ZsKECdqwYYPmmzr776Sj/y6ntXAAAAAAICA4vJ49eycLRrz8kJtzWbSB/A
rt9enV+Zv18hJP2n5rhyz46AOMXy/n2Z3DOedd54SEhIOZcoU/7WxY8fK6XTqnXfekec/nU3Jysv7+97/rzjjvvl
CTl5+crISFBU6dO1fjx45WamqqOHTtq+fLl6t27tyRpGzlbmI0ZTXh/ZaxXbFVjvEOTMnDIUt6dKnKuVsSeNeQwB80w/ZEOE+2LV41o3ZeTbxo/Wx/RJ5ZXl2aKCTPQmvMlrd7gRRGdXVJLsZ5BCrrRuOBIRu/fTBLM7eAEUhKVp60uEotXpJpOpTfIDKHD/vpmhGLNe97RLPSZLuQFYXo3JK2ocrVVZDQYY5Ix4P7svVyG6+jmlWdHSWyE1j0GsI3tyRNIz+/+w/qi3+rUGzkmmUrs+Y36JI9rNQNIhrOCPvs2lX/CFxZzVqkUvhzE2rmrDH6xOaDDbv6iEz8A/hR7i7fOMJ0bLH5l9yl2FB0wTAHZTK8ta98FRY4RbxiAnuTDsRjP5xVWP8GSLfHdRzaS5gc2yJsdtWrGzwi8PMv6hFzT4ck3rpnCZ3hYJ8gGMqUfGcGLBXqxqnFpQMlmi1FaFqtbXSzeGRRSxtCALnOPgKeiahlw1BHcGlutWRSg5SaLPjxV37XjbDK5oqX4XxuAzCGbmcb6npnkGa5VMW/ibVsZGNWMW2SnFd/g+3n5tCGsQTWMJYWSzTH24DK6jDy7e5HjvNkh2Rzvw4sq1yMVS7xqaF5Apgu+bB+sXgRm1Egm0S0WZXTF40YBbLVgK5tK/6qN7TT4+v9tkj6QPXmTSq+q7VqvQ3ZuIqjaYnKzSiV/KuqqtfBHFl6XbZWyv0xAa3TNhAc0FJI0vknhSrCX2KNkrl1Yx/gzX1jWEr5kTGAMtFVPM+B1pPo96LHVq/KfjxwNX3WK+UKPVaWKYLZSXixCoxXBWBtsnqoUAHnqjvs4G92XAJbIx4LSa6zzv4Cmrz5srVeZEPx/GvlkKqPX4s5dcMY6cl91LnkZyJX0EhhDZo5oCUX8RNRW2ed9jivOPspGR6AAAAAYBAE3V/tJz6xSa0eBpl2yt1aP5oEZVrnAqdsl0aTCSN8eBbl9pWk0ZUj8HcH3Lg2lUz9XG8AAAAAAAA0K2Z5qQTcVq3OyoFAAAAAUd2XpKPBbXm+d9rO2q1tQYHhC3PjYH/zT7a7mXpt8xYzNsmTKS+1fzVxLjAp4DEHBCQ4YUk7CvPU/Vy/dqMsH+42e+lWsv/PerAXJ5WHJxuShEUxe+KZmzY5N8DYe4X8gaW/zTPT2+ZRbdwvNJ1XEPHSbXHbj8Y+n8+WYW3gfWL3xI1Iz6uy1fvPHBOwz2Ey1ULqeAW0vbxJaWHjWSzgGAAAAUlpSXEp1gw6HOjk8xYrM/cGp0J87uIKRgZ34b5f0uvbmpWxnx9X9ycNKDKFc4v3WCI4/4vtzgctH7PRL1Aj60+DRpVbIYWbQ9CV6JbsXLDVcZBGtVQQdXaLx/bpu7qI6X5LqLjSmZ1CH/oA8NS4S1bM81e+dTVp0lB0xI1AAAAAyU5FfqGlgbLsg8FMZnmeUbH+6U2OYB4Rl1qyDYOsCXy0tZAAAAAP8qRzAK9zMwOZbIchN0B6ATktVBgZhFm/l6xglhRXFtpt6AJ3kW9y6D8AuF71ETOWUgwJmwHUSTNBG8LDA16h3hY3GLLEmB+7NRkgCGjUM3HEAUH7MFC9ZaGCTCVYIWmVf0MxYVgYYWvVJDNLaYy0c1zULWQi9uXGQZmBZBVXdMmGTYDRoNM8C0CWSUq6AnFt4Z7Gs8XxCg3s0RavD0BHrAVVWhSZ1fJUmzRmONJJakOUKaXyxAnpuVkNf1MXC6AZJbpQXSKPrmOS4VPX8fgZkk/Z0YoRKXOb5G/ogLNU+gMsmkeKl1hJnJfOgpQtrnuG6QBzTNbXQzDqVhJu6FWnSr/5VI5U6FaSwSkNmnJ65ISn2OZnqVs8C3KAQuuFF/oybgUHQ==

+nqO8OBTtdUeYHadesjdqqoRLHv3tguXQTEhvRZmyP7xP9vgWSrjOMUlS3k/vKPuTh5R4xTMyjKpnTHrKipT5z
t1yNO2q+IsflCUkUu7cfbI4wiRJPp9P2TMfkWGxKe7Cf8kSFKKC5Z8p64N/Kfm6V2QJcqhk21IVpy5Q/LiH5c7
dp4OzXpCzRU9ZQyLlLS9W3o/TlTD+kZr9xKBBKyp365YZK/XpLafLYbeaHadeY+41AAAAANQz6ZnZmvHp18qps
ChVHIxRYyxWWcOif/sIiZQkladvlDs/W41G3qGguOYKimuuRqPuUEXGNpXtXnvEpytY8rFsEY3UaNREBSe3kzO
qUc4WPWWPTpIkuXP3qWLfZsUMuOXBSW1lj22imOG3yOeuUHHqAkmS62CaHCldFJzURqEdB8kICpE7POuSlDvvL
YX3GClbRHwNf2LQOKVmFOjf7K9X4yj1AAAAAKAeKSwq1rSPvtDufVnaGdpJbvbRqzHu3H3a+9KVSn/1Ou3/8mm
5C7IlST6PS5JkWO3+sYY1SDIMle/dcMTnK922VEGJbbT/s8eV9uIE7XvrLypcPdt/3/+8tt+WUhuGRYbVrvK9h
wqUoLgWqsjcJk9ZkcoztxlawhudrLK9G1SRtV3hvUZX3ycAOIr3lu3R56vTzY5Rr/HVHQAAAADqCY/How+++FZ
rUreoKL6rDrCPXoOJTmqn2JF3KP7ifytm2C3y5GUpc8Y98paXKDi5vQy7Q7nz35LXVSZvRZly5O2RfF55inKP+
JyuvEwVrvpGtuhkJYx7SOE9Rip37msqWjdXkmSPaSJrRJzyFkyTp6xIPo9L+Us+lqfwgDxFOZIkZ8teCuOOWJn
T7tDBr59To1F3yGIPVs63Lytm+KOqXPWNOl+/SZnv3KWK/btr5XOFhuufM9dpx/4is2PUW+ypBwAAAAD1xE/LV
mnBkl/kSGipZaXsoleTnK16//aL+BYKTm6nva9cq+JNCxXebZjixvxDOd+9rMIVXOqGodCOgxSUOEoyjCM/qc+
n4MTWih5OlSQpKKGVXAd2q3D1NwrrMkSG1aa4C/5PB2e9oLOvjJcMixzNu8vRspfk++1pos6YoKgzJvh/nbfwX
Tmad5dhsSp/8QdKvvYllW5bpoNfP6ukq1+o7k8N4Fdc4dFf3l+lz24ZIJuVeWXVjVIPAAAAAOqBPekZmjnre9m
coVpaniyfjlIeodpZHGGyxzSWO2+fJMnZoqca3/SGPCX5MixWWRxhSpt8uUKiEo/4HNawaNkbNa1OzR6bopLNi
/y/Dk5sreRrXpS3vFg+j1vWkEhlTP+bghLbVPmcroNpKt44TOlXT1LR2jlyNOksaOikQtqfqYOzXpC3vESW4JB
q+AwAVVufXqBXF2zXbWdX/XsUJ4+aFAAAAAACXGlZmd77fJb2H8zTDkc7lXg5cbK2eStK5c7LkDUOptJ1aOikL
I4wle5eI29xvkJa9zvicwQ37ihXzt5K11w56VUebGEJDpU1JFKunHRVZG5TSJvDn9fn8+ngty8p+uzrZQlySj6
vfF73/wL/758+7wm+U+DETZq7TVuyCs2OUe9Q6gEAAABAAPP5fPp67k9atX6zfIkdtLvcaXakBiH3hykq27NO7
vwsle1N1f6ZjOqGRaEdBOmSitbOUXn6JrlyM1SOYZ4OfPaEwvucL3vsb6cRZ73/TxWs+NL/64g+56t832blL/5
Qrtx9Kt44XOVrZius5yj/mOJNC1W2Z61ceZkq2bpEWR/cp5A2p8nZoudhGYvWfCurM8JfJAY37qCy3WtVnr5JB
cs/lz22qSyOsJr6FAF+FR6v7vxojTxe37EH47ix/BYAAAAAAtiajVsOe/4iRTaK17ySmGM/ANXCXXhAB758Wp7
SAlmdkQpuOlGJVzwjaOikpEMz7HJ/nCZvaZFskfGK7D9O4X3GVHoOV26mgksL/L8OTmqruAv+T3kLpilvOXuyR
SYo+uwbFNbpLP8YT1GOcn94Q57iPFnDohXW6WxFDhh/WD5Pca7yF3+oxMuf/u35k9spou8Fyv7437KERKrRqDu
q+bMCHNnavfl6dcF23XpWa7Oj1BuGz+cLuJqOoKBAkZGRys/PVORE7W/+OnNzRq2/JoD67cJ2SWZHAADUELO/d
OX9djA3X8/8d5r2ZmZrb1R37Shjlh6AuivIZtHXt5+hNgnhZkepF1h+CwAAAAAByOPx6KOvvtWOPekKSmxNoQe
gzqtwe3Xnx2tZhltNKPUAAAAAIAAtWLJCC5evVuPGyVpSGG12HAA4LmvS8vTajzvMjlEvUOoBAAAAQIBJz8zWZ
9/+oNAQpzZ7E1XkYbtOAIHjue+3aFt2kdkxAh6lHgAAAAAEEI/HoO9n/6DsA7lyNGqiDcWhZkcCgBNS4fbqwS8
2mBOj4FHqAQAAAEAAWbpqnZauWqemTZK1KD9KPhlmRwKAE7Zw2wF9uyHT7BgBjVIPAAAAAAAJETl6+Pv92vux2m
3YrTgdcQWZHAoCT9sjXG1Xm8pgdI2BR6gEAAABAAPD5fPp67k/anb5PcYmNtaIw3OxIAHBKOnJK9cZPHJpxsij
1AAAAACAArN+8TfMXL1dyYrxWlUTJ7ePHOQCB7+X525WZX2Z2jIDEnwIAAAAAUMeVlJZp5qy5qqhwSaGx2loSY
nYkAKgWJRUePT4r1ewYAYlSDwAAAADqu09/WqLUrTvVomkTLSuI4HAMAPXK56v36ZddOWbHCDiUegAAAABQh+1
K26dZ8xYqNiZK2d4w7S13mBOJAKrdg19ukNfrMztGQKHUAwAAAIA6yuv16os585VXUKiERrFaVhBhdiQAqBHrO
wvOwS9pZscIKJR6AAAAAFBHrUndohXrUtWOcZK2lYUox2O3OxIA1Jjn5mxRmctjdoyAQakHAAAAAAHVQRYVL38x
dKK/XK2doqFYwSw9APZddWK7pi3eZHSNgUOoBAAAAQB2OdNU6bdy6Xc2aJGl9UZiKvVazIwFAjXt1wQ4VlbvNj
hEQKPUAAAAAoI4pKi7RrHkLFRwcJNkdWlMUZnYkAKgVOcUVenPhTrNjBARKPQAAAACoYxYs+UU79qQrJTlRG4r
C5PLxoxuAhuP1n3Yov8Rldow6jz8ZAAAAAKAO2X8wV9/9uFjRURHyWezaWBxqdiQAqFWFZW5NWcRsvWOh1AMAA
ACAOmTOT4uVlX1QSfGNlFocqnJm6QFogKYu2qmCMmbrHQ1/OgAAAABAHbErbZ9+XLJCCfGN5DUsWs8sPQANVEG
ZW1MX7TI7Rp1GqQcAAAAAdYDP59OcnxYrv7BIjWKitKUkRKWceAugAZuycCcn4R4FpR4AAAAA1AF7OjOOfMOGJ
SXEySdDaznxFkADl1/qOowlu82OUWdR6gEAAABAHbBgyQoVFBYrOjJC2OqdKvLYzI4EAKabvni3PF6f2THqJEo
9AAAAADBZema2Fq9Yo4S4WEmG1jBLDwAkSel5pfp2Q6bZMeokSjOAAAAAMNnCZSuVm1+oRjFR2lXmUL7bbnYkA
Kgz3ly4O+wIdRKlHgAAAACYKOvAQf24bKXiYqNlGAYn3gLAH/yyO1fr9uabHaPOodQDAAAAABMtWr5EaP2O
brRyXTVkVwWZHAoA6561FzNb7IOo9AAAAADBJTl6+Fiz5RTFRkbJYLNrELDOAqNJXazOUXVhmdow6hVIPAAAAA
EyyeMUaZeO/qIS4RnJ5DWOtdZodCQDqpAqPV+8s2WN2jDqFUg8AAAAATFBYVKx5Py9XZES4rFaLtpc65fLxIxo
AHMm7S3er3OOxOOadwZ8YAAAAAGCC1Rs2Kz1rv5LiGOmSUktCTE4EAHXbgaIKfbkmw+wYdQalHgAAAADUMq/Xq
4XLVynIbpfNZlN2hVOHXUFmxwKAOu+D5SzB/RWlHgAAAADUsq0792jrzj1KjIuVJKVyQAYAHJdfdudqz8ESs2P
UCTazAwAAUJX169ebHQFAPdK5c2ezIwCVLF+zQaVlwaFuzo5iOmXoAA
AAAUIty8wuObPU6xcZESZK2lYTIwwEZAHDcPluVbnaEOoE/OQAAAACgFq1av0n7D+YqLiZGkrSt1GlyIgAILLs
OlmjF7hyzY5iOUg8AAAAAaonb7dbJCA6W1WpRvtuq/eVOsyMBQJ37Jp+y5t371XiQmNJ

EnbmaUHACfl67UZKnd7zI5hKko9AAAAAKgly1atU4XLpVDnoTKPUg8ATk5+qUtzU7PNjmEqSj0AAAAAqAUHc/O
1Yl2q4mKjJUkHKuzKd9tNTgUAgWvmyr1mRzDVCZd66enpuvzyyxUbGyun06kuXbrol19+8d/3+Xy6//77lZSUJ
KfTqaFDh2rr1q2VniMnJ0cTJkxQRESEoqKidN1116moq0jU3w0AAAAA1FEbtmxTTl6+Yq0iJEk7yhzmBgKAALd
gy37lFleYHcM0J1Tq5ebmasCAAbLb7Zo1a5Y2btyoZ555RtHR0f4xTz31lCZNmqRXX31VS5cuVWhoqIYPH66ys
jL/mAkTJmjDhg2aM2e0vvrqK/3444+68cYbq+9dAQAAAEAd4vP5tHzNBtntdlmth34M28XSWwA4JS6PTz9sarh
LcG0nMvjJJ59USkqK3nrrLf+1Fi1a+P/d5/Pp+eef17/+9S+df/75kqTp06crISFBn332mcaPH6/U1FTNnj1by
5cvV+/evSVJL774okaOHn//Oc/Sk5Oro73BQAAAAB1Rkb2AW3Zsdu/9Pagy6YCzwn9OAYAqML3qVka26uJ2TF
McUIz9b744gv17t1bF198seLj49Wj/Rw+9/vrr/vs7d+5UZmamhg4d6r8WGRmpfv36afHixZKkxYsXKyoqyl/oS
dLQoUNlsVi0dOnSKl+3vLxcBQUFlT4AAAAAIFBs2LJd+QVFiooIlyTtYJYeAFSLn7YeUIXba3YMU5xQqbdjxw6
98soratOmjb799lvdfPPNvu+stf/qJp06ZJkjIzMyVJCQkJlR6XkJDgv5eRZmz4n+PhK92O2m2JiYvxj/ujxxxx
GSk/yMlJeVEYgMAAACAaXw+n5atWienI11iGYUiSdrGfHgBUi6Jyt5bs0Gh2DFOcUKnn9XrVs2dPPfbYY+rRo4d
uvPFG3XDDDXr1lVdrKp8k6d5771Vr7vVq7/fr7/Iy0trUZfDwAAAACqy96ML03au0+N/rf0ttBt5dRbAKhGc10zzI5gi
hMq9ZKSktSxY8dK1zp06KA9e/ZIkhITEyVJWVmVP5VmVP5lZWVn+e4mJicr0rryJodvtVk5Ojn/MHwUBysiIqLSBwA
AAAAEgtRt01VYVKyIsFBJ0t7yYJMTAUD98n1qwzws44RKvQEDBmjz5s2Vrm3ZskXNmjWTdOh9XIzCw9GWiN2d09d/v
6CgQEuXLlX//v0lSf3791deXp5WrFjhH/PDDz/I6/WqX79+J1GAAAAAKCu8fl8+mXNBjkdDv/SW0o9AKhe6Xm
lSs1oe0cvnFCpd8cdd2jJkiV67LHHtG3bNr377rt67bXXXd0utt0qSDMPQxIkT9cgjj+iLL77QunXrdOWVyo50
VljxoyRdGhm34gRI3TDDTdo2bJlWrRokW677TaNHz+ek28BAAAA1CvpmdnalbZPjWKiJElen5RBqQcA1a4hLsE
9oVKvT58++vTTT/Xee++e++pc+fOevjhh/X8889rwoQJ/jF33XWX/9OX09b889q0ePioqKNHv2bDkcv20E02PGD
LVv315DhgzRyJEjdcYZZ+i1116rvncFAAAAAHXA5u27VVhcrHrijYC7VFhcrIjwMElSdkWQPKKnwn9GMYA0A4NMQluLYTfcB5552
n884774j3DcPQQw89pIceeuiIY2JiYvTuu++e6EsDAAAAQEDZuGW7bDbb4WMIgdNNmbpwNF5WoU1nC+z
vJXRAAAAABQAwqLirVl5x5FR/520B/76QFAzfD5pGU7c8yOUaso9QAAAACgBuzYk668/AJ/qVfmNXTAZTc5FQD
UX5R6AAAAAIBTtmNPmtwej+z2Q7PzffA4Ot2F1lg8tDWpBwAAAAA4BT6fT+s3b1dIyKEHPJ7Ny8EHy
kCq+hPLfN5FQAUP81pH31KPUAAAAAoJpt2bFLhmHIYjn0I9d+l1iPz0AqHENaV89SjOAAAAAqEbl5RVav2W7I
iPC/ddYegsAtWPN3nyVuz1mx6gVlHoAAAAAUI327Mt1KrgCtP16Bi1glIPAAAAAKr
R3owslZdXyBH822BqQ0NZgkupBwAAAADVKG1fpgzkDkGEc2kOv0G1VqddqcioAaDjWpzNTDwAAAABwA
nw+n7bs2K0Z2wOBD/NWbpAUDt2pxZaHaEWkGpBwAAAADV5EBOng7k5ik8LzNR/jUMyAKB27TpYrDJX/T8sw2Z2AAA
AqrLFHmt2BAD1SGezA6DB2Je1X0XFJUqM++3PMWbqAUDt8vqkLVmF6tokyuwoNYqZegAAAABQTdIzs+XzSU6rt
TZ/IpdSDwBq3aaM+r8El11IPAAAAAKrJzrS9p+stl+OxSjxGNRhY8fuwCgtm1qAPvq8acLAAAAAFQDl8utHbvTG
7/fTy3ex4BABm2JRZYHaEGkepBwAAAADVIHP/AeUXFlY6COVFLAMAAACAarAva79KSksVEuz0X2PqAUDt8vqkLVmF6tokyuwoNYqZegAAAABQTdIzs+XzSU6rt
2OmHgCY42BxhbILy82OUaMo9QAAAACgGmQfzJVkyDAMSUYvXzd70+yblUZTNfwMPNy1NSn+kAY9lgpMPAwT32frUepBwAAAADVIGv/QVkslX/EYqYeAJh
nx/5isyPUKEo9AAAAAKgGu/dmKDjYWelaDgzqNnWxHpAAFGp2Aj9AAAAAAAgG7Gvz
yafjKM8CgBQk9Ip9QAAAAAAR5OTl6/SsjI5ncH+ayy9BQBzUeoBAAAAAI7qQE6eSkrL5HSElLrOzHIr3bBmpvnL0TISwLDsW6a9oBJrXQTX0rsvFQp1S9IPLg2oLmp3mbSF
FAAAAABzVwdw8eX0+Wa2//YhVQqkHAKbaX1gul8cru7V+ruKqn+8KAAAAAGrRwdw8Wa2VSzypWbxKKMfqhdYz0Cav6lOdnG5oqOLJOHUeANvSvuTLGFLbPJWbfd1Q+F8pbgDDkZZSxJzJLzk2HcZlcNE9wFgxdAsvRagS1GSYHvTd0ScPbaS+Z1AgkxrjeNVtCR5CxHnDF+oAvJZCpYG5zWsTiQSU+OHvYFX6XJi+sL4hwDSzYWeJ6/oTQgNMxUfcJLRIJYMqvkFBqp8aLHyK/2lfRCTU6QGypHTWe20HljkcPX1OFiQ+k9XS3VP4Wyp+LXprzVO96tFq/Ms+tTIkz5jo1xNVYhXj1mIq7i1ltlSQp9yapT3vyYWSM1e0o6SvDiQbTxuLOmtzNjYe6t+L1SbM2
ACgbigoc5sdocZQ6gEAAADAKSgqKZXb5ZHdZvNfYz89AKgbCstcZCksMlx3ZCcbEDfmEvNNjG2f5rrSg6sVBRQK8QeT41Ifi7VHSjdRZ3nWEN8d67K7UNNMfmaoIDAdmAAFxjEpgMTnx2T8y9R2hvhWtwNdA8SstcZPtxl6ltxyuIHuKpbYPTG5+2+KJYGrUF9jhBkH5MDq+UQnHOzwXAKvDXuLNqHmRAvDxHQqYH9TPytBwWlH1pxAKBsT+stE7qOHsE9D6H5mAoSALAXu+38pG9z5SOK
1QyEz9QAAAAAAVSkqPnwT9jJm6gFAnVBQykw9AAAAAEAViopL5PNVvubyGVUPBgDUKmbqAQAAAACqlJ0XL7vdV
umah1IPAOoE9tQDAAAAAFSpqlLPTakHAHVCcYVHHq/v2AMDEKUeAAAAAJwkn8+nvIJCBdvtla5T6gFA3VFUT5f
gUuoBAAAAwEkqr6dQWXm3H8o9Vh+CwB1R0E9XYJLqQcAAAAAJ8nlcsvj8cJFTP1AKDuqK+HZVDqAQAAA
MBJcrnd8nq9v6h1GOmHgDUHeVuj9kRalGkRagSlHgAAAACcJLfcrfQCt2Wbew5wBWx5x4cN8VVp3JFjN+0
8Ph/LbwGgDvP66merR6kHAAAAACfJ4/F/FUuade/VzoBQCByVtPp+pR6gEAAADASSwJ3hqacz9WxmBwAAAACAQHVo+e3hB2VYjfr5AyQajiRLriY531ALb5rZUYDj5vF6ZDEMRUVGLruutf5XUiNzQtUgSj0AAAAA
AOElujOdez+EHZVgo9RDAxgUv1X2WNxXuKTQ7CnBifv1SnJ/zhxv1c1MESj0AAAAAOEkud9UHZbDPEQJRuFGi5
51TNcS7UKKXRn1iqZ/1V/18VwAAAABQC9xuj3w+nwyDmXoIbIPtqXra/qrivPvNjgJUP4vV7AQ1glIPAAAAAE7
BHws9SaqfPz6iPrLJrUed7+si7yxZ6+kJoQClHgAAAACgEqvVIp9Ph83WY6YeAkFXa5qet76glr59v+1FBtRHB
qUeAAAAAOB3LIZFVUzUo9RDnXen4yvd4P1AwZb6eYAAUAkz9QAAAAAAv2e1WmQYhrxen6zW33b3OsoiM+Nz8DzV
aHDRZ3bSZ36hoOGw0sxPUCEo9AAAAADhJVqtVhmHI5/Pq9w2JlZl6qIMuD/5J9/jeUrjKzI4C1K6QGLMT1AhKP
QAAAAA4SVaLRRaLIa+vcolHqYe6JNoo0vPBr2uQlrN3Hhoei01yRJmdokZQ6gEAAADASTq0/NYi3x9KPYfFa1I

ioLLhQev0iF5SnArMjgKYwxmjKjc/rQco9QAAAADgJFktVlkMg1IPdU6wXHrC8Y7O982RpX72GcDxCYk1OOGNo
dQDAAAAgJP0+4Myfo9SD2bqZdupZ62T1ExZLLcFQhuZnaDGUOoBAAAAwEmy/G9PPWbqoW7w6v+cn+sqzOwFGR6
zwwB1Qz09JEOi1AMAAACAk2az2WQxLPJ6K5d4Tko91LKWlv2aHPSCOvp2/P4gZgAsvwUAAAAA/FGII1g2uOOut
7vSdWbqoTZd75inO7zTFKoKs6MAdU8Iy28BAAAAAH8Q4nQoyGaTy1W51LNbfLLKJw8bmqEGxRqFejH4FZ2u1cz
OA46kHs/U4397AAAAADhJNptNoSHOw2bqSczWQ80aHbRKc+x/P1ToATgyDsoAAAAAAFQlPDxUmfsPHnbdYfWo2
Gs1IRHqM6fK9bRzqkZ6F8jCRFDg2DgoAwAAAABQlajwcGbqoVb0t23TfyyT1Nh3QKzsBo5TPV5+S6kHAAAAAKc
gKiJM7ipKvVCrx4Q0qI8s8ugB5ye6zPuF7AZlMXBCOCgDAAAAAFCVkJAQVTVtKpxSD9WgrTVDk20vqK1vD7Pzg
BNlWKTQOLNT1BhKPQAAAAA4BaFOR5XXIyj1cIpudXyn27wz5DRcZkcBAlNUM8kWZHaKGkOpBwAAAACnIMTpkOQ
77Hq47fAlucDxSLTk6cWgl9VH6yWL2WmAABbXzuwENYpSDwAAAABOgdNxaKae1+uVxfJbA8PyW5yMi40X6f98r
ylKJWZHAQJfo7ZmJ6hRlHoAAAAAcAoiw8MUHByk8ooKf8EnSU6rV3bDK5ePqVY4tjCjVM8Gv6lhWsTeeUB1qec
z9fjTBQ2Wx+PRey88pZuH9NOl3VrqlnP666OXn5PP99vSidLiYr3+OD91w6BeurRbS/111CB9+/70oz7vku++0
d1jR+iKPu11WY9W+vuYoZr/+ceHjXno2vG6ql8njW2frJ2p6w97nrcef1BX9euoGwf30o9fzqx07+fZX+qxP19
5Cu8eAAAA1SUyIkyOoCCVlVccfo8luDgOg+yb9L39zkOFHoDq06h+l3rM1EOD9dnrL+nb96bp9ideUErrdtq+f
oOm//MOhYSFa9SV10uSpj7xoNYvXaS/PvWi4hunaPWiBXr9oXsVE5+gPmcPr/J5wyKjNPbPf1Xjlq1ls9v1y/z
v9dI/71BkTCP1OHOwJKmstETte/XV6ee01iv33XXYcyz/4Tst/PpT3ffGe8rYvUMv/9/f1f2MQYqIjlVxYYHef
e5JPfDW+zX2uQEAAMDxiwwPk9PpUFlZuRT5h3s2tw646u8m7Tg1Nrn1sPMDXez9Rjbj8HOZAZyiOJbfAvXS5lW
/qM+Q4eo1eKgkKb5Jin76+jNtW7f6tzGrf9HgMRerc7/TJUnDLrlccz54W1vXrj5iqffr2F+dd+X1mv/Zh9q0c
pm/1Bt8/kWSpOy9aVU+R/qOrerUt79ad+mm1l266a3HH1D23jRFRMfq7acf0fBLr1RccpNTefsAAACoJlarVXG
x0dq28/Dv7ZiphyPpbN2rF2wvqJUvneW2QE0IS5QckceF8BYfosGq12P3lq3eKH27dwuSdq1ac0h4m3g2b+N6
d5by3/4TgezMuTz+bRuySLt27VD3QYMOq7X8Pl8Wrv4J+3buV0de/c77mzN2nXS9vVrVZSfp+3r16qirEyJTZs
rdcVS7di4TiOvuO07E3iwAAABqVFJcnMrKyw+7TqmHqvzN8bU+sd6rVka62VGA+quez9KTmKmHBuyCG29TSXGh/
jJyoCxWq7wejy6b+A/NNH2hf8z19iV+7WzcO6iWrzSbDsO0gnnJVVMhiseqGBx4
77iJQknqc0VgDR1+oey4eqaBgh25/4gUFOOPO2oP36rbHn9e3703TrHfeVHh0jP78ONNq2qZ+7xMAAABQ1zWKj
aqON/Ovoin18DtNLDmaHPSiumszU2yAmlbP99OTKPXQgP086wv99OVMTfzPS0pp3U47N23QW489qNjj4BJ11wTh
J0jdvv6kta1boHy9PVVzjJtq4fIlef+ifio5PULf2Fk2aHLc09mktuv
1OX3H6n/9cfTn5GXU8/U1a7TZ+8+oKe/eIHrZg3Ry/e8xc9PfPbk/9EAAAA4JQ1io6ST4dWahjGb2spo2xuWQ2
vPJyA2+BdFrxI9/qmKFxlZkcBGoZ6fvKtRKmHBmz60w/rghtuMrAP79mrmay/qrAvGqbysVO8+/
4TufnGKf9+95u06atemDfrizVePWupZLBYlNWshshSWrRobP27tiqma+9eEK13u/t3bFCf/M/E4/fPKeOvQ
+TZExsTr93D/ppf/7m0qLiuQMCzup5wYAAMCpi4mKVJDNpgqXS2S2MYTGkWJtb2RyWOWBFGcV63vG6BvuWs
XceUJsaQKnHXxehwSovLZNhqfy/gMVilc97aNmEx+2W2+U6whjvCb2Wz+uVu6LipHL6fD799957dPU9D8gZGiq
v1yuP2/W/jIf+6fV6Tuq5AQAAUD1ioiIV4nSotPTwfrk1oRP/c5DhR6A2sXyW6D+6n3WOfrk1
UmKS2p8aPlt6np90fW/OnvseElSSFi4OvXpr+lPP6ygoGDFJjfR+kVf68sP31Hvv/7dnZ8PuWb8aaa5hSfz17
SZv73RbXq3FUJTZvLXGhlQvmasEXn+jGjGB8x73P6YwL5LgI8zhYJMtKVk0lSf7Dlq1ajxvdQL/YwS
VL7nn304eRntGX1Cq388QfdtRNhQz5M5hnMc/MDwIwLPFrfpR0UdaneeXTdCZddJazXMqKvZR20XbDq6rs
13JnYq8bh98gc+EOeb0YZ6nvvD6hJumA/9ltMDhAIVPrFlOE7wZz2Rgdxkr/Ldgx6C/+x9KzKW7P6GUXwXwPPn2t
x9aGkgqRsGthwb7c/gw66Aiie9BY+d7eS6UaZRfGs8lnbwzpWxfyO8aC7bn0Y/QXWeaXcWK7PE2n7PaoVv7Sa9
8urVevVtBZzSa9cDOPuD6tjg9ymFfSzrmvAsJXfZ8Ra0wx+vuyPMUP5W8Wb8t2GgRzGgiOZfFkhu4BBUJiCkdK1X
zHPoVxXvv7YZ6s+mqVfH/nSQW6bUJmS77VtfYAqmXV1ysupA5TUIn4Jm5F2xKy0zCO2XWFu+9XnrmGx3fN1kCCoSA
m8aZvqeGdxRqhLyzpx3vx5z3ZfTvO+1qNNOd6+bv44kZwdfmWZH6vP8LWBN9/hu7FXB9u0x6T0jF6AC2c2+Zb9ZJG
mGPxDsLU6aikpIQHzHdvKMpvYWoSGhsbeOYSL1oNxfvr6lq1I4U0w7Y4sXs3u0FlHfaShfNv3n4N/ZiS/GcMlVvfSrv
K0uFHQVd4AYSLUMfgC7Vm3kNTvpNyEpSDXLVrTRRZG/+8bD+qzfdzoSYtT7dMlxAwRlg30T5UV+xJBy7BW0qrlzWGD/
+u+uqd5ynTZxHOw+sfRbqhyKdI5iMwcdI+hWdjPDR3vRKt06NdSXZ/MqZ/dHYqr0fI3tt/NnKOmqddCd+h1ghHB6H
dSwoDm01zv+iOxWZdTR48QTFfXw/T8MAwK1FjrvZb2Zo+r/0duFZ/W+C0j9E9qjOQMo8I+NeWR+mmMIyUEEpp0zdQO
8eLAOkX3Hzn3O0yvaMnqg3rpVj+8wGoKaqAK9qeLmX97uX8JTVNJoVaqK3doK5AkWSDvkTtcbRmc2v+ZfCTNl6+yUv6E
GO0xkD1iwXJH/Ssq+XM7wfKzBSgu9QkrbT5nS2hTXhPaGXYLfVsGg8w4Ed3ERNG8u6KiOIwhC3a58bTW9ad5Xw2b8W
3x4eWwOHvYTdiSQ+nMbZ3q8Ur9Vt+sP9hjG2qs3VcGMrN36c2VMsifFfmLkq1GjFh2ZcJHrGFPaZh1ZeWu7WH3OqFp
1LQW3+PK3GuRZfS2hj5oZVIPDSIIUV0L/bvmw6RxyKnyw2XPSU6zbuNfxpYlrYZrdOaRm0/g5VaD2cZjnhEzGZmYD
1fJuAgq0kv04utfBRq04Y8Crvbmd1OJiIm3d9HbSO/M5RlIdaXxjKVd20J94y6Kx3DEu8SNr3RqG7AR0cf6mb8AH
JGEaIUe4uPRGkpM7dB9LmJ+cDj5SXvWp4bJxdmakBwrxUQv1AyT0AE0rtwBYuYkFzE+ywkwqPm4y0vZWyrd0jZJ
NIJqxTuDPM6/o8RC+YpcX7s/GIiZ/xZcPi0Q2EBPFeVaAJD7K/4gKqHt8LoSvSa2+pqqYbT/mkw4pmXaLCmkqmzE

4KDgxUREVHpAwAAAADqmpZNm6i84vACL8ruVgh7rQU8pyr0ovN1vWA8rVhLkdlxAFSlgeynJ51gqTdkyBCtW7d
Oq1ev9n/O7t1bEyZM8P+73W7X3Llz/Y/ZvHmz9uzZo/79+0uS+vfvr3Xr1ik7O9s/Zs6cOYqIiFDHjh2r6W0BA
AAAQO1Lim8kw3L4YRmSlBRcbkIiVJd+tu2aE3SXRvvmyWKYnQZA1QypaX+zQ9SaE9pTLzw8XJO7d65OLTQOVLG
xsf7r1113nf72t78pJiZGERERuv3229W/f3+ddtqhTQqHDRumjhO76oorrtBTTz2lzMxM/etf/9Ktt96q4ODga
npbAAAAAFD7khPiFOJwqqSO7LB99RoHl2t7acgRHom6yiKP7nPO1OXez2U3Di9rAdQh8R2kkBizU9SaEz4o41i
ee+45WSwWjRO7VuXl5Ro+fLhefvll/32r1aqvvvvpKN998s/r376/Q0FBdddVVeuihh6o7CgAAAADUqoRGsQoPC
1FR8eGHZSQzUy/gtLFm6kXbJLX37ZKYnQfUfQ1o6a1UDaXe/PnzK/3a4XDopZde0OksvvXTExzRr1kzffPPNqb4
OAAAAAANQpdrtNLVMaa/najUqMb1TpXpjVq0irW/meap9bgRpws2OObve+oxCDQO6AgNHASjOO3gYAAACAatS2V
XO5XO4q7zFbr+5LsOTpQ8fjukdvKcRCoQcElAZO8q1EqQcAAAAA1apFSmMFB9lVWnZ4gUepV7eNDV6u2bY71Vf
rzI4C4ETFtJTCE81OUauY9w0OAAAAA1ahp4ORFR4Yrr6BQTkflwwCTg8tlyCcfG7TVKSFGmZ4LflPDtZC984BA1
cCW3krM1AMAAACAauUIDlaHNi2VX1B42L1gi0+xdpZO1iWD7Js0137noUIPQOBqfY7ZCWodpR4AAAAAVLO2LZv
L4/HI5/Mddq8xS3DrBKs8esz5rqZYHlaSJcfsOABOhT1EajPM7BS1jlIPAAAAAKpZi5RkhTidKi4pPexeM0eZC
Ynwe52s6fo2+F5d5vtKNuPw4hVAgGk9VAoKMTtFrWNPPQAAAACoZoOT49UoJkp5BYUKC638g2Z8kEthVreKPPw
4ZoaJjm/OZ+/7chhVn1AMIAB1PN/sBKZgph4AAAAAVDObzab07VqrsKi4yvvNma1X65ItOZrpeFgT9Y4cFgo9o
N6wOaS2I8xOYQpKPQAAAACoAa2bp8jr88k3mZ3CFMz3BgAAAIA
a0DylsSJCQ1RQVKyoiPBK9xKCKhRi8ajEazUpXcMQaZToOcfrOtu3lCktQH3VQJfeSnxZAwAAAIAakRgXqxZNm
+hgTt5h9wyD2Xo1bYh9g+bY/36o0ANQP1mDpHbnmp3CNJR6AAAAAFADDMNQzy4dVFpWJp/v8BNWmzsPPxkXpy5
ILv3HMU2vWR5TvCXf7DgAalLLsyRHhNkpTMPyWwAAAACoIR1at1B4FOAo5LB6VsQS32nSz7
dYL1klqrgzJMDsNgBrXgJfeSszUAwAAAIAa0yQpQc2aJ0tAFUtwLYbUjCW41cSrfzg/14eWf6m5kWF2GAC1wWK
X2o80O4WpKPUAAAAAoIYYhqFeXTqopLTqbgWbtWYJ7yppZDujL4H/rz74PFGzxmBOHQG1pMVByRpudwlSUegAAA
ABQgzq0bqmwUKeKiks0u5ccVKEwq9uEVPXDNY4F+tp2l7oYW82OAqC2dfyT2QlMR6kHAAAAADWoaeNNEpSQnVrk
E1zCkdiGHl304uhijSG87ntED+q/CLOVmxwFQ26zBUgdKPUo9AAAAAKhBFotFPbt00E+gqIqRPjDlKZFcL2vxF
ZFBa/Sd/e86UyvMjgLALJ3GSCwEx6HaffAgAAAEAN69C6zUcGhzWoRGsereseOR/HpDEm8
U6APHE7pXbyrE4jI7DoA6xZB6X2t2iDqFUg8AAAAAaknH1i3VJDlRWfsPVHm/Q2hxLSeqOy4M/kXf2v+uflprd
hQAdVHrIVJsK7NT1CmUegAAAABQS4KDgnR6786yWq0qKCw67H6QxaO2zEawlqJwW2LpPvCrBDkqjEtakDMGJ3T
OB2QchlIPAAAAAGpR907t1KVDG2Vk7ZWviv4q59Wq7Qc7rJc0pOCX1AgmzfyzwaAXkCS93Y5G+n5Xx7V9dD5t2gYmo
TnDqFoCgnZTM9nA8+nOv9nu6klzZnWGNTxvebJV7b10fYSMrXJVLn/8lRP61Jm2/Qc/xfeWIpXj2qO8ZR8ryFq8B
2drHZcYSnfNJJUnvIK1VMp4YgZ8aR4LOaaZ8kM3FcPtFXTK4lj4g5Bz3G+KRBBpEFeJfWBAU8DkKUrDKl7i7Fdl
Xf2KCcXcmMcB7rtLtCI7BWnvOSZS8Jvr1iaTvMdgr1rIu6dAFHAfN4ANZI0q56aMhQdCSD9+rR8v2UI6BDNKX+ZbI
2HCV2ph5xjrcClOSeyP6tEXNl7X3Y05TzAUGaUZ5I0zQDrnwPUJ8Ucg1uKwBxGZqtbsmo1dtNfLpKfy7Rd+5GzNFz
1ncDYqVQXRHWUOCmTM04DJaJD3s8bxlBzDz6u+8z9VbMrsT5jSP1VYtIyD1mkl7R
aOyOnLYmowtAb9VHIV5WLhoVHXLK0qNz5V2j0LKMxLpzMcVsZcOEGIgPeBwlVlP3c+ZTmeH0bbrmsbP2eq+6Nuu5HOnBFHFVXfz+qvR9VUPBZ+Ssd1bLqWRXDB8eiSaUWjcl0
nUZ3QK3P8NUXmfMEDmrnefR7YTPC9ejY4ntepHSXqnqeH5sk8j1Fy0e5PfXU3DzWxWXFsb2Q6NFa5o3mTgmyF23dhZ4eRrxjaXdUwGuM06T7T3e7wmKf4XWrBmq07d6u
...

```
mM8+7QM6wMGXvTZMkvf30Ixp+6ZWKS25Sc58cAKiCxWLRiMEDFOJOHvEk3I6hxQqzums5mRQsl553vKmXjSfUy
EKhB6AGDJgoBYeZnSKgUOoBAIAa0a5Hb61bvFD7dm6XJO3atEGbVi5Tj4FnS5JcFRWSpKDgYP9jLBaL7EFBSl2
x/Lhfp6K0VB63W2GRUZIkr9erFfPnKrl5Sz103aW65vQu+se4UVr6/Sz/Y5q166Tt69eqKD9P29evVUVZmRKbN
lfqiqXasXGdRl5x3am+fQA4KW1bNtNpPbooPSO70nYFv7IaUs/w2i3Vett26vuguzRG38tiHHs8AJyw2NZSv5v
MThFwKPUAAECNuODG2zRg1Pn6y8iBGte5qe68YJjOu/IGDRx9oSSpccvWapTcWO88+7iK8vPkqqjQp69P1sHMD
OXuzzru13n7mUcVHZ+grqefKUnKP3hAZSXF+vT1yepx5lm6f8p76jt0hJ6+/XptWLZYktTjzMEaOPpC3XPxSE2
+d6Juf+IFBTtD9NqD9+qmB588tGx4xBn656V/0p6tm6v/kwMAR2AYhoYN6q/oyHDtz8mtckwbZ6ni7BW1kMarf
zln6l3rfUqxVH2ABwBUi1HPSLbgY49DJeypBwAAasTPs77QT1/O1MT/vKSU1u20c9MGvfXYA4qOT9BZF4yTzW7
X3ZOm6OV//U1X9esoi9Wqrv3PPDSTr4rZKVWZ+dqLWvTN5/r39I/9++X5vF5JUp+zh2vO1TdKklpO6KzNq37Rt
+9PV6e+/SVJl9x+py65/U7/c304+Rl1Pf1MWeO2ffLqC3r2ix+0Yt4cvXjPX/T0zG+r81MDAEfVtHGSzuzXU19
8N1+NoqNksVSei2EYOoCoPH2+PO4+1czUudbWTE2yvaiOvp2qoZcAgEO6XCy1HGx2ioDETDOAAFAjpj/9sC644
TadMWqMXmrXroMHnX6TRV9+gma+96B/TqnNXPfPZ95q+fJPe+Gm17nvjXRXl5Sohpekxn//zKa/o09df0n1vvKf
m7Tr6r4dHx8hqsymlddtK45u0aqMDGelVPtfeHVu14MuZGv+Xu7Vh6c/q0PsO0RcbE6vRz/6QdG9eptKjoJD8LA
HByhpzRT/GNYpS5/2CV9xvZ3eoYWvUpuafqJsdcfWG9Rx2NnTXy/ADwK58jUhr+mNkxAhalHgAAqBHlpWUy/jC
7xGKxyuc9fBZeaHiEImNitW7Erz+u+12eodZdule7Zg4LUunM3pf9vL79f7du1o
8rDL3w+n/57/z26+p4H5AwNldfrlcftkiT/P71ez7HfMABUo4RGsRoyoJ9ycvPk8VT9NahXeKFCLNX39SnOKNT
7jid1r6YoxOKqtucFgCMxhtwvhcWbHSNgUeoBAIAa0fusc/TJq5OV2ZunLVqqTGbiIRGm9Vv9Vv3NG+Mf8PPtLr
V/6szLTdmvZ3Nl66Nrx6jNkhLqfMdg/ZtI9f9E7z/z2N7ifvj5Z773wtG559FnFNU5R7v5s5e7PVmnxbzNWzr/
uFvO86wvN+XCGMnbv1DfvvKlf5s3RiMuuOizn9x+9q4iYWPU5e5gkqX3PwlIrIGP
kMAcHSD+vdWk6QE7c2sej+7IItP/SLzq+W1/hSOUt/a/6bTtKZang8Ajqlxb6nXtWanCGjsqQcAAGrE9f96RO9
NekqvPXSW8vCg4eVHR8gs655ApdfMsd/jG52Vma+sSDyj94QFFx8p8RRLGcZvfw/57XvT5XV6
D9/vaHSuHG3/s2/R16/c87VjQ8+oZmvTdabj96n5BYtddek19WhV79Kj8k7sF+nvPqCHvnMN1+n
Rm65UZGysbn/iher6lADACYmKCNe5Z52hKe9/qtKycjdh28i38pZpi0lZUovd5zUazhVrmecUzXCu4CTbQHUG
p9hlTH6ecnCXLNTYfiq0ie9jisoKFBkZKMVERFR668/c3NGfmasd0OBAIoVoqK14r1foCGAEBrA5ywRQW2Yj4uWmcqYTa0Vat24hwzi8ect3WzUz016eEzzRYoB9i542XlSypep9+wCgxpx2qzSCvfRO0ZUoAAAAANRRQ
UF2nT/sLIU4HTqYm1flmEibR13Djv9AH6s8etj5vqZZ/k2hh2rzxMJyGf8u0MTZZf5rZW6fbv26VLFPFSrssQK
N/bBEWUXeoz7Pg/PL1H5ykUIfK1DOkwUaOr1YS/e6K4159MdynT6lWCGPFijqiYLDniOn1KfR75Uo7LEC9fhvk
VZlVN578tavS/XMz+Wn8G5xKnwRydJZ/zQ7Rr1AqQcAAAAAdVi7Vs111ul9lJ194IiHZnQPL1SE1V3lvd/rYN2
nWcH/pyt8X8hmBNyiLdRRy9M9+u+KCnVNqFwx3DG7TF9uceuji51acHWo9hX6dNEnZ1MHjO+oxjjXxrmm/bjIC5kFVmyp8iN6bu8hDYUvGhgRsp8U2Wip2k6jh0uqeTD0Pa+8CDG5DjhTYnh9nbWO6LT5t6gx1tMoeSJa/bl9hOa+gD4AWl/TLQRtvubf4VPWlRiH6pc6G9h6KWDkdnyMSQ6IimIbPtvlxHp/zFHpJoodXTt0aRcEvCl3hXJ1CkTSY9pnNDfdqN9o0XHD08lFN3Ig6qz2XJ6qVRTeDVxrGKbJ9mz8ANh6Npe3V+j8XLHqYH4bTNJ3PP1qhsa+f7m5vj3PvZDM95x7Y3XRvlqJgWhgWJ3MJSQGa5x3Mbbqjx0e3vN0rh3TzOnKxfiSC/q33e/7Nt6Gqf2d2Z1J5oNpyFy6MT08Pp3z9L5rSi6R3BF8TBfGRsOVV99M2YfOe2C9iS27b+bQMy1d7jz1T1Gdzr6dj6v2BDpzNKo4S+SrhKszSkk2Yn4rxPKLUYdX5AKiBuHiQxp9bC9a08W1ff89w/d3bXfFi4iqqIgLHsX3X6PPdbeXHJmJK3vb/bOXW1ue9tmEWE9DWr++qMGGrW82GKXfCqWnYomdxXyQzHvGxAPr3JOzPnJMf9e4Xo1SvkiD6MJvDyHKhAFQoK6C5L6q6ZRiA6o8N55Zy+xfxDjpGd+F+s6mN9g5ViDzwoJnXM7k+mGFKH6OJKuQ+OiRzEVhG+4HGfWl2IX0NaItK2WxWoE9FOPlE4jtk1CCUNDBTEqiWMDYIhYPMwrj9b+mC8xTiaeZLM8xvYHPN4mvzVtq4pEOVa5VvhWtVt0e8yxUmUDqzoJCp+1/Jl1Gol8GuWuMYeCthhDyHhB/ymmi+k0FmTZk7ltlz/c06m5ldu9+7Hl5ZY2F4Gs4yc2LNoR+82+TLR6ZzQ8nP0Xr+PNG4wJ0v6N1cCZHhm0eXUzYH2mcAcC5YSc2mx2tq1EbaXz8yl3z20vVkPVUnzWY0VmYXw1I03z+q4Kly3fBX7Jd6e7N3MdEUR+Sd2+Zd9GUpmVKTRSUfcDp6CwoHjJM1lE0lqvW3b+KcGXw0zj5bnCGoTDSkb9c7dq8Gt73w3ZNv4c5YZJwT+ylCk1LDQwz+iCKz9HvHMHS8lDvHVhYkm2WTqiQ5LP+r5kzZbbzzEbqJ9gwUAXO5ENqRR9YEp0KFgC1M1fCjeNhbRKqZ8T3y9NxXL57NI2xrt+Cl4XDSuAFFDp9Jq+7KqPvR7XSeF2rF9qRIQhD4oYGJ0o9mI6R0OfS9IzVhj4XEYKXk0vfJfEKM8M=
```

gMlPsU6DfVpbNVPl4SqU/xv5dv988o1bc1vy3F7/PfQqc/zrgrR40a/1RzfbnNrW45Xb1/wWyF4W98g/bLPo35
vFKtvY6seGFw5D6qPK6at7MMfNjtGvWb4jnRsUh1WUFCgyMhI5efnKyIiotZff+bmjFp/TQD124XtksyOUOfwt
RZAdTLz66zZ37ui/iotK9Nzr03XktkfKaTioFKSk9SlRy/ZbJXnbpxWNFstyzn9FkDt8Vidst68UGrU2uwo9Rr
LbwEAAAAgADkdDl0w7Ey1ahSsDm1aqkeffocVepLOS+jZKrRE1X5AAA2W8adJFHq1gFIPAAAAAAJUh/YddPn1t
yvYblVZSVGVY9xGkH4K/5Ncsld5HwCqU3nnS2XpNs7sGA0CpR4AAAAABLDupw9Rx54DlLlnh7weT5Vj8mxxWhx
2rgJu7yUAAaUssrWCx7xgdowGg1IPAAAAAAKY1WbToD9dqvgmzZWZtuOI4/YGt9E65+m1mAxAQ+K2OBR85UeSj
VO3awulHgAAAAAEuMiYOA0adYkMw6KC3ANHHLfeeZp2B7WtxWQAGgKfJO+fXpIR29LsKA0KpR4AAAAA1AOtu/R
Wr4HDlZO1T66K8qoHGYaWhI1QjjW+dsMBqNeK+t6hoO4XmR2jwaHUAwAAAIB6wDAM9T9njFp37qnOHZuPuL+ex
7Drx/DzVWqE1HJCAPVRTrORCh/5oNkxGiRKPQAAAACoJ4KdITrnomuVkNJS+3Ztlc9X9dEYJdYI/RT+J3lkreW
EAOqTA+EdFX3l22bHaLAo9QAAAACgHomOS9Q5F10tZ1i49u/bc8RxB+yNtTx0aCOmA1Cf5FrjFX7d5zKsNrOjN
FiUegAAAABQzzRt3VGDR18mdOX5UQ/O2OHorE2OnrWYDEB9UKQQece/q+Ao9uc0E6UeAAAAANRDnfsOVN+zRyt
3f6bKSoqOOG5VyCBl2JvVYjIAgazcZ1PuOS8otk0fs6M0eJR6AAAAAFAPGYah/sMuUOc+Zypzzw65XRVVjvMZF
iOMOO/5luhaTgggOHh9htJ7/UMpA8aZHQWi1AMAAACAesseFKSzL7hSzdp2VvrOLfJ6vVWOc1kcmh8xVsWW8Fp
OCCCQ7GhxuVqMvtPsGPgfSj0AAAAAqMdCwyM17OJrFRufrMw92494Im6xNVI/RFysEiOOlhMCCAQ7YgarxeXPy
zAMs6Pgfyj1AAAAAKCei0tuqiFjr5LNHqSc7IwjjiuORuuHiItVZjhrMR2Aum6vs4MaX/+urDZOuq1LKPUAAAA
AoAFo1bGHzhw5TqVFBSoqyD3iuAJbrH6IuEjlhqMWOwGoq/bZmiryuk8VHMIs3rqGUg8AAAAAGogeZ5yjXgNH6
EDGXpWVFh9xXJ4tXvMixqrCCKrFdADqmjSjiexXzlR4oySzo6AKlHoAAAAA0EBYLBadOWqcOvc+U5m7t6u8rPS
IY3NsiZoffqFcstdiQgB1xS5vsrzj3lZc0zZmR8ERUOoBAAAAQAMSFOzQORdfq/Y9T1fGrq2qKC874tgD9sZaE
DFGbrGPFtCQbHcnquS8l9WsQ0+zo+AoKPUAAAAAoIFxhoZpxCXXq03XPkrfuVmuivIjjs22N9VP4efLI2stJgR
gliOVCTow6HF16DPY7Cg4hhMq9R5//HH16dNH4eHhio+P15gxY7R58+ZKY8rKynTrrbcqNjZWYWFhGjt2rLKys
iqN2bNnj0aNGqWQkBDFx8frrrvuktvtPvV3AwAAAAA4LiFhETr3OpvUpnMv7d2xWa6KiiOOzQhqroXh58nLvBC
gXttUnqC9vf+p3mefL8MwzI6DYzihr8gLFizQrbfeqiVLmjOnDlyuVwVwaNmyMyot/22Djjvu0JdffqmPPvpvpIC
xYsOL59+3ThhRf673s8s8HoOaNUoVFRX6+eefNW3aNE2d0l339/9rOrAAAAAMAxhUVE6dxdL/6xWHborfccmuV2
uI45ND2qtn8NGyit+0Afqo9TyRGXO/acG/elyWa3MzA0Ehs/n853sg/fv36/4+HgtWLBAAwcOVH5+vuLi4vTuu
+/qoosukiRt2rRJHTpOOLFi3Xaaadp1qxZ0u+887Rv3z4lJCRIkl599VXdc8892r9/v4KCjn26UkFBgSIjI5W
fn6+IiTjX/SZm7OqPXXXBFC/Xdi006T+iK+1AKqTmV9nzf7eFTge+Tn79dXbL2n31g1KadVBVtuR99BLKd+i/
kWzZBOrrYD6Ym5N5orL6/p8Gjr7sqP//o245pbnT+fn5kqSSi/1AKqTmV9nzf7eFTge+Tn79dXbL2n31g1KadVBVtuR99BLKd+i/
kWzZBOrrYD6YmN5orL6/p8Gjr7sqP//o245pbnT+fn5kqSYmBhJ0ooVK+RyuTRO6FD/mPbt26tp06ZavHixJGn
x4sXq0qWLv9CTpHOhDh6ugoEAbNmyo8nXKy8tVUFBQ6QsAAODhlxbcVvMix
qrccNRiQgA1ZX15krJPu49CLwCddKnn9Xo1ceJEDRgwQJ07d5YkZWdnKygoSFFRUZWOTUhIUHZ2tv+YQwu9g/sP
7qvOpEmT9NBDDx13NoAAUHGPqbHqUl75SZyUY+bYb0JjeaJ25oHZwnPbAUnMIDcJu+/dpm4Qsg7yG8pqscsX/d
/vVeVxx9/XJGRkf6Pl4JSUk40NAAAAAKhCdKMEjbzsZiU1b6292zzfj6/Eccex+exPNiRyvIguzT4FAtq68sQ6ef
r/OPO9SCr0AdNKl3q233qr169fr/fffr/fffr848Vbr33nuVn5/v/0hLS6vx1wQAAAACAhiY2IVmjJtyihCYtDhV7Xu8
RxxZYY/Vd5GXKscbXYkIA1WVpWQvlnn6fzhh5CXvoBaiTKvVuu+02ffXV5o3b56dlaVG
p+VlaXExET/mD+ehvvvrr38d0OfBwcKiIio9AEAAAAAqH5xSSkaNeMvTJLqL6PvEQZ9ma1mBDAqfD
IolnFXVQx8P80YOQ4CrOAdkKlns/nO2233aZPP/1UP/zwg1q0aFFpf8+ePWW32/X999/7t23evFl79uxRenq6JKl
Kl///5at26dsrOz/WPmzJmjii2IgIdezY8VTzAgAAAACgGiQ0aqaRl9+shBR7GwDbEZbZ1+KBqdUCK0bFi6kW6Ph7P5
AJ6NMQfqwqK/Ch9yhASPGUugFuBMq9W699Va98847evfddxUeHq7s7GxlZ2ertLRUkhQZGalrr71Wd9xxh+bNm6eVK1
fv3zRv3
jytWLFC1lxzjfr376/TT60NkjRs2DB17NhRV155pdasWaNvv/1Wf/3rX3XrrbfKbrdX/x0CAAAADADi4+t8hAAAAAOCEJTVtpTH
XTFSzNp2O20d3uqKspGyE1jlPq8WEAE5Eri9COwtOV9OhN+n0YRfjImlsls1NRB5zQf8FXXnlF+fn5G
jx4sJKSkvwfH3zwgf+Y5557Tueff77Gjh2rgQM3KKrIKSkpmvvf/78uv/xyvXXfXIBa3JVX3jcvXZrooYe
q71OBBAAAAAE5ZbEJj/emqv6hd937at2urSoK2jzgr1u/bkIsJKSbp+XrFgvMvi4yj/e7aYe2Jl61K5mtz4wf
D6fz+wQJ6qgoECRkZHKz883ZX+9mZSav01AdRvF7ZLMjtCncPXWgDVycyvs2Z/7wpUh7KSYs39dLrWLpmnmPg
khUfFHnV8csUODSj8SnYdedkugNqxtVcP1R0O+kjLlbfs8+TYVC61xdUswAAAACAo3KEhGr4uOf+j5yj+4X
znZR//Lt31BLTU3cpyKLeG1lBDAH3llaG5JBy20DdCIS/9MoVcPUeoBAAAAAI7JZrdr4OhLN2fhNLdL6Vf/9jn+aP
paAu/cmyJmhV5hdLtLY44BkDNqJBdH+d31+7YQTr/qr+qQ8/TKfTqIUo9AAAAAAMBxsVgs+kfHUWbq2rRwz2kLGRZ/5AbZbDbt2
7VVXq/3iOMrLE4tCL9Aq0LJcagwTcvpJbUdpguuuNjZtLsNhYLr1PkMhYeH
67sMp2rs9VY1btpfVaj3SA5Tq7KsDtmQNKPpaId6i2g0MNCC71Vif5bZV637DdfaYK+QMDTM7EmoQf1UCAAAAA
DhhLdp305+u/qSU1oqbesGuSrKj3zqVzc+ur/qsSU1oqbesGuSrKjzp+v72JZkVeoX325rUTEGhA3LJqnquHPiroqV4jLtfwcddT6DUAlHoAAAA
AgJOS1LSVr/6r2rZobvd6obvSd2xWUd1rWUUHeUceXW0IOP/xCrXaeIa/Y3wuoDrmWWL1dOEDr1FbDL7lBA0aMlc1uNzsWa
gGlHgAAAADgpEXXgPV/9VvQadq0333UAzmEXAqUo066vX9PeSJbas/XXm7uvQ

```
bxIEYDQilHgAAAADglISEheuci67RiPE3yh4UrLRtG+V2VRz1MfvtTTQr6gpl2JvVUkqg/ig1QvSt/VzNTE9Q4
zZdNeba09S8XRezY6GWcVAGAAAAAOCUWSwWdT1tsOKSU/TDp29r95b1iktuqtCIqCM+ptwSonnhY9W5dIk6ly6
WRUeZ4QdAkpRub65virvpYE6FuvYbpLPGXKGwo/x/hvqLmXoAAAAAgGqT1LSVLrju7+o1cIRy92dq/749x1yOu
z6kv76LvEw51rjaCwoEGI+sWhp8pt7OaqdyilPDLrpWIy+7mUKvAaPUAwAAAABUq5CwcJ1z8bUafskNstpsStu
WKrfLddTH5NgS9W3k5VoVMlBuFpUBleRZY/WJMUrf73WoaZvOGnvj3ep55jBZbfy/0pDxXx8AAAAAUOOsFou69
T/Lvxx3z9YNimvcTKHhkUd8jM+wKNXZR3uC2qhv8RwlufbUYmKg7vHIok3BPTTnYJI8MnT6OWPUf9gFcoSEmh0
NdQClHgAAAACgxiQ3a60Lr/+7fvz6A635+QeVFBWoUWKTo57QWWyN0ryIi9WifIN6FM+Xw1dWi4mBuiHTlqKfr
f21KT1P8cmJGnjeJWrTpQ+n28KPUg8AAAAAUKNCwiI07OLrlJjSUgu/+VBp2zYqsWkrBQU7jvq4ncGdtM/eQj2
L56tFRWotpQXMVWKEaWXIQK3MC1dpUYE69z5TA0ePV1RsvNnRUMdQ6gEAAAAAapzFYlH304coPrmpfvz6Q+3ct
EahEVGKTWh81JlH5ZYQLQ4fqV0VHdWneI7CvAW1mBqoPV5ZtMXRQytsvZW2Z7dCIwwNHXuNup8+hL3zUCUOygA
AAAAA1Jrk5m009oY7NXTs1bJabdq9Zb3KSoq0+biMoOb6OupqpTp6ySuWH6J+ybI10azIKzS/orN279yhlNYdN
fb609Vr4HAKPRwRvzMAAAAAALXKHhSsPoNHqnm7Llo0+xNtXr1U9mCH4pObymK1HvFxHsOuVaGDtSu4g3oVz1O
8O7OWUwPVr9QI1arQgdqilsrctUPBwU6dds756n/OBXKGhpkdD3UcpR4AAAAAwBRxSSkafeXtat2ppxZ/96n2b
N2gRskpCouIPurjcm0J+j5yvJIrdqhbyUJFe/bXUmKgenhlaKuju1Y7+isjM1tlJdvVvG0XnT78AqW07shhGDg
ulHoAAAAAANNYrVZ17jtQTVt31OI5n2n9sh+Vf3C/ElJayGazH/Wx+4Jaap+9hZpVbFLXkp8V7s2rndDAKciON
dXKOEFKK7LpYNpWxSYka9B549W570DZg4LNjocAQqkHAAAAADBdREwjDRt3nVp16qFF387U3m2piopLVGRM3NF
nLRmGdgd30J6gtmpVvk6dS5YoxFdce8GB45RpS9G6kNOV7m2krF07ZQ8KVq9B56rfkNGcbIuTQqkHAAAAAKgTD
MNQ6869lNy8rX6Z/41WLZqjtG2pSmzaUkHBjqM+1mdYtc3RXTuDO6ld2Up1LF2uIF95LSUHjuzXMi/LmqyDmXt
VUrRNzdt1Uf9zxqhZ284stcVJo9QDAAAAANQpIWHhGnjeJWrZsbsWzvpYu7ask90epLjkprLZg476WI9h1OZnP
2ON7qaOpcvUrmyVbHLXUnLgN5m2ploXOl/77U1UlJ+r/RnrFRufpDNHjlOXfoNYaotTRqkHAAAAAKiTmrRsp7E
33KXNa5bqlwXfKH3nZjlCwtUoqYms1qP/OOuyOLQmdKC2OHuqc8kStSpfJ4u8tZQcDdnvyzxXRbmytqXKFhSk3
gNHqO+QOYpulGB2RNQTlHoAAAAAgDrLHhSkzn3OVJvOvbRxxSKt+HG2OrZuVFhUjGLik2WxWI76+FJLmJaHDdU
GZ1+1LVul1uXrWJaLGpFpb6p1zl/LvAod2L1drooyNW3TSf3PGaPm7bqw1BbVilIPAAAAAFDnBTtD1OOMc9S2W
1+tW7ZAq376Tnu2rFdkbJyiGiUesywpsUZodeggrQs5XS3KN6hd2SpFenJqKT3qswx7U62vosxLatZaPc8Ypnb
d+x1zTOjgZFDqAQAAAAACRmh4pE4b8id17HG6Vi/+QWsXz9XuLesVHZeoiOhGxyz3PIZd2xzdtS24m5Jcu9Wub
KWSXDvF/CmciAojWDuDO2hbcDfl2xpR5sEUlHoAAAAAgIATEdNIA0eNU6feZ2jVwjna8MtP2r1lvRolNVFYRPS
xn8AwlBHUXBlBzRXuyVHbslVqWbZBdrlqPjwC1gFborYFd9Xu4PbyGHa5Ksp1YPc2uSrKKfNQ6yj1AAAAAAABK
zYhWUPHXqXOfQdqxY+ztXnNUuVk7lNOQpLCIqKPaw+zQmuMVoQOOVrnGWpZvl5ty1Yp3JtfC+kRCFyya1dwB21
zdFWu7dAhF66Kch3IoMyDuSj1AAAAAAABLzGlhUZe9md1PW2wVi38XjtSV+tgZroiomIV2ShBVqv1mM/hsgRrs
7OXNjt6qrFru9qWrVaCa48s8tXCOOBdk2ON0zZHN+OK7iC3ESTp1zIvzV/m9TpzuNp26OuZB1NQ6gEAAAAA6gX
DMJTSqoOatGyv7PTd2rR6iTb+slBp2zbK4QxVTELy8ZUvhqHOoNZKD2otp7dIzco3qXl5qmI82TX/JmAqt2zaH
dxO24K76aA9yX+9tLhQOdkZ8rhdlHmoMyj1AAAAAAD1imEYSmjSXAlNmqvXwBHauna51i6dr8w9OyRJMfFJCgm
PPK6luaWWMG1y9tYmZ29FeA6q+f8KvjCW59YbHlmUZW+qPUFtlRbURi7LoaLO4/Eo/2C2CnMPKMgRopRW7dW57
yC17dqHMg91AqUeAAAAAKDeCouIUo8zzlHnvoOOa9MarVv+k3ZvXqf9+9IUGRunyNh4WSyW43quAmus1oYMONq
QAYp1ZSilYotSKray/14A8siqTHsz7Qlqq71BrfxFniSVlRQpJztDropyRcbGq9/QP6lt175Kbt7muH+vALWBU
g8AAAAAUO/Zg4LUpmsfte7SWxl7tmvTqsVKXfmz9mzdIGdouGLik2WxW0QPCj7u5ztoT9JBe5JWhw5StDtLKRVblVK
+RZHe3Bp8FzgVFUaQMuzNlR7USnvtreS2/Pbf2+vxKD9nvwpyD8geFKzk5m3UqfeZatWph8IioswLDRwFpR4AA
AAAoMEwDEPJzVoruVlr9R40UlvWLtaJYeW5vp8PkVExyo8utFxHazxq1xbgnJtCVobcoYi3QeU5NqpBFea4t3
psvsqavDd4FgKLZFKD2ql9KCWyrY1kc+o/N+1rLRYudkZKi8rVWRMI/UZPEptu/VR4xbtTuj3AGAGSjOAAAAAQ
IMUER2r3oPOVdfTztKuzWu1fcMq7di4Wnt3bJLVYlVkbLzCCIqOPa++9X+XbGinf1kibnH1k+LyKcWcpwb1HCa4
OxbnSZZO7Bt8RygyHDtiStd/eROn2liqwxR42xu12qTD34KFZefYgJTZtpU59zlTrTjOVHhVjQmrg5FDqAQAAA
AAatKBgh9p27au2XfuqKD9Xu7as046NK7Vv51aVlhQpPCpWETFxsu2nqYKhGbVwKq9YRyxbHsURxVyJjBd0V5BkbnVHfMHI8UlV2oMd6btPTkn58Iri/jZ3zYdYJ7aEHpTQ8p7a1RzlUfxb9+tpMqvN8bgftYa7xvDFtUmSc2VWJrUe+6xlq2o4J+jm0TZXfK9yI3zM0hR0/WpZEpNQ==
GEJSW9LOUo9VCgFVOXSA4oQUjgQ6KHApSqlpSWHoh57QHCkBy69IJBAldYI1Ivp8jf/E/9L7Hjm2YNNWBNQodg
eHvN6SaPHz29m7O9Eysh+63nmyQe55uPFssxg51xG2qeyYeHO4rZ9KkPts6mnsOKvcu21U18Md7WRTNU3Lrtdq
H2++NZeWMjM1PlMnTudhfm5NPsGsmnbjtx82/ey69pvZ+c3r0uzr3+VXwlOh6gHAAAAn9P/P4Lvplv/M9OT53L
ivbdz/N2389e/vJGTH/xv3p84nk7ZSbPVn4HhkfQPbfhyV1ItikzXN2a6vjHv9l738XLZzlD7XPrK6fR1ZtPqz
KTVmU2rXNp2ZtLXmU2znE0t5Qq8+s9vIY3MF83M11qL26KVmfqGTNY3Zqq2GO5masPJF/mMwrLMpbmLmZ2ezMz
U+Sxcmk+tVkv/4HB2X7831+4Zy9XXXJetO7/pyrV8LYh6AAAA8HcaHB7J4PBYrtOzln/b91+ZPn82J957Ox++M
5G3/zKeO8ffy9mTH6Qsy7T6BzIwNJL+weHUViA6lUU9k43NmcyVV3hd/sAyzfLC5ejXV86k2bmQWtopyjK1dFK
kTJFOinJpm/ITa4uPS8osFL2ZL1qLt9pisJv7xH6n+PK5odPp5OLsdGanJnNhejKdTjs9zVb6B4fzrRv+OVdf8
+1suWpXtl71jQyNbPrCn3EIVSfqAQAAwAooiiJDI5syNLIp37rxX/LvB+7K1NnTOf7uRI6/93Ym3vqfnD35Qc6
ceD9lWaZWr6fVN5Bm30Ba/QPp6W2uTpgqiswV/Zmr9ef8yn/3FVGWZdoLlzI7PZnZ6cnMXZhNkaQ1MJShkU25f
uyW7Bi9ZvEO6KtGO2x99gUy4OtC1AMAAIBVUBRFhjdtyfCmLbnun/41//G9/8650ydy8v2/5typEzn14Xs58d5
Eps+fzanzZ3Jpfi5JkUZPI82+gTT7+tPqG0ijp3fdHIXWbi9k/sKFzF2czfzcxdnE2n3U7KpN7TSP/AUHaMX
ptv/MOebLlqNFuvGs3Ilh1Op4VPIeoBAADAGiiKIhu3bM/GLdsvr5VlmbMnP8jU8e9bhlBKkx9y57y/b9
mZnoyk6dPZmHhUpKk0dObnt5mGj29S7eey1/XarVuvaxrJMp9PJpaVgN3/xQuYuXkh76TXUavX0tvrSb
46rs3n71dm4dUeGRjZnaGRTNm3bmYGhDV2JmGVZ5o477ki9Xs8f//fTTeeihhzI+Pp5du3at+WzaUQ
9AAAA6JKiKDIwtCEDQxuyc/fHF8HodDqZmTy5U8dz6sN3MnX2dGamhUvzi8qy
5RJarVa6kuxr6enJ/VGb4pasRTKisvBrCiKFCmSoli8XsWy/cXHdNrttBcupdle8SHth4RPbS+m0O2
avZaeZivNVl82bNqazduvMOsjmDGzZme+OPmDI1sTm+ztbb/2H9DURR59tlns3fv3jzzhzQkRiY
iIPPvhgfvOb3wh6fKWIegAzUphZ3M5M5+LS2szU+Uyd053pyX0
ZnZrMwqW5lGW5FN2WtsnH+4s7KfPR/R/v1+r11OuNNBo9qTUaafUPpq9/MHODg+kbHEpf/2Caff3D3NZ
nqbfRkcHsnQyOYO+/orderw6Ohonnzyyfzk Jz/J/v37s3v37tx3333Zv39/7r777m6PB8uIegAAAFAh9Xo9/YN
D6R8c+puPjHrtZOnU2JRlOmVnMfR1OimXfb3oimXfb3OmCxue3qb6W22OtvsS2+zlXpj/WeEgwcP5P5vnnn8+
+66K+Pj43njTe6PRZcYf3/bwQAAICvqY8CIF/MsWPHcuONN Z+ZPf/pTnnvuuWzdurXbI8EVvhqfpAkAAADwFbF
t27YcOnQoe/bsyZ133tnteeBTiXoAAAAAn9BoNNL4GpxuuTHWJegAAAABQMaIeAAAAAAFSMqAcAAAAAFSPqAQAAA
HzCI488kj//+c/dHgM+k6gHAAAAAABUj6gEAAABAxYh6AAAAAAFAxoh4AAAAAAVIyoBwAAAAAAVI+oBAAAAQMWIegA
AAABQMaIeAAAAAFSMqAcAAAAAFSPqAQAAAEDFiHoAAAAAUDGiHgAAAABUjKgHAAAAAABUj6gEAAABAxYh6AAAAA
FAxoh4AAAAAVIyoBwAAAAAVI+oBAAAAQMWIegAAAABQMaIeAAAAAFSMqAcAAAAAFSPqAQAAAEDFiHoAAAAAUDG
iHgAAAABUjKgHAAAAAABUj6gEAAABAxXYt6v/71r7N79+6OWq3ccsstefXVV7s1CgAAAABUSlei3j
h/+8IccCPnw4R48ezeuvv56bbropBw4cyIkTJ7oxDgA...
```

```
    },
    {
      "cell_type": "markdown",
      "source": [
        "1. More than half of the applicants own a car (59.69%)\n",
        "2. Not much difference in their approval rates"
      ],
      "metadata": {
        "id": "lNQwgHt3FREm"
      }
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "lfoJLZQxKFl8"
      },
      "source": [
        "#### Propert_Owner"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "Vzfk0Ox000Gv",
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 507
        },
        "outputId": "94371917-d932-4bcb-f1a8-bba52028f3a4"
      },
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 1400x500 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAABPgAAAHqCAYAAACZYfCtAAAA0XRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAACGLEl
EQVR4nOzdd3hUZcLG4WdKZia9QRoEiPTeBQbIKDoSnEVxbaLZV3RxXV1dT/bujZ0bYhiQ8EVe8G0IigoIr33n
kBIQkjvU873BzoaE1AkmZNMfvd15VpyzjszT7KY8vAWi2EYhgAAAAAAAA0SlazAwAAAAAAAD4/Sj4AAAAAAA
AgEaMgg8AAAAAAABoxCj4AAAAAAAAgEaMgg8AAAAAAABoxCj4AAAAAAAAgEaMgg8AAAAAAABoxCj4AAAAAAAAg
EbMbnaA+uLz+ZSZmanIyEhZLBaz4wAAAACSJMMwVFxcrJSUFFmt/Hs7AAA4cUFb8GVmZio1NdXsGAAAAECtMjI
y1LJlS7NjAACAIBC0BV9kZKSkIz84iYqKMjkNAAAA0KKoqEipqan+n1cBAADOVNAWfLVfoUZFRVHwAQAANDh
jIAAKCus0kHAAAAAAA0IhR8AEAAAAAACNGAUfAAAAAA0IgF7R58AAAAdcHr9crtdpsdA41sIbDab2TEAAEAtQs
EHAABQC8MwlJWVpYKCArOjoJGKiYlRUlISi2gAAICToAGCsQGqKiYlRUlISB2kAAICAo0ADAACoxY/lXkJCgs
LhIkpKTk010BAAAmgIKPgAAgF/wer3+cq9Zs2Zmx0EjFBoaKknKyclRQkICX68GAAD1jimAGAAAAOVNAWfLVfoUZFRVHwAQAANDhnguS4AAKh3HLIBAADwCz/uuRcWFmZyEjRmP75/2MMRAAA
0RIkpKTk0l0BAAAmgIKPgAAgF/wer3+ci8+Pt7s0GhkQkNDJUk50TlKSEhguS4AAKh3HLIBAADwCz/uuRcWFmZ
```

yEjRWP/7dYf9GAAAQCBR8AAAAR8GyXPxe/N0BAACBRMEHAAAAAAAAANGIUfAAAAGj0zjjjDE2ZMsXsGAAAAKbgk
A0AAIDj8N62gwF9vXEdf98prEuXLtWQIUM0atQoffLJJ3WcCgAAAA0JM/gAAACC0MyZM3XDDTdo8eLFyszMrPf
Xq6qqqvfXAAAAQO0o+AAAAIJMSUmJ3nzzTV133XUaPXq0Zs2a5b/39ddfy2Kx6JNPPlGPHj3kcrk0cOBAbdy40
T9m1qxZiomJ0dy5c9W+fXu5XC6NHDlSGRkZ/jH33HOPevXqpRdffFFpaWlyuVySpPT0dJ1//vmKiIhQVFSULrz
wQmVnZ0uStm/fLovFoq1bt1bL+/jjj6tt27b+9zdu3Kizzz5bERERSkxM1GWXXabc3Fz//dLSUl1++eWKiIhQc
nKyHn30OTr9/AEAADQ2FHwAAABB5q233lKnTp3UsWNHXXrppXrppZdkGEa1MbfccoseffRRrVixQs2bN9d5550
nt9vtv19WVqb7779fr7zyipYsWaKCggJNmDCh2nPs3LlT7777rt577777rt577z2tXbtWPp9P559/vvLy8rRo0SLNnz9fu
3fv1kUXXSRJ6tChg/r166c5c+ZUe545c+bokksukSQVFBRo6NCh6t27rt1auXKl58+Yp0ztbF154YbXsixYt0gc
ffKAvvvvhCX3/9tVavXl2nn0MAAIDGhD34AAAAgszMmTN16aWXSpJGjRqlwsJCLVq0SGeccYZ/zN13362zzjpLk
jR79my1bNlS77//vr9Ic7vdmj59ugYMGOAf07lzZy1fvlwnn3yypCPLcl955RU1b95ckjR//nxt2LBBe/bsUWp
qqiTplVdeUdeuXbVixQr1799fEyd01PTp0/Wf//xHOpFZfatWrdKrr74qSZo+fbp69+6tBx54wJ/1pZdeUmpmpqq
rZv366UlBTNnDlTr776qoYNG1YtPwAAQFPFDD4AAIAgsm3bNi1flwXX3yxJMlut+uiy7SzJkzq4ObNGiQ/89
xcXHq2LGjtmzZ4r9mt9vVv39///udOnVSTExMtTGtW7f2GXL2eVZr+y2tbdL+q9atbXq9VPLsNA7wKWrz6RKHyi/4xSxXTcJW6cIl
wQVJvf8ZUaEiq0GnQGU/ZyoR3GNyIc2f0nTT1Ijvq0hljiU3fEgX+5tnJ2QGA9ujbjwj3CDFs0yuLNf5AxkRXWH+zdv
y/pyOy9Pn36qFOnTpKkdevW6auvvlJERIT/7cd7u3bt0q5du1RVVeUvHn+eHwAAoKliBh8AAEAQmTlzpjwej1J
SUvzXDM0Q0+nU9OnT6/S1wsPdj/sxSUlJGjp0qF577u77mq677jr//ZKSEjj3339cF7Qx3NmaA8FWJRFJjX
86dJ5QZAAAgGB13wbd48WI98sgjWrVqlQ4ePKj3339fY8aM8aM8d83DEN33323XnjhBRUUFGjw4MGaMWOG2rdv7x+
Tl5enG264QR999JGsVqvGjx+vJ598Ehm5/Xef1/XUfa/ZHEfX4+f/f97NNs9h8xNXNXU/ZVcbvDg31zrcfq3v9GVzTQ=/=/RN4jd0
I2Ix+PRK6+8okcffVQjRoyodm/MmDF6/fXX/bPhvv/+e7Vq1UqSlJ+fr+3bt6tz587VnmvlypX+5bjbtm1TQUF
BtTG/1LlzZ2VkZCgjI8M/i2/z5sOqKChQly5d/OMmTpyoW2+9VRdffLF2795bW+/Pn366N1331WbNm1kt9f8U
bVt27YKCQnRsmXLauQ//fTTj+vzBQAAECyOe4luaWmpevbsqaefrrrW+w8//LD7+w8//LCmTZumZ599Vs5gg3VWLVN4eLhGjhy
piooK/5iJEydq06jMKZNmj9/vj7++GMtXrxY11xzjf9+UVGRRowYodatW2vVqlVlw3VvWv3vMaKmrqqfYjMZhDWdqnQkL3lSBFRd/mVjDPh8gGrjAjPOhzeG2OdQu3XlXyOzcjdfZo8o9bfgM0kKv5EG2ZnRlZq5cRVePbPwMhiNlmgmnApG7DMYiAC5nRgJU1jYhmUlsy/h5P9GRYC6h+wSy9mKUmdNTQjDazmFmRb/6P7Tw89pC53oXSx6GF69tEQAhs6FRYwa4JPpvn3+zWTThT1N/mKWMKGLiR3PvkS8/MNMqTfg7oUlLSVp55sW7uH8a2KIwCv+PtgxrrxC5UA2jCvC9M2kLUIRB1VfNQbFtrWsNSX2zkPBRbgwyrXtQAA0r5NjHWw5N9pGxeLNrTr5R08nb/dv9ZUwJQ6XpX/vfTL+vSzUHIP1Ph+CB1XLqVpRuXBB0sEk/lR9Y+Ixrq0XSZrZr/vLwZlWsYRqS6WM2mxf7njujXSbmC7TI2sKkd5OOEfZvE+vvZy6OhmrxfRsWS9aXCeaGOS2V8ssrbnJoMTO2VLG7Y9gvG3MzjuLdlnyeyRM2gD8YOSVUhHHMKyu5lxMAcs3O8FNzGgYkrt9TdK9nrTZbD2a2RQzWqGwv8cRIf9VkuaPEF+Y9GgXHrA6KFxSfRt9Ea8bLFnSWlS4MDdM/2RXpR6U5Rj2/Sjv0j8Rf9QqYsFdsdFjx/RTYnSgmNT0SWuvOOHrdMqWyzxCQyEhg6Nbjzbn+xpWy7ZD0zpZ8zvfZh0ERDQ4eqkmxnrEBKqXevpeS/tErj8F5aaHwzlYuxM/OUK3i6gNnNuIGA5ZDv/tnMOqp7Lb5bGSYP7i2u3ZKcHDRT2pdVk7dbX9y4nKqp1CrvXzUA2YjQ3zh57QbDrYR+yDKOvmV11vL4Yy+5ZtZkmHMTMNzb8W3iUKGcUB3TQbYLUZMpKuIqhkWsyx6fsZYWjnLctL6FsIjuPUbC2AXjbmpOHo1XH8D8zqeR2DrvLF27Jdn/mCRHxiVrNFzpoZw1tCQrKGOAFtg2dWBaOQwKJbpQgpxDD+aLo0fePu3QSX3n3yjL3ix+kg1KQcGBSlv0HsJIrBFzhP3aEWi81P8+cIj4wPHexDYMhIhRNYgcKcE5f4cqFJNXcJj6vepbahtJyWQm3VK+vpqBpw0dxgfCgRKqWqMA/8+tAVNQcBTGwy82/vLQdXnE+Ew3ZJ/Sh9cGt1dpYYgv1MrE9xvRy4w0xEU7cvStFNY8Dx3wdoYxXrnCO9j3QpyQzzOwmPPXo3fEhjH0Ggh+vGrjlajFYeTwSu0w8LHWM1X2kpoQK3UnfTRq7bhWGJk7m7g6Vn7fLg2lE3zJo2cqzcCLQ3fzUyrm2Pdd/yNyS0WhCXjpFRK0Su9DJZqlJDTj6A4KfQFKGr3SXt8NvwQAfT2EAvv+oaWdF2kOZwP2ohgJjFQkxyRGfd/CwS69M1lXQ/

```
FGjNH78eK1fv15vvvmmvv32W/9zSEfKxYyMDH311Vd655139MwzzygnJ+dX/p8AAAAIDJboAgAABKGxY8eqV69
euvvuuzVz5swa9x977DENGzZMd955pySpQ4cO2rx5sx555BFdeeWViouLk81mU2RkpJKSko75W10mTPH/uU2bN
rrvvvvv0l7/8Rc8884z/utvt1vTp0zVgwABJ0uzZs9W5c2ctX75scJ598siSpoqJCr7zyilq0aCFJeuqppzR69Gg
9+uij/gzh4eF68cUX/UtzX3jhBf/jwsPDJUnTp0/Xeeedp6lTpyoxMVETJkzQa6+9pmTJkk6UhIWFBRo/Pjxk
qQHH3xQEydO9H8c7du317Rp03T66adrxowZSk9P12effably5erf//+kqSZM2eqc+fOv/H/DQAAgPrFDD4AAIA
gNXXqVM2ePVtbtmypcW/LlI0aPHhwtWuDw/Wjh075PV6j+t1vvzySw0bNkwtWrRQZGSkLrvsMh0+fFhlZWX+M
Xa73V+OSVKnTp0UExNLVurVq385Z4kDRo0SD6fT9u2bfNf6969e7V997Zs2aKePXP6fP27ixIn6+uu
vlZmZKUmaM2eORo8erZiYGEnSunXrNGvWLEVERPjRo4cKZ/Ppz179mjLli2y2+3dvjfwAAAANAQUfAABAk
DrttNM0cuRI3X777fX2Gnv37tW5556rHj166513rvHj166N1339WqVav09NNPSzpyCEZd+3mR91v1799fbdu21RtvvvKHy8nK
9//77/uW5klRSUqJrr71Wa9eu9b+tW7d0O3bsUNu2besyPgbAAQL1giS4AAEAQe+ihh9SrVy//gRQ/6ty5s7755ptW
VLt2pIlS9ShQwfZbDZJksPh+NXQwfbZLP590jjz4qq/XIvx2/9dZbNcZ5PB6tXLnSvxxx327ZtKigoqLbMNT0
9XZmZmUpJSEkff997JarTWy//LjmDVrlkpLS/3lkpLS45atmwpq9Ww1PcZYsr8oqMcWgfcwrnHHjxr1bXGvnSN5J
VzrZKFQ59S6jvsCSrRTrMToErG3bjLIVlq3VxZR7nZb20gzqqYgLnXZG6TbE3RcXRQHdbYBV/SEGBc3V3tQuGz/lKpJ
```

8/ZCuO6OtrjujrZx2m9mROEhQ8AEAAAAAGoUFW7J1z4eblJHPclwEr0qPTO98uUMfrs3UfWO66ZR2zcyOhEaAg
g8AAAAA0KBl5JXp3x9t0pdbcsyOAgTM7txSXfLiMo3t3UJ3jO6s+Ain2ZHQgFHwAQAAAAAaJJ/P0Avf7NbjX27
ndFwOWe+vOaCvtuXoX+dO1oX9Us2OgwaKgg8AAAAAOODszy/T399ap+V78syOApiuoMytW99Zr/mbs/XQuO7M5
kMNVrMDAAAAAADwc2+vzNDIxxdT7gG/MH9ztkY+8Y2+2spydVRHwQcAAAAAaBDySqt0zSsrdMs761Va5TU7DtA
g5ZZU6k+zVuj/3t+gcv47wQ8o+AAAAAAApvtqa46G//crfbGZmUnAbzFnWbpGT/tG6zIKzI6CBoCCDwAAAABgm
rIqj/75zlr9adYK5ZV7zI4DNCq7c0s1fsZ3evLLHfL6DLPjwEQUfAAAAAAAU6xJz9eIx77WmysPmBOFaLQ8PkO
Pf7ldFzz7nfbmlpodByah4AMAAAAAABNzL3+7WBTO+0/6CSrOjAEFhTXqBRk/7Rh+vzzQ7CkxgNzsAAAAAAKDpq
HB7dfMbq/TJpkNmRwGCTmmVV5NfW6MNBwr1z5GdZLVazI6EAKHgAwAAAAAEREZeqS5/YYn25LvNjgIEtecW7db
mzCJNv7iPosNCzI6DAGCJLgAAAACg3s1fn6GRj31FuQcEyDc7cnXe9G+15WCR2VEQABR8AAAAAIB6df97K3TNa
+tU5mG5IBBI6XllGvfMd/poHfvyBTuW6AIAAAAA6kVphVt/fmGRlh2olES5B5ih3O3VDa+v0cYDhbp1VCfZ2Jc
vKDGDDwAAAABQ57Zn5mvY1C9+KPcAmO25xbt15cvLVVBWZXYU1AMKPgAAAABAnVq4YZ/+MP1bZZWbnQTAz32zI
1d/mL5Euw+VmBOFdYyCDwAAAABQZ2YvXKdr5qxXhY9fN4GGKD2vTH98dqnW7y8wOwrqEF9xAQAAAAAnzOfz6T+
vL9I9X6TLw6+aQIN2uLRKFz//vb7dkWt2FNQRvuoCAAAAAE5IaVm5/vLOJ5q5tlgGv2YCjUJplVd/nrVCH3LCb
lDgKy8AAAAA4HfLzj2sSx7/SF8csEoWTucEGpMqr09/e2ONZi3ZY3YUnCAKPgAAAADA77I7I1MXP/Wl1hWHmxO
FwO9kGNI9H23Wfz/fZnYUnAAKPgAAAADAcVu3dbcuefZb7a6MMDsKgDow/auduv299fL6DLOj4Heg4AMAAAAAH
JfFqzbqitmrlOVl5h4QTF5fnqG/zlmlCrfX7Cg4ThR8AAAAAIDfxDAMff7tSk1+e7MKjDCz4wCoB59vytY1/1u
lSg8lX2NCwQcAAAAA+FU+nO9z53+rWz7cqSIxcw8IZou3H9L1c9bI7fWZHQW/EQUfAAAAAOCYPB6PXv9ovu6en
6Eia6TZcQAEwJdbsvW3N9awJ18jQcEHAAAAADiqispKvfz2J3r42OMqskWbHQdAAH26IUs3v7VWPkq+Bs9udgA
AAAAAQMNUUVmpmW9+qOfXV6jQHmd2HAAmmLs2U067TQ+N7y6LxWJ2HBwFM/gAAAAAADVUVlZp1jsf64V15ZR7Q
BP35soM3f3hJrNj4Bgo+AAAAAAA1VRVufW/9z/V86uLVRASb3YcAA3AKOv36f5PNpsdAOdBwQcAAAAA8HO7PXr
tg3masSJfeSHNzY4DoAF54Zs9+u/n28yOgVpQ8AEAAAAAJBO5LffNjz7XcOszdTgkwew4ABqg6V/t1Ivf7DY7B
n6Bgg8AAAAAIK/Xq7c/ma+Xvtmhg46WZscBOIA98OkWzd+cbXYM/EydF3xer1d33nmn0tLSFBoaqrZt2+o///m
PDOOnI5UNw9Bdd92l5ORkhYaGavjw7jw4dqxY0e158nLy9PEiRMVFRWlmJgYTZoOSUlJXUdFcPPzPPDDeuqpp/xjHn74
6D/fbVee5ztzI4DoIHzGdLf3lijjQcKzY6CH9R5wTd16tDVgnL1kRBpaPp9bVgXPPPVeJiYmaOXm/9r48eMVGhqqV199/
35RUVGdfUwAAAAAEGy27NitV/7/7VJvtbVVghJkdB0Aj9uWWbD3w6Rbdc4W4Xs6MOWXU+g++tt97SnDlz9NprrQ2n
16tWaPXu2/vvf/2r27Nl1/VLPPjgg4qa/paamluvrAQAAAEBjtf9gtma99a/99i99
GX7zI7RZNV5wXfLLbfotttu1uuommm/Tggw9KkpSkiRJ2dnVj1Pzs7TkUWJ2d+ndm/Jz7BOetKSlJOTk/x6
PR3l5ef4xh+Tm5mrw4MEKCwtTRnp9DrDw1DVYNOzs7723
OtKSlJOTk61+x6
PR3l5ef4xv3T77bersLDQ/5aRkVHXHxoAAAAANHoFRcV6V+aOLHp/OhLc2OAyCI3P3BJn2z45DZMZqk0i/4y
srKZLVWf1qbzSafzydJSktLU1JShYsWOmnQoEGUGXLlmnQoEGSpEGDBqmgoECrVq0vq3y
6nU5FRUVVewMAAAAA/KSislKz3/9Q/5QK7Yf1L7Q9mbHARBkPD5DN76+RgcLy8F33nnn6f7779779cnn3yiv
Xv36v3339djjz2msWPH+SpIsFoumTJmi++67T3PevVj4ap
WLNHkyZM1YcKE33SCLgAAAACgOq/Xq/e++Ew3uXW5HH1Mo4BUrq8/zM4jaYQAGXo4mAQoEPRnUKT7dY/vcA7nK9w/1QtiQ
9UdUk42k46mnntKdd96pSUpS96I4ScKE/5JSx8pMk4QgH3k4M4uTjxs7FX5Q1JU2iU5
ALl9SlHdCqPH6OtfG8hFdbZPxSNp/LKA6RuodyiYnBJPekRTpTzbz/rXEP1cTAZfk1Fc6nVVj1Ca+4Vk+mfB/Gk3
dwnNLJc9bHZ/TpaKSGqXBGWhv/cPJG2MK8bYd8AlacfJnpTdk1X0NZ8LbT95PfclqzlZWlsTlyQtEe+Z7i/m
mWcLQnUQmqoKpiyVRg8qmTxRyYOxBpKMSvqbCbUNlCTD4nsaavRJBePj4T9OKwkBLjqbb0hxNKM2VkZwIzNNB+U=

K6IauNcaX3V7JwpP12wH/vAkoqMjYrsM1qOpPaS4VXBoleU/dadSpk0Q1aHyz8uoudIxQy51P++JcTp/3Ph0jf
lqypX8pVPqnjNpzo87yklX/GEJKnywFZVHdymuOHX/P4PHjDJvz/arCHtmik+wvnrg/GbUfABAAAAQBD4cd89p
8up1e4W8rE091cVff+07FHN1Gz0FP+1kJikGuMs9hDZImJ/8/P+cnZf/0ibtP+piarK3ilXarefPa/zqM/rPpy
h8M6nKSSuhSJ7jlLJunmSJMPr0eEvnlb8qBtlsdp+cyagocgrrdLdH27S9Ev6mB0lqFDwAQAAAEAjV1ZeoTlzP
9Xh/CKVJ/VQXsmxZ5jhiPKdy+RK66NDcx9URcZG2SLiFdn7HEX2GlVtXEX6BmU8NVFWV4RcrXoo5rTLZAuN+s2
v46sslaQaMwNLN3+t0s1fyxYeo9B2Jyv6lAmyhhyZ4edISFPFvvWK6DlS5XtWK6R5G0lS0bJ35Urt7l9CDDRGH
68/qPN7ZeusLhwCVFco+AAAAACgkfvsq2+1cet0NW/VVp8URJodp9FwF2TJveZTRfUfo8RBF6ry4A7lL3heFlu
IIroPkySFpvVRWIdTZI9JlCf/oAoWv6Kct+9W0qX//U0z6AzDp/wFL8jZooscP5R0khTe5QzZo5rLFhmvqpw9K
vh6ltx5B5Qw9v8kSdED/6jDnz+jA89dJXt0ouLP/pvceQdUsnGBki77rw5/Pl0Ve9bIkdRe8WffIKszvF4+R0B
9uWPuBg04KU5RLv5Boi5Q8AEAAABAI7Zlx259vug7NY+P1bLS5izNPR6GIWdSO8WefoUkyZHYVu7cfSpe+6m/4
Avvcrp/uKN5G4UkpCnzuatUkb5BoW16/epL5H0xQ1WH9ilp4sPVrv98lqCjeRvZIuKU88b/yZ1/UCGxybI6w9X
8D7dUe0zW6/9S7Jl/Vummr+UpyFbK1c/p8LynVLDkdcUNver3fhYAU2QXVeqBT7boofE9zI4SFKxmBwAAAAAA/
D6lZeV6+5P5Ki0rV25oqnLcnJp7PGwRsQpp1qratZD4VHmLDh31MSExSbKGRslTcPBXnz9v/gyV71qhxIsfkD2
q2THHOpM7SpI8+Zm13i9ZP19WV7jC2g9URcYGhbUfKIvNrrBOQ1SZvuFXswAN0RsrMrRkZ67ZMYICBR8AAAAAN
FKfffWtNm3fpaSWbbSy+LfvCYcjnC26yJ23v9o1d94B2aMSjvoYT1Guf0XFsoXHHXWMYRjKmz9DZduXKnHC/bU
e3PFLVTm7JUm2iJrP6y0rVMF3byhu+LVHLvh8MnyeH256ZBi+X31+oKG67b31Kq/ymh2j0aPgAwAAAIBGaOvOP
fpi8VIlNovXqvJ4uQ1+vTteUf3PV2XmNhUufUvu/EyVbv5aJevmKaLPaEmSr6pc+V+9pMoDW+UpzFb53rU69N5
/ZI9NVmjaTyeAZr/xLxWt+sj/ft78GSrZ9LWanXeLrI4weUvy5S3Jl89dKUly5x9UwZLXVZm1U57CbJXtWKbDn
zwmZ2o3ORLSauTMW/C8ovqPkT3yyCxAZ8vOKt30ldy5GSpeN0/OFl3q89ME1KuMvHL994ttZsdo9NiDDwAAAAA
amYrKSr332QKVlpXLHt9Ke/NCzY7UKDmT06j52P9TwaLZKljyuuzRiYoderUiup55ZIDFqqqcPSrZuEC+ilLZI
uIUmtZbMadeKov9p4MB3PlZcpYX+d8vWf0pJCn79durvV78OVMU0X24LDa7KvatU/HKD+VzV8ge1UxhHU5R9Ck
TamQs371KnvyDanbuzf5rkX30VdXBnTr4v7/LmdxBMYMvrstPCxBws7/bqwn9U9U+kUOCfi+LYRiG2SHqQ1FRk
aKjo1VYWKioqMBPVX9v26/vxwAAv2Zcx2SzIwAA6pjZP6ciOHy68Bv9792PdVLrVppXnKLD7L0HoJE7tX0z/W/
SALNjNFrM4QYAAACARiT9wEF9suAbxUZHKcMXTbkHICh8syNXX270NjtGo0XBBwAAAACNhMfj0XufLVBeYaGaN
U/QyiJmgQIIHvd/ukVVHg6N+T0o+AAAAACgkVi+dqNWrd+sNi1baENppMp8NrMjAUCd2ZNbqtnf7TU7RqNEwQc
AAAAAjUBpWbk+Xfit7Ha7DGe41peGmx0JAOrcUwt3qKCsyuwYjQ4FHwAAAAA0Aou+X6ld+zLUqkWyVhRFyWvw6
xyA4FNU4dH0hTvNjtHo8B0BAAAAABq47NzD+nzRd4qJjlK+Eapd5aFmRwKAevPK0n3KyCsz00ajQsEHAAAAAAA2
YYRj6YtFSZR/KU3JCM31fGC3JYnYsAKg3VV6fHvl8m9kxGhUKPgAAAABowHbuzdA3y1YrKaGZ0ivDlON2mB0JA
OrdR+sztWF/odkxGg0KPgAAAABooLxerz5ZsFglZWWKj43W2pIIsyMBQEAYhjRt4Q6zYzQaFHwAAAAAA0ECt3rh
VqzduVasWydpf6VIus/cANCFfbsnWtqxis2M0ChR8AAAAAANAAlVdU6JMFi2WxWBQRHqY1xZFmRwKAgDIM6emv0
FH3t6DgAwAAAIAG6NsVa7Vt1161bpGszEoHe+8BaJI+2XBQe3NLzY7R4FHwAQAAAEADU1xSqi8WLVVERLgcjhC
tZfYegCbK6zM04+tdZsdo8Cj4AAAAAAKCB+X7NBu0/mKUWic2VXRWizCqn2ZEAwDTvrdmvzIJys2M0aBR8AAAAAA
NCAlJaVa8G3yxQeHia73c7sPQBNnttr6PnFu820OaBR8AEAAABAA7J87Ubt03BQLRITl0u2K6PSZXYkADDdGyv
SlVtSaXaMBouCDwAAAAAaiPKKCi34dpnCXE6FhDB7DwB+V0H26cVv9pgdo8Gi4AMAAACABmLlus3anbFfLZISV
eCxaW8Fs/cA4Eevfr9PhWVus2M0SBR8AAAAANAAVFZW6ctvl8kZ4pDDEaItpeGSLGbHAoAGo6TSozdXppsdo0G
i4AMAAACABmD1xi3atS9dLZIT5TGkHWVhZkcCgAbn9eUZMgzD7BgNDgUfAAAAAJjM7fZo/jffy2azy+V0aHd5q
KoMflODgF/ak1uq73YdNjtGg8N3DAAAAAAw2ZpNW7V99z61TE6QqpB+W5wIAajNn2T6zIzQ4FHwAAAAAYCKfz6e
vv1shi8WLAJdLuVUh0uR2mB0LABqs+Zuzdai40uwYDQoFHwAAAACYaNe+/dq6a6a69SEptLkraw9x4AHJPba+itl
Rlmx2hQKPgAAAAAwEQr1m1UWXmFIsLDVOWzaFd5qNmRAKDBe315unw+Dtv4EQUfAAAAAJikoUF4wnlJe6T3AgNBgUfAAAAAJjA6
/Xq2xWr5XQ6ZbPZlFXpUL4nx0xYANBoLNyao6zCCrNjNAgUfAAAAAABggm2792nzJzSTJ01g7z0AOC5eH4d
t/IiCDwAAAABMsHzNRlVWVik8LFQ+Q9pLwQcAx+2jdZlmR2gQKPgAAAAAIMBy8wq0Yv1GNf9h770D1U5VcrgGA
By3HTkl2pFdbHYM0/EdBAAAAAACbM2mrTqcV6hmcUcKvj0VLpMTAUDj9cmGg2ZHMB0FHwAAAAAEkMfj0TfLVys
8zCWr1SqfIe1jeS4A/G6fUvBR8AEAAABAIO3cm6H0AweV2PzI4RoszwWAE7M9u0Q7c0rMjmEqvosAAAAAQABt3
rFblZVVCgs9six3N7P3A0CENfVZfBR8AAAAABAgbrdHK9ZtUmREhCTJZ0jp7L8HACeMgg8AAAAAEBC79mXoYPY
hNYuLkcTyXACoK1uzirXrUNNdpst3EgAAAAAIkE3bd6myiuW5AFAfPl3fdGfxUfABAAAAQADUtjx3H8tzAaD0f
NKEl+lS8AEAAABAA0zcm66D0blqHh8jScqqcqiK5bkAUGe2ZhVr3+FSs20Ygu8mAAAAABAAm7bvUlVV1UJdR2b
tZVY6TU4EAMHn2525ZkcwBQUfAAAAAANSzqiq3Vq7frKjICP+1AxR8AFDnvtt52OwIpqDgAwAAAIB6tvMXp+dW+
SzKdYeYGwoAgtDS3YdlGIbZMQK0Ogg8AAAAA6tnm7btUWeWutjzXkMXkVAAQfPJKq7T5YJHZMQKOgg8AAAAA6pH

```
P5906zdsVGRHmv8b+ewBQf5riMl0KPgAAAACoRwdzcpV96LBioqP81w5UOUxMBADBbcmupnfQBgUfAAAAANSjv
RkHVFxaqsjwIzP4Sr1WFXrYfw8A6svyPXlye31mxwgoCj4AAAAAqEc792bIYrHKaj3y6xen5wJA/Sqr8mpNeoH
ZMQKKgg8AAAAA6onb7dGm7bsUHRXhv8b+ewBQ/5bsbFrLdCn4AAAAAKCe7D+Yrdy8AsVERfqvUfABQP37rontw
0fBBwAAAAD1ZE/GAZVXVCgs1CVJKvbYVOazmZwKAILf2owCVbi9ZscIGAo+AAAAAKgn23btld1ul8VikSTlujl
cAwACwe01tOVgkdkxAoaCDwAAAADqQVl5hbbt3ltteS4FHwAEzsZMCj4AAAAAwAnYtz9TeQVFiqbgAwBTbDpQa
HaEgKHgAwAAAIB6sCcjU+4qt1xOh//aYQo+AAiYjZkUfAAAAACAE7Anfb9CHD8VeiUemyo4YAMAAmZ7VoncXp/
ZMQKiXgq+AwcO6NJLL1V8fLxCQOPVvXt3rVy50n/fMAzdddddSk5OVmhoqIYPH64dO3ZUe468vDxNnDhRUVFRi
omJ0aRJk1RSUlIfcQEAAACgTnk8Hu3JyFREWKj/GstzASCwqrw+bc8uNjtGQNR5wZefn6/BgwcrJCREn332mTZ
v3qxHH31UsbGx/jEPP/ywpk2bpmeffflVbLli1TeHi4Ro4cqYqKCv+YiRMnatOmTZo/f74+/vhjLV68WNdcc01dx
wUAAACAOpdzOF9FxSWKiAj3X6PgA4DA23SgaRy0Ya/rJ5w6dapU1P18ssv+6+6dPnUkU/TnHPOOX/TnV7Ns8HuX5fnH/NLtt9+uwsJC/1tGRkadWAAAAAqDNZh3JVW
l6uFsmJ/msUfAAQeBszC3WhUs2OUe/qfAbfhx9+qH79+umPf/yjEhIS1Lt3b73wwgv+++3b73wwgv++3v27FFWVpaGDx/uvxY
dHa0BAwZo6dKlkqSlS5cqJibGX+5J0vDhw2W1WrVs2bK6jgwAAAAAdepgdq4Mw5DN9t0vXBR8ABB4G5vISbp1X
vDt3r1bM2bMUPv27fX555/ruuuu04033qjZs2dLkrKysiRJiYmJ1R6jov5eVlaWEhIRq9+12u5i4vxjfqm
yslJFRUXV3gAAAAADADOkHDspm++lAjVKvlQM2AMAEWw4Wy+czzI5R7+q84PP5f0rOrTp48eeOAB9e7dW9dcc42uv
vpqPfvss3X9X9UtUtU8+OCb/9EsAAAAAADY/X69XujP2KCA/zX80A/tLXOC77k5GR
l6dKl2rXOnTsrPT1dkpSU1CRJys70rjYmOzvbvbfy8pKUk5fnH/NLtt9+uwsJC/1tGRkadfDwAA
AAAcDxyDuersKhEkT8r+Iq8zN4DALPszS0zO0K9q/OCb/Dgwdq2bVu1tb1a9u3bt0ajBfu1ifr1q0lHTlwIykpSQsWLPDfyo
qOrJlyzRoOCBJOqBBg1RQUKBVq1b5xyxcuFA+n08DBhq5d9q9XWbnCw0L914o9d
X6+IQDgN8rID/6Cr86/y9x000065ZRT9MADD+jCCy/U8uXL9fzzz+v555+v555+XJFksFk2ZMkX33Xef2rdvr7S
5551KSUnRmDFjJB2Z8Tdq1KZ8EAFBBBB2Z8Tdq1Cj/0l63263JkydrwoQJnKALALAAAAoEHLysmV4uV4T0q7cFXzAw+ADBNRl652RHqXZ0Xf
P3799f777+v2229XvEd6/S0tL0xBNaOLEif4xt99+u0pLS3XNNdeooKBAQ4YM0Yc0bx58+Ryufxj5syZo8m8mTJ2v
YsGGyWq0aP368pk2bVtdxAQAAAKBO7dt/UDZ79UKPGXwAYJ6mMIPPYhhGUB41lU1RUpOjoaBUWFpqyXPe9bQcD/
poAgs+4jslmRwAA1DGzfO5F/TIMQ3c8M1l25eQVKTflp//DZB5PkNup8hyQAwG/QKSlS86acZnaMesV3GAAAAAC
oI+UVlSoqLpXL5fRfq/BZEKY7PN/iS7fZQAAAACgjuQXFqmsvFyhYWH+a0x8s/AMIuQKRdGGsOqU4nBuiKlj2FZZFf/ZB5PkNup8hyQAwG/QKSlS86acZnaMesV3GAAAAAC
oI+UVlSoqLpXL5fRfq/BZKPcAwET784N/iS7fZQAAAACgjuQXFqmislKhzp8KPpbnAoC5Sio9yi+tMjtGvaLgA
wAAAIA6UlBUrPKKSrmcDv81DtgAAPMF+z58FHwAAAAAUEcKCovk8/lkt/80a48ZfABgvmA/SZeCDwAAAADqSH5
RcY1rRczgAwDTMYMPAAAAAPCbHM4rkMOhHIPAEyXkUfBBwAAAAD4DTJzDlXbf0+SKij4AMB0h4orzY5Qr
yj4AAAAAKA0eL1e5eYVPzE3QlqcLHr10AYLbCcrfZEeoV32kAAAAAoA4UlZSqrLxCLtcvCj7DcpRHAAACpaj
CY3aEekXBBwAAAAB1oKCwWBWufdW6Lp9FnkNfu0CALMVMYMPAAAAAPBrCoqLqVVlZJafjp6uVHa9X8+kB2j3lDwAQAAAEAdKPuh4Pu5KmbwAUCDYBhSUUXwzuLjuw0AAAA
A1IGyisoa15jBBwANR1F58B60QcEHAAAAAHWgrKysxjVm8AFAw1EYxPvw8d0GAAAAAOpAYXGp7HZ7tWtuZvABQ
IPBEl0AAAAAwDEVl5TKbrdVu+ZmBh8ANBjM4AMAAAAAHFNpeblstu7lstuoFn4cZfADQYJRUsgcfAAAAA0AYyisqZf9
FwQcAaDi8PsPsCPWGgg8AAAAATpDX61O+VFJQUfADRkwVzw2X99CAAADcfGjRvNjgAgSHTr1s3sCAgilVVueb0+
WzMoQCAhspnBG/Bx3cfAAAAADhBlVVV8ni9F9NfbgAwA0HME8g4+CDwAAAAB0kNfrk+HzyWqWqfqhG8P4qCQCNTxD
3eyzRBQAAAAAARzfRtUQ3Wd40Owbwm/kMnwyfoZjoSFmtP81tc4f8XVKaecHqEQUfAAAAAJwg48d9nX4xgw9o7
FpZD+tflpcVbpSZHQU4PlZJxfnVLoUY5eZkCQCW6AIAAABAHaht5VcQrwZDk+DTDNcz1Hs1Htbg3SeVgg8AAAA
AANTwz9CP1dW3xewYQN2xBG8NFrwfGQAAAAAEiPHDXD0W6LafWZHQWoW5bgLfg4ZAMAAAAA6kmIhV340PhMC5+tk4wSs2MAdc/mMDtBvWEGHwAAAA
iOHr0LTbI/LafWZHQWoW5bgLfg4ZAMAAAAA6kmIhV340PhMC5+tk4wSs2MAdc/mMDtBvWEGHwAAAACcoK0doss
MKDQ251qXaYSx2OwYQP0IjTE7Qb2h4AMAAACA0lDbXD0HBR8akVhfvh50viwr5z8jWIXGmp2g3lDwAQAAAEAd+
eWeew6W6KIRmRnxnCKNIrNjAPWHgg8AAAAAcDQWi0UWi+Wnpbo/YAYfGou/0j5TH996s2MA9csVY3aCekPBBwA
AAAAnyOlwyGazye1fbW2bHA0ofM/gAAAAAAEfjcjhkr63gY4kuGjiL4dHM8BlyGpVmRwHql
yNCstnNTlFvKPgAAAAA4AQ5HCEKsdvl9VQv+DhFFw3d/aFvqo13r9kxgPoXxLP3JAo+AAAAADhhFotWYWGuGjP
4bBbJxomkaKAGWzbqIs0zOwYQGEG8/55EwQcAAAAAdSI8I8iHDw1TqFGmaaHPy6aaf2eBoBQaY7aEeU3aCe
kXBBwAAAAB1ICIOtMYMPomCDw3TM+EzFe/LNTsGEDgsO0QUAAAAA/JqI8PBaC75wCj40MBfavtEZEZvqVmxwACixl
8AAAAAIBfExEeKp+3ZpkXYfeYkAaoXaIvV3c7/ieL2UGAQAtrZnaCekXBBwAAAAB1wOl0yGKpWZWtE2tjjDA3HS
xEzFG6UmBODCLzY1mYnqFcUfAAAAABQB1xOp2ry9xRBwYcG4u+OD9TVt8XsGEDgsOKQBbgLeg4Zgcwsq/rW
ZhlHjOjP40BBOOW5dZ33P7BiAeeJOMjtBvaLgAwAAAIA6EBERJkny+arvRkcGHxqCsc10hIw+Q05VQ9+UzcV3pmnZx6Uit9lRAHPYH
FJUC7NT1CsKPgAAAACoAzGrGRkqp2pqp2Pczqk1U1Z/YBgfJI2P/U0rff7BiAeWJaS9bgrsDsZgcAGiKv16u
3pj+qxR++q4LcQ4pNSNSZYy/UBddN8W+cXJB7SP/77/1at2SRSosL1aXfQE264z61tDn6tN/0Hdv0xrRHtHvTe
```

h3K3K8/3f5vnXvF1dXGbFrxvT6Y+Yx2b9qg/EPZunX6TA0Yfna1MR/MnKG5M5+RJI296nr94c9/8d/bvm61Xvj
37XrorU9ks/OfOILP9pB4syMACBLdzA6AoBMdFSGXw6GKyiqFulz+6xbLkX34irz8bIbAG2pdqzHGArNjA0aKC
+799yRm8AG1mvvC0/r89dm66s779eQni3TZzf+nuS8+o0//N1OSZBiGpl7/Z2Xv36fbnnlZ/33vCzVPaal///k
iVZSVHfV5qyrKlZjaSpfe/C/FNE+odUxleZnad0qqq+96oNb7e7dt1htPPaK/PzZDNz36jF5/8mHt23Zko1yvx
6Pn7/mnrv33VMo9AACAAIu0jFCoy6nKyqoa9zhoA2aINEr0q0tFWeX79cFAMAvy/fckZvABtdq2ZqX6Dxupvmc
MlyQltEzVN5/M1c4NayVJB/fu1vZ1q/T4R1+pVfu0kqRr7nlIk4b01LefvK/hf5xY6/02695L7br3kiS9+mjtB
V6f04aqz2lDj5rtw06dat2xi7oPHCJJat2xsw7s2anWHTvrg5kz1LnfQP9rAAAAIHBsNpviYm00NyOzxr1Iu0e
qcpqQCk3Zc+EvKNaXZ3YMwHxBfoKuxAw+oFYde/fThqXfKnPPLknS3q2btHX1cvX+oXhz/7CvisP50w9pVqtVI
Q6HtqxaUa/ZWnforIN7d+tQ5n7lHNivzL271ap9J2Wl79XC997UJX/7Z72+PgAAAI4u0aGZKiora1znJF0E2mX
2hTrFV7+/mwCNRhNYossMPqAWY6+ZrLLSYt14zmmy2mzyeb26ZMpt0u28cZKkFie1U70UFnr1sQf1l39PlTMOT
B/Pfl6Hsw4q/1B2vWZr2ba9LrnpNt375wmSpIl/v10t27bXPX+6UJfd8n9a++3XevPpR2W32/Wnf/1HXfsPrNc
8AAAA+EmzuNgap+hKUjQn6SKAWvqy9K+Q18TZLsAPmsAMPgo+oBbfffahvvnoPU3579NKbddRe7Zu0ssP/b
MMeEqJbp83UM3f8XVcM6CKrzaYeg049MsPPqP/voiMnXK6REy73v//V+28pNDxCHXv10w1nn6qpb3+qw1kH9fj
fr9OMBd8rxMFyEAAAgECIiYqo9Xp8iDvASdCUvRQxQ6G+o+8NDjQpFqsU28bsFPW0gg+oxSuP/Edjr56sIaPHS
Dqyz11u5n699/xTOnPshZKktt166NG5X6q0uEget1vRcfG67cLRatutR0CzFuUf1ltPP6b6b7Xn1PO9avVkqbk/x
vXo9bmXt2q3XHzgHNBAAAOFRFR0bKYrHI5/PJav1pR6RIm1cOiO9VBrskoX79y/mOOvh2mBODaDiiW0p2h9kp6
h3fXYBaVJZXyGKt/p+H1WqT4as50y88MkrRcfHK3LtburrrlZ8Uop8Xp+8np/+ddj
r9crnY78XAACAQImJipTT4VDFLO7StVikOGbxoZ711g792fKh2TGAhiW5l9kJAoIZfEAt+p15lt59dpqaJ7c4s
kR3y0Z9N0s5DR0/wT/mu3kfKSo2Xs1SWih9+xa9dP9dOcM/Zto/b1RcQpIuvflflko4czrF/13ZJksf
t1uHsg9qzZaNcYeFKbn1kT4Dy0lJlpe/xP0f0/gzt2bJREdExap7Sslr0dUsW6eDe3brhoSclSe2699fNuO
vFC5R7MlNVqVUpa23r5HAEAAKCm6KgIOZ0OVVZVKSzUVe1esxC3sjhJF/UkxKjUjPBnZfex3yNQTYs+ZicICAo
+oBZX3XGfXp/2s2J6/93YVHn/1EXXl2vPZh/1EXXDel2vPkZh6QxWKt9
ph/jB3hf//Dl57Vhy89q679B+ne/7OrSdq1cZ3uvuIC/5hZD90jSTpjzIW64aEn/NcrK8N/+T39//Fn/8o/
4pBRNuuM/evpfN8nucOiGh56U0xVaV58WAAAA/IroyAhFhYerqLRUsdFR1e6xDx/q0xNhryjd9DsGEDD06Kv2
QkCwmIYATgRwARFRUWKjo5WYWGhoqKifv0Bdey9bXxhBXDixnVMNjtCg8PXVwB1xayvsWb/nIr6N+0Vt/TdqnX
q2LZNtet5brve05RgTigEtb0tK/SO4wlZOTYX+AWLdFu65Ar+77fswQcAAAAAdahVi2R5vTX3QY6xe2SjgEEdi
zUKNdX5EuUeUJtm7ZtEuSdR8AEAAABAnUpsHi9J8vl81a5bLVIsy3RRqHZMYCGKaVp7L8nUfABAAA
AQJ1Kah6v8LBQlZaV17jHPnyoS9eEfK5+vrVmxwAariZywIZEwQcAAAAAdSqxWbwiwsMoMo+FCvOowDutn+ptkxg
IatiRywIVHwAQAAAECdCgmxq1WLZJWUltW4l0DBh7pg+DQz/Bk5jQqzkwANlzVESupudoqAoeADAAAAgDqW1jJ
FVe6aZV58iFs0i6+WRwC/3b2hb+gk3x6zYwANW2IXye4000XAUPABAAAAQB1LSmgmQ5JhvD/Z1GKRkp2V5oRCU
Bhg2aKJ+szsGEDD14Q02JAo+AAAAACgziU2i1eo06uHvqcbPKaHQVo+FJPNjtBQFH
wAQAAAEAdS0qIV0R4qEpLax60kcIMPvx008NeUnNfjtkxgMah7VCzEwQUBR8AAAAAA1LFQl0stk5NUXFJa415ci
EehVmZg4fiMsX2nYcYSs2MAjUNiNykyyewUAUXBBwAAAAD1oHP7NFVU1tyGxnAI1FE5u
9J1HwAQAAAEC9aNMyRTabTVVVNU/TTWEfPhyHl8JnKMIoNjsG0Hi0G252goCj4AMAAAACAetCmZYpio6zm
GEfPvxWNzo+Vg/fJrNjAI1HSLjUapDZKQK0gg8AAAAA6kFkRLjatm6pwqKaBV+U3asIm8eEVGhM0mqfJlvfMTs
G0LiknSrZHWanCDgKPgAAAACoJ53apdW6RFdimS6OzWJ49GLYM3KIvyfAcWmCy3OlABR8Dz3OkCwWi6ZMmeK/V
lFRoeuvv17x8fGKiIjQ+PHjjlZ2dXe1x6enpGj16tMLCwpSQkKBbbrlFHg//wgUAAACg8WjTMkbaD9to4ao
wIREai6mhrynVl2F2DKDxaYIHbEj1XPCtWFLzzz1nHr0nrpenanemmtbrEpSQspvMFnqUnKnhHlnNfMVe8K1
atUo6rOxayNOqOT0XSGmCiQfVgaA2h
8YtOk+LZmpzBFvRV8JSUlmjhxol544/14JQXFxsb6rxcWFmrmzJ167LHHNHToUPXt21ezZs3Ssm3b9gfD2lcvv/yyxdBmD0Ta+EpqFzDfXUxqO9abfRAbv2EmLbG6/NedMzubQqlpsbJUnv/KSIaxnTmG84i6t5c3LbnTFm8i8AWaLsb5aqSQ7FtfPUVVVanf3JH8BzHCV/1vu62c6G3FtO0BmTjDVZTsWFSCHPuJwpdcB/9ebeIQ3a3MUcNBMt7iRZhr5FyoImLB/kDTI7tMYQlVxTeRWAPQ9vMmXnr0jiPTc56oRnSFbKDU5RvinyBKUfWXAWn7+jLxXbpWB8EDvAD/p4kPs5j0pJsQbAk72VB8qZSmPnc0Zj3g0k1Kf+uTpP3DHJJ0m2qtZpvOT4jxwVG9Y7XXkSN7axoECqrGCkRedsv+q8RgCmjeSCx9rK7rjMZn7R2zJv5FDcVcqlsTPs8Ggk0FjFnvlsFPWmGJxCpLgVBFC9VFS9wwA6ORD7hNbgmXTcXpgRnFVakEx9S9zh0Ua7vzKOkwBJiT4XBWTWD6TOr1wMhQ0VP7kBALC1iE1hL0p22T4eR2n4bb3WcNVEyZw4LEwNvvnWaRhrPAv6y8jIi2eL69vEkl68gLaGH8QydSqxWoeRMpaDoQWrGkeXZmDJQJh7vNvG5hwoBXGLsOxpfhRjwRhsqLEswtQmLsf7E6LL+qlKZ2iKqwL5DgbtUEvVQAJWoDV96zPRsKVB2WJ+v8I7rjlmyPCoVrK7DbU2A8N48d6Mn34gKxqUmlsRhBQ5gNaWmhO7HWEARNvfWbyuMsZXWhBGCGM4ML3dEkhnoUhvXKCnvb7F4PS9yX4ntOELyl3O5Fo/Qy4MLCLXYH13Mte1O9atWXDWVT3LGbxDfHVEdlaHOR7HdlwZWF8fJCWvhqz5t9vw0Wqg72a4wl5x6VHNm6A1TC4n+7TtCcCpCqLcTWVZDLt/mOmOuTUkyrLMZw8OAqFo2JE+xq+m7bH2XoqAo1L2yVeZ+8VWnt1+zlhM7Nk8U60GDWOYaHhdTTXpUV/aM0fzJHVjdzvyY5ljkDTzUOjqrNMFDZoZGyYiD6M6Grzmkr+9PLW3BbngOtU9K5VebFHLzVSxvDV7v1pDg8/lplNPPf9mO1LSHNhHzbTpG7tr7vW3U0pLhPYpyOo7qRgrU1XoTm0Wd5pUPvLGQD2zj16zkMn8dHoXQVYkGfmKHCWd4FjIVTNL7HO30tMh21RZVsDKoU5U3wRy1o7m4Rr0lK0v7Uqn/5nuuXb1eAtTMA3vjjVwUSNEmkjc8Q+4tSOULRfW9pqz1Pk3/Nl/vyu1/7yGeP0kdAZFaEI4X+HvCAvBjoQXEp7OOAVVU0BZiBXoq6kOb3LePd2lFDcr0MQi2OYL3DFnqZvU0G+Owp4kQhwb1K8+/gjNzrHdyRRzUTKG6mXZZqN1tdbMivHZVB+U6tpxOB1ETZPr/3B0/Nw4Nu+0a6wTB+/9AD01+3/1p5d9PgA5j2Wnpjy+MFgm4ZA2HdgPg4UE4V1tWX0R2rXmdnfnwQY9I8mjzLXx2Y7tWpVW2Dg2xmfYM6K9dyV3gn7Yvw9GBXmnoTaSZp+MMfVy9r3YlT6UVvpt0EZg5hUJkjmeCjTJgn8XE2f21n+AAAA

ANARdOrSVzWZTZS2HBv64TBdN122099TZt93sGEDjFZ4gnXSm2SlMVed78EVGRqpbt27VroWHhys+Pt5/fdKkS
fr73/+uuLg4RUVF6YYbbtCgQYMOcOBASdKIESPUpUsXXXbZZXr44YeVlZWlO+64Q9dff72cTmddRwYAAACAetU
+rZUSm8XrcF6BUpISqt1zWg0lOyt1oDJwWxyh4eihnZpknWt2DKBx6zpWstrMTmGqejtF91gef/xxnXvuuRo/f
rxOO+OOJSUl6b333vPft9ls+vjjj2Wz2TRoOCBdeumluvzyy3XvvfeaERcAAAAATkioy6U+3Tup4GjLdDlNtOm
yG1V6LuxZhchjdhSgcetxodkJTFcvp+j+Otdff13tfZfLpaefflpPP/3OUR/TunVrffrpp/WcDAAAAAACo2uHd
vrsqyWqqnLL4Qipdq+NqOJLCgOZnKbbpDwe+oqSfZlmxwAat7iTpJb9zE5hOlNm8AEAAABAU9PhpNZqHh+rw/k
FNe65bD5O021iRlhXabS+MjsGOPh1b9qHa/yIgg8AAAAAAiAs1KU+3Tor/yjLdDuFlQU4EcwSYxTqEddMDlcB6
kJ3ludKFHwAAAAAEDBdOrSV1WJRldtd414rV4XCrF4TUiHQngt/QdG+ArNjAI1fci+pWTuzUzQIFHwAAAAAECA
dT2qtZnExyssvrHHPapE6MIsv6EOKma8BvtVmxwCCA4dr+FHwAQAAAECARISHqWfnjsorKKr1fsewMllYthmOW
huZusX+htkxgOBgtUvdxpudosGg4AMAAACAAOrRpYNsVosqq6pq3Iu0ezlsI1gZPs2MmCGXUW52EiA4dP6DFJl
kdooGg4IPAAAAAAKoa4eT1CI5UVmHDtd6n8M2gtPdoW+rnXeX2TGA4DHwOrMTNCgUfAAAAAAQQC6nUOP691Jxc
akMo+ZyXA7bCD79LFt1mT4xOwYQPFL6SKknm52iQaHgAwAAAIAA692ts2KiImrdi4/DNoKL06jUM6HPyS6P2VG
A4MHsvRoo+AAAAAAgwFISm6t7p/bKyT36Ml0O2wgO08JeVoIv2+wYQPCISJK6jjU7RYNDwQcAAAAAAWaxWDSwb
O9ZLLUfthFh96olh20OeufbvtdZxjdmxwCCS/9Jki3E7BQNDgUfAAAAAJigW4e2Sk1JUlZObu33IOoCnAh1Kd6
Xr/scs2RlJiZQd2xOqe+fzE7RIFHwAQAAAIAJnE6HBvfvpeKS2g/baOGsUnxIzdl9aBxmRjyrSKPmHosATkD3C
6SI5manaJAo+AAAAADAJH26dVZMdFSth21IUg9m8TVK1zs+VS/fBrNjAMFnwF/MTtBgUfABAAAAgEmSE5qpR+c
OOpSbV+v9NFeFIm2cvtqYtF06bS9bXYMIPiOHiIl9zA7RYNFwQcAAAAAJrFYLBrQu7u6ssFLaL1J1Zf
I2GxfBoZtgMOQOOSAHq3EBm7xOLBR8AAAAAmKh7x3Y6qXWqDmTl1Hq/Q1iZwqzeAKfC7/FA6Btq7dtndgwg+CR
2lzqda3aKBo2CDwAAAABM5HCE6MxT+quyskpuT83luHZm8TUKgyObdaE+NzsGEJyG/p9ksZidokGj4AMAAAAAk
/Xv2VUtkxOVmX2o1vudw8rkYhZfgxVmlGpa6POyif+PgDrXop/U8WyzUzR4FHwAAAAAYLLwsFCdeUp/FReXyuv
11bhvtxrqHl5qL8Fs+Ez1S8L9fsGEBwGvp/ZidoFCj4AAAAAKABGNinhxKbxyn7UOl1FUZfwUjmZxdfgXGhbr
NN935sdAwhOrYdIbYeanaJRoOADAAAAgAYgNjpKpo/sp7yjHPbRnOHYaAEMPsMKNyhdgXrRbrjUepDZKRoNCj4AAAAAaCAS4
uNO2oA+yj2cX+ssPptF6h7HLHLc1I8+wYMQjnCXcj4Gza73fNrRCCSF
VAU6Fn+ui3brW+r7ZMYDg1elcKaW32SkaFQo+AAAAAGhAkprH67ST++jQ4TwZhlHrmAHRhQFOhR/ZDI9eCJuhE
LnNjgIEJ4tVOpOTc48XBR8AAAAANDCnDeyn+LgYZR06X0v9RIdbaa7yAKeCJP037H9q4TtgdgwgeHW7QErsYna
KRoeCDwAAAAAamOSEZho+ZKBy8/Ll8XprHdM/qkhW1T7DD/VjmHW1zjcWmBODCF4h4dJZ/zY7RaNkNzsAAAAAA
KCmoYNP1tJV65SRaW01BY17kfZveoaXqoNpREmpGt6IoOSPeqaKWsth5/gxM1YUaUZK6u0t+DI57drgk13neb
Q2e1DqoOzDEPnvFameTu9ev+iUI3pFFLbOOmSLP+u/UCah4c7dctgpySpzRPF2ldYvSh/cJhTtwO5cn9vgU+Xv
1+uVQe96ptsOytjQ9Um5qe5Uue+VqY/9QrR+C5Hz4HjcNo/pKgUs1MOShR8AAAAANAARUdG6Owzh+iF199VRWW
VXE5HjTG9Iou1vTxUlT6bCQmblufCn1eMr/aDT3DiWkZZ9NBwp9rHHZmXOnutW+e/Ua4111rVNeGnv99PfF8li
yy/6TkP3ly9/P5shOeTPqyoUcbde4ZTV/f96Vqk46fnv/mLCrWIsmjmH8J1x1eV+scXFXrnwjBJOpsb3bJaRLl
XV+LbSYMmm52iOWKJLgAAAAAOUKf07anO7U5S+v7MWu87rYZ6R5QEOFXTc3nIQp3iW2l2jKB2XscQndM+RO3jb
eoQb9P9w1yKcEjf7/9pifraLK8eXVqll853/abnTIqwVnv7YJtHZ6bZdFJs9Sok0ll9bPjPCr4th3y6oqdD7eN
turJniLbkHplhWFBh6I6vKvXOOb8tC36DUVMle81/yMBvQ8EHAAAAAA2UO+nQ6GGnymKxqLiktNYxXcJLFWvnR
Nf6kurLOu32OWbHaFK8PkNvbHSr1CONSjOye6/MbeiSd8v19DkuJUUcf5WRXeLTJzs8mtS7ZoHOOLdVin+4WL2
fK9EjSyrl8f2OZLdnklVf7vbIZxj6YpdXPRKPvPYtX1To+v4hSo2mVqkTHc+R2g83OOWjxhJdAAAAAGjAenXpq
L49uuj71evVuf1JsliqLO+OWqTTYgrOYW4zGb9x6SJ+u5kRzyjUx4nFgbAh26tBMOtV4ZEiHNL7F4WqS/MjBd9
N8yp0SqpN5x9jz71jmb3OrUiHNK5z9RrkxgEO9Um2KS7Uou8yvLp9QYUOlhh6bOSRmXn/Pculaz+uUJsnStQj0
abnznVp8T6P1mZ7NfUsly58u0wrM70a0dauaWe75LDx3+DxMkLCZDl7qtkxGjOKPgAAAABowKxWq0YPO1Ubt+3
U4fxCNYuLqTGmucOtruGl2siBG3XqDufb6uDbaXaMJqNjM6vW/iVChRWG3tns1hVzK7ToSqt25vm0cK9Xa64N/
93P/dIatyZ2D5HLXr2A+/sgp//PPRJtctikaz+u0IPDnHLaLWoRZdXH14T5x1R6DI18tUKzx4TqvsWVinRYtG1
yhEbNKdNzK926YQBLTI+X5YzbpJhWZsdo9JhLCgAAAAANXNvWqTr15D7KOc4tvsliRNk+AkwWv3tqhK
y0fmR2jSXHYLGoXZ1XfFJseHO5Sz0Srnvy+Sgv3eLUrz6eYh4plv7dI9nuPnI47/q1ynTGr9qXrP/fNPo+2Hfb
pqj6/Xr4NaGGTxyf/ab6/9MA3lRrR1q6+KTZ9vdej8V3sCrFZNK5TiL7ex39/xy2xmzTwerNTBAVm8AEAAABAI
zDi9EFasW6jDubkqkVSQo37DuhITEF+uxwMxPSBZcQo1IzwmfI7qOwMZPPkCq9Or/PdOiqPtWX5nafUarHRzp
1XodfX7I7c41bfZOt6pnO66dNr83yyWqREsJrzofacsir1zZ6tPaHmYReQ3L/cAaI22fIW3sniKMwLFZZzn1Cs
lFN1QVm8AEAAABAI5DYLF6jzhisgsIiVVRdtenwqvmqpuCbZqb5LUKtqqtJ+diNtpeone31L9wJmiSkNvb3bXOntvaYZHT3xfqXVZXu3O92nOe

rdu+rxCl/YIUWxo9aW8hmHomo8r9PhIp/+U3cGpNr2w2q0th7x6ZZ1bg1N/vUDETyx9/ySl9jc7RtCgJgUAAAC
ARmLo4JO1dtM2bdq+S53apdU4cEOSBkQVaX+FS2U+yobfY7R1uUYZi8yO0eTklBq6/P1yHSwxFO2OqEeiVZ9fG
qaz2v722mLbYZ8KK41q197Y6JZhSBd3qznTz2m36I2NHt3zdaUqvVJajFU3DXTo74NqloHPr3IrMdyic382Y/C
eM5y65N1yDXixVKPa2XX9yey/91sZMa1lOevfZscIKhbDMIxfH9b4FBUVKTo6WoWFhYqKigr467+37WDAXxNA8
BnXMdnsCAOOX18B1BWzvsaa/XMqGr9tu/bq0edfkcvlVEJ8XK1j9pS7tCC/9ns4ulhfvhaF/OtRRqHZUYCgZVh
ssvzpU6nVQLOjBBWW6AIAAABAI9KxbRuNOG2Qcg4dlttd+x5xaaEVauMqD3Cyxm9mxPOUeOA9s5z6d8q9ekDBB
wAAAACNzKgzBqtj2zTtSd9/1DGnRBfKaWHX/9/qLyGfqY9vndkxgKDmTeopnX6b2TGCEgUfAAAAADQyEeFhGnf
2MNntduUV1D7jLMzmO6kxBYEN1ki1VYZusr9ldgwgqPlsLtn++DKn5tYTCj4AAAAAaIR6dG6vMO/pr8ysHHm83
lrHtAmtUJfwkgAna2QMn14MnyGnUWl2EiC4nf2QFN/W7BRBi4IPAAAAABohi8Wi8846XWmtWmpvRuZRxw2IKlJ
8SFUAkzUu94W+rjTvXrNjAEHNO/YsWfv9yewYQY2CDwAAAAAaqZioSIOdeaYMw6fC4tpn6tks0tDYfIWwH18NA
y2bdLHmmRODCGoeV5zs454zO0bQo+ADAAAAgEasX8+uGtyvt9IPHJT3KEt1o+1eDYnmdNifCzXKND3OedlU++c
MwIkzZJFt/PNSeLzZUYIeBR8AAAAANGJWq1Xjzh6mtq1aate+o5+q2zasXB3CSgOYrGF7OuwlNfMdMjsGENR8f
f4kS/uzzI7RJFDwAQAAAEAj1ywuRhPOHyWnIOTZhw4fddwpUUWKtbsDmKxhGm9bojON78yOAQS1qubdZTvnIbN
jNBkUfAAAAAAQBHp07qDzhp+u3PwClZVX1DrGbjUONDZf9ia8H1+C77DudcyWxewgQBCrcsTKcdnbkt1pdpQmg
4IPAAAAAILEqDMGaOCvbtq9LONeb+OlXmyIR4Oa8H58MyNmKNyo/UASACfOK7tsE9+QopLNjtKkUPABAAAAQJB
wOEJO8Ziz1apFsvZkHDjquI5h5erUBPfjm+L4UN19m82OAQS1yuH3ydZ6oNkxmhwKPgAAAAAIIonN4nXRHObJZ
rXoUF7+UcedEl2oFs7al/IGo07aq79a3zM7BhDU8k4ao7Ah15kdoOmi4AMAAACAINOnWyedfeYQZR86rPKKylr
HWC3SsNh8xTSBQzdshkcvhM2QQ1VmRwGCVl5EJ8VOnGl2jCbLbnYAAADQtHi9Xr01/VEt/vBdFeQeUmxCos4ce
6EuuG6KLJYjW56P75RS62Mvu+UOjZn011rvlZeU6PVpD2vZl5+p6PBhpXXuqj//33/UrnsvSZLH7dbrT07V6kU
Llb1/n8IiotTjlFN16d//pbjEJEmSu6pSz9zxD61Y8LlimiXo6rsfUM9TTvO/xtyZzyg384CuuvP+OvyMAEDds
1gsOnf4adqbkalVGzarc/uTZLXWnN/hsBoaEZenD3ObqcJnMyFpYEwNm6NUX4bZMYCgVWKPVcRVc2WxUTOZhc8
8AAAIqLkvPK3PX5+tGx56UqntOmrXxnWa/q+bFBYRqdGXXyVJevGbtdUes2bxQj1zx80aOGLOUZ/3mTtvVvqOb
bpx6lOKS0jU4g/f1b//dJGe+ORrxScmq7KiXLs3b9AFf52uC89cJce+uuVevjdeZKk+W++qt2b1uu
BNz7SmsUL9cQ/rtdLS9bLYrEoe3+6vnznDT34zsexTDnkPakH1DbNqm1jouyezU8Nl+fH
o6XLwjPlj3DslbjjPlmxwCCllsh0kWvyRHDoRpmYokuAAAIqG1rVqr/sJHqe8ZwJbRM1aBR56rn4NO1c8Na/5j
Y5gnV3pYv/FzdBgxWUmrrWp+zsqJc33/xqS7/xx3q2n+gklun6aIb/qGkVm30+euvSJLCI6NO90tvavDZf1CLk
9qpQ6++uur0+7Vr03odytwvSdq/e6f6DR2hVuhF+XmSp0fvuU2X/eP/FBYRWb+fIACoQy2SEjT
h/LNls9lOMCf3qOOSnFU6NaYgcMECJNwoOeOhL8qq2k8UBnBiDEMqPvNBRbQ/xewoTR4FHwAACKiOvftpw9Jvl
blnlyRp79ZN2rp6uXqfh7R6OQINGz/hqM/p83jl83oV4nUg4OXhUdGSpd/hM2m+/VmzzREXFxmnxR+8pxGOHF40/e5iKioqVX1h01HHtw8rVM6I4gMnq3
3PhLyrWl2d2DCBo5XS+QnGnX212DIglugAAIMDGXjNZZaXFuvGcO2S12eTzenX7S6Ngmah3/9dy3FBoeoQE
jzjnqc4ZGRKhjr75655kn1PKk9opu1lzffjJX29euUlKrNrU+pqqyQq/+934NGNGs4NGT3GPytv6PgJ2rd9s6aMPkORs
XG6+YlnVVJYoDemPaJ7X3lHrz0xVUs+/UCJqa11/QOPKT6RpSgAGj6LxaKRZ5yi3i3PwCfbLgGzkdDoWFumod2y+
yWEUeu/ZUhAY4Zd2baP9aQ3xH/0ceACfmQPIIpVz0pNkx8AMKPgAAEFDfffahvvnoPU3579NsKbddRe7ZussP3
O0/bOOXFrz7hk49d6wcztp/Gf3RjQ8/paf/ddPafXXddfXofWW02ndSlu4aMHqNdm9bXGOtxu/XolGtlyNA19zzkv24
PCdHVdz1Ybez026fonMsmac+WjVq+YJ4enful5r74tGbed6duferF3/lZAIDAslqt+uO5ZymvoFBLPEX
gmp+eugxSKdHpuvklybDrkdJiStGylGtv4v5FXJMDsJEJzSI/sq5c+v+Q9Ig/lYogsAAALqlUf+o7FXT9aQ0WP
UumNnnXXH+BTrvXv3NPf1+Ri7eeUyZe3bplPHHB2VcWy31BP/jwb3LeEtYgwC2ehzr+Q3HWWWUq6u+Bp7X+uU+LPG0LVqCn7+kBkOJhEQL
6f7Hc/W9q2eP7HUUGFD1PW9c7ZLu+bfF7eUdgtTfb94vAHlkRmf32Cu/bgLCfR8+9H9jZwbtcqdFIeUUl5L4aB88sBK4/YHkUv7oO7Z3d5jO87OLQqhR
z2eTzeeX1eCRJXo9bPp/3N2cCgIYiNjpKf/75OjfFokJW2fHH1Z0Iyz5bnO8q06+7WbHAILSf
qUo9PK3FB599J+hYA4KPgAAEFD9zjxL7z47Tau+/lI5+zO09Ew5bOBP5n+mjWcxpw1qhq48pKirX084807Ciz9+658kJ
9+upL/vfXfPO11nzzlbL3p2vdkkW6+4oL1OKkdho67iJJR8q9//7tau3auE5THpkun9M/EM5yj+UI3dVVY3nf
/uZJ9TntKE6qUt3SVKnPv3l/fxPtXfbZn0252V16t/OjBBYqSlJuuKP5yk81B9B7OPOi7U5tM58YcVZfM
EMN2J6aGdmmT9wOwYQFDK9MWpcswLap7a1uwoqAV78AEAgIC66o779g9Pqh/3Tt
XHffvKBDMPQkNFjan2erPS9Ks7/6WTI3nsQR3OOqiImBgNPOscXXLTbf7lI3nZWVqx8AtJOs1jzqr2XP+
e/Y66DTjF/3769q36bt5HevT9+f5rg0aeq03ueO3Ll+rOiWOVktZhvHgDATN07tddFfxil9+cq9y8fDWLi
611XLjNp7PjD+vjw/Eq9TbsXx/tRpWeC5+hEF/jKSSBxuKQN0r5I55S115DzI6Co7AYR5uT3cgVFRUp0jpahYW

FioqKCvjrv7ftYMBfE0DwGdeREzp/ia+vAOqKWV9jzf45FfiRYRh699Mv9d68BWqRmKDoqKPvSVrosemT3GYq8
9kCmPD4POV6UedpodkxgKCT7w3XvlMfVc+zJnCoRgPGEl0AAAAAaIIsFovOH3GmRpx6ivYfzFZxSelRxObbvTo
7/rBcloa5/+hI60qN1ldmxwCCTrHPqd397qHcawQo+AAAAACgiQoJseuSMWdr20AB2rs/UyWlRz95NjbEo7PjD
8tpqf30XbPEGIV62PmSrArKxWmAaYp9Tm3peqt6nTuJcq8RoOADAAAAgCbM4QjRpeNH68xB/bQn/YBKy8uPOjY
+xKOR8YcVOoBKvhfCn1e0UWB2DCCo5HlDtabDP9R3/BTZbA13aT5+QsEHAAAAAE2cy+nU5Rf8QacO6KPd+/arr
LziqGMTHG6d0OCW614V8oX6+9aYHQMIKjmeCK1qOOWDLvybbPaGfbg0fkLBBwAAAABQWKhLV/7xDxrUp4d278t
QRWXlUcc2d7h1brNcRdjMO7G2tZGpf9jfMO31gWBOwB2tFa3/qtMuulEhDqfZcXAcKPgAAAAAAJKkiPAw/fmiM
erXo6t27E5XZVXVUcfG2L06r1muYu3uACb8geHTS+HPyGUcfaYhgOOzpypOy1pcpTMvvkHOODCz4+A4UfABAAA
AAPyiIiN01SXj1LtbR23fvU9V7qMXeOE2nOY3y1VCyNGLwPrw79C31Na3O6CvCQSzbZXNtSzpMp11yWSFRUSZH
Qe/AwUfAAAAAKCamKhIXX3JBereqb2279qrqqqjl3wuq6Fz4g+rpTMws+l0tmzRRHOakNcCmoL1Fcla3+rPGn3
5FEXGxJkdB78TBR8AAAAAoIb42Ghd0/EC9ejcQdt27VV5xdELPLvV0Ii4PLUNLavXTC6jXNNDn5Nd5u39BwSTF
WWp2t3+Go2+bDLlXiNHwQcAAAAAqFXz+Fj99YqLNKBPd+3cm6GS0qMXeFaLdEZMgbqGl9Rbnmlhs5Tgy6m35we
akm/KTlJ2j7/q7Ev+wrLcIFDnBd+DDz6o/v37KzIyUgkJCRozZoy2bdtWbUxFRYWuv/56xcfHKyIiQuPHj1d2d
na1Menp6Ro9erTCwsKUkJCgW265RR4P/0oDAAAAAIEUExWpaydeoDMG9tPe/ZkqLCo+6liLRRoUXaT+kUWSjDr
NMcb2nc4yvqnT5wSaIrfsmlvSU2X9JmvkhZM4UCNI1HnBt2jRIl1//fX6/vvvvNX/+fLndbo0YMUKlpaX+MTfdd
JM++ugjvf3221q0aJEyMzM1btw4/32v16vRoOerqqpK3333nbPnq1Zs2bprrvuquu4AAAAAIBfER4Wqj9fNEa
jTh+sA9k5OpxfcMzxPSNLdFZcnkIsvjp5/XgjX/c5ZstSJ88GNF3FCterRScresg1Gjb2coU4nGZHQh2xGIZRt
/+s8guHDh1SQkKCFi1apNN0O2FhYVq3ry5XnvtNV1wwQWSpK1bt6pz585aunSpBg4cqM8++OznnnuuMjMzlZi
YKEl69tln9c9//lOHh2Sw+H41dctKipSdHS0CgsLFRUV+Kmm7207GPDXBBB8xnVMNjtCg8PXVwB1xayavsWb/n
AqcCI/Ho/fnLdRH8xcpKipCSc2bHXN8vtuuL/PiV0i1n9DrfhD2gHr6Np7QcwBN3QGjueYW91LPERdr0FljZLP
ZzI6E0lTve/AVFhZKkuLijmzWUrGrVKndbg0fPtw/vFPbrv672Vp9ErR1lO6dKm6d+/uL/ckaeTIkSoqKtKmT
ZtqfZ3Ky6yoVFRVewMAAAA1B273a7x5wzXhPNqOb3xhPHqbS0XBmZWcccHxvi0fnNDyn1BE7Ynez4hHIPOEEbvCfprZK
BOvm8P2nwyHGUeOGoXgs+n8+nKVOmaPDgwerWrZskSsrSw6HQzExMdXGJiYmKisryz/m5+Xej/d/vFeb6dOnKzo6
2ucXpq/y3bjmp/fftSjhtMh
NHR0f631NTUOv5oAAAAAABWq1VnnzlEV/zxDzIMQ7vT9+tC5yw2yuiWMe7L197petG69snmBhounyy6su
KHlpQ1VtDL/iz+p1xjiwWFrsHo3ot+K6//npt3LhRb7zxZRn2+jCTp9p9ttvIyOj8lSs8j3l8TAAAAAJoii8WiM
wb1017UzUXj1Ooy6ltu/bK6/UeY/zPkL6pYw2Lzf/O+fBbDo5lhz8ihqrqKDTQpFRaXXivsq1lhvTV64nXqOWgo5V4
QO7GNEI5h8uTJ+vjjj7V48ux0pKUlVVUqKCioNosvOGAxSv0OztbSUlJ/jHLly+y+vnjL745FLxrUvr2Ih7ciIi4Or8AKgo+c1AJFnpK3X4H8zOkzmlcM4J4DjCmeYBbz/qxmvOG5N+AoG+VKZqfBpOe55dXL+kxFmQ0dpC3D3s8mRUcq5SpMG93bB2n
jFxUVmHl9T4EOvbqt53f9ZsGh8K7BTOWxG7+cIb6f3e3L24wgq5KbAtkKq5M6sUQrqalSTaWJHA93XA0AqVjPFTX
lJ1zKrlrj/Mhyo7Op0FOUkq8ecmAcKvwQ+jxhmCICe36tzhVr2oIStdlT9mZRLmlVVtGEvOOUIdcFM73Ebf+/tMuUEAj
wtDTLhIWRh2WxWDkuhaKfbRmdXjRY0ZlCwVbjmHOZO7GjbRHz7xwJ0cNlSXQRw5T0yEJBtgxV1CmbFEbXsC6fs+CzT
Wy51uc2xYZ/ckTxQHeFhpPBOYOqqFP9tyI8X3Cp88njQOtvnLSUjX4FeMXA37WbQ2Wbqn7jBsCO7YrY2WrZ07Gpc
mvqmZRdxwo5ws1tmhrLpHlK7c5m0U+zqfxkcJ8N2KqRPSaK7Gx/zNQpkZmB+h/m4wIG3tS0CsJfBZaW0f/GwM1hTv1
9S3wC6ahxoi0lORJmOn4qhgX0HHSfn8c7ggu9TJDH2OKmHugRvRXOdjQZ7kH

2ixaI+rm/Y72qln2bdqV7meZbto1LyyaWtoX60L6afMA/vl82Wqx4DTderoixQd19zseGgEKPgAAAAAAKaKbZ6
kcy+9Xm279Nb3X36gjB2bFJuQrKjYZsecteS2urQyYrh2ubqrf+mXaubJCmBqoG4cDGmllWFDlVHgUUHuDiW3b
qtBZ41Rh54DjrlsHfg5Cj4AAAAAg0lsdru6nXya2nTsrhVffaJ133+ljMOblZh6kpyu0GM+Nt+eqC+iLlFq1Xb
1KP9O0d68AKUGfr/DtkStCxuivd7myt69R2ERUTp19EXqe9pIhUVEmROPjQwFHwAAAACgwYiIjtUZ509U++79t
HT+XO3eslYOV5iaJ6fK+ivLdjOcHZXh6KA2VVvUvew7RfoKAxcc+I2KrLFaHzZYe2xtdehgutxV6erQo79OGTl
Oya3amhOPjRQFHwAAAACgQbFYLGrZtpPGtrpZm1Z+o2ULPtK+HZsUn9hCUbHxv/Zg7XV2OT5HJ51UuVHdyr9Xu
K84MMGBYyi1RmhT6EDtcnZXYUGe8rI3K6FFaw0cfr469zlFNjsVDX4//vYAAAAAABoke0iIeg4aqrSOPbRs4cf
auHyRCg5nKyn1JDmcrmM+1rBYtcvVQ3ucXdSuYr26li9XqFEaoOTAT8osEdocerJ2urqrotKt7F3b5HCFasDw8
3XymaMVGRNndkQEAQo+AAAAAECDFhXXTMPHX6GOPfvru8/f197tG+QMDVOzpNRfnfXks9i1PbSPdrm6qOPFWnU
uXy6XURGg5GjKyizhPxR7PVTlNZSbka6qynK16dhDg0eNU8uTOh3zEBngeFDwAQAAAAAaPIvFolbtuyq5dTttW
vmtVi2ap/27tigOIkrxSS1ksx3711uvJURbQvtrh6unOpavVoeKtczoQ7OossZqu6u3drm6ye2zKvfgAZUXFyo
x9ST1O/1sde5zikIcDrNjIshQ8AEAAAAAGoOQh1O9ThmmDj1O1pZVS7Tqm8+VsWOzIqJjFZeQcuyDOCR5LA5tC
huoLaH91apqmzqWr1G8NytA6RHMDoa00jZXX2WGpMlnGCrIyVJRQa6aJbbUqaMuUJd+QxQaHmF2TAQpCj4AAAA
AQKMTFhGpvqePUqc+g7Rx+WKtXTJf6Ts3KTImXrHNk2W1Wo/5eJ/Fpr3OLtrr7KJm7kx1rFit1KodssoXoI8Aw
cAju/Y6O2ubq48K7c1kGIYKcrNUcPiQYpol6PRzL1HPQWcqIjrW7KgIchR8AAAAAIBGKzwyWgOGnacufQdrw/J
FWvvdl9q3fYOi4xIU2zzpN+1xlhuSotyQFIV6i9W+cp3aVayXyygPQHoOVmXWCG139dJOZw9VWUOPFHuHslR4O
EdRcc005OwL1GPgmYptlmh2VDQRFHwAAAAAgEYvMiZOp4wYq679hmj9919r3dIF2rdtg2KaJSo6PuE3FX3ltki
tDxuijaED1aZyizpWrFGs91AAOqOxyLUna6urjzIcHWRYrPL5fCrMzVJhbo6iYuN1yqjx6jnwTMU2TzI7KpoYC
j4AAAAAQNCIjmuuU8/5o7r1P1XrvlugDSsWa9/2DQqPilVs8yTZ7SG/+hw+i127Xd2129Vdzd37dVLlRqVW7ZD
DqArAR4CGptwSpn3OTtrj7KJ8+5EZeR53lfJyDqqsuFBRcc1Oyoix6jFoqOISkk1Oi6aKgg8AAAAAEHRimyfpj
PMnqsegodq2bpk2LFukzD3bZLM7FJ/UQq7Q8N/OPIdCWupQSEutMIarRdVutanaopSqPbLJW88fAczkVoj2O9p
rj7OzskNaybAc2dOxvLREeTmZ8nk9apaUqsEjx6lDz5MVHdfc5MRo6ij4AAAAAABBKy4hWYPOGqNepwzXrk2rt
X7Z1zqwZ7u8HrdimiUpMibuNy3f9VnsynB2UIazg0J8FUqt2qE2lVuU4Nkvq4wAfCSob15ZlRXSRnudnbXf0VZ
ey5HZnoZhqDg/V/mHshTicCq1bWf1GHC62nbrK6crtF6yXHnllZo9e7YefPBB3Xbbbf7rc+fO1dixY2UY/J1Dd
RR8AAAAAICgFxoeoW4nn6bOfQcrfccmbV61RDs3rlJe9gFFxsQrpnmibLbf9iuy2+ryL+EN9RarddU2tancojh
vTj1/FKhrhqRD9hTtdXZWuqOjqqw/FXZejOf5h7JUXJinyOhY9TplmLr2G6IWJ3WSzWar92wulOtTpO7Vtddeq
9hYTuHFsVHwAQAAAACaDJvNprROPZTWqYdys/Zr65rvtWnFYu3ftVUhTpfimifLFfbblu9KRw7m2BraT1tD+yn
Ke1iplTuU7N6rZp6DsspXjx8Jfi+P7MoJaamDIW2O39FOpbZo/z3DMFRRWqL83Cy5qyoVl5Csvqddok69Byg+s
UVAcw4fPlw7d+7Ugw8+qIcffjigr43Gh4IPAAAAAANAkNUtqqSFnX6A+p47Qjg2rtOH7r5r5R9YK+qKsoVFhmt6Pg
EOZyu3/x8RbZ4bQqL1yYNVIivUonudCW79yrZvVcRvqJ6/EjwawqtsTroSNPBkDbKCWnpX377o6qKchUczlFZS
aFcoRFKbtVW3U4+Xe2791NYRKQpmWO2mx544AFdcskluvHGG9WyZUtcqBxoOADAAAAADRpYRFR6jnoTHU7+TQ
d3LdTe7dtOLa1y5RzYK+8MUFddc9pBfP4H3R26rU/ud7bXf2V6SFOU9rOSqfUp271GCe7/s8tTXhwNJb
otDWfZUf6n38l6P/K4q1SYd0glBXmyhzgUn9RSJw89V206dldS6kmyWq0mJK9u7Nix6tWrl+6++27NnDnT7Dh
owCj4AAAAAAADQkRlTLU/qqJYnddLdCvXZuvCvXZuWKHMvTskGYqKa66omHhZj3MPtiJbvIpC47UttI+sh
kcJ7v1Kdu9Vc0+mYj05nMp7gtwWh/JsicoNSdbBkDY6ZE+RYan5/5HX61Vxfq6K8g/LYrGqWe/LYrEoplmCBg4/X2mde6p
FWkeFOBwmpD+2qVOnaujQofrHP/5hdhQ0YBR8AAAAAAD8QojDqbDqbROPZXWqaf+v727iY37rvM4/hnP2DO2xw9xH
p0mIWzapqWERKXQ7e5SaIcO0IcUURQsqFiB5oekHQAoIeeDj2xpFLz4hKsLuoVQttSdu0eapjx0+
xx/bMfw8JpiFJtw9JZv7h9ZKsGf/GM/P1ZWS/9f/9///9vTrz+cqrVWkYnNqc5Ov6hY
1+nUsvpgd05PbA7SVIp2hlvT2Xj2ulMrJ3OxrUzGWtPuTrvNbRTzfnqppyrbbv0NZm56kRyjasht9trWZw7n7n
pqbQ77YyOTWT/A/+e2++5N7vuuCf1waGb/Bt8OA8++GAOHTquUIOe05LHHHuv2OPQgzQ8AAAAAAEfL+vr7P/Sr3zM1A
+/P/PnpvPnqS/nzH1/I9/IO3/5U97+y+mkSIZGRtMcmh2bh2+lL
4O5ORzkw+3CuXX5FkrjqxHv0mq9syU9ucTuXaOaMoiixfWMzC7HTQKmkOTqeOz5zX+7Y97+nsvvVaTr7xao6/8
6q0x44dy4EDB7J3796bvJ/mbNmLB8Amq4N+Nm/GRsMcgb8H/1a/SGn3Ndz1B8B3ZfQKa9aRTjl8r9h7fqnzpIvV8NwyVVbR17SDgC35zj2l7bGlWW8YD2m
wXEEszlSvydkGT4XvTkH/yWLIrelbc/ZpxkiAYrVjzJYDVwPiufG7pqh27WzvN/EwbCnxqyxLmRtPzprF
2IcNjG1KrffDz9r1Xu9Kfqf7bMtX/tyu39neWM9qeSbNzPiPt8xnuzGakfT7N9mwGi4XSxr90Klnqa2a+bzwL1
fHMV8ez0Hfptjqetcr/v212bXU1C3MzWZidnt1NfWh4Yxv3JL9D/xHbtt9RyY/cpot577du5kiefrv9dd/cXvpot577du3L4888kief/75D7HHaYgUkwA8JCPF7tdnAFfaWZ
xqXC3GYRPJLLTbOxAJMAzokMG/4VvbuAHexDYpmM5JJymQjH2BXRLg5Uzo2JSjcLgpVCKZ+aU1THIMK3HVizWnrN/r3a2b3b
pbo9CjxL4AAAA4E0qVCoZndiU0Oy1NuevAP6e9tpazJ9/Mq9z/V4tqBa7OZGUXbaQzLNGdQKGm0N/OXk+bPxwVHyM5/ihyaCF10kXEuSrMb/5+kGj6vcy2VcJM/YcKuFYdbenNRFtHYqoWx2pNz08prxgjpkHbyuR6NeTUm8lwFVZKXvJtp8wmk6cDs8k6EM1eZmXNNJx8I7/OxLC/QXG+pP52DUumCqWQlA6QkSgsjP9J6f
A6PQEWBGbJPIrRrkwXQlk1y6z4M6ExeKOAdJLqEWaocdmQ2rIvnn3RyFCrufpVsVbkxKHvWc13JDutTh
2VbFwEJ6KrvgoRsX5AoRVtCmHgV5dVAdbOxgmo9VgMAeqewebCpuaW9X0cfDqGKQwALyrBYXW36HpJxA2gnCZJJcTRTArsRTU7U2z8DvsWvcu5jW0GZNk
xGJRvs1EtCAeS1mt00fKnN0NPVQSnFFmrLD2UnimFlDFw
wGNVOPzSxYnD1x7pd+2L0U8OQ4F5mk1n0m+6xXHzWsPDhfzLWRxZ6RQjfz6kYTXvBZh0ImWrNMvHx3Pj4AsP7L6hP5iKrdoJbWuVbFVd
+Jbl3A6uQJmj7R9RtUs3YSf3LO7Gjyd5jnHyZmDUd+9bkjfHZ+EhD2Bp5H4/2n4Rz3+2wk6NdU6NVmHIRNpCFq0QW+p1pm6EMDtyc3FLAC8QgE7p1DKAqp2WO7nBrHVFVSvDVw/+c9Qq0I3X4vu0Nwa4rBl8+jFDd9Ss8h4dtKvzG5y7PHjk4ht9v7ZH+hkp8qyf/ItZgq2b7Q7Zh9YDwXKx0uAqE0s1WsMdQQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

```
alf7M1jZlNpuu+TO1YiX1zlIanQsZKFqpFa30FyvpL1YycOl+X/H3V/f922xF5cq1Tvouxru+xnrEa12636o0s
lqpX/PceB9UURRpLV/I0uJ8lhbms7rSSpIMDo9k49bt+ae792fyE7dnctftGWqOfKz36hW//OUvr1jbvXt3Wq3
WzR+GUhD4AAAA4Dobao5mzz33Zs899+Zf//O/cu7MO5k6dSLnzpzMyTdfy9SpE5k+ezIrreVUUsnA4OB69Osfq
F+X6Pf31ioDWasOZLE6dt1f+3opiiKrK62LMW9xPq2lpSRFBuqDGWqO5JN37c/23bdnYstkNm3bkQ2bJ9PX19f
tsaHrBD4AAAC4gfr6+rJ5cmc2T+5cX2stXcj0u6dy7szJnDv9Tt5548+ZOXsqUyffyupKKK5W+vgzUBzPQGMxAv
ZGBxuANC3/dUBRFVlrLWWktZWV5OSvLS1ltLadIkf7+gQw0j2Ry157ctvv0bNp2Wya2bs/ElU0ZqDe6PTr0JIE
PAAAAbrL64ND6lt6/urAwn+kz7+Tc2V0Z0v12pk6dyPlzZ7K8tJi5mamsrawklSJFkfQP1C8FwMbF23oQfT12v
rlOu52VlYvxbmX5YsxbXWld3LFbVC7+Do1GGoPD2bbjk5nYuj2btu3Ixi2Tmdh62y2z3RZuBoEPAAAAesBQcyR
DzbuyY89d62vtdjuc+fXrxC7MHs+87PTmT5zMtPvns7S4wuLMxlZXkpGsx7NULp4XsFrrT7Va4V7VaS7VaS1+1m
r5qLdVLt++3tbVIceVap0i7vZb22mraa2uX319bXX//4tJTK5XKeoQcHh3Pzs13ZWLr9oyObOxzdDzN8YmMjE1
kaGTMNlv4mAQ+AAAA6FHVajWjGzZmdMPGKx4riiJJLiwtZnJvJ/Ox0lhcXOlpeunhBiguLWZw7nwvzs1m6sJCVp
QtZW1vN2morK8vttNvtFJ33Xl3Glt/K++9W7kUDWup1frTHB2/eN7A5kiGm2NpDA1f2lI8mHrj4rbi5uiGNMc
mMjjcvGW2FOMvEvgAAACghCqVyqqWj/kayefuu9/3ZoijSXlvL6vHoXKre6lrK62O1772xVzrxrf/m6tr68v/QON1
BuD6a83HHUHPUTgAwAAgFtcpVJJrb8/tf7+DA43uzuzOcJ3J7QAAAABQYgIfAAAAlJjABwAAACUm8AEAAECJCXwAAAAAQIkJfAAAAAQIkJfAAAAAlJjABwAAAAAFBiAh8AAAAAlJjABwAAAAAQIkJfAAAAABQYgIfAAAAAlJjABwAAAAAFBiAh8AAAAAlJjABwAAAAAQIkJfAAAAABQYgIfAAAAAlJjABwAAAAAFBiAh8AAAAAlJjABwAAAAAQIkJfAAAAABQYgIfAAAAAlJjABwAAAAAFBiAh8AAAAAlJjABwAAAAAQIkJfAAAAABQYgIfAAAAAlJjABwAAAAAFBiAh8AAAAAlJjABwAAAAAQIkJfAAAAABQYgIfAAAAAlJjABwAAAAAFBiAh8AAAAAlJjABwAAAAAQIkJfAAAAABQYgIfAAAAAlJjABwAAAAAFBiAh8AAAAAlJjABwAAAAAQIkJfAAAAABQYgIfAAAAAlJjABwAAAAAFBiAh8AAAAAlJjABwAAAAAQIkJfAAAAABQYgIfAAAAAlJjABwAAAAAFBiAh8AAAAAlJjABwAAAAAQIkJfAAAAABQYgIfAAAAAlJjABwAAAAAFBiAh8AAAAAlJjABwAAAAAQIkJfAAAAABQYgIfAAAAAlJjABwAAAAAFBiAh8AAAAAlJjABwAAAAAQIkJfAAAAABQYgIfAAAAAlJjABwAAAAAFBiAh8AAAAAlJjABwAAAAAQIkJfAAAAABQYgIfAAAAAlJjABwAAAAAFBiAh8AAAAAlJjABwAAAAAQIkJfAAAAABQYgIfAAAAAlJjABwAAAAAFBiAh8AAAAAlJjABwAAAAAQIkJfAAAAABQYgIfAAAAAlJjABwAAAAAFBiAh8AAAAAlJjABwAAAAAQIkJfAAAAABQYgIfAAAAAlJjABwAAAAAFBiPR34fvzjH2f37/t1pNBq5//7///8788/9///+DoKTO
b+J577rkcPnw4R48ezR/+8Ifs378/hw4dytmzZ7s9GgAAAAD0jJ4NfN///vfz9a9/PY8//ng+9alP//WmGh
oby85//vNujAQAAAEDPqHV7gKtZWVnJ73//+xw5cmR9ra+vLwcPHsxvf/vbqz6n1Wq11lWqtfz8705skmZubu7H
DXsOFhfmuvC9wa5mbG+72CD3H5ytwvXTrM/avf58WRdGV9wcAbjO9GfimpqbSbrezd
fLJJ/Pd7373ivWd03fekBkBAODjmJ+f6f9f+uj/vf8+Mdp2/2+Hf+EkEOhasCm3ncGzkq0BgHB37Mpvh3JaALS4/2D5PJtPZvfexM9DaAnN2u4QoHuLnyfgt
ubm4uO3fuzIkTJzI6OtrtcQBuGT5f6XVFUWR+fj/bK18fHxGzUiXDejo6P+AQW4AXy+0sscuQcAXE89eZGNgYGBfPazn83zzzzzzz+/vtbpdPL888889/ngQce6OJ
kAAAAANBbevIIviQ5fPhwz33zzzzx5/Oc/nx/84AdZXFzM448/3u3RRAAAAAKBn9Gzg+/KXv8v5x333033033/nOd
3L69OkcOHAgv/nNb6648ez9GjR6/YWg7Ax+PzFQCAfzSVoiiKbg8AAAAAQIkJfAAAAFBiAh8AAACUmsAEAAECJCXwAAAAQIkJfAAAAFBiAh8AAACUmsAEAAECJCXwAAAAQIkJfAAAAFBiAh8AAACUmsAEAAECJCXwAAAAQIkJfAAAAFBiAh8AAACUmsAEAAECJCXwAAAAQIkJfAAAAFBiAh8AAACUmsAEAAECJCXwAAAAQIkJfAAAAFBiAh8AAACUmsAEAAECJCXwAAAAQIkJfAAAAFBiAh8AAACUmsAEAAECJCXwAAAAQIkJfAAAAFBiAh8AAACUmsAEAAECJCXwAAAAQIkJfAAAAFBiAh8AAACUmsAEAAECJCXwAAAAQIkJfAAAAFBiAh8AAACUmsAEAAECJCXwAAAAQIkJfAAAAFBiPR34fvzjH2f37/t1pNBq5//7///8788/9///+DoKTO
```

```
      ],
      "metadata": {
        "id": "qW_OybNpFBzC"
      }
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "EIL3OoNaORW9"
      },
      "source": [
        "#### Gender"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "kDYesa73zUit",
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 507
        },
        "outputId": "738512e9-c214-42ee-d669-6332cb2b9102"
      },
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 1400x500 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAABPgAAAHqCAYAAACZYfCtAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAACHX0l
EQVR4nOzdd3hUZeL28XtmMpn0mSSQBgFCkd5BiCC9CbIoWFAsuCq7igV31/b+FF0bllVVRrKvrCigIdgUVRVRQp
PfeIQFSgPRkkqnvH6zjRgICJjkp38915drNOc+cuWcMIbl5nvY/OY/H6/XwAAAAAAABqJbPRAQAAAAAAACcOwo
+AAAAAAAoBaj4AMAAAAAAABqMQo+AAAAAAAoBaj4AMAAAAAAABqMQo+AAAAAAAoBaj4AMAAAAAAABqMQo+A
AAAAAAAoBYLMjpAVfH5fDpy5IgiIyNlMpmMjgMAAABIkvx+vwoLC5WUlCSzX9vBwAAv1+dLfiOHDmi50Rko2M
AAAAAFUpPT1fjxo2NjgEAAOqAOlvwRUZGJGSjrxg1NUVJTBaQAAAIATCgoKlJycHPh5FQAA4PqszfXz8tyo6KiK
PgAAABQ43AbGQAAUFm46QcAAAAAABQi1HwAQAAAAAAABQi9XZe/ABAABUBq/XK7fbbXQMQRJbrzOpu4lGmQ0H2vAf
Wq1UUi8XoGAAAoB6h4AMAAKiA3+9XRjYjqQ9ZZ7R27VPG8Ll3aEZbe5u9G/VvlZGGTo4
48/1iWXXBI47/f79dBDD+mNN95QX16e+vTPo1dffVWTtWrUKjMnJydHHtt9+u+fPny2w2a9y4cXrhhRcUERERGLN
p0yZNnjxZq1evZsVsOGDXX77bfrnnnvu+X2vFgAA4Az9XO7FxcUpLCyMSeDBs2kGCyTJu3FL//OMfNXbs2JPPO/B
fbpmzpypJQUPPRrsgHXNt7MCkSZM4Z2GhRYsGWzPr
264YbbtCkSZM0Z84cSVJBQYGGDRumoUOH6rXXXtPmzZt122232eFwaNKkSdX3xQIAANRqXq838AYyJibG6Dh
dDQUE1Sdna24uLiWK4LAACnnMv9/vP+cEmU7kZfH6/X0/+qv/3tb5/Px8xcfHa8aMGRo/fry2b9+u9u3ba/Xq1ere
udu3aafXqXq1erRo4ckaeHChhg0bpsOHDyspKUmvvvqq/vSnP8ZkZMhms0mSHn30Uc2fP187duwmOc
KqQ/q///sI/t7+aWsWKGuAs+FOnXtGAlpKQE/pH7Z/ycCgAAKlAAK1ul7ZZPacrYnQ+Oi66+pTtmpCn7ZZW+sSPMasi
rvy8/MVFRV1ri8QAAKlul7qK7f/9+ZWZQlkAfO0OnXgwAGlpKQE/pH7Z/ycCgAAKlul7qK7f/9+ZWZmasi
```

QIYFjdrtdvXr1OvLlyyVJy5cvl8PhCJR7kjRkyBCZzWatXLkyMKZfv36Bck+Shg8frp07dyo3N7cyIwMAAJwSy
3JxrvjaAQAA1alSN9nIzMyUJMXHx5c7Hh8fHziXmZmpuLi48iGCghQTE1NuTEpKyknX+PlcdHTOSc9dVlamsrK
ywOcFBQW/89UAAAAAAAAANV+lzuAzOrRpO2S32wMfycnJRkcCAABANRkwYICmTJlidAwAAABDVOoMvoSEBElSV
lZWuR3DsrKy1KVLl8CYn3cV+5nH41FOTk7g8QkJCcrKyio35ufPfx7za/fff7/+8pe/BD4vKCig5AMAAJXuo5O
Z1fp8Y1uf2y6sy5cvV9++fTVixAh9/vnnlZwKAAAANUmlzuBLSUlRQkKCFi9eHDhWUFCglStXKjU1VZKUmpqqv
Lw8rV27NjDm22+/lc/nU69evQJjli5dKrfbHRizaNEitW7dusLluZJks9kFRVV7gMAAKC+evPNN3X77bdr6dK
lOnLkSJU/n8vlqvLnAAAAQMXOuuArKirShg0btGHDkknNtbYsGGD0tLSZDKZNGXKFD322GP67LPPtHnzZl133
XVKSkoK7LTbtm1bjRgxQxQjfffLNWrVqlZcuW6bbbbtP48eOVlJQkSbr66qtls9k0adq0UezYSn3bPDTwVmmJ
XVKSkoK7LTbtm1bjRgxQjfffLNWrVqlZcuW6bbbbtP48eOVlJQkSbr66qsVHBysG2+8UVu3btW8efPOwgsvlJu
hBwAAgIoVFRVp3rx5uuWWWzRq1CjNmDEjcO7777+XyWTS559/rk6dOikkJES9e/fWli1bAmNmzJihh8Ohh/hTz75R
K1atVJISIiGDx+u9PTOwJiHH35YXbp0Ob///+e9y08WmpaVpzJgxioiIUFRUlK644AS/xdu3bJZDJpx44d5fI
+//zzatGiReDzLVu26+AQAA1DZnXfCtW
bNGXbt2VdeuXSVJf/nLX9S1a1dNnTpVknTPPvPPffrpph9ttVn27Nlq06aaNBg8erJE
jR6pv3756/fXXA+ftdru+/vpr7d+/X927d9df//pXTZ06VZMmTfq9rxcAAAKDOe++999SmTRu1bt1a11xzjf7zn
//I7/eXG3P33Xfr2Wef1erVq9WwYUONHj2630qJkpISPf7445o1a5aWLVumvLw8jR8/vtw19uzZow8//FAfffS
RNmzYIJ/PpzFjxignJ0dLlizRokWLtG/fP1155ZWSpPPOO089evTQ7Nmzy11n9uzZuvrqqyVJeXl5GjRokLp27
ao1a9Zo4cKFysrK0OhVXXFEu+5IlS/Tpp5/q66+/1vfff69169ZV6nsIAABQm5z1PfgGDBhw0g+nv+I0Mumm
RI488csoxMTExmmnNnzmmfp1OnTvrhhx/ONh4AAEC99+abb+qaa66RJIOYMUL5+flasmSJBgwYEBjzOEMPaejQo
ZKkmTNnqnHjxvr4448DRZrb7dZLL72k0OUuIXKzJkz1bZtW61atUatrnn3n3Urnn3++pBPLmcfNmvqrLKOnFLlc2bN2v//v2
BeyHPmjVL7du31+rVq9WzZ09NmDBBL730kh599FFJ0s6dO7V27Vq999574lcQHUQ79sIjJrbTt7lrnrV73e/W/1XSfk0B6q/0/u0Fr4SdJpU7VeBJ04qV6uzfZNdlqd1vfGpy5mXAZlHHL87zaVvhI0GWgmQyRV+31aDzG
nbt2qWkpCS9+eabeueddzR480By+QEAAOqrOrOLLgAAAKSd03dq1apVuuqqqyRJQUFBuvLKK/Xmm2+WG/fz/ZG
lE//42rp1a23fvj1wLCgoSD179gx83qZNGzkcjnLnJDVs2DBQ3EnS9u3blSdNmhQoXiTp3LlzZmrWPH/v3786zKmTjZ31bzYlAhfvsbvO98l4YeYkyc8HCfpRoUfwuc
lE//42rp1a23fvj1wLCgoSD179gx83qZNGzkcjnLnJjmjZtGij3JGn79u1KTk4ut9FZu3btyj1u
ooWb//PjxOnDggFasW
CHpxOy9bt26qU2bNpKkjRs36rvvvlNERERgpL4ZkjRs36rvvvlNERETg4+dze/fu1d69e+VyuQLF4+LLF4//mBwAAqK8qdRddAAAAGOvNN9+Ux+NRUlKS
goNOW6SQM+kFjEHsQMmnBvQeWWdkcjlHAA74dbHB7W2l
3NtYkvx+v2w2m1566aX0i
J3NtYkvx+v2w2m1566aVKfa7w8PCzfkxCQoIGDRqkOXPm6KKzz9JJhYmMX4A0IKxzjbtz
Ho1mzZunZZ5/VsGHDyp275ptsmkXTg4B7ZJHm8PexgmTt3aS5w6d1Td8pdUNGC
J49e35XZgAAgLqIgq+K/O+NqgHgXo0MHoCABqEY/Ho1mzZunZZ5/VsGHDyp275JJL90677wZmw61YsUJNmjS
RJOXm5mrXrl1q27ZtZuWutWbNhWDbMmsBx3586dysvLKzfm19q2bavPP/+8/vPPq4vXVjWzQDbmWjHVJvVyDRW0z98xjUTR9ap46HXTMjQ7ZvBzbjaPD0eBv3aHxMS/7bzMpfF+96ckvALPK7jZeUdPx9KEbY75ZC7Bt7e2c6dRbPcHAm2Oa
bm6sbb7xRHTpOKPcxbty4cstOH3nkESUtSlevFhbtmzRxIkT1aBBA11yySWB+yEySWB8larVbffffrtWrlypWvXauLEierdu3egDJWoYMPXFMvaHHAOnbHN/PGDbcKDpx3ITQgYjUAQIIAomQFAGuLW47Rydcl7W2f1wCl0Lbgzk
3eg8KvIkCFD1LFjRO2YMEHr1q3TqlWrdN1116l///I2
dpdWrV2vv3r366quvdMMNN8jr9SoiIkI33nij7r77bn333XfauXOnTx8MjxFIKUYoWgpAnSs2hqFJYImh9RSwjrjtCeUsmnmPFeIeVjwFvV8CYNaM0Ci1PjmG1vDmY7A4cqRCvgEQD/dAJ4DPOf5bbvz5XKmRDYqfkpY/MyAzWpMX9pbuOXnbbnN21nX+uV6P/sMn1Sy6nNn12AN+oYcy7PhVmN44Fsk99wOpT9QudZXHtMHjYHaWCeAe3V+9RvK+7/I0Wll9uJ3MwK1ukw7VyQ9sjH9xNZ5FgRBVEDDYk8nRIGCmm0ud7u1pyZl4WdRY3ryqNbjf5yu7q4wn56+rdMyswbnoPjKFSCKU3P62fCljsBnpi0bj5MjpUUsJ7A2nymZyhHI8VpCWE8EWLpVHJXTT4C7bPq1d9zDdEBdt5c3Xn+MdKYdymzGqfFrtRXaNwthTaHsC5bUr8CPstv0OtQvTzgqD4UsPpewRAmnn5nb8k4XPiSjcBPoqtpqfSxqXznWqN5AvQp6YrWnZWy9jnMh3pnsxE9u0a8DXr1XTjnhnHRgOkq5KjWyvqhWF2k9+0qmHoyMZ+5wFmWOM8dT3WD4J7k2jk5sK9xDKj9cGhaNx9nlAzqDPlR5tgYt9Oi6tL7bhcCQfQ
kSW1FZKM4dVmbL9edbBBK4YCMIbsMNUh5Zq+QPsjcABvScaoa6jl9NlMaP3lrnOA9A7lRHbSRz9Pz+UBu3t2wA7Y5C7GxfCaUbmOHi58i3GBwWLm5Myu4QfQ1fRQbybEEQAsOpjWKqoaVK1Zq4yUGm8f2bwAHh5cKvIU1jfaBBNCrTNiAhFd6JrDuvWSv8GRUaYbBl9nWOCdNAz5MMn7rLIkJ2nCaoqpQ+XF6mSW0RrRUxG3svJgWS6zU8IU6Jea1FjqHVLAIUOyGZ6AYLAbFW0zXJBW6jNvTzVRrKCv95UTwN6rhvXNDxO8TeMxdO3yejEPjRoDqA8AKqd6b+Z05jdMP9RLrO9KGb8xG96ke5Pq1e5XUp5Ag8XrMB7Z55zuBOiZFvQUbPUPtcqnSSzhnXTvNhnrwRwdRqSeNX9S8O42HoKOaxtMRHB7dMmkfhMnKDzBuaXPTMuKWG8KAvGmz9dHpZ6Yx4N0KZs2Yp3Y6ccTYvUqoyPeUJyOhPmOXk1DxArXW6YWaJLVZb9lHKLOqWGd3IPiwcXFhmrqM3SRl2PbG1zP7NQjxyHUZR7HwzD6zLbD7JbVaTfzUJQxpaAgr3IpJx1iEr4iWvCKXKbGk4Xe3rP9uUHbzGMNzJ0dsU3fEeS0AW0MaF1QLnh5FL3pjNJqdbWfIS6GzSJkuEd4AQKqKHyUL7yNfC9LS20mmCkMWXGKQgVhZDhOnkl38KWXnGrWV5vOZ9Q6QQBzxSyOVbTp7Z3AGb9AN99SJpq8HQNHcq6qxzpBI2iBbJGf5oB9ASrAnbsOS5Yz2I1gOnHfqoH4EWm8hnI7BvKU9PFkLXQUG75vLzsqxYzlqNfbYWCHqLeC99ttcmzxvP1Fec0nNP3dLv9s9vyIMBI1HApVLaWNsdgMntPzEdPRYmB3ay3AzHHyqNn1GmSzN2g9yCUHUwkV7AiOVtjtWKzFsoJDjW1YVMpfd6MEuCFbsFTAR/6Uh8iNxs3v3CQX+VXMkUbD6tF5yoomCC+uN0Y9ZfUdFeHfHLwgsZzbtcEnHhCr0rz0JAVIH+QlvXe08NH03VoFRUmz/8oH1m0SkiNhXfqQPN1sUUfDSs/YpbOuy9W9hwB

XOmtm/fftK57du3q0+fPuWO9enTR7t375bX6z2r5/nmm280ePBgNWrUSJGRkbr22mt1/PhxlZSUBMYEBQWpZ8+
egc/btGkjh8NRLluTJk3UqFGjwOepqany+XzauXNn4FjHjh3L3Xdv+/bt6ty5s8LDw8u9jv993IQJE/T999/ry
JEjkqTZs2dr1KhRcjgckqSNGzdqxowZioiICHwMHz5cPp9P+/fv1/bt2xUUFKTu3buflB9AzTRixIjA7OOfP1J
SUoyOBQBVhoIPAACgjurXr5+GDx+u+++/v8qe48CBA7r44ovVqVMnffjhh1q7dqlefvllSSc2wahs/1vknameP
XuqRYsWmjt3rpxOpz7++OPA8lxJKioq0pKq/9Cdt2LAh8LFx40bt3r1bLVqOqMz4AKqJzWYLzD7++cNisRgdCwC
qDLvoAgAA1GFPPvmkunTpEtiQvxHkPCFNPPvmkunTpEtiQ4mdt27bVsmXLyh1btmyZzjvvvMAvwcHBwb85m2/t2rXy
y/PlSSPx6M1a9bo/PPPlyTt3LlTeX15atu2bWBMWlqajhw5oqSkJEnSihUrZDabT8r
WnfS4CRMmaPbs2WrcuLHMZrNGjRoV0Netzdt27ZNLVu2rPA52rRpI4/Ho7Vr1wZmIf6cHwAAoCZgBh8AAEAd1
rFjRO2YMEHTp08vd/yvf/2rFi9erEcffVS7du2d0lSHT58WMeOHavw+i1btpT
b7daLL76offfv26e2339Zrr712Ojir1arbb79dK1eu1Nq1azVx4kT17t07UPhJUkhIiK6//
w5dccUVp73/34QJEwKP27Jli7777jvdfvvvtuvbaaxUfH19u3Lp16/T444/rsssuk81mC5y799579dNPP+m2227
Thg0btHv3bn366aeBTTZat26tESNG6E9/+1Mg/0033aTQ0NDfePcBGGXBggXllt1ffvnlRkcCgCpFwQcAAFDHP
fLII/L5f0WOdevWTe+9957mzp2rDh06aOrUqXrkkUc0ceeLEco87cOCAWrRooYYNG1Z47c6d0+u5557TU089pQ4
dOmj27NmaNm3aSePCwsJ077336uqrr1afPn0OUERGhefPmlRvTsmVLjRO7ViNHjtSwYcPUqVMnvfLKK6d9bWFhY
frqq6+Uk50jnj176rLLLtPgwYP10ksvnXTTt888/X5s2bSq3PFeSOnXqpCVLlmjXrl268MIL1bVrV02d0jUwk1C
S3nrrLSUlJal///4aO3asJk2apLi4uNNmA2CcgQMHll1t2/+t/5ACAusbk9/v9RoeoCgUFBbLb7crPz1dUFS1P
/+WLVuq/TkB1D0dOnQw0gJQL5WWlmr//v1KSUlRSEiIOXFqvRkzZmjK1CmnXdL68MMP65NPPPtGGDRuqLVdVO0t3
XkNE/pwJ13cSJE5WXl6dPPvnE6CglXpnHK6fLq2KXV06XR8W7PIFjZW6zvjmx9uwTymxDqIQlLsWO5BJgeDyOMdb3
H7i+InPg4PMigqxyh5mVVVSI1leiXBtQ73IMPAAAAAIA6wunyKr0OgVBn5Tmpx9yUhFjORqQ37apaiITpwlYNdGG
rBqgZQNFhfC9GXUPBR8AAMCvWCwjWO0nd79dayA7r/2/+t/Z+1oyS7spyRkcAUAd0MDoAAAABALXQ4z6kXvtmlD
9cdlpd77KEG8Pr8+nj9YX64OsXo2Bth3kZQvMJqIZ9KGIM9VRDmmHFCTK+I1WYFwCM43c+mjr
bN/bl/C93cr7lCrBajY6EOoOADAAAAANRqTpdXL3L3+Ut9IC6odRtpdwMAQCc+bGQKpgdsBi+nqaUaxT/UrB+bUvM
qLAo+AAAAAB+ZVdWod5fk65PPNhzR0cIyGQqTGhl6YYANrDjYrRzHFmEcnp/H/Flb/W9dWvP2ZFsL4ep3
BXK3Yd1xrDuRqfVquil1eo2PVTjkyvTtjmx7apaiQIKW2iNWFrFrRqpX6uGahIbNdLr2PVCeCeAWZCALBGIbnqtGhhsdD6gUFHwAAAAA+4ULHdygzz91QHzg
UnuMOOg5quFK3Tx+sPaiQ6ZRuqZXU43Emg4NEbp38P4OexQwpRlIYBdGNVMOYjM8xmK5MzAXLz6W9pMWqmdpUqh4o
hHVrQJpMfJJeswSG9zgB1zQ6zWq/pJGaQIZB1VHmHzf0IxGTf/MDIgxgoZJZbd6ZlS4RtQ/s2fHBTmV5D1S8gHmVdm7ttqYTOMACk8MN/hogQa
LgR57EkUsDzZxCC+74KcVaLvt1gcpIVTFMgbJTgS5BNkKtebXyfSSXX9cyGfXuH+2nP4rNfwtKPzCWpW92rmLH2Ka
1O6/AaLKwEeaAsnWkDxjP4/YICBfVH54vvSrZaeMD2xiJjhRXBJHKBmodjMHmNVoiEfwUmPSyLaW+K77KomhehR
bfG0fTevbmGLSxVOmbLn1/e1m3DGEaBOiz/z/xZk/Rfzfp/DeLpVWd61MWpgZPoEmPdDk7z5ZHaJxd54jE7gvoo
jRhSl5MKjbOMN3WmPyjEGuwKX70Ev/WoaKaU8cW6f/jsbtk6sA5j1eQV0cW5eFK/ue1dBfKLwJtwXl4ix5p3m
eYfwVu3V5O4IBiPEc2J/Nx30rJh6zZFvfGUncK5tX4b0Q5/uQWWqZTfQ3gDLRRVp3Bgep9RaAM/3Tzo1F2jYJHwuA
BqBUpzg3MJ9t3DbDhn4aYXa7wyBFW9EhvRTIzZE7nIeVEVsI1XC4Aw6M/bupeTTuaYOpSd6w7qO8XmNNk68

AAAAAYJjiMo/+9t563TRrjXKdHqPjALVKZkGprv73Sr3wzW75fH6j48BA7KILAAAAADDEhvQ83fL2amUUsMwQO
Fden1/Pf7NLK/cf1z/Hd1FcZIjRkWAAZvABAAAAAKrdO8v367JXl1HuAZXkp73HNfKFH/TD7qNGR4EBKPgAAAA
AANWmzOPV7e+sOgOfbpPHZ3QaoG45VuTSdf9ZpacX7pCXJbv1CgUfAAAAAKBapOcU66JnF2v+FmYYAVXF75de+
X6vxr++XBn5TqPjoJpQ8AEAAAAAqtzCjQc1/NnvtC/XbXQUoF5YfSBXI1/4QSvYZbdeoOADAAAAAFSpRz9eq1v
e3awSr8noKEC9klvi1nVvrtJ7a9KNjoIqxi66AAAAAIAqUVjq1g3/WqI1GWWSKPcAI7i8Pt3zwSbtPVqk+OaOk
cnEn8W6iBl8AAAAAIBKtycjV4Oe/Pq/5R4Ao/1ryT79+Z21crq8RkdBFaDgAwAAAABUqqVbO/SHl37QOVKjkwD
4X19tzdLl//pJmfn84axrKPgAAAAAAJXmnW836sa3N6jEazE6CoAKbDlcoDEv/6jNh/KNjoJKRMEHAAAAAPjdv
F6vHpnznR78Kk1uUe4BNVlWQZmu+NdyLdySYXQUVBIKPgAAAADA71JUXKJJLy3QfzYWy2/i1OygNnC6vbpl9jq
9+eN+o6OgErCLLgAAAADgnBO9nqsbX/tGmwrD2CgXqGX8funRBdtUUubR7YNbGROHvwMFHwAAAADgnOxLz9Af/
/2jDpSFGROFwO/w7KJdKnF7de+INkZHwTmi4AMAAAAAnLX1O/bpT2+vVbaXcg+oC179fq+cLq8eGt1OJhPTcWs
bbo4AAAAAADgry9Zv1R9nUu4Bdc2Mnw7ovg83y+fzGxOFZ6nSCz6v16sHH3xQKSkpCgONVYsWLfToo4/K7//li
8Pv92vq1KlKTExUaGiohgwZot27d5e7Tk5OjiZMmKCoqCg5HA7deOONKioqquy4AAAAAIAz5Pf79fWy1Zo8b4t
y/ZR7QFOOb026pszbII/XZ3QUnIVKL/ieeuopvfrqq3rppZeOfft2PfXUU3r66af14osvBsY8/fTTmj59ul577
TWtXLlS4eHhGj58uEpLSwNjJkyYoK1bt2rRokVasGCBli5i5dqkmTJlV2XAAAAADAGfD5fPro6yW6+9M9ylOEOXE
AVKHPNh7RrbPXyeWh5KstTP7/nVpXCS6++GLFx8frzTffDBwbN26c4778jv9yspKUl//etf9be//U2Sl
J+fr/j4eM2YMUPjx4/X9u3b1a5d061evVo9evSQJC1cuFAjR47xUoUOHlJSU9Js5CgoKZLfblZ+fr6ioqMp8iWf
ko5OZ1f6cAOqesaOTjY4AAKhkRv+cCpwLl8utOfO/1vMr8pVvsRsdBOA16XdeQ71+bXeFWC1GR8FvqPQZfBdcc
IEWL16sXbt2SZI2btyoH3/8URdddJEkaf/+crMzNSQIUMCj7Hb7erVq5eWL18uSVq+fLkcDkeg3JOkIUOGyGw
2a+XKlRU+b1llZmQoKCsp9AAAAAAB+n7Iyl/7z/gI9tzyPcg+oZ5buOqpJb6+Vm+W6NV6lF3z33Xefxo8frzZZt2
shqtapr166aMmWKJkyYIEnKzMyUJMXHx5d7XHx5d7XHx8fOBcZmam4uLiypOPCgpSTExMYMyvTZs2TXa7PfCRnJxc2S8
NAAAAAOoVl8utGR9+rlfWF6sgyGFOHAAGWLrrqP7y3kY23qjhKr3g+e++99zR79mzmzNmjmTdP111m3YFZfJL
UsWNHHTx4UNOmTdP/2s2m62yX
w4AAAAA1DtutOdvf/ylXl5bqPygKPjAKgB3lmRpuiwYP11WGujo6AClT6Dr6SkRGZz+ctaLBb5fCfWa6ekpCg
hIUGLFy8Oncx8oKNDKlSuVmpoqSUpNTVVeXp7WrlWa7Wrl0bGPPt9/K5/OpV69elROZAAAAAPBfbrdHcz75Qq+tOka5B
6CcF7/do//8uN/oGKhApc/gGz16/Xcc8/j3Dc2Mnw7ovg83y+fzGxOFZ6nSCz6v16sHH3xQKSkpCgONVYsWLfToo4/K7//li
8Pv92vq1KlKTExUaGiohgwZot27d5e7Tk5OjiZMmKCoqCg5HA7deOONKioqquy4AAAAAIAz5Pf79fWy1Zo8b4t
y/ZR7QFOOb026pszbII/XZ3QUnIVKL/ieeuopvfrqq3rppZeOfft2PfXUU3r66af14osvBsY8/fTTmj59ul577

```
Gw3Ok6t5c7LlHv9F4rqeYniU69QWcZu5S5+XSaLVREdBwfGlexZpWOfPS2/u0yWiGjFX/moLGGnf99dRw8o8+2
/ye9xyRQcqrhL/0/BDZqccbaIjkPlyj6gI2/eKktolBqMuVe+0iLl/zhb8VdNU+7St1WyfamCHAmKHXmngiJZo
o2aZeqnW9WzWYySY5hpWpmYqw0AAAAAdcDBQ0f06dffKSI8XKvLEuX18+veOfP7ZYtvoej+1ys4voUiu4xQROf
hKtzwRblhIU06KfGG6Uq45hmFpHTXOU+fkrc477SXtsYOOvGY655TZNeLdOzz5+U6lnbGOUyWIMUOuOWN//ymE
q9/XiGN2yv32zcV2X2OXFn75Ny9XIk3vChbUhvlfvP6ubx6oEoVlXn0f59sMTpGncN3fAAAAACo5Vwut95fsEg
5uQXKiWyubHew0ZFqNUtEtKy/mlVnjU2Wt+BouWPm4BBZo5Nka9RGDUbeKZPZrKJNX5/22iaL9cRjEloquv9EB
celqHDNZ+ectfTgJrmPH1Rkt4tVmrZJoc17yBwcorA2fVWatvmcrwtUpaW7jurTDYeNjlGnUPABAAAAQC23+Me
VWrtlu2IbN9O6wiij49R6tkbt5M45VO6YO+ewgqLiTv9Av19+79ntEuo/h8cEHutxKWfRq4odfptMZovk98nv8
5446fPK7/edO3WB6vDogu3KL2FX3cpCwQcAAAAAtdj+tMOa/8OSOSIjtbIkXl6ZjI5U6OX1HKOyIzuVv/w9uXO
PqHjb9yrauFAR3UZJknyuUuUumamywzvkyc9WWeYeHfvin/IUHldY676B62TN/X8qWDs/8HnukhkqTd8iT36WX
EcPKHfJDJWlbS63I6+3KFeurH1y52ZIOnHPPlfWPnmdhSflzPtprkKb91BwfAtJJ4rJkl0/yZW9X4XrFiikUdu
qeHuASnGsqExPLtxudIw6gO02AAAAAKCWKitz6bOFXymvoFDepI46WsDS3MpgSzxPDS/9P+Utmam8Ze8qyB6v6
EE3K6L9QEmSyWyWO+eQjn6yWF5ngSyhUQpOaKWECU8puGHTwHXcuZmyOQsCn3uL83VswXPyFufIbAtXcMNmirv
iEYWmdA2MKdzwhfKXvRv4PGvOfZKk2JFTFNFxSOC46+gBlez4QYkTXwwcC2vTR6Xpm5U5+15ZYxupwei7K//NA
SrR3NXpGtutsXo2izE6Sq1n8vv9fqNDVIWCggLZ7Xbl5+crKqr6p6h/tDOj2p8TQNOztnWiOREAAJXM6J9TUbc
s+GaJ3vn4CzVuOlSf5SXLxcYaAGqZVnER+uLOC2W18P3r9+DdAwAAAIBaaM+BNH2++AfFRju0qawB5R6AWml3d
pH+tWSvOTFqPf4GAAAAAIBaxllaqvfmf62ComJZ7AnaXRJmdCQAOGcvfrtHB44VGx2jVqPgAwAAAIBa5uuly7V
5x26lNGmsFQUO+dlYA0AtVubx6YFPthgdo1aj4AMAAACAWiT9SKa++v4xnxhoAar8f9xzT55vYz
+BcUfABAAAAQC3h8/k0/5ulyskrUHRsA6OuYKMWAHXHUwt3yOXxGR2jVqLgAwAAAIBaYsO2nVq5frMaNE7UxqI
oOXOWoyMBQKVJyynRrOUHjI5RK1HwAQAAAEAt4Cwt1by27yhvIwFApXvx2z3KK3YbHaPWo+ADAAAAgFpgyYq
12rxnr5FnvV5oITtLKfLt8bKwBoA7Kd7o1ffEeo2PUOhR8AAAAAFDDZRO7ri+/+1H2qEj1+MOVXhZidCQ
AqDLvrDio9JwSo2PUKhR8AAAAAFCD+f1+fbH4R2Uea6k+IZaVxhpdCQAqFIur0/Pf7PL6Bi1CgUfAAAAAANRg2
/fs14+r16tRQpyy3CE64rIZHQkAqtwn6w9rV1ahOTFqDQo+AAAAAKihXC635i2b4XqVlZYpx2LWW2XsA6gmfX3r
mq51Gx6g1KPgAAAAAoIZatWGLNm3frWbWbJSTpSFqxMZu8BqEcWbcvSurRco2PUChR8AAAAAFADlZaV6eulynYa
lVoSAiz9wDUSy8u3m10hFqBgg8AAAAAaqBVG7Zoz8F0NU6K8EHAAAAAADW
Ms7RUXy9drpBgm2zBweycC6Be+9eSfUZHqPEo+AAAAACghlm5fov2pR1S48R4pZfal0O0NjoSABhm/sYj0pRbY
nSMGo2CDwAAAABqkBJnqb75YYVCbDYFB1u1ntl7AOo5j8+vf/+w3+gYNRoFHwAAAADUICvXbw7M3st2WZm9BwC
S5q10V26xy+gYNRYFHwAAAADUEMUlTii1aulyhISEKtlq1tTjc6EgAUCM43V7NXH7A6Bg1FgUfAAAAANQQK9Zt0
v70w2qcGK8w2qcGK8Sr1n7naFGRwKAGmPmTwfkdHmNjlEjUfABAAAAQA1QQXOLUoh9WKDwsTFZrkLYXh8snk9GxAKDGyC1
xa97qNKNj1EgUfAAAAABQA6zasEUHD2eoUWKcvH5pe0mY0ZEAoMZ544f9f98nh9RseocSj4AAAAAMBgbrdHS1
UhwsKxBQdrrDFWpz2J0LACocQ7n0fXN9iyjY9Q4FHwAAAAAYLCtu/Zq267MRLgFOauzrd6Ag1DgU
fAAAAABjI7/dr2er18nq9Cg0JUWZZsI67g420BQA11tJdR5WR7zQ6Ro1CwQcAAAAABko/kqmqA6BJGbvA
cBv8fml91YfMjpGjULBBwAAAAAGWrl+swKi+swoKi+SIilSx16wDpSFGRwKAGu+9Neny+fxGx6gxKPgAAAAAwCB5
6cfV6xUQ7ZDKZtKckTH6ZjI4FADXe4Tynftxzz0gYNQYFHwAAAAAYZO2mbco+lqP2MTAUDtMY/NN
gIo+AAAAADAAG63R0tWrlVoaIgsFou0u4OU67EaHQsAao1F27KUU+wyNQOkaNQMEHAAAAAAbYvHO039qcdVmJcQ0n
SnpIwgxMBQO3i8vr00ToEYTP51NoiE1+v7SP5bkAcNZcHp8+WsNxRYECj4AAAAAqGZ+v1/LV6/XsbxcJcTHKscdr
f333nsZrmAV+ywGpwKA2mdwdpE2pOcNhZHcNwFHwAAAAAUM227tyrB3+XAjUFeUMAjQX4t0d0
Qjfx+v1au36wQW7DMZrM8fml/aYjRsQCglvp6a5bREQxHwQcAAAAA1SjfAAAA1SjtcIb2HjwUJ6bVhoit59fzQDgXOO/V
qwdmQVGxzAUf4sAAAAAQDXavHOPioqLFRkRLonluQBQGRZuyZLf7zc6hqEo+AAAAACgmng8Hq1ct1n2iMhAscpB&xHwAAAAABUIb/fr43bdinEZ
pPJdKLUY3kuAFSd5fuOGx2h2lHwAQAAAEAVOpqTq4OHjijaESXxGD1HwAUCV+WkvBR8AAAAAoBLtPZCu/MI
iOaIiJUl5niCV+iwGpwKAumvNgRy5Y1QrCj4AAAAAqEI79hyQ3++XpdRuyoNWvWBM77/X5NnTp
ViYmJCg0N1ZAhQ7R79+5y18jJydGt13r+UUVFRVURFwAAAACqxP60wzqhG2+8UVFRVURFwAAAACqxP60wzqWm6dohz1wjBl8AFD16
ttGG5Ve8OXm5qpPnz6yWq368ssvuvW3bNj377LoKjo4OjHn66ac1adIkzZw5U6tWrZC9fuffPq3i35tbfJ/mg2ZoRMZM2fLaJI4Xi5aFqPfSbbIM7B
```

BW7du1aJFi7RgwQItXbpUkyZNquy4AAAAAFBl9qYdktvtVojtRKlX5jMpzxNkcCoAqPvWHsyVy1N/7sNX6X+zP
PXUU0p0TtZbb70V0JaSkhL4/36/X//85z/1wAMPaMyYMZKkWbNmKT4+Xp988onGjx+v7du3a+HChVq9erV690g
hSXrxxRc1cuRI/eMf/1BSUlJlxwYAAACASrdr7wHZgn+ZsXdi9p7JuEAAUE+Uun3akJ6n81NijI5SLSp9Bt9nn
32mHj166PLLL1dcXJy6du2qN954I3B+//79yszM1JAhQwLH7Ha7evXqpeXLl0uSli9fLofDESj3JGnIkCEym81
auXJlhc9bVlamgoKCch8AAAAAYBRnaak0HMpQVGR44BjLcwGg+mw+nG90hGpT6QXfvn3790qrr6pVq1b66quvd
Mstt+iO0+7QzJkzJUmZmZmSpPj4+HKPi4+PD5zLzMxUXFxcufNBQUGKiYkJjPm1ad0myW63Bz6Sk5Mr+6UBAAA
AwBk7nJmtgqIiRUZEBI5lUfABQLXZdqT+TP6q9ILP5/0pW7dueuKJJ9S1a1dNmjRJN998s1577bXKfqpy7r//f
uXn5wc+0tPTq/T5AAAAA0B0DmVkq6zMFbj/nt8vHWUHXQCoNluPMIPvnCUmJqpdu3bljrVt21ZpaWmSpISEBEl
SVlZWuTFZWVmBcwkJCcr0zi533uPxKCcnJzDm12w2m6Kiosp9AAAAAIBRDqQfkdlsls l04p570Z4guf2V/isYA
0AU9h4tqjcbbVT63y59+vTRzp07yx3btWuXmjZtKunEhhsJCQlavHhx4HxBQYFWrlyp1NRUSVJqaqry8vK0du3
awJhvv/1WPp9PvXr1quzIAAAAAFCpfD6fdu07oIjwsMCxoyzPBCxPBPYBq5fb6tSur0gY1aLSC7677rpLK1as0BNPP
KE9e/Zozpw5ev311zVlyhRJksmcWv+9t30bOS7FRMTjlv7MqmpW4/5OqQoJvzJwo5H3s+zg7XqG1OmMaTrB5jmAgYjfkBrcajwd8DUJBqTmT0JS2F8bJ3+zgtY5NXbAhmA5QgCoXajXoz8SFVl2tbnIZYfOiw5vffbubjTn08LJQzdcH0Vs+J5XBVvtqPO1ShdUWTjFbXFdLfK5NIZdQZ6HDtZTRpDAu3QrKYOR7BiGlSkzPQXsR3ynRmR+MRfuJcVOGwRVxHBkdNqxQ/+C1RxUDGQNJvVg+R59zZ09D+eUhlzNqpt+nXnjqVR79B+WMMUUeImg+Oz9O3VDP/eNvcIMtfs9N9PVtRaCbhYt5fTiFqzzO8m/ESlC/2Ovj2WSFUyz6yYcprQytvJ9J7x4ymCWv7Bnwp2unNY5zmEOLmB6ia5sTwBkRe4ppWWfEa5tnlIcMEfRLDAT9m/nF9hjtlbmctEWN8NGnwRmpw8qmTHiScLtXQE8xw7XUCbmO5fePLedpHjM0mI0vsGVLp9dF++/kgt9pnH3zpjjjpsoc4Ugne7swfyyZGwjl4V3Nn8MuXWRtiw91fZdvgAw+J4ok20nlBZFY+o5Gqne0x7brHaBDsfdHzPswmfAqMqz9FxoR/uOCOhrNIKV4qYs/1ysSS3m4QqdHwK7JWq3GNSBd+VR+Vh/oV2U5d0Rz0mu4n15dkW6LYtBkyIlTd76D12BjvPe6YSuhK9BT+lXQNrPCjCrh7+hbtFRIiIFGqNUd8y2T/ELwQ6g6WxW7e5S0S0IOc4tOwA7PtJ7l3n2ghbC3oeuxs06bPlEK5YzIfw6DfN3xoDRu51PxfQKnYdz1pj5XvxPSDNwDPhNDg5r+GxbCH6zjYPgpHOa59hSRlTwyxDbpRlO7V6F3PX08w5hdGBXkR5X95UXbh6ZvXR6GjfT/yi0/7cyIGvs3F+kzmVLx2P5ksRwpHGFpmHrNHpzHv/dZeX0Nh/ZiYZYxdhqFIrqLK+fhPDkqd5jh5HWuZ+OPJR5dRd9LPkuOgthZpVyhV3m2Mg83CykXmhSjcDY9NOO6Vh1mHeRTQRVkrbLsNV6NpR5I8Hs4Yh3qCeVuWj+KiQKhOKlXXHDaCrgpxHAw7pXQbt2Mn8PVlbVm0hqZ8KJwOC8fNFjn6MWU6Mr0yqjBwKNNkJcLsIsfUW1TqfDu1FjC4hLx+6hJrn6LkI6gGI+3j4bRPuq3WJ/W5gENjtBZSZvHZMihLkPfRRYRVyAfGw8HzQoD+RdsNJD9aoN99lkH3NRHpbUKD8sGxGcEbNzMHqKd/bQTOJB8qsnXnb0NxeNGpJVlWnFqB3ZJ2I3Q1OwHd8LdV4KXBQGWKNXE+jyi/tuUTz1SJqbJ/2P+yp/BLaNvRR7Y9CZgS4wX8nI0LGF5dfLeY0fydkxGZ5Ab+BjPHKHdvSoF4EKuYGnIC6NA6/bVd/gKXVgghqfAEqJD8Gg0ZtgmqdnVzt1Kk/dMhAKRf/rIYjMS3bfiiC5XQ1uKNwNPMGbKwNWwDG5HS6PE1aM3X2EFwt+XU2jRz5SR/xcXYEfBiLG2tvMYCY0pEqTBKt8lfSTsIXahNdE57HcMiCDgCAFEbdHWnRxoXAWfAuOTShWZeK0okFrTDJr1fXwhh8xZRzyNBJxmAHYHL9dCZz+dk5nk9lImOmQkZgoFBXbZRsmGNvfP2lukEgfWSQZEzt0mYkKtJFkIxwnJGQi3HJg8SGrSL6xhPfj9RuaAKQsm3+Gy/Q6r2XdOspRhDn62zTGbP1nf2jOuqAIbv5BZ9Wx0sXJMbjuDcPhzWXHxVNhaE5k3kfGX5ZSMhBqhvtFUEb3Y+yaF1hGVRSMKmjI9b1OO2WIRuq5mNNzJv0D2/6gOs+oFOh2K7BdSFJ+Uk5Rl+uJRYsuF8PSCgPYwNGeXG6j2J2EHUtQpmVsijg5a+G0VqTDvjMJU4tHwz2AuMYYP1U9YZ5KBQJgUc8LiWm+8tk+PP23+X9rV5gc5F7I8dLN2BIHknEMzBLI6grg6Yp5LTHtQ=

z/Rns0bJEkZB/Zp18a1en7+d2rSqrUkadLDT+rGvp314+cfa8jlEyq87o71a9SmW09dOHps4Lp9R12i3ZvWSZK
y0tMUFhmpPiPHSJI69LpAh/btVo+BQ/XDgo8VFBSk3sNGVuVLBwAAwBkIDwtTsNV6Yifd0F+OB5mkCItXhV5+/
ULVauPapn9Hva5gExu7AOXEtjQ6QZVjiS5wh1p37aHNy3/Ukf17JUkHdmzVjnWr1LXfIEmS2+WSJAXbbIHHmM1
mWYODtX3t6lNet03XHtq7dZN2b1ovScpMP6h1SxerW7/BkqTEpikqczq1b9tmFeblas+WTebl
NODj1fJ6wUAAMDZCQ8LkfW/M/h+zc4yXVSxJHe6Zoc9qwizy+goQM1Tx++/JzGDDzhjl066TSXFhbpjZD+ZLRb
5vF5dPeU+9fvvzLtGzVuqQVIjvfPcNP3570/JFhqmBTNf1/HMdOUezTrldS8S8cPVYFuTl6YMIl8qrCpIl8vv98no8Gjb+O
o378x2SpAi7Q7c/+YJevPd0ucpKNWDMZep64QC9/H9/0UUTblDWoXQ9eetEeTweXTn5r0odcXG1vB8AAAAozxY
crNCQEBUUFZ90zhHk0aEyA0KhXrB7cjTP9rhig5xGRwFqppjmRieochR8wBn66cvP9MP8jzTlHy8ruWVr7d+xV
W898ZCi4+I18NIrFGS16p7pb+qVB/6i63u1k9liUafUC0/M8P0f+v4XW1b+pI9ef1E3T31CrTp1U2baAf3niQf
1/ivP6/Jb75Ik9Rp6kXoNvSjwmK2rluvgzu266YHHNHlYH9317CtyNGio+64YpXY9e8se26DK3w8AAACUZzKZl
BDXQFnHck46xww+VBWbz6l3LQ8p0bjA6ChAzcUMPgA/m/XMo7r05tvUd9QlkqSmrdvq2JFD+uj1FzxXw0iskSS0
6dNKzn3yj4sICedxu2WNidd8Vo9SiQ6dTXnfu9KfV7w/jAvfoa9q6rUqdJXpt6t0a9+c7ZTaXX0nvdpXp9Ufu1
51PTVdG2gF5vR61Pz9VkpTYrLl2bVynno0GVcE7AAAAgN/SOCF0qzds0ek4BR+qgsnv0Vu+qWoXcdzoKEDNxj3
4APyszFkq06/KNrPZIr/v5Nl54ZFRssfE6siBfdq7ZaN6Dhp+mus6Tyrxfv7cX8HMvw9efUFd+w5U8/ad5PN65
fP+cgNdr8ctn48b6gIAABilQUx0hccdFHyoAtNdj+mCiMNGxwBqNotNimpsdIoqxww+4Az1GDhUH742XQ0TG51
Yort9i+bP+JcGjRsfGPPTwvmKio5Vg6RGStu1Xf95fKp6Dh6hLn0HBMZMv/c0xcQl6Jq//r/AdefPeF0pbTuoV
eduyjy4X3OnP6MeA4fKYrGUy5C+Z5eWffGZ/vHx15JO3PfPZDLpmw/mKLpBnA7v26uWHbtU+XsBAACAisU47DK
ZTPJ6veV+lguz+GQ1+eT2M8cClePBOuc02rHL6BhAzReTIpnr/vdeCj7gDN3OwGN6d/rTev2R+1Vw/Lii4+I19
MprA/fJk6Tc7CzeNePJh5R8/JkfDOAOYc7kuu2VKuesc03JYJtMv31wuu2WKTCaT3n3haeVkZSoqJkY9Bg7V1VP
uK/c4v9+v16berYn3Pqzg0kpVLdN+6feePT/yeNy6BJki0kVLdN+6feePT/yeNy6aYHH1NsfGLVvQkAAAA4rdhou0JDQuQsLVNEeFi5c
/Ygj465gw1KhrrkT87/6MboNUbHAGqHerA8V5JM/orWANYBBQUFstvtys/PV1RUVLU//Oc7M6r90QHUPWNbU9j
+Gt9fAVQWo77HGv1zKqpWaVmZ7nn8eclkUlxsTLlz3+c6tMcZdopHAmfmUucnetbxvsym0vmrPFD5LvybNPhBo
1NUubo/RxEAAAAAqkmIzabYaIecztKTzjWwug1IhLqkn/N7Pe34kHIPOBuNexqdoFpQ8AEAAABAJUp0jFdJadl
Jx+OCXQakQV3RsWyDXr0/JauJTfWAs0LBBwAAAAA4W/ENG8jv851OPNbqllnMvMLZa+Lap1kR0xVmZhYocFaiU
6TwWKNTVAsKPgAAAACoRLHRdvl1Yp00/2UxnSj5gLMR48nW3NAnFW05edk3gN+QfL7RCaoNBR8AAAAAVKLYaId
sVqvKXCcvyWWZLs5GmLdQ84L+riRrkdFRgNqpnizPlSj4AAAAAKBSJTSMVUR4mAqLS046R8GHM2XxuzRLD6lVS
K7RUYDaq3EPoxNUGwo+AAAAAKhEkRHhapwYr8Ki4pPOxbFEF2foFfcj6hGeaXQMoPayhknxHY1OUWOo+AAAAAC
gkrVKaaLSCnbSjQzyKtTMLqg4vcdKn9LwqH1GxwBqt8QukiXI6BTVhoIPAAAAACpZ48R4+SX5KthNtyHLdHEad
zj/pWscG42OAdR+9Wh5rkTBBwAAAACVrlFCnMJDQ1Vc4jzpHMt0cSrjne9pimOp0TGAuqEebbAhUfABAAAAQKV
LaBgrR1Rkhffhi2cGHyow1LlIjzk+k9nkNzoKUDckn29OgmpFwQcAAAAAlSwoKEgtmyWfciddq+nkpbuov7qVr
dGLjrcVxNcFUDnsTaTIBKNTVKsqL/iefPJJmUwmTZkyJXCstdLRUkydPVmxsrCIiIjRu3DhlZEL4ZWWVe1xaWppGjRq
lsLAwxcXF6e6775bH46nY5H5m/Gx2lC0FvELtkKsYQxPpZ3qxjhFRp0YZtfEKhZk5NJyn2fVC4pAzSOXx/jWmvZVFw6SwKlg58lL1EjSVQs/bIGO7vxTm8VMzxsx0a9M5z8sL
lsLAwxcXF6e6775bH46nquAAAAABQKVKaNJLXe/KGGhaTlGg7eQMO1E8tXLs0I+JlhZj4fReoNC0HGZ2g2lVpw
bd69Wr961//UqdOncodv+OOO7Rg4UJ9/PHHWrFihTIyMnTZZZcFxnu9Xo0aNUoul0urVq3SG2+8oaVLlw4dOxq1ny8aaR2qG/zjs6cXvlSbsnc62u27yxq7n93ZjS8XJMzmR6D/IV4uHXm+S2dONOv+zp/mAvNvjDoavAaZ04pqsjxLxcrkcDM0vD9JfzEHcjrmzM5L+wKPjtD7vMjWq4sDsTUq0TRjm3xTBXKAgw8kuw9jOMXjq+TZQl1YVhzNmOj0zAAAAlNdvrvR/AhMOQDuOyCvOWaqmqM1TpgVoi1kz0rGdhFcnZw1qxN9ZCVzJpFR8Fes+7uqYWZr8JaIEeXP0YdGy5K1lm+ZJCqwbXbRfuUn0vpCzrWpO1yVRI7CfW3lcSKgdGxY3Oei74TykLvMGdMZVLaFCd1D2rfycVLAHkPuqpXa7F0/SC9WN1LcLZ0wHS/fwKmU8f/dWXxG2Sd6u91h5Kk7pZKsDT7H9gUNi8ZeGRdOb0ebN+/LQz
AAAAAqPmS4hsqKiJchcUlJ51rGhPMWwgCWk5xwrOZKGgDJyJZkx2sb8ZYVp9u0tsZAXoNzn8/YRRb4IswdzQR2g6q0vFtmcJh2/Mk4ojuHOlpfF4MGQo4IthYpU/Cx9g8L6X9xVb7ZezJh8SSjUv4YNSaJA7wa/YLtA7zgRXLBm4jPBaVaPmS4hsqKiJchcUn34evMQVfvRfhzdc866OKs5789QHgdwgKlZoPMDpFtauygm/y5MkaNWqUUu742
rVr5Xa7yx1v06aNmjSpol2ucQsWwHjQhirSeMbrkUjOzbKlnk9erAWVAAAAAc2cTyoQqPvnBoJ7Tg9Zj9y8rK0JrvFuevHH0XJ5bQX09ebuJ6LaQm2+yj7eocv2ms/cSgS0GNZjq8NRHR2v8kiaCjlRpSAAAAAP4PYdMO+cTmHd5AAAAAB+V0KxEO5v0wq5cKx3GVkvwkfxzijUyLD4sHoAkpnpqS7v6L
ig7xyBLGbbn1l9ZVptmmqUmz5RkcB6p6UCyVrqNEpql1QVVx07ty5WrdunVavXnX3Suc
zMTAUHB8vhcJQ7Hh8fr
8zMzMCY/y33fj7/87mKlJWVqazsl38FxKygo+D0vAQAAAAB+F5PJpE5tW1R5rHnn0VnmnmeRtWmQMt2sTC+wmhydt5vbFF/QMNy7

wQelQldq4tunfUa8r2OQ1OgpQ97UcKpkr/U5OtUb9feUAAAAAUA3MZrO6tG+t4mJnheebh1Z8HLVbkjtds8OeV
YSZ+ywC1aIe339PouADAAAAgCrXommyLBazXG73SedSQp0yieWbdYndk6N5tscVGOR5C1QLa7jUapjRKQxFwQc
AAAAAVaxFO8aKtkcpr4LddMMtPsWzm26dYfM59a7lISUHn/zfGkAVaTdGskUYncJQFHwAAAAAUMUiwsPU7rzmy
snNr/A8y3TrBpPfo7d8U9Uu9LjRUYD6pesEoxMYjoIPAAAAAKpBl3at5fP75fGevOFC81CnzCzTrfWmux7TBRG
HjY4B1C/RzaSmfYxOYTgKPgAAAACoBm1bNVeMw17hLL4Qs1/JIaUGpEJlmVr6nEbbdxkdA6h/Ol8lmUxGpzAcB
R8AAAAAVANHVKQ6tT1Px3PzKjzfkmW6tdafnG/qj441RscA6iHTiYIPFHwAAAAAUF26tDtPkuTxeE461ySkVDa
Tr7oj4Xe61PmJ7nV8a3QMoH5q1leKbmp0ihqBgg8AAAAAqknbVs3VIMahYxXM4rOYpBRm8dUq/Zzf62nHhzKbu
H8iYIgubK7xMwo+AAAAAKgmkRHh6t6xnY6fYjfdtuHF1ZwI56pT2Qa9Zn9LVtPJm6YAqAbBkVK7MUanqDEo+AA
AAACgGnVu11pBFotKy1wnnYu1epQQXGZAKpyNpq59mhUxXWFmt9FRgPqr/RgpOMzoFDUGBR8AAAAAVKM2LZqpU
UKcso/lVHi+HbP4arQYT7bmhj4ph4VdjwFDsTy3HAo+AAAAAKhGNluwenftqMKiIvn9J9+7rVlIqcLMLPusicK
8hZoX9HclWouMjgLUb9EpUtMLjE5Ro1DwAQAAAEA169i2lcJCQ1VUXHLSObOJe/HVRBa/S7POkFqF5BodBcD5N
xudoMah4AMAAACAapaS3EitWzTTkeyjFZ5vE1Yii9iZtSZ5zf139QjPNDoGgBCH1O16o1PUOBR8AAAAAFDNzGa
z+vXqJp/XJ5f75I0aQiO+pYQ6DUiGijxe+pSGRu03OgYASep5o2SLMDpFjUPBBwAAAAAG6NzuPDVOjFdm9rEKz
7PZRs1wp/M1TXBsNDoGAEkKCpF6/dnoFDUSBR8AAAAAGCAOJEQXnt9VBYXFFW62ERfsVkOry4Bk+NlVznma4lh
qdAwAP+s8XoqIMzpFjUTBBwAAAAAG6dG5g6IdUTqWm1fheWbxGWeo82s96pgvk8noJAAkSSazdMEdRqeosSj4A
AAAAMAgCQ1j1bNTOx09llPh+RahTkVYPNWcCt1LV+slxzsKMvmMjgLgZ21GSbEtjE5RY1HwAQAAAICBenfvrOB
gq4pKSk46ZzZJXSKKDEhVf7Vw7dKMqFdkM1GsAjVKn7uMTlCjUfABAAAAgIFaN2+q1s2bKSOr4s02zgsrYRZfN
YnzHNHcsKcVaS4zOgqA/9WOj9S4u9EpajQKPgAAAAAwkMViOYXnd5Pb7Zbbc3KRxyy+6hHhzdc866NqGHTyTEo
ABmtzp9EJajwKPgAAAAAwWNcObZQU31CZ2cziM4LVV6bZpqlKseUbHQXAr8W1k1oNMzpFjUfBBwAAAAAGCw8LV
f/ePZRfUCiv9+SNHZjFV4X8Pv3b+5A6hx01OgmAilxwh9jO+rdR8AEAAABADdD3/K5KiGuojOyKiyZm8VWNZ12
Pq39kmtExAFQkrr3U6UqjU9QKFHwAAAAAUANE26MOuM/5yssvYBZfNbnX+aLG2bcbHQPAqQx9RDJTXZOJ3iUAA
AAAqCGYxVd9Jpa8rT87lhsdA8CpNB8gtRpidIpag4IPAAAAAGoIZvFVj9HOBXoweiG39QJqKpNZGvqoOSlqFQo
+AAAAAKhBzmQWX3SQu5pT1R2ppcv0rGOeLCa/OVEAnEqnK6XETkanqFUo+AAAAACgBjmTWXy97fkGJKv92pZtO
b+j3lCwyWtOFACnEhQqDXrA6BS1DgUfAAAAANQwvzWLr5HNpaYhzmpOVbs1cqfpnfDnFW52GROFwOnOvkWyNzY
6Ra1DwQcAAAAANcxvzeKTpF5RBbKIZaZnwu7JOVzb44oNohQFarSwBlLfu4xOUStR8AEAAABADdT3/K5KjD/1L
L6oIK86sOHGb7L5ijXX8pCSgwuNjgLgt/S/VwqJMjpFrUTBBwAAAAA1ULQ9SsMuTFV+QaFc7oo31egSUaQwM/e
TOxWT36MZvofVNvS4OVEA/JaYFlKPG4xOUWtR8AEAAABADdU/tYdapTTRwUMZFZ63mv3qGVVQzalqjxfdjyol4
rDRMQCciSEPSxarOSlqLQo+AAAAAKihwkJDdPGQ/vL7/SoqLqlwTMtQpxpa2Tji1x4qfVYXR+O2OgZ+5dXVLnV
6tUhROwoUNa1AqW8W68vd5WeoLk/3aNDMYoU/cWJMv7eK5XSf/n6Thwt8uuYjp2KfLlTo4wXq+GqR1hz5ZXZrV
pFPEz9xKunZQoU9XqAR7xRr9/Hys1//8lWpYp4qUPLzhZq9qXym97e6Nfrdiv8MohKOGCS1+4PRKWq1IKMDAAA
AAABOrXvHturesa1Wrt+stq2ay2QylTtvMkmp9nx9dqyBJFPFF6lnbnG+qRui1xodAxVoHGXSkONsahVjll/Sz
A1ujZnr1Po/mdU+zqLl6R6NmF2i+/va9OJFIQoySxuzfDKf5ks71+lXn/8Ua2BKkL6cEKaGYSbtzvEpOuTEg/x
+vy6Z55TVLHO6PkxRNum55S4NebtE226NUHiwSfN3ujVns1tfXxuu3cd9+uNnTg1vaVGDMLPyS/36v2/L9M11Y
dXzJtUzfmuYTBc/b3SMWo8ZfAAAAABQg1ksFoOeOl+REeE6mpNb4Zi4YLfOC2N2kSSNdX6sux3fGhODpzC6tVU
jW1nVKtai82ItenxwiCKCpRWHTsymu+urMt1xfrDu62tT+ziLWjew6Ir2VtmCTt3wPbWsTMl2s94aE6rzG1mUE
m3WsBZBahFzovLYnePTikNevToqRDObnbjmqxeHyOmW3t1yYqbe9mM+DWhmUY8ki67qaFWUzaT9uSdmDd6zqFS
39LCqiZOKpSqY+t8rRTczOkatx1cnAAAAANRwzZs21qALeir76HF5vb4Kx/SKKqj3G27OL/1OTzk+ktlO+uWcq
Bm8Pr/mbnGr2C2lJluUXezTysNexYWbdcGbxYr/R6H6zyjWj2meO17ns5Oe9UiO6PL3SxT3TKG6/qtIb6z9Zdl
62X8fHvI/JaHZZJItSPox7cSfmc7xFq054lWuO6+1R7xyuv1qGWPWj2kercvO6o5ewZX/BkBK6Cil3mZOijqBg
g8AAAAAaoGh/VLVODFehzIyKzxvM/vV15FXvaFqkE5l6/Va1AxZTfW75KwNNmd5FfFEgWyPFerPC5z6+MpQtWt
oOb7cE+X1wOvKdHM3qxZOCFO3BIsGzyo56X55/2tfrk+vrnGpVYxZX1OTplt6BOuOhaWaueFEydemgVlN7Cbdv
7hUuU6/XF6/nvqxTIcK/MooOvGcw1sG6ZpOVvV8oOgTP3Vq5iWhCg+Wbvm8VK+NCtWra9xq/VKR+vynWFuz+Rq
rDH6TWRo9XbJw97jKQMEHAAAAALVAjMOukYMuVLHTqdKyijfVaBJSplah9W+pblPXPs2KeFGhZvdvD4bhWjcwa
8OfI7TypnDdOiNY139Sqm1HvfL9d+Lln7pbdUPXYHVNtOj5ESFqHWvWf9af+r+tzy91S7ToicEh6ppoOaTuwbq
5W7BeW3viMVaLSR9dEaZdx32KebpQYY8X6rsDHl3UMqjcvfOeHhCiPXdEavMtEbqOrVXTfnBpSEqQrBbpsaVl+
vGGMN3U1arrPnFW5dtTb5h63SI16mZOjDqDgg8AAAAAaok+PbqoXasWOph++JRjetvz69VS3QbuLMONfVIOS6n
RUXCGgiOmtYwxq3uSRdOGhKhzvFkvrHApMeJERdGuYfmqom1Ds9IKKl6aLkmJkaaTH9PArLT8Xx7TPcmiDX+OU
N69kcr4a4QWXhOu4O6fmjsqrkV2HPPqnc1uPTrIpu8PeNSvqUUNw826or1V6zJ8KixjGfjv4Y9pLg1+OOgYdQo
FHwAAAADUEjZbsP4wtL+sVqty8vIrHlOPluqGeQv1rvURJVqLjI6C38Hnl8q8UjOHSUmRJuO8Vr7M23Xcp6an2

```
eCiT7JFO4+f2WPsISY1DDdr93Gv1hzxaUwb60lj/H6//rSgVM8Nsyki2CSvT3L/9/I//6+Xfu+c+U1mmS55VbK
GGh2lTqHgAwAAAIBapGObVhp4QU8dycyWx1Px5gP1Yamuxe/S25qqViEV7yyMmun+bOq19KBHB/J82pzl1f3fl
Or7A15N6GiVyWTS3RcEa/oqlz7Y5taeHJ8e/LZUO475dGPXXza5GDyrWC+t+mWZ+l29bVpxyKsnfijTnhyf5mx
26/V1Lk3u+ctj3t/q1vcHPNqX69On09wa+naJLmkTpGEtTr7/27/XudUwzKTRrU+Uf32aBOnb/R6tOOTR88vL1
K6hWY6QU+/qi9Mz9bpFatLb6Bh1DncyBAAAAIBaxGQy6Q/DBmjn3gPan35ErVKaVDiutz1fh8tsKvFZqjlh9Xj
N/Xd1j8oyOgbOUnaxX9d97FRGkV92mOmd4k9sjDHOvOXblN42lXqku74qVY7Tr87xFi26NkwtYn6Zn7Q3x6djJ
b/M2OvZyKKPrwzV/YvL9MiSMqVEm/XP4SGaOOmX2XkZRT795WuXsor8Sow06bpOVj3Y33ZSvqwinx7/oUw/3Rg
eOHZ+I4v+mmrTqDlOxYWbNPMSZp6dK19Oc5lZmlslTH6/vO5OLCOoKJDdbld+fr6ioqKq/fk/2plR7c8JoO4Z2
zrR6Ag1Dt9fAVQWo77HGv1zKuqOdVu268W33lWOPUoxDnuFY9JKbfo6J7aak1W9JOqf1NWOTUbHAHAW/DLL9Mc
vmb1XRViiCwAAAAC1UNf2bTTogvN1OKN+LdW90/ka5R5QGw24j3KvClHwAQAAAEAt9PNS3VYpTbQ//cgpx11gz
5cjyF2NyarOVSXzNMWx1OgYAM6Sp2l/mfrfY3SMOo2CDwAAAABqKXtkhMaOHCyz2XTKXXWtZr8GR+cqyOSr8Hx
tMcz5lR6Nni8TexsAtYo7tKGCrnhL/OGtWhR8AADAUF6vV+++8LRuGdxLV3VurluHpur9V57Xr28TfGjvbk275
Xpd26O1ru7aQvdcdpGOHjlO2msvmPmGbh/RV1d1bq5JA7rrrWkPyVVWGji/dP5HmjSgu647v63emvZwucdmHOr
XbcP7qqSosNJeKwBUha7t22hgas/TLtWNtnrUx15xAVgb9ChdqRcds2t9SQnUNz6ZFTT+bSm87tOLtKZhF1OAA
GCoT954WV+9O1O3P/mCklu21t4tG/XS/7tLYRGRGnXdTZKkzLQD+r+rL9Hgy8brytv/prCISKXv2algW8gpr/v
D/I/OzrNPaPLjz6pl1546cmCvXrr/Lkkm3XD/wyrIPa5XH/ibbpv2vOKTm+rxP12rjr37qMfAoZKk1x+5X9f89
f8pLCKyOt4GADhnPy/V3bXv4Gl31WOV5lSGK1i7SsIrPF9TtXDtOltRr8lmqri8BFBzuS+8V7amqUbHqBco+AA
AgKF2rl+jnoOHq/uAIZKkuMbJ+uHzT7Rn84bAmDn/fFLd+g/SdXc/GDiWOKTZaa+7Y/OatenWUxeOHhu4bt9Rl
2j3pnWSpKzONIVFRqrPyDGSpA69LtChfbvVY+BQ/bDgYwUFBan3sJGV+EoBoOo4oiI1duRgvfTWXB3PzVNstKP
CcRfY83XMFawcj7V6A56jOM8RzQ17WpHmMqOjADhLxUl9FD7oXqNj1BssOQUAAIZq3bWHNi//UUf2TvUrVLXfoMkST6fT2u/X6ykZs31yI1X6YYLOuq+KOZp5Tdfnva6bbr2ON6tm7R7O3pJUmb6Qa1buljd+g2WJCU
2TVGZO6l92zarMC9XezZvVNPz2qkoPO9zpz+jmx58vApfNQBUvq7t22h4/1RlZB2Ts7TiQizIJA2OyZG1Fix1j
fTm6T3ro2oYVLd2AQbqg1JbA4VfO5v77lUjZvABAABDXTrpNpUUF+qOkf1ktljk9/k07u4H1K3/IDyjY4o46hHk0KBNxkNf6v87UnvkHXW+uk+RgEY4x+/Pb1lQ+2AYfNSz2vHrPEcoZMptGS41gC5iWabSdI+Z9E/6VZGP1mJ4Vaxr0F8cj/sUnIh+QtT9DM7Sk7Tx5V+tWFM7TvO3rhn4ziRnbTr9SQvzG1+S27/Ij1QU5z8QrQUL3lI
```

```
AAOqhC3pOOdgRg5Wbn6+8glPfDy8qyKuRDY4p7HeUfLc439QNjrXn/HgAxsnxhuvwwBeUcv4Io6PgNCj4AAAAA
KAeMplMGjmor4b1uOCHM7JUVFJyyrH2/87kCzmHku8y50e62/Ht74kKwCDHvOHae/7jajdgnNFR8Bso+AAAAAC
gnrJYLLpi9DBd2Ku7DqQdVrHTecqxDqvnrEu+gaXfaprjY5lNFW/mAaDmOuoJ19b296nbyOtkMpmMjoPfQMEHA
AAAAPVYiM2miZf/QX3P76Z9Bw+pxFl6yrExVo8uijOum8n3m9ftVLZOrOTNlNXO++/fB6B6ZXsitDbldqVeOkk
Wi8XoODgDFHwAAAAAUM+FhYbohivG6ILunbX3YLqcpacu+WL/W/IFn6bka+bao7cjXlKo2VOVcQFUoSxPhFYl/
On9rpysYFuIOXFwhij4AAAAAAAKDwvVjeMvVWq3Ttp7IF3OOrJTjmOQ7NaoBscUWsFy3QbuLMONfUp2y6lLQgA
lU6Y7Uqua3KpBE+5SWESUOXFwFij4AAAAAACSpIjwMNO4/lL17NJBe/YfVGmZ65RjY6OejW5wTJEWT+BYmLdQc
61/V4K1uDriAqhER9xRWptyq4ZMuFNhEZFGx8FZouADAAAAAAREROTrpqvGqmfnEyVfmevUJV9UkFejGxxTlEp
k8bvOtqaqZUhe9YUFUCkOu+1a13yyBl99pOLDI4yOg3NQ6QXftGnT1LNnTOVGRiouLk6XXHKJdu7cWW5MaWmpJ
k+erNjYWEVERGjcuHHKysoqNyYtLU2jRo1SWFiY4uLidPfdd8vj8QgAAAAAULXskRG66eqx6tqhjXbtPSiX69T
3OvMWHVWzTa/p4aKn1DO865TjANRMO8ritL7FnRpy1eOKCQs3Og7OUaUXfEuWLNHkyZO1YsUKLVqOSG63W8OGD
VNx8S9TtO+66y7Nnz9f77//vpYsWaIjR45o7Nixgfner1ejRo2Sy+XSTz/9pJkzZ2rGjBmaOnVqZccFAAAAAFT
AERWpm68ep87tW2vnvgMVzuRzFhdpzO8L1cBUKG/yhUoLbmVAUgDn6qeSZtrd5nYNveoWyr1azuT3+/1V+QRHj
x5VXFyclixZon79+ik/P1+ik/P18NGzbUnDlzdNlllOmSduzYobZt22r58uXq3bu3vvvvzyS1188cU6cuSI4uPjjJUmvvvfa
a7r33Xh09elTBwcG/+bwFBQWy2+3Kz89ZlZ1f6cAOqesa0TjY5Q4/D9FUBlMep7rNE/pwJn63huv
v797odat3mHWjRLVljoiV01Xa5SbV0yXxGeHKX27a+oKLvk96tbyfdqU7rO4NQATscrsxYWtpG/yzUaevkfZQs
JNToSfqcqvwdffn6+JCkmJkaStHbtWrndbgOZMiQwpk2bNmrSpImWL18uSVq+fLk6duwYKPckafw4SooKNDWr
VurOjIAAAAA4L9io+265dordEGPztp3MF1FxSXyetzatmyhQsu0q2ev1BPlniSZTFoXPlDrwvqrSmeSADhnZQr
W3PyuUrfrKPfqkKCqvLjP59OUKVPUp08fdejQQZKUmZmp4OBgORyOymPj4+OVmZGysjKVl
f2yjXtBQUFlvQwAAAAAqNeiIiNO89XjFBJiO7fLVsmbtUvBRRnq2qOHYmIbnjR+R2gPFZsjlVqOUEHiXupATVG
gSL2b2OnJvf6gIeMmKtgWYnQkVJIqncE3efJkbdmyRXPnzq3Kp5F0OYnMPu90e+EhOTq7y5wQAAACA+iIsNEQ3X
D5Gg8/vqBBntlo2b6bEpManHJ9ua61v7FeqyMxSdKAmOOJvqBl5vdSi35UaetkNlHt1TJUVfLfddpsWLFig777
7ToOb//JNPyEhQS6XS3l5eeeXGZ2VlKSEgITDm1VdfrfL5fn7gIz09vRAQ0AAAAAgONiqGyaM1
O233qmo8HAdyzh02vE5QQlaaL9GR6zNqicgrBt8DTW+85UpY75owZZdeq2swTajI6GSVXrB5/f7ddtttuvjjj/XNN9
Wt99+q5SUlHLnu3fvLvqVqsWLFweeS01N1YFwKz7/4lZXi6XpaamavmaVPmzmmPcYwdZeq2swTajI6GSVXrB5/f7ddtttuvjjj/XNN9
Wt99+q5SUlHLnu3fvLqvVqsWLFweOdy5U2lpaUpNTZUkpaamavmaVPmzcrOzg6MWbRokaKiotSuXbsKn9Xn9dmsykqK
qrcBwAAAACgclmtVvUfdYWGXDZRPp9XGX/j36HR7N7rMofo+cqy2ZPQaM1Nlc5ds
xwACVfg++yZMna86cOfr0008VGRkZWDNnt9sVGhoqu9GuiaCc39iXvmoxjOoy8iRKCevX8yRF9XBRCmvs3m9YWXnH2vYqAfDI6eCnZHwDe6sYL7I
zt2rXTtddeq6efflqZmVkaKLjaXtnxn39x1HTk
...
```

tOWWv71x9q0/Dvl5xxVXKNmsgYHn/axRRaHfogaozh3uroVf6cY79FqSg2cuwxrU20Iu1CZ7khl7tun8CiHBoy
+WtOuHKaQMHaNxtmh4AMAAAAA1BgRUQ4NGTdRzVp31PJFn+rw/p0Ki4hSbEJjmc3m0z4225pc7v58of6SakoNn
LnjlgRtCL9QGZbGOpaRrtKSYzqvU0/1GTFOCcnNjY6HWoqCDwAAAABQo/w8m69Jq/basnKJ1iz5Qgd3bVFMXKK
iohucftmiyaS9IZ1OONZGbZxr1bp0nWz+0uoLD5xCgTlaG8P6Kt12nooL8pR9ZIsaxCdp4JgJat/jQgVZrUZHR
C1GwQcAAAAAqJFsIaHq3n+EWnbsrrVLF2rzyiVK271VcY2aKjQ88rSP9ZiCtSUsVTtCu6tl6Ua1da5VqL+4mpI
DvygxRWhzWKr22TrIWepU9p5tsgYFq2ufIUoddqmiG8QbHRF1AAUfAAAAAKBGs8c01KBLrlWbrqlatXi+dm9eK
5kyzuj+fB5TsHaE9tSukK5qXrZF7ZyrFeErqKbkqM+cpjDtCO2uXSFd5XR5dfTAHvl9PrVo20U9BlykZq07sYk
GKgOFHwAAAACgVkhq2lJ/mHin9mxZo5WL5+vQvhOKjbCrwRncn89nCtKekC7aa+ukZq7taudcJbs3p5qSoz4pM
Edre2gP7be1k8vj19HONLnLSpXcoq16DLhILdt3lyWIOgaVi68oAAAAAECtYTabdV6n89X0vI7asmqp1nz/udJ
2bZGjYYLsMQ1/c0aU32TWflt77Q9up2TXbrV3rlSMN7ua0qMu0xqUqO0hPXUouKW8Pp+OHzmkkqICJTZprh79R
6pN196yBtuMjok6ioIPAAAAAFDr2EJC1b3fcLXq0F3rfvhKm1Yt0cFdm2WPiZM9Nu43Z/TJZFK67Tyl285Tomu
/2jtXKs5zuHrCo87wyaxDwS21I6SbjlkbyefzKTc7Q4W5x9QgobEuvOhyte95oULCwo2OijqOgg8AAAAAUGtFx
TTQgDET1LH3AG1etVRbVi1R2q4tioyOVXSDBJktlt+8RkZwijKCU2T3HFOLss1KKdvGzrs4rVJTiPbaOml3SGe
VWKLk9/uVfzxbeUcz5WgQrwFjJqhTrwGKsEcbHRX1BAUfAAAAAKDWi41vpAGjr1KXCwZry+oftHnFdOrbvVXh9
mjFxCXKYvntX3/zgxpoXdBAbQi7UMmu3WpRulnxnnnSxDQJ+lmNpqF0hXXXA1lY+U5B8Pp/ysjNUkHtUEVHRSh1
6ibr0HcrOuKh2FHwAAAAAgDrDERunviPGqXPvgdq2dpk2/rRY6Xu2KzQ8UrHxjRRktf7mNXymIB2OtdVBW1tFe
HPVonSzmpdtVai/pBpeAWqaEnOEDgS30QFbO+UFNZQkeTxu5WSnqaQgT44G8bpg+Di179FXDROTDU6L+oqCDwA
AAABQ50Q6YtRr8Gh17NVf29ct14Zl3+jw/p0KDglVg4TGZ7zZQZElWhvD+2lTWF81cu1Vy7LNSnAfkFn+Kn4FM
JLLFKz04FY6YGunrKBk6b+bt7jKSnU887BcZU41TExW6pAxats1VVExDQx0jPq0gg8AAAAAUGeFRUSpe7/hat+
jr3ZuXKn1yxYpM22vTGaLohvEKyzS/ps77Oondt89ZGulQ7ZWCvMWqHnZViW7dinae6waXgWqg1dmZVqbab+tr
Q4Ht5DXdGK2p9/vV3FBrnKyM2Q2mZXQpLk6pw7WeZ16KjQ8wuDUwAkUfAAAAACAOi8kLFydUwepbbcLtGfLWm1
bu0zpe3foaEaaIqKi5WgQryBr8Bldq8QSpS1hqdoSlqpwb56SXXvU2LVHDTxHmNlXCx0LStR+W1ulBbdWmTksc
Nzn9SrveLYKc48pNDxSbbr0VoeeF6pZ605ntNQbqE4UfAAAAACAeiPYFqJ23fuobbcLlHXogPZsXafta35UxsE
9kkxyNIxXRFTOGc3qk6Rii0M7QntoR2gP2XwlauTaq8auPUpwpylInqp9MTgnHgUp29pYGdYUHQ5uriKLI3DO7
/erpKhA+cez5XaVyR4bp9ThY9WmS2/FN252xl8XQHWj4AMAAAAA1DsmkOkJySlKSE5Rz/4Xaf/0Tdq+brnSdm/
V8YxDCou0K7phwhnfq0+Sysxh2hfSUftC0sridyvRdUDJrj1Kcu+TzV9aha8GvyXfEqMMa4q0WJsp29pYPlP50
qTMWaK849lyFhUoNDxSyc3bqHXX3mrZvpsi7NEGpQb0HAUfAAAAAKBes4WGqU2X3mrduZeOZR7S3tKt+rtHXNMmW
m7ZPf75ejQZwi7DEym81nfE2vyRq4Z5/J71NDzyEluNPU0H1YsZ5MZvdVMZcpWFnWJjpiTVGGtdn/b+9uYps69
zy0/47PsY+d2I6T0K/gQMpbp2+8lMK0iysxU00iZRYzI1VdFAkSqWolWLFjU9RVd203tKwg3SC6ibroohukLoq
qqW6rO3PnDnAL9IYQ8v7ixPG7z5lFqHu5SZleQnJ8zPcjWT5+HO0/EbHIV098lDXja76mXCoqPTejTHpBwVBIy
a6UXvuXf9f0fS+qfds0duvBVwh8AAAAAABodVdfW1dKbV0pvfy7fv3l5h918w//qTs3/ktjt/5XAdNULNGqWKJ
Fpvnbf5x2jYCmgz2aDvZIkgJuRS3lKbWVx9VWuqdk+b7C7PDbkLIszVvtmg6mNBHcqVmrW66xNsg6lYqWFue0N
D8rwzDUnGzXi0f/Tc88d0Dbdu6VaZFJ4E/8ywUAAAAA4G8EQ7b2vHhYe148rIXZYKY3fuamfbvy37v74J43fuSl
JisabFWtOKhj6bSfn+JljmJoNdms22K3rkVck11W8Mq+28rjaS+NqK48r6qQ342XVhtmjc7NG91as7q1
JLZsm7Qk6RKuazMOoIyi/Mql0uKJ1q1/9Vj2v38IfXseV52pGHdxz1pJ0+e1GeffaZ33nlHFy5ceOi+U6O6ZN
PPtGJEyc0OPDy8JfOgvhD4AAAAAAB4hOZkh5qTHXrhyP/bVj0+UrVrU+P83NfrnP+mn63/Q0Pl4sM
gwtWa1aslp10/yWJCniZJcbGBUMmnAyNyOypVFncWn7W9mtJxeUC6
zJMMIKBpPaNfzh7T3pVe0Y+8L/PipV+AXnGA7/L3/zaw5BkJmm3AAAAAPD4PjmAtYtvRSMjuzj2fKYlsUSpkcTVmc/JbvTp5S8A6Q1y9CJ0U0wWj
4jNRzFlQtJJW2M0q7ORkO
9kHx9kHxzkF5GzwVf/9yrJUMmxlAzFlzLgygUQ15K2YTVoJxB65I289juMot7KsTHpe+ZUVmZapaFOznj34j+r
Z87y6enYp2ZWS+ZiBdTMNDQ3p0qVL1cB38eJFDQ4vJFDQ406uuvv/Z2MPgagQ8AAAAAgA2ygkFtf2aftj+zT+6D
pvWzMRdTY/f1djt65qbHNfE6C05TllW0FZDLK7GaJ0CdnjjZ2s1DOXMmHJmTNOP6oeuq5CbV9jNynZyCjtZhd0
VBd2iDNeVIUcBrV4briNDrgJ/dWzowZrrqGJYKhkhlQxbRcNWybBVCoR+WRKrUjFuVjbC8W7t2K4K+axyK8vKr
WRULhTkylWkMaZk53Y98w8H1LVjlzpTu9QQjWubC8ePHdfbfsWY2OjkqSrl27pitXBAAAAAnqB
AIKDmtk41t3Vq70tH5LquVpbmVz6/EnUyjUSthZWzS9P3VCrkFTAtRaIxhRuiCsm7e75EMQ0UjoqIiU
u1tbJOOGvOKhfyDmLesYj4nyZUdblCkMaade19Q947damnvUrIrpZb2bgUC659co1a1tbVpYGBAw8PDcl1XAwM

```
DSiaTXo8FnyPwAQAAAACwiQzDUDSeUDSeUO+zL+mVYwMq5HOanRjTzMSYJu7e1r3bN5TNLGlxZlKVSlmGDAVtW
3akUXakQXY4ItMKbny3X41wXVelYkHFQl6lQk6FfE7FXE6u6yhkRxSJRrW9d5+6d+5Wa3u3Wjq2qaW9SyE77PX
oT8TQ0JBOnz4tSTp//rzHO6AeEPgAAAAAANhidjiibb17ta13rw689s+qVCrKpOeVnpvW4tyMFuemNH3vL5qbu
q/scloLM5OqlEsyZMgwAwrZEYXCYVnBkCwrJCsYlGkFa2Y3m+u6qpTLKhZyKhXyKj64lEtFGYbkulIwZCtkhxU
KR9TVO62uHbuV7NyulvYutbR1KdzQ6PXL2DT9/f0qFosyDEN9fX1ej4M6QOADAAAAAMBjpmmqqaVNTS1t6tnzy
3q5VNJyel6Z9LyWF+a0tDivhdlJzU3e0/LivIq5rLKltMrlksqloiTJkORqdeegFQzJtIKrITAYlGlakiEZMiT
DkGEYDx8/2CFoGL+suY6jSqWiSqUsp1J+cF1RpVKRUy5X1x3HefBnrQY+07IUCoUVCoeVaG1Xc1uXEsl2xZpaF
IOn1NjUrGgsocZ4Qqb1dOUJOzR1/fr16jGwUU/XdxAAAAAAAD5iBYNqTnaoOdmx5r5yqaR8bkWF3Iry2ZXV4+y
K8rms8rkV5TLLWk7Pa2VpUdnltIrFgsrFglzXleu6klZDnP7mdnVNq9eGEVDAsmSapkzTUsAOZTdEZIcjshsaF
Y5EFW5oVLihUSE7rGDQViQaU7SpWdF4sxrjCdnhyJb+vflBPB73egTUEQIfAAAAAAA+ZAWDigZXP9vv//PzySt
KhfxqxHMcOa5TPXZdR47jSK4r569uu44j07IUDNkKhsLVX6t92nbcPQnDw8OPvP+LL77YkjlQn/iOBAAAAAACgz
hmGsbrjjp1OQF2qjU/fBAAAAAAAAPBYCHwAAAAAAAAACAjxH4AAAAAAAAAB8j8AEAAAAAAA+RuADAAAAAAAAAfIz
ABwAAAAAAAPgYgQ8AAAAAAADwMQIfAAAAAAAA4GMEPgAAAAAAAMDHCHwAAAAAAACAjxH4AAAAAAAAAAB8j8AEAA
AAAAAA+RuADAAAAAAAAfIzABwAAAAAAPgYgQ8AAAAAAADwMQIfAAAAAAAA4GMEPgAAAAAAAMDHCHwAAAAAAAC
AjxH4AAAAAAAAAB8j8AEAAAAAAAA+RuADAAAAAAAAfIzABwAAAAAAPgYgQ8AAAAAAADwMQIfAAAAAAAA4GMEP
gAAAAAAAMDHCHwAAAAAAACAjxH4AAAAAAAAAB8j8AEAAAAAAAA+RuADAAAAAAAAfIzABwAAAAAAPgYgQ8AAAA
AAADwMQIfAAAAAAAA4GMEPgAAAAAAAMDHajrwnT9/Xjt37lQ4HNbOf13XffeB3/4d0QS1fffVGjrPm6Q90
nPmjM6dO6cffvhB+/fvV19fn6anp7OeDQAAAAAAAKgZNRv4PvzwD7397Xvf7f7tcaEyCLMBoO7Pvex6gvQ8
AAAAAAACgZlheD7CeYrGo77//XmfPnq2uBBQIBvf766/8/766/r222/XXVfHUhCLKjeTqfTkqSlpaXNHfZZXDPLn
LWJ6+/KEsUEBCLtgs5lWnk3A1nk0VHo3F+U8ObprXHVcEDPUFKQD3Jnm8BlADq3e3TDbVwQ1DJh/a7rQAxhdXIWsIUlXH
CrREhdRPeWjPVQdm4DArzpUzjjCzxSS+ZrZo0jGymrCEQeiCT0i+x6KSuzaHM2w2nozRLHV7nLmv5S/yTsYNdwAAt9hhJBdx8RWUsBvkdXzGwNOJcFXDIsrBwAAAAAAAJ+g6E2WHQ20bzJiZrMy/z5v9fW1cJUbp9zTIk8/
A7xAAeKbizGbdoMZQV8x0H+3ILK8EPknVE3/Vb7v2JMDEBBvbAQmAE9gZtlXQ0dDF/YMQ3bn6g1QNy9eGZa+y4aCrb9MUYdj
N5W+5ebe+6jMMTbaDH0VGQD4BtVWkfRAN2zTALHc+jM4rhmppWuKTl8AvrojEkyAfGuUkAsLE5rkGr/kUFavzEvb+3NPEA+fKdBmpmXFDQYrZa39HW6RWV9o
Wj/MpOWM0m1ZmRE56n+bNwpz/pF0oovX1Lt4XtegAuNUhPN00TUkxK0rbAvJfK1EF/0OfNoeakVNfOvE/0Fu
UOeMj5eBQoDNnmSIZyb5ycjUt/VjcsTPJZe6bVSxsUMv6eZ4KlUh+N3nYUZyjz6vZ/5PYb+Yt3DbwIVBGvSPjx7/kJDq8JKvi5zJRLXdhPr
XAAAAAEAAAAYYiTKoPOk4HAAAAAAAAAAoGkmiDYJ/BAAAAAAAAMzD4IfQ+LBb6LJqJDL8KZn3bGNz8nXs8vOqLY3JpX1a+Pd5Bv8/S6IHWe+hkJR33qBzO2AWy8b3u+X8
eUDkeI93cMQkzl0gx0fGgIqHHGN+O6QWI0LfvlB8ABKbOgFrM2IswhAvFHYM5x+F13wAnmgVjvDahY6n/8/e4j/fvBp9GWAAAAAAAASUVORK5CYII=\n"
```

That attempt was wrong; ignore.

```
      "metadata": {
        "id": "jgMqzZwIE0pX"
      }
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "ADeE_-bBzM2g"
      },
      "source": [
        "#### Type Income"
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "analyse_categorical_column(\"Type_Income\")"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 507
        },
        "id": "6Hl7PuUj2DUc",
        "outputId": "785f83c1-261c-4e22-eefb-7598b77d0f40"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 1400x500 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAABT0AAAHqCAYAAADGeeqIAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAADZtUl
EQVR4nOzdd3hUZf7+8XtmOnshFQKhSRGkCYKoSFFQdFVY2yKKYkMsqGv7rrosFtS1/ey7KyuoYFsBO4pI74TeO
wHS26RP//2BjsYEJJDkZJL367rm0pznOefM8bfcM8bM5JOnmDwej0cAAAAAAAIAmhaInAAAAAAAgCaFoicAAAAAA
AAACAJoWiJwAwAAAAAAIAmhaInAAAAAAAAgCaFoicAAAAAAACAJoWiJwAwAAAAAAIAmhaInAAAAAAAAgCaFoicAA
AAAAACAJsXP6ACnwu12KyMjQ+Hh4TKZTEbHAQAAQAPzeDwqKSlRcnKyzGb+jg8ICqfLLomZGRoZSUFKNjAAA
AwGCHDx9Wq1atjI4BAACARsYni57eGSn3ijYiIMDgNAAAAAGlpxcbFFSUlK8nwsBAACA3/LJoucvU9oojIioe
gIAADRjLHUEAAAAmraAEgAAAAAAIAmhaInAAAAAAAAgCaFoicAAAAAAACAJsUn1/QEAAC+xeyyeFwGB0DPsT
f318Wi8XoGAAAAPBRFD0BAEC98Xg8ysrKUlFRkdFR4I0ioqKUmJjIyZUAAACoNYqeAACg3vxa8IxPT1dQSkAjFK
5wUj8ej8vJy5eTkSJKSkpIMTgQAAAAfQ9ETAADUC5fL5S14tmjRwug4kMLjDHEAAAAAAAD5WSSkdGpkbeOEieYdeG
qfcN6yxeDpAWqN2SWMc/AkrgFeYJw+i5+BisPDrY1AvO8bRk8ItKjl0YHSWYqZlAlB5OFOpA+Pp9Gjpu4GM/9tvjTR2J
IHjc0PhfXnBuGUm7UKgD4fDs7oGOAbm/9lhOmmb5wOKeAHKr1nr5a7VbbHfLjr6Ov7gNz3x5Sm2xjQMpAOq5NNjo
ozrzszFSzycMHKMPfrgQOOKhxqQhJgDzuozxMSPs+vGqVoJJwJhQCLR5N2rLjwrktChn7D9jhULmrGkwGqQ+LnT6vY
0Xay/0mLeq3DmZ6+uhnQvKzM5KAGrRBa7aAhxyEQVxGT/zjJzZ6rPEKBDCT+ZRMcSiDDRLRqtvlHOaIRHhAcfo
pNhXt3xz7rPdG3rZNOENUf6kx/uEWaKJC01QrtuPAmuXOI0TMa07Hf7r6SKp6yldrp03vA5DQlZVqPbYvWIOK
```

```
tWiSTyaRvvvlGZ511loKCgtS/f39t3brV22f69OmKiorS3Llz1bFjRwUFBWn48OE6fPiwt8/kyZPVs2dPvfvuu
2rbtq2CgoIkSenp6briiisUFhamiIgIXXPNNcrOzpYk7d69WyaTSTt37qyS95VXXlH79u29X2/dulWXXHKJwsL
ClJCQoLFjxyovL8/bXlZWphtvvFFhYWFKSkrSSy+9VKevHwAAAGCkWhU9n3/+eb399tt64403tGPHDj3//PN64
YUX9Prrr3v7vPDCC3rttdf0zjvvvaPXq1QoNDdXw4cNVWVnp7TNmzBht27ZN8+fP19dff60lS5bo9ttvr7tnBQA
AcJo+/fRTde7cWZO6ddINN9yg//73v1X+OCtJDz3OkF566SWtXbtWcXFxuvzzyy+VwOLzt5eXXleuaZZ/T+++9r+
fLlKioqOnXXXVflGnv37tXnn3+u2bNNna+PGjXK73briiitUUFCgxYsXa/78+dq/f7+dq/SdZZ5yhs88+WzN
nzqxynZkzZ+ovf/mLJKmoqEhDhgxRr169tG7dOs2bNO/Z2dm65pprqmRfvHixvvjjiC/3www9atGiR1q9fX6evI
QAAAGCUWq3puWLFC1lxxRUaOXKkJCk1NVUfffSR1qxZI+nYKM9XX31/uK644gpJO/vvvvv6+EhATNnTtX111
3nXbs2KF58+Zp7dq1OvvssyVJr7/+ui699FK9+OKLSk5OrsvnBwAAcEqmTZumG264OZI0YsQIWa1WLV68WBdee
KG3z9///ndddNFFkqQZM2aoVatSoDhXRrBwAAAGCkWhU9n3/+eb399tt64403tGPHDj3//PN64
+/PnasmWLDhw4oJSUFEnHPk+deeaZWrt2rfr27asxY8bojTfe0FNPPSXp2OjPtLQ0ffjh5KkN954Q7169dKzz
z7rzfrf//5XKSkp2r17t5KTkzVt2jR9+OGHGjp0aJX8AAAAQFNQq5Ge5557rhYsmkks
ukSQdOHBAWVlZGjZsmPecyMhInXPOOVq5cqWkYxsCREVFeQuefekjRs2DCZzWatXr26vvabDYVFxdQAAAANSXX
bt2ac2aNbr++ms1bdqOKvOGDBjg/feYmBh16tRJ0Pd3759vV937txZUVFRVfq0adP7t2t2
GW/CUpBO7diglJcVb8JSkrl27VjnvuuuuuO8GDB7Vq1SpJx0Z59u7dW507d5ZO7DPawoULFRYW5n38Orzv3z7t2
7dPdrvdW4z9bX4AAACgKajVSM9HH31UxcF6ty5sywWi1wul5555hmNGTkpSVlSVJSkhIqHJeQkKCtyOrKOv
x8fFVQ/j5KSYmxtvn96Onap//OMftYkAABwyqZNmyan0l1BorH41FgYKDeeONONr1XaGhorc9JTEzUkCFDN
GvWLPXv31+zZs3ShAkTvO21paW6Ll9fzzzzc1c7NykpSXv37jzCAhjd58mTNnTtXGzdurLF9+vTpmjRpkoq
KihoOFwAOVrUqen766ae66eaeaOXOmZs2apTPPPFMbN27UpEmTlsxs+ZLPk+XC4yuiHhvLbz
QnQ9HXr1s3oCAAAAzidTr3//vt66aWXdPHFF1dpu/LKK/eeffVWdu3bJHF1dpu/LKK/XRRx95RO2uWrVKrVu3liQVFhZq9+7d6tKlS5S5Vrrvu
3zjuVfdeuXSoqKqrS5/e6d0miw4cP6/Dhw97P09u3bXg9PDDuv7667V/4/qa4X27t1bn
3/+uVJTU+XnV0vVv27eXv76/Vq1dXyz9oOKBavV4A0Jy98847euihh6GNavV4A/Jy98847euihh1RYWOj9eVtaWqro6GgNHDhQixYt8vZ
dtGiRBg8erL1791bZeK6uXHvttbrO0kvr/LoA4KtqNb319ixY3X/X//8fdr6tSpko6NO
pDk3V30F9nZ2d62xMRE5eTkVGl30pOqKCjw9vm9wMBARURE+rVhXzu+Nsb+Gz9WnFsPkqZ
cpOyZogULFmjr1q0aN26cWrRooSuvvNLb7u/vr3vuuUerV69WlqaxoObp/79+3uLoDUZNmyYunfvrjFjxj9+
vVas2aNbrzxRg0aNKjK8kCjRo1SSUmjJjkyYoMGDB1cZl4x4kQVFBTo+uuv19dh/BbTo+uuv19q1+a7Vv3z59//33uvnmm+VyuepdX+2Vy
WFqbx48froYce0k8//eTNbzbX6qMhADR7gwcPVPmlpdatW+c9tnpUiUmJmr16tWdS7WSVyu/Wa6xoB1cgdVePhMkH2c+aXOx8fjK
dPp/MN+wcHB1WZVAkBzVqtPtuXl5Z5dOlFnQbdblHS27ZtlZiYqAUFBd7z7HY7+vfvr9dffVWS1K9fP02YMEGPP/64EhMTtWPHDj3/+
8fX766Se53e4q6OoBAAAYYdq0aRo2bJgiIyOrtV199dVat25dlbU3J06cqD59+igtLc27d+9cD+dWquVvgsB9CbX6TsHBc5oH5U8Sy0+o8uObZdm2Kzl4AfMMsfY/rSPnPHrbYIAk
M/SI5OVnLly+Xy+XSdfr07vu2vSpEmKioryfpb8zAYDLmmYYn8FasJgmLrxdF6sDBTBqOCuHdYn6ZMU0IsFbVR6
MOfEhKiRx55RH/5y5y55RH/5y1
SkpqdqIziuuuEJt27b1rr38y/HBgwfLZRPp3nvvvVX8qIKCgnTeedp7dq1VfqZTCZ999rVR3sKCv6bzc2eq2R
q9963b5/atWunu+++Wx6PR20+ZP9RnT1dUVJS3ffLkyerSuddd51KSkjyaLFg6vjWyeu6QhQe0Kqrk2pjqnRmzr4gb
SlJr7zyii688nADBCraa3X3755XrmmmWfUUnVrnXnmmdqwYYNefyvl3XXLLZKOfUifNGmnSnn76aXX
s2FFt27bVE088hTNmTcsuh8TsN3xfPbGDbrg1Zj8j9bXaorv5+MTqnbgAb7N5oyy2X2bddOp2w9+M8GDbi3+2JXHCeKf
9dPHo/H+8vteeed94fL34waNUjRo2qsW3cW3y5MmaPHtW7fWF1988YdZ/nkk2ywsDB
98MEH+uCDD7zHHnrooT+8JwcGqsGDB2vhwoV69NFHJR0bf0nwwww/L5XJp4cKFuvDCC1VRUaHu1frl1tu0cMPP
6zPP/9cM2bMUJs2bfTCCy9o+PDh2rt3r2JiYryYrzXffTRR/Xiiy+qXbt2io+6OPh2rt3rJ2iYrzXfJfBTRS/
2ffv2ae7cud5ZDNdcc42x2Yn2ijjTDwU5TjqS4Q0KEZh0U6u
T3xxBO66671JOTo+TkZN1xxxx6168sknvX0efvhBlkZhlZWV6fbbb1dRUZuZHOO+88zZs3T0FBQd4+M2fPGFNlN13S62hQ4f
KbDZr9eOjRei211+ruWQEAAAAAOAAGDx6sSZMyMyelOqqKiQhs2bNCGZkcDjDzOjzvvvSJJWr1wpm82mCy+8ULfdd
pumT5+uSy65RJL0n//OdeadOq/PFpypQpuiiii6rdb8WKFbrsssvt0t7/9TQ8++AJs7ndbk2fP3eG
SpLFjx2rBggV65plVFJSoRkzZmjWrFkaOnSoJOm9999995jMBBkAJqQNWRc/w8HC9+uqevXVXV4/bx2QyacqUZoyZ
cpx+8TExGjWrFmluTUAAAAAAI3OhRdeqLLKyMq1quFaFhYU4ApepHbt2slqtcr
hcGjgwIHe8/39/dWvXz/vnjO/d1zb4jYv3+/HA5H1XWmI/zWz+/HD/2lXcXo9c4cPjBgB
yMj1a1Tp9q+BADQKLFaPQAAC1de0GF8ng8VdZO+71x48apqKiowTIBAaltTChQs1aNAgscf
WV05JSdGKFSu0cOFCDRkypypFbXDQ0NrXYsLi5O/fr100cI+fYuFcIa/v3+fa7+yWsY6V07+FJV3+cVr7+dv
gYPHqxFixZp0aJFuvDCCDCC73HL7jgAn333dsa2aNBg8erPIDEDLly/39E9HFq7dq7B29d7W/9
/raCgIAOfPrzKpkS1la5d0/n7+1fZRMltEadoGfJlSRQvOSiAotOOJjGa5VdtWRq1tOqwEE0AaEwogIAAAAcBoGDx6sZCuWcZuWaePGd6RnpIOaNAg/
```

```
etf/5LdbtfgwYMVGhqqCRMm6KGHHtK8efOOfft23XbbbSovL9f48eNP6l6hoaH65ptv5OfnpOsuuUSlpaWnlDk
8PFw33XSTHnroIS1cuFDbtm3T+PHjZTabZTKZTumaANCYUPQEAAAAAOAODB48WBUVFerQoYMSEhK8xwcNGqSSk
hJl6tRJSUlJkqTnnntOo0eP1tixY9W7d2/t3btX33//vaKjo0/6fmFhYfruu+/k8Xg0ZWVnVLul19+WQM
GDNBll12mYcOGaeDAgerSpUuVjYgBwFeZPB6Px+gQtVVcXKzIyEhZrVZFREQ02H23bt3aYPeC8bp162e62ZOBADwa
ZWVlTpw4IDatm3LL084JSf6HjLq8yAANGVlZWVq2bKlXnrppZMeeQoAjVWtdm8HAAAAABNw4YNG7Rz507169d
PVqtVU6ZMkSRdccUVBicDgNPH9HYAAAAAYIjU1Va+++qrRMQCgWXvxxRfVo0cPDRs6TGVlZVq6dKlatGGhhdCwAO
GOUPQEAAH5j3LhxMplMeu6556ocnzt3bq03dqBoBwBozHr16qW0tDSVlpaqoKBA8+fPV/fu3Y2OBQB1guntAAC
gwTXkOtmnskZzUFCQnn/+ed1xxx212ljCC19jtdgUEBBgdAwAAAKhzjPQEAAD4nWHDhikxMVFTp049cZZ
555pgIDA5WamqqXXnrJ23bhhRfq0KFDuv/++2UymU44O44SvTl119W9+7dFRoaqpSUFN11l0qLS31tk+fPl1RUVG
a03euOnbsqKCgIA0fPlyHDx/29pk8f3r/3rX0pJSVFISIiueYaWa1Wb59x48bpyiuv1DPPPKPk5GR16
tRJkrRlyxYNGTJEwcHBio2oCAVFRVVyXzfdpyJAh3q+XLVum888/X8HBwcDlbbtm01c+bME762AAAAwOmg6AkAAPA7FotzFzz77rF5
2nLli2aPHmynnjiCU2fPl2SNHv2bLVq1UpmMzj3s/s157Tdu2bdOMGTP0008/6eGH67Sp7y
8XM8884zef/99LV++XEVFRbruuuuq9Nm7d69mz56tuXPnavt3b
ZWVafjw4YqOjtbatWv12Wef6ccff9Tdd98tSRo6dKiioqL0+eefe6/hcrn0YySefaMyYMZKkffv2acSIERo9erQ
2b96sTz75RMuWLfNeQzpWcD18+LAWLlyo//3vf3rrrbeUk5Pz/8lAAAAgFPD9HYAAIAaXHXVVerZs6f+/ve/a
9q0adXaX375ZQ0d0d0lRPPPPGEJOmMM87Q9u3b9c9//lPjxrLBaLwsPD1ZiYeMJ7TZo0yfvvvvqampevrpp3X
nnXfqrbbfe8h53OBx6403M4550iSZsyYoS5dumjXXq16+fJKmyslLvvv++WrRoIUn673//K7Nq+zLD6CNZJaz
nnXfqrbfe8h53OBx64403dM4550iSZsyYoS5dumjNmjXq16+fJKmyslLvv/++WrZsKUl6/fXXXNXLkSL3O0kveD
KGhoXr33Xe909r/85//eM8LDQ2VJL3xxhu6/PLL9fzzz2apfHjx0s6VjgtKirS6NGjJUlTp07
VmDFjvM+jY8eOeu211zl1zZRo0CC9/fbbSk9P13fffac1a9aob9++kqRp06apS5cuJ/Jl/lfAwAAAKgdRnoCAAAcx/PPP
68ZM2Zox44dd1dp27NihgQMHVjk2cOBA7dmzRy6Xq1b1Ox13bNlSAeHhGjt2rPLz81VVeXu7t4+fn5+n5yO
YSlLnzzp0VFRVVJVvrlq29BU9Jg9gNxut3bt2uU9Yrre07YsUM9evTwFjx/eR6/PW/MmDF1jhgIRMjIyJ
EkzZ87UyJEjFRUJUnatGmTpk+frrCwMHw6rvd0nDgHhs2CE/Pz/16d0nWn4AAACgPlDOBAAAOI4LLrh
Aw4cP120OPPVVZv9zh48KAuu+wdw96SXqZVsPYySk+bGT9/T6Q+aZ+zenGE4Tfu/+Sr/cuVkrlVbR29sDTtg
oaSk5MlSatWrZLZbK6OfRzBi+8ay6TBNhiJ4ZTu1TQ8OvuvG9Qtww+BAhcJD74ejbPQaNb2uqmtKZn4p9DUdcE7wf8Mxtv/S
RTIWY6bYXaPv+stU6rzcHERNQ6/fD6jzOJvC7bAOY4tVU1ei36LTnnBTRt8xxxGDgFT/2nsjYkUIVnfx9kf0fUPQ
3vP7cbpkc5u76ttdlA1qxSfGR9IHfxgwt3+fTNdpEedPL4ypWsNBsaLFqYh2PMbnuTtYFnsYDBLTEyxLvvDLPzfb
nlMbvbhtBg4ykm5QQWdXenbZ9vOhVBsX3E8MP+uGqnBGh5X3B2JjAyt5hQJLLfO/+DTjN6FJTmb1SM+FdeYncVfvhC
ilIFSa1ViwAeHfDcX/VQWr+9MWQ6HH9Nav5uuR+eP9aauM56Vzr3dk9BYmjvyRpbqXzX8hqOBakNOo/mjYz3
wIQWLwqFfIxrJUrenHB8VVBh4hFHNTxMKcgWvxJ+LtTpmCB+u8z0EW5/t4aufhhE7E7mKWk7h2ZHR2lORH6k+8ZAu3a/
GtMbcBCsxrVVMbyr7xF/B5PQqjZfaS16zCm6dDhV47vpxwy/1K7uHjk0SmFUaO/cLVLG+pdjnGWhchLRPinzEs6NfEkT
Xxq8O7ZhIlxgwWsyYDpY+WxH8UsM8sjTyr36AGKtOpSmYnO5BlOpDx+EgdjgBbjkm3cb2BeKYYQXdrP7WbQPh9yn
ohBGkaB43qZXomSQK4bUZeuzu1uo5Iz8nJhe0+wAl/EHdi8pmDDwD6S0vG44dn8zjhcRFaCUndEuJnPx7D8nV7x4
Ai6K/nPpjBBhVjVZ9bRCTw+14ajGC7zL7yRnT3TPFgZr/A5BUBJaQG0VWaIPMkIoTt0VRyZ2xkHSp5wZaSVUaFM4O6
T3RHu6fjJbLBAeP2Q0pTYHHKFNl0CiRbEoSbN0Q/mC6btKA5NL8rGs6Efm9MyrppfdhbUYKdnXGMp3K7J6u1lTw7E7Jr
IrZEhOWRG+OdsxuWvC1pQQlqL8RfbDqHrITxh+R0oUFwWhEI/CtGJ4zP+OOk4Ft8/pA10nYyIEHn0U+V3UYhKv4BUIk
7QbgKaYEq7Eypj5xlZlDlb0HsR/QuKA5OLYPRaqVyqqxtJzKDMdZgDdYg1ohZTRBJd5G5CEtKfmTFQqzEMaPtJaJn
0XgvLTxZ3HfDWR3x0EdgUuxx67QRCCtwzhZWXc/KuIjyIiZG1uvIxJI0ltfoVGdwHfWdiAT2nqWwGUHlwvMcf4b2
xKSwPhgoW9NGiXH9xULRuE0++nM0JVxbhAlr2dh3CHhwlhVUVOzJgZx+yaZT1MBFEmR/iwbjUBl9iFkkXKZzq7Z3F
fd+z0aIM7LP2lgq3qkW47ahHhqPABUpbNnpT4w53i1q2ThJqxnnxuGrfYjcMcanE6zRY3adwsNQA79dLCFGlaAyD
pXGIzx8eU5ZdH4HDOL7xNaIXBnEV5vcE4vxBkKv0dHxxLpxvSzOE9Lmzh3ZH+9xWiMgJGKt9Mfl5YjaLkd8HE1Tp
AJPbdh+0MQiZG7ltY16IgI1Utz+3AAWAPjFhQbJGRjdxsMGPnKq/3QUnS2XzrVc9O0i6d8U+g1kyw+MUKFWmUN8
Wm0k3ryZ5/q80dbMxXz0WGgh1fhEhpP+c2WYuVYHkMemW+RFjqb9eJh8a8QDFXSZn8PnUcRrRzeTAf5Y8UwTH5Ur
Hk4+fzajjVUdlfxHgWJwMMzQI8Li0bcQFeEoOzRzRvvg4MLWxIm2aFcd3T05jpDhG7lDI+f4L2pr7UZvjBdt7Oxa
pVwp3fsYNhR6fKMDrKIBDhwRJwbrfNBRMb6DH/l/n7a4AwrOn6Ao6SNTVcu8B8mrk/M0FdsmKqVs2A9mm3oAGp
U1WIV3p6fWrMkSAbuAfH1Md7x1dDz1KYMSh3IM+Tt0EnBMGgxWSGbJBpwm9Ly52MAtR89v/QZcZPOF2f36tnRdrs
-----
```

```
HvtmcqdkbjmrtwQKxRLFvyi2xadbqdM1ana7wID8N65KgK3u11HkdWshiZgQoONhR9AQAAPXOYrFQwAIANGkut
OdLdufq8/VHNH97tmxORnQ2JSWVTs3ZcFRzNhxVfHigLu+RrFG9W+rM5EijowE4DoqeAAAAAACcomOZVs1ef1R
fbMxQXqnN6DhoADklNk1bdkDTlh1Q16QIXX12K13Zs6WiQwOMjgbgNyh6AgAAAABQC5UOl+ZuOKr3Vx7S9sxio
+PAQNszi/WPr7Zr6rc7dVHXBI3p31rntm9hdCwAougJAAAAAMBJOVxQrg9XHdIn6w6rqJy1qvEru8utb7Zk6ps
tmeqcGK5x56bqyl4tFeTP8j6AUSh6AgAAAABwAsv35mn6ioNasCNbbjYlwh/YmVWiR2dvOfPzduq6fq1144A2S
ooMNjoWOOxQ9AQAAAAA4HcqHS79L+2IZqw4qDO5pUbHgQ8qLHfo7UX79J8l+zX8zETdPDBVZ6fGGBOLaDYoegI
AAAAA8LOSSoc+WHVI/112QHmldqPjoAlwuj3eqe+9WOfpnqEdNbhTvNGxgCaPomct7PaPNToCGlA3owMAAAAAa
DCFZXb9d/kBzVhxUMWVTqPjoIlan16km99bqx6tInXPkI4a1jXB6EhAkOXREwAAAADQbBWU2fWfpfv1/oqDKrO
7jI6DZmLTEatufX+durWMON2DO2r4mQkymUxGxwKaFIqeAAAAAIBmp6jcrncW79cHKyl2wjhbjxbrg/T1DkxX
PcM6ahLuydS/ATqCEVPAAAAAECzYXO6NGPFQb3x016msaPR2JlVoomz1qtrUoT+NrKLBnZoYXQkwOdR9AQAAAA
ANHkej0dfbsrQP7/fpSOFFUbHAWqOPbNYY95drcGd4vR/l3ZRx4RwoyMBPouiJwAAAACgSVu9P1/PfrtDm45Yj
Y4CnJSFu3K1ZE+eruubovsvOkMtwgKNjgT4HIqeAAAAAIAmaW9OqZ77bqd+3JFtdBSg1lxuj2auTtcXGzMO4cL
2Gn9eWwX5W4yOBfgMc2O6p6amymQyVXtMnDhRklRZWamJEycqNjZWYWFhGj16tLKzq765pKena+TIkQoJCVF8f
LweeughOZ2sowIAAAAAqBvldqee+Wa7Rry6hIInfF6pzal/fr9LQ15cpG82ZxodB/AZtSp6rl27VpmZmd7H/Pn
zJUlXX321JOn+++/XV199pc8++0Oy+LFy9WRkaGRo0a5T3f5XJp5MiRstvtWrFihWbMmKHp06frySefrMOnBAAAA
ABoruZvz9ZFLy/Rf5YekNPtMToOUGcyrJWa0Gu9bn5vjQ4XlBsdB2jOTB6P55TfBSZNmTvv/5ae/bsUXFxseL
i4jRr1iz9+c9/liTt3LlTXbpO0cqVK9W/f3999913+uSSS5SRkaGEhARJ0ttvv61HHHlEubm5CggIqKn6K9W/f3999913uuyyy5SRkaGEhARJOjvvvKNHHHnlEubm5CggIOKn7FhcXK
zIyUlarVREREacav9Zm7+IvKs3JqE5JRkcAAADHYdTnQQCNVOZRhf7+5TbN387ITjR9wf4W3Teso249r638LLU
azwYOG6f8f4bdbteHH36oW265RSaTSWlpaXI4HBo2bJi3T+fOndW6dWutXLlSkrRy5Up1797dW/CUpOHDh6u4u
Fjbtm077r1sNpuKi4urPAAAAAAAcLk9enfpfg17eTEFTzQbFQ6Xvt2cqa83Z+iznCaHQdolE656Dl37lwVFRV
p3LhxkqSsrCwFBAQoKiqqSr+EhARlZWV5+/y64PlzfPHx8x7lo7X0Vt/sy3dkh
```

```
p3LhxkqSsrCwFBAQoKiqqSr+EhARlZWV5+/y24PlL+y9txzN16lRFRkZ6HykpKacaGwAAAADQRGw8XKTLX1+mp
7/ZoXK7y+g4QIPbmVWiP7+9Qo/P3aLiSofRcYBG5ZSLntOmTdM1l1yi5OTkusxTo8cee0xWq9X7OHz4cL3fEwA
AAADQODlcbv3z+50a/dNa/dZybc9kJiCaN7dH+nt09Qdt9tFgLd+UYHQdoNPxO5aRDhw7pxx9X/1OzZs73HEhMTZbfbV
VRUVGWOZ3Z2thLte7191qxZU+v/8ZU+Vav+zu/kufmgQGBiowMPBUogIAAAAAmpDd2SWa9PEGbc8sMToKOKjklNh083t
rNbZ/G/3fpVOUHGAxOhJgqFMa6fnee+8Pj5eI0e09B7r06eP/P39tWDBAu+xXbt2KT09XSCZm6SXuTdBAu+xXbt2KT09XQMGDJAkDRgwQFu2b
FFOzq9/eZg/f74iILUtWvXU3OAAAAAIAmzuM5tnbnZa8tpeAJnMAHqw5p5OtLtflIkdFRAEPVeqSn2+3We++9p5tuukl+fr8MFI2MjNStt99+6++/X1FRUbvvvV9XvD9XsWbv1tVVyVefKeAOlBBnpPFvQhgm+VPXUxKPX+ixSxcCAAADpxbbd6u7DvNIKoVmVTSOvgZi
9p5tuukl+fr+eHhkZqfHjx+uBBx5QTEyMIiIidM8992jAgHq37+/JOniiy9W165dNXbsWL3wwgvKysrS448/r
okTJzKSEwAAAABQo6NFFbr/o/Vac6jI6CiAT9ifW6ZUerYKT5evWq3v6AO3v9auVUxZK8IT7AP0lYTUZHAhpcr9RiQ6o9LT03XLLbdd
Ua3vllVdkNps1evo2Ww2DR8+r+XG+99aRC30R4uv2oKm2ZUdk8MYbNjnqpEUSZ6f6Lcxx8KjtIxJ5oDc7A1dKrmrrSYu2lOtoeqfbBUxR4ThUv9EpErcdl
pNzt6rc4TY6CuBTnG6PXp6/Wwt5qTUWoOZGABmXyeDweoOPUVnFxsSIjI2W1hUhUREdFg9529K7PB7gX
jjeqUZHQEAABwHEZ9HgTQcCrsLj3yv436cN6cnOW0VEAnxcSYNHHT6tjI4CNJhT3r0dAAAAAID6sD+3VJe8s
pCCJ1BHyu0uPfgPpJv1tzhbZnYRyaRvNA0RMAAAAAOGh8ueGwLn1ly5ob4VXPMsoAGmc7sKkurZbd5ApFfb3O5I9lUx3HdoIh55pPfu3SlEc+6R3gH7E9evOqPhoUYXRUYB6RU9E
TAAAAAGA4p8uthz9eq3s/2aqzShZv/i4DDDj/U1UNfU9i6sTDEbKuAK0XrCEqaKpSh/+bvr9v7xA8CAqgAAAAAAyVWVUS1/6UZ9uzDE9pf2+K9Wo5/uYGy2xi+7ufEGjhCExgAPP8+4/LtSdDPXjlrO/36eR9rO/39Sh1/6G5+Fwh
JY7NO69NXptwR754FYvwEmh6AkAAAAAMMySHRm66MWftLvAYXQUoFlxe6SX5+/Wbe+ny1pB5SjYQTRsY7JxdtrGm4eixUhU9Af4XyXuexNGkAAAAAuvmkR5hvwFEZ/3F9sdARCk7/z1QlzLCz9YyXvPW4qhyLnHLedNltVjc5BRnVIAmcyxlpNdTa+XXv63mZAAAAAeqn4uAr7vkgrSCT/6eHS4ZnvaWm7HdL8Cv4WE2qt1+n0oCJC
M8e5P2zRuxnqVOk1GRwGarYW7cnXV28t1KL/M6ChAnaLoCQAAAABoUB6PRw9+sFRP/3BQblHwBIy2P7dMV721Qsv25BsdBahdUuhhJ6wHCtU4ExvQbMudDPUEeyNK
mmHCoyOAtQZip4AAAAAgAZTUlapUS9/vBFRsdBcBvFJTZdf1/VugUaZWAJgqUPIjOFLKyJCkrVqS6pqNBowvi1UOpqFzeGDVLoCJnAJT3dlgKgGzYYFAbKxRJiqm64zMWgGNuk7MUOBjbWI1onz7qwlNQwVKg47+3BgYYmIKL0VJX4oz91QwZk8vm1qY7yrY4IJPmjm7njdWSyUbxOFQFNuUQF/VEMDBzO/TeB5eqG6Ck1fQrXhbGQ9P4YEmBDh2zrPVOKn02r8Qrq9TxVdnmdg2xcQrq9gFO86cRt7p8ZsOsRvTp6ptTnxkq8WlH+6TEy4TEDphh6sTsH9UGM+xedkN+D6a7u8ofiK0ZsqubYsa9VIEwY3SvjNXyWDtoRWGNWuNxwX7RaKDkL8Oeth6c0YNjD29dvtfKR0W7cD4lUp8ktEwENK1sCWcUdkKvBCjPyHrK8IDZKNvnSLeeffq9+k6hR4SzJv8EJp+BSWsOHm3dtlgVxIbhl0LDWY2+MqKvE8pSiWUhLfxGkwNSjxjWY7/vNs4W+DTZTMqlsmmW9lhjyQgvWZG6jyZDw06mfgBnx8F8g+cN5o0xWqO+emo5IaTBf5AJmjYQZwxMSsKQyr5nPSHl1qMjjSfhZuPnBiJTBwAkAIxnRnAuAgDcTLYJs1wvPYTyOSzXiVMVeeyZLe0yV6Wn76hPPoQ+0HTSm4HQygqrrJBpeVSpVtdRX6ZW6z/4iPBFSbVmMFXxAzBD0MaK0oSMKjWgGWUQ+1vKgxSXyYS4N5oQbVHyfWVPAeeGRZ9Okyp1dQV7ycSbrc6++9k+eDVs0yPMvC2w2K3WT7wLhkgvBycPHVBvC1lNbPr5V48PU+wLnoZ9b8mFmv0xdF0e2SuqUeHN5Qcm1CUKj2HXmDKkFvl0QrfG2PjC9S4CVxRfKR2jQ8U0UKLdDY7eYxwcVvfTDFBq0E4vE45xXRhKF3Kb4ZJFrZ07uYsVwTCXsMxY/i81xYVYJqRMyMhPmTGBFnKsp7xWYZ4VK6sjQvVvOUHa6lZWmdqdgAdH7rd5KffPzBWwAXScL1+nEMydJPXu9b2Jq/jNBVldhcfDqZjIb/f7jqTtnxpOZx3HpGUPZ1vqn4/o9ZTAmC2hGbq1l7qBmmyY1Sm
```

pmrFUcqdSSgtdFxAPiwEptTt0xfq5ySSq0joJnxMzoAAAAAAKDxsNnsev/zr7Vk+xEdCOttdBzUQtGymbIu/6j
KMb+YVmp52zuSpJKN81S2fZHs2fvksVco5b6PZQ4K061rSlLWrEdl07y1Sp+wniMU0/xuSZKrokT537ysyvQt8
ot0VotL71NAQntv3/wf3pZ/VIIi+o2q/ZOGTzhaVKHbZqzTp3c0UKCfxeg4aCYoegIAAAAAJElut1v/+3a+flq
zWQcj+srtMRkdCbXk36K1Eq595tcD5l8neHocNgW366Pgdn1UtHhGnVzzF2E9hivqvBu8X5v8A73/bl35idz2C
iWN+38q2fCt8ue9rqSbXpUk2Y7ulD1zl2KG3X7SeeCbNh2xavKX2zV1VHejo6CZoOgJAAAAAJAkfb94pb5buFx
Honqoz0VvdBycCrNFlrDoGpsi+l4hSapM31xn1/yFyS/wuH0c+YcV2uUC+ce0VHiPESrdNE+S5HE5lf/Dm4odc
a9MZkb/NQcfrUlXnzbR+n0fVkZHQTNA0RMAAAAAoLWbtmn2dz8qK7S9sl0nnvKMxstZmKEjb94ok8VfAS07K3r
QTfKLiK/3a5ZtX6Sy7YtkCY1ScId+ijz3Opn9gyRJAfFtVXlos8J6DFfFgfXyj0uVJBWv/lxBKd0VmNTxtPLBt
zw+d4v0TI5Ql6QIo6OgiWMjIwAAAABo5vYf0qIPZ3+tbHe49noSjI6DUxSY1Emxl96v+Kv/oZiL75KrKFtZMx+
R21Zer9cM7XqhWlz2oBKuf1YR/a9W2daFyvv6JW97ZP+rJbNFR/91q8r3rFTsJffJUXBUpVsXKHLgdcr//g0df
We8cuc+J7et7LReAzR+lQ63JnyYpuJKh9FR0MQx0hMAAAAAmrGi4hK9//lX0lJYoZ2hXSXW8fRZwe3P/vWL+LY
KT06kI2/forKdyxTe4+J6u2Z4zxHeLgFxqbKExSjn47/JUZgp/+gkmQNDFfenh6pcN+uj/1P04FtUtm2RnEXZS
r7tX8qf97qKln+kmCG3nlJW+I6D+eV68NNN+vfYPjKZ+JmD+sFITwAAAABoppx0pz7+4jtt23NQByJ6y07hV8S
mxBwUJv+YlnIWZTToNQ0TOkk6Ni2+JqWb58scFKqQjv1VeXiLQjr2l8nip5D058mWvqX0sqJxm789W+8s3m90D
DRhvKMBAAAAQDM1b9FyLVmzXoVxPVTgDDA6DuqY214hZ1GmLKExDXpNe86xQpYlrHofV7lVRSs+Vsyw036+oFs
et/PnRqc8HnedZUXj9+IPu7R6f77RMdBEUfQEAAAAgGZo47Zd+uKHRXJEtNJee6TRcVAHCn+apsr0LXJas1V5Z
IdyZz8jmcwK7TpIkuQqLZQ9e78chZmSJHvuQdmz98tVUeK9RvbH/6fitK90+pq0wkwVLf9Itqy9clqzVb5ntfK
/eVmBKd0UEN+2WsaCBf9WRN8r5RfeQpIU2KqLyrYtlCPvsEo2zVNgy6719vqg8XG5PXrg000qYX1P1APW9AQAA
ACAZiYzJ0+z5n6rUoe01a+10XFQR5wlecr76p9yVRTLEhypwFZdlTj2JVlCjhW1SzZ+K+vyj7z9s2c9KkmKvXS
SwroPkyQ5CrMUWFF80tc0WfxUeWiTStZ9KbejUn4RLRRyxrmKPPe6avkq9qfJWZipFpc96D0W3vsy2TP3KvODB
xSYdIaiBl5f9y8MGrWjRRX6+xfb9PK1PY20gibG5PF4PEaHqK3i4mJFRkbKarUqIiKiwe47e1dmg90LxhvVKcn
oCAAA4DiM+jwINAUVlZV6/b2PtX7LDmXE9VW6LdjoSACgt8b01qXd+T0cdYfp7QAAAADQTHg8Hs3+boHStmyXJ
6kLBU8Ajcbf5mxRTkml0THQhNS66Hn06FHdcMMNio2NVXBwsLp3765169Z52z0ej5588knFx8crODhYAAAADQTHg8Hs3+boHStmyXJ
9e6pco6CgQGPGjFFERISioqI0fvx4lZaWnv6zAQAAAAAc19pN2zR/6SpFxrfU+vJYo+MAgFdhuUMP/2+z0THQh
NSq6FlYWKiBAwfK399eff3333333nbZv366XXnpJOdHR3j4vvPCCXnvtNb3j3zzzTvtNb3zzjtavXq1QkNDNXz4cFVW/lqtHzNmjLZ
t26b58+fr66+/1pIlS3T77bbfX3bMCAAAAAFSRm1+o/30zX5JJG12t5fAw8Q9A47JoV64+WHXI6BhoImq1puejj
z6q5cuXa+nSpTW2zweJScn68EH9Rf//pXSZLVValVCQoKoKmT5+u6667Tjt27FDXrl121du1anX322ZKkefPm6dJ
LL9WRIOeUnJz8hzlMNgTU9AQBovFjTE6gdl8uldz78n5auTpM9qafW17Fb04DGKdjfom/uPU/t4sKMjgIfV
6s/7X355Zc6++yqdfXXTVVys+PlJ4Cl
Jw4YNk9ls1urVq0/YlZ/3+QAAAAAfmfxqjStXLdRscmp2lTLSGfUMEWDFfbUHrwJAY3bXrs5+hV59RnTbT7lvr0vnadPkLw6HX+GMk/BGSyTI96m/rUxbVasbWIoRQQEHSLXYqHXUbPWjKj6+Srufn59iYk6Zr4uKnZ35lw4hMoK8fX7/VX1/rJmMpmM6YWkmkG9uhMyFCMAAAA/Lr+wcbVZsSGMjZEZsJbPPXt6wfAgRXkuRlyXL
+AQkPw6OKfmrMScQKamqOPvqSZV5GdhVBbwoY+rCcj
...

tfozyBNAsVThc+s/SA0bHQCNF0RMAAAAAfMzileuUm1egxPgW2l0eolJGeQJopj5YeVBF5ezkjuooegIAAACAD
8n0y9fi1Wlq0SJaMpkZ5QmgWSuzuzRtGaM9UR1FTwAAAADwIUtXr1defqHiY200tyJYJYzyBNDMTV9xUMWVDqN
joJGh6AkAAAAAPiInv0CLV6WpRWyMTIzyBABJUkmlU90XHzQ6BhoZip4AAAAA4COWrl6vnLwCJbSI0RFboKx0f
6MjAUCj8P7Kg7I5XUbHQCNC0RMAAAAAfEBufqEWr1qnFjFRMpvN2lYWanQkAGg08krt+mJjhtEx0IhQ9AQAAAA
AH7BOzXpl5xYoIS5WRU6LjtgCjY4EAI3Ke0xxx29Q9AQAAACARi6/0KpFK9cq9pdRnqVhkkxGxwKARmVHZrFW7
Msz0gYaCYqeAAAAANDIrdm4Rdl5BUqMi5XdbdLeimCjIwFAo/T+ikNGR0AjQdETAAAAABqxispKLV6VpoiwUJn
NZu0qD5HDw69yAFCT+TuylWmtMDoGGgHeKQEAAACgEdu0fbc0Z2QpKT50Ho+0nQ2MAOC4XG6PZq10NzoGGgGKn
gAAAADQSLndbi1ZvV5mi1kBAf5KtwWqx0VndCwAaNQ+WnNYdqfb6BgwGEVPAAAAAGikdu8/pB179ys5Pu7Y1+U
hBicCgMYvr9SmH3dkGx0DBqPoCQAAAACN1Ip1G1VZaVN4WKgqXWYdrgwy0hIA+ITZ648YHQEGo+gJAAAAAI1QV
m6+1m7ervgWsZKkvRXBcstkcCoA8A2LduUqv9RmdAwYiKInAAAAADRCazZsUWFRsVrEREmS9lQEGxsIAHyI0+3
RFxszjI4BA1H0BAAAAIBGptJm07K1GxQRHiaTyaQCh5/yHQFGxwIAn/I5U9ybNYqeAAAAANDIbN+zXxnZuUqIi
5HEBkYAcCq2ZRRrZ1ax0TFgEIqeAAAAANDIpG3eLo/Ho8CAALk90j6mtgPAKZm9/qjREWAQip4AAAAA0IjkF1q
1cdsutYiJliQdsQWqwm0x0BUA+KY5G47K5fYYHQMGo0gJAAAAAI3I5h27VVBkVWx0pCRGeQLA6cgtsWnZ3jyjY
8AAFD0BAAAAoJFwu91atX6zAgICZDab5fJI6ZVBRscCAJ82b2uW0RFgAIqeAAAAANBIHDycoX2HDishLlaSdNQ
WKIeHX9sA4HT8uCNbHg9T3Jsb3j0BAAAAoJHYvG03SssqFB56bLf2A0xtB4DTlltiO/r0IqNjoIFR9AQAAACAR
sBms2vl+s2KjAiTyWSSm6ntAFBnftjOFPfmhqInAAAAADQCew8dVmZ0ruJij+3anmUPkI2p7QBQJ37Ylm10BDQ
w3kEBAAAAoBHYvnufHA6XggIDJTHKEwDq0oG8Mu3JLjE6BhoQRU8AAAAAMJjd7lDalh2KCA/1HjtE0RMA6tQP2
xnt2ZxQ9AQAAAAAg+07dFiZ0XlqERMlSSp0+KnE5WdsKABoYn7YxrqezQlFTwAAAAAw2K59B2Wz2xUcdGx051F
boMGJAKDp2XzUqrxSm9Ex0EBqVfScPHmyTCZTlUfnzp297ZWVlZo4caaJiY2MVFham0aNHKzu76tDh9PR0jRw5U
iEhIYqPj9dDDz0kp9NZN88GAAAAAHyMy+VS2pYdCgsN8R7Lo0gJAHX045FW7ss30gYaSK1Hep555pnKzMz0PpY
tW+Ztu//++/XVV1/ps88+0+LFi5WRkaFRo0ZZ52l0ul0aOHCm73a4VK1ZoxowZmj59sm6eTYAAAAA4GM0Z
2QrIztXsVGRkiS3R8q0BxicCgCaphUUPZuNWi8S4+fnp8ExGrHrVarpk2bplmzZmnIkCGSpPfee09dunTRql
r1L9/f/3www/avn27fvzxqmn9Mgjj2jy5MkKCOCNHQAAAEDzsmv/QZVVVCg5SqVOvwl8PDS
mQAUB9W7ssz0gIaSK3fSffs2aPk5GS1a9d0Y8aMUVX6uiQpLS1Nr8Y8Y8KTJNBpB9UrdgeV9OqSVq5cqe7
duyshIcHbZ/Zxhbn84L2/tc2bdt097kAAAAAgM/ZvG03ggICZDKZJDG1HQDq08H8cmUUVRgdAw2gVkXn5R9n
9e8efPO9ttv68CBAzr//BUQ6fXPFgW7Q5CQkJyso6tjtWVlZXcQCKQwYAEAAAAgMoIHd/8nTWqU
9e8efP09ttv68CBAzr//PNVUlKirKwsBQQEKCoqqsqYhIQEZWVlSZKysrKqFKWOth9tO5Ennnik2
A+Lqi4hIdPJyhqMhw7zGKngBQv5bvZbRnc1Cr6e2XXHKJ99/v/PNVUlKirKwsBQQEKCoqqsqYhIQEZWVlSZK
Mc/6u36AAAAAGCEA4ePylpSqvapKZIkp0fKYT1PAKhXK/f16+qzU4yOgXp2WgvTVjBL/0Xd7Q9vw9dRfFjK2
uoqKiKn2ys709a4AmJiZW28391l69rWif0F4899pisVqv3cfjw4a4JDQAAAACNwsHDR+VyueXVRdvp7g5iQQ5bS
E4FAE3byv1sZtQcnFbRs7S0VPpL690kjf39/LVViwwNu+a9cupaena8CAAZKkAQMGaMuWLcrJyfH2WbBggSIiItS
/viIiItS1a9fj3icwMFARERFVHgAAAAC0bzVej7bs3KuQ4CBaZh3nuh7P1wOfnk3MIPJXk47Y1fMX0M0tf/6rFi
xfr4MGDWrFiha666ipZLRbdcNNNio0bj6gVHjpF/3YXvf8Ll5H/ru0PD8VGsxt7dZ72x8Gg3JBkDHQ4p3ukF/+
p0yZ9//33+evzxxzVx4kQFBrJuDQAAAIDmIye/QBnZeoVlsmZbq9uf61zfmWimWz70zYtyqlFXZN46iSpZ90G
HzFxcXpvPPO6vqxQXFydJeuWVx2Q2mzvMzH7U6vDYbDNHz5cb73lvd8i8Wir7/+WhMmTNCAAMUGhq4qm266SV0
mTKnbzWUAAAAAjdyhwxxkqLi1VUvyx36c8HimPk4A0CA2HynSn/u0Mjo33oXlqFO3
zxunzZt2ujbb7+tzWOBAAAAoMnZd+iI5JEskMT8IpdFtk8p7UCGQDgJG0+YjU6AuoZ76gAAAAAOMBcLpe27Nq
jsNBQ77FcRnkCQIPZnlksh8Q08x1vMEgIZjd7q1K6vE6BioRxQ9AQAAA
KCBZebkqaysXKEhiasXKEhIdhd5jjPQEgIbFFPemjaInAAAAADSwjOwcufXrep5uj5TvrNWWCwCA07T5SJ
AQAPbf+iIAvx/LXIWOP3kYhMjAGhQjPRs2nhXBQAAAIAGZLPZdfDwUYWHhXqP5VXIVTN3kvjh2VSavEw9tNmcbB9IAGtzu7RJU0l9ExUE8oe
gIAAABAA8rKzVNxabZbnCwn5dz70Iqe0A00Ccbo805pcZHQP1hKInAAAAADSgzJw+lVdWKjAIqiddQ6HuMoicAGGNfDkX
PpoqiJwAAAAA0oKNZOTJJMvnbrcDgCH255YaHQH1hKInAAAAADSgPQfSFR6P3a7ZGKXRYDEwFA87U/j
5GeTRVFTwAAAABoIDabXZk5uQ+N+XVqu9XpJ49MJzgLAFBfG0nZdFH0BAAAAIAGkl9kVXlFpUKg7zHrKznCQC
G2Z/LSM+miqInAAAAADSQ/MIiVVRUKvg3RU82MQIA45TYnMopqTQ6BuoZRU8AAAAAaCD5BUXyeDxsKggOYV
gBgLEZ7Nk0UPQEAAACggeQVFFV2fQx8LEAAAAgAZyODNbAQEBVY6VUfQEAEN1FlUYHQH1gKI
nAAAAADQAt9utjOwchQQFVjleTtETAAyVXWwz0gLqAUVPAAAAAGgAhdYSlZaWV9nEqNJtkksmA1MBALLZyKhJo
ugJAAAAAA0gv7BI5ZU2hfym6MnUdgAwHiM9myaKngAAAADQAIqKS0gS+Rw0BTg7+89xtR2ADBeTjEjPZsiip4AAAA
A0ABKy8olSb9PZGekJAMYrKLfL4XIL4XIbHQN1jKInAAAAADSA4tLSKuNyasfK3fxKBgBG83ikzCKmuDc1FD0BAAAAoAHkFhTKz2L1Ppeq
AOwFpdUGeUpMdITABqLbKa4NzkUPQEAAACgAeQWFCowIKDKsUpGegJAo5DDZkZNDu+wAAAAANAACgqtCjwr3L
MQdETABqFkkqH0RFQx3iHBQAAAIB6ZrPZVVpeUb3o6TEd5wwAQEMqszmNjoA6RtETAAAAAA0pZcWmZ7Ha7Av2rF

j3tFD0BoFEos7uMjoA6RtETAAAAA0pZaVm5bA4H09sBoJEqZaRnk8M7LAAAAADUswqbTQ6HU/5+flWOM9ITABo
Hprc3PRQ9AQAAAKCe2ex2uVwuWSwW7zGPR3J6+JUMABoDRno2PbzDAgAAAEA9s9sdkskkk+nXkZ2M8gSAxoORn
kOPRU8AAAAAqGc2u12/L3GyczsANB5lNjYyamooegIAAABAPbPZHdWOuSh6AkCjUWZnpGdTQ9ETAAAAA0qZzW4
3OgIA4ARcbo/REVDHKHoCAAAAQD2z2x3y8Ps0ADRaFD2bHoqeAAAAAFDPKiorZfrdbHZ+vQaAxoOaZ9NDORMAA
AAA6ll5RaUsFkvVg6zpCQCNhofh+E2On9EBAAAAAKCpszucMpurFjn59Rq+wCKXWpqLlGLJVOtzoZLMhUowFSl
OhYqVVTGy6g3/VJ29lqXUIwWilA9f4HA4FRYaoqDAAO8xS2qqpAsMy4S6R9ETAAAAA0qbxyNRDkIjEiiHUiz5a
mUuVEtLgZJMRUowFaqFrIpVkaJlVaTHqjBPmcxy/3qiR9Uq9umxsfr+ijwN3B+rq5e6lZyR26DPBTglFaVVvpU
t4WGGRUH9oOgJAAAAAPWMaZNoKOGmcrU2F6qVpUDJ5kIlmo6NzGyhIsXIqmhZFeGxKtRTXvXEGoqZJ+uXcv7yd
vla3k66YE+irl7mUEJW/uk8FaBBmcysANnUUPQEAAAAgHpWOzhPyqCojThTiXeKebK5QAmmIsWbitTCVKQYj1V
RsirCXawg2aqeeBrFzJNl/t36tEs65mlJR2nwrkSNXuZQfA7FT/gAip5NDkVPAAAAAKhnNdWcLCbKns2dWS4lm
61KMRccm2L+83qZ8SryTjGP8hQr3FMsfzmrntwAxcyTVWG1SkHVjy/slKeFnaTzVodq9CqnWpbbqncCGgsLRc+
mhqInAAAAANQ3j0cyVR0N50fRs8kKkEMplgLvepm/TDGPU5Fif55ifmy9zFJZ/mC9TF9gdthP2L7sHJuW9fXow
mUBGrXWpUS7q4GSASfP50dvdATUsdMqej733HN67LHHdN999+nVV1+VJFVWVurBBx/Uxx9/LJvNpuHDh+utt95
SQkKC97z09HRNmDBBCxcuVFhYmG666SZNnTpVfn7UYAEAAAA0PTWt6elP0dPnhJkq1NpccML1MiM9xQrxlMv02
+qljxYzT1ZkRLikkhN3Mpu06AK3Fp0njd4cr5EryhVmLW2QfMDJMIeFGh0BdeyUq4xr167Vv/71L51lllVjt9
///365ptv9NlnnykyMlJ33323Ro0apeXLl0uSXC6XRo4cqcTERK1YsUKZmZm68cYb5e/vr2efffbOng0AAAAAN
EIej6famp4UPRuPWF0JWlsK1Mp8bIp5oqlI8T/vZB6jIkWp+Of1MiurntjEi5knq7ioQIo4yVFyZpM+71mgOWd
JV2+M14iVZQotLqvfgMBJsISFGx0BdeyUip6lpaUaM2aM/v0f/+jpp5/2HrdarZo2bZZpmzZqlIUOGSJLee+89d
enSRatWrVL//v31ww8/aPv27frxxx+kVkJCgnj176qmnntIjjzyiyZMnKyAgoG6eGQAAAAA0YiaTZDG55fKwjlz
9cCvJbFVrc4FaWgqVZCpUgrlQ8SpSC1kVrSJFeayK8JTIX46qp1LMrBW/kCBJtZuy7jZLn/Qu0Gc9pOvXx+uiV
aUKKS3/4xOBemIODzM6AurYKRU9JO6cqJEjR2R2rYsGFVip5GFVip5paWlyOBwaNwaNmyY91jnzp3VunrrVy5fKlSv
VvXv3KtPdhw8frvHjx2vbtm3q3bt3teex2Wyy2WzefxcXF9fH2wEAAACABuFyS+6RtQhu3KzCfgTZ/8XXLHexwaAuTr
tPdhw8frgkTJmjbtm3ql8uWvXtLsvDZbL8ueFxcXH0ndAAeFbzQqqWqRj+V+a0a+6dq8cCKcVGb6dhCU0tPr8yt6c+fr
GRW0Z7Z2dlKTEyUJCUmjmrNm
jVVrpudne1vq0lgYKACAwOrtUdFRZ0z2u71V9ymgGBbi4/VgOQGvOMbBgYAIAB8snUybgmTM7RxQaPSBTfsbs
W1/xyEaX7jE6QG15fcOrugrXPGBcCgm7t/qJa4tS3KU8FcQPyMTTqRn4NfGrBuzsLBF9LgGlEXvnkiZ/Zp3Amb9PZXrO0dYFFx8A6G+e7Vov+SCz7
EBT3s1ccbQe87+cBzUz+5s+SQjeLpMXaWFmt0Q9ewBfOOHkpbhN9IU7mwfwQJvjfA26Aq/Ly7bmfeLPaKDrmFd7BbVCnQjsPsQwFkIP+/3n/9lf
h4xJ9L+RFYW2m9xdgjdPfYnfk+GlmhpnY3q++3Bi0s9V7HUfeUL01V6JTt6a/Ijp0rm7wG5H/6fmFk1S6RsuEKkw75yL0j70v/5zXt4lfL5v/kfdk
mjwO04MixChZfAcQWPqIMGJfnY1SDlKYm9iRpBbq5JIJTV5QWYOdk8J/4o8OuJ0fHgwBeESAO+HNfIlGOaJ+RCBlzRHK4WL+ND1ZkWcmJvA66Sk
IoC1P5XRZzMdVPWa3tK5wBhJzj5J+kVGvd/zbakpl8HM5nz8/t58C4hrk7pnHkv8ZgSw9ANgn7JLJJY5/w+MHWAWpS7N79x8S7jzhMeWF3B1t0BP
MdsVJfoYA3yv1HDKaSLvHwUd7hrONHaO/7zsQC4dXwNaYlbTznBVbncpDTzYXZPBJcz8LtAN+XYeYqyjBTi5gHwtn+eaPTTdE7X3b7b3Kr+YLjD
YnuA6MTCVyqkk8iZR8HrSp6rHN4ymKSUXrGufXHhHHYEk0CrEqN5uq55hE3YIDgEVPXmTn2ZlRO+iY5pXqPZmKk9QkQeUWtnsQK3w1a5vRJIgxPGj
d6jPqVD5o5cG4JBKphMRt+Gb6/sfkhJ+jvnX0EsMZ60q0JkXObEkM85W1zGPPzLdzl3uyqOkt3a8E6bWXpw8S6nYTlf7j72eKuvBVPHsgH4Ap
UUl4/o8eBPJ4TiM5Q5Ru4KutE1bNvHHX3OjTj6mv7Qjk/CH2mD8Mpbc0l/GqK+01DkZ0HD31e/eW+L4rZPHa3ibc2kPcUo9Fr9JWwC6+ubp
ZKaoKF0Kxgl/6ZEZkU1Rhot5VfUQkX43HrNz/6kSp1CzuJd/Fn3AZ4jCgGMHoIVVrpozoqRWm4pXc6QTfjFN4PGZeWIWw9EXjlMrzsuHU3/7
O0cWUhdEWRKqTG6s8W8v/1ZQbjVhIhoBTOSsjvtTk9zRMdF7QAjqdpbUMMhHFAnI8w3ZG6Oui/CM0Gsxbcl/ozqPs4S4R3P8RAHsAUm9cKvK
3/J5jgAvd1UVkKBAMtRL0WJGWbaJ3IH+l5c9NP9l1ftgWrXBlj5Rg9TPgSTUXijjbTqVdw8+jhyLL1wD9fYPuEFMGqwMPcyMu4VWWjZ8Q+tOUT
+h+vd1WP7ETYdfPeNKFLnM/UU/uBj0Oq/dvKG6FsqYgJodyQnNrBDTJi6BLXmFTkF7uDn/yx3xpS8Pz0UXO2Z3wKgmq3WDDK7pq4zNDNQnNu
Bp8fOfAaiVjWRAlnjRJ2DAsY6j0g+Rp9jcDq5fNJaJ6HLcNpPZ5E2Bu6VmRjQTpknN3E5Ih62BzpmDe8YTKqf3ATnbiqx9ReA2oUKP4TZhO
qa/3AFxTHWjP+zg6YoZ5dQNo3fgvj7e2BEDhYZeVZ2y2kiJmPY3+fL0o5YnSWdA8/ms63N4pjmc6P/gVBLY0IIcZmHa+EsFh9qh7Sx1rX3
HXfcQmd47MWQFGXFRt4+9e+vDmoX5R+Xp6B/LbsZZ+ZJN1fWyb6gPrg/g9v5lT7LQ9nPGh9gZZv3c0b3j7a98/p55t5h1/hP4V3eoP/+PPNXfc
PpTs7MUXbLvYq/DeAeY0HyWUbNX6Z2NvbDeDUEN6qmr2/0+YFm+cmx09JK8ybbDeeBAqUCZp0KGc+YZ2Ssg+YhmL4cWnuX9Ex0p19S7HvbKxa
9edYFMTqxTb//h+qfwh6j5b2bPfvw+yjxb7OefNlXqzP7RUKZrqJ4FKE6Bsf6SyX1WoqmrzD/b+QwR+QNjvNkwcd4N3eHWdMmvnJZNrlsqK
8w/cPxo5+TNgqw16UtKchNyx7wSNl7XmVNGPhqLhJWL5l+rB3U3VzqtOXSp+XZGwc0ggPCoNWMuYVxVh5k7qvAxZzc2UBW7ps4WYhIAF4mAm
99M6zp/k3nqE9aeD7i4cXZVfkd0lvs6evSXq0D7/OJJgeeR9Wh+6zD2C/ZjdpD8fLO+Wc7ucgjzDp91x7vPwPbDmwwf3ouhM+nGqC/XPSh+
8MxEtp3qPPXNC0z3p4HldN2qW+SbVrWd/0Tb1zN9L0Fmh17jcn57JnN92fzbNuZ6XUjaquZwXzu5Xdr/93mmlX1eRq9WjG6ebvCvzZPW/R
0G+3mKkqqrCt8u8t6Ex4zJc/mevNbUYZbjNG+b5J4ZQt4BbUvTn9jqXwY1Nz8PcE6vXWT0P1oOZQXvDLQQqjL4Uy5f+LLNXGAr8S9Hw2iG5
QURqHYpM3Zv6vMxBUzZQl6W/M7cJyhTkdJ1eoTkHmTtmVmh5W8RZMTdWXNtg5tFFWnwHyzt/B7ELA0l7dMvLJ+mTlI8ACx8ONlB82dVHHgf
Oe+fkLmnHWk2FvAw6hLmCu9GVDcYSf04LLpV0k/mVk3sD7bAw3Rp8zDzlBQWvmhcmLy6nlALm0S1ijdpTdB+6QGZjOeAHr7MLAGSNbU3OX1
8GfZt1PK5jfNb6dX6X8TQh5UlwpUvdTJMZ58fjvKYU+Q/5AYOGUGRpSh7cc3pHh6AbuP6sB5cXSn2Yb13RzcbZFFuzDXzIz9DqiUgO04v6F
3TdexSIutVwXpQrDhWnjl9vG3TIoI7o7M80LAB9KTn0kA9fnXCe3zu+b/O+Mwtn8OAhpjVD1ffjV49vTM1mqSe0KspkYF8wS83XTZwlk4
FG8T0x4fWsmrpNs5kxRtmmN0uVqi8QNbAmeGvtQcMWf9XKEdZdrxgXPR+/6zDEnd1TMPiUb0X9hF7NNmztQUzWFbmdN0a0GNV0t3AHsN6

```
MpSZfUanU80J1y5JgnbErt1EWaX8rxq9qqebdgnyl+Purb3DIcc9xeTx60CNTd8e2UFpFuYp/U+A95HBoU2Wlv
khtq46Bx/7o8veEBF2wr0TfFhfrz1FR2m+zq19IiLoFBatbULCey8nWEYdD0X5+ejE3V9dFRSv50GsTNwX+SYz
0bIoa888mnKS3nnhQm1Ys0b3Pv66Xv1ygHgMH6R83X6v87ExJ0tx339S3H/xXd0x+TlM//VpBwSF66ta/yG6rP
O41Yx0SdMOD/6cXPp+nF/73nbr1H6jnJ96s9D27qvQbdvUYvbt0o/cx9qHHvW2fv/0aKsvK9M/Pv9eZ/c7V008
85G3bvTFNezZt0MibbqvjVwMAAABofEKCg2TO3qbKvctVvm+VTDnbFV5+WInmYhU6A//4AkAj1pimtx/PwdgKP
XlVgZ4YH6Y9Z8TJY/KF1JLd49FXxcUaFRkp0wkyv5Wfpxg/i0ZHRdVwjWMF0MDfnG82mRRgMml9RbkkqVNQoLZ
WVsrqcmlbZaUqPR61DghQWnm5dtgqdUN0dN0+sUbGP4mRnk0RIz19nK2yQqt++FaPvvmezuzbX5J07T1/1bqF8
/X9R+/r+vse1tfvv6s/33mf+g0dIUm65/nXNH5gD635cZ70G3lljdft0+TiKl+Puf9R/fDx+9q9KU2tO3byHg8
MDlZ0XHyN1ziyf48GjrxCyW3b66JrbtD8Tz+UJDkdDv1r8i0a8NRLslgsp/sSAAAAAI2e2WxWv706KCsmWC3bt
JfJ/0v4E3dJpWTfYWA64PSYG/fAySr2xFXob6Mr1Ck7X00WBKrdvjyZGvHIzwUlJSpxuXRV5PGXwkgrL9dsq1W
z26TW2N42IFBJfn56JS9XkxMSFWw26/2CAmU5ncr9eaf780LDdHlEha45dFBBJr0mJiYp2GzWl0xsPZuUpI+Li
jSzsFDRFosmJyZ6R4w2Ff5Mb2+SG0np49x0l9wul/x/9wMnIChIO9PWKPtIuopyc3TWued720LDI9TxrF7atTH
tpO7hcrm07Ju5qiwvV6eeZ1dpW/rVbI3rf6YmXT5YH770rGw//5VIklI7ddWWVcvlcjq1cdkitTmjqyRp7rS3d
Ga/c9Whe49TfdoAAACAz4lukSCT3FUKnpJUYokyJhBQR8xqvEXD49mVUK7Hri7UlJsidLBdXKN9Br0tVp0fGqp
4v5qnlpe5XXo0K1P/SEhUtF/N49r8TSa91rKVDtrtGrB3j/rs3qU15eU6PzRUvx08eneLOH3frr2+aNtWw8LD9
Z/8fA0IDZGfpHfy8/Rh69YaHRWpxzIz6uGZGouNjJomRnr6uOCwMHXq2Uf/e+tVtWrXUZEt4rTsm7navTFNia1
TVZSbIOmKio2rcl5kizgV5eWc8NqHdu3Q/11/uew2m4JCQvXwG9OU0uEMb/t5l12luORWiolPOKHdO/TBi88o4
+A+Pfz6NEnSVbffrX9PflR3XTxA8ckpuuuZl5RxcL8WzflMUz/+Uv/6+yPauHyx2nfroQlP/VOh4RF1/OoAAAA
AjUdETAs5HdXX7iyxNO1po2j6TD4xwb1m25LK9PC1ZeqREaWxi/3U+mCe0ZG8jjocWllepv+X3PK4fdLtDh110
DTx6BHvsV9W8+y+a6e+adt0rQMCdGZQk0aktlWJyyWHx6MYPz9de+igugUF1Xjd/Tabviq26vPUtpptLdLZISG
K8fPTiPAIPZ6VpTK3S6HmJjJz02SSf2Ki0SlQDyh6NgH3vvC63vy/B3TboN4yWyxq17W7zht5pfZt23xa101u2
14vzpmv8pISrfz+a73x6H2a8sFsb+Hz4mtv8PZt06mLouPiNXncNcpKP6jE1qkKDY/Q/S+9VeWaf7/pat348ON
a8tVsZR8+pNe/W6q3n3hIn735isY9+vfTygsAAAA0ZqHhkVIN48koesLX+eJIz9/blFyqTddLvQ9HaewSi1qm5
xsdSX0sRYqxWDQoL0y4fdoFB0iL1LZVjv2/vFyVud36v/gEJf5u86Hwn5eY02i3a1tlpe5tUXWAlHRsM+TJ2Vl
6JD5eoWaz3B7J+fMSAL/80+X7/8m9/BISZAoIMDoG6gHT25uAxNapeurD2Zq5fq/+vXCdnv/sWzmdDiWktFHUz
+ttFuXnVjnHmperqBY1r8X5C/+AACW1aav23c7SDQ/+n9p07qpv3n/3uP07ntVbkpR56GCN7T99/rFCIyLUb+g
IbVu7Uv2GjZCfv7/0HXGZtq1ZUYtnDAAAAPiesIgoSccKCr9VYqboCd/WlAoL61NKdf8Yq178S4wyW8UYlsPt8
Wi01aorIyPl97sNjB7NzNDLP8/qDDSb1TEwsMojwmxW6M/HA34+d15JsdaUl+mw3a4FJSW69XC6hoaFaWBoaLV
7/89qVYzFT4PDwiVJvYKDtbq8XJsqKjSjsEDtAwIU0YT25wg8o6PREVBPG0nZhASFhCgoJESl1iJtXLZYY//6u
BJatVZUXLy2rFymtl26SZLKS0u0Z/MGDb/+xlpd3+P2yGG3H7f94M6tkqTo+0rFVGtBvj576xU9PWuuJMntcsn
pPDa1x+l0y0121SoLAAAA4GvCo2Ll7x8gp8Mu/4Bf1+R3mANVaQpWkKfCwHTAqf0ljYx01po2xVozVhq4P0bXL
ZUSMgoa9P4ry8uV6XRqVGRUtbZMh6PWheZcp1Mv50Qoz+lUnJ+froiM1J2xLar1y3M69a/8PM1q08Z77KzgYI2
LjtGdRw4r1s9PzyY2rfUvgzp1+uN08EkUPZuADUsXSfIouW17ZR06oPf/+ZRatuugIaOulclk0mU33qqr/vfP/l
JTaVvEtW+uj115QdHyC+g0b4b3G5HHXqN+wEbr0hlskSR++9Kx6Qe
190s56n3BUIVHRevQ7u16b+pkdT27v1I7da2W8b1nn9TlN9+h2IRjPxw79+6rJV/8Tz0HDtL8T2eqc+++9f0y+
YytW7caHQENpFu3bkZHAAAADSg8MkYBQcGyVVZUKXpKUrElWkF0ip7wTU1hevvxLG9XrOXtpEF7Y3XNUrfisgo
b5L4DQ001vVPnGttmtG5T4/FfPJtUfSfysdExGhv9xyNXAXxcszko/qJLCX/9
yZS3I0+uP3KvC3ByFhIerTacueuLdueuLdWeoxcJAkyc/fX5tXLNXMM96VraJcsUnj6n/xpfrzhEnV6Qd17
wuve49dMuZm7d26WY9eM1IdzuqpayY+WD8vDgAAANBIhEZGKzA4RPbKCunnqe6/sFpaKN7Z9HZERvPQ1Iuev1j
cwarFHaShu2L152UuxeYUGR0JdSTwDEZ6N1Umz+8XlPEBxcXFioyMlNVqUREw+34PXtXZoPdC8YblcmYIfuM9
Gw+G0kJAKf0qM+Dw0n6902p0rx/p5Jat69yvEPlJvUr+9GgVMDpeTa2iz6KKDM6Ro06eGeURi91KDrPanQUnA5
/f3VenybT7zZ8QtPQlNYbBgAAAIBGLTaxlRw2W7XjBLCz80LlId9xWphl/aqGiWP745qsC27al4
NmENcefTQAAAABgiKiYOMnjrna8yNJCbn49g49qihsZnaxvzizS7beW6cPL41lQcHW50HNRSIJsYNWm8qwIAAAB
AAwmLipYk/X6VMbfJT8WWP95kBGiMms0anidkNunL7oW69fZyfXJnEqiKH76isAz0hodAfWIoicAAAAANJDIm
Pifd3Avr9ZW4BdvQCLg9FFY+JnZpNk9C3XbHRX6bHi8SiNCjU6EPxDESM8mjZ9NAAAAANBAouMSFRQSpsqy0mp
thRaKnvBNzX6k5++4zdJnvQt06502zbkoXuXhFD8bq8B0nY20gHpE0RMAAAAAGkhgULBiE1qqoryGoicjPeGjK
CzUzG2RPjq7QLd0s0mrIQmqCAsx0hJ+wy8pSf4J/NxtyvjZBAAAAAANKKlNe9krK6odz/dLZDMj+CSTh5GeJ+K
0SB+ck69bJzj07eAEVYYEGx0JkkJ69TQ6AuoZ76gAAAAA0IBi4pMlVd/MyGXyZ4o7fBKFhZPj8PNoev983XaXU
```

```
z9ckCBbcJDRkZq14F69jY6AesbPJgAAAABoQDFxifIPCJTDbqvWluPf0oBEw0lhTc/asfl7907AfN1+l1sLzku
QPSjQ6EjNUnDvXnVyndzcXE2YMEGtW7dWYGCgEhMTNXz4cC1fvtzbx2Qyae7cubW+dmpqql599dU6yemrJk+er
J49e57SuX51GwUAAAAAcCJRLRIVHBKmirISBQRWHemV499KXSrTDEoGnBozNc9TUhHg1r/Oz9cH/cwatypB56Y
VKcBW/Y8hqHvmkBAFda6bTYxGjx4tu92uGTNmqF27dsrOztaCBQuUn59fJ9dvjOx2uwICAoyO8YcY6QkAAAAAD
SgkLFxRLRJUWcNmRjl+rRgzB59j4bv2tJQHuvXWoHzdeZe0rH+CHD5QTPJ1wbl6yWSxnPZ1ioqKtHTpUj3//PM
aPHiw2rRpo379+umxxx7Tn/70J0nHRmtK0lVXXSWTyeT9et++fbriiiuUkJCg0e3bVz/++KP32hdeeKE0H
Tqk+++/XyaTSSaTydu2bNkynX/++QoOD1lZKSoruvfdelZWVHTfnpk2bNHjwYIWHhysiIkJ9+vTRunXrTvp6qam
peuqpp3TjjTcqIiJCt99+u84991lw98sgjVe6Tm5srf39/LVmyRJL0wQcf6Oyzz1Z4eLgSExP1l7/8RTk50d7+i
xYtkslk0OoIFC3T22WcrJCRE5557rnbt2iVJmj59uv7xj39o06ZN3tdg+vTpJ/3fh6InAAAAAADSwlm3PkK2ivNp
xhzlIRZYWBiQCTh0jPetGaZBLrw3O1113mbWyb6Kc/v5GR2qyQvr1q5PrhIWFKSwsTHPnzpXt0KN0165dK0l67
733lJmZ6f26tLRUl156qRYsWKANGzZZoxIgRuvzyy5WeniSmZNTmZmZko4VS0eMGKHRo0d
r8+bN+uSTT7Rs2TLdfffdx805ZswtWrVSmvXrlVaWpoeffRR+f/8/XWy13vxxRfVo0cPbdiwQU888YTGjBmjj
z/+uMr61J988omSk5N1/vnnS5IcDoeeeuopbdq0SXPnztXBgwc1bty4avn+9re/6enW265RZJ
07bXX6sEHH9SZZ57pfQ2uvfbaY+oqpdQSXPnztXBgwc1bty4avn+9re/6aWXXtK6devk5+enW265RZJ
O7bXX6sEHH9SZZ57pfQ2uvfbaP/zv8guT5/erZ/uA4uJiRUZGymq1KiIiosHuO3tXZoPdC8Yb1SnJkPtu3brVk
Pui4XXr1s3oCADgs4z6PAjUlR3rV+iL6a+qVfuuMpurjkU5u3SBzrBtNCYYcAqmR7bXSzEOo2MOOTHl/rplWYR
6b86Xn8NpdJwmpc1HsxTSq27W9Pz888912223qaKiQr1799agQYN03XXX6ayzzvL2MZlMmjt2rrt
2668847vQXH1NRUTZoOSZMmTfL2ufXWW2WxWPSvf/3Le2zZsmUaNGiQysrKFBRUfYOsiIgIvf7667rpppuqtZ3
M9VJTU9WrVy/NmTPH2yc3N1fJycc66aefvEXOc889Vxdc1IGee+65Gp/funXrdDm3x+PzkWUlCgsLEyLFi3S4MGD9
eOPP2ro0KGSpG+//VYjR45URWFgoKCNHnyZM2d01cbN2484WtXE0Z6AgAAAEADi2/ZRkHBYTVPcfdvZUAi4NS
xkVH9KAhx6MWL83Xvnf5K65Ugl/bstQFU0iIgrt3r7PrjR49WhkZGfryyy81aGZaWpsO8/WbN3tfHYXkqaIY
1//qi5cd+uigqKkphYWHasWO0Hd6Tn8WzatEnTp0/3Xa7deDAgrRrPeeCBB3mqnLJn
v375aX+/ss8+ucs24uDhdfPHFmjlzpiTpwIEDWrlxytW7dWeHi4Bg0aJEnVnuNvC8RJS
ccGoP12GvypougJAAAAAA0sqkWiImJaqKzEWq0t2z9FbplqOAtonCgs1K+8MIeeH5GvSXcEaGOPBLnqYC3K5iy
kd2+Z6riAHBQUpIsuukhPPPGEVqxYoXHjxunvf//7Cc/561//qjlz5uiZZ57Rz77/FQeXHjXRnXy3l12u/2E55WWl
uq0O+7Qxo0bvY9NmzZpz549aa5at++fY3nTJ48ldu2bdPIkSP1/qRmXRv3lbkN6JgfLDMzWDcRnpT1/pJu/QT7LbHMzsXKKa4BZ3zXZHSgFaCkPIumFEJPcuwJMpmgwuZM6LJYhuhcX9uEm5P2SrbbhauPfrFfl1uljjPfi+1vF3v3mMnZS1GlWm9IEPfIfKoZqWRMPHg8M5xmQ6UkiNhGHIW2CuL6m01qP2aoqPN6ClwW6PR6WeG+3iKLp2a1uBgDv++/3T1srZWWkWcqUBNgWcT3ruMQXk7DmEoiIAI2xXRbc9xLDNrhA0CSpPnGhlMqRUCvUqa7s2CzWK3p8WZuBGbWCwmIm8Xov8rOvJUqwRBVk0L+jCXtXjHTJJRmIUmWQ
M455xy98soruuCCC9StWzc98cQTuuKK4Jdtjmnz7Z6Nr4YbrIA4mHtxhb0n9ydJAMZKfxd45bXDPCSqYVGm/MWG1tsvG3LxvpfeOAF4xRQmu9DFoPU0cu2kpAQKR3sftRVIUKJkb1+ZF12oUh7ifBkw3Cw5HIVWwyenTH
```

```
0ZMgQXXHFFdq2bZsk6f7779dXX32lzz77TIsXL1ZGRoZGjRrlPd/lcmnkyJGy2+1asWKFZsyYoenTp+vJJ5+s2
2cFAAAAAD6kdYeuimqRoOLCvGptTlOAMgLaGZAKODms6dm0zO9cpDtvLdN7V8SpqIXvFz8jr7zS6AgwSK2Knpd
ffrkuvfRSdezYUWeccYaeeeYZhYWFadWqVbJarZo2bZpefvllDRkyRH369NF7772nFStWaNWqVZKkH374Qdu3b
9eHH36onj176pJLLtFTTz2lN998U/Ya/qIJAAAAAM1BYHCIOnQ7W6XWmjcUORjAFHcOXhQ9m6bvuhbq9tvK9MF
lcbLGRBgd55T4JSUp5JxzjI4Bg5zymp4ul0sff/yxysrKNGDAAKWlpcnhcGjYsGHePp07d7d1br1q21cuVKSdLKl
SvVvXt3JSQkePsMHz5cxcXF3tGiNbHouLq7yAAAAAICmpG3n7rJY/0Sw26q1ZQS0U6Up2IBUwB+j6Nm0fdW
9ULfdVqaPRsarJCrc6Di1Enn55TKZ2c6muar1f/ktW7YoLCxMgYGBuvP00zVnzhx17dpVWVlZCggIUFRUVJX+C
QkJysrKkiRlZWVVKXj+0v5L2/FMnTpVkXERlZWVVKXj+0v5L2/FMnTpVkZGR3kdKSkptYwMAAABAo9aqXWdFt0iQNT+3WpvbZNGBwK4GpAL+mIW
iZ9NnNmnOWQW67Y4KfTYiXqWRYUYn0imRV15hdAQYqNZFz06dOmnjxo1avXq1JkyYoJtuuknbt2+vj2xejz32m
KxWq/dx+PDher0fAAAAADS0gMAgdezRT2XFRTW27wvq3rCBgJNkdlP0bC7cZumzXgW69Y5Kzbk4XuXhoUZHOq6
g7tOV2I71kJuzWhc9AwIC1KFDB/Xp0OdTp05Vjx499P/+3/9TYmKi7Ha7ioqKvTPzs5WYmKiJCkxMbHabu6/f
P1Ln5oEBgZ6d4z/5QEAAAAATU3bTt3l5+8ve2VFtbZiS6xy/FoakAo4Maa3Nz9ui/RRnwLdOsGmL4cmqCKs8e2
OzihPnPbCBm63WzabTX369JG/v78WLFjbdu1a5fSO9M1YMAASdKAAQO0ZcsW5eTkePvMnz9fERER6tqVqRoAA
AAAmrfk1DMUk5CsooLqU9wlRnuicaLo2Xw5LdKH/fJ16wSHXvh2coIrQxrH2sMnfXXmp0DBisVkXPxx57TEu
WLNHBgwe1ZcsWPfbYY1q0aJHGjBmjMhIjR8/Xg888IAWL1QO3XzzzzRowYID69+vSbr44ovVtWtXjRO7V
ps2bdL333+vxx9/XBMnTlRgYGC9PEEAAAAA8BX+AQHHq0muAyouL5PFULySlB5whu4nfndC4sKYnHH4eTe+fr9s
n0PX9BQmqDAkyNE/EpZfILzra0AwwXq2Knjk5ObxxhvVMnDR06VGrXrtX333+MViyiuv6LLLLtPo0
aN1wQUXKDExUbNnz/aeXR06rKlCl1+6wAAAAAwEd16NZHIeGRKispqtbmDqO6vQG64YYbd0ONN2rKlCl1+6wAAAAAwEd16NZHIeGRKispqtbmMvn
rQGCXhg8FnIClhgI9miebv0fTBubrjrvc+H8RNmCjPkjTfSNNxpyXzQuJk9Nfz5s5IqLixUZGSmr1dzv63
pXZYPeC8UZ1SjLkvnyfNR9GfY8BQFNg1OdBoCF4PB7N+e/L2rslTa3a89Ua3ad67WHunM1Ujr+wYkA2q2yz9Sf24VaXQ
MNEIhNrPGrYrWuWlFCrDZGuSewWf3UeqHHzbIvdC4nfaangAAAACAumMymdSl1wC53S45HY5qbC5q7Va/OGX6tzYgG
VAz1vTE8ZQHuvXWoHzdeZe0tH+iHIEB9X7PmLGM8sQxFD0BAAAAoJFp27mHbyWCXWP7zzqCzGzgRcHys6Yk
/Uhrk0uuD83TXBLNW9EuQ09+/Xu7j37KlwocNrZdrw/dQ9AQAAACARiYoJFRdep+UmthjRsaZQa0VZE1l1oBkQ
HWM9MTJsgY79erQfE28y6LVZyfI6e9Xp9ePHjNGJoulTq8J30XREwAAAAaoQ7deis4JEzlpcU1ltu8M6tPAiYC
amX1vqxAYrDDEqZcuyte9d/prXa9Eufx0v/hpDglR1NV/roNaCrqtqQ0AAAAAKgTiSnt1Kp9Zx3csVmh4dU3ZJB2oJDmHi
TkY2EU9ypcp2FNuQLrGY+pSm2bvdGhnnlvBfiadm2LR88MC1anFr6097viqQj8ecCqjxK0wgF/7dG5x/BFh4+Z
WaMamqmuqDm9v0bwbQr1f78536aH5Ni1Pd8nu8uisBIueGhyowW2P/apdU0HRTXMrtPCAUx1jzfrvn4LVK+nXe
078pkLtos168FxjdriuK+zejl0VF+bQCyPyFD8wQ00XxuqsrXmyuFyndK3IK6+UJTy8jhPClzHSEwAAAAAaIZP
Jp07nDJLb45bDXn3Y7fJT3uCejZ8sEZm8SGnJvYN0KrxoZo/NkQt7t3Txh+Uqs/9QwfX9Di
Dwe6eIPyuVyn7hYN6KDRZkPhnkfH40OqdJ+2awK0d3STzeFK032UPVIM0uyj8qVVeqWJD2zxKYSm0fr7wjVhW3
8dNtXFd5zVx1xavVRlyb1r/+NXeob09txunLC7Zp6ab4euCNQm7snyFXbKeomk2JuHFsCHoCAAAAQCPVv
msvJaa0VX7W0Rrbdwf1lLLOZT+Cbd0OoxvUM0JnxFvVItGj6FUFKt3qUlvnraLHb+wTogjZ+So0yq3eSRU8PCdT
hYo8OFp24WBdoMSkxz0x9RAebvG155W7tKXDr0YEB0ivBoo6xFj03LEjlDmlrzrGi5448t67r5q8zYi26vY+/d
uQd0+5weXTn15V657JgWcymGu/tSyh6oq5kRtr19GX5euj2IG09M15u88mVrcIGD1JgamqdZR+aH9J7mvWf6iO4M7ZmCsH6vRUcrQHsm8D8qWdSk
UfOCgzhowRLaKcrlczmrtdn0w9gadZUCyxsv686DYm0Cai4lldo/e2+BQ2yiTUiJPXBcdNCp+H+WqNMbpZrwd
YXyy93etthgkzrFmvX+K7B453R79K82u+FCT+vw8hb1Hglk/HXDK6fbo+31OnZVw7PgLy+26MNVPZyc3jQ1
XmN6OunYkyqYpfyrQI7eFaEeXPy5+trjrrgZKBl9C0RMAAAAAGrFOPfopJj5Jj8X7/sLt8WjSv
EoNTLGoW3zVgujJba+0ep+aPDVWA5fhFzxEd/PT+VcFacGOInh8WqMWHXLpk7Uk368MUQ
bslwKn1qioKdL9PJKu+aNCfGOCH30vED5maX2r5Vqzk6npv0pSHvyXZqxyaEnLgjQnV9XqN3/K9E1n5XLWum7h
UMKC6gvh2Iq9fcrC/S38aHa1Sleb1P1/2fDBg1ScLczDUiHxo6fTQAAAADQiIWERajnaBDiZj70Reqt
KhAnhpG1FWaQ7U
nqIcByRqfid9UamuOSx//Obha25ju/tpwR6gWjwvRGbFmXf0/clU6j19ovK6bv/7UyV/dEyy6sr0/vv5LiNZmu
LXo4LFp8x6PRx0/rVR8qFlLbw7RmttCdWVnP13+UzS46NCIOMMmnW6BAdmhSuxeNC1TX0oju+rtQ/LwrUzC0
07S90a9fdYQrxN2nK4urrtvoKC9PbUc/2tajQE6MK9MT4MO1pH1vl067F3RMNy4XGjT8HAkAztHXrVqMjoIF06
9bN6AgAgDrQudcA7dm8TgW7M+BKGlkMtlcAlt3x7cTx0qN8tfjjhr0bh7u/rZCX+9xasm4ULWKqD6+JzLIpMggizrGSv1
bWRT9fInm7HDq+u7+J3X9dtFmtQgxaW+BW0PbST8dc0nr3U4VPhKuiMBjo8/eGhms+ftLNW0TQ4+eV31H9vc22
BUVZNIVnf016pNyXdnZX/4Wk67u6qcnF/lw0dPj/uN0QB3YE1ehv11ToS5ZkXpoX1slmCMV3L270bHQSDHSEwA
AAAAauZj4JHXq2V9FeTk1jva0mU0a7U7uHo9Hd39boTk7nfrpxhC1j7jX3M9nmMPm+vkRygeKXYrv9yjpPBjB
c5yx7Fzf78Pkdkk1bQpfG6ZW10W2PT6JUGSJJfn2IZGkuRwSy4frhv6UfREA9uRWKZbBm5V2ZR7jI6CRoyiJwA
AAAD4gG79LlBwaJhKrYU1tu8IPlsOndyoxaZk4reV+5UrKfSPt2q+Lkoub/QralLbUrLcCnd6
```

taKw05d/VmFgv1NurTjr5Mf079Rqjk7jo2ULbV79NAPlVp1xKmDRW4t20/UFR+Xq0OMWcPbHztnQIpF0UEm3TS
3QpuyXNqd79JDP1TqQKFbIztWn1Q56ftKPTggUC1/HoU6MMWiDzY7tCPXpX+nOTQwxXc3NWL3dhhhaOuh6hLPW
p44Pqa3AwAAAIAPSGrdXp16nqMNy39UWGSOTL/bOMNmDtHuoF46s3KNQQmN8fa6Y4XKC2eUVzn+3hVBGtczQEF
+0tJ0l15dbVdhhUcJYSZdOMaiFbeEKD7013FAu/LdstqOFe8sJmlzzrENh4oqPUoON+ni9n56anCgAv2Ove4tQ
syad0OI/vZTpYa8Xy6Hy6Mz4y364rpg9UisWsD8fq9Tewvc+uCqX4vSd/cL0LoMl855t0z9Wlr09wuD6uX1aQi
s6YmGZjaZdU8vRnnixCh6AgAAAIAPMJlM6nXeRdq1aY1KigoUER1brc+O4LPVwbZZgZ5KAxIaw/P3iBO2J4eb9
e2YkFpdJ9jfpO9vCP3Dc85OtpxUv+Ed/DS8Q1iVYyH+Jn169R/n8gW+OOYVvmpk25FqH9Xe6Bho5Ch6AgAAAIC
PSGjVVp179deGpT8oPCqm2mhPuzlY24LPUe/yxQYlRLPlkWT6w171omxXmfK+zVPFoQo5i5xqfU9rRfT5tYi9d
VzNm3gmXJOguEvjamzL/ylfBT8VyJF3bCRxYMtAxV8Rr/Czwr19HEUOZX2SpbJtZXJVuhSYFKi4y+IU2TdSkuR
2uHX0v0dVsqFEfpF+Sr4xWWFn/lr8zv02V458h5LHJp/2a9Cc+Jn9dFfPu4yOAR9A0RMAAAAAfITJZFLPc4dp1
8bVKinKVORO0i2p9dgf1UsfKTQp3FzV8QDRbJpnkMWiau9vmVlDrIEVfEK3019OrtXd6tVOVrOu3lOrof48q8uz
I417TP9pfiVcnKiAhQJJUtKxI6f8vXe2ntFdQy2NLERz5zxG5y91qPam1/ML8VLSqSIffOqyAyQEKbhOswkWFq
jxUqXZPtFPp5lIdfuewOr/WWSaTSfZcuwoXF6r9ZEYr1taf0/5ZrcJbGRODPoCNjAAAAADAhySOSlXnnv1VkJN
Z4O7ubpNFGOIuMCAZmjOTUcM8JYWfFa6EOQlVRnf+ln+Uf5VH8fpihXYOVUB8wHGvGdErQuE9whWYGKjAxEAl/
DlB5iCzyvf+unZsxd4KxQyLUUi7EAXEByj+T/GyhFhUcbBCkmTLtCm8Z7iCWgYpZmiMXCUuuUpckqSMGRlKvCZ
RlmAWB6iN6MBo3d3rbqNjwEdQ9AQAAAAAH2IymdRz4DCFhkequDCvxj5HAjsq24+RUMDvOa1OlWwuUfQFOSd9j
sftUdGqIrltboV0+HUd1uAOwSpeUyxnqfPXPg63QjsfW+c1KCVI5XvK5ba7VbqlVH5RfrKEW1SOokgmf9Nxi7Q
4vvt636fIwOOPOAV+i+ntAAAAAOBjElqlqmufgVqz8BuFR8XKbK4+nmV96IUabp0pMztrowEYOdKzNgqXF8oSZ
DmpgmPl4Urtf3q/3A63zIFmtb6ntXdquyS1vqu1Dr99WDvv3ilZJHOAWa3vba3AhEBJUvT50ao8XKk9/7dHfuF
+SrkrRa4yl7LnZKvto22V/Xm2rKutCogPUMvxLeUf7V9vz7sp6N6iuOZ1HGVODPgQip4AAAAA4IP6XDBCu7esV
WFupmITWlZrL/RLOMHArmpn22ZAOjQ3vlHylAqXFCqyf6TMAX888TUgKUDtp7SXu8It61qrjrx7RGOfbestfGb
Pzpar3KXUh1NlCbOoZH2JDr95WO3+r52CUoJk8jMp+caqmxQdefeIYi+KVWV6pYrXF6vDUx2U+22uMj/MVOt7W
tfLc24KzCaz/nbO36pt3gacCNPbAQAAAMAHxcQnqc/5w1VaVCCXO1ljnOOh58lhOv66hUBd8YWRnmW7ymTPsit
6OMlNbTf7mRWYEKjg1GAlXp2ooJQg5c/PlyTZcmwqWFCgluNbKqxrmIJbByv+yngFtw1W/oL8Gq9XuqNUtqM2x
Q6LVdnOMoWfFS5zoFmR/SJVtrOszp5nUzS642id2eJMo2PAx1DOBAAAAAAfdVb/IUpIaaecjEM1tleYw7QpeGA
DpOLz1PiLnoVLChWUGqTg1sGndgGP5HEcWy7CY/t52YjfPW2T2aSaVpRw293K/CBTyeOSj/VxSx7Xz9dyeuRxs
wzF8UQFRum+3vcZHQM+iKInAAAAAPiokLBw9Rs8Uk67TbbKihr77A7qpXxLQgMnQ3NjZHHBVelSxaEKVRw69v+
APc+uikMVsufbf+1T4ZJ1rVUxF8TUeIODzx9Q/o+/jtDM+izr2MjQXLsqD1ce+3pnmaIGREmSApMCFZAQoIzpG
SrfXy5bjk153+WpdFupInpXXy8098tchZOVpuA2xwquIR1DVJxWrMrDlSpYUKCQjiHVzsEx9/a+l82LcEpYOxM
AAAAAfFjnXgOOde1SHdq9VSkdulTvYDJpTdhFbGqEemXk9PaKAxU6+PxB79dZH2VJkqIGRqnVbaOkSdbVVklSZ
P+ai2f2HLucJb8uE+EsdurIv4/IaXXKHGxWUEqQUh9MVVi3MEmSyc+kNve3UfZn2Tr06iG5K9OKTAhUy1tbKrx
HeJVrVx6plHWtVR2mdPAeizg7QmU7y7T/2fOKTAxUqztbnf4LOQR1i+2mOR1HGxODPsrk8Xh87l2vuLhYkZGRs
lqtioj44x3X6srsXZkNdi8Yb1SnJEPuy/dZ82HU95gkbd261bB7o2F169bN6AhAvTDq8yDQWB3YuVmz331REdE
tFBoRVWOfXmWL1KUyrWGDodno26aDKs32P+4InCSzyayZl85UtxZ8nsWpYXo7AAAAAPi41E7d1bnXAOVmHJbb7
a6xz+aQgSo18OcC1A9f2MgIvuWqDldR8MRpoegJAAAAAD7OZDJpwEVXKiouQfnZR2vs4zL5a13oOAZOhuaCkif
qUkJIgu7vc7/RMeDjKHoCAAAAQBMQE5+k/kP/pIpSq+zH2dQoI6CdDgV0auBkaA4Y6Ym6YpJJTw18is2LcNooe
gIAAABAE9H9nAvVrktPZR3er+Nt37A2dKjKTWENnAxNHSVP1JW/dPmLBiQPMDoGmgCKngAAAADQRPgHBOjc4aM
VFBIma0FujX3s5mCtDhvewMnQ1DHSE3UhNTyVae2oMxQ9AQAAAKAJadWuk3qdd5GKcrPkcjpr7JMZkKrdgT0aO
BmaMkqeOFOWkOXPD3pegZZAo60gifAzOgAAoOHt9o81OgIaCPtdAkDz1PfCS3VgxyZlHzmg5NSONfbZEDpIiY5
ORbgLGzgdmiJGeuJOTegxQV1juxodAOOIIzOBAAAAoIkJCYvQucOvkjwelVprLmq6TP5aGXaJ3PxaiDpAyROn4
8zoM3Vr91uNjoEmhnc3AAAAAGiCOnbvq7MGDFHu/2/vvsOjqNo+jn93N7OnEFIgIYSQEHqvUoJgqNIeQUEQRVE
BEQEFVJoNUCyg2BXQVwR5RFBUikgHpQYQQguBOBDsJAdJ25/2Dh9WVAIlCAuH3ua69dGfOnLln9uyQ3DnlSMpVh
7mfcg5hm3udAo5MiiKTobSn/DOuZlfeaPoGFrOlsEORIkZJTxERERERkSLIZDkSGVnSpWJ4ej+pKuW+8O9Pie
cQgswMimKlPKUf2ponaGE+YQVdhhSBCnpKSIiIiIiUkR5evvSuG1XXnFxcSD11PNcyhsnMSq+2Zg6KUoOp
6f8Ew1DGnJfzH8HERMVWo2bglIiFGERMVWo2bglZ04cJTsrM9cyFy3erPZM9cyFy3erPZqjU2JK/mH1HIkvwJdAxnXeFx
hhyFFmJKeIiIiIiIiRZjJZKLu3e/nffnZtnlEN8ARVbGHlFmQyRHcjlfiFsZ1TNHBbDnd/nffzc/Ao7FCnCl
PQUEREREREp4lzwv2oyqeZSPYbIWOpc7ohn+6jSDBdaE8ZBOGdhOCbdcrv6dZ8mrkuvzgbGwXzRBnX0EVtL

O69ePYsWK4eXlRefOnTl27JhDmZSUFNqOaYOHhwclSpTg2WefJecq/+CKiIiIiIjIjWFxcqLpvdOoGRnDoX27r
jq/5OnnkqzzbF7AOcntTPO85Xp88OGj1h9hMVsKOxS5Q+Qr6bls2TL69evHb7/9xqJFi8jOzuaee+7h/Pnz9jL
PPPMMP/zwA7NmzWLZsmUcPnyYTpO62fdbrVbatGlDVlYWq1evZtqOaUydOpWRIOfeuKsSERERERGRXHn5+nN3h
x54evlw6uihq5bb61aZRLeaBRiZ3M5MSnvKNVgMCxObTaS4V/HCDkXuIE75KTx//nyH91OnTqVVEiRJs2LCBxoO
bk5qaymeffcb06dNp1qwZAFOmTCE2NpbffvuuNevXqsXDhQrZv384vv/xCUFAQ1apV4+WXX2boKGMHjOaFxeXG
3d1IiIiIiIicoVSZctzV+suLPzmM86fS8XT2zfXcps8muBjPUPJ7LOFHKHcbpTylGsZVHkQtcJqFXYYcof5V9N
upKamAhAQEADAhgObyM7OpnnzP4dBlC9fnvDwcNasWQPAmjVrqFy5MkFBQfYy8fHxpKWlsW3btn8Tjoi
IiIiIiORRlXpxVG3QjBOH9pOTnZV7IZOJVd5tOGNR7yy5NiU95WriS8TTs2bPwg5D7D7kD/OOlps9kYOHAgDRs2pFKlSgA
cPXoUFxcX/Pz8HMoGBQVx9OhRe5m/Jjwv77+8LzeZmZlkZmba36elpQFgs+HaeI/hBnDezxKOudkhZNcmy2ooNeR6pjAc9hVVp9Jjws/jzmaMlYtbdsLtgjc
jkmF5Z5d+SiyaOAI5TbiSn36WHlDlfOtRzj7hlX2GHIHeofJz379evHH3/8wYwZM25kPLkaO3Ysvr6+91dYWNh
NP6eIiIiIiEhR5+HlTYv/PEyxEqEcTUm66sJGFyw+rPBujxUtQCK5M6uvp/xNkCmIqe2n4mTJ18yKIjfMPOp69
u/fn3nz5rFkyRJKlSpl3x4cHExWVhZnz551KH/s2DWFCg4PtZf6+mvvl95fL/N3w4cNjGRvBcvLyhbaxWYnn4Zf6q1EbLJyp25p/JNzhjHRJSsqLb7OQLHKyfBuAq/mFLcs4aLiO5IcLMB7iWY4j6sm4E8IzqcXEo69MtvOpqRDmS2OcaRR37DcFbo8qv+/PTAcXiP3++EWvVvh5c/wUZKQcF/Lq9pVGJJqD/zQdzrcozp3SJ15uBlFRERERERFPkvL/mH0yWOgoRNAAM6fr+iW5/b13u5o1qt+qAc7kBAh/D8rfZvhbixfoTU0g6BHj/DXbsmzOAuMxGLh9w+Sv+yKt6CHF8BS0IpfS0PcrzTW0XIMZpg8TpaPVR1DyAtBIR3N7D3mTo6avyGoczMdbuQ8xuYxrFsJTcsV1NCo8PP0M2VRAKV6XnjtOHNJQUKtrHjcCzW2XBLZeQ7mvSSIIA6WuX2GY/Au3NN2dl2rhKIRlxC5GLcr8JlR0MHE8sv3FuuXxK2t8j1KIkUd8pMkvJRAkdqfkWYOUn1p0+PAuEhpQlLPp0oaKLErSMewYa5rQ2oKIsWL5UwvHbm6IiIiIiHzqBF9IIZIdBIRKbbzs4I1iXsq2/NqAqgYSbAC6ulsl9z++d0fTmsvTbB3CjqKAhBMOQ2m+dGX4cQSmoKbajIh7D5O+N5bWWtMW5RMZa9M/SGZdMyhXyWfOU3fbxY4HQeNWhiMUP4CYB1ORgV+fCKnSB6WqjqrKzprU2tzwuiNoqvbXLj1lC2SkHhMGeKCmMS1wY8rIqQzWNuDySZbNaNBTKJwXs3U7Yl1Mcg+cpwoDqgX8EpdKfdsrPxwh0MHg+L/nwjIQxzd4Gg6bK9hYPlL3RlmnTq3dr0pk+3+KsUKCgyk6M2b4mvQTymuKlA+aoZVTbb3YxOvLV5fIj7D+vsrL2GkLyD6vx5ojYYFr2vPf0tKaoFYK7p5Y9i/ssuGNjlTeSKOCMw0tM3tS8N9HsAK9+Ps0RP4XFBbndGvMYtBPQmbtO0MJ6+1i+WnVSUEJNpLbWNfCfi8bs/7BQOCLPlvXN9+Tt7+ObKw6WyQpULSFi+fO
eGc48zIKiOpGVmzsEMRyd9z6YMPPiA1NZWmTZsSEhJif82cu2336Zt27Z07tyZxoObExwczOzZs+37LRY

gq16lGpbgM2rVjC+iWLeOmL/9rLTBo6gIASwTw4+HkAZn/OLmUrVSEoPIKcrCw2LlvMsu+/pc+osVfEsHnVMo7
s28tT4yYCEFW5Kof2JrFx+a+cPHIYs9lMaBkNwxAREREpCCaTiUp1GuPq5s7Cb6dwaO9OQstEYzbn3s8mw+zJY
p8u1D7/C2UztxVwtHIzqEfV7c3ZcKbk3pIOCG3Ao1OeJTAgsLBDEvnXlPQUkXx79MVX+HrS63z8OnDSTp3CvOQ
QLbr24L6+z9jL1G3Rij6jxzH74/f4/NURhJaJ5NlJnxBbs669zMnDhzCZ/vzxKOPiBT5+6XlOHz2Ci5sbJcuU5
enX36Vh6/YO58/MuMinL7/AoLc/tP8gXSw4lN4vvszk55/BycWFp8ZNxNXN/SbfCREREREH5q3JVauPq4cmCmZ9
yYE8ipSJjsDjl/munzeTE714tOWsJpPqFZZi1vvttTXN63r48rZ6U3F2SemXrOfu+3gT4BhR2SCI3hMm41riDW
1RaWhq+vr6kpqbi4+NTYOedvfNIgZ1LCl+nmMIZbqN2ducorDYGamd3ksJsZyI3U2H9PCi3r4iICAYOHMjAgQP
zVH7q1KkMHDiQs2fP3tS4boZ9+/ZRpkwZNm3aRLVq1QolhuOH9jN/xscc3LuLkpHR111cMjhrHw3Tf8TVyCigC
OVG6x9cmWXuqYUdhuRTsYxiBCcFO6RaE7q164avt29hhyRyw6gHuoiIiIhIEXbO6FGeeuopIiMjcXV1JSwsjHb
t2rF48eLCDq1ArVu3jj59+hR2GAUiLCyMIOeOUKlSpTwfO6tXLzpO6HDDYihRsjTtHx5IVKUaHNq7kwvpadcsf
9QlggW+3Ui1qIfZ7cp823WnkpDUEMrsKOPnuM7Ovq+3Ep5S5Gh4u4iIiIhIEbVv3z4aNmyIn58fb7zxBpUrVyY
7O5sFCxbQr18/duzYUdgh3hDZ2dk4Oztfs0xg4JOzP53FYiE4OLiww8A3IJC2Pfqz9Puv2PLbEnz8i+NXPOiq5
dMt/sz3fZA66b9QJmt7AUYqN4KGt98+LIaFkkdKUiarDFO6daFRrUaYTPoEpehRT08RERERkSKqb9++mEwm1q5
dS+fOnYmOjqZixYrV4YoMGjSI3377zV4uJSWFp4QJ6Bm3SEiF/bKLrt6gWUXNb9ER2+kyHr28u18eJ+8JiPTjQa
0s6cHR7Q0rcPJG7Wk72N5Qcgp8ba/QnZgzHnt//SLpLlsuVgHIOKmnwoCIGgBZUtThHuXULdhJ2i6n0bUZfLBKN

WCp5ZPT1vGT42H+qerAu7oFLZSgx4aAD1q9fXgkVyx1HSUORERESkiJo8eTJWq5U6derw7bffsnv3bhITE5k0a
RL169cHoHnz5lSuXJnu3buzceNG1q5dS8+ePWnSpMkVQ8gLwsiRI/niiy8YM2YM27ZtIzExkRkzZjjMn5mbUaN
G8fXXXzNq1CgSExPtiwPlJioqiuzsbN5991327t3Ll19+yYcffpivOMuVK0dOdKSgozZswgKSmJSZMm2XtxAly8e
JH+/fuzdOlS9u/fz6pVq1i3bh2xsZcWDhk4cCALFiwgOTmZjRs3smTJEvu+vn37cuDAAZ566il27NzB3Llz3LlzGTV
qFIMGDcJsNuPm5sbQoUN57rnn+OKLL0hKSuK3337js88+u6a4cS2rnn+OKL0hKSuK3337js88+u2qs69evZ8GCBezatYsRIOawbt06hzIRERFs2bKFn
Tt3cvLkSbKzs+nevTvFixenffv2rFixguTkZJYuXcqAAQM4ePBgvu5Xbkwm
E+Wr1aNzn+coV6U2h/fvIfX0ies
dxG636vzs15PjTiX/dQxy45i0lFHhM6Bsdlmi90RjO22j/d3teeqhp4gMi7z+sSJFkJKeIiIiIiJFVGRkJBs3b
iQuLo7BgwdTqVIlWrRoweLFi/nggw+AS4mnuXPn4u/vT+PGjWnevDmRkZHMnDmzUGK0j49rx5LFy4kNq1a1O
vXj3efvttSpcufc3jmjZtyqZs/j++++pVq0azZo1Y+3atbmWrVq1Km+99RavvvoqlVKVV4wd0zZfcd577
70888wz9O/fn2rVqrF69WpGjBh32+xWDh16hQ9e/Yk0jqaL1260KpVK/uCQlarlX79+hEbGMvLli2Jjo62L6J
UsmRJfvrpJ9auXO0Pid8SIEQwePJiRI0cSGxtL165dr5hP9bLHH3+cTp060bVrV+rWcupU
6fo27evQ5nHHnuMmJgYatWqRWBgIKtWrcLDw4Ply5cTHh50p06diI2NpXfv3mRkZDjMXfpvFQsK5d6HBtCodRc
upp/j8L7d1x3unm7x4xefrmzwaEq0lqq4JainZ+HytnnTMK0hnomeBPkH8WT3J+nSugue7p7XP1ikiDIZeZ3h/
BaSlpaGr68vqampN/Qf2+uZvfNIgZ1LC1+nmJBCOa/3DyGYbDnjw0s++F
rjh9KISisDO6eXtc9ztt6mjrpiwjK+fe9T+WfGxVYkdle5wo7jDuO2TATmxNLwKEAzqWfo17VenRp3YXAgMDCD
k2kOOlPYiIiIiIlLoTCYT5SrXonhwKVb8+A2Jm1bj6u5J8Cwa87nes4SwGLfrkRkbqf6+WW4GxcKMGq5TAs
ZFbzi1uJUv1idE/tOYPYyO6N9D5rVa6bFikT+R98EEREREREREuWX4BwbTpkc/SsdUZvXC2RzYvYvY2gsEjcPK49T
HefawUOOUdS9cIqojI3a7h1ATPpfhcYV8OVqtlV8T/tz+Hjh6lYtiJd2nShXOlyhR2ayC1Fc3qKiMgtwWq18vX
E13ny7ro8UDWSvi3qM+v9t7naLCwfjRnzX5IkHhzJv2yTXrnf/1NJ65924erBnNgzWjGd61HRuX/+pQ5mjKPsb3f
4SH61fiwZrRTBj4/0PDCO/rmQQnZWJh0fe4oHa0bTP/4uNq9e7nD8nM/e59OXX/iHVy4iIiJ/Z7FYqFo/ji5PDCO
2RgOOH97PicMpV/254LJssxvrve5mgW93TjoFF1COAkouFASTYSIqJ4qW6SOx9hqc0XuGNk3a8HSvp5XwFMmFn
ksiInJLmPPJZBZ8PY1HR7zkzKxB+X0WPwC8z59H1++vLKlWh/X/QzuzZvIKDE9X+ZKRYUwoODn+f1b+fz+n9/plK
9hozv9zApu3cCkHHhAi/1fgCTycToqbN4dfpccC2czPGvkQ/ZFFBbN/D/2btvCazN+oEWX7rwzpJ/9l65jB1P45
ZvpdHtm2A28GyIiIgJQLKgk7Xo+RXy9xXlXkXrz+8PUzTkEs9OnGWs/mZJjcCiBSMd9+y4XcVgJsATT
PbE7EsQj27t1LqeBS9H2wL93adcPL4/tgk7N62n9t3x1GzaHIASpcJY8/jqSpS30XgzO4rq7V5iMd4CCbDqhgsDSmNCd2B6tzF96HPWJMWqopuq62XW2SnHVpQzRnUOPg+hiGkxZtdD7qBhQ4ELwpXeGwrOuoXzQmsm3KgWaVv0z0W1xxYT1DCj5H/Xx5i8+PyVbEOkbcYnZ3xXyoIGqgmLyVgpHkJXBBK8g+ONE6JIk8ZhpkZ3BxNwQD7tg4XXBbC7IdPBvAoJVdKGbU7T1W0XF16j3Xg7XqrTRhkN0cs0h4+GAF0ETE7bnCtxTeZnYgWzfR8I/GbNe8yHIkTfNTRc1tJh3Q47wFXCyW3OcIHwMhqUxy
cu4daze4hvFwMQWHhfPHGy6Sd0Y1vQDE+Hj2MHkNesB8rIiIiN5bFyYnqDZtTKjKGFT/OZNfW9bi6uRMYEo7ZY
rn6gSYTe9yqst8lhooX1xKdsQkncgou8DuMelTdHJ42TyrnVCb4QjD7D+7Hw92DDs070LJxS3y8tJCfyLXouSQ
iIreEmOq12LpmJYeTkwDYt2Mb2zf+TvVG1wEhhIVwrDCZc
HZxsZdzcXXFZDazY8NaACJiKrBjw1oyMy6ShIp/oFB+PgHsPyH2Ti7ulK3Rat/c+kiIiKSB4EhYdz70N06fY
knt5+7N/1B2lnTl33uGyzGwmejZnn9zB7XStgw1QA0d55lFy4sdwMN2pm1aR1RmtcDrmQnJJMhagKPNPrGbq07
qKEp0geqKeniIijcEjr26c+F8cY0RDmBRvqYa1lZNKsp0DWrAxWErb+Zq8ZZkx6g5e++
C/T3xnPqp/mEhRWmn6vvUWxoJAbd2NERETEzsnZmcp1m1A6uhLrlvzIlt+WcPbUMYLDInFxvfYw9gsWH37zasU
Ot5pUv7CckOz9BRT1nUHD228MZ8OZ2JxYYnJiOH/uPDuP7CSwWCAPtn+QpnWb4naddi4if1LSUOREbgmrf/6eF
T/MZuCEyYRFxZC8Yxt4jp2IemPLfz45ae88eOCTKb89dAILVOWCd8t4sK5c6xZMI/3hj3NS1/OJiw
qGt+AYgx+5yM+Hj0cn778DJPZzF1tOxBZoTJms3nUOjrHr1JTvyDtYvn8+acX5jz6WQ+e
2UEz7376Q27LyIiInIlH/9iNOvs3XFs27kZNO8figeXwmy+/dp/DsO4lWOLzH4Kz9lH1wkqKWY8VUNR
Fm3p6/jsWw0J0TjQVcipANiQfSsZsMhNXL462cW0JDtTCXCL5paSniIjcEr5442Ud
sfvEtexC4kbfif11Ekeb1bbfozNamXa+DHMm/YJH/669qp107u4EFK6DABlK1Vhzx8J/PjFpzzx0usAVLurKe8
vWkPamVNYLE54+vjS+66qBIWF51rf1t9WcWDPLp585U2+eP1lajRuhpuHBw1a3cvPX3XK9RgRERG5sUwmE+FRF
QgJL8sfzz++LvSdn1B/41QvDxL37dP5IedYngqEsEoV17qXTxN4rnHCmgyIsmM+rp+U+YDTN1rWWpmF0RV5s
rR44f4WzaWcqXLU+7Zu2oWr5qvv/gLyKXKOkpIiK3hMsZF+3rDl1msVz6GZCFHzdrJTsbrMdV8z94+Ws2//5aZj/5H/50W40bt+ZZ
h275utchs24NJfn3/j4FwNg628rST11kt1ktpx91xRJiszg09ffp6n33gPi8WCzWaFnEsxnWnOyL7OXERGRAuPs4k
1u1pQJrYqG5bN5491y9y9m/6w8CQ8Px9Pa97vGHXSI57BJJUNZ+K138jaCcgwUQddGjnp75YzJMhFvDqZJTBU+b
8d0HuPkmZ0EBIbwYPsHVKnCR7uHoUd5g0xevRo5syZQ1QJCQmGHIncYJT1FROSWVqNQSR+qs
PUz+iWef7AfD2D8DbP8DhGIuTE/7FS1AyMsq+bXSvLtRp3pLWDz4CwP+4CwP+9+RrVGzcjMKQkF8+ns2Led2xbu5oRn
063H/PrtzMoVbYcPgHF2Jmwgc9fHUHrf/oUbjZkRWqAxA+Rq1+eKjvX+5pC+RrqmTvX
HH38UdghSQCpVqlTYIYiI4FesBLE1Ui8Bvjy

FEXHSbN6ZknzoYzkTmRRFuj8bR5cvLMSfaf3E8x/2L8J/4/NKnbhGJ+xW74eXv16sW0adMuxeDsTHh4OD179uT
555/HyenmpoaGDBnCU089dVPPIZIbJT1FROSW8OiLr/D1pNf5+KXhpJO6hX+JIFp07cF9fZ/JVz1HU/Zx7sxp+
/vU0yd5d+gAzpw4joe3N6VjYhnx6XSqNmxiL3NoXxJfvT2W9NSzBIaG0fmJAbTr1eeKulN27WD1/B9487tF9m3
149uybe0aRnTvSGiZsgycMPkfXL2IiIjcKMFhkbR5sB8Vazdm3ZJ5JO/YisXJicDQ0ji7uFz3+OPOYfzqHEax7
MOUz9hAWNYezNgKIPLbm3p6XpunzZPonGjKWsviZDhxNu0s245tw9/HnzZN29CsfrObPm9ny5YtmTJlCpmZmfz
000/069cPZ2dnhg8fflPP6+XlhZeX1009R15kZWXhkodngBQdJsO4/f4ck5aWhq+vL6mpqfj4+BTYeWfv1Bwvd
5JOMYWz+rLa2Z2jsNoYqJ3dSQqznamn552jMHp6FtbPgyJye7Hm5LBry1rWLfmRQ/v34OLqRvHgsDwlPy/zsKZ
RLmMzUZlbcDUybmK0t7cP/Mvxvl9mYYdxyyluLU5MTgylbKUwYyYtPY2DRw/i6e5Jnap1aNGgBeGhuc8jfyP16
tWLs2fPMmfOHPu2e+65h3PnzrF06VJeeOEFvv76a86ePUulSpUYP348TZs2BWDq1KkMHDiQmTNnMnDgQA4cOMB
dd93FlClTCAm59LPm0qVLee6559i2bRRvO6zs5UrFiR6dOnU7pO6SuGt9tsNl555RU+/vhjTpw4QWxsLOPGjaNly
5YA7Nu3jzJlyvDtt9/y7rvv8vvvvzNjBNCAiIoL
evXuze/du5syZQ6dOnZg6depNv9dy69AfY0RERERKRIsjg5EVujAV37vUjrBx6nWIlQjuzfzeH9e8jOyluC7
oLFh82ejZjj34ffPVtw1lL8Jkd9ezLfdt2pbh6TYSI8J5wWGS1okdWCcFs4Fy9cZMfeHZw8fZIG1Rsw5NEhPNL
5kQJJeF6Nu7s7WVlZ90/fnzVr1jBjxgy2bNnCfffdR8uWLdm9e7e97IULF5gwYQJffvkly5cvJyUlhSFDhgCQk
5NDhw4daNKkCVu2bGHNmjX06dZ4kTfffJMJEyawZcsW4uPjuffex30B/DCCy8wZMgQEhISiI6O5oE
HHiAnJweApKQkWrZsSefOndmyZQszZ85k5cqV9O/f36GOt24uIiMhNscv5x
s9HJbcmzegpIrc6Vzd3qtZvRvnq9dm1eS2bVi7kcEoSFosTgaHhuLi6XbcOq8mZJLcqJLlVISg7hXIZCZTMSsK
ioe+AVm8HcDFc/pyv0/DEMAxS01M5euIoZrOZauWrEd8ongpRFQp1RXbDMFi8eDELFizggQceYMqUKaSkpBAaG
gpcmoNz/vz5TJkyhddeew2A7OxsPvzwQ8qWLQtA//79eemll4BLoy9SU1Np27atfX9hwgSGDh3K/fd
fmrt//PjxLFmyhHfeeYfJk/+cKmrIkCG0adMGGdDFjxlCxYkX27NlD+fL1GTt2LN27d2fgwIEAlCtXjkmTJtGkS
RM++OAD3NwufaebNWvG4MGDb9Stk9tMvnt6Ll++nHbt2hEaGorJZGLHLoGg2XvjwjR44kJCQEd3d3mjdvfkW2/vT
p03Tv3h0fHx/8/Pzo3bs36enp/+pCRERERERERK7F1c2dynWb8MBTI2nXoz8Ic/1HHMOZ
6X3vczxf5wNHnGcsQTexKhvD3dq0tNsmAmzhtEosxEdMzpSPac67lZ3Tpw+wfY92zmTeoYaFWswoOcAnn7oaSq
Wq1hoCc958+bh5eWFm5sbrVq1omvXrvznP//BarUSHR1tn3vTy8uLZcuWkZSUZD/ew8PDnmAEKBQAJxbnwzAAAm
ATQq1cv4uPjadeuHRMnTuTIkdynO0pLS+Pw4cPE2UYZs5zN12Fah7qH2FHXXvqHZDDUq2U5nGz7t+fOsfaq0AEB
NZiM5Odl+XK1atf7JrZIiIt89Pc+fP0/VqlV55JFH6NSp0xX7X7X3/9dSZNmsS0adMoU6YMI0aMID4+nu3bt9sz7
d27d+fIkSMsWrSI7OxsHn74Yfr06cP06dPzfyUiIiIiIiIiIiIiIiIiItkRnJ2caVirbuIrlKHPX+sZ90OqXziOdydWaw7+gcF
4+QbkKTGVafZgp3sNdrrXwD/nGJGZ24jITLwj/68o4a3GxBoCyTCGkG4NRwXLsORm5OTw+GThzmbdpZifsVo2
agl9avXp2x42ULt2XlZXFwcH3zwAS4uLoSGHhuLk5MTMmTOxWCs2LABi8XiUP6Viw1ZTP/zs77DOZTPx1iiZgpU6Y
wYMAA5s+fz8yZM3nxxRdZtGgR9erV4/vLPBf5KqptuqPB6vt6NtGD/ZgTmt4uk3wlI+RqbFt6epPvW8VA/p6ntgpNlVz
aoVrVq1ynWfYRi88847vPDhf+aqtPFmj2LDY7aaM8NSIqgtugOFvpgMFPsUjbbQYxqF6/mn8uXgBdNa8mUoCAa3ZNzZAPpLmacS2vU0hi
1iIiIiIiIyM3k70JCbIOGRFepQ8qqebWxbt5Kk7Zs4dfQQ3v7F8CsehMWSt1+bzzgFscEpiEOSOHk7YTebEnHQ
ftwIucmX8Wt4U5YMMTb5k2ENYIIawRexp8JwYsZFzl8/DCZmElgil1VeNW1KlSh6DiQYUY7ZU8PT2Jiopy2Fa
9enWsVivHjx+nUaNG/6Pm6/6tdWYYPH079+vWZPn36Ff+vPn0x7cF9VKxT4aNLFvX7BqFU2/NMjTgH8Q7qsuw
MH27duvuB6Rv7qhbD3pKWpygNmNaIDVWNRN8lBWOSYrSS2bjRyb+ttcZS1t/6GW2hgpmKoRGo+KIx35kh7zn1Dhu+b
u76iz8iQQSyQiQmbp5GBLyw5sT6ovPKbRsaP4/n2LrvwyUDxLpwuLaxaqsgvlSjk5YuSpZzpbOcFzqcrTQS2WWP5
EIcPnqAvAtWYfbejPvxw0BLRwuvlMT5TqZbwsPzjh/hlfCMGD2fibSAp79cZCg8LgGs7zEx3DdFaWL2iCqG5eWF0UBDd
S2R6rzGqkzDXJPRkUDXMvWdbfPLwWQcgj6Vfq7HS9T3HMP2mxK9r4e3THb3HfSOOP+JQU0Dhx5HKvsMUHA+vfHP
VWiUbwDhqRR6bx+eqS2MgQkVgHaRyXX0dPA2WuDw4OL/Zb3fVqB6LFOlGI+y+ecX5B6nZmwwEomE1C++pQ7DTUF
zNpb7K4XTFkuwTPvLRW/jtOc8vcqQMYgAg21b4bdaHakXyWULKcq6hC0ZcA7BxA88uESOg5WwuGyYkR+xsXAtKWhvvi
r+vbrhi8GXjvfFfo4Gc4jH/w62ZpM/BFIlD7RN+Qg7v9xeOgkAHjZvnTE9ng70LMhf8BtaPyTu+zlgTE1mRFm
2i1a4n4ar8+/gCKd7/yRtMQI5iTvv5UmfTyK8bOr7K2Zu+Dg8KVG+ZpOJJ21zStv3hyJiz1FvBWJfZY17J+D6UIbuCAeWdV1JRhKuY+OJMRb5iQxBcNNbZB2jNC+6PmTOQNsNUt2hO0WLabijaOq3Y+MUbbLbC7Zo6t1XYu6RfS+aRexQDBZz8g6hfA6uCEZ9yEtHyvO0Fb5k5lrL0b4nZ3yQOzvG+M8hGujIEOVyvbYgvrGJVdH8I4Ty

```
cw0mgMW628wRn77cnQd2Mizfxim60220hI1+br703Z3Fbccx/m5XUMAz0pp3lx0kT5FhzK0ZXjBYNW1AtthoxZ
WJwy8fnLSKF44YmPcuUKUNwcDCLFy+2JznT0tL4/fff7T0469evz9mzZ9mwYQM1a9YE4Ndff8Vms1G3bt0bGY6
IiIiIiIjIDePl60/luk2oVKcxxw7uY+/2TWzfuJrjh/Zhzc7B08cXH//iu0Rx/s/LMsye7HOtwD7XCmAY+FuPU
yL7EMVzDlE85zCetvSbdEU3zq08p6fZMONn+FHMVoxitmIEWYPwwCPXsukX0jl+6jgXMi7g5+1H9QrVqV25NhX
KVcDP269gAxeRfyXfSc/09HT27Nljf5+cnExCQgIBAQGEh4czcOBAXnnlFcqVK0eZMmUYMWIEoaGh9iHwsbGxt
GzZkscee4wPP/yQ70xs+vfvz/3336+V20VERERERERERERO0SWZzKZCA4rQ3BYGWrHteHg3p3s372NPVvXc+JwCjnZ2Xh
4++DjXxxX99yTa9eonDN0QZxxCmInNQDwsKYRmHOY4jmHKZ59GH/rCczcWkPJb6Wkp4fNg+K24hQzLiU5A2wBW
Mh9Hlabzca58+c4k3qG8xfP4+7mTkTJC0pUqUPl6MoEBwZr+LrIbSrfSc/169cTFxdnfz9o0CAAHnroIaZ0ncp
zzz3H+fPn6d0nD2fPnuWuu+5i/vz5uLn92fX7q6++on///tx9992YzWY6d+7MpEmTbsDliIiIiIiIiIiBQcZxdXy
pSvQpnyVWgY35kj+3ezf/d2dm1Zy6mjB8n0ysTN0wsvH3/cvXwwm83Xr/RvLlh82G/xYb9reQAsRjZ+0Sfxs56
49Mo5iZ/1JK5Gxo2+vDzL29J0N56r4YqvzZditmKXEp22YlfMyfl3WdlZnEk9w9m0s1itVrw9vQkPCadK+SrER
sUSWSoSSx4XqxKRW9e/WsiosBTWxPVak0H0ooWM5GbTAjNSENT0pCBoISMRkSvlZGdzJGUPB/YkkrQ9gVPHDnH
x/DlMJj0ePn54+/rnexj89XhY0/CzXkqGelvP4GVNxct2Fnfb+ZveE/MHr3CeD7w5dTsbzngb3njbvC/99y//7
4LLdY83D0PP3pwXzm0xWCjmV4zYqFhiI2MpG15WPTpFiqAb0qeniIiIiIiIiIiFxaACmsbCxhZW0pf09HTh8/wpG
UJA7t3cn+XVs5eeQAWVmZuLi64enti4e3L84u/24B3wsWHy5YfDhMpMN2s5GDpy3tf0nQVLysZ/GwpeNqu4Cbc
QE32wVcjIx/lRg1Gwb8gzXonQ1nXA1XXHHF1XDFxXDBw/BwSG66kb9FgwzD4GLGRdL0p5F2Lo2cnBw8PTwJKRF
CtdhqRIVHUSasDJ7unvm0V0RuH0p6ioiIiIiIiNxEJp0JYkGhFAsKpVLtRmRnZXLsQDJHUpLYt3Mrxw/v5/jBf
eTkZG0x00Hu5Y2Hlw9uHl43pPehzeTE0UsA5ywBVy9kGLgaF3H7XyLUxZaBk5GNhZxL/zVyc0LSf/ce2I23tyf
+fn/Wd97JmzI5bpgxY8KEGTNm49L/0/0/x0bfkpuuuF6xavo/kZWdxbnz5zh3/hwXLl4AA9zd3PHx8qFu1bpUi
KpA2fCyhJYI/UfTC4jI7UlJTxEREREZEC50ziSqmy5SlVtjy149pwIf0cJ48e40SRgxw9sJdDe3eSdvokxw/
tx4QJFzd33Dw8cfPwxMN4+Yk7kwmMk0eZJo9SL100S80nME/LJLKEbUcttfLvvFh/Z3NZiP9Qjrnzp8j/UI6V
qsVJ4vTpXk5g80JLhNNqeBShAaFEhIYguu/7D0rIrcvJT1FRERERERECpGHlzfhURUIj6oAXJoP9My5ySw4coA
Thw9w5EASp48uJd3Czd3T1zdPXB2cSOyc1IahkF2TjYXMy5yZXMy8iGEYmDDh6
eGJn48f1W0rU7pkaUoGlSQ0KBRfL98icw9E5N9T0lNERERERETkFuLk7ExwyXCCS4ZfvTQ89K2rMwUk8d5+zJY5w
9dZwT/+sVej7tD0f0niI7Kx0TCQzDhJ0zMy6ubv97uePs6orFyfmGJQRNmPi3SyLbbDayc7LJzMrlYsZFzKZTNPbB
AYGBs50zri7uuPh7kGp0qUICQwhJDCEkoEl8ffxx2w23/D/CAsFQbd/vMyQg/AfhO4bG+4fcSfIYX7E8wVvR7+G
XQIBvAP6+/vj7+0PkpNSFi0SfnhwiIiIiIiIityGTyYSXrz9Xrz9evv5X7DMMg6zDMMiSyeN3uJzmSqsMAkmeC
bh4ngvpaaSnnuH0qfw/kqwXuY8wZzpyCpZ0g/3l7hwS+1gzmlwl1Wz8lmT7hA4/zxnyi7ktQVlsVXX7BAwFBnDzBA
emm7ulPAsgY+XD77evvh4+eDu6o6726WXh5sH7m7ueH144e/rr7k3ReSGU9JTREREREREGy3xIzwSPr9Lz6SdO
kWZ2Io88sLLRFWudkXZj0YNZeHML3l4+BjaPvTYVeu0Wq18896bB/+W86ePIF/iSDiOnbhPO8OtA8jnPnuBFb
+NJdTRw/j50xCZMXKdBs4j0iqNQDIzsrk/ReHsG7xAvyKl+CxUa9RtUFj+znmfPwuxw4d4P6wO0qO4nJhk5XStDN1
WTC1c0dVzd3CAq9ZlnDMLDZbFhzssnrJzr7035xs+3ubNQfjL+PZ22dex0zuicViwfK/nqCwX/nqCX/+vi4oKHmwcuzi6
aX1NECpWSniIiIIIInLHeH/EYFJ272TA+HcJKBHE8u+/ZczDXXnnx6UUCwqxl/t90c/s2ryBgBLB161zLzbGNkaBtiqMNP9e+0tA
J4aN5GwqBiS/tjMe88/g4eXN216PgpAaEQkj454laCw0mRlZDBv2se83PsB3lu4Gt+AX/s3baF12b8wKb
lv/L0kH58vmoLJp0JYwdT+0Wb6bz+7c837b6IiMid7XIPTYvFgourW2GpCZcZHfFv5Ez
yEvUrF2PUJKl6HrU0MIDo9gwddf2Mud0naET195kaffmIwlD30I7dy0ntp3x10zaXNKlAqjfsu2VG3YhD1bE+x
lGrXrRNUGjQk0K014uRh6DRvNhfRz7N+5HYCDe/dQq9k9hJeLoWX3XqSdPkXamdMAfDx6GD2GvICHl/eNvSEiI
iIiIkWYkp4iIiJyR7DlWLFZrTi70s4Z5uLmxo4Nay+VsdmA2vd+kvByMXqmN6Z6LbauWcnh5CQ9u3Yxo6
Na6neuFmu5b0zslg08//w8PYhonwFACJiKrBjw1oyMy6ssHIp/oFB+PgHsPyH2Ti7ul7etkiIiIncxD
W8XERGRO4K7lxcx1Wry3/ffoVRkOXQDTLyzXiHnHgEcGmousVioU2P3nmut20f/lw4f44BrRtjtliwWa1
0GziMxu060ZRbv2QRbw9+ksyLF/EPDGLU5GL097N/13YGtmmKt38Ag9/5/5kPTUs8yY9AYvffFpr8zn
lU/zSUorDT9XnvLYSi+iIiIihcSUlPERERuWMMeP1djJj+iMea1KJrlTL8+OVkGrVsj8l86a9lv/XqMsx4Z/n
Qr4UXVv/8PSt+mM3ACZMji4ohecc2prw2j/8dc/4lusOE70zjw2cqxDv
e8NH0jrHr1JTvyDtYvn8+acX5jz6WQ+e2UEz7376Q27JyIiIiRZGGt4uIiMgdIzg8gpf/zbZfbdzDx0vM37
WT+TkZBMUVprEDb+TeuokjzerzXVw7ivYhgnDh9k2vgxPNGsz3/H9r1eozZH
7/rUM7Nw40Q0mWIrlaTfq++hdnJicX//TrX0rf+tooDe3bRqvvD/PH7GmoObabhwcNWt3LtrVrbug9EREREE
pitTTU0RER044bh4euHl4kJ56loSVy+gx5EXq390aKvUb0ZR7+dFuNG7fmWYdu161rsYGLZjMjn9HNpstGDbjm
```

jEYNhvZWZlXbM/KzODTl5/n6Tfew2KxYLNZIedSXdac7EvvRURERETkmpTOFBERkTvGphVLAYPQMmU5uj+ZL95
4mZKRUTTr1BUnZ2e8/QMcylucnPAvXoKSkVH2baN7daFO85a0fvARAGrFteDbDycRGFLy0vD2xD/4YepHNOt8P
wAZFy7w7YcTqd3sHvwCgzh35jTzp0/h9LGj1G/Z7ooYZ73/DjUaNyOyQmUAyteozRdvvExcp678/NUUyteofVP
ujYiIiIhIUaKkp4iIiNwxLqSn8dVbYzl19Ahefn7Ua9Gabs8Mw8nZOc91HE3Zx7kzp+3vH33xFb6e9DofvzSct
FOn8C8RRIuuPbiv7zMAmC1mDiXvYemAWaSdOY23nz9RlavyylffXbFCfMquHaye/wNvfrfIvq1+fFu2rV3DiO4
dCS1TloETJv/LuyAiIiIiUvSZDMO49tirW1BaWhq+vr6kpqbi4+NTYOedvfNIgZ1LCl+nmMJZGVft7M5RWGOM1
M7uJGpnUhAKo50V1s+DIiIiInJ70EJGIiIiIiIiIiIiUqQo6SkiIiIiIiIiIiJFipKeIiIiIiIiIiIiUqQo6Sk
iIiIiIiIiIiJFipKeIiIiIiIiIiIiUqQo6SkiIiIiIiIiIiJFipKeIiIiIiIiIiIiUqQo6SkiIiIiIiIiIiJFi
pKeIiIiIiIiIiIiUqQo6SkiIiIiIiIiIiJFipKeIiIiIiIiIiIiUqQo6SkiIiIiIiIiIiJFipKeIiIiIiIiIiI
iUqQo6SkiIiIiIiIiIiJFipKeIiIiIiIiIiIiUqQo6SkiIiIiIiIiIiJFipKeIiIiIiIiIiIiUqQo6SkiIiIiI
iIiIiJFipKeIiIiIiIiIiIiUqQo6SkiIiIiIiIiIiJFipKeIiIiIiIiIiIiUqQo6SkiIiIiIiIiIiJFipKeIiI
iIiIiIiIiUqQo6SkiIiIiIiIiIiJFSqElPSdPnkxERARubm7UrVuXtWvXFlYoIiIiIiIiIiIiUoQUStJz5syZD
BoOiFGjRrFx40aqVq1KfHw8x48fL4xwREREREREpAgplKTnW2+9xWOPPcbDDz9MhQoV+PDDD/Hw8ODzzz8
vjHBERERERERESkCHEq6BNmZWWxYcMGhg8fbt9mNptp3rw5a9asyfWYzMxMMjMz7e9TU1MBSEtLu7nB/s2F9
HMFej4pXGlpnoVyXrWzOOdhtTFQO7uTqJ1JQSiMdnb55ODDMAr83CIiIiJy6yvwpOfJkyexWqOEBQU5bA8KCmL
Hjh25HjN27FjGjBlzxfawsLCbEqOIiIiI3B7OnTuHr69vYYchIiIiIreYAk96/hPDhw9nOKBB9vc2m43Tp09Tr
FgxTCZTIUZW9KWlpREWFsaBAwfw8fEp7HCkiFI7k4KgdiYFQe2s4BiGwblz5wgNDS3sUERERETkFlTgSc/ixYt
jsVg4duyYw/Zjx44RHByc6zGurq64uro6bPPz87tZIUoufHx89Mub3HRqZ1IQ1M6kIKidFQz18BQRERGRqynwh
YxcXFyoWbMmixcvtm+z2WwsXryY+vXrF3Q4IiIiIiIiIiIiUsQUyvD2QYMG8dBDD1GrVi3q1KnDO++8w/nz53n
44YcLIxwREREREREREREpQgol6dm1a1dOnDjByJEjOXrOKNWqVWP+/PlXLG4khc/V1ZVRoOZdMb2AyI2kdiYFQ
e1MCoLamYiIiIjIrcFkGIZR2EGIiIiIiIiIiIiI3CgFPqeniIiIiIiIiIiIyM2kpKeIiIiIiIiIiIgUKUp6ioi
IiIiIiIiISJGipKdc0+jRo6lWrdpV90+dOhU/P78CiOduTxEREbzzzjt5Ln87t6t9+/ZhMplISEgo7FDkL673L
BMREREREZGiRUnP29CHH36It7c3OTk59m3p6ek4OzvTtGlTh7JLly7FZDKRlJROU2Lp2rUru3btuil1FwVHjx7
lqaeeIjIyEldXV8LCwmjXrh2LFy8u7NAK1Lp16+jTp09hh1EgwsLCOHLkCJUqVcrzMb169aJDhw43L6hbRK9ev
TCZTJhMJlxcXIiKiuKll15yeJbdLEOGDLnjvndFwYkTJ3jyySDw/H1dWV4OBg4uPjWbVqlb2MyWRizpw5+a4
7v3+MKYrOxwARERERKcqcCjsAyb+4uDjSO9NZv3499erVA2DFihUEBwfz+++/k5GRgZubGwBLliwhPDycsmXL5
uschmFgtVqvW87d3R13d/f8X8QdYN++fTRs2BA/Pz/eeOMNKleuTHZ2NgsWLKBfv37s2LGjsEO8IbKzs3F2dr5
mmcDAwAKKpvBZLBaCg4MLO4xbVsuWLZkyZQqZmZn89NNP9OvXD2dnZ4YPH35Tz+vl5YWXl9dNPUdeZGVl4eLiU
thh3DY6d+5MVlYWO6ZNIzIykmPHjrF48WJOnTpV2KHdNGojIiIiI3hnp63oZiYmIICQlh6dKl9m1Lly6lffv
2lClTht9++81he1xcHJmZmQwYMIASJUrg5ubGXXfdxbp16xzKmUwmfv75Z2rWrImrqysrV6684txJSUlERkbSv
39/DMO4Yhjy5V4jX375JREEfj6+nL//fdz7tw5e5lz587RvXt3PD09CQkJ4e2336Zp06YMHDjwht6nwta3b19
MJhNr166lc+fOREdHU7FiRQYNGuTwGaWkpNC+fXu8vLzw8fGhS5cuHDt2zL7/8j39/PPPCQ8Px8vLi759+2K1W
nn99dcJDg6mRIkSvPrqqw7nN5lMfPTRR7Rt2xYPDw9iY2NZs2YNe/bsoWnTpnh6etKgQYMregHPnTuXGjVq4Ob
mRmRkJGPGjHHoiWcymfjggw+499578fTOtJ/3hx9+oHbt2ri5uVG8eHE6duxoP+bvPareeustKleujKenJ2FhY
fTt25f09PR83d+hQ4cSHR2Nh4cHkZGRjBgxguzsbPv+zZs3ExcXh7e3Nz4+PtSsWZP169cDsH//ftq1a4e/vz+
enp5UrFiRn376yX7ssmXLqFOnDq6uroSEhDBs2DCHe2Cz2Xj99deJiorC1dWV8PBw+334+/B2q9VK7969KVOmD
O7u7sTExDBx4kSHz3fatGnMnTvX3gvy8nf7wIEDdOnSBT8/PwICAmjfvj379u3L13261VzurVe6dGmefPJJmjd
vzvfff09mZiZDhgyhZMmSeHp6UrduXYdn3OVnzYIFC4iNjcXLy4uWLVty5MgRe5mlS5dSp04dPD098fPzo2HDh
uzfvx+4skebzWbjpZdeolSpUri6ulKtWjXmz59v33/5c5w9ezZxcXF4eHhQtWpV1qxZ43A9K1eupFGjRri7uxM
WFsaAAQM4f/68fX9ERAQvv/wyPXv2xMfH547p8XwjnD17lhUrVjB+/Hji4uIoXboOderUYfjw4dx7773ApfsL0
LFjROwmk/19UlIS7du3JygoCC8vL2rXrs0vv/xir7tp06bs37+fZ555xv69u+x6n+nfXetZk5f6cmsjDRoOYOj
QoQ7nOXHiBM7OzixfvhyAL7/8klq1auHt7U1wcDDdunXj+PHj9vKX/11fvHgxtWrVwsPDgwYNGrBz507g0ndqz
JgxbN682X4Ppk6dmo9PSERERETk1qak52OqLi6OJUuW2N8vWbKEpk2b0qRJE/v2ixcv8vvvvxMXF8dzzz3Ht99
+y7Rp09i4cSNRUVHEx8dz+vRph3qHDRvGuHHjSExMpEqVKg77tmzZwl133UW3bt147733HH5J/KukpCTmzJnDv
HnzmDdvHsuWLWPcuHH2/YMGDWLVqlV8//33LFq0iBUrVrBx48YbdWtuCadPn2b+/Pn069cPTO/PK/ZfThTbbDb
at2/P6dOnWbZsGYsWLWLv3r107drVoXxSUhI///wz8+fP5+uvv+azzz6jTZs2HDx4kGXLljF+/HhefPFFfv/9d
4fjLv8inZCQQPny5enWrRuPP/44w4cPZ/369RiGQf/+/e3lV6xYQc+ePXn66afZvn07H33OEVOnTrOioTp69Gg
6duzI1q1beeSRR/jxxx/p2LEjrVu3ZtOmTSxevJg6depc9f6YzWYmTZrEtm3bmDZtGr/++ivPPfdcvu6xt7c3U
6dOZfv27UycOJFPPvmEt99+276/e/fulCpVinXr1rFhwwaGDRtm75Har18/MjMzWb58OVu3bmX8+PH2XoCHDh2

idevW1K5dm82bN/PBBx/w2Wef8corr9jrHj58OOPGjWPEiBFs376d6dOnExQUlGucNpuNUqVKMWvWLLZv387Ik
SN5/vnn+eabb4BLw667d0liT+AdOXKEBg0akJ2dTXx8PN7e3qxYsYJVq1bZE31ZWVn5ule3Mnd3d7Kysujfvz9
r1qxhxowZbNmyhfvuu4+WLVuye/due9kLFy4wYcIEvvzyS5YvX05KSgpDhgwBICcnhw4dOtCkSRO2bNnCmjVr6
NOnz1WfUxMnTuTNN99kwoQJbNmyhfj4eO69916H8wG88MILDBkyhISEBKKjo3nggQfsCfCkpCRatmxJ586d2bJ
lCzNnzmTlypb0COR4ROPj+mkEZr6T2ZOh+9ooNpZCY8uH3vtzXV28iK8yjwbxfUC6PjjW3x3MTe+9SjPpjrB/U
lCzNnzmTlypUO3ymACRMmULVqVTZt2sSIESNu50Or0i73zp0zZw6ZmZm5lrn8x7spU6Zw5MgR+/v09HRat27N4
sWL2bRpEy1btqRdu3akpKQAMHv2bEqVKsVLL71k/95B3j/Tv7rWs+aftpHu3bszY8YMDMOwl5k5cyahoaE0atQ
IuNTL/uWXX2bz5s3MmTOHffv20atXryvie+GFF3jzzTdZv349Tk5OPPPLII8Cl6WkGDx5MxYoV7ffg7//2iIiIi
Ijc1gy5LX3yySeGp6enkZ2dbaSlpRlOTk7G8ePHjenTpxuNGzc2DMMwFi9ebADGvn37DGdnZ+Orr76yH5+V1WW
EhoYar7/+umEYhrFkyRIDMObMmeNwnlGjRhlVq1Y1Vq1aZfj7+xsTJkxw2D9lyhnTD19fXobyHh4eRlpZm3/bss
88adevWNQzDMNLSOgxnZ2dj1qxZ9v1nz541PDw8jKeffvqG3Jtbwe+//24AxuzZs69ZbuHCVnbuFYjFSUlLs27Z
t22YAxtq1aw3DyP2exsfHGxEREYbVarVvi4mJMcaOHWt/Dxgvvvvi4mJMcaOHWt/Dxgvvvvii/f2aNWsMwPjss8/s277++mvDzc3N/v7uu
+82XnvtNYcYv/zySyMkJMSh3oEDBzqUqV+/vtG9e/erXmfp0qWRQrVsz+/u/tKi/eeOMNo2b
Nmvb33t7extSpU3MtW7lyZWPO6NG57nv++eeNmJgYw2az2bdNnjzZ8PLyMqxWq5GWlma4uroan3zySa7HJycnG
4CxadOmq8bar18/o3Pnzvb3Dz30kNG+fXuHMl9++eUVcWRmZhru7u7GggULUrlr3reyv12mz2YxFixYZrq6uRq9
evQyLxWIcOnTIofzdd99tDB8+3DCMS20CMPbs2WPfP3nyChLly7N9dyXn2WXhYaGGq+++
qpDmdq1axt9+/Y1DOPPz/HTTz+177/83UxMTDQBz69+5t9OnTx6GOFStWGaz2bh48aJhGJfafocHfJOf+R
K//3vfw1/f3/Dzc3NaNCggTF8+HBj8+bNDmUA47vvvvrtuXRUrVjTeffdd+/vcnkt5/Uz/aRo4fP2440
TkZy5cvt2+rX7++MXToKte37p16wzAOHfunGEYf/67/ssvv9+9jL/PjjjwZgP/ffvxciIiIiIkWJenreppo2bcr
58+dZt24dK1asID6msDAQJo0aULVa9aV5wrJSWFFF
iaMHLkSAYPHnzd2CiIivD29ra/DwkJ4fjx4wD8/PMP+nrf19fYmJicn3PbiVGX/pnXMtiYmJhIWFERYWZt9
WoUIF/Pz8OA+z9tj216GNgENv3cs9ESRxrywLSMjg7SONODSMM2XXnrJ3sPKy8uLxx57jCNHjtDU/YObWUHErzQoX0NzTn3
CNHjnDhwgX7cX9vJJwkJCdx99GAAX375buvvpuSJUvi7e3NZUN0YgG9+ZGevPk7OcKPQOBgvLy8ePH
FF+29u0BSj+JHH32U5s2bM27c0Idh/AMGDOCVV16hYc0GjBo1iii3JSYmUr9+fSYRWx8+fYceg2
JhIZmZmvq538uTJ1KxZk8DAQLy8vPj44485ePDgNc+xefNm9u/fj4eHh8N3IiMjg6SkJHr16sVvv/3GmjVrAEhLS+KfXq1cvpLoRatzO1ebQ5HOt4OWaVfJQ6YzOPjsmq4oJGiCIAA9NxoQJ9qXDtNH5dC/E2ugsVTkzrNb
NOnTq8yr5TwZj1uWSyjtSMZ1tfFIiIsYcn8nxUlRn8Nmv83etHnDOPWvt16eWzuJoxyLvIEaS1ORgWXKAXhNoQLKc4jSHAJdEOhQUv+uEH3wIK6DUwSn2qu92CLMUvmO80oiC03b0+cp3ktSb3cJYLwJrtIySiIFTjtqlo0tEOAb3cBLMSvwZO6Y0VH5cZ2qGtC6CVuR40ch1kxjWHW5dC6Q2MlqQy6zyOoWvlyyGR9XwvHlmu3XcStDwSTCAOrkyBUUQRdXwXBbyN3oPePtDBF8MQLBYvBHDMwCYRSYNrfsCDGvAYh4XBNeH2SOQoMNjN6N/d+r0BNyYVgHGg/j1j+YDPeUE/tQ0UU79Jm8k8P0SDEEuHu5VzDqvz2w3hZNI/ebrFcUjoSjJkQGsTPcTuRH9WNVZH1a7TEYsm6oNo0CL9nydg/7TPR7sG/uqX9oEN5ZX2E3u/qvn9PE8TYa

```
C9enPbt27NixQqSk5NZunQpAwYM4ODBg/m6X7e66OhounfvTs+ePZk9ezbJycmsXbuWsWPH8uOPP+apjuTkZIY
PH86aNWvYv38/CxcuZPfu3fbP9e+effZZxo8fz8yZM9m5cyfDhg0jISGBp59+Os9xDx06lNWrV9O/f39779W5c
+dec9EbybtTp07RrFkz/u///o8tW7aQnJzMrFmzeP3112nfvr29XEREBIsXL+bo0aOcOXMGuPS9mz17tr1nZbd
u3a7o3RgREcHy5cs5d0gQJ0+eBPL/mV7vWfNv2oinpycd0nRgxIgRJCYm8sADD9j3hYeH4+LiYn+Ofv/997z88
sv5u8H/uwfJyckkJCRw8uTJqy4YJSIiIiJy01LS8zYWFxfHxYsXiYqKclg5ukmTJpw7d46YmBj7/F3jxo2jc+f
O9OjRgxo1arBnzx4WLFiAv79/ns/n5eXFzz//jGEYtGnTJtcejHnx1lUb9+fdq2bUvz5s1p2LAhsbGxDsOni
4LIyEg2btxIXFwcgwcPpplKlSrRo0YLFixfzwQcfAJeGGs6d0xd/f38aN25M8+bNiYyMZObnMmYUSc3x8PPPmzWP
hwoXUrl2bevXq8fbbb106d0lrHte0aVNmzZrF999/T7Vq1WjWrBlr167NtWzVqlV56623GD9+PJUqVeKrr75i7
Nix+Yrz3nvv5ZlnnqF///5Uq1aN1atXO6yFD179iQ6OpouXbrQqlUrxowZA4DVaqVfv37ExsbSsmV
Loq0jef/99wEoWbIkP/3OE2vXrqVq1ao88cQT907d2yHxO2LECAYPHszIkSOJjY2la9euV+1l9fjjj90pUye6d
u1K3bp1OXXqFH379nUo89hjjxET0tWrUIDAxk1apVeHh4sHz5csLDw+nUqR0xsbH07t2bjIwMfHx88nW/bgd
TpkyhZ8+eDB48mJiYGDp06MC6desIDw/P0/EeHh7s2LGDzp07Ex0dTZ8+fejXrx+PP/54ruHDBjAoEGDGDx4M
JUrV2b+/Pl8//33lCtXLs8xV6lShWXLlrFr1y4aNWpE9erVGTlyJKGhoXmuQ67Oy8uLunXr8vbbb9904cWMqVar
EiBEje0yxx3jvvffs5d5880WLVpEWFgY1atXBy79l+Pv70+OEDBg1o164d8fHx1KhRw6H+l156iX3791G2bFkCA
wOB/H+m13vW/Ns2Or17dzZv3kyjRo0cvguBgYFMnTqVWbNmUaFCBcaNG8eECR/L9BvuR4Q7H8+eCRPyfnP/p3PnzrRs2ZK4uDgCAwP
5+uuv812HiIiMitymTkdcUVkZvk/PnzlCxZkjfffDPPPU9FREREREREREREESuxqmwA5A7z6ZNm9ixYwd16tQhN
TWVl156CcBhuKKIiIiIiMg/paSnFIoJEyawc+dOXFxcqFmzJitWrKB48eLEx8c7rBD+vzIzMxk7diw8++yw

C9XfwXyWTWt6sMa4s+7hI0

```

```
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 507
        },
        "outputId": "12f9b25f-bc23-4422-a097-81895651ccd4"
      },
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 1400x500 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAABQ0AAAHqCAYAAACuimiwAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAADlZ0l
EQVR4nOzdd3hUZfrG8e+U9J6QCgFC71JFR0kKK/qzsIsoFhTFVVGxYUfX1bU31F0bCip2EcWCBem919ADBNJ7n
WQyM78/0NExQSmTnJT7c125J0e855xnYgLJnfd9H5PL5XIhIiIiIiIiIi8guz0QWIiIiIiIiIhI/aLQUER
ERERERERERERDwoNBQRERERERERERERERPCg1FRERERERERERETEg0JDERERERERERERER8aDQUERERERERERERDwoNBQRE
RERERERERERERPCg1FRERERERERERETEg9XoAmqL0+kkLS2NkJAQTCaT0eWIiIiiADgcrkoLi4mmISEBs1m/wxREZH
6qdGGhmlpaSQmJioMRhbdhoiIiIjVJTU2nRooXRZYiIiIjUqNGGhiEhIcDRb8ZCQ0MNrkZERE5KiioiSSExPd3
6+KiIiI1EeNNjT8dUlyaGIQkMRERERqXe0hY6IiIjUZ9pERUREERERERERERERERERDwoNBQRERERERERERERERERETEQ6Pd01BERE
ERERETEQ6Pd01BERETEGxw0B3a73egypAHx8fYHBQUFqdiIIlIlIlIDl8tFRkYXYDBQUFRpciDVB4e
DhxcXFqdiIiINlkJDERERkRr8GhjGxMQ0i8i8vloqysjKysLADi4+MNrkhERETk5Cg0FBEREfkDh8P
hDgyjoqKMLkMkcamICAAAAyCsrKIiYnRUmURERERFpkNQIRUREREREESdDN6dOkHw
0QYtCRZTpY+d0RERKShU2goIiIiIiIiiHhQaioiIiIiEiDN2TIEKZrmWPGy2Yy+AiiIiIiIiJyA0bvS6/R
5l3Q8ue67K1eu5KyzzmLFUqFF88803Xq5KREREREREERERBo7zTQUERERERaYRmzJjBLbfcwpnzQVEtb6MORER
ESk7ig0FBEREWlkSkpK+Pjjjj7nxxhxZSPpK+PjjjnxxhsZPXo0M2f0JbtGgRJp0Jb775hh49euDv2Dv788GNmzb3b3G
NmzpxJeHg4c+f0pX379vj7/2/++tf/+7y62tX1Va2s+vs7drgAn+3N5EeTdXCU4VH0xtTHiNlc+wwPHBRPhHJvk5P/mv






































Vv1Vt27YtPj4+rF69ulr9gwcPPrF7ZPPLK8GLs9aWDdRsGRp+A2NsHKdCcMqC9WLecLF6Zb8DzjCL3zgHpXULHcHgGqSb









































































































































```

```
Up+fj7h4eE1jpk5cyZTpkyhoKCgTmtrTP7sc8jo71NFREREjodmGoqIiIiIiIiIiIgHhYYiIiIiIiIiIiLiQaG
hiIiISBMyZMgQXC7XMZcmAOyYMEFLkOVERESaOIWGIiIiIiIiIiIi4kGhoYiIiIiIiIiIiHhQaCgiIiIiIiIiI
iIeFBqKiIiIiIiIiIiIB4WGIiIiIiIiIiIi4kGhoYiIiIjISWjdujUvvviiOWWIiIiI1AqFhiIiIiKNyIQJEzC
ZTDz55JMex+fOnYvJZDqheykUExEREWm6rEYXICIiItKQbNu2rU6f161btx+xt/fn6eeeeoobbbriBiIiIWqiqf
qisrMTX19foMkRERREQaJcOOFBEREWlkRowYQVxcHE888888cSfjvv888/p2rUrfn5+tG7dmueee859bsiQIRw8eJD
bb78dk8n0p7MUn3/+ebp3705QUBCJiYncdNNNlJSUuM/PnDmT8x9fkyJGkpqa6xzzzyyCP07
NmT119/ncTERAIDAxk7diyFhYXXuM4dOwKwdetWhg0bRkBAAFFRUUyaNMn9/B9++AF
/f38KCgo8ar7tttsYNmyY+/1ly5Zx9tlnExAQQQGJiIrfeeiulpaXu81lZWVxwwQUEBASQlJTE7Nmz//RjKyIiI
tLQnXBouGTJEi644AISEhIwmUzMnTvX47XLatGnEx8cTEBDAiBEj2LNnj8eYvLw8o8foKOHh4UycONH
jGOuALVu2cPbZZ+Pv709iYiJPP/30ib86RkSbIYrHwn//8h5dffpnDhw/XOGb9+vWMHTuPGsXXrVh555
BEeeughZs6cCcCcOXNo0aIFjz76K0np6aSnpx/zeWazmzp7N9+3ZmzZrFzz//zNSpUz3GlJWV8fjjj/Puu++
yfPlyCgoKGDDdunMeYvXv38sknnzBv3jzmz5/PxoObmmmzzGLFiwgF27dvHjjz/y9ddfU1paysRI4mIiGDt2
rV8+umn/PTTTOyePBmA4cOHEx4ezuef+6+h8Ph4OOPP2b8+PEA7/
BZWpqKgsXLuSzzz7jv//9L1lZWX/xf0JERESk4TrhOLCOtJTTTjuNV199tcbzTz/9NNOnT+e1115j9erVBAUM
XLkSGw2m3vM+PHj2b59u/ubvSVLljjT3+aKiIs4991xatWrF+vm88vSCkn8GJh+ZfqtWyptJcuhQNG5m3
8MT179uThhx+u8fzzzzz/P8OHDeeihh+jQoMTJkxg8uTYz5Ye/eFHLXLV1TyzDAEjsEBDBsHTtcEDV+/Ii+cP
NatWzNs2DAee+wxPvnkE48xdrudV155hf3hQEDBtCnTx9mzZrFihUrFKYGXMtKDIzW65Nf+4kSpd1ZoXWy2
EWmCSUrjS8bHOGLJgeBS4g1NtemSTYRSsXNMjeuTrgbMpm5qrSxSR9tz0SnRiTsb/wEKFxgBZ3bJ
ERTuKMIjV40lUCvlNHOqVdQTjh/tzKgIOrbkoKDJMPeLqK8VzLx5IdKmenQpCVqpHDBgG51
jTSAZvLCvuvCHj1lBErfKDIQuY7Y2bM4mYQwEnp7f8Ruw79jc03nFqjKV9BcGZYUODk54G3rkLmbsAAkzY7jTVX
HDvQ4NnDgQPbs2YPD4Tih5/z008MHz6c5s2bExISwpVXXklubi5lZWXuMVarlX79+rnf79SpE+Hh4R61tWzZk
ubNm7vfHzBgAE6nk127drmPde/e3WMfw+TkZE477TSCgoI8Xsfvrxs/fjyLFi3y+F5z9OjRhIeAA7B582Zmzpx
JcHCw+23kyJE4nU5SU1lJITk7GarXSp0+favWLiIiINFZe3dMwJSWFjIwMRowY4T4WFhZG//79SXnVgc9CsFq
Dycvn37useMGDECs9nM6tWr3WMGDRrk8Q8jhyJEj2bVrF/n+PUzm+TU+u6KigqqKiIo83ERERkaZs0KBBjBw5kvvv
WnnHgwAHOP/98evToweeff8769et59dXXXfddK1IqKyu9/frfh4PHq1+/frRt25lPc2fz73ed44b0p6mGGqkc
m3a5H7bvHkze/bsoW3Gr35F+XjsTGxnocj42NdZ/LyUJ1Je7x67m
augA+8cQT/0tf//LOCxERERFpJJ588kl69uxJp8lC7hmmvq5dZr+mKJpNVuPhJmE4Cii9MsUVI5ID1LEkPvb/5Mb2mw
vf8fPz8997p577mHFihVMnjyZTs2sWfPHr788kt3I5SPgY4Dq+/rZZt9i56p0M0qwCQbMdG1BAEBAX/x0RcRERFpuLwaG
v76DV1mZqbH8czMTPe5ulio4ap3mqqqqyMvL8xhTr7Oz+/4w/8vPzIzQ0101ONNRERERODRRx/F6XR6HOvduzeffPI
JH33O7EzK8y+7tbkHs0ssrO+bsu0qUjujoOhGFKKqiiD2nfdTOPDSmnrQw28Sgg2bzzzoPcKJz+ahdcvKVOmnl/MwzeTQLyo2iHrCb
889XH755QwOJDg4GA+/vhjjjzHt2rXjkksu4bzzzuPcc8+1R48e/Pe///3T1hq3T1xYYGMj3339PX+14e/fr14+9//zv
DEe++yEhYXxww8/cPPNN90nOTx+aNWvGtGnncbtDRp0qmVVhHERERERERREZEGz2Z3sC+7hL1ZJezLmFvdg17Mks4mFt
GpcP51zeoZVVOF8UVVRRXVHGKoBwoqHGcyQRRb6/hIgBJDULpGNcKJ3iQmgXEE4y/j6VO6xYRT6eOPLk+M3rZh
5Ynl19aniwiIn9Fy5Pl1VGl5soh4Q0lFFbsyitmXVcKerGHuL2/BsVKVGcJTdnWW6EcQAVQp8eu6MIehrL4GQ2HIggWqQhQQYIOqaEBL1WEFqPv7NDh91C7rGbmQodtDOwQfPhvVdIpgNx
```

6xYXQMjIQs9lkdHkiTcIJzzQUEREREREREe9LzStj/cF81h3MY/3BAnZlFDX6cPBYHE4X+7NL2Z9dyrdbM9zHA
3wsdIgNplNcKH1aR9A/KZJWUUEGVirSeCk0FBERERERERaljVQ4n29OKWHcwnw2/BIWZRRVGl1XvldsdbD5cyOb
DhXy8LhWA+DB/Tk+K5PSkSPonRdEuJtjgKkUaB4WGIiIiIsfQSHdxkTqgzx1p7Fq3bs2UKVOYMmXKSd+jsXVN/
ysOp4sNh/JZsjubtQfy2JxaSLndYXRZjUJ6oYOvN6Xx5aYOAJoF+3F6UgT9k6Lo3yaSjrEhmExa0ixyohQaioi
IiPyBj48PAGVlZQQEBBhcjTREZWVlwG+fSyK1YcKECcyaNYsbbriB1157zePczTffzH//+193w0pvW7t2LUFBW
hL6V4ptdpbszmFBciYLd2WRX2Y3uqQmIaekgm+3ZriXNYcH+jCwXTNGdo1jaMdoQvz1d7PI8VBoKCIiIvIHFou
F8PBwsrKyAAgMDNQMBTkuLpeLsrIysrKyCA8Px2JR50+pXYmJiXz0OUe88MIL7l9y2Gw2PvjgA1q2bHlK97bb7
dWC78rKSnx9fYmOjj6lezdmh3LL+Ck5kwU7M1mTkofdoZnHRisos/PNlnS+2ZKOr8XMme2iGNk1jnO6xNIs2M/
o8kTqLYWGIiIiIjWIi4sDcAeHIiciPDzc/TkkUpt69+7Nvn37mDNnDuPHjwdgzpw5tGzZkqSkJPe4+fPn89hjj
7Ft2zYsFgsDBgzgpZdeom3btgAcOHCApKQkPvroI/773/+yevVqXnvtNRYtWkRBQQH9+vXj1Vdfxc/Pj5SUlGr
LkwsKCrjrrrv48ssvqaiooG/fvrzwwgucdtpp7hqefPJJXnjhBcrKyhg7dmyjCR6dvyw7/ik5iwXJmezJKjG6J
PkTlQ4ni3Zls2hXNg98sZU+rSIY2TWOkV3jSIwMNLo8kXpFoaGIiIhIDUwmE/Hx8cTExGC3azmZHD8fHx/NMJQ
6de211/LOO++4Q803336ba665hkWLFrnHlJaWcscdd9CjRw9KSkqYNm0aF198MZs2bcJsNrvH3XvvvTz33HP06
tULf39/Fi1axIIFCwgNDeXHH388Zg3/+Mc/CAgI4LvvviMsLIzXX3+d4c0Hs3v3biIjI/nkk0945JFHePXVVzn
rrLN47733mD59m3atKm1jOtt25ddwmfrD/PFhiNkFNmMLkd0gtMFaw/ks/ZAPo99k0zn+FBGdo1lldPd42seGG
F2eiOEUGoqIiIj8CYvFogBIROq1K664gvvuu4+DBw8CsHz5cj766COPOHDMmDEe17z99ttEROezY8cOunXr5j4
+ZcoULrnkEo+xQUFBvPXWW/j6+tb4/GXLlrFmzRqysrLw8zu61PPZZ5917ty5fPbZZOyaNIkXX3yRiRMnMnHiR
AAee+wxfvrpJ2y2hhW2FZbbmbc5jc/WH2ZTaoHR5YiXJacXkZxexIs/7eGOxHDG9Uvk/O5LIMhPOYkOTfrMFxE
RERERacCio6MZPXoOM2fOxOVyMXr0aJo1a+YxZs+ePUybNo3Vq1eTk5ODO+kE4NChQx6hYd++favdv3v37scMD
AE2b95MSUkJUVFRHsfLy8vZt28fAMnJyfzzn//OOD9gwAAWLlx4Yi/WAE6niyV7svls/WF+3JFJRZXT6JKkDmx
OLWBzagGPfb2D83skcOnpifRuGWFOWSJ1SqGhiIiIiIhIA3fttdcyefJkAF599dVq5y+44AJatWrFm2++SUJCA
k6nk27dulFZWekxrqaOyH/VJbmkpIT4+HiPmY2/Cg8PP/4XUc/szTq6/HjuRiO/bspKKx18vC6Vj9elOjE2hLH
9EhnTuznhgccOOkUaC4WGIiIiIiIiDdyoUaOorKzEZDIxcuRIj3O5ubns2rWLN998k7PPPhs4uqTYW3r37k1GR
gZWq5XWrVvXOKZz586sXr2aq666yn1s1apVXqvBW1wuFz8lZ/HmOv2sSckzuhypZ3ZlFvPvr3fw1PydjOwax7h
+iZzZNgqTyWROaSK1QqGhiIiIiIhIA2exWEhOTnb/+fciIiKIiorijTfeID4+nkOHDnHvvfd67dkjRoxgwIABX
HTRRTz99NN06NCBtLQOvvnmGy6++GL69u3LbbfdxoQJE+jbty8DBw5k9uzZbN++vd40QrHZHXy+4TAzlqWwP7v
U6HKknquscjJvcxrzNqeR1CyI689uw5g+zfGzag9kaVwUGoqIiIiIiDQCoaGhNR43m8189NFH3HrrrXTr1o2OH
Tsyffp0hgwZ4pXnmkwmvv32Wx544AGueYasr0ziYuLY9CgQcTGxgJw6aWXsm/fPqZ0nYrNZmPMmDHceOONfP/
9916p4WTllVby7soDvLfyILmllX99gcgfp0SUcv8XW3nxp91MPCuJ8WeO0IliNU6SRMLlcLpfRRdSGoqIiwsLCK
CwsPOY/nrVp27Ztdf5MOT6/3+hZREREpK4Z/X2qiMD+7BLeWpbCnA2HsdnV2ES8J9TfypUDWnHtwCSigv2MLkf
klCj+FhEREREkSZh7YE83liynwXJmTgb5fQZMVqRrYpXF+5jxrIUxvZNZNKgNrSICDS6LJGTotBQREREREREG
rVNqQU89d1OVu7PNboUaSJsdifvrjzIB6sPccFpCdw4pCOdYkOMLkvkhCgOFBEREREREkUZpX3YJz36/i++2ZRh
dijRRVU4XX2w8wtxNRxjZJY67R3WkbXSwOWWJHBeFhiIiIiIiItKoZBbZePGn3Xy67jBVWocs9YDLBfO3Z/Bjc
iZj+7bg9hEdiAn1N7oskT+lOFBEREREREQahcJyO/9btI+ZK1LU4ETqJYfTxYdrUpm7MY1rBrbmnOPaEurvY3R
ZIjVSaFhLdvtEGV2CHIN6J4uIiIiINC42u4OZKw7wvOX7KCy3G12OyF8qtzv476J9fLQ2ldtHtOfy/q2wmE1Gl
yXiQaGhiIiIiIiINEgul4tP1x/m+R92k1FkM7ockROWV1rJQ19u571VB3lwdBcGdYg2uiQRN4WGIiIiIiIiOuD
sSCviwblb2XCowOhSRE7Z7swSrnp7DUM7RvPg+V3ULEXqBYWGIiIiIiIiOmAU2+w8/+Nu3l15EIeanEgjs3BXN
sv2LuHGIe2YPLQdvlazOSVJE6bQUEREREREREBqErzan8djXO8gqrjC6FJFaY3e4mL5gD99tTefJMT3oOyrC6JK
kiVJkLSIiIiIiIvXakYJyrnlnDbd+uFGBoTQZe7JK+MdrK3jkq+2UVVYZXY40QQoNRUREREREpF5yOl28szyFc
55fzMJd2UaXI1LnnC6YueIA5zy/hMW79TUgdUvLkOVEREREREKTe2Z1ZzD2fbWFjaoHRpYgY7khBOVe/vYZLejd
n2vldCA/ONbokaQIO01BEREREREETqDZfLxYxlKZw/fakCQ5E/mLPhCCOeX8y8zWlGlyJNgEJDEREREREREqRdyS
iq4+u01/PvrHVQ61BlZpCY5JZXc8uFGrn93HXmllUaXI42YQkMREREREREEx3OLd2Zzz3CKW7MkxuhSRBuHHHZm
c99JS1qTkGV2KNFIKDUVEREREREMQwlVVOHp67lQlvryG/XB1iRU5ERpGNy95cxcsL9uBOanaueJcaoYiIiIiIi
Igh9meXMGnWavbm2IwuRaTBcjhdPPfjblan5PHCpT2JDvEzuiRpJDTTUEREREREROrcB6sO8LcXFyswFPGSZXt
zOG/6Ulbs1RJ/8Q6FhiIiIiIiIlJnCsvtTHx7JffP3U6Fw+hqRBqX7OIKrpixmud/2IVDy5XlFCkOFBEREREk
TqxN6uYUc//zILdatwgUlucLpj+814uf3MVmUWaySsnT6GhiIiIiIiI1Lrvtxzm/JeWkF6sZicidWF1Sh7nvaT
lynLyFBqKiIiIiIhIrXpm3kb++cEmbFqOLFKncksruertNXyO5pDRpUgDpO7JIiIiIiIiUitslXauf3MxS1MrA
JPR5YgOSVVOF/fO2UpKTin3/qOTJpO+FuX4aKahiIiIiIiIeN3BzHzOefK7XwJDETHa6Ov2c+P7Gyiv1JRfOT4
KDUVEREREREMSrlmw7wOjpSOgtsxhdioj8zvztGVz6xkqy1CBFjoNCQxEREREREfGaGT9u4tr3t1LiOG5YIvXRl

sOFXPTqcpLTi4wuReo5hYYiIiIiIiJyylwuF/e+t5h/LzhMlX7UFKnX0gpt/OO1lSzclWV0KVKP6W9yERERERE
ROSUVFZVMfPU7PtpeghqeiDQMJRVVXDdrHbNWHDC6FKmnFBqKiIiIiIjIScsvLOKyF7/h58MuoOsRkRPkcLp4+
KvtPPndTqNLkXpIm0yIiIiIiIjISTlwOI3rZyxlT3mw0aWIyCl4bfE+KqucTLugi9GlSD2iOFBERERERERO2Lq
tO5nyOUYOOOKNLkVEvODt5SnYHU4evbArJpO2GRCFhiIiIiIiInKCFq7awNS5O8kmzOhSRMSL3lt1ELvDyX8u7
o7ZrOCwqdOehiIiIiIiInJcXC4X3y5ezR1f7FJgKNJIfbQ2lbs+24zTqX1KmzrNNBQREREREZG/5HK5+PyHZTy
6IIOis5YkizRmczYcocrh4vmxp2G1aL5ZU6XQUERERERERP6U0+nkg28W8sTSXErNanoiOhR8tTkNu8PJ9Mt64
aPgsEnS/3UREREREREE5pqqqKt7/8geeXJJDqTnI6HJEpA59ty2DG9/fQGWV0+hSxAAKDUVERERERKRGdnsV73/
xPc+tyKXEohmGIk3RT8mZ3PDeOuwOBYdNjUJDERERERERRqcZWUcHMT+fx0uocCi3hRpcjIgZauCubqZ9tweVSc
5SmRKGhiIiIiIiIeCgrt/HOx1/yxro88q3NjC5HROqBLzYe4cnvdhpdhtQhhYYiIiIiIiLiZquo4J1PvuTdjXl
k+8YZXY6I1COvL9nPjGUpRpchdUShoYiIiIiIiABH9zCcPedbPtqYTbpfotHliEg99Ng3O5i3Oc3oMqQOKDQUE
RERERERHA4Hn37zA7NXHyTVP8nockSknnK54M5PNrNib47RpUgtU2goIiIiIiLSxLlcLr76cTGzFiez37+dOeW
ISD1X6XByw3vr2Z5WaHQpUosUGoqIiIiIiDRhLpeL7xev4J3v17HHrwNgMrokEWkAiiuqmPDOWlLzyowuRWqJQ
kMREREREZEmbOmaDbw7bxE7/Tri1I+IInICsosruOrtNeSWVBhditQC/YsgIiIiIiLSRK3ZtI1353zLVmsHKrE
aXY6INEApOaVc9+46KqucRpciXqbQUEREREREpAnaunMPsz79ii2uVhSbAoOuROQasI2HCvjXvO1GlyFeptBQR
EREESkidl74BBvfzyXHbZwssxRRpcjIo3A7NWH+HRdqtFliBcpNBQREREREWlCsnLzePvjuezKd3HAp6XR5Yh
II/Lg3G1sO6KOyo2FQkMREREREZEmotxm473Pv2Z7ai57AzrjUqdkEfGiiion/3x/PQVllUaXIl6gOFBERERER
KQJcDqdfPbNT6zclExKaE8qXfpxUES873B+Obd8uBGnO2V0KXKK9K+EiIiIiIhIE/Dz8jX8sGQlRyJOo8jpa3Q
5ItKILd2Tw/M/7ja6DDlFCg1FRERERERQaua079/DZNz+SHphEuiPE6HJEpAl4ddFeftyRaXQZcgoUGoqIiIiIi
DRiaZnZvPf516RX+LLPFWdOOSLSRLhccMcnm0jJKTW6FDlJCg1FREREREQaqZLSMt797CtSOrLZF9hJjU9EpE4
V26r453vrsdkdRpciJ8HroaHD4eChhx4iKSmJgIAA2rZty7///W9crt82wHS5XEybNo34+HgCAgIYMWIEe/bs8
bhPXl4e48ePJzQ0lPDwcCZOnEhJSYm3yxUREREREWmUHA4HH3/1PRu37yK32WmUOKxGlyQiTdCuzGKemr/T6DL
kJHg9NHzqqaf43//+xyuvvEJycjJPPfUUTz/9NC+//LJ7zNNPP8306dN57bXXWL16NUFBQYwcORKbzeYeM378e
LZv386PP/7I119/zZIlS5g0aZK3yxUREREREWmUvl+8goUr12COac/+imCjyxGRJmzmigOs2JdjdBlygkyu308
B9ILzzz+f2NhYZsyY4T42ZswYAgICeP/993G5XCQkJHDnnXdy1113AVBYWEhsbCwzZ85k3LhxJCcn06VLF9auX
UvfvnOBmD9/Pueddx6HDx8mISHhL+soKioiLCyMwsJCQkNDvfkSj8ucXel1/kw5Ppd0jDe6BBEREWnCjP4+VZq
Gbbv2Mv3tD3BYA1jm7ESFSztTiYixmocH8P3tgwj2O6znhsLr/3KceeaZLFiwgN27j7bW3rx5M8uWLeNvf/sbA
CkpKWRkZDBixAj3NWFhYfTv35+VK1cCsHLlSsLDw92BIcCIESMwm82sXr26xudWFRQVFTk8SYiIiIitLUFBQ
V8/FX31NmqyDZ0laBoYjUC0cKyvn3vB1GlyEnw0vx7r333ktRURGd0nXCYrHgcDh4/PHHGT9+PAAZGRkAxMbGe
lwXGxvrPpeRkUFMTIxnoVYrkZGR7jF/9MQTT/Cvf/3L2y9HRERERESkwXA6nXzx3c/sSTmII6EHaSV+RpckIuL
28bpUzu0ay/DOsX89WAzn9V85ffLJJ8yePZsPPviADRs2MGvWLJ599llmzZrl7Ud5uO+++ygsLHS/paam1urzR
ERERERE6ptVG7awcOVagmNbsaEk3OhyRESquXfOVvJLK40uQ46D1OPDu+++m3vvvZdx48bRvXt3rrzySm6//Xa
eeOIJAOLi4gDIzMzOuC4zM9N9Li4ujqysLI/zVVVV5OXlucf8kZ+fH6GhoR5vIiIiIiIiTUV6Vg6ff/sTFqsP6
+3NcWAyuiQRkWqyiyt48MttRpchx8HroWFZWRlms+dtLRYLTqcTgKSkJOLi4liwYIH7fFFREatXr2bAgAEADBg
wgIKCAtavX+8e8/PPP+NOOunfv7+3SxYREREREWnQ7PYqPv5qPkcysskN6OB+lY/RJYmIHNM3W9KZtznN6DLkL
3h9T8MLLriAxx9/nJYtW9K1a1c2btzI888/z7XXXguAyWRiypQpPPbYY7Rv356kpCQeeughEhISuOiiiwDo3Lk
zo0aN4vrrr+e1117DbrczefJkxo0bd1ydk0VERERERJqSn5atZs2mbUS2aM2aohCjy6l3qopzKFg0k/L963FVV
WANjyfqvCn4xbcH4OBT59d4XfiQawjrP6bGc4f/dy2Ooqxqx4N7jSbq3Bvd71ccSSZ/yXtUpu8CkxnfmDbEjHO
Us48frio7ufOnU7ZnFZagCCLPvYmA1j3d1xau/hxHUTaR5/zzFF69SP300Jfb6J8USUyov9GlyDF4PTR8+eWXe
eihh7jpppvIysoiISGBG264gWnTprnHTJ06ldLSUiZNmkRBQQFnnXUW8+fPx9//t0+U2bNnM3nyZIYPH47ZbGb
MmDFMnz7d2+WKiIiIiIg0aHsPHOKrHxcRHhrCRlssTi1L9uCwlZDx/lT8W/Yg5h+PYA4Moyo/DbN/sHtMi5vf8
7imfP86cr+bTmDHgce8b/zVL8AvK+oAKnMOkvXxgwR1+u2aiiPJZH7yMGED/kHkiBswmS1UZqVgMh1dnVe8eT6
VGXuJu+JZyvevJ2feM7SY/D4mkwl7QQYlm78n/uoXvfSREKlfCsrs3P/FVt66up/RpcgxmFwul8voImpDUVERY
WFhFBYWGrK/4Zxd6XX+TDk+l3SMN7oEERERacKM/j5VGpfSsnJeeOs9kvekYE3owuLCSKNLqnfyF82k4sgO4sY
/fdzXZM15DFdlGbHj/nPc1+T99Abl+9aSMOkNTKajwW36u3cSOLon4YOurPGa3B/+i9k3kIghE3DaKOh9fgwtb
pmNJTCMzE+mEdJzFIEdzjzuGkQaojev6ss5XdRNuT7y+p6GIiIiIiIiUvtcLhfzflzEtp37SExsyZriMKNLqpf
K967GN6492XOfIPXl8aS9cyvFm+Yfc7yjNJ/yfWsJ7nHucT/D5bBTumMRwT3OcQeGjtICKtN3YQ4KJ+O9u0h9+
QoyPrgX2+Ht7ut8Y5KoOLwDp7OCW8oGLMGRmANCKdm+EJPVV4GhNAn/mrcdm91hdBlSA4WGIiIiIiIiDdDOvSk
sWL6GuJhmbLFFUe6OGF1SvWQvyKB447dYIxKIHfsoIb3OI3/BG5RsXVDj+JJtCzD7BpxQYFe2exVOWwlB3Ya7j
1UVZABQuOwDgk8bSezYf+Eb25bMjx7AnncEgODu5+ATkOTajJsoXPkJzS68B6ethMJls4kccQP5S97jyOvXk/n

xQ1QV55zCR0Gk/jqcX84rP+81ugypgdf3NBQREREREZHaVVFRydzvF1JuqyAophXJ0YFGl1R/uVz4xbUjYvDVA
PjGtsWec5DiTd8S3H14teElW34iqMsQTFbf435EyZYfCGjTB2tI108ee3QnsOCeowjucQ4AkbFtsR3cTMnWH4k
YPAGTxerRNAUg55sXCelzAZWZ+ynfs5L4a16maPXn5P/0BtEX33/CL1+kIXhj6X7G9GlBUrMgoOuR39FMQxERE
RERkQZm6ZoNbN21h1YtElheGIZLzU+OyRIcgU+zlh7HfKIScRRlVxtrS91GVd5hgk87/qXJVYVZ2A5uJvi0kdW
eC9T47Koang1g07gFe+5BQnqfj+3QFgLa9MXs609gp70wHdp63DWJNDSVVU4e/mr7Xw+UOqXQUEREREREpAHJz
Mnl6wVLCQkK4oAjgmz78c+Ia4r8mnfBnnfY45g97wjW0JhqY0u2/IhvXDt8Y9oc9/1Ltv6IJTCMgLaeHWCtYbF
Ygi0pyj2+Z7uqKsn78X9EjZyMyWwBlxOX85d93pwOXC5ntWtEGpMlu7NZkJxpdBnyOwoNRUREREREGgiXy8XXP
y4hPSubyJg41hWpA/dfCe13IRVpuyhc+Qn2/DRKdyyiZPN8gnuP9hjnrCijbNeyYzZAyfzoforWz/M45nI5Kdn
6EOHdhh8N+n7HZDIRevoYitbPo3TnMuz5aRQsee/oTMYanlGw4iMC2vTFN7YtcDTsLNu9gsqsFIo3fI1/886n8
mEQaRAe/yYZu0MBeX2hPQ1FREREREQaiC3Je1i2bi0J8XFsKAmjwqV5IH/FL74D0Rc/QMHiWRQs/xBrWCwRw64
nu0tQj3GlyUvABUFdBtd4H3t+Bn7lRR7HbAc24SjKdu9Z+Eeh/S7E5agk/+e3cNqK8Y1OIubSf+MTEe8xrjL7A
GU7lxI/4WX3scBOA7GlbiVj9j34RDWn2QV3n8zLF2lQ9ueUMmvFAa47+/hn+OrtMbl+3Z21kSkqKiIsLIzCwkJ
CQ+v+t29zdqXX+TPl+FzSMf6vB4mIiIjUEq0/T5WGq9xm4+n/zWTfgVSiW7ZjTna09jIUkUYnxN/KoruGEBXsZ
3QpTZ5+LSUiIiIiItIALFyxluS9KbR0bM664hAFhiLSKBXbqnjux91GlyEoNBQREREREan3Dqdn8t3CZUSGhZJ
PIAdtAUaXJCJSaz5Zm8qh3DKjy2jyFBqKiIiIiIjUY06nk69+XEx2bgHxsdGsVfMTEWnkqpwuXv55j9FlNHkKD
UVEREREROqxzcm7WbNxKy1bxHO4wp+MSu3zJSKN3xcbj3Awt9ToMpo0hYYiIiIiIiL1lN1exfyFy3E4HIQGB7G
+OMTokkRE6sTR2YZ7jS6jSVNoKCIiIiIiUk+t27KdHbv30bJFPAfL/cmx+xpdkohInZmr2YaGUmgoIiIiIiJSD
9kqKpi/aAUWqxV/P382aJahiDQxVU4X0xdotqFRFBqKiIiIiIjUQyvWbWZ3ykFaJsRxw0ZPbpWP0SWJiNS5uZu
OcCBHsw2NoNBQRERERESknikpLeP7xSsICgjAx8dHswxFpMlyOF1MVydlQygOFBERERERqWdWrN/MoSPptIiP4
VCFH/maZSgiTdiXm9JIOWzDOqfQUEREREREpB4pLinlp6WrCA4Owmq1sr0k2OiSREQM5XC6eHmBZhvWNYWGIiI
iIiIi9cjydZs4dCSD5nEx5NqtpFX6GV2SiIjhvtycxqHcMqPLaFIUGoqIiIiIiNQThcUl/LROFaEhwVgtFsOyF
BH5hcPp4r1VB4wuo0lRaCgiIiIiIlJPrFi3icPpWSTERVPuMLOvPMDokkRE6o1P1h3GZncYXUaTodBQREREREES
kHigrt7FwxVpCQ4KwWiwklwXiwGR0WSIi9UZhuZ25G48YXUaTodBQREREREESkHti4LZnDGZnEx0TjcEFyaZDRJ
YmI1DvvrjxodAlNhkJDERERERERgzkcDhatXI+P1QcfHyv7ygMod1qMLktEpN7ZkV7E2gN5RpfRJCgOFBERERE
RMdiOPfvZnXKQhNhoADVAERH5E7NWHDC6hCZBoaGIiIiIiIiBXC4Xy9duxG6vIigwgLQKX3KrfIwuSOSk3vp+e
wZZxTajy2jOFBqKiIiIiIgY6HB6Jhu37SQuJgqA7drLUETkT9kdLj5YfcjoMho9hYYiIiIiIiIGWrVhC4XFpUS
EhVLmMHPI5m90SSIi9d4Hqw9hdziNLqNRU2goIiIiIiJikIKiYpat3UhkRBgmk4l95QG4MBldlohIvZdVXMH8b
RlGl9GoKTQUEREREREExyLrN28nKySO2WSQAe8sDDK5IRKTh0BLl2qXQUEREREREExAAVFZUsWrWOwIAALBYL+XY
ruXZfo8sSEWkwVqfkklmkhii1RaGhiIiIiIiIiIAbbs3MPB1DQS4qIBzTIUETlRThd8vSXd6DIaLYWGIiIiIiIiIB
li3ZTt0lws/X19cLtin0FBE5ITN25xmdAmNlkJDEREREREGR0paTV8DWnXtpFhUBQHqlLyUOq8FViYg0PJtSC0j
NKzO6jEZJoaGIiIiIiEgd27ZrL/kFhUSFhwFamiwiciq+0mzDWqHQUEREREREpA65XC7WbNyKr68PZrOZKhccU
GgoInLStES5digOFBEREREREqUOpaRnsOXCImKhIAA7Z/Kl06UczEZGTtTOjmD2ZxUaXOejoXyYREREREZE6tHX
nXopLyggNCQZgb1mgwRWJiDR8mm3ofQoNRURERERE6khVVRWrNmwhOCgQk8mE3WnicIWf0WWJiDR487akG11Co
6PQUEREREREpI7sSTlEaloGMc2OLk0+UuGHE5PBVYmINHwpOaVsPVxodBmNikJDERERERGROrIleTcVlXYCA/w
BNMtQRMSLvt6qJcrepNBQRERERESkDpSV21izaTvhYSHuY6kKDUVEvGbxrmyjS2hUFBqKiIiIiIjUgR179pOZk
OtOVAQAeXYrpQ6rwVWJiDQeuzKLySq2GV1Go6HQUEREREREpA7s3JuC0+nE18cHgFSbv8EViYgOLi4XLNuTY3Q
ZjYZCQxERERERkVpmt1exJXk3YaFamiwiUpuWKjTOGoWGIiIiIiIitezA4TSyc/OJCAsFoNJpIrPS1+CqREQan
2V7FRp6iOJDERERERGRWrb3wCFsFRUE+B+dXXi4wg8XJoOrEhFpfLKLKOhOLzK6jEZBoaGIiIiIiEgtcrlcbEn
eg5+fHybTOaBQ+xmKiNSepXvURdkbFBqKiIiIiIjUotz8Qg6kHiEy/OjSZJfr6ExDERGpHdrXODsUGoqIiIiIi
NSifQdTKSwuISzkaBOUvCor5U6LwVWJiDRea1LysNkdRpfR4CkOFBERERERqUW79h3AZDJhsRz98StLDVBERGp
VRZWTtQfyjC6jwVNoKCIiIiIiUksqKirZkrybsJBg9zGFhiIitU9LlE+dQkMREREREZFacuBwGtl5BUSEh7mPZ
dl9DKxIRKRpWKeZhqdMoaGIiIiIiEgt2XsglcrKSgL8jzY+qXCaKKyyGlyViEjjtyO9CIfTZXQZDZpCQxERERE
RkVqyffc+/P1+65ScXekLmIwrSESkibDZnezOLDa6jAZNoaGIiIiIiEgtKCOrJzUtg9Df72eopckiInVm6+FCo
OtoOBQaioiIiIiI1IIjGVkUlZQSEhzoPqYmKCIidWfLkQKjS2jQFBqKiIiIiIjUgiMZWVRWVuLn+1tQmK2ZhiI
idUYzDU+NQkMREREREZFakJqWgdlsxmQ6uodhYZWFCqfF4KpERJqO5Ixi7A6nOWUOWAoNRUREREREvMzpdLJr/
wGCg7QOWUTEKJVVTnZlqBnKyVJoKCIiIiIi4mU5+QXk5hUQEhzkPpatOFBEpM5tPaIlyidLoaGIiIiIiIiXHcn
IoriOjJDfzTTMrbIaWJGISNOORfsanjSFhiIiIiIiIl52JCML18uFxfLbHoaFCg1FROrcVnVQPmkKDUVERERER
Lxs34FUfHx+CwkrnSZsaoIiIlLndmUUU6VmKCdFoaGIiIiIiIgXVVRUkpJ6xGM/wwLNMhQRMYTd4eJIQbnRZTR
ICg1FRERERES8KCOzm6LiUo/QUEuTRUSMczC3zOgSGiSFhiIiIiIiIl6UnZePraKCQH9/97EihYYiIoY5lKfQ8

GQoNBQREREREfGi3PwCAEwmk/uYZhqKiBhHoeHJUWgoIiIiIiLiRdm5+R6BIUChQO1QRESMckjLkO9KrYSGR44
c4YorriAqKoqAgAC6d+/OunXr3OddLhfTpk0jPj6egIAARowYwZ49ezzukZeXx/jx4wkNDSU8PJyJEydSUlJSG
+WKiIiIiIh4zeHOTPz9/DyOaaahiIhxNNPw5Hg9NMzPz2fgwIH4+Pjw3XffsWPHDp577jkiIiLcY55++mmmT5/
Oa6+9xurVqwkKCmLkyJHYbDb3mPHjx7N9+3Z+/PFHvv76a5YsWcKkSZO8Xa6IiIiIjX2O1VZOXkERjw236Gp
Q4zVS4t8hIRMUqqQsOT4vVfdz311FMkJibyzjvvuI8lJSW5/+xyuXjxxRd58MEHufDCCwF49913iY2cuYw
bN47k5GTmz5/P2rVr6du3LwAvv/wy5513Hs+++yzJCQneL1tEREREOSU5RUWUm6zERYa4j6mWYYiIsYqrqgir
7SSyCBfo0tpULz+666vvvqKvn378o9//IOYmBh69erFm2++6T6fkpJCRkYGIOaMcB8LCwujf//+rFy5EoCVK1c
SHh7uDgwBRowYgdlsZvXq1TU4iZrYIAdU4WEalXtET5xHk9NNy/fz//+9//a
N++Pd9//z033ngjt956nK7NmzQIgIyMDgNjYWI/rxmMjKIiYnxOG+1WomMjHSP+aMnnnniCsLAw91tiYqK
3X5qIiIiIiMifyisox0FwYLX+1vikzKmlySIiRjuYW2p0CQ201//1cjqd907dm//85z/O6tWLSZMmcf311/Paa
695+1Ee7rvvPgoLC91vqamptfo8EREREGRP8otOLri6ffdk23qnCwiYjjta3jivB4axsfHO6VLF49jnTt35t
hQwDExcUBkJmZ6TEmMzPTfS4uLo6srCyP8uLo6srCyP81VVVeTl5bnH/JGfnx+hoaEebyIiIiInUpMzsHi8UzJCzXTEMRE
cMdKbD99SDx4PV/vQYOHMiuXbs8ju3evZtWrVoBR5uixMXFsWDBAvf5oqIiVq9ezYABAwAYMGAABQUFrF+3j3
m559/xulOOr9/f2+XLCIiIiIi4hWHOzMJ8PfzOGZTaCgiYrjC8kqjS2hwvvL4j7+23386ZZ57Jf/7zH8aOHcuaN
Wt44403CjBeONN4Cj0/SnTJnCY489Rvv27UlKShuKhhx4iISGh4rIKhhhhKGBiy66CDg6M3HUqFHuZc12u53Jkyczbtw4dU4WERE
REZF6qdxmI7+wyKMJCigOFBGpDwrL7UaXO0B4PTTs168dFffx9Zurqx/4Zxd9uzZzDh2M2mxkzZgzCI/OjmpDUVERYWGrK/
jJp0iQKCgo466yzmD9/Pv6/4/d19uzZTJ48mQsPPDh2M2mxkzZgzTp0/3drkiIiIiIiIJeUVJaTmWlnZDgII/jCg1
FRIynOPDEmVwul8voImpDUVERYWFhFBYWrK/4Zxd6XX+TDk+l3SMN7oEERERacKM/j5Vas+hI+n864XXiI1uR
mDA0QkRLhe8nR6PC9NfXC0iIrUpMTKApVOHGV1Gg6JfeYmIiIiHhBaVk5lfYqfKy/LeiqdJkUGIqI1AOFZZp
peKIUGoqIiIiIiHhBma0Ch8OB1fpb92QtTRYRqR+KK6pw0hvlYttao3/BREREREEvKCsrBw42vzxV+UOy7GGi
4hIHXK5oNhWZXQZDYpCQxEREREREES8os9kw/WEpsmYaiojUH2qGcmL0L5i5iIiIiIiIgXlJfbcOK59K3CpfOMRUT
qC4WGJ0ahoYiIiIiIiBcUlZR6LE0GcGCgFBGpNxanhiIiIiIiIl5QUFyR6LE0GcCg0FBGpNxQanhiFhiIiIiIiIl5QUFyC7+86JwO4FBqKiNQbBeWVRpfQ
CgOFBEREREREER8YLComKsPp6hodOgWkREEpLqyCofRJTQoCg1FREREREREROkcvloqikhpmGBtUjIiLVOVz6W/lEKDQ
UEREREREE5RXZ7FRUVdqx/CA2dWp4sIlJvOJwKDU+EQkMREREREREREZFT5HA6Cl1lcmMyeIaF+PBURqT+cmml4Qhq
oiIiIiInCKnO3U0NMQzNNRMQxGR+sOpmPYnRKGhiIiIiIiIiIjKXI6nUdDQ5NmGoqI1B+aaXhiFBqKiIiIiIiIeRIKIFco/aV8QqxQqx/PURERERERERET
+jNP1yOzDPOwsVPdkaQz08d1G56LDBNi6YPNticmlKEHqt19/kRMeGor5d9tGnFalz90ToY+WiIiIiIjIKTrmT
EMtT5ZG4HbrHH5qbaUseSfnLsjkUFJ/OhP6YvNphcnha3R5IsdkK6j0eN+i3+ScEIWGIiIiIiIip8jpdOGCaqG
hfj6Vhm6ITzJdnDtpW2BmWP9OHIqK4tqv19Bm7zIAjiT25HDLAZQEtMFUFWhwtSJ/zmzWL3JOhEJDERERERERGRU
3SsmYYiDdOdPnPACX44GVdk4o2OORyJCOHuz/wILSymeeommqduAiA7pi2Hks6dotcZAIkb7Y4d7+XM
KDUVEREREE6RO+WEGkJDiOm77kvD1ce+hR707e73b8jfwbu+bdkVU8wd4y3c8OEw7QtK3Oejs/YRnbUPgNyAK
JKb96cgqjv+/sOxmyx1Xr/IH2mm4YlR92QRERERERERZFT5HIdffvjj6NWhYbSgE22v+fxvq/Jxd8zbQAUhTmYNtH
GOuYhNV4bVZ7LWXu/5fzVT3HG8vsI3vcJ5YU7jhepC6YXJW1jhepC6YlIKdEMOOFBEREREROUVOpxMX1WcaK
XoGRx6pdvx2jvBNVRfyrcU4fOHlq8opXZnIuYsPY3bV/PkeXFXK6amLIXRZfUtsNJDO2D2XWFuDOqe2XIuK
m5cknRqGhiIiIiIjIKTrWnoYKDaWhusU6m5ryFV+TiwtS8ni3/W9h39sDOjkUFcc183LxqfzzzmYRWRyVJuaSt
GshTpOZ9FZ9FZ9SGvenxL/JFwOf2+/DBEPvn6KwU6EPloiIiIiIiK1xEehoTRAvc17GRGacczzt1vS+bwoID
YTx2ySR0fzD1z/AkuLDqu55hdTpofWEvzyDFYYOdCH2ORERERERERETpHVa
sVsMuNOOj2PKzSUBuiBgDlYHMc+bzXBPyusPIfnoF1xJUy53MyDH4fQOq/4hJ8bk7GTmIydABRGtOBQ28Hkh3W
myhFO9R1DRU6cf5CvOSUOKAoNRURERERETpGP1YrZYsbxh9DQ1+w8xhUi9dNploPOcmz+y3FXlOzl3fBOZPuUe
hwvCndy3zVl3Pi+P4N+aZpyMsLyD9N93WwAyoIiONhuCLmR3akkGlyNs5vFDxs/5Ks1bzGk2yX8feDNxxy3cMv
nLN3xFfklWQT5h9GrzSD+/Tr8LFWD8SOdc/PV/yX1bt/wNfqz4X9r6Nf+xHucxv2LWbN7h/4598g
7WH5olQaCgiIiIiInKKfKwWzObqMw11dLk6tjPhwWR8jIi6zbwY2ePXzzeIqoRe24uA4A4+3WrGVWLFqNzvQRrr
KLM2n8+YvgC+o9A3gULuzyY7pTbk5HpyNI9Y4mLWT5clf0zyyzZ+OW7tnAV+mTVxXsgoO896ipwEYc
+ZNx3XPrQdWsG7vz9w8+imyC48we9GzdG7Rj+CAMMorSvh67dtMGf3McccuLUNpG/iIiIidcj
X1weLWTMNpWHrbDlCX8fG4x5/efE+YuxBxzz/yXmVzLg4ikqr94Ia38py2u34gQGLnmTwwttpl/oBwY49YPnzB
iz1WYW9nJk//4fLBt1BgF/In45NydxOm9hu9Gs/nKiQODon9qVvu6EczN513PfMKDhE+4TTaBXdkb7thuHvGOh
ucToAc1e/wdld/o/IkFjvvsh6wDfg6IxwOX76aImIiIiIiJwiH6sVi9mMO6HQUBquB/0/xcLxf85aTS7GZ5b+6
ZifOhXy2FWBlIYeO1w8WRaXk5b7lnP6OhcZtuB2uux9nbDKrZgtZV5/Vm36eN1lLdGt5Bp1a9PnLsUmxXUnN2c2
BrKN7P+YUpbH9OBq6Jp5+3PdsHtWWQ9m7Kaso5lD2buxVfl+lZSc/YpNvF3nlh9YyWJp84zcsUERE
RE5RRaLBR9fK5K5Wl5R7HfbU8WRqI9pYMznCsP+Hrrq48xMyStuQH2485ZmdsGXde48O/PoOkNi3vVMr8U3GHtxB

3eAsAedFtSE06m4KQjjgcodTXRirr9v5Mas5epl783+Ma36/9cEpthbzw5W24cOFOOjirywWM7D3+uO/ZJbEf/
dqP4Ok5N+Fj9ePKoffga/Xno2UvceWQqSzdMY/F274g2D+MywbdQXxka2+8VMP5BykOPFEKDUVERERERLwgKCC
AwiLPWVdmE/ianFQ2OsYNOng84P8ZFseftEw+BosJxh7O5fVOoX86Li/Qzu1X2Hngm1i6bs882TKPW2T2fiKz9
wNQHBJDarsh5EZOxe6KrDeNVPJLsvh8xatMHv1OjU1MarI7bRPfb/yAS8+6lVYxnckpSuOzFa/y3fr3+FufK4/
7nqP7Xs3ovle73/923bt0at4bi9nC/A3vc/8/3mLbwVW8u/BJ7hnz2im/1vogQDMNT5hCQxERERERES8IDAjA4
aiqdjzI4qCyqn6EFCI1aW3O5izHmpO+/kbfAt7PC6E08s9n81VZ4F//l8vEZrGcsyQLs6tuZuKGFGfRZeMnANj
8QzjUbhDZOadRQRy4TrVNy8k7lL2b4vICnvr8n+5jTpeTfelbWLJ9Li9eNx+z2bO+b9a+w+ntz+HMzqMBaB7Vh
gp7OR8ufYGRvcef1DOz8g+xds9P3Pv311m58zvaxfcgJCCc3mOHM3vxM9gqy/D3DazFjOTdOEzDE6fQUERERER
ExAtCggKpqmGmVrDFQX6VfliV+usB/8+wOqsH3sfLYoLrKpy8dJx9kmecmcuBqEiu/boIn8pjL2uuDf62Yjps+
4YOfEOVxZfUtmeSGdeXMmsLcNbt12nH5r25/x9veRx7f9EzxIYnck7PcdXCPYDKqgpMJs9w1j3O5Trhe7pcLj5
a+gKXDPgnfj4BOF1OHL98LjicR/8+c7oax96s2tPwxCkOFBERERER8YLgoEDsVTWHhiL1VYI5jyHOVad8n2ttR
3ivsiN5vuV/PRhYOLGQw+GB3POZL8FFf95MpbZYHZUk7V5EOu5FOIHO1n1Jb34Gxf5JuBz+tf58f99AEiKTPI7
5WvOJ8gt1H3/35ycJC2rGhf2vA6BbqwEs3PIZLZq1o3VMZ7KLjvD12nfo3nIAZrPluO75eyt2fkuwfxjdW58JQ
Ju4bny7/l1SMnew49Aa4iJaEegXXBsvv85ppuGJU2goIiIiIiLiBUGBAbic1ZdbKjSU+uxB/zn4OE99tp8ZuK7
EztORx3/Nrtgy7rrGhOc+iyTuSOO1SDkeZqD5gXUOP7AOgOy4ThxufRaFQe1xOowLzfJKsjxmFo7qfQUmTHy99
hOKS3MIDginW8szuODOiSd876KyPL7fMJs7LpruPtY6phPDe/yd/313PyEBEVw59B6vvI76QDMNT5zJ5aqjTQT
qWFFREWFhYRQWFhIa+ucbstaGObvS6/yZcnwu6RhvdAkiIiLShBn9farUnm9/Xsb7c76mS4e2Hsf3lfuzMP8Ek
hSROhJrKmSp3234Uum1ew5v2YUsS8kJXWN1wP3fRNJte5bX6vCmwvDmHGo7mPywzlQ5I6ivnZjlz426oRtte8U
YXUaDopmGIiIiIiIiXhAWElTjcc0OlPrq/oA5+Dq9FxgCXFtg48moE7umygKP/l8e1zaL5dw6bJByvMIKjtB9/
QcAlAVGcKjdYHKielBJdL3pxCx/LbRZgNElNDgKDUVERERERLwgNCQYs8mEw+HAYvmt2YBCQ6mPIkOljHIt8fp
9xxft553wrmRaik/42rfPzOVAsOgmzivGp9K7Yaa3BJblO2nLXGAulb4BpLY9i6yY3pRbEsCpiKU+C4tWaHiiF
ImLiIiIiIh4QVhIMH5+vtj+EHYEmp2YqV8zpOQeCPgCP1dFrdx7YkHZSV/7c4dCHr3Kn5Kw+t98w7eynLbJPzJ
g8VMMXng77Q7NJtixByz1M/BsygJCffH1NybUXbRoESaTiYKCgmOOmTlzJuHh4XVWO/FSaNjAORwOPnzpaW4c3
p/LTmvDTecM4NP/vsDvt6osyMnm5XuncN3ZvbisZxv+fd3lpB3Y/6f3PbRnFO/fch3/HHY6Yzol8PWsN6uNmf/
hLG7/v+Fc0acDV/TpwH2XXsCGJT97jHnniUe4un8XJg3pw5J5czzOrZg/j//886pTePUiIiIiIvVHWGgIfr6+V
FR4BgYmEwRptqHUI+GmUka7FtXa/S8rSiGuKuSkr98VXcadE6rIaN5w9gK1uJyO3L+CO5e+yLAFt9Nl7+uEV27
FbDn5AFW8J9xLswxfe+O1QkJCqKqqch8rKSnBx8eHIUOGeIz9NSyMj48nPT2dsLAwr9RQlzR3toGb++arfP/hL
G558iUS23Vk37bNvHL/7QQGhzD6qutwuVw8dfO1WmjITW3L8df1WHys3PvfdwgICmbezDCPEKzxvpW2cmITW3L
mqPN558lHahwTFRvPFXfeT3yrJHC5WDj3U566eV566+RqemfMDLdt3ZO3PP7Dsmy946KOPST+4n/8+cCc9zxpMaEQUp
cVFfPDCUzz8zke1+NEREREak7IUGBBAT4Y7NVn7OVbHFQ7NCPX1I/3BPwJf7O8lp9xnUFpTzW7OSvzw+sYsr
4Iu7/LpYeWzO9V1gdiTu8hbjDWwDIa9aG1DZnUxDSEYcjFDVSqXveWpo8dOhQSkpKWLduHWeccQYAS5cuJS4uj
tWrV2Oz2fD39wdg4cKFtGzdfQbPpI+Q0YQ0yKRAaPO57SBg9m7dRMA6Qf2s3vzzeiY
9/CTtuvekeZt2THrkSptNpZ988Ux79uue0+unjqNsOZfhI+Pb41j+gO7lz6Dh5PQug0JSW0Zf/u9+AcGsXvze
gCO7N9D19MH0K77aZx9/sUEBAeTdTgVEeeeYyRl11FdEIL735AREREREQMYjabiQoPq7Y8GbSvodQfwSYbF7o
W1vpzLiO+QPwpzDYEcFrgsfNz+XZoLE5Tww3aInP2c9qaWQxecD+nr/8XcYWL8DHngDHngM1pdGlNRliMdOLDjhO7E
h8fz6JFi9zHFi1axXIUXXkhSUhKrVq3yOD506FAeeeQRIvPXJ2ZLxgEhgYyMUXX0xubm61Z/3vf/+jbdu2+Pr6rF
jR9577z33ubuuovzzzz/f/f6LL76w2rXXjbfeOqXq9Cwewgavy9bVy4jLWUfAAd2bmfnhjXXOG
jQMAPsv37D4+vm5rzGbzfj4+pK8Hi62sjI69uwLQKuOXdm3bQs27aFSpuNuJatSV6/mv0
7tnLelRO99nwRERERkfogpllkteXJkcK9S9qf5fEEegqrZNnTSrwzNmnpHLa2MisfvVPKGlIQkuzqbLx
k85++eHGbjiXhKzv8GfI2DSLxZqU1hMzassT8bQoUNZuNzmpC34yzpeSgDp4vwjqbjm4mtfxhbufGdmxPGdW64
XM3HiRCZPnsymTZtYuoXePv4bcsOdZj1cvC9Rno3sbCK78RAGd+ucltfunnr74iAqdxK2bOqHbwY5BZlQQ+nq9K1H+SaGrQNCNc/KGN8+sPowzhb5NGtZAAEoOWNbi99vlVfu+5oTNdk1ph9Xo3+OF/g8GmQ3MJaNe5dqEVYHEZqWqDpH+xaEBxkzBU0UXiCX1QZtRDSTzLuW2luntmilJd5ivhs9kkAmPmnR2Pn+
/L9h7P47v23CYmI5J+PPkArGSjCECzyAiczuqzhyJ97AZUI+IpkArG8FOdPe/vxQd4M7wbadaiU77
XovaFHLkygPs+9SW4sMQL1RnPr6KU9tu/pT3fUmWxcrjtQDLj+lJqTQSnj9HlNSqRCUFeu9fQoUOZMmUKVVVVl
JeXs3HjRgYPHkmVRUVDB06FD27/fsKHDh1YsWKFxyzBzXyTSqQHnTwYK5OzEXjtknVEnnj8nmTP11FVd6lpnubl1a9wqu4evtW32njyUU/VA8TCX2gsNjDAR2IdkLSh6c6dswLiWk9BGZe4e2Khr2ZJG
XQqB9w0EbAAOT0MCt8O35+bQ0HAFkKJ7rFKeurKm3ErEUyoLCDq4Ee6ZMtYGgxCxWUkOuB1JwQXX/ZaQxe24n6M4o5vdTwyfafS04MlfqP+9tu1uYrW3yoWzMo5rDAjgsUXOzbLKWd2Ik9s7WdMwZWNkUT2JBd6g3yDJx36nSi28bb48bCTV24NJ9xpi4kfVtSPEBU3sZ8QNfDKtuh2vadxC1ZRMaKMG6XfDuh5mNqVVwcbrqKHJrO5tFmFuhFcm1GHCPiG4tDHKnLX6Fg9zbhQ6NkSpn1J7m6lJMNFvcmbvmoLSTTl2MWL9JaeSc5xRZq3Rb0Zj6ZmXw1VpA0HFF1C4R8Gkd6GQrqUGG3cHNhfpa3L8p2bFY3CHVgeQ4bOcRJ49sMB83zWJWkqKfQ1YfQKDHBDH5qvCxaG0FWNHTfbBfdEhkB73KQG3ijC0NX2k2K1Hd1rjfYb1o8vK19ucuUkuPmUQeVVL3eB5sfbjiwHgmH3T9w7R3hSuz3mY4FPzOeiDWRgs6ai1Q0Kuf/10lgQg3GMxUQ4EyLQFKTFjOvhKO3CzRiQeKcMAD0hO

EREExGhhITV3fI1SaCj1wBOB3xDkqtvA7YaCYq/da090OXdcU0V6iyiv3bO+sDqqaL17Mf2XPMeQn6fQMeUdQu0
7MFlsRpfW4JktJsJjvTfTcMiQIZSWlrJ27VqWLl1Khw4diI6OZvDgwe59DRctWkSbNm1o2bJlteuTk5Pp37+/x
7EBAwZUGzNw4ECPYwMHDiQ5ORmA8PBwTjvtNBYtWsTWrVvx9fVl0qRJbNy4kZKSEhYvXszgwYNP+bVqpmED9+4
z/+bi6ydz1uiLAGjVsTM5aYeZ88bLDL14LABtu/Xgubk/UVpcRJXdTlhkFPeOHU3bbj10+fk+vr5HG6H88py92
zbxzbtv8c9Hn6429vD+PSyeN4dn5/zAz59/SOe+ZxAWGcWf/s/Xn3gDspLSggIrvmbLBERERGRhiD0l9DQ5XJ
h+t0ebMEWJ35mBxVOi1GlSRPnRyWX8kOdP/eS4oO8Ed6NI16YbQhQEFDF7ZcXNtgGKcfDDDQ/uI7mB9cBkBPbk
cOtz6IguD1ORzBqpHJiwqIDsFi8N2euXbt2tGjRgoULF5Kfn+8O5xISEkhMTGTFihUsXLiQYcOGee2ZNRkyZAi
LFi3Cz8+PwYMHExkZSefOnVm2bBmLFy/mzjvvPOVnaKZhA1dRbsPOhxmDZrMFl9NVbWxQSChhkVGkHdjPvm2b6
TdspNfrcTld7nOUPY67XLw+7R4m3PMwAUFBOJ1OHFVHf9v663+dTu3hICIiIiINW3hoML4+PlRUVp9ZGKV9DcV
AUwLmE+LyTnB3ov5Z4N3n/tog5ZthsTi9sIKuvmuWuYueq2cwZMG99NnOBDHFy7Ca84DqP/dLdd5cmvyrXxucL
FqOyGPfwEGDBvHdd9+xZs2aGvczBOjcuTOrV6/2OPb7Biq/jlm+fLnHseXLl9OlSxf3+7/ua7hgwQJ3DUOGDOH
DDz9k9+7dp7yfIWimYYPXd+g5fP7adKLjm5PYriMpydNyEO7k3n78WnOGz6Kn
mcNcY+Zfs+tRMbEccWd9wNHG6gc3ndOmXGV3U5uZjopydvwDwxyzyx8/7n/OGvQMKLjm1NeWsLSr79g+5oVPPT
WB9Xq/OnTDwiNjKLfsHMB6NS7H5+88hy7N61nw5KfadGuGuA0GhYbX1YRIRERERqRPhoaEE+PtRbrPh/4emDZE+d
tIq/Y5xpUjt8aGKyO3zDcuYLio+xOvh3TjspdmGv5rVP5eDURFc/1UxPjUOIGqMwgqOELb+QwDKAiM41G4QuVE
9qCAGXIO/QDOZkfG1ExrefPPN2O12j2XAgwcPZvLkyVRWVh4zNLz11lsZOHAgzz77LBdzzD
77LBdeeCHff/+9x36GAHfff
Tdjx46lV69ejBgxgnnz5jFnzhx++um3PUkHDRpEcXExX3/9NU8++SRwNDT8+9//Tnx8PB06dDjll16nQsIG77sH
H+HD607zx6HOU5eYSERPLOZdeyT9uut09Jj8rk5lPPkjhbg7hOTEMufAf/P3GKR73yUk7gslk9rjmrovPdb//1
duv8dXbr9G13wAefe9zAArzcnj5nlvJz84iMCSEVh0789789BbH3DaQM918wU52Xz+2kv858Ov3Mfa9+jFBdfcw0M
3XEVYVBS3PPmSNz8sIiIiIiKGiAwPJSIslLzCIiLCQj30aV9DMcpk/x8IcxYYWsON+UU8EO39+y5qd7RByr2f+
RJSODgapByvwLJ8Om35EvgSu9WfQ+3PJiumF+WW5uBU3POryATvb4M2dOhQysvL6dSpE7Gxse7jgwcPpri4mI4
dOxIfH1/jtWeccQZvvvkmDz/8MN0mTWPEiBE8+OCD/Pvf/3aPueiii3jppZd49rnnuOWWW2jdu5nuN5xXUHEHA/kJP3+ApBnfQ9gssTsUQ9zfxZ
i21+bXQ8n0+BNdCOPvbHdnxrTnS10GGawzvL6Ha42VsUi3bXXs/vMxE+tNx+tnj5bZM+SP9Zeb8OuOzg+j
dOxIfH1/jtWeccQZvvvkmDz/8MNOmTWPEiBE8+OCD/Pvf/3aPueiii3jppZd49tlnue222OhKSuKdd97xmDOYE
RFB9+7dyczMpFOnTsDRINHpdHplPOMAk8vlapTzWYuKiggLC6OwsJDQ0NC/vsDL5uxKr/NnyvG5pGPXjIiI
idcHo71Ol9r35wecsXrWeTu2SPI7n2a3MyY4xqCppqsw4WBd4G5HOPKNL4bzEbqR6ebbhr8LKrfzrsyASDufWy
vObEofJTFpSf9ITTqfUrxUuR9Oe4XzlYwMIbRZgdBkNkuauioiIiIiIeFGL+FicTme14+HWKizag0zq2I3+C+p
FYAhwU15hrd27MKCKOy4vZHOP2L8e3MhZXE4S96/k9GUvMXTBHXTd8xrhFVswWOqNLq3OBYX5KjA8BZqvKiIiI
iIi4kWxOVGYTCYcDqdHx06zCcJ97OTaff/kahHvMeHkGvM3UD3DNsT5pan8z96NQz61M9vQaYHHR+dyZbNYRi/
KxlxDeN8UxR7ZSuyRrQDkN2tNatIg8kM74XCE0tg7Mce1DTe6hAZNoaGIiIiIgXxTaLIjDAn7LyckKCPTfgj
lJoKHXoOv/FNHNmG12Gh5vzC7gnpnYXPb7XP5dDURFMakINUo5XRM4BInIOAFAa3IyD7YaQG9kNuyuqUTZSiW+
nhqunovF9RoiIiIiIiBgoJiqSkKBASsvKq51TMxSpS9eZ5xldQjXnlR6mtb32g5zF7QqZdpUfxeEhtf6shiqoJ
Icumz7j7J8fYeCKeOnM/gZ/joDJYXRpXhPfVqHhqVBoKCIiIiIi4kU+PlYSm8dTUKurWU9SN672XOKsM8P
oMmoOOb9u9ljcl6ycOybYOZIYVSfPa8j8Kkppv/1bzlzOHwYtuYM2Rz4hyLkfzA33FxO+fhaaJSoOPhUKDUVER
ERERLwsqUUClZXVA8IoaxV+Ju2zJrXvn+avjC7hmEaWHiGpDmYbwtEGXKdeXsim09Qg5XhZHVW03rOY/kueY8j
PU+iUMoNQ+w5Mluq/CKnPYYpNCMZsb956NtU17GoqIiIiIiHhZbHQULsDpdlcmEy/fZDq8kE8X4VHLCpm6fUnsv8V
hLvTDO6jD91S34ud8TUTSThNMN/zsvlimaxnL9QDVJOhBlIOLiBhIMbAMiJ7cDh1mdRGNwBhyOY+txIJU5Lk0+
ZQkMREREREREvi2OWhb+fL7aKCgL8/T3OKTSU2naTZW696Zh8L0eUptHG3p39PoV19sz3T8/lYFQEN3xZgm9FR
ZO9tzFplrmbZpm7ASgMTyC17WDywpgWI5eqcfMoUGoqIiIiIiHhZbHQUwYGBlJSWVvsNE7SvodvesSiS3z
XkuhMNbqM43JbXi63xdZtLLG0bSFpVwVw/6e+hBQU1+mzG5uwgjTC1n8IQHlAGIfaDSan2WlUEGN4J2aT2URsm
1BDa2gMtKehiIiIiIIlwUFBtCyeRyFxSXVzkX4X4VBFgbjzdSaV+ucXnC6NLOG7DytJoV0d7G/7evmbl3D6hUg1
SvCigvJCOW79i4MJ/c/bSu2md8QWBroNgrjKknqjmQfj6a57cqVJoKCIiIiIiUgu6tG9Dma35aQDaM0HYGXOVHkY0/TZFx
O8mkhwHjC7jhNyal2vIc4sCHNx5eP357dnP
ZwQMsK63+y4YP8vMZsW8vPXfv4tKDB9hS7nnPp7IyOWPPbobt28u8Is8l3/OLi7jpcN3NPvWpstFm50+csfhpB
i+8nfYH3yO4ajcmS939vRevpcleodhVREREREESkFrROTMBqtVBZacfX9/grIHymsCbEFzDxQfnWhexeo
OLI32ld3Z41t3exv+ymmGJ87LZXx0DBf8nHPKDVIKHQ7GHzrI6YFBvN4ikUiLhYN206FmyzGvWVdexpmBQUxpF
k2IxcIXhYXcdPgwH7VqTZdftjb4rqiIp7KzeDg2lh7+AbyXn8ekw6l8SGKKuVhSXFfF1UxFuJiRystPNgRjp
nBQYRYbVS7HDwUnY2MxJbntJr01kW15PElFUkpqwCICuhG4dbnU1RYDucjqBae258OzVB8QaFhiIiIiIiIrWgV
fN4wkNDKCgqJq9ZZpMc5zTQUbxvhs412jr1Gl3FSbsvLYXKcz18PrCWz++VxMCqCf35Zgq/t5L82Z+TlEufjw3/
i493HWvj6/uk198V4znS8PTqan0uKWVRS4g4NZ+bn8Y+wFuwSAAPFpeWMqewkOujothfUcnp5gYF08w+gm
38AT2ZlcthuJ8Jq5dnsbMaFR5DgY9zPyo1qS20Zv80E44HGF4rZGKCZp3iPDOvZo4LU8WERE

RERGpBcFBgbRrnUhBUfVmC6FWB8EWY/b6ksbpLp/PjC7hpA0uT6dDZbihNSxrU8hDV/lSFBFy0vf4uaSEbv7+T
DlyhLP27uGSAyl8WlBwQvdwulyUOp2EWY7OTqx0udhhs3FG4G+z8swmEwMCA9n0y7Lnjv5+bLPZKHQ42G6zYXO
5aOnry/qyMpIrbFwRUT8DtIjcA/RY+x6DFzxA/3WPEF/wM77mbDCd2ozP2NahBIb+eVgrx0ehoYiIiIiISC3pl
C4Ju73mcDBeXZTFS4b47KSTc7fRZZyS2/OyjC6BlKhy7ri6ksMtm53U9Yftdj4qKKCVry9vtEhkXHgE/8nKZG7
h8S+9ficvjzKnk1EhR8PLAkcVDqCZ1XOhaJTFSk7V0b9bzgoK5oLQUMYePMD96ek8ERdPgNnMo5mZPBwbx0cFB
Zy3fz/jDx5kTOX9nOUcVJJD502fc9bPjzBw+b0kZn+NP4fhJJpGte6uBjfeouXJIiIiIiIitaR1iwR8fX2wVVT
g7+fnca6Fv4092tdQvOAu388a3F6Gf3RWeQadKnuw07fAODqKAhzcdVkBd38fS59NmSdOrdPlopt/ALdHRwPQx
d+fPRUVfFyQzOVhf73H3tdFhfw3N4eXm7cgynpicc3kZtFMbhbtfv/VnBwGBAViBV7LzeHL1kksKi3hvvQOPmu
ddEL3rmt+laW03/4d7fmOKouVIOkDyIjvR6lvIjj+egZh6x4nF/pKdZppKCIiIiIiUktaNo8jIiyEwqLq3VAT/
Sqw4DKgKmlMBlj3ONWxw+gyvOL2vEzqw5eE0wxP/S2XuefE4jQff2wSbbXS1s8z1Grr60t61V9vRfBtURHTMjJ
4PqE5Zwb9thQ53GLFAu5Zhb/KdVRVm334q/0VFcwrKuSWZtGsKS+jb2AgkVYro0JC2VFRQamz4STMVkcVrfYup
f/S5xmy4HY67Z9BqH07JkvNHamDI/1o1uLkl5iLJ4WGIiIiIiItSTA358OSa0oKK6+r6Gv2UVzf5sBVUljco/
fZ95qH2G4M8sz6WwPN7oMtw/65vLyP8Kp9Pf768FA74BAUio9tx04YK8kwfrnTUi+KSrigYx0nolPYHBwsMc5X
5OJLv7+rCordR9zulysKiujp39AtXu5XC4eyczgnpgYgsxmnC6och1NYn/9r6MeBLMnwwwkHNpA3+X/ZeiCu+i
R/CKBecux+pS4e6i07q5Zht6kOFBERERERKQWdWjTiip7FS5X9Z/UkxQayinoY02hh2Ob0WV41R15GfVituGvl
rcp4sHjbJByVUQEW8rLeT03h40VlXxdVMinBQVcFhHuHvN8dhb3pqe53/+6qJD70tOYGh1Dj4AAsquqyK6qotj
x22zACRGRfFZYyNzCQvZVVPCvzEzKnU4urmHJ82eFhURarAwNPlpvr4AAVpeVsbm8nFn5ebT19SX0lyYrDV2zz
D2cseUDBv14D4N2PUfPOD207xr01xfKcVNoKCIiIiIiUotaJybg7+dHua16A4JW/jbM9SkhkQblPr/PGt3nzxn
lWXSpR7MNAQ5ElTNlQiWprf58Flv3gACmN2/BtOXFXHgghddyc7k3JpYLQn8L93Kqqki3293vf1pQQBXw76xMB
u/b6357Iuu3/RT/FhrK3dExvJyTzSUHD7CzwsbrLRKrLU/Oqari9dwc7o+NcR/rERDAhIhI/nk4le+Li3k8Lv4
UPxr1kzVtP8OWziC+U8xfD5bjZnLV9OuuRqCoqIiwsDAKCwsJDQ2t8+fP2ZVe58+U43NJx8b5l6SIiIg0DEZ/n
yp1r7LSzgNPv0xJWTnN46r/QPt9biSpFf4GVCYNWXdLKl/63IcZp9GleN3qgGiuiw2gvq27Njvh7h+i6LPxxBq
kSNOIveACmj/ztNFlNCqaaSgiIiIiIlKLfH196NO9M4VF1fc1BEgKqHlDf5E/84D/p40yMAToX55Nt8pwo8uox
mmGp0bl8sW5MSfUIEXqRsi55xhdQqNT65/lTz75JCaTiSlTpriP2Ww2br75ZqKioggODmbMmDFkZnom9YcOHWL
O6NEEBgYSExPD3XffTdVxdBwSERERERGpb7p0aIvFYqHiDO0SQEuU5cR1thyhn2ODOWXUqjvy0jC56tlUw1982
CePl8aGU3GcDVKk9pkCAgg++2yjy2h0ajU0XLt2La+//jo9evTwOH777bz948Pv30UxYvXkxaWhqXXHKJ+7z
D4WDO6NFUVlayYsUKZs2axcyZM5k2bVptisIsiIiIiIlIrOrRpRWyzSHLzCqqd8zO7SPCrvt+hyLHc7/cZlkY6y
/BX/Ww5dKus3uijvliZVMSDV/tQGKltJuqDkBEjMPtrmwdvq7XQsKSkhPHjx/Pmm28Sm28SERHhPl5YWMiGTN4/vn
nGTZsGH369OGdd95hxYoVrFq1CoAffviviBHTt28P77790zZ0/+9re/8e9//5tXX32Vyhp+MyciIiIiIlKf+fv5O
atbJwqOtURZXZTlOLU1Z3Kmc53RZdSJO3Pr72xDgIORNm6/uoJDrf+8QYrUvrCLLjS6hEap1kLDm2++mdGjRzN
ixAiP4+vXr8dut3scj4+Pp3Q0dut37duxMbG+seM3LkSIqKiti+fXuNz1dRUUFRUZHHHxEREREREREEp
L7o2qEdFr0Zykp7tXOtAsoxaYmyHIcHAj7DgsPoMupEn4ocutfj2YYAJf407ro0n7W9Y/96sNQKa2wsQQMGGF1
Go1QroeFHH33Ehg0bEobOKJJ6qdy8jIwNfXl/DwcvDwcI/jsbGxZGRkuMscH06OnT927mSeeeYZwsLC3D/x8fF1
0cz1Nq1i0OHDjHglzXoAwYMY0vWrWWR1/nI5l4bL66Hh8OHD2bp1K5s2bXK/9H99Bi4Jn4iISH3StNbzGYzlfYali
j72wgwN
41lp3Li4s35DHWuMLoMQ9yde7j+zzb8xarWxTygBil1xr9rV/zatTO6jEbL66Hh8OHD2bp1K5s2bXK/9e3bl/H
jx7v/7OPjw4IFC9zX7Nq1iOOHDjHglzXoAwYMYOvWrWRlZbnH/Pjjj4SGhtKlSxdvlywiIiIlInOrVNIjoqg
tz8wmrnzCbooNmGcgwP+s/Bh+phc1NwWkUePRvIbEOAQ2qQUmc0y7B2eTOODAkJoVu3bh5vQUFBREVF0a1bN8L
Cwpg4cSJ33HEHCxcuZP369VxzzTUMGDCA844A4Bzzz2XLl26cOWVV7J582a+//57/HnzwQW6++Wb8/DTNV0RER
EREGqbAAH96dulIfkHNS5Q7BZapIYpUE20q5BznUqPLMNRdDWi2IfzWIGVNn5pXS4oX+PgQev5KfK
FF17g/PPPPZ8yYMQwaNI4uDjmzJnjPm+xWPj2Pm+xWPj666+xWCwMGDCAK664gquuoopHH33Ui4kREREReRES8plun9ljMJ
ioqK6udC7E61BBFqnkg4At8qf75OpTOqMijVOXDmW0IgNnEs+fm8OnIGBwWi9HVNDqh54zAcjXmsnlcjX
KX2MVFRURFhZGYWGhIfsbztmVXufPlONzScd4o0sQERGRJszo71PFeLaKCh5o71PFeLaKCh5+7n/kFxbRsnn17OOP2vz4MS/Kg
MqkPoo0lbDS/1b8XDajSzHcdt8ILksIxWVqeDHGQdDufmLMvzK9f/RW1rNfp/APn2MLqNRU09qERERGROuT
v58dZ/XpRVFxKTKTM4Ev0qCFJDFPnFfQFzFRj+omtlPn0a2mzDX6xqVcT9V1spiNIvi7zBr3NnBYZ1QKGhiIiIi
IhIHevdvTPhocKFbf29Bsgo5BpQZUJfVNmKmU81OLjS6jXrkr9xBmV8OMMlIjbEy52qYGKV4QecV4oOtoEhr
mV5qIiIiIiEgDlhAbTfd07cnMzq3BEfc3xfEc1RBFgasA8AlzlRpdRr3StLKBPRcOdrVfm5+SuS/NZ1VcNUk6WJTycO

```
PPPN7qMJkGhoYiIiIiISBOzmUyc0ec0TCYTtorqDS6CLE4S/bUktSkLxMZFrgVGl1EvTc052GBnGwJgNvH8OTl
8OkoNUk5G+N/HYPbzM7qMJqEBf5WJiIiIiIg0XNO6tKVlQhwZ2Tk1nu8SWFbHFUl9MjXga4JcWqZek072Qvo24
NmGv/q0Vx4vXBpGRYC/0aU0HBYLEZddZnQVTYZCQxEREREREQP4+fly1um9KS6puSFKC/8KonyqzOKUxi+QCv7
Oj0aXUa81+NmGv1jTqoj7JlgoaNYwG7zUteChQ/Bp3tzoMpqMhv8VJiIiIiIi0kD16taJyLBQcvMLazzfM7ikj
iuS+uD2gG8JdhUbXUa91tFeyOmNYLYhwOHwCqZcVc6BNtFGl1LvRU2caHQJTYpCQxEREREREYPExzSjZ9eOZOX
U3BCltb+NCKu9jqsSI/lRyaX8YHQZDcLdOQcaxWxDONogZeo/8tQg5U8E9u9PYK9eRpfRpDSOry4REREREZEGq
n+vHlgsFsptFdXOmUxwmmYbNim3BnxPqKvmmafiqYO9iP6NZLYh4G6Q8vHf1CClJs3+eYPRJTQ5Cg1FREREREQ
M1KV9G1q3SCA9M7vG82OCygm1VNVxVWIEKw7Gm74zuowG5Z6clEYz2/BXn/fM44VxodgC1SDlVwE9exIOYIDRZ
TQ5VqMLEBEREREREacp8fXOYflZ/Xp/9GXZ7FT4+nj+mmU1wWkgxSwsiDKpQ6srN/j8S7iwwuowGpa29mAG2JJY
HFBxzjD3fTsYnGZRsKcFZ6cQ31pcWE1sQkBRQ4/iS5BIOPHWg2vGOL3bEJ9wHgOyvsylaXORFegUmHxOB7QKJG
xuHX7yfe3z6h+kULCvA5Gci7u9xhJ8Z7j5XuKaQguUFtLq9VYO1rGlZzP1X+/HQp2FE5GjmaZRmGRpCoaGIiIi
IiIjBTu/ZjfmLlnMkI4vWiQnVzrcPKGdjcQglDvOI11iZcXC1+RtwGl1JwzM1N4VLmkfhMFX/4DlKHex/bD9Bn
YNodWcrrCFWKjIrMAf99ezE9k+2x+z/2zhr6G9ff6U7S4kcFklAmwBcDheZn2Vy4NkDtP9Pe8x+Zoo2FlG4spD
Wd7WmIrOCIzOOENw9GGuIFUeZg8zPM2k9tfWfPv9ogxQ7/5obTev9Nc9Ebgr8unQmZMgQo8tokhrXHF4REREREE
ZEGKDDAnxFn9aekrIyqqupLkcOm6KG9DRu1G/wXEumsuSGO/Lk29mLOtNW8t2H2N9n4RPnQ4roWBLYJxDfal5B
uIfjF+NU4/vesIVZ8wn3cbyazyX2u9V2tiTg7Av/m/gSODKDFdS2w59opP1AOQEV6BUGdgghICiD8jHDMAWYqs
ysByPgkg8hhkfhG+f5lDeW/NEhZ2S801/F8MBqhZjf80+gSmiyFhiIiIiIiIvVA/17daZkQR9ox9jbsGFhGoNl
Rx1VJXTDh5Frz10aX0aDdnZuCpYa9DYs3FRPQOoBDrxwi+ZZk9k7bS96ivOO6595pe9l5205SnkmhdE/pn451l
B/92rQEHW1g4p/oT/mBchylDsoPlOOqdOEX60fp7lJsB21EnRN1/C/ObOKFETl8dF7Ta5Di264tIeeeY3QZTZZ
CQxEREREREkXogJDiIYQNPp7C4BIejejhoMUF3zTZslK71XOyOM8voMhq0JHsxA20h1Y5XZlWS93MevnG+tL6rN
ZHDIkmfnU7+svxj3ssn3IeEqxNoeUtLWk5uiU+kDylPprhnEf6Ry+ki44MMAtsH4t/iaPOSkO4hhAOIY9+/9nH
4rcOOuL4FJj8Tae+mkXB1Ank/57H73t3sf2w/tiO243qNX5yWx3OXhWILrHkvxsYo+rbbMJlMfz1QaoVCQxERE
RERkXpiQJ/TSIiNJj0rp8bznTXbsFG63jzP6BIahak5KVhcf5iJ5wL/1v7E/T2OgFYBRA6JJGJwBHkLjz3bOC/
ej8ihkQSODiCwfSAtJrYgsFOgOd/X/HWZ/l46tsM2Em9M9Dgee3EsHZ7uQPvH2hPaJ5Scr3MI7hKMyWIi+6ts2
tzfhojBERx+4/Bxv8Z1icXcf7WZ/GZhx31NQxVw2mmEnqNZhkZSaCgiIiIiIlJPhIUEM3zg6eQXFuFwVG/qYDW
76BdaZEBlUluu9FtKnDPD6DIahVZVJZxtC/Y4Zg234p/g73HML8EPe679hO4dmBRIZVZlteNp76VRtLmIpHuT8
InOOeb1FWkVFKwsIOaSGEp3lhLYMRBrqJWw08OwHbS5lzcfj6MNUspJaRt9Qq+hoYm+8w6jS2jyFBqKiIiIiIj
UI2f27UlcdBSZ2TXPamoXUE60T/XwQhqmGy1fGV1CozI1Z7/HbMPA9oFUZFR4jKnMqMSn2bEDvpqUp5bjE/7bN
S6X62hguL6IpKlJ+EYfu6mJy+XiyKwjxI2Lw+JvweV04XIcbWviqvqlvckJdsOu93Nyz9h8lp/eOBukBA06m6D
TTze6jCZPoaGIiIiIiEg9EhkexpAB/cgrKMTprJ4kmEzQX7MNG4WxfqtIcB4xuoxGJbGqlEHlv802jDo3irJ9Z
WTNy6Ii8+hsv7xFeUQN+60RScanGR5LhHO+z6FoQxEVmRXYDttIn5106Y5SIodHusekv5dOwYoCEv+ZiNnfjL3
Ajr3AjrOy+tds/uJ8rCFWQnsd7fAc2D6Q0uRSyvaWkfNDDn4Jfu4GKifqpeE5fDQ6Boe1ETVIsViIvftuo6sQw
Gp0ASIiIiIiIuJpYL+eLFi2mqycPOJimlU7H+dXSZJ/OSm2ptMQoTGabJl7wjPM5K9Nzd3POoAYqkwOAtsEOvK
WlmR+lkn2l9n4RvsSf3k84WeGu8dXFVRRmfvb7F2Xw0XGRxnY8+2Yfc34J/rTemprgjv/Fkbm/XxOT8SUJ1M8n
tl8YnMizo747d6FVWTPy6bNg23cxwLbBNJsVDMOvnAQa6iV5tc3P6XX+OWPPA5GhDBlTgX+ZTU3a2lIwi+5BL/
27YOuo9GZMGECBQUFzJO797ivUWgoIiIiIiJSz8RERTJkQF8+++ZHoqMisFiqzyI6PbSIQzZ/HKizaENOke96W
joPGV1Go9SiqpTB5cEsCCwEILRnKKE9Q489/voWHu9HnxdN9Hl/vl9gt5ndjqsWa5iVjs91rHY85sIYYi6MOa5
7HI8NiSXcO8GPaZ+GEZld6LX71jVzYCDRt97itftlZ2czbdoOvvnmGzIzM4mIiOCO005j2rRpDBw4OGvPqSOnE
/R5k5Yni4iIiIiI1EPDz+pPi/hYDqdn1ng+xOqgW3BJHVcl3nKrdY7RJTRqU3P2YXU1rXlSaWEV3H51OfvbNdw
GKVGTrsca7b36x4wZw8aNG5k1axa7d+/mq6++YsiQIeTm5nrtGSfDbj+rj+xRjbj+xRjxGUWgoIiIiIiIiJSDOWEhfK3oWdRW
lZORWXNjU96BpcQYD7+rqtSP/zNdzNtnCl/PVBOWoKjjCHlQUaXUefKfZzc+498lvVveA1SfFu3JvLaa712v4K
CApYuXcpTTz3F0KFDadwWqFaeffjjr33Xcf//d//+cec9111xEdHU1oaCjDhg1j8+bN7ns88gzZ9dff53Ex
EQCAwMZO3YshYW/zeZcu3Yt55xzDs2aNMsLIzBgwezYcMGj1h/+9//+/L//9+Lj/+z+CgoJ4/PHHcTgcTJ4kaS
kJAICAujYsSMvvffn3SSx7NnzZrFl19+iclwmQmQysWQ+iclkwmQwiQM1MZO3Ys4eHhrEZGcuGFF3wAH3tQ6HgzvuuIPw8
HCioqKKjYOnUqLteJf0YoNBQEREREEamnzurXi07Jt7HTaknF/kj4Sd/Ed8/nbQJTcLUnH34NLHZhr+
aPiyHD86PxmFtOK8/btpDmH2P3YX6RAUHBxMcHMzcuXwOpqKiocw//vEPsrKy+O06/7fj29e/dm+PDh5OXlu
cfs3buXXTz75hHz5jF//nw2bt3LJOCL9/ZywiTDAmk/9dzQ+B1+T2ghS5BLsXRGOHhRuFbwKhUD7zFBhZhQw9vvr7pD8G8q
y7bXX4nQ6adGiB/Z9++ik7duxg2rRp3H///hO+B5v3bw0o61+YJoocGZwNeyBgj2hJcpTS57a+sYfRsZQwK2Qg1+gBLgYIvL0IORk
UtZvnw5wcHBjBo1isplspffsH03HPPPMXn5n++22WLVtGx4eX3zxQE/D/E/uk4kaGCioiLClwkNPTYexf
```

```
Uljm7Ouv8mXJ8LukYb3QJIiIi0oQZ/X2qNDwbtiXz0luziYtpRkhw9ZlTLhfMzWlGrt17P2xL7Rnhs423LP8xu
owm486Y7vwQ1HD3+DtVvVODGOSDlNDz/kbz55/3+n0///xzrr/+esrLy+nduzeDBw9m3Lhx9OjRg2XLljF69Gi
ysrLw8/NzX9OuXTumTp3KpEmTeOSRR3jsscc4ePAgzZsfbVgzf/58Ro8ezZEjR4iLi6v2TKfTSXh4OB988AHnn
38+cHSm4ZQpU3jhhRf+tN7JkyeTkZHBZ599BtS8p+H777/PY489RnJyMibTOT1tKysrCQ8PZ+7cuZx77rkkJCR
w++23c/cXairqqpISkqiT58+J7Q/omYaioiIiI1GM9u3SkX89uHDySXuPyMpMJzggtgga3GLFputNXexnWp
bty9zbZ2YZwtEHKPRNM5EWHGV3KMZmDg4m5995aufeYMWNISOvjq6++YtSoUSxatIjevXszc+ZMNm/eTElJCVF
RUe5ZicHBwaSkpLBv3z73PVq2bOkODAEGDBiA0+lk165dAGRmZnL99dfTvn17wsLCA0NpaSkhEOHPBsd9e3bt
1p9r776Kn369CE6Oprg4GDeeOONatf90ebNm9m7dy8hISMjI7HZbOzbt4/CwkLS09Pp37+/xqr1Vrj8/9
K0/3KERERERERaQDMZjOjh53Ntp17yM7LJyYqstqYeL9KOgaWsaus6e3h1pCc7bOLzo6dRpfRpMQ7yhleFsT8J
jzbMD2sktuvrmLa3Gja7s02upxqom+ZjE+M9zpJ/5G/vz/nnHMO55xzDg899BDXXXcdDz/8MDfddBPx8fHufQJ
/Lzw8/Ljvf/XVV5Obm8tLL71Eq1at8PPzY8CAAe6lwr8KCvL8+/mjjz7irrvu4rnnnnmPAgAGEhITwzDPPsHr16
j99XklJCX369GH27NnVzkV7sYkMaKahiIiIiIhIvZfUsjmDB/QlKzsXh8NZ45jTQ4sIVFOUem2q72dGl9AkHZ1
t6GNOGYYq93Fy3z/yWXpG/WqQ4tepIxFXXFGnz+zSpQulpaX07t2bjIwMrFYYr7dq183hr1qyZe/yhQ4dIS/ttX
9lVq1ZhNpvp2LEjAMuXL+fWW2/lvPPOo2vXrvj5+ZGTk/OXdSxfvpvPGY4Avr6+OBy
ef6/37t2bPXv2EBMTU63usLAwwsLCiI+P9wgfq6qqWL9+/Ql/rBQaioiIiIiINADnDjqT+Nhokz1Xjez+ziz
LCmO5uqvutv3Uc3x3ajy2iSYh3ljCgLMLqMeuHloTnMri8NUkwm4h5+GJPFUiu3z83NZdiwYbz//vts2bKFlJQ
UPv30OU55++mkuvPBCRowYwYABA7joov44YcfOHDgACtWrOCBBx5g3bp17vv4+/tz9dVXs3nzZpYuXcqtt97K2
LFj3fsZtm/fnnvfee4/k5GRWr17N+PHjCQj468n0NQw89xNqlaz3GtG7dmi1btrBr1y5
ycnKw2+2MHz+eZs2aceGFF7JO6VJSUlJYtGgd956K4cPHwbgtttu48knn2Tu3Lns3LmTm266iYCghP+GCo0F
BERERERaQCiIsI4b9jZFJeWUvGHZW+/ah1go7V//W540FTd6/cpJqOLaMIO2/A3X3XP56nLgikPMjZIjbjyCgJ
79aq1+wcHB90/f39eeOEFBg0aRLdu3XjooYe4/vrjBu3DgOHjxIbGys+
z7t2rXjkksu4bzzzuPcc8+lR48e/Pe//3WfnzFjrzSrr7ySW2+9lZjjWG59ww03cMkll3DppZfSv39
/cnNzPboyA1x//fVO7NiRvn37EhOdzfLlywkMDGTJkiW0bNmSSy65hM6dO3Nx4kRsNpu7wvqdqd97JlVdeydVXX
+1e+nzxxRef8MdQ3ZNribon11/qniwiIiJGMvr7VGnYm02nvr/9u47KoqrDQP4sw12Yekd6YwYDGFFewYd7+d7h1Ydj6f2h
oUWOLLdbYYoo9MZrYYzTGaGyJRpPYNWqMsWvsXQRFsAHSe9+d7w8+N64UUYEFfH7nBsnqpzuqtzYgN
PwRPKu6FFgmXSXG9mhrZAnsI1Je1JM8wA7Z5xCXq4Ghb59PrWpjvzJZ12GUG7bJepj9mx7MYxLLvG2Zqwvcdu6
EWC4v87ZfxezZs7Fr1y5cvXpV16HoBH+KEBERERERVRAKuRzdA1pDJpUqeEjV8OEjwZlJ/zsThuXA1
Lh70GNvQ40o42x8NCgN9zxKdvGMlxKLUeXrr8t9wpCYNQiIiIiIqpQ6tTwPZcDEYKhxWaxTC2Fmh8NCvfRoiIikauquCFT9wNMjhMuZy
ojXmMBuqrug6DAFiqMxGQxrkNn5clE/D5uwk44HDoKhbt4xaozfBpCENT
yD8UUSh5ZqZJEH01ZR1brr8d4hR8IrXJenEg1x0+TUd9otTIJqTjF13cgot++G+DIjmJGPZv1kvrfdJshrv78i
AxYIUKL5KhteqVFyM0L6vgmJU6PprOky+TobhvGQ0WpOKh0n/nf0kQ5kw/yYZjktTsPm6dly/38pBl1/TX/FsX
9+kuLvsbViAFS3j8HMXK+SW8gIpMg93WIOdU6ptlKTZs2e/tUOTASYNiYiIiIiKhxYPP6NAgA6WLaCuJgxY95HNnUA7+fayCvdHLl2VJyBDQbF0aZBLgQH8D3B6txOL2cpjJ/
zs2NF6N5uvTUd1SjGODDDHH9QyVm+OlD/v/c097gHGy5kYO/BhhiQVs5hu3NQGx6NUH/k9xJ+IpZG6EvAtH+yXhpzSbJUZ6E
DexsWaF/tBHzTvxQXSJFI4LBwIUR6eqVTP5U4Jg2JiIiIiIgqoIZ1a6FZwN/J1XNXc/AEwxJ+p4+DgSKlXBPdlcFZlwV5RdLy7SN
l3/DOhQNrO93/GQYW5rObrXKLwX3ZNkNcYdyMTmHgrIipEN+OZ0FhxNxFgfqEDjKhK4monRvqoUVc3/O3jaP5n
o6CHFgnZy1LeToKq5GF09ZbA2zCsTFKtGSxcJGtpLONdLBmN9EcIS8gbCTj2ciVENZXAyKdvUxOS4u9AXmLgqy
DX7VEwdLEKctWmJ12O5Zgzk1auXeL1Uepg0JCIiIiIiqoDEYjG8Gj15ZB2NwP3IDQw7VGf1
rMDTUgoAB0zPwsa8allLinXMnuBcNLSToNfv6bIL6q1Ox5lK2Vp377+aimrkYAb+kwXphCpqsTdUaGl3XR
oKLESokZAi4FKFCRo4Ad3MxTj3MxTj3MxeUoFcY3Kfvknbk6C++k6Zd5uxXFU+NsTBiYhrvVrEusTv369WE5ckSJ1Ud
lg0lDIiIiIiKiCsrKwgw9OrWBnr4MDx8lQajV+ADfm9/RGKmAuFxlqlL8OcAA3GkrTrbA3WhwPvWM3lVCjrxeR+/nsMxrdCLWh2BKNOYdHjKrPfV9MJ
cyGh7kYh943wKiGehh/MBMbr+Yl1DqPTBKRmA1+fzkKHqlL8NcAA3avL0GNbBo6H5517gLsU79eRodGaVAzenYG
N3RQw1ANG7c/ED50UWHUxB54rUtFsXRpuRZfdHJyT4u5Bzt6GhcqSCZjWMx7HfW0hiF4+lL1IJiZwXLYMIknxk
tVUfjBpSERERERERERERERERERERVIE1rlcbfo0b4NGTKgKWU3ZTJbL+Q3LkIM4Dn6qf3UdhsalCBW+PZeNDd0UEL1CAkgtAN5
2Esxrkzf0eEQDPQz31sMPl3I0+wEg0FOKiT76qGcrwafN9dG5/PNcjcXZL0e6NN8KNUUp0ryHD/JPZa0sqh
UwCzD2RhVNDDDCsvgwDd5Xdit9m7G1YLCv9Y7Gpi+VrL5AiiERwXLwYMpuS67VIZYdJQyIiIiIiogpIJBKhV3dfVPWww4WrMWWuetCvlaSS1bqcgl7YJUG34zTwUFazlsYc6g1LY4SmGvKkZJj0lfHWrfIYhwsoa
t4eJUBWGPnhRazsMgA9UN0sowsrfXdPl2yMpRL8OTD3MRnSbAaWkqpF8kQ/pFMh4kCZj8VxZclhXcQxUA7IxEq
GmlnTaoYSnWrIxsaSCCVAzUtJIUUEYosM47sSr8ciMHX7bWx7HwXPg5S2BlKEbvWjJ4rD5Li77G1
YDPtrJeDr/obIUBq88rGmQz+AsnmzUoiKygKThkRERERERERERBWcuakJvLj0Ff31ZYjJPMJa/b7Kp6mT4qSTVXYYX0EUvJ2pNcZIsDGp10ixzGhJMFCpPeDrDp
```

Alooz6t6zCODKgjw/VRhrj64X8f9kYifOyrhOPvF54IauYoQXCc9rD2kDg1nP+/cImeRIRG9gWUiVfD2SR/jOZ
BEDByXyaWtNeHUk8ElRrI+f+hzz6ryi5nCBN1Njqxt2GxXLdPw8eDgVgbO2IfI65dG3aTJpVaTFT6mDQkIiIiI
iKqBBp41UAHf19ERcciIzOrwDISEdDGLAF6nN+w1Hwu3wEZyn7hmdRsAVejVLgalTdEPSxBjatRKjxMUsPCQIz
a1hKtD5kYsFWK4Gn5Xy/BNpvSsOL8fOnliU318e9jFeadzMK9eDW23MjBj5ezMabRf73zPvbVw7abOVhzKRv34
tVYcT4be4NzMbpR/h58ay/nwMpAhC6eeSs8N3OS4p+wXPz7OBdLz2ahppUYpvI3nD/vFU2MC4Fczd6GxRFtVPw
FUlRKJaqu+h4iMdNOFRnfPSIiIiIiokpAJBIhMKAVGtapidDwh1CrCO4MGktV8DNNLNvg3hKWomSOFO7ppO2LE
SrUX52G+qvzhqBP+isL9VenYebRghPIBQmNVyM2/b/7plEVCXb2UeDXmzmo/XOqvjyRhWUBcvSvI9OU6V5Dhh8
6y7HgTDa8VqVi7eVsbO+tQHMn7Tnwnqaq8dXJLCx/R67Z1riKBJN99NFpSwZ+u52L9YGK1z3912aizkGXNCYNi
ytbmrdAytEiFkgRRCI4LFkMqZVVGUdHJUOkCEIZdv4tO8nJyTAxMUFSUhKMjY3LvPOdwZFl3iYVTw9PO12HQER
ERG8xXf+eSpVfZHQslvz4M2LiE+Du4lhouX+TjHEzTVmGkVV+Sww2oof6kK7DoFeUJJahnaMLMsTFT7ASOPGWG
d4/kABpjvb8ncoPR8JxwgTdBEUlijONiYiIiIiIKhE7aOu8F9gBErEYObHxhZZrbJwMa85vWGLMRanoKBzXdRj
OGvJ6G8peXpCO/FkrAfP7GSL9uQVSRM2bMWFYiTBpSEREREREVMl4166Ozm39EBOfgLSMjALLiEVAW/N4KCXlZ
5XfiuwTxW7IhUxdhOGvaUJcCBRqLoryqm7Yp2HKIAFPTBXIcagCjxUrdBOSlSAmDYmIiIiIiCoZkUiETq1boGl
9L9x/8BgqlarAcgYSNQLM47kwyhsyEqWjq3BU12HQGzASchGYKn15Qcon1jgHXw7Vg+umDZDI5S8/gCoMJg2Ji
IiIiIgqITO9Gfp16whXxyq49+ARCpvO3kyWi7bm8RCjUk53XyamyvdCIaTrOgx6Qx/F34UBexu+Mqlagm9aLYa
RvYOuQ6ESxqQhEREREFRJWVlYYb+3TtCoa+PpzFxhZaz189GC66o/FoMkIkeOKLrMKgEKIVcdEvl3IavRAAmu
49HU/dmuo6ESgGThkREREREJWYV3UPdAtohfikJKSkphVazsMgAw2MksswssrhY8V+GAqpug6DSshHccEwVHO
IbXG9a9IZ77f4QNdhUClhOpCIiIiIiKiSa+/ng5ZNGyH8cQQys7IKLVffKBWeBoUnFkmbHNnohb9OHQaVIAOoO
D1VouswKoTmogaYEfiVrsOgUsSkIREREREUSUnlUrRv3tHNKpTC3fDHiInt/AVk5uZJMFBn6sAF8dExQEohRR
dhOElbFxcCHsbvoRzgg2Wvvs9xGKmlSozvrtEREREREvAUMDBYbO6YYa7q4Iuf8AanXBKyaLRUBrswSYS3PKO
MKKRYZcvIdDug6DSoEBVOiZwt6GhbHLscNPAzdDbmCg61ColDFpSEREREE9JawMDPBOPe6w8HWBvfCHha6orK
eWECARRwMxaoyjrDiGK84BBMhUddhUCkZGx8CpVqh6zDKHbNsM6wIWAEbcxtdhOJlgElDIiIiIiKit4ijvS2G9
AmEkdIQDx5HFlrOUKJGR8tYGDBxmI8UKrwvOqDrMKgUKaDCuykiXYdRrhhmGeIbn29QzbGarkOhMsKkIREREREE
ROVumpocbBvTsDEFQIyo6ttByJlIVOjFxmM8o+d8wU8frOgwqZWPY21BDP1sfO2pPgO9tH12HQmWISUMiIiIiI
qK3UJP6Xni3YzskJacgPjGpOHImUhV7HD5HBDUGi/frOgwqA3KoOYu9DSHLkWGi6OROums61CojDFpSEREREREER
E9BYSiUQIaOmLd1o3R+TTGKSmpRda1vT/iUMFE4cYIT8KC3XhvTOpchkdHwKjt7i3oSxbhrH2Y9G3dV+IREygv
m2YNCQiIiIiInpLicVivNuxHfybNkDYoyfIyMwstKypVIVOFnFvdeJQBDWGivfqOgwqQ3KoOTtF11Hohl6WHoZ
ZDsOggEEQi5k+ehvxXSciIiIiInqL6enJ8H6PzmhS3wuh4Y+QkZlVaFlTWS46vsWJw8Hyk7BWR+s6DCpjo+Lvw
vgt622ol6mHASYDMKzzMEgkEl2HQzrCpCEREREREdFbTmlogOF9e6BxfS+Ehj8sMnFo9hYnDkeK9+g6BNIBfaj
R5y3qbaifqY/eBrOxMnAk9GR6ug6HdIhJQyIiIiIiIoKROpCJwyLO1z8NW3WkrsMgHRkZfxfGagNdh1Hq9DPOE
SgNxKjuo6CQv129Kyk/Jg2JiIiIiIgIQF7icNh73YudOOxkEQelJLcMI9SdUZLdug6BdEgfavRNVus6jFIlz5C
jm6wbxvYaC2Olsa7DoXJAqusAiCqrmzdv6joEKkTt2rV1HQIRERFRuWVspMSw97pDEAScv3IT7q5OUMj1CyxrK
stFV8tYHIo3R1xO5R3G+K7+eTioH+s6DNKxkQl3sdXYEOniwlcar6gMUwzRzaAbPuz1IUyNTHUdDpUT7GlIRER
EREREWoyNlP8fqlwboeEPkZlVeI9DA4kanS3i4KBf+MrLFdO4yU5dhODlgAwC+lXC3oYmCSbopeyFOb1HM2FIW
pgOJCIiIiIionyeJQ4b1auNe2FFJw5lYgHtzeNRzSCtDCMsG13OLsNZ/UDXYVA5MTzhLkwrOdyGlk8tOcusFOb
OGsEhyZQPk4ZERERERERUIGMjJYb17YGGdWu9NHEoFgF+pknwNkouwwhL3wTpDl2HQOWIDAL6J1WCBYAEwP6xP
d61fxdDew2FkaGRriOicohJQyIiIiIiIiqUiZESw/v1RKN6tXE37CFSO4qez83bKBV+pgkQQyijCEtPgOw6qqr
v6zoMKmeGJd6DmcpQ12G8NrEghlO4E7q5dsPgnoOhNFDqOiQqp5gOJCIiIiIioiKZGCnx4fu9OLpZYzx4Eon4x
KQiy1czyEB783jIRBV7/rfJett1HQKVQ1IIeD+5Yq4aLlPL4Bbqhu41u2Nwj8EwVFTc5CeVPiYNiYiIiIiI6KU
MDRQYOisQXdv5IzYuAVHRsUWWd5BnobNlLAzEFXMoZ2vZbVRT3dV1GFROfZB4D+YVrLehYbYhqt2rhr4+fTGo+
yAo5ApdhOTlXIknDefPn49GjRrByMgI1tbW6NatG4KDg7XKZGZmYsyYMbCwsIBSqUTPnj3x9OlTrTIPHz5Ep06
dYGBgAGtra3z88cfIza2YmXwiIiIiIqLKQE9Phj5dAtA38B1kZGYh/FEEBKHwYcgWslwEWsXARq/wuRDLqynsZ
UhFkELAgOQcXYdRbOap5qj1sBYGBwzGux3ehUwm03VIVAGUeNLw+PHjGDNmDP79918cPnwYOTk5aN++PdLS/lt
Fa+LEidi7dy9+//13HD9+HBEREejRo4dmvOqlQqdOnZCdnYOzZ85g48aN2LBhA2bOnFnS4RIREREREdErEIvF6
NCqGYb17Q49PRnuhj2AWl34MGRDiRqdLOJQ2zC1DKN8M82kIaihCtJ1GFTODU4MLf+9DQXAPsYeXnFeGN5zOAJ
aBEAs5qBTKh6RUNS/hUpATEwMrK2tcfz4cfj5+SEpKQlWVlbYsmUL3n33XQDAnTt3UKNGDZw9exZNmzbFgQMHO
LlzZORERMDGxgYA8MMPP+CTTz5BTEwM9PTOXtpucnIyTExMkJSUBGPjsl82fEdwZJm3ScXTw9OuTNq5efNmmbR
Dr6527dq6DoGIiN5iuv49lagk3Qy+hw2/7UbEOxhUc3OGVCotsnxYhhwnEk2RI5TvpMVuw3moq+Lv8/Ry6Ozds
dQsW9dhFEgmyOD4wBGeck8M6jEItTxq6TokqmBK/UmdlJQ3Qa65uTkA4NKlS8jJyUHbtm01ZapXrw4nJyecPXs

```
WAHD27Fl4eXlpEoYAEBAQgOTkZNy6davAdrKyspCcnKz1QURERERERKWntqc7xg3pC3cXR9y5F4as7KKTJ66KT
HSzioG5tPwO62wovQ8vVcF/dxK9aHDiPVioyt/qwya5Jqh6pyrqmdbD6PdHM2FIr6VUk4ZqtRoTJkxAs2bNND1
7oqKioKenB1NTU62yNjY2iIqKOpR5PmH4bP+zfQWZP38+TExMNB+Ojo4lfDZERERERETOImcHe4wbOg/1a9fA3
fsPkJqWXmR5E6kKXa1i4KEoupyufK7/O8Qo1QF5VImIAQxOKl9zdlbJqAKH2w5o4tEEY94fA5cqLroOiSqoUkO
ajhkzBjdv3sTWrVtLsxkAwGeffYakpCTNx6NHjOq9TSIiIiIiIgKsLMwwemBvtPRpiEcRUYiMji1ygRSpCPA3S
ORzkORIylGCrq7kAeqprus6DKpgBiaFwrIc9DaUCTLUiKkBm/s26Ni8IOb3Gw1rC2tdhOUVWKklDceOHYt9+/b
h6NGjcHBwOGy3tbVFdnY2EhMTtco/ffoUtra2mjIvrqb87PWzMi/S19eHsbGx1gcRERERGVDSOlIYb17YH+3
TsiJycHd8MeQKUqfIEUAKhumI4uVjEwkuSWUZRFmyb/g7OM6ZWJAQzRcW9DK5UVaoTWgEWyBfp37Y/+XfvDQGG
gO5io4it6ltrXIAgCxoObh507d+LYsWNwdXXV2t+gQQPIZDIcOXIEPXv2BAAEBwfj4cOH8PHxAQD4+Pjgq6++Q
nRONKyt87Lihw8fhrGxMWrWrFnSIROVihCZha5DoEJwGRQiIiKiOiGVStGxdQs42tti884/cftuKKo6O8JAIS/
OGEtZLrpZxeBMkglCM3SX5KgpeYKGqis6a58qtveTQrHBpCZiJGW7SrhIEKF6ZnVI7OrhaOeIfl36oY5nnTKNg
SqvEu9pOGbMGPzyyy/YsmULjIyMEBUVhaioKGRkZAAATExMMHToUEyaNAlHjx7FpUuXMGTIEPj4+KBpO6YAgPb
t26NmzZoYMGAArl27hkOHDmH69OkYM2YM9PX1SzpkIiIiIiIiKkFe1TOwacQA+HjXQdjDx4iJSyivL5YQCuzR
LQ1i4dCrCqjKLVNk/8OMYruGUlUmLzehpll2qZSrUST+CaQhkjRoFYDTBgOgQlDKlElnjRctWoVkpKSOLJlS9j
Z2Wk+tm3bpimzdOlSdO7cGT179oSfnx9sbW2xY8cOzX6JRIJ9+/ZBIpHAx8cH77//PgZYOHIgvvviipMMlIiIiI
iKiUmBtYY5RA3qjZ8e2SElNQ+iDR1CriO7KuSgyOdM6GlXLeJEUDOkUmqoulWmbVPkMSLoP6zKa29AlxwU179e
EKlqFLq26YHS/ObC1Kng6t9IiEomwa9euYpd3cXHBsmXLvt4XdiwYUO+hXzfJiKhqNlpK7Dk5GSYmJggKSlJJ
/Mb7giOLPM2qXh6eNqVSTu8B8qvsroHiIiICqLr31OJypogCLh4/Ta27j6AJ09HqBHO9vZceF54hx+kkE2S
oJaUe4wbDFWipOlPq7VDlt9nYDV9blN4cnQpBgTppdZARmgFba1vObN8TPvV9IBKJSrSdqKgofPXVV9i/fz+eP
HkCa2tr1KtXDxMmTECbNmO0ZczMzIo9IjQmJgaGhoYwMMibiEmHnzp3o1q1bicZekjjIyMpCSkqKZOu9tU+J
zGhIRERERERE9IxKJOKhuLdhZW2Lzzv24cjMY9rZWMDc1KfI4FOUmbPWzcLaU5zpOEceguep8qdVPb5f+yfex3
rQWnkpSSrZiAXDPdYfdUzskxyejoVdD9OnUB/bW9iXbDoDw8HAOa9YMpqamWLhwIby8vJCTk4NDhw5hzJgxuHP
nDoDCF6otjWVYnHWKys7OhBQKKBQKKBQKHQQVfleQaqsnExERERERET3jYGeDcEPc1ePnUP6oXMbPyQnu2
EPkqoqev1BeBnMdTpP/ASnKx+rNVDkMTSzZ4fXGamOOzGgJ4xBjCNkC3uv8Hsa8P6ZUEoYAMHroaIhEIpw/fx4
9e/ZEtWrVUKtrVUKtWLUyaNAn//vuvt9fX3xySefaNUTmDmUyGEydOAMg/PoZAOMg/PPllWrZsiXHjxmxmHChAkwMzODj
YON1qxZg7SONAwZMgRGRkZwd3fHgQMHNMeoVCoUOHToUrq6uEgUPPTxLfffqtV+8DBg9E9e3Xse3h6e
nJ8LDwyESibBt2zb4+/tDoVBg8+bNiPT4nmgSbBt2zb4+/tDLpdj8+bNN+bN8TPvV9IBKJSrSdqKgofPXVV9i/fz+eP
```

ImKNDKx8OHJ5mpztM1qi4ZpDfE49DHSM9LRrW03TBk6Bd61vMs8YfjMypUroVKp0LhxY2zfvh13795FUFAQli9
fDh8fn0KPMzQ0RLdu3TBjxgwEBQWhb9++ZRh1Hg8PD1y8eBGHDh1CSEgIZsyYgQsXLpRo/Tt27MDVq1dx7do19
OvXT9PLEQCqV6+Otm3bYsSIETh//jyuXLmCESNGQKFQaN7Ptm3bwsfHB926dcNff/2F8PBwnDlzBtOmTcPFixd
LLNaSwKQhERERERERlRvPeh2OHdIXbk5VcDfsAR5FRGn9YV4YB3kWeljFwMckEfrighdWMRWloZNwrISjJipYj
5QHqJKr3WPWUG2IptlN0TajLbIjshH6IBQ1qtbAxMET0btjbxgri+5hW9rc3Nxw+fJltGrVCpMnT0bt2rXRrl0
7HDlyBKtWrSry2P79++PatWto0aIFnJycyiji/4wcORI9evRAnz59OKRJE8TFxWn1OnxTS5YsgZmZGXx9fdGlS
xcEBARozV8IAJs2bYKNjQ38/PzQvXt3DB8+HEZGRpDL83pCi0Qi/Pnnn/Dz88OQIUNQrVo1vPfee3jw4IFmjsb
yQiS87iyX5VxycjJMTEyQlJQEY+Oy/4bbERxZ5m1S8fTwzL8CUmngPVB+ldU9QEREVBBd/55KVJFkZGbixLnLO
HjsNCKjYmBrYwkLM9Ni9cDKVotwK8OQN1KVyBb+6y8z32AL+qr3lWbYRFp2GTlhhiVgoDZArdxacM11RWJSIiK
jI2FjYYP2zdujVdNWkOsXb3g9VSyPHz+Go6Mj/v77b7RpO0bX4bwSzmlIREREREE5ZJCLkeAvy+8a9fAoeNnc
OLcJUTH3oeTgx2UBgZFHqsnFlDfKBU2yXdwLSIFsfYtIBOrESgcLaPoifKOSYvHEWU7WAtVkZmeieCIYCgNlHj
H7xOEtAiAtYW1rkOkEvTPP/8gNTUVXl5eiIyMxNSpU+Hi4gI/Pz9dh/bKmDQkIiIiIiKics3Kwgzv9+iEpt5e2
Hv4BC7fDIJEIoZTFTvovbC66ouigy+gpjoBTb1Noc5Kgjir4GHLRCUtWWyEs4ZNcNWgLiyy1QiLDINapUbD2g3
RqVUnuDu562zeQio9OTk5+Pzzz3H//n0YGRnB19cXmzdvzrcSdEXApCERERERERFVCO4uThj/QV9cun4be/8+g
bv3H8DE2Aj2NlYQi/NP2R8b+Qii1KeoWscLCikAqQl2G4yAR+Y1VM+4BAMhtexPgiq9RIkJzho2wTVFHeQKIkQ
8jUBSShI8XDzQ0b8jGtRqAIlEouswqZQEBAQgICBA12GUCCYNiYiIiIiIqMKQSCRoXN8Ltat74PjZizh4/DRuh
4TC2tIcVhbmWj23Ht+5DBOFDI7OLpptKpEMdxQNESKvB7esW6iRcRFG6sSyPxGqdO5lmWDdIz2I6vaAQm6M6Nh
oxCbEws7KDoFtAtGiYQsYKIoeVk9UnjBpSERERERERBWOgUKOd1o3h3edGvjn9HmcPHcZt4JDYW1pBisLc8RHR
wApkXDzqgVRAbOQ1SIp7snr4p5+HdjlhKNa5lXY5YRBjEq5ViiVEhUkCNevjhB5fUSqTXDpxn6Ir1+EmZUtrM2
tOeudXvBr6AcLMwtdhOrOypgOJCIiIiIiogrLxtICfQPfgX/Thjh29Thj29gJOnb+Cm8H3kP3kFszOpXB2cSu6ApEIk
XquiNRzhaEqCR6Z1+CWdQNyIbNsToAqpHSRIe7J6+KuvC6yxAZQq9XISIiGm6k5norF6N6uO/wb+3ORE6rQmDQ
kIiIiIiKiCs/exgr9unVES59GOHrmPI78ehNWJqZIjHOKUOubAuc8fFGaxARXDf1w3cAXzlnBqJZ5BRaqp2UQP
VUUcRJbBCu88UCvGgSRBGq1GonRkUiKj4GJuSX69BmBGg2awcbeSdehEr0xJg2JiIiIiIioOrC3sUL/7p3Qrmk
dBFO6jZvnjuNhyEOYmVrAzMoW4mIsQKEWSREmr4UweS1Y5ETCI+saHLNCIENOGZwBlTfpIkOE69dAuH5H5NJEqtA
AC5uTlIiIlAWlICTCys0KJjb3g19oOZla2OoyUqOUwaEhERERERUaVjbecI687voZ5vG9y6eAr7/DR/6DR/duQ19
hCHMbe+jpy4tVT5zMDnEyO1wwbAOH7HtwyboD25xwSKAu5TMgXcqFFI/13BGmXwtRMicIoryeqpkZaYiLegJVb
jYsrKugcat0qOHtC1MOQ6ZKiElDIiIiIiIiqrRMzK3g27476jRpiZDrF3Dj/HE8fRQGtVoNMytbE3MtFZcLox
KJMMD/Rp4oF8D+up0OGaHwCXrDqxyn+DlR1NFIAB4KnVEmH5NPNKvhlyRXt52QUBKQhwSY6MgkcpQxbUa6jRpi
aq1vKEwVOo2aKJSxKQhERERERERVXpKEzN4t2gZYt8SDkBm5f0oOw09cQ9/QxlCbmMLO0hURavD+Rs8QGuCe
vh3vyejBQJcMl0wjOWXdgpoot5b0gkqaGCHFSWzzWc8cDvepIlxhr9qlUuUiMeYqUpDgYGpmidN/1GrYHE4et
SApxjB3ooqOSUMiIiIiIiJ6a8jO9OBeuwGq1vJGJGF87j1sRJj3FY0wW1FExj
nxqFKzn3YZ4fBKvcJxBzCXC7lQIYomTOe6FXFEz03ZIkNNPsEQUB6ajISY58iNycb5la28Ov0HjzrNYGlrUOxe
qUSVRZMGhIREREREdFbRyQSwcreGh7ty7j5vkTeBIWgujHmTA0NoWJhRVkevrFrjNZaoFkqQWCFIO
gVWfBLucB7HPuwy47HAZCWimeDb1MuliJJ7K8JGGUzAlqkXY6JDszA4lx0UhPSYLC0AiOVWughrcvPLwawEBpX
EitRJUbk4ZEREREREETOVlMYKlG7sR9qNGiGiPAQ3A+6hpCr5/DOOX2oVWoYmZnD2Myq2MOXASBXrI9H+tXwSL8
aIAgwUOXDPjsM9jn3YZEbBTGEUjwjyoEMsTJ7REsdEKHnigSpTb4yqtxcJMXHICUxDhKpDJa2DmjSpgtcq9eFj
YMLexXSW49JQyIiIiIiIiIAEokEjlVrwLFqDfi064ZH924j+OFVSCBCpKGGCBpEdnCmaGxwPJEqh4THBgQsGpkkbx7OhBFQpk5dEXXDgT7g0W7gB/4vH2Z8L3K1Ry
2SJDa4BaaQqrOgmVuJKxyn8AyNwKWuVGQCdmld2JvgSyRHDHSKoiWOSBa5oAEibVmxePnqdVqpCYlIDk+Bmq1C
ibmVmjUshOq1qoPB7fqkOnp6SB6ovKJSUOiCk6lUuG3FYtxYs92JMbGwMzaBq2698a7oyYU+J+x1bM+wV/bfsa
Qz+ag86Dhhda77btF+G3lEq1t9q5V8d2Bk5rXCTHR2LTwS1w/cwIZaamwd62KniM/gk9AJwBATnYWvp8+BReOH
IKppTWGz5qHur5+muN3/fQ9YiOeYNiMr97OMhARERlSg9fTmq1vJG1VreaP5OLzy8exP3bl5GePB1PLp3G1K
pDEpTcyhNzCCRvNqf1rlifUTpuSBKzwUAIBLUMFHFwio3AlY5eYlEpTq5FM6q8kgVGyNWaocYmQ0ipQ5IklgAh
fQMzM3JRkpiPFIT46EW1DA0MkG10o1RrW4juHh6cfgxUSGYNCSq4Hat+wfDmanjUZ0/0/ha07J0JvXsOKzyfCQGm
ETgOHaZU9d/gAQq5dgrm1bbHqdvTwxKx12zSvJVLtFcK++2Q80lKS8en3G2BkZo5T+3ZiycSR+0aPA3Cr6YXD2
37B/VvXMW/rXlw58Q+WTRmDdaevQyQS4enj/j7ty1YsP3Am18EIiIiIqJSZKA0QvX6Pqhe3wfJ8bF4cPcW7t2
8hCdhIXgSFgJBLcDQyBhBKU3Poyw1eeVirIBIjUWqNRKk17srrAQDk6lRY5kbCJDcOJqq8D2NVPCRQlcIZl1+5k
CJRaolEiRUSJVZIkFohUWKJHLG80GMEQUBWZjpSEuKQnpoi1VgCIzMLeDVtCedqtVHFtRpMpkOZH4jMzK04/JjoJZg0Jkr
ggq9cRKM2AWjQsrBESf378K9G1eysU9jcTaudMxY+0WzBs5oFh1SyQSmFlZF9721YsYsPutreNSpDwB4B4d
9QE7N2wBvdvXYdbTS88vn8PDVu3h5OHJ2wcnbBp4ZdITGoiHibkFfpz9KQZMmQYDpdHrnTgREREkQ4Ym1vCq4k

/vJr4IzUpAZEP7uHx/WDcD7qGhOhIZGdmQKYvh9LEHEpjU4glkpdXWoBMsRKP9TzwWM9DsOOkqKFUJ8IkNw6mq
tj/JxNjYaRKgKSCr9ScJZIjTWyMVIkJkiQWeUlCqRVSxKaF9iB8nlqlQnpqMlIS45CVmQF9uQKmljao69sGjlW
rw97Z45VWxSYiJg2JKjzP+g1x+LdfEBEWCnvXqgi/cwt3Lp/H4E9na8qo1WosnzoegUNHwcnDs9h1Rz4Iw7AW9
SHT14dnvQboP+kzWNk7/Nd2vYY48+ceNPBvA0NjE5w5sAc52Zmo1dgXAODiWRPH9/yBrMwMXD11DGZWNjA2M8e
JvTsg09dHk3bvlNh1ICIiIiIqa0oTM3jUaQSPOo3QolMfREc8QGT4PdwPuoqoR/fx+H4UBCFvoRUDpTEUSmNIX
jOJCOT1SEyRmCNFYo7H8HhuhwC5kAYDdSoMVCkwUD/7SNV8VqhTddZLUQUJskRypEuMkSb+/8cLX+eKXmOuwdz
cHKSnJCM9JQlZGekQiURQKI1g7+w099oNUMW1GmwcXF9p8Roi0sbvHqIKrvuIsUhPS8H4jn4QSyRQq1ToN+FT+
HXpoSmza81KSCQSdBowtNj1etT1xtj5y2DvWhUJ0dH4feViTH+/O5btOQqFMu8/dJOXrcbiiR9icNNakEil0Jc
rMPW7n2Dn7AoAaN3zPTwIuY0JnVrCyMwck5f9gNSkRGxdvhBfbPoDW5Z9g9N/7oaNozPGzFsCCxu7kr04RERER
ERlRCqTwd7ZHfbO7vD2COBKQhwiHtxFxINQPLp3GOnxMYiPjoAgCJDp6cNAaQwDIxPI9PTffJisSIRMkRKZYiX
ipYVMRSQIOBcyIBOyIBOyIROyIdV8znluew6kQjbEUEEkAPj/Ks8iCBAgglokhoC8D7VIBBWkyBHpI1ssR7ZIP
+9rUd7X2eK8r9WiN0895GRnIz0lEWmpycjJyoRYLIaBkQlsndzg5F4TNlWcYVXFmcOOiUoQk4ZEFdyZA3twcu8
OTFiOEo7ungi7cwvr583SLIgSevM69v+8Fgu3H3qlH57efq01X7t41kS1uvXxYevGOH1wD9q+2w8A8Ou3C5Cek
oxZ67fB2Mwc5/8+iMUTP8TcX3bC2bMGpDIZhs+cr1Xvis8moOOAoQgLuonzRw5i8a6/sWvtSvwOdwamfre2ZC4
KEREREZEOiUQiGJtbwtjcEtXr+0AQBCQnxCI28hFiIh/hyf0QPHOShtiIh8jJyYZYLIHC0Aj6BgaQKwwhlemVf
OJLJEKWyABZMCjZekuYIAjIycpERnoqMtPTkJWRDkFQQSKVwdDIBM7uteDoXgPWVZxgbe8MQ2NTJgmJSgmThkQ
V3KaFX6L78LFo3qkbAMDZswZiIx5jx4/foVX33gi6dA5JcbEY2bqR5hi1SoWN38zBvo1r8MM/54vVjqGxCexc3
BD1IBwAEPUwHAc2r8fSvUc1Q55dqtfC7UvncHDLBoyc802+Om78exqP7oVg1NzF2LTgS3j7tYbcwAC+73TFgc0
98pUnIiIiIqoMRCIRTMytYGJuhaq1vAEAmelpiI16jJjIR3j6OAxP7ocgPTUJyfGxyM3JBgBIZXqQKwyhrzCAv
sIQUpmsUiXIVLm5yMxIQ2Z6GjLTU6HKyYEAAXp6csgNDGFpWwW2jlVhbm0HS9sqsLJ35pzoRGWISU0iCi4rIxM
isVhrm1gsgaDOG0bg37Un6vi00Nr/5bB+8Avsidbd+xS7nYy0NDx99ABmXXv+v92M/7eVv221Ov8kzNlZmVj75
ef4aOEKSCQSqNUqIDcvRlVuTt5rIiIiIqK3hNzAEA5unnBwy/sHvFqtRlpyIpLiopGUEIukyIPVEIukuGjERD5GTORDpKc
kISkuGrm50QDyfueW6elDpqcPqZ7ef1+XRg/F1yQIAlS5ucjJykR2diZysrKQnZ2J3OwsAAIEARBLJJArDKEwV
MKxqiesq7jA1MIappY2MLWwgcJQWW7Oh+htxKQhUQXXsFU7bP9hOazsquQNTw66ibObVqN1z/cAAEZm5jAyM9c
6RiKVwszSGlXc3DXbZg/ujcZtO6Dj+x8AADZ+MwcNW7WHlbOD4qOjsG3FIojFYjTv3BOAUMXNHbbOrvhh1lQMm
joTRqZmOP/3QVw/cwKf/bApX5y/f78M3n6t4cb3/bBsRNi38Eq169MGBzetR3btRvmMqups3b+o6BCpC7dq
1dROCERERkYZYLIaRqTmMTM3h8Nx2tVqN1KQEJMVHIyk+FskJsUhJjEdSE+KQ1ZGG1KR05GRnaXooiiACx
CJIJFKIJZK8z2IJxBLJ/19LIP7/NolEkq8TAiBCXmJPAIS8zwIECGo1VKpcqHJzoVLpcqcqHJzof7/Z9Vzn9UqVV7bECAg7+8
OPT05ZPpyGBqbwsHSGqaWtjAyMYOhkQkMjExgYmENYzPLN1oghohKB5OGRBXcsOlz8evyBfjxi8+QHBcHM2sbt
OszAL1GT3yleqIehiMl7zOu5pJJZOHo2UxAQYm1ugRoNGmL9tHOzMLQDkTfQ8bfXP+GXxPMwfNQiZ6WmwdXL
F2K+/RQP/Nlp1Pwy5gzMH92LxzsOabT4BnXHr/FnM6N8d9q5VMWHRyje4CkRERERElZNYLIaxmQWMzSzgWFV7n

HXqVE299+7dw/bt27Fjxw5IJBLN9tmzZ+Prr7/GsmXLIJVKcf/+fa14QkND0aFDB8yd0xfr1q1DTEwMxo4di7F
jx2qSl4MHD0ZERASOHj0KmUyG8ePHIzo6uoh3iyoq9jQkIqoEMlJTsW7eTIxs3Qh967rh8/e64N6Nq1plHofex
fxRgzCgoSf61a+Kqe++g5iIx0XWe+bgXox7pwXeq+0KiV1a49LxI4WWXT3rE/Ssbo99G9dotuVkZ+HbqePwfoN
qGBvQHNf0nNA6ZtdP32Ptl9Ne/YSJiIiIqMxJpVJs2LABGzduhKmpKZo1a4bPP/8c169ff+mxU6ZMQadOnVCtW
jXMmTMHDx48wL179wAA3333Hd555x1MmTIF1apVw+jRo/HOO+8WdwsePFi90jRA66urujRowcmTpyI1at
XF3nc7t27tXpFFqRObTqYNWsWPDw8MHDgQDRs2BBHjuT9HnzkyBHcuHW7ZsQYMGDdCkSRNs2rQJx48fx4ULF
6BQKKBUKiGVSmFrawtbW1soFIpC2zI3N8fy5cvh6emJDz74AJ6enkhPT8fnn38ODw8PfPbZZ9DTO8OpU6c0x0y
fPh2+vr5wcXFBly5dMGXKFPz2229a9WZnZ2PTpk2oX78+6tSpo9ner18/DBkyBG5ubnBycsoXz/z589G/f39Mm
DABHh4e8PX1xfLly7Fp0yZkZmYiJCQEBw4cwJo1a9C0aVM0aNAAP/30EzIyMoq8plQxMWlIRFQJfD9jMq6d0YH
x33yHJXuOoG4zf8wZ0gdxTyMBAFEPwzGtXzdUcXPHnE1/YMnuI+g1egL09OWF1nnn8gUsnTwa7ti0U7/0Ljt
h2wYOwHeBhyJ1/Zc4cPIOTaJZhba8+HcnjbL7h/6zrmbd2Ldr37Y9mUMRAEAQDw9PFD/P3bFvSb+GkJXgkiIiI
iKk09e/ZEREQE9uzZgw4d0uDYsWPw9vbGhg0binzez87u7/tcSc3nocsVIjiQsnkTOxBw4uHnqB5+tudcedyf/a6+Or4FkEnFdl0gIhiUOIDjvOzFSmaiarM0PNrJRPrf4q/fOJu57UoLDS+1tMvDaM5DDF7s/EcUwYyUcUz/Idu/NqmrNWaTUchp3LDAK2Ms8Q1XeRJqrahgAR15tH97VoFzKrw9PewxBiDpUPdXb7LuYaSrYN8vQVQCE0A1NAKmbnEJ+7NG/PdYKKPYTEJwM3mvgTM4fEyTyrOg4n19yVLtNr3mV7W5X9HrVe8jqOHPg31lKnUNl3VjUFZdXS0cVaJKVJt5QPuj5Aaa6I1CtNfGIo2nSIo4VX3J9J3g1JvkGn0s8L/hsRb66mXZKdS5dX8jBK5Pl6Jgen/9Vy1fFKeuF9JFMsuaQq/Ri8V/8CCDymVxtzgFn2K6YbZntmHvGd2ZZbHHPCcTlrjszJcZvMvyvMGlj3m4oxBmtFWMyQbg3iEWM8OnbX2Ah2zc3nXsjRJtWxe2GV2jumBkV2Y8IDdG4EZT9zRrdbPtmnz4yjQ7+vnpcVr6VcW3Sjc3AGs3CIVUixhpEcRR2eE3PThEBsuNmnT5oXZprN9Sb8bx5nFtEfXR7TBgbGCxSCb4iRXc9ZKOwIa8Y69t7j3oKcmqXOeQ5RDB7T4BGzVsiy0aFqPUG+Gtnp4n6j8s/9DD1qbYoV8vTkHsmFmPUqUYGUsfsPpUtPS3B7mi6Mu4HV6rT95qYSP1kkcDuvgFaxcfdOc2Qn0cNQ6EZqXUP7gUoqsa4sgH2bhWG1FMNomLm3lfXThGBTjwsJ0Jxb5X1pzbQbj3HgVB3uHEhQCaaoMKB3hmL2dLzOZLFY4vTyrE8XdmTUZ2Gkv+D1vMxbjCKhOVuH1jZGLcEY8a2zjqEn4GZ7cvlFa/e0jHMbEu3UQAbqLgO1rX7LClDZi9Ri5fRwhYvJFeiFFRBuJN8W8H3YTVSwcaQjyVwo8lFgEk6dUdZvOHoREEGFZdjsKhYzTuAO3rjI1ax/VTF3zWz2E3XEbd5ZiMc6BuckyHpXuJMGw2kO0DsNmRdC5NH9SVHF8jTZwmxbd9xqxnDSqahHoVcVbfDdqr2lW+nKxjvanNqK7nPjmA+KgjfbRWHQj/N+t3BMmdAoFdK75BW7VqX7LWwBWgZDPaB3dCJeS53YhCSGBs6ZDH6cK2C9pztHJTrgmoFiTpeG+2M61xGCNLGjhgAgrc6EiahUhBoyxEg4Z9nWEkNPZmh5sgGAG0pT0zScH5t63t+9/fuh3NS9Y0M2e+TxfhOaNu6zaLtPXeQQ0eoAuP5Fl1s0NCCpPrCAnuDu6NhxhKAEFEwJ4i7eTOu5DhjTpzgvG7eoe4ykh/Y5Hb3/eMQdZ0HQx+SnDcsvtoQ1t99XN5IUAlRS8snSONdGSpNKCnnqjdXCMMEqvEqHBkodZeKt0VmRPP5WcS5x9RlOmYm/tWiOk9HHF1AHn8vXcsyWyCaD4j4gx9JdK7ZUp8vPGZp6HIlqjqjjnUMpjbsbQa2/O8n3N95qMLVqaA4xpjEVQuQhcZ3J0xnGqq5+Ykp8ZyeUp7ZG3SWqTdeeihYHBgIfJsdV3BRbFzrC49AW+KStwFVdiBjEqDn2T8VxOpz2FYjaP7wX8tZ4JNlVg0hyVHKP8y4+eJ2CZTLS4YIW9pi2Vr4bMhRMSyr5qiSzIXZjvNqu8S2jLvbMSLUyXRfeN6w+oX8O5+TTLWXxIdZvXQtkIuDQ2QR2E0BK+dFMnsOGNKfPMl8MDlQl8xTcBFkl5ksF3HDmfFIRqlOr3/AMmC7vZtqUX8hpQB0wvRLsUPJBv/xE1bnvPp8KU4HyWQ8b8HhDbrzJyE4R0oaMBaMxZ8WkQ1h9N8/yxIZbLF/OsC5B6wRL03LlDkWf0Nx2jsYfW6YGP97+lMR9UxAAFMp3tf5DmXt8iUaivaK7yupUMhKKnDG2c9mT0bq8HAaWw4hm9fPWm8Lb0Bv2Kp04tn0nqaMUJfhV/aLGYvijXMUAFNYnK+QmK6D3ZvmNWMcfqzRA28HMjXv5j5uMaf0d7YV+jTOgT8jPIbN2CzE5pMYzYcPaLbTDmkUCRmRSp1yqxxg3c0fdoVJzaBK04pM6DRqsRtK8fWuU7ikBPn9k2+htrSgmBPDAVMo+w+mnnAM0MgwV7xEb8Jm2UFMdwdWs8T5PYCqaY2kRUSwb0VGWO3uwHaSyKmWHhYYYQCzI+NxsCwApP1PkInxl3eKbBcqDLnVGt6M0twxTCXqAZXs+2hp/Oko/Cz6NNiUtX9R+QKtN8efdUKkPMnvK0Y5d2jGybJ0T18z+PO8ADFjjVEXqgwahFkeiwOdw+1h7FqJ6oC04O+m5kE4mxgP72v/7WnMMCgTZJPv3uuG1P0tfz9SOM7gsUMyZL4gjrYtJ7o99y2cHcJYqIhTv6HeLdXTgB6x4Si/9sgTIFgFP6JMJ0Gn4uhq9H6ufxKcr/zSBHo+wdUw+HqS8XxPVL+fodMFHOyiaYwkrY6xz5m+tZeMM

mDYmIKhG5gQHkBgZITUrE1VPHMWDKdMj09OBeuy6ehGmvJBcRfh9W9g6F1lWtXgNcP3sSnQf998vb9TMn4Fkvb
6U0/649UcdHe87DL4f1g19gT7Tu3idffdlZmVj75ef4aOEKSCQSqNUqIDfvFxRVbk7ea3pt3UeMRXpaCsZ39IN
YIoFapUK/CZ/Cr8t/ydj6LVqiaft3YF3FCVGPwrFl6deYO+J9zNu6t9Cefqf278L92zfwzR9/FiuO5IQ4/L5qG
dr2fl8rth9nf4rR7X1gbe+IOV8tRkT4fRzb+Tvmb92D1bM+wdXTx1G1dl2M+nIhDI2M3+xivKVeNkRdEAR8M+Y
DSGRSfPr9eigMldi74UfM+aAPvt13HHIDgwLrVZqYoueHH6GKmzukMhkuHvsbKz+fCBNzS9Rv0RIAYO/ihmEzv
oKNozOyMzOxb+OP+HJoX6z46wxMzC20VlK/cuIfLJsyButOX4dIJNKspL5ge9HzrBIRUeW2aNEitGvXDoaGhjh
w4AA2btyI77//vsTqb968Ofr27Vti9RG9DZg0JCKqBK6cPAZAgL1rVUQ9CMOmhV+iips7WvfIS94FDh2NJZM+R
M2GTVG7iS+unDyKiOcP44tN//OHdPkn42FubYv3J38OAOg0YBhmDuyJPet+gHfLNji9fzdCb13Hh18sBAAYmZn
n66EmkUphZmmNKm7ueNHv3y+Dt19ruNX0AgBU926ETQu/RKsefXBg83pU926U7xgqvjMH9uDk3h2YsGglHN09E
XbnFtbPm6XpbQYAzTt105R39qwBZ8+aGNPOB7fOn8mXAAaA2MgnWDdvJmau2wo9/aJXGQSA9NQUzBs5EI5Vq6H
P2Mma7YZGxpi4WPuX/lmDemHg1Ok4sXcHnj56g08OnMSqGR/j95VLMfjTWa95Fd5uzw9RBwBrBOec3L9LMOQ9M
vw+Qq5dwtK9RzULF42Y/TWGNq+LU/t3om2v/gXWW7uJr9brzgOH4diu33Dn8nlNOrBFF+2ewoM/nYOjf/yKB8G
3UcenBVdSJyKilzp//jwWLFiAlJQUuLm5Yfny5Rg2bNjLDyymqVOnlldRG8LJg2JiCqB9NRkbF4yH3FRkVCam
qJpu47oN/FTSGUyAECTdu9gxOyvsePHFVj31QzYu7rh4+VrUKNBEOOdsRFPIBL9N9Vtde9GmLBoJX5d9g02L/0
adi6umLpiHZyqVX/l+B6G3MGZg3uxeOdhzTafgM64df4sZvTvDnvXqpiwaOUbXAHatPBLdB8+VpMYdPasgdiIx
9jx43eapOGLbB2dYWxmjqgH4QUmDUNvXUdSXCw+7hGg2aZWqXD74r84sHk9tl4P1/RQzEhNxdxh/SA3NMTUFT9
p7r2C/LN9KwyNjdG4TQcsGDcUjdt2gFQmg2+Hzti6fOEbXIW328uGqOdkZwMA9J6biFOsFkOmp4egSxcKTRo+T
xAE3Pj3FCLCQjFg8rQCy+RkZ+Pwtl9gYGQMl+p5Q9FdPGvi+J4/uJI6EREV6rffftN1CET0AiYNiYgqgWbvdEW
zd4qeo6VNz75o07PwIRlf/Lw93zbfDl3g26FLseMoaB5DIG+xjJWHTmttE4vFGDFrPkbMKnpOHCqerIxMiMTa6
5uJxRII6vxz1zwTFxWBlMQEmBUycXWdpi2wdM8/WttWfD4RVdzc0X3YGE3CMD01BV807QeZnh4++35Dkb0Sk+L
j8Pv3SzF3yy4AeUnI3P8vmJLLYepv5GVD1Ku4ucPSvgp+WTIfH875BvoKA+zb+CPioiKREPOOyLrTUpIxwt8bO
dnZEIsleXMSNvPXKnPx6GEsnTwKWRkZMLOywax1W2FsZgGAK6kTERERVURMGhIREVUCDVu1w/YflsPKrkre8OS
gm9i7YTVa93wPQN4KyL+tXAyf9p1gammNqEfh+HnhXNg6uaJe85aaemYP7o3GbTug4DPT1sfQKFU5utZKlcYwMjUT
LM9PTUFXwzti6yMDHy08Dukp6YiPTUVAGBsbpFvrsT182aiy5CRmqRQde9GOLH7D9Rr5o/Dv23mMPU38LIh6lK
ZDFOX/4Tvp0/CoCY1IZZIUMenBer7tQYKmBj9eQpDJRbtPIzM9DTcOHsKG76eAxshZ62hy7WbNMOinYeRkhCPw
79vxuIJI/H1b/thYmGpWUn9eSs+m5BvJfVda1fip7kzMPW7taVyjYiIiIio+Jg0JCIqqSGTZ+LX5xuqBg+G/qBuHvtMgBgT
Hvtue9W/XOOlg6OmtdXTh5D1MNwjF/wnWbb0/2H4N7N6/i0lGTk5uT
AxNwCn/buhKq16xRZt1gshp2zKwDA3bMRGzjz5kqvtKT+4/E4raSg3MICdsyvsnF1RrV4DjAlohiN//IoeqlVk4QxOXwgB
TlqOu1vFUtFcZOv5sheqI8PsIvXkN743/+JXaEtRq5P5/jsSiyuRkZ+XbzpXUiYiIiCoGJg2JiIiIiKoGXDVEHg
DMH98LYzAKW9lXwMCQI676aiUZt0mgNUX9xJfUdq79D1dp1YOPkgtzsbFw+fgTMZGeamZ6O7T98i0at28P
UygYpCfE4+OuYjHvtbK4ui6da9gCH7+/TOJnc35NvVOjRuvUUcicxKKYjsmiiaPRgd59BBhHLe/fw3QHxW6fn0yKRuSnLI5cbgUewWlsiPwwawKeWUcmBKJswZlOt03F/3GZgZRr72NiZfnLLmjZOMjeprcXqCwLwGVcEwlOlkOcFj3jbAwu1z9/MYtAJy3ZeS7Sro0s9xiH9l+Nzl3a4U9fWRFa+rW5r9Z8rbg/RkQmQHePlyxsxumSRPdeEkIP2HTKHvgUIthkfaDv0EYHDMGmU5WDs9+T2V+cxQqnGAE+q3UVuV6eSG/pTLvj+Uy2P7VIYxbBlhD+s8mi+4wG1sM0BpvlR1bc/tbW/JYIOD1JrOoCwgZvjJFXNrTGS02PbE+Ue1WOVVEPDjYG82OUwuJ0ZbzH0N8gSfuRHSTRfY+nx3bsMTiRUcDgfdWnvNxQ==

zVlIiMj8c477+gwyqINGTIE06dPL7X6K1KC09fXF5GRkTAxMdFpHFFRURg3bhzc3Nygr68PR0dHd0nSBUeOHNG
UedX7qiJ+L/bp0wchISG6DqPCKM7zqDxjoq/s/fDDDzAyMkJubq5mW2pqKmQyGVq2bKlV9lmyOM7Orlw8J4mIi
IiofJDq0gB6c460jti6dSuWLl0KhUIBAMjMzMSWLVvg5OTORnXn5ORAJpNpbcvOzoaenh6srKzeqO7yrmfPnsj
OzsbGjRvh5uaGp0+f4siRI4iLi9OUsbW11WGERVOpVNi3bx/279+v61DKzLN780U5OTnQO9PT+fsVHh6OZs2aw
dTUFAsXLoSXlxdycnJw6NAhjBkzBnfu3AHw6vdVef1eLOr9UCgUmucVvVxnke6UNDPCCofWrVqhdTUVFy8eBF
NmzYFAJw8eRK2trY4d+4cMjMzIZfLAQBHjx6FHk5Fk5MTPD09dRkyEREREZUz7GlYCXh7e8PPR0E7duzQbNuxYwecn
JxQv359zbaDBw+iefzPmMDU1hYWFBTp37ozQ0FDN/vDwcIhEImzbtg3+/v6Qy199BXt7e80fFS8
OiUxMTMSwYcNgZWUFY2NjtG7dGteuXdOK9euvv4aNjQ2MjIwdOhQZGZmltJVeTOJiYk4efIkvvnmG7Rq1QrOz
s5o3LgxPvvsM3Tt2lVT7vlhoM+u344d09CqVSsYGBigbt26OHv2rFbda9asga0jIwwMDNC9e3csWbLkpUO91q5
dixo1akAul6N69Er4/vvvX3oOZ86cgUwmQ6NGjQrc37Fubk5bG1tMXv2bKOyDx8+RGBgIJRKJ
YyNjdG7d288ffoUQN4QtTlz5uDatWsQiUQiUTYsGFDgW09u4fmzZsHGxsbmJqa4osvvkBubi4+/vhjmjJubw8H
BAevXr9c67pNPPkG1atVgYGAANzc3zJgxk5OZr9z3o6rl27Fq6urpo/gEUiEVatWoWuXbvCONAQX331VYHD7
k6dOoUWLVpAoVDA0dER48ePR1pammZ/dHQ0unTpAoVCAVdXV2zevPml17Ooo0ePhkgwvnz59GzZ09Uq1YNtWr
VwqRJk/Dvv/9qyj1/X/n6+uKTTz7RaicMzg42NDdasWYOOtDQMG
TIERkZGcHd3x4EDBzTHqFQqDB06FK6urlAoFKhevTqWLl2qKRcaFVaCxcDgbGxsoFQq0ahRI/z
9999a9UdGRqJTp06a92PLli2v9RyqaIrzPHrZeT/7Xlm9erXm+d07d28kJSVpyly2cAHt2rWDpaUmOO/vj
8uXL2vFUtD31svujujdmzZ2Pjxo3YvXu35llx7NgxAMCjR4/QvDcrHIGBgQgPD9/PQu3dvmJqawtzzcHIGBgQgPD9ccq1KpMGnSJM3Pral
TpOIQhFK4ypWPp6cn7OzsNNcayOtRGBgYCFdXV61nzrFjx9CqVasCn5MbNmyAk5OT5mdWQYnqVatWoWrVqtDTO

```
WaN++vWZodFhYGM6ePYv+/fu/Vn3PPP/eP7v2z78fNjY2AKD1fqxcuRINGjSAlZUVlEolfvzxx3zvR2HPipe9H
6mpqZgyZQpq1KgBU1NTKJVKBAUFaeoPDg6GVCqFt7e35hh3d3eYmZlpXhfnOVRRFfU8Ku55Ozk5oUqVKprXPj4
+UKvVCA4OBgA8ffoUw4cPh4eHBOxMTGBsbIzU1NRi/Ywozr3xomvXruHevXswMjLSxGxubo7MzEyEhoYiKSkJk
ZGRWj83pFLpS+8l+k/Lli2RlpaGCxcu4OTJk6hWrRqsrKzg7++vmdfw2LFjcHNzK3AO5OL83A4KCkKzZs2OtjV
r1kzzzc8LU1BR169bFsWPHcOPGDejp6WHEiBG4cuUKUlNTcfz4cfj7+5fwmRMRERFRSeBCKJXIBx98oBlauXLLy
nz7u3TpAmdnZ6xZswb29vZQq9WoXbu2ZhjYMwUNnSxo2/NSmWcoDuVyOdu3aoV27dpgxYwaGDRu
GWbNmFbki9fOLAjwbGvziEPDiSk1NBZA3D+KLf7hJJJJCj9uzZw/atWunmeOvOLECefG+bqwvU1BbRbX/LDE2Z
84cBAQEwMTEBFu3bsXixYu1jins3izOPTty5EiMHz8+3z4nJ6cSX9nXw8MDIpFIs9jJq+jfvz/Gjx+P7777Dlu
2bIGXl5dWgu91vOz9ePHe3bp1K6ZMmYLFixfDx8cHRkZGWLhwIc6d06dVz+u+H1OmTMHhw4exaNEiuLu7Q6FQ4
N133833fCpKZXOOPVPY82j06NElct6DBg1CXFwcvv32Wzg700NfXx8+Pj4v/RlR3HvjRampqWjQoEGBc4WW18V
9Khp3d3c4ODjg6NGjSEhIOCTn7O3t4ejoiDNnzuDo0aNo3bp1qcbRsmVLHDt2DPr6+vD394e5uTlq1KiBU6dO4
fjx45g8eXKptk9ERERERr4dJw0rk2TxQIpEo35xhcXFxCA4Oxpo1azTzoJO6darE2vb29kZUVBSkUqmmB+OLatS
ogXPnzmHgwIGabc9PxF4R1KxZU7NAxevw9PTEhQsXtLa9+Pp5NjY2sLe3x/3791+pZ9nu3bsxVsSI144TyHu/H
jl6hEePHml6G96+fRuJiYmoWbMmAEBPT08zT1VJO3PmDJydnTFt2jTNgcPHpRY/d7e3rh9+zbc3d0L3F+9enX
k5ubi0qVLmsVkgo0DtRYIeBXm5uYICApAjypUrMX78+HyJl8TExIEITPIGBgRgxYgQQOHjyILVu2aHOPlZXTp0/D1
9dXaw68kuy9d/r0aQwePBjdu3cHkJdQen5BDE9PT+Tm5uLKlSua3p/37t1DQkKCpkxxnkMvetXnUHPP++HDh4i
IiIC9vT2AvOevWCzWLG51+vRpfP/99+jYsSOAvEVjnl+Q7n+ytrWFsbFxg3XZ2djh37
hz8/PwAQPM9+XyPUyraswVOEhIS8PHHH2u2+/n54cCBAzh//jxGjRpV4LHPfm4/78WfcbNDp06cxaNAgzbb
Tp09rfk4AefMarlu3DlKpFB06dACQ1Oj89ddfERISwvkMiYiIiMopDk+uRCQSCYKCgnD79u18Fyy98vdDMzMxgYWGB
3/8Effu3cM///yDSZMmlVjbbdu2hY+PD7p164Zff/0VDXp164a//voL4eHhOHPmDKZNm4aLFy8y8CAD766COsW7c069evR0hICGb
NmoVbt26VWAwlKS4uDq1bt8Yvv/yC69evNetzl6OozLm82eS3BDbA16sDPS
4pcrGTOnDmYP38+li9fjpCQENy4cQPr16/HkiVLCiwfHR2Nixcvaq1W+TratmOLLy8v9O/fH5cvX8b58+cxcOB
A+Pv7a4YHuri4ICwsDFevXkVsbOwrLxpTFA8PDzx8+BBbt25FaGgoli9f9fXqILIHzyySc4c+aMZoL/u3fvYvfu3
Zreup6enujQoQNGjhyJc+fO4dKlSxg2bBgUCsSVrt7ly5UqoVCoObtwY27dvx927dxEUFITly5cXOVTfONAQ3bp
1w4wZMxAUFIS+ffu+dgyvy8PDAxcvcGTNmFJnwfp36ny1mc+3aNfTr10+r12v16tXRtm1bjBgxA
ufPn8eVK1cwYsQQIKBQKzfdPcZ5DFdHLnkfFPW+5XI5BgwbhAH2rVrOHnyJMaPH4/evXvD1tYYWQN578PPPPPyMoKAj
nzp1D//79i3W/F+fecHFxxEcHIzY2Fjk50Sgf//+sLS0RGBgIE6ePIImwsDAcO3YM48eP1yyI9NFHH+Hrr
7/Grl27cOfOHYwePfq1E/dvq1atWuHUqV04evWq1jBgf39/fd9/rF69GtnZ2QUuggggIA48ePx8GDB7Fo0SLcvXsXK1a
sOFr1GAA+/vhjbNiwAatWrcLdu3exZMkS7NixjTGm3RKbN2+GnZ0OdqlWrVvInT
kRERERvjEnDSsbY2LjAHhtisRhbt27FpUuXULt2bUycOBELFy4ssXZFIhH+/PNP+Pn5YciQIahWrRree+89PHj
wQDM3Wp8+fTBjxgxMZGD58OFasWPHa9TZr1gw//
PADlixZgrp16+LgwYOYOHFikcOBIhw0bhrVr12L9+vXw8vKC78/NmzYAFd1wLL7927F40bN843p+WrEolE2L1
7N8zMzODn54e2bdvCzcMFt27Zt05Tp2bMnOnTogFatWsHKygqq//vrrG7X5vK5du2LixInxYO3Ys6tWrhzNnzmDG
BklVn+dOnVw/PhxhISEoEWLFqhfvz5mzpypWZSsoP/BgxAtbW1q/dppubGy5fvoxWrVp
h8uTJqF27Ntq1a4cjR45g1apVR7bv39/XLt2DS1atChw7rHSNnLkSPToQN9+vRBkyZNEBcXp9Wz7E0tWbIE
mZm8PX1RZcuXRAQEJCvN9mmTZtgY2MDPz8/dO/eHc0HD4eRkZHm+6ej7u703r06IGOHTuiffv
2qF0njtYK9j/99BMSEhLg7e2NAQMGYPz48bW634zmDzejhcPGYnPXc54o2LAhrKyscPr0aRgYGODEiRNwcnJCx49U
KNGDQwdOhSZmZman2OTJO/GgAEDMGjQIM3Q52e9Ual4WrVqhDyD/7+/khJSYGnp6dm/tOXNW3aFGv
WrMG3336LunXr4q+//9LO6dO1ynTr1g3ffvstFvY1ffxuY9v0mQz2OUtimh5aeBl8LMzvnGLk6Crhpr
9Wcz5CIiIioHBMJrzs7PxGViOHDh+P0nTs4efjKidTXtWtXXNG/eHFOnTi2R+ojKq8ePH8PRF//SEBERERVRCc05CojC1atAjt2rWDoaEhDhw4gIObN2r19nlTzZs318nwVaLS9s8//19//yA1N
RVeXl6IjIzE1FEu9vbEj7r7PWGOb5JJllZOTg88888oLLbY9lIdTe8gGcRw4XKThIjRLb+ZCWGHj2ZlemIdqyFyZoPrHUahnuXLl2PYsGElVj97GFJlZOTg88
//xz379+HkZERfH19sXnz5nwrQRMRERERERERERGb4/BkIiIiIiIiOsKFUIiIiIiIiIiIiIiIiEgLk4ZERERERERERERERER
ESkhUlDIiIiIiIiIiOsKkIRERERERERERERERERWlh0pCIiIiIiIiIiOMGlIRERERERERERERERERWpg0JCIiIiIiI
iIi1MGhIRERERERERERERERERERERZEWJg2JiIiIiIiIiIhIy/8AmUmndu+7mgkAAAAASUVORK5CYII=\n″
       },
       ″metadata″: {}
      }
    ],
      "source": [
        "analyse_categorical_column(\"Marital_status\")"
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "1. Most people are married (67.76%)\n",
        "2. People with civil marriage have the highest approval rate"
      ],
      "metadata": {
        "id": "o38jdeJSDgd3"
      }
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "urBrJuA8ypnn"
      },
      "source": [
        "#### Education"
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "analyse_categorical_column(\"EDUCATION\",90)"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 507
        },
        "id": "s450TEdo4ElP",
        "outputId": "35a74570-fa6a-4f2e-e8d6-63f06709b9f9"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 1400x500 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAABPoAAAHqCAYAAACdlCCQAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAADqSEl
EQVR4nOzdeZyN9fvH8dc5Z/Z9YTYGYx/7GiKEUJKtJCRlaRNaLH0rZC8poZTIEP3aKUtFClki+zbZd2YG+s8z5
5zfH5NTk7GPOTPj/Xw8ziPnvj/3fV/37Zhmrrk+n8tgtgtVqtiIiIiIiIiISJFmtHHcAIiIiIiIiIiIiIiIiicuuU6BM
```

RERERERERESkGlOgTERERERERERERERpBpToExERERERERERKQaU6BMRERERERERESkGlOgTERERERERERERERpBpToE
xERERERERERKQaU6BMRERERERERESkGHOwdwO1isVg4e/Ysnp6eGAwGe4cjIiIiUuxZrVaSkpIICQnBaNTvk0V
EREQKWrFN9JO9e5bQ0FB7hyEiIiJyxzll6hSlS5e2dxgiIiIid5xim+jz9PQEcr7R9PLysnMOIiIiIsVfYmIio
aGhtu/DRERERKRgFdtE36Xpul5eXkr0iYiIiBQgLZsiIiIiYh9aPEVERERERERERERKQYUKJPRERERERERESkGFC
iTORERERERERERpBgotmv0iYiIyNWZzWaysrLsHYYUIY6OjphMJnuHISIiIiJXoESfiIjIHcZqtRIVFUV8fLy9Q
5EiyMfHh6CgIDXcEBERESmElOgTERG5w1xK8gUEBODm5qaEjVwXq9VKamoqMTExAAQHB9s5IhERERH5LyX6RER
E7iBms9mW5PP397d3OFLEuLq6AhATE0NAQICm8YqIiIgUMmrGISIicge5tCafm5ubnSOourSZOfrO4qIiIgUP
krOiYiI3IEOXVdulj47IiIiIoWXpu7mk71799o7hDtOjRo17B2CiIiIiIiIiEihoYo+ERERkUKsZcuWDBO61N5
hiIiIiEgRcMMVfevWrWPKlCls27aNc+fOsXjxYjp37mzbb7VaGT16NJ988gnx8fEObdqUWbNmUalSJduY2NhYX
njhBZYuXYrRaKRbt268//77eHh42Mbs3r2b559/nj///J0SJUvywgsvMHz48Fu7WxEREbmi7w6cK7Brda1y8x1
bN23aRLNmzWjfvj3Lly/Px6hEREREIq2G67oS0lJoXbt2nzwwQd57n/77eZPn06H330ZEs3b8bd3Z127dqRn
p5uG9OrVy/27dvHqlWrWLZsGevWrWWPgwIG2/YmJibJ6ayZcuybds2pkyZwpgxY5g9e/ZN3KKIiIgUJ3PnzuW
FF15g3bp1nD179rZfLzMz87ZfQ0REREQkP9xwou/+++9n/PjxdOnSJ9VquadOm8rzzz7hnxv5J9VquVadOm8frrr9OpUydq1arFggULO
Hv2LEuWLAEgMjKSn376iRTuOaNWvGjBkz+OKLfRTqlevTo8ePRg8eDDvvvvvsrurd2
tiIiIFGnJycl8+eWXPPvss3ToOIGIiAjbvjVr1mAwGFi+fDm1atXCxcWFxoOb51pHyIiAh8fH5YsWUKlSpVwc
XGhXbt2nDp1yjZmzJgWMBv76669c8b7
33ntUqFDB9n7v3r3cf//9eHh4EBgYy0OPP86FCxds+1NSUujTpw+IiIiIlK85esafceOH
SMqKoo2bdrYtnl7e9OoUSM2bdoE5Ey38fHxoUGDBrYxbdq0wWg0snnnzZtuY5s2b4+TkZBvTrl07Dhw4wQFFxcXH6
GLCIiIkXIV199RdWqValSpQq9e/fm008/xWq15runtupMnRl7e9OoUSM2bdoE5Ey38fHxoUGDBrYxbdq0wWg0snnzZtuY5s2b4+TkZBvTrl07Dhw4QFxcXH6
4+nR48euc5x+PBhvv32W7777jt27tyJxWKhU6dOxMbGsnbtWlatWsXRo0d59NFHAahcuTINGjRg0aJFuc6zaNE
ievbsCUB8fDytWrWibt26bN26lZ9++ono6Gi6d++eK/a1a9fy/fffs3LlStasWcP27dvz9RmKiIiISPGVr113o
6KiAAgMDMy1PTAw0LYvKiqKgICAY3EE4OODn55drTFhY2GXnuLTP19f3smtnZGSQkZFhe5+YmHiLdyMiIiKFzdy
5c+nduzcA7du3JyEhgbVr19KyZUvbmNGjR9O2bVsAIqzXXn/CbWsrCxmzpxJo0aNbGPCw8PZsmULd
911F5AzXXfBggWWULFkSgFWrVrVrFnzxnezFN69ejFz5+5ejFs5kzGjRsHR5FT5bdu2jYU
LFwIwc+ZM6taty/rniFxERERG5HsWm6+6kSZPw9va2vbn2a8xHlKpkQBBaj8gjaFllxebF3Xu5B2Ty9Dn9Mc2P8ov7HAwOUY
y4iIiLFw4EDB9iyZQs9evQA8uPPLwkcffZS5c+fmGtekSRPbz35+flSpUoXIyEjbNgcHBxo2bGh7XbVqVXx8fGxjbbd1udscenYoeOxR
aU7ZsWVuSD3KWHgkNDc31UW1atVyHdejRw+OHz/6/AeRU89WrV4+qVassCsGvXLn777Tc8PDxsrvv7jhwlw
pEjR8jMzLQlIP8dv4iIiIjI9crXQF9/fn5OTk3Y3aeM/N39HzZvrbeZXzx92cbLYS9X10rK/U+42sY/21vHh19Ph7v6FLR3+T3i7AW/OQ3x0Y5GmM1Oz9kXzcfZyeF/u1vNHuaRHLzdzYPG2/fDRKiERA+SomWVX4fSAhccLsTaq2aKGKOqm+Bqd3PkuF2F9+UyaeTv1LW0uu+KhGGRU9FZuOk57E8AfGmZrcH43cqBy5gnVGu+ks7Y09HSwyOOiSw8CtdTJQj8rjbGmmJiFU3VKnhmWEUWVb8DDZGQ+pc95uzh4sp0cUtLA5z7Vjn8nwOzkW7pPoFVNc0c+FxqHbM/yUxx/VFVJiFkC7HD7umP+wPpc8lWuGzPfZzLa4MPvQaUcpfvSwpLvzTmYhcgqHAEmmmTcFwaKsbUcMBPeMr3b6yoXxvwPT92w0aLYMEPlxj7FsZ9NJMppNJRp2ZAQYvMNJRpCUO9QAB0XDN9n6mxj0b8mwwNQbEnu6n9+hzLBNvLpHMexZQi57uI65MmNRspfT3HmPJdFFx0WuvJH5gq8UBIkW8r1nBgqrtmM0HphPqMAASAlPhFVIbFRmvs8tB2uGnpo0Hbe3bb/Qss9RJTSMJBmHvrJH0j1gdQlM4AR3Ivwks3Lr9fLlbQxPjUsPONBzWFaSYQWh0XGYq/n/+/ZL4CsMxZgVv6YHmBJzZ2sdQ2TiI9mMxI9aD5Mc13gw8WdsDrdcNv+1e2g5bhHs9d7vfZtknbmWcUwhpgA4NVFcX5TLGqZJ0rYVQHsXvG3a7rNG1HaM2jSGfgRaOUj1RmnbHTQYjMv8p2K4rrc2aKSpAGdOcp0mtAgYI1YbBnNOl4kWQDHzaX7XcWPqdOeB1bVrdjH5bXiyiInMNzRTtfZG+CnHThq2N2zcbWtLXQoKmD96AwKZp7G1ha7hHQm17D4jjVN8ZtqGTNP0OXZTaN5SW2gfd6a4OXEddBHg0BvW6+7e8+iAUPjtMAwnC4+xdBxv9ts8+qT3x7b7aCdrqj8t8++YBEGk6MeOy8oCsTN3PsBHDDVnfDvKtA8Vch8lRy5kyzNs/3HQKNu/6M0z8EpmXlq4bb3JWcMuAEqtjHvF0W3rhFxAQ/S0d3eznMp+ahs4eJ7PYmZV2iuA5SuM5nMMmPAyGaLhvhFlMbTf4YPBlfKcN8NmIfEJg9v6c6ZY/YKhwbfMRbKHLFUQ9efv5ESBUwaRPAX0yfZJfd9Z6FoGYVQ4FRdqKYWmkJubxA60G6v+HXtK0TZPL4Xre+2lXGMeSXeUUkWrcvNF/dgdT+gGFG0DkW6pOcomo9a8Kh/jU4Rq5IcdrhF5z88s/zv+zP/mA+0jwZHxHo4o9SEWs0IMlrSDsmbTe6wJRM0Lc7u7fHZuhqJJP8ruk9fmQ1PqGGJ3zZAlTWqxsjlHDYQaVW2JbqKyWrT8u9W3Ho0pBWQ+pLCtO7yJ/dA+/BCN1wo3FTqG4H+Hu5p6bfQ7ZJW7gxYOQfoIDxHDgxl8aOLPRsJ3AwbnxFBHN0V2vS3j6bJyt+pHmk/AYYW+UsWGfN5cF4oXlqTF2kD2n03f3j1Mmw6+dH1hQnHEg5Dkl2PxC0h+FwT3n4S4A3TEFnyHXTiSk0Fm3uhsmGjRy9KTQ3ZZHtIe3l23XJfazNpnBZkPuXKuCzoMvnyfSPp3TdLdX6h0dW4jDpxfTAUfHW2x9U3FmvgYcYiggJM48N9ehbY5/zb3+qdLj0u8sKDoq5cA23X+R9F1A9m7GMb8fh9fXMySChTkhQyqn+jpp8KSXeEJ9VcGW3MG4uA8h4mSZ/yTXBXPb0XRDIENqwJtK1Dq9Mwa9QWgnDHC3jaTrbDSaxK1aIjXmh+Qc5L6F1kuUeCkwNeVb7/n8D5/2c1+cxUOXJ6eRr9oxB3XZ+tpYxyuyk9eiPzVX8XyR73+QHcm4pdTMmWVfsnN6vmx/XM16h94mVE8pXUNcRM+yTO+LdVGTQJwdnAVuAvI9nWbiX/W/W6xl6xevI3wrWRxqM9nvUhQgKHNWDWMZ+SkeOQ3g9KPMxVOyKEHWzV8/95bImKv+3fWmpx3hnuYh94KbdFQNK3jgQ47YzeDLyy+0SfvOxKSe8+DeAYRrp9nCTJJ6/EBB0T2/NYXEIQRqWr+BWSUcCTHJe96v0KDyR+84/CDNeS24NeFbTDlOgtD+0W/4zmtkgD7LB7xE3Oq/Sz3dTpFI0GXhsgMs8x9FF3oEgDaeANSmmVFaspMJHYJ93lcEWIsmnBqHvdBv3mSsUN1IdUXZjz+B47oEGBBOvnNYgTdSvK0qNqWLHSrlPfI6vjU8MhVmUC8QkUOzjnuKN4jUAIKZvH5bs+lTxUENdVeXs+OzXgVTd3qq5EWRjcGb8EbjA5a7nC7yLxLvZWwLDb3jMdCDhyWGFPEhJJhb2OF6CWGLx9AnsCLLSXnWFmiZkr+o6KgL4Nm3tYtXt/5MvCa4bP8UGyB3Pn9Ab2CWAmpx3h1iHNyDvxIYnzXIszSw8Nnt5YtYVvsXo4xLhKnTo3bGSbMdNh4l4mmwzNhRCR/RMGyebeWctnmHxUg3Rb6lsAvz0XG/b7C1UdiwyhOZyqSSwMOsLIyDQvjsd4QpBfMuNXYNs6VpkpYNdMLbSvsnSnw/5gBz/ySDxlqrbuNXGkeG6c52OAg7cxm7FP8IF2bIAZsN/i8gNvR3kbzJqVM8Zz/lakbtJLwNZbsQuPmsyQPf5cCo0gwcwzhWYrZ9qW/TV2TJX5kz+vXIqQ7vBKOKPoFfrWm5DUnWMtF90IE1sJoUvZuFlnhHEbs7O+RNcB0a4E8dJOdg5whN7rPvx+oY9FhaE4ZD0NUcMYE4uLwCqZ1tiXVlB8LDPY5E4XJJThUbqa9mNpQ/CEmuhoQ9b24IKt6iXrWl0G7xB/OmTFzHRtPvItVH1F2qxSL+4c/X86vxOpa4xlyUT+gzJnJtd8gUrL/CfO6bnPA/DHxK9aqyNcbtxQlgGQBDnzMpQkIRg06vgMqWZUHEF2C9bnfM/hTszqN2ukjUEkW6R/b2jcX+XRd5s0Ay8zzx1BpdVyfUkQf5WUHfbjOxu3jKjrs4mgsMgAlbsN3PpS86sZbR0RuHGxr8lUB3Sv9W1Mqgk3G7NLCrncw9v8fTpjBuy0xAmBORKMt0GKsqOHOfh3dQmDcnVYUmPj13hRRk5Uvn6jq/9ccOBnixzlzxUtgcN6sT1/vpIQFAsCRZpAG8G5NtlvaTUTu3POkZbQ2XpvjqzxT5wOFw2zBWY9q6vrqk/NVGx9SkCxerjuZ0F7LNMt5IEk/Rej

fy3NuGnn36iWbNmtjEPPvggR44cse0/fvw4BoOBL7/8khYtWuDi4sLs2bPx8vLim2++yXWuJUuW4O7uTlJSOhW
f38aNG3FOdMzVsfvfvvnmG2rWrImrqyv+/v60adOGlJQU2/45c+YQHh6Oi4sLVatW5cMPP8x1/OnTp3nsscfw8
/PD3d2dBg0a5Fqvc9asWVSoUAEnJyeqVKnCZ599lut4g8HAnDlz6NKlC25ublSqVIkffvgh15gVK1ZQuXJlXF1
duffeezl+/Hiu/RcvXuSxxx6jVKlSuLm5UbNmzVzrn0JOsmvQoEEMHTqUEiVK0K5d05566ikefPDBXOOysrIIC
Ai4rCv6JSdOnKBjx474+vri7u509erVWbFiBfDPtNrly5dTq1YtXFxcaNy48WWFevNBQBg8
en0uZZ2RkMGLECEJDQ3F2dqZixYq2eP47dfd67r2oSMs0czA6iV/2RzNvwzHGLt3PwAVbeXjWRtq+t5ZGE38h/
I2fqPTajzQY/wvNp/zGve+sofXUtbR5dy2tpq6l+ZTfaDr5V2q/uZIK/1tB7TdX0vzt3+g4Yz2952xm6Bc7mLr
yAF9tPcXmoxc5l5B22b9REbGff026KyIiIiIihY+HhwceHh4sWbKEExob4+zsnOe4Rx55BFdXV3788Ue8vb35+
OOPad26NQcPHsTPz4/ly5fTpUsXXnvjzNRYsWEBmZqYtQQM5lXeHDh3ihx9+wMvLixEjRvDAAw+wf/9zVdamo
q77zzDp999hlGo5HevXvzyiuvsGjRIgCmTp1KREQEn/n2376KeHh4UydOp3XFixfTqlUr23VSU1lJ46aWXqFWrFsnJy
YwaNYouXbqwbqwc+f0X0tajhw5kqlTp1K3bl1cXFzYtWsX8+bPN4+bN4+GHH7aNufTe09Pzis/ysv
HHnuMt99+my5dupCUlMTvv/9uS3WrSIUaNGMXPmT0rWrcuOHTsYMGAA7u7u7uPPHEEyQnJ9O0iRQtKlSrFDz/8Q
FBQENu3b8disQCwePFihgwZwrRp02jTpg3Lli3jysefpHTp0tx777722ON58803efvttsJ9qxcnTpz
Az8+PU6d00bVrV55//nkGDhzIlq1befnll3PefR3p6OvXr12fEiBF4eXmfPlyHn/8cSpUqGUqJCrinH+/Pk8++yzt
infFy9epHnz5pw7d47g4GAgZ13V1NRUHn300OOyf5/PPP09mZibi1q3D3d2d/fv34+HhkWvMsGHDeP/99wkKCuJ
///sfHTt25ODBgzg6g6OnLkyBHat2/P+ffqwadMmpk+fTu3at127BgXL
lzIM57rvffC5sTFFPacSWDvmUT2nkngr6gkLiRn5Os1LFZITMSsiIS3rquOcHIyU8XMjPNiLWqW8qVHKmxqlvPB
OubkqWhG5eZq6m080dbfgaequiMiNK85Td6VgaOpu0fXtt98yYMAAOtLSqFevHi1atKBHjx7UqlQlULyKmQ6tChA
zExMbkSgRUrVmT480EMHDiQu+++m/Lly7Nw4cLLzn/oOCEqV67Mhg0buPvuu4GcJFBoaCjz58/nkUceISIigie
ffJLDhw9ToUIFAD788ePGjh1LVFQUkLOe5YsvvsiwYcOAnE7ZYWFh1ah3uRDQEJKiXZs2cPNWrU4Pjx4
4SFhTFt2jSGDBliG7dlyxbuvvvuTp06RXBwMDExMZQqVYpfmvvmFFFiilaXPHZVa5cmffee48OHTpctcm/79u3Ur1+
f48ePU7Zs2cv2V6xYkXHjxxvHYY4/ZtoDfP54VK1awceNGZs+ezSuvvMLx48/O77PimTZtSvXr1XGuad+/en
ZSUFJYvXw7kVPS9/vfvrrjBs3DshJhHp4ePDjjz/Svn17/ve///H999+zb98+2zlGjzhJW2+9dSWSdopwuw8++CBVq1b
lnXfeAXIq+hITE9m+fXt0OEP4+/vbkm7/VatLbp168dv23dpSvEXX3xhSxTEX3xhSxTGx
sZSunRpIiIi6N690/3798dkMvHbBjlu/fjOtWrQgJSWFkydPdPUqVKFVatWdWwSvYjw+x3dnr3I+vbOw9e7Lav9UQ
sGw9fZNuJOPadTSAxPdsusVwvgwHC/N2pUcqbhuV8aVapGEl1300dlkixp4o+ERERERERZE7QLdu3ejQoQ0///f
/zxBz/++CNvv/O2c+bMoW/fvuzatB0GDBiQ5/kjjIyNxcHCgUaNm3+/v5UqNqKWLqFHk+U/3cXpWM2xbsDrdX
FyMhI2zY3NzdbkgA/wJdwAEhISOHfuXK5z5zB0KBBg1xAw8dOsSoUaPYvHkzfy5csFW/nTx5MJzx6Mtcvg//dFRvgr
rvuonrl6syfP5+RIOeycOFCypYzNUyPTS/Pmza/43CIjIzl79iytWtW7Nt25aHH34YX19
fUlJSOHLkCP369rll3LkCP369cv13LKzs22NIHbu3EndP9mXw7bdqU999/P9e2SwlbAHd3d7y8vIyMtczB
WjSpEmu92azmYkTJ/LVV19x5swZ19cjIwM3NLde4vBoE9e/fn9mzZN8+HCio6P58ccfWbVq1W/+ccf+fXXX/o8H4DBgwf
z7LPPsnLlStq0aUOaUO3bt1yxf/f+Pz8/HJ9jnbt2sXOu+bttVaAAVqsVi8XPCsWPH2LNnDyaT6arJ25u594JktVrZf
y6RDYcv8PuhWuhC2w9Hkdal1tlu8dwMqxWkdUhKKhJ9IiIiIiiJ
3CBcXF+677z7z7uu+8+3njjDfr378/o0aPp27cvycnJBAcHs2bNmsu9u1T15Orqessx/LchhsFguOH1vTp27EjZs
mX55JNPCAkJwWKxULUUxKNGDTIzM30Nc3e/vHqof//+fPDBxo8PPHHf8+cHJ/8+SQJHEw//jV5z8dtq/PePbXL+AaQGM8+oDwr/MBDUo+p/ALllGrhgo8SSKcpHkWzJbaVezsleZvvtlIdHK8zc4HKdxsRAG9NE2g
rqjv//+ezpl6nTVMQaDgaZNm/Lmm2Y+yY8cNJycWLx4MYGBgYSEhD06NHLnmlYWBiQ80x37txJbGxsnucDw
3rYd3yYYNG677mV46x5YtW3Jt++0PPy48FfnqHxl7Lr07+/vT+f0fZZwE1bwtPrz+XUhiqQBgUk3b55teva1hIaG8
swzz/DdqN9vrKn/BAsLVDccmv8
+/r60rVrV4YNGbvbtm0PbbFVcCrKUdXPQzazZZt2Z0PvbE6efaTx5NDR/o83IRBO
ePcj0zmbFihWMGDGCSpUq0alTJwYMCMDHH3+Mp6cnI0eOpFSdUtdMlP3bkCFDmDx5MpUqVaJq1aq8++67xMfH5
JLPPbYY0ycAcD0UuLzVJW6TrcnqBzvOGEQkn2bPQRmYpeQOIydDSaxpd7OeCwXgzsX8oqlr8L3XAnrx0DhXbq4rZ2xFbs
swzz/Ddd9/x8ssssv88knnlwxvri4OA4ePEh4eDiQ81nd3//L3xBlIyR8Uzrv44tFoWhnE39N8ODdNy/dXcgRXLb
Tk5UbN0tbQ2J0Hc2Su5r/mDZOR7wrPMyH2o4ObA0RkeopA9rtiwiZONl/0/9LoScy

```
55hs2btyIr68v7777LtHR0dedU0zfvz8PPvggZrP5msnToUOHcv/99105cmXi4uL47bffbM/rkrFjx+Lv709gY
CCvvfYaJUqUoHPnzgCMGDGCxo0bM2jQIPr3729r6LFq1SpmzpxJuXLleOKJJ3jqqadszTh0nDhBTEwM3bt3vyy
eW733m3E+KYMv/zzJ/205xZn4tNt2naLCYoUtx2LZciyWMUv306xiCXo1KkPr8EBMxitX3IrI5VTRJyIiIiJSz
Hl4eNCoUSPee+89mjdvToOaNXjjjTcYMGAAM2fOBHKq0lasWEHz5s158sknqVy5MjI69ODEiRMEBgYCOQ0Jvv7
6a3744Qfq1KlDq1atclWKzZs3j/r16/Pggw/SpEkTrFYrK1asuGxa6dW8/PLLDP/PLLPP744zzxxBM0adIET09PunTpY
ttvNBr54osv2LZtGzVq10DFF19kypQpN/Q8+vXrR2ZmJk899dRVxy1dupS77rrLVtGYFy8vL9atW8cDDzxA5cq
Vef3115k6dSr3338/kJMAmzNnDvPmzaNmzZq0aNGCiIgWI0Wfk5MTK1euJCAggAceeICaNWsyefJk29TS
/777/POO+9QvXp1Pv74Y+bNm0fLLi2v+37LlCnDt99+y5IlS6hduzVYfffQREydOzDXm9ddfp169erRr146WLVs
SFBRkS6xdjzZt2hAcHEy7du0ICQm56liz2czzzzz9PeHg47du3p3Llynz44Ye5xkyePJkhQ4ZQv359oqKiWLpOK
U5OOeu51apVi7Vr13Lw4EHuuece6tay6hRo3Jdd9asWTz88MM899xzVK1alQEDBtiqV//rVu/9Rvxx9CLPf76
dJpNW887Kg0ry5cFssbL24HkGfraN5m//xsxfD6nKT+QGq0tuPlHX3YKnrrsiIjd0XXflWsqVK8fQoUMZ0nRon
vvVdVeKg88++4wXX3yRs2fP2pJHeXnooYdo1aoVlycjK1SpVi3rx5d03a9abPcz0dcYua9CwzX287zfw
Nxzh8Pu9koJydk8lI+xpBPNm0HHXL+No7HJFCTRV9IiIiUuj17dsXg8HA5MmTc21fsmTTNm1aP
kYnIkVFamoqR44cYfLkyTz99NNXTfIBIBNGvWjMcee6yAoiuaLBYLMTExjBs3Dh8fHx566CF7h1RopGWamfP7UZp
0XsObS/YqyXcLMs0Wfth1li4fbqT3nM38eTzvNS1FRGv0iYiIyN8Ksjr9ZqqyXVxceOutt3j66afx9S2+v83Pz
My8ZvJBRG7O22+/zYQJE2jevDmvrqu/aTp48SVhYGKVLlyYiIgIHB/2Imx/l47WHi08z2Dqf
YWX/4AusPX6BJeX+GtKlE4/L+9g5JpFBRRZ+IiIgUCW3atCEoKIhJkyZdddy3335L9erVcXZ2ply5ckydOtW2r
2XLlpw4cYIX3wRg8Fw1WrAd99915o1a+Lu7k5oaCjPPfccycnJtv0REePGGh4+PiwZMkSKlWqhIuLC+3atePUqV0
2MWPGjKF0nTp8/PHHHhIaG4ubmRvfu3UlISLCN6du3L507d2bChAmEhIRIRQpUoVAPbs2UOrVq1wdXXF39+fgQMH2
q6/cuVKXFxccjUogJwmBq1atbK9X79+PffccQ8urq6EhoYyePDgXGXnOmAc+HgHiBQCtrUzmYyt1FkwcedWZT0fE0pM2ldP1pl9k+2UadKsSKlWqhIuLC+3atePUqVO
2XLlpw4cYIX3wRg8Fw1WrAd99l5o1a+Lu7k5oaCjPPfccycnJtvOR
```

rTiVXHzZNNy/O+BcBxNSt5K8aOvWCIiIiIiIiJ/y87OZtLX64nYlYhZPzIXS/M2HGf36QQ+6FmPIG/94lOKF6W
vRURERERERICY2AQembqMubtSMGOydzhyG2O7EUeH6b+z4XDe6/WKFFVK9ImIiIjchL59+16zQ+2YMWOK1bRdE
ZHibMu+I9z/7mp2xDnaOxQpIBdTMnl87mZmrTli71BE8oOSfSIiIiIiInLHslgsLPp5E098touL2c7XPkCKFYs
V3vrpL/63eA9mS7FsYSB3GC04ICIiIiIinek1LRO3v58JQsPWsg2KMl3J/t88OmiE9KZObMubk5KlUjRpYo+E
RGRO5DVqt9Yy83RZOdEiovY+ARenPU98w8ayDZouq7A6r9i6DH7D84nZdg7FJGbpkSfiIjIHcTRMecHmdTUVDt
HIkXVpc/Opc+SiEhRdPpcNE9/sIyfo92xGvRjsfxj9+kEus7awNHzyfYOReSmqB5VRETkDmIymfDx8SEmJgYAN
zc3DAaDnaOSosBqtZKamkpMTAw+Pj6YTOpGKSJFO4Ejx3llwVr2ZJQA/S9Q8nAqNoOes//gi4GNKV/Sw97hiNw
QJfpERETuMEFBQQC2ZJ/IjfDx8bF9hkREipotO/cy6uvN/GXW1zG5upikDCX7pEhSok9EROQOYzAYCA4OJiAgg
KysLHuHIOWIo6OjKvlEpMjauHUno775k8OE2DsUKSKU7JOiSIk+ERGRO5TJZFLSRkRE7ggbt+5k1NdbOGwoZe9
QpIhRsk+KGqO6KiIiIiIiIsXWhj938ObXfyjJJzctJimDxz75g9NxamYmhZ8SfSIiIiIiIlIsbfhzB299/TsHK
G3vUKSIiO7MoO+8POlIO7InUrgpOSciIiIiIiLFzh/bdzPt61/ZawgDdZiXfHA4JpmBC7aSmW2xdygiV6REn4i
IiIiIiBQrew8c5sOvV7LDUBGLQT/2Sv7ZfCyWV77ehdVqtXcoInnK9694ZrOZN954g7CwMFxdXalQoQLjxo3L9
Y/AarUyatQogoODcXV1pU2bNhw6dCjXeWJjY+nVqxdeXl74+PjQr18/kpOT8ztcERERERERKUaOnzrLrC+Wsdk
cRrZB/Scl//2w6yxv/3zA3mGI5CnfE31vvfUfWs2bNYubNmzl/7u+n7zHsWz5k1xX/x9Z3ubhMmURGRvLWW2/x9ttvM2PGP9NuYt99+m+nTp/PRRx+xe
fNm3N3daRcuvZZxQa8vr6ry+t++faxatYVly5axbt06nNx36jg4+cmN/hioIiIiIiISDERczGWT/5vMeuSA8kwutg7HCn
GZqO5wrfbTts7DJHLGKz5XG/64IMPEhgYyLi7Na65L8/OHtaHX32SUhIwMvLKz9vpfXDAeP1l+PJm9ZFCzv5tQnIHQ5jQOHjvAsBv/dpF4BJfr3NFXwSfx86y6XVQOEwZlVsQRvtHvAywsKdyYjqQ/HY4zUwUWnIn7Jm6Fy8WhJO3r9zdVbHcJH8JqfNCLxkCdPmgAAX+QoaTmLYp4OPK+jcEKxUmCJJqAUEJKtd/5uiKafsWyJsCwcaXpWXq/XqaZhbz0aXsqfiBzqzAr7jE0jctSh6IWAONp2hEUHPmg/ZzZZMbNwpP8BWAk8aQlrgL9T0HTwcnkCagTa6VOmpUGSRftqhEckDqYjHsDw0ETmR2OvqiPlJaYk7aEQzvLz0xqrfedE9LbI/wYThkpzjgeSi8QwN2umfO+jiRy1ZqOw9rQAqa1P2eOCf2G9vkRuwxqp/CFc1zHmvYb6AZJSXG8VMxJ04ZU0fT1RdnLNRdpIY0GYQwOUCUvaTLgpGD2LaBg7oMCYMeWwxvC2ezFEVYeHJ5fqxmgpGWECpfbKEAWQiK+d/J1h3DwSKIjmFZeFWdhkPSNC+m9zCKxKo8kPzN3QGmOUPikuz2whDa1Rt/BTxgFh5XQ3V8DfH6t20h/R/tPkTuGdLzbQUm7Ilo9F7sUPpS+jIVKKH0l2TjgZfbTLtDqAPDyg1fsHPMtGaiwF4ksG6SQd0FozXfJJA5gJ/XMBSk3XtV+rMuh81NIIZfK2PnTz76NXSrCRtQaHWK2nI5IQgrvhVa9ptzw6iH/oO0LqYn+QeNyJHkIs+Y9hUAHMJ8DnFmFaoQx4pxLfVuaWx1eVJUO+pWQpNOlhbV4O/xRbvb3xqbs2t7WZ0nLIa4Pno+EMxNsVFDZEMLVuuwlRiUhhuIzXuY9sTQbsF3T1FHlsFhFPiprg+6CB1cGxCAUV2nG+V+AXPd5a5qPJq7W4SwL2dWmMbnkZB2zPVNDfjmTrZfd3aJM3VZwXy6KaWJaKFXtJ1nJ9IzevhMlo+Q6V5Gzg/bddJvx2Dg+kmImmr0J+lWa89LeiHhyp9j/8yLEk/WG6g5R2BnwxaxNO+tyFP72DbtGgSU2fCrnJqHo3lZ1c8LjXh1BvpZVzN63QpjkDsQgr2LrvLWlAnj7ROmwI4mnN6XZ/m7j/ihSyEOaIyzdEk+y87WOMtC9M3TutbFDcm3DAz6q4xaFgEDs/YPZbxJvYbyydDGuHThe5vrZJDtyGa+Mqrt8yxAKhHGCM1n/rfbxoZaHhSFwH8ntF7L/6YxSJOoMgYxgwbUFZe3mChE5N8ZQTxLyTqEPhEETF/7E8LXr7qhRFtfUwnCvZc6bSsjsEKo0cfZFixs5xn+PyeqWvTm8IW1Gz7ZAB8bdsrKP5ikEbr6yzt4Nj7CfbTmlmCWcIfgPEoVdgBqG4IzzYk5JJT7TQRZSihRkbIfZdSMX4mpTjRbfK/V+bdw/6b4f//Z

+61pye/x+6AI/7Y2ydxhyh1GiTORERERERAq1IydOsXrDFix+5YjKcrFbHF4NO5Fx9gAJm74iK+4sKfvXkLzrJ
zzqdfhnTKOupET+TtLOn8iKOOvitqWkHd6CZ7OHbGMuLJtK3NoI23vPug9gSU8i7pfZZMWeIfXInyRs+hrPuv8
6b8POJO36meQ9q8m6cIrYnz/EmpWOR802l8WZvOtnTK5euFVsBOQkEtNP7CbjzF8k/vk9jv5lMLpoSmlBmPxjJ
JnZSqpKwdEafSIiIiIiIlJomc1mvl+5hoTkVP7yrmXXWJyDK1Oyy2vEr51P/Ib/w8E7EN9WA/Cofq9tjFvlu/F
v9xwJf3xN3OrZOPiVomSX/+FSurptTHbieTD8U3fj4FWSwO5jiVO9h6RPB+Hg6Y9Xg4fwatTNNsY9vDnm1ATi1
y/EnBKHUOB5ArqPvWzqrjkljoRNXxHUe8o/cYdUweuuLsR88yZGN29KdHjxdjweycPxi6ks2HSc/veUt3cococ
wWK1Wq72DuBOSExPx9vYmISEBLy+v23697w6cu+3XkNy6Vgm2dwgiIiLyLwX9/ZeI3Bk279jDzIgvSPWrxLaOk
vYOR+SGebk48PuIVni7Oto7FLkDaOquiIiIiIiIFEqpaeks+2UdBpMDkRl+9g5H5KYkpmczb8Mxe4chdwgl+kR
ERERERKRQ+n3Ldg4dPOm6XwVSLaZrHyBSSM3bcJyk9Cx7hyF3ACX6REREREREpNBJTEpm1bpNeHh4sC/N297hi
NyShLQs5m88bu8w5A6gRJ+IiIiIiIgUOhu37eLOuRiSvcqRYlYfSSn65q4/RkpGtr3DkGJOiT4REREREREpVBK
Tklm9fgsenu7sTVVzHyke4lKzWLDphL3DkGJOiT4REREREREpVDZt382ps1FkeJclWdV8UozMXX+UzGyLvcOQY
kyJPhEREREREESkOkpJTWL1+M56e7hxIUzWfFC8XkjP5ce85e4chxZgSfSIiIiIiIlJo/LFjD6fORuHiFOJMlpO
9wxHJd59p+q7cRkrOiYiIiIiISKGQmZnFuj+24ebqwsEMVfNJ8bT1RBx/RSXaOwwpppToExEREREREkUJh36EjH
D99Fr8SgRxJc7V3OCK3jar65HZRok9ERERERETszmq1svHPnVgsFk5afMi26sdVKb6W7DhDcka2vcOQYkhfOUV
EREREMTuzkTFsPuvQwSW9Odgqpu9wxG5rVIyzfyw86y9w5BiSIk+EREREREREsbutu/aRkJiEwc2PuGxHe4cjc
tst261En+Q/JfpEREREREETErlLTOln/5058vLO4mq5qPrkzbD4Wy4XkDHuHIcWMEnOiIiIiIiIiJiV38dOUbU+Qs
ElPDjaLqLvcMRKRBmi5Uf95yzdxhSzCjRJyIiIiIiIIna1e/8hLBYLSbiSoGm7cgdZuluJPslfSvSJiIiIiIiIiI3
SQlp7BjXyR+Pt4cSXO1dzgiBWrr8ViiE9PtHYYUIOrOiYiIiIiIiIN1EHj7Gxbh4/P18OKFpu3KHsVjh531R9g5
DihEl+kREREREREMRudkcexGqFDIOTpu3KHWndwQv2DkGKESX6REREREREREExC4Sk5LZtf8g/r7enFE1n9yhNh+9S
LbZYu8wpJhQok9EREREREETs4sjJO8TGJ+Dr482ZDGd7hyNiFOkZ2ew6HW/vMKSYUKJPREREREREREE7OLoidNYLFY
cTA6cyXSydzgidrP+OEV7hyDFhBJ9IiIiIiIiIiIiUuAsFgu7Ig/i7u7KxSxHMiwme4ckYjfrD5+3dwhSTCjRJyIiI
iIiIgXuXMwFomIu4OvlyVlV88kdbuepeFIzs+OdhhQDSvSJiIiIiIiIIhIgTt+6gxJySl4erhzXok+ucNlma3sO5t
o7zCkGFCiTOREREREREREArcwaMnMRqNGI1GLmQ52jscEbvbczrB3iFIMaBEn4iIiIiIiIiIiBQos9nMXOeO4enhTrrFQ
JLZwd4hidjd3jNK9MmtU6JPREREREREREECtT52HjiExLxdHfjgqbtigCwR4k+yQe3JdF35swZevfujb+/P66urtS
sWZOtW7fa9lutVkNGkVwcDCurq60adOGQ4cO5TpHbGwsvXr1wsvLCx8fH/r160dycvLtCFdEREREREREREQKUFTMB
ZJTO/Bwd9OOXZG/HTmfrIYccsvyPdEXFxdHO6ZNcXRO5Mcff2T//v1MnToVX19f25i3336bDn0Dnn89FHH7F582b
c3d1p164d6enptjG9evVi3759rFq1imXLlrFu3ToGDhyY3+GKiIiIiIiIhIATsXcx6rxYrJZOKiEnOiAFissF8NO
eQW5ftCCG+99RahoaHMmzfPtiOsLMz2Z6vVrp03j99dfp1KkTAAsWLCAwMJA1S5bQoOcPIiMj+emnn/jzzz9
pOKABADNmzOCBBx7gnXfeISQkpL/DFhEREREREkQJy4vQ5TA4mAOKztT6fyCUHo5NpUM7P3mFIEZbvFXO//PADD
RoO4JFHHiEgIIC6devyHef2YAksPdDYAks8nOEYkUHidjU+0dghRx+Z7oO3rOKLNmzaJSpUr8//PPPPPvsswwePJj58
+cDEBUVBUBgYGCu4wIDA237oqKiCAgIyLXfwcEBPz8/25grZWRkkJiYmOuSkJCQ3/u+d2j/IjmEGVpd6Ilv6hnH9uLkN1IgTuOLnaCjJcP2eDTHxM+x0V8wqLqtOI6ZS3Q3lndWG5mg
ZJTO/Bwd9OOXZG/HTmfrIYccsvyPdEXFxdHO6ZNcXRO5Mcff2T//v1MnToVX19f25i3336b6dOn89FHH7F582b
c3d1p164d6enptjG9evVi3759rFq1imXLlrFu3ToGDhyY3+GKiIiIiIhIATsXcx6rxYrJZOKiEnOiAFissF8NO
eQW5ftCCG+99RahoaHMmzfPtiOsLMz2Z6vVyrRp03j99dfp1KkTAAsWLCAwMJA1S5bQoOcPIiMj+emnn/jzzz9
pOKABADNmzOCBBx7gnXfeISQkjJL/DFhEREREREkQJy4vQ5TA4mAOKztT6fyCUHo5NpUM7P3mFIEZbvFXO//PADD
RoO4JFHHiEgIIC6devyySef2PYfO3aMqKgo2rRpY9vm7e1N48aN2bhxIz179mTjxo34+PjYvgABtGnTBqPRyObN
m+nSpctV58zIyCAjI8N2OykpKb9PTUQkjXsXtKvhNbeX8nMtJ9CUzvKz/gKn7kCRUK6zUG0Rp9Bn5g6hbDXUFFbnRCGg
+ZE9AZZv0u+74U2gOtXqCSKl89rmdK2skkAd7uwxdunIzZIFQ+jVIYuaDtekNuNKfWf9BD9PtFbbP6ZEC4lwaKHuOhtLFVK+Q2DX5d35wOd8zG7g4rvfwRa2e
ItlU9IkUuORWnRJ/cmnz/imqxWKhXrx4TJO6kbt26DBw4w4kAEDBvDRRx7e3rZXaGjobb2ei
IiI3Li4+ETSOtNxdXFWNZ/If5xSRZ/conxP9AUHB1OtWrFuRVc28LDwzl58iQAQUFBAESfP59ST0M2dsiQAQUFBAERHR+caExUWFBRETEx
Mrv3Z2dnExsbaxlxp8uTJJCQk2C6nTp3K71MTEREREREREREyT/xiUlkZZtxdHAgevzieSiqbtyq/I90de0aVMOH
DiQa9vff/9NhQoVAExbpVasmRJcq2bkp1p87r2LDz8s7O64/SuS+xVpe3Kd4CxVTwvaNSzGtlQQX0iftlwAtQQK6
8vHK9REREREREpHCJS0jEABgMBl0XiifxHfGoWielZ9g5DirB8//XJiy++yEP+NXQvmAVUvbY6P+MNl4k6OJh4+B0tpj4EYZ2hYdJ0uJWTYQPTj7MCRv8+rtrrn4dfgqS/ZrwZpzvx7+ZDZl
RvI+WI+dOhQxo8fT6VKlgLC+ONN94gJCSHcRqWsHwpKBzPZdMhtBlYNZ1QRmKO4qQ/xaZPf6PDf4n4yEfNzHPwJM8SYZ/bEPzSKrmhYqTx3/ncYWCTkFK+0fZPbT2MhNtwLYaw4tGqI7mDmYHhfNAY6wfOH0wLLQ2WpSvydzOju7S
YAEhRok/kMmfiOvAKdrR3GFJE5Xuir2HDhixevevJhXX32VsWPHEhYWxrp16+jVq5dtzKuvvkpKSgoDBw4kPj6eVf/ykQFmTpwlhm2Hz74slYFKSpWkGXvKFLvCphaPStcE6/lVVppr0FPwlOq1Bcy0l2oPrLTHA5XfhGPTmjD
FkzfvrpJ1xcXGxjFi1axKBBg2jduVGo5Fu3zuoxffr0/A5XRERERERECtCZqPO40ODsBkGFRIw6R/4pPRWUWf3Dy
D1Wq12juI2yExMRFvb28SEhIKZBrvdwfO3fZrSG5dqwTbOwQRERRHH5l4L+/ktEih6LxcLwCe+RmpZOSFAAyy/4c
y7T2d5hiRQqH/WuT/saefcnELkW/fpEREREREREECkR6RibpG+GZk4OamiT+RKEtNUSc3Ty2OREREREREREEpECkpWe
QbTbj4pJTxZdhNdg5IilKrYvI2Hzd5hT4nAKCMOvzdM4h1TJc2zM+RMkFE9ERtRhzIkx+LYagFfDjdhvhfbyYyagFfDTrnPt2MFS
TtWkJOQDYBjiTL43POYrhUa2MbErv6ElL2rMTi64t1yNNEVPzNOmmsyWtcbfQu6Z1aUmEUllXrzFZvbi/LXelL2ribg4dH5ep8J4RSTRt2bNGu69917
i4uLw8fG5rbGJiIiIiIiIiEjxopXjp1/vbtk/Ju6ELY8Mv18G8Xk0x3zO11GYPGbZiNIP+Bkc2++S0+nbCBWsAiOssH9oFxOPlWt7NfLPaRg6LRvmPVc2dSB49RGXO
UqJXEfsr3PwafoWtQ+s03xf+VIPfgZZOXvRZZOIc1d7OH9VeY53aq7AfSzH4b3BiHQVLDeDL1nYzvxYPqz3q90gAbvGPwE7frdPWVZe6BHwZE5xdaFDJRmPtgiU3bzo5wL6tHDnnrqYWdpn1z+g8S4QbrGVELxMbUQTkD
Pnw0g9fBmUiLXEtB9HL4tnyT2pxmYUxMAsGSkEL9ucFoY/Zq0C0ZdnLBh9MQDuQcErZZSDBw/a0
4wiQ19VRURERERERpECkZ2SQlW3G0cEBixWyrPqRVK5P4p9Lk3Kz3DoO9a8+FUogx+7Z7H4OhM8p5VeY53Dq6M771
P4V6tBZjy7mDrVrERrhUa4uhXCke/Uvg274PRyYWMswcAyLp4Cpfml28K9WgSMTm626r+43+bhWfcBHLwC8

v1ebzbRN3fuXF544QXWrVvH2bNn8zkq+3F1dSUgIP+f8+1mtVrJzs4u8Ovqq6qIiIiIiIgUiLTOjJyKPpMDZk3
bleuUdnwnmecOkrJ/LSfeepDUg5swGIy4lKtDxpm/LhuffnofUQuHcer9xzg5tSvm5Iukn9pz2bjspAtcWPoOp
95/jBPvdOHOh32xZKbhXKoqAFmxZOk9sIGTO3sSu3p2TpWgbwjpp/eRfnofyXt/xWox5/v9pmfd+DmTk5P58ss
vefbZZ+nQoQMRERGXjVm6dCkNGzbExcWFEiVKOKVLF9u+zz77jAYNGuDp6UlQUBA9e/YkJiYm1/ErVqygcuXKu
Lq6cu+993L8+PHLrrF+/XruueceXF1dCQONZfDgwaSkpNj2lytXjvHjx9OnTx88PDwoW7YsP/zwA+W7YsP/zwA+fPn6dTp05
4eHhQq1Yttm7dajsmIiLisqnBV7uXvEyePJnAwEA8PT3p168f6enpl42ZM2cO4eHhuLi4ULVqVT788MNc+zdu3
EidOnVwcXGhQYMGLFmyBIPBYJv+u2bNGwAz/++CP169fH2dmZ9evXY7FYmDRpEmFhYbi6ulK7dm2++eabXOf
eu3cv999/Px4eHgQGBvL4449z4cKFq97TlSjRJyIiIiIgUiPSMTi9WKOWjAau9gpMgwJ8cC4HVX11zbTW4+m
FPiLhtvdHTBs96DBPacTEj/WRid3Uk78idJO3/655zpyUQtHI4lMx1rdgZYzFjSk/Fr+zxOJcqQGXOM1P1rcav
WEqPJgaStS/Fp1hujozOxP38IFgsu5Wpzdu5zRCOcZpvumx8sN/GP46uvvqJq1apUqVKF3r178+mnn2K1/nOi5
cuXO6VLFx544AF27NjB6tWrueuuu2z7s7KyGDduHLt27WLJkiUcP36cvn372vafOnWKrl27OrFjR3bu3En//v0
ZOXJkrhiOHDlC+/bt6datL79k/fr1DBoOKNe49957j6ZNm7Jjxw46dO0jA448/Tp8+fejduzfbt2+nQ
oUK9OnTJ1f8/3ate8nr2YwZM4aJEyeydetWgoODL0viLVq0iFGjRjFhwgQiIyOZOHEib7zxxBvPnzwcgMTGRjh0
7UrNmTbZv3864ceMYMWJEntcbOXIkkydPOiQWLFjARx99xL59+3jxxRfp3bs3a9euBSA+Pp5Wr
VpRt25dtm7dyk8//URdTdu3e/4j1djZpxiIiIiIiISIHIyMjEABgMBq7wM7zIZVzK1s75b5ma1zXeKbACToE
Vb08Nji44eviRcXofnnXaA5D4xzc4eJWgZOcRZCeex5KRSuqB9cSvjcC5VFWyLp7GsWQ5Sj74EgDnFryEydOfh
E1fY3R2w+QVQMq+3wh56gPSDm/h4vJ3Ce77frcr+Um/nHMnTuX3r17A9C+fXsSEhJYu3TtLVu2BGDChAn06NG
DN99803ZM7dq1bX9+6qmnbH8uX74806dPp2HDhiQnJ+Ph4cHNk6Rbw4WbOoUKECU6dOBaBKlSrs2bOHt956yt
NWrFOOHDgWgUqVKTJ8+nRYtWjBr1ixcXFwAe0CBB3j66acBGDVqFLNmzaJhw4Y88sgjIwYMYImTZoQHR1NUFD
QZfd6rXv5r2nTvGvXz/69esHwPjx4/nl19yVfWNHj2aqaqVOn0rVrTjI5LCyM/fv38/fv38/HHH/PEEO/w+eefk52dz
cKFC5kxYwbVqlXjzJkzDBgw4LLrjR07cDBgw4LLrjRO7lvvvuuw+AjIwMJk6cyC+//7ciRgpkzzZ1K3blO
mTpxo08enn35KaGgoBw8epHL1yle8t7woO0SciIiIiIFwmK12P6sPF/h4UomM9xn8Du+nDUb8E414Z1ixCvNg
GeqBfcOC+7p2ZgsVqwGIxajEStGrAYDGIz/2mbAajDatlkN0e8vjf1nn+Ff2/7e/vexubb9/T7bbOY5g5EuZ7Y
zC3g4/gR1YjyYd/E4qY4OPHsh8u9rG+BSXBjg7+u8lp1ORloCHas1pUnCKcDAxIMbqBtanfjvxnPk3EG83X1pX
6MVO/xL47/pK1rUbsf02NM8HnUIq9XAu7Gn6WQ2893ulXgYjTQuW5dTWRnOwZGMcg147cf36Z8Qh4uTW8614T/
/NVxhe85/MzIz8fXywt3NjRBcr/vvrm/fvpw6dYTotW7awePFiABAwcHHj4ZO03euLdG3c+fOPJNSl2zbtoY
8awa9cu4uLisFhy/q2SaWqERkZSaNGjXIdd6hAiHhpycmuXbvYsGEDLnSPLAuHeTLTtGeHg4ALVq1bL
tDwwMBKBmzZqXbYuJicn15O+fhnJOBGdLU9Pf4QjTHbCAgrWxAh0HjyJXPcE81TjTV52ts9EzXFk7ETjqUXvKL+d
Xo6bV6dLZMKvWfdXrtrKB4vk2xuUrxyJ1GUvVDKNJvbUL6/tWZOtUo9y1MVMwlHGtv8kx3wFMoiYS1Q/qO9IIQbb
PY+C6sqJ+sMgq5Ne9NgVKd4v7EfuLRoevxwVKWWzNn7FPw8fnJ1wG9ouGH+9atP/xBAKY/b1+n6IHQ/zWGtzk5
tEaAp61ZdI6Mdg3eIY/bNkzBHP2V9+DOb4DVEZA8vGiMD0hjM0QkAzIhbmr1hgUCLF2XRv0sbMcxd3J/mmk0sj6A
8nf7AL6Nyz5HexFAdsHnVWN0qNvKdfKEpUHRqo3CxT9JfX5MOmJBZFVYmqvdOwUrMnX0dBYOcaGwbT5iTEvSyzk
2OjwRrBtjKjLEcOb0yTYeeV/KhUbwgMxRU2Q2zOOnXXkXqgk4NR4IWWOx3UNBx2swLhjRncI8WD+iHjO0kSdjJY
uDyEEaG9mIXbHAq4v5Bsk46TI3/Fj1CIyMuCFAcK6dqQtvKxabfCxmd2aUvPSJOXhUh/Gf2Ydsy7j7+GVnavh8E8
BIvyKETfhqbCzkYk0PArJZL1vX0D1Q6Xx+7JKc5qKuVqRf/9RvYy2+l0tuOnj6qGcCNgMBqeihGA8FZDrWZ9pPkH
BsX9d27wR67y7ssOHjxUbzB1r8nlrHcFbcE8PKLfW9+u8FTHwJLWeKkVxE7nzy0+Ktm++Y4iq+rqx3+6fqdZu
/TJbc1Dbwy/LwEbbfDp8/QncCIf5pM1x7ff0W9BY31dUEjZ8nqTHN5eXtelvaKTYXMgGrNwM3xTb+3CprNh1Pbb
GO+DnrX4ce3OjqiG5m/Kvq8OueWjOV/s6O6ivdfOWSvs3ZrE6egkF/DcLg2aRUh0UTYn8Ai/LvV5O/GoVNusg+
YwzjAd0BuNGDfg6Kw8ylbGdvoPDa/8iHDnV2wgM9F2fU9e25P3fHHSKr5fGxIWczo5XhT0DPdmW4xGAaB/HbmL9Xs
PL8ewnZxT1DnBW0iGkwXkqRpgZoYfyDwJSeD1yIwdv88yLoHNTdl03nxV0BpOcVXaN+hVLqNCEzLNsnrL+vTw9iS
PBjlKG0DHUzA7GGKqtCq2OoQ1EKuarYM3a6P5AoAsDIGDMYa4p5aeDlPj41mOywaw4iS4F8IYU2BWEbVxZm7dJa4

JEiVYvHgxjRs3vqwZR2JiIo899hhPPfUUkZGRADRu3NjWfGPJkiUADBoOiMjISLp16OZ6erqtMcbw4cNZvHgxG
zduZPrO6Xz11VdUqVLFVgG4ePFihgwZwssvv8zevXt5+OGHWb16NcOHD+fgwYOMHjOawNbwYvfu3bRoOYLPP/+
cSZMmcfDgQVatWkWnTpOICQlh586dDBs2zNY9+MCBA/z888/MnDkTgBkzZrB37146duxoe2aXODg4YDQa+f333
3nllVf43//+xz333MOKFSt4//2cNR2ff/55oqKiiIuLo2LFihw5coSff/6ZJ598ErP5xjswq6JPRERERECoT
B8M86ZSYDmLBiVm2f3bSybObBwDPXHOfvYGGOwOVGp11k4OFXZjt6syvIkWzXW/+7y3IxEB1qJToU/lm58d8Jx
UufGSPeacHwbDT3PVKFuOICiE4ysCUFvFIsTCzli1tqNulpmThlZFLKOYn9VarazvLhhQt8lxDP/DJlAajn6sa
57CwW/vOeYHJMNLvTOvmvEg4ONHRzo6aLCOGOjuxNTyfLauW5EiV4rkQJGh86SAUn51t+FpcYb2Bq6qFDhyhZs
qRtLbl/SOtLw2KxMHDgQGrXro2HhwdDhgzhww8/5KuvvvqJ58+bUqVOHXbt2ERERQf/+/cnMzCQ+Pp4vvviCRx9
9lFOnTtGiRQvKlClDy5Yt2bx5M++++y533XUXU6ZM4amnnnuKLL77gO08/pVatWgQGBmI2m/nkkO+YPXs2FSpU4
NFHH8OVV2JiItnZ2XTt2pWyZXOe//33329bo+69994DoHXr1oSFhdGgQQOcnZ1tjTFOnTqFi4sL3t7e9OvXDy8
vL5o3b25bM/Cdd96hb9++PPfccwB89tln7Ny5k/fff59Zs2bRrVs3APz8/HjsccA+OGHH/Dw8GDOnDmMGTMGo
9GIk5MTQ4YMoUqVKlSpUoW1a9eyaNEimjdvTs2aNRk7diw9e/akTp061C9fnq5duzJjxgwWL15M69atbfdbvXp
1+vbty6xZsxg1ahQ+PPj64ubnZkpwhISHUr1+fIOeOOLlzZzIyMihbtizt27fHaLzx+jwl+kREREREREKRA5BTO/
ZPEcTJaSLPkO3RHuTFWCxM8vsKQdzHfFd3tmMbdpJEWZeQL99J86+7KCbeO2zoX25xuJjM6k7S/3+9OSOCQXza
mMiac/J2I+jqK7PRsSj9TGoDYVfGUdHOlrIcnfikOJBxO5MejcbSvVIqjYSXxSM3mEZOB/vsj+fjiBdp7erEnP
Y2v4+MZkOfzh4OpKRzPzGRSUM56aTVcXDiWmcm65GSisrMxGgyE3UAV3rUY/1NReDWtW7cmPj4+z3OJCQn07dv
X1pG2a9euVKhQgTp16rBt2zbKlCnDSy+9xJo1a1i6dCkvvPAC27ZtoOePHnh5ebFoOSKGDRtGpUqIiLiyM9P
R1HROe2b9/O9u3bcXNzIzU11dZsw9nZmQEDBvDaa6/O9u3bcNnTmTmjVr0q5bPnk2LFi2AnGnA+J0+
exNXVlWbNmtmOs1xGbsVBMhpxnHXXXexdu3aPK8RGRnJwBDc23r378/77//77//PkePHgVg/vz5zJgxw7bf3dO
y8vRo8eTZ8+feTZ8+fejSpQu+vr506t/PfaUQtVL1ZPNmtm0smLmTmjVr0q5d2bPnk2LFi2AnGnAJ0+
exNXV1WbNmtmOs1qtHDlyBMhpxnHXXXexdu3aPK8RGRnJwIEDc23r378/77//77//PkePHgVg/vz5zJgxw7bf3dOdL
y8vRo8eTZ8+fejSpQu+vr506dLFNuaRRx5h0aJFRdH4+Pjw2effYbJZKJr166c03eOzMxMHBwciImJyXXtBgO
aMGTIEIYMGWLb9t577zFmL9t57zF79mwgp7L7Lw999999/59dff+Wee+7J855uhBJ9IiIiIiUiAcTKZL2T5AiT57GmL6jmDLu
WsPvAJXLDyZcpInU+CIoyfzfMryml1s2icbUfIwyR9qxNI6/ddz2Pur/ogDwaepD6QGlyY7PJvPiP2u4WYG/Vpx
h9/lMDCYDTgFO+D4WwKGWnow0xv09ykTIzjJ8/HUOMO5dxNPHhRb3VcBYoRQbk8E7BTxTzDgkZTL2+DGmliqF8
e/PbpCjI68FBPJa1DmcDAYmjchOVV1dizKP76u3SsmVLPv30U3OU3bt2oWjoyNVqlyalZcuWfmzhri4OFsCv
utnHJv5ttuF8jUWkymVi1ahUbN25k5cqVzJgxg9g9dee43Nmzfj5uYGXL0xRn4/h/h3AxDIqTi+UgMqTi+UgMQ+GdNvhMnT
rBmzRoGDDRqEyWTi1VdfpWbNmri7uzN06NDLGpjk9Tz69DLGpjk9Tz69OnDyJEj2bRpExs3biQsLCxfknygRJ+IiIiIgUEGd
nZ7BYsVqtGAwGnLVOn134mGN52vWn3DNkbOGFrCTGn9+LBVjuUYYvvXzY65SI+UbLBa/AI9yDGhE1rri/9IDSu
d773+eP/33XXjPPq44XXnW8bO9PA/O5ePk4KvEm4J1mIijJmYAkBOomlWRUShV8ksErxcrZVDNuqVm4pmbilJ5
xS12HTV6eN33sv4WHh/Ptt9/a79TxOqXznlm99xzDOlJSbz33nu2u2L7Li2ZPHkycXF9bzl
atXj/379OxYsVbjs1gMNC0aVM3MZ+0aNm3KqFGJCSEo0eP5mqMn0+1atn/TnnwT+vsHUV0bzq+1lS+d
9b9iwgSeeeMK2bcOGDVSrVq6G8vIfOTdu3aBUCjRoOIDg7G39+fe+65969ewM6Y9jVEOHjx4Xdfx9/c
Js3jO2bNvHkkO9ed2zXokSfiIiIiIFAgXZyeMJiMWiwwTyYSTMX8SQXJj3nJbhJs1/yvvjEDH5JN0TD5JjMm
FCJ8wfnY3EGNKzpfn192nuhvovG/z5/zj5ZefJSM
aeaca/sTnDvYJyD/lln79z/nSN+fTwGZwNBDwfhc7cPCa5mElxT2XI8gfgN8ZR9sWxeIWCOmCiR7ERwsiMBSY6
USDbim2zA5+/1BD1SsnFNy8IlNRPH/1R/AZjySF5dTUJCAjt37sy1zd/fn+eee45EE
aN56aWXbGu/+fr6UqtWLRYtmRrLtG8eXO6d+9OVlZWroq+ESNG2Jpr90/fH3d3d/bv3
1atq2bUtAQACbN2/m/PnzhIeHAzmdfgcPHoy3tzft6dy5M5MmTSI4O/dy5M5MmTSI4O
JgdO3YQEhJCkyZNGDZsGN27d6du3bq0adGpUuX8t133/HLL79fxcd4zPPPMMU6dOZdiwYfTv359E9uE0cnxGoxRdnVm
VFYWPjw8vvfQS33RzzzcTDRs3bsTX15d3332X6OjSo60fe9u/fncffCBCz2ZwrKXmr0OgrxMxmM1/NnNmq6H741/sJ5f
AMCubdLdx5+dqgtGx9/4TyfvTOBXRvWkpKUQLUGjen3+nhCyprW/rmusX76E915+joqt2zHyg3m27V/OeIfK77
nYtRZHBydKF+9Jj2HjqRy7/XoAZGVm8OHrr/Dn6p/XKHKlHU4ldE4tLqMtvVx5n
bo442AykWO2YzKZcFair8DVMkfSxrD5tl8nwJzO8IuRDL8I6lDWegydyJ8uKWQasm7qfMbyqfkHlU4ldE4tLqMtvVx5n
cTQRODMAp2AmDg4GknUmcmXsGBy8HPGt6YrVaOTH9BAaTgTKDy2BynXh5wscn3KcShMrYXQ2krgjkYRNCZR7p
RwZORmcmXsGj5oeOHg6YE41E/1tNOWGl7tiDBYjxHhlEuN1eRLvv5yyHQlOdMb3gpUq7iGUMXjjExbAjdTOrVm
zhrp16+ba1q9fq9fP+bMmcOKFSsYNmsVKvv5Y7JMpoi2d8X2YpQMmcTrvYTPDMgkyJ5pW5pImEnZ0ms9smWZOnUq7999/
tWqVYu1a9fy2muvcc8992uvcc8992C1WvNstnEtX15erFu5jmnTppGGzYMd3d
3atasydChQ3OemZMTK1eu5OWXX+aBBx4x4gOzubatWq8cEHHwDQuXNn3n//fd555555x2GDB1CWFgY8+bNs93f9ShTp
gzffvstLL774IjNmMPUcu+5i4sSJPPXXUU7Yxr7/+OkePHq/VDu3a4ubkxcOBAONnfuTEJcWnVdo02bNgQHB109enV
CQkKuO7ZrMVitt1BPWoglJibi7ZelQNaICXl5e1z7gFnFzbFzeXBiKvldnSYS3F9RDvGMhtCqFuqGG1QYf4x96rTvsHtQ7eu9XZ38fm9e3WQ9hWI

AQd+vTHarXyvx4PYXJOoO+IObi6e7AOYjY71v/G+8vW4vL33PYriTl9itd6dSYwtAwe3j65En2/L/OOL/8SBIa
WJTM9nWXzZ7Ppp2XMXLkRbz9/Vnw2l5+/WMDLO2azY92vLJn7IZ9u2I3BYCD69EnG9evJ29/+iJtH/pQc56Vrl
bx/gyMiIiL2UdDff4lIOXP6XDRj3p1FSX8/3N1c2ZLoye7k2/czg1xupesbVLYescu1kwOOfO4dxlIPF447JFx
3Aw9zupkjo48QOieEmB9icC3jesWKvrwcHnOYz1qeBHYLJCMqgOMjD1FxQkVcSuUkDaOWK38N+YvAhwPxa+HH+
RXnST+eTuhzoQBEDo6k7NCyuJV34OzEGZyDnSnRrsQN3//1Wv3IagLcAm7b+aVwSE5OplSpUsybN4+uXbvm23n
zb7VIyXcHdmylYet21G/ZhoDSoTRp/yC1m7bg8J6dAJw7fpSDu7YxcPRkKtasQ6nyFRk4ZjKZ6emsX774quc2m
81MG/Y8j77wMoGlLy83vqdjV2rf3ZygOLKUqVSFviPHkJqcxIkD+wE4ffQwDVq1pUylKrTv1ZfE2IskxsUCMHv
MSB5/5bXbmuQTEREREZGix8X5n4o+AE+T2c4R3VkeZZXdknwAHtZsBsYfYunpPXxxNpH2KV64W65eoQdw7rNze
Nb2xKP6jTWpsFqtJO9PJuNcBu5VchoiWLNyapOMjv9kGQ1GAwZHA6kHc6Yzu4S6kHY8DXOKmbTjaVgzc6YApxx
MIf1E+nWt/3ezTAYT/i637/xifxaLhZiYGMaNG4ePjw8PPfRQvp5fib5CrErdBuzZtJ6zx3K+EB//ax9/bd9C3
eatAMj6ey6/k/M/6wgYjUYcnZyI3PbnVc/99Qfv4u1fgjYP97xmHFmZmaz6ciFunl6Uq5oz17xclWr8tWOLGel
p7Fy/Bt+SgXj5+rFu6Xc4OjvT6L77b+qeRURERESk+HJ1ccbBwYGsrGxAib6C5GRJ51XXJfYOw6Z6ZjxTYvay/
sQhXr9gJDzTB6P18hRF/B/xpJ1II/DhwOs+tznVzP6n97Ov/z5OvHuCkN4heNTISRI6Bzvj6O9I9NfRmFPMWLI
tnF9+nuzYbLIT/v5c1vTEu4k3R948wuk5pyk9oDQGZwNnF5wl5IkQYn+N5eDIgxwdf5TOM+n58OD+5u/qj8moT
tTF2cmTJwkMDOTzzz/nOO8/xcEhf1fVOxp9hViXgYNITUli8APNMZpMWMxmeg4dSfOOOSWdpcpXpERIKRa+O4l
n3nwLZ1c3ls2fzcWoc8Sdj77ieSO3bWb1t18wdcnKq15/62+reO/lZ8lIS8O3ZCCjP/OCL9+c3yyO6taDEwf3M
7RDSzx9/Xh52kckJ8TzxfQpjF3wDZ9Pe4sNK74nMLQsz098F/9ATbMVEREREbnTubm64OHuRkJSTnMGT4dsOOd
O5xjl9H/4WOPsHcZlHLDyaNJxHk2COw7uzPMpyy9uVmJNKWRezOTc5+cIGxaGOen665SMLkYqjK2AJd1Cyv4Uz
v3fORxLOuIR7oHBwUCZF8pwZu4ZIp+PBCN4VPPAo5ZHri7EgVOCCezyT3IxZkkMHtU8MJgMnP/hPBXHVyRpVxK
nZ5+m4pu33oX2kjKeZfLtXFI4lStXjtu5ip4SfYXYxh9/4Pel3zHOnQ8IrViFY3/tY97EObamHA6OjgyfPpcPX
3+JJxpVw2gyUavJPTkVf1f4OKQlJzN9+GCeHTfFlrS7khqNmvLO4lUkxcWy6utFTB36NJO/Wo63fwkcHBOZMGp
SrvEzXx3KA4/341jkXras/ompS35hyZwPmDv+DYbPmJNvzOVERERERIomg8FAYAk/os9fBMDDZCYnu3Kdi7XJT
Qk1n+ZRlzX2DuOaSmen8MaF/bwBrHIPYdJREwcTzRweffifQRZIPZjKxdUXqT6nOgbj5Z8dg9GAc2DOzDfXsq5
knMvgwvILeITnVPW5lnOl4riKmFPNWLOtOHg5cGTsEVzLueYZV8bZDOI3xVPhzQrE/x6PWxU3HLwc8L7LmzNzz
2BOM2NyzZ8qvAo+FfLlPHLnUqKvEFswZRxdBgyiWYfOAJStEs6Fs6f5bvYM7u3SHYAKNWoxdckvpCQlkp2Vhbe
fPyO7d6BCjVp5njPq1HFizpxiOrP/tG62WnI6XT1SPZQZP/5OUJlyALi4uRFcNozgsmFUrlOf59s1ZfU3/OfXp
1+47Lx7/tjAqcMHeXb8VBa8PY56zVvh4ubG3fc/xI+L8m9RSRERERERKdqCSpbgz137ADAZwM1oIdWiqYq3Ozv
uC3Hk5rrd2st9KWdp7GNl93O+LPUMYZ2bA+dNKZyZewanICdKdiiZZ5IvL1arFUvW5R2eTW45n7uMqAzSjqURO
PXyBhhWq5Uz888Q1CMIk4sJq8WK1ZxTWGPN/rvAJh+bR4d5h+XfyeSOpERfIZaRlo7BmLs82WjM+cLyX+6eZ3
tzh4/ypG9u+gxeFie5yxVviLv/fBrrm2fv/8W6SkpPPW/sfgHXbmls9ViISsz47LtmRnpzBn3P4ZMmYnJZMJiM
cPfX/DM2Vk57OVERERERABfn9xduXOcsknNVKLvdmlh2cJd7LZ3GDfF09lA05JmmnIKLLDdOZ+HHBzJcnfCpXR
OE4/TsO/j4OtAOCNBAJxfdh7Xcq44BThhybaQvCuZ+I3xhPT552fdhCOJmDxNOPk7kX46nXOLzuFVzwvPGpc3l
IxbG4eDpwNedXM+t26V3IhZEkPq4VSS9iThHOKMyT3/Pr+q6JNbpURfIdbg3vv49qPplAwulTN1N3IvSyM+plW
3HrYxG39aipevPyVCSnHyYCSfThhFw9btqdOspW3M9BGD8QsIovfL/8PJ2YUylavmuo67pzeAbXt6airffvQ+D
Vu1xadkIElxsfzO+Txio6NoOr7jZXF+/eEO6jVvRflqNQGoWq8hC6aM496uj/LjonlUrdcwvx+NiIiIiIgUUT5
enljJqZQyGAz4O2ZxNtP5msfJTbBamOTxJYZ8rDizp3oZF6mYmUKNpHTuii3NYg83jl08lmvmtyXDwtpxVNQ1
4XRyYhTsBOhAOPxbuRtG5OdkM25L85hTjDj4OOAz9O+lOxU8rLrZSdkc37pecq/Xt62za28GyXal+DEeydw8HK
g1IBS+XqPFbyV6JNbo0RfIdb/9fH83/S3mT32VRIvXsQ3IJD7Heun2cR557OTYmLiaaiMljSLh4AZ+SAbTs9AgPP
zsO13kunD2DwXADC5eajJw5dpg1g78mMS4WTx9fKtaszfhFiylTqUqusScP/sXGn5YydfEq27Ym7R5k35ZNvNG
rCyFhFRj6zgc39wBERERERKTY8fHyxNHBRFZ2Nk6OjpRwyCFRA1cY8w8hXa/q65/wh4Qh9EuDI4
z7M8ynLb5ZsEo2pBHYLJLDb1TvO+t/nj/99V1+zHsDB24EqU6tctj2gUwABnS6f5nurvJy8KOl2ecKxuBszZgx
Llixh586dBX7tli1bMQaDgcNm3bFWLF9O5c+frOueaNWu49/p1cNM9neYx6a5+LmYnCHHJ/vlBEG6+1s9WFHiYmJe
Ht7k5CQgJeX17UhUPuEXfHSheXzyLgq5V1M1XRESkMCno779779Ep+3JwnZZJr6OuXpiRm6cjzm
ejR7DcLPeGVlUC7DcowxfevmwlykRcxEsY6wbUJcF9y+4rrF9+/YlPj6e6eJUuW3N6gCsCNJvqOHz9OWFgYO3bso
E6dOrdO7etJ9EVFReHlIiIiIit8jPxwtfby+SUnISUF4mM45FMClT2E1yW3jHJPk
gJ7nRMfkkC8/uZuWpKB5PdCfA7GHvsG5IjRI17B2C5CEoKOi6k3y3U1bW9TXUUaJPREREREREREREREECozRaKR82dKkp
KYBYDCAv2PR6ghb2NUwH6Ct4Q97h2E3AeZOhl+MZPXJ/cyKyqBpmg9OVkd7h3VNdUrWueljW7ZsyeDBg1tx
+fn4EBQUxZsyYXXGPPi5+umnCQwMmxMXFhRo1arXs/jXj7OxMuXLlm2jDp1aaq7jy5Urx/jx4+nTp0+nTp
w8eHh6ULVuWzFUy9lWVuWH374gfPnz9OpUycXTtOnTp11fuZM2c2O4eH

```
huLi4ULVqVT788EPbvrCwnM7EdevWxWAw0LJly+s67kosFstVn5vBYMhVOblx40bq1KmDi4sLDRoOYMmSJRgMh
ssqErdt20aDBg1wc3Pj7rvv5sCBA7n2f//999SrVw8XFxfKly/Pm2++SXZ2dq7rzpo1i4ceegh3d3cmTJhwzXs
BJfpERERERESkgIWGBGGx/FPFV1KJvnw11WMhJlQlCdAsLZqPonaz9sQJXohzplyWNxTSBcxql6x9S8fPnz8fd
3d3Nm/ezNtvv83YsWNZtSpnPX2LxcL999/Phg0bWLhwIfv372fy5MmYTDkdg7dt20b37t3p0aMHe/bsYcyYMbz
xxhtERETkusZ7771H06ZN2bFjBx06dD0Dxxx+nT58+907dm+3bt10hQgX690nDvleJS01NZcKECSxYsIANGzYQH
x9Pjx49uJJFixYxatQoJkyYQGRkJBMnTuSNN95g/vz5AGzZsgWAX375hXPnzvHdd99d13E389z+KzExkY4d4O1K
zZk22b9/OuHHjGDFiRJ5jX3vtNaZOncrWrVtxcHDgqaeesu37/fff6d0nD0OGDH//v18/PHHREREXJbMGzNmD
F26dGHPnj25jr8aNeMQEREREREGRAhVUsgQGg4FssxkHk4lg5wz2pBStaZaFVXd+oYr1iL3DKHQ8rNkMjD/EwHj
Y5+RDhE8ov7tmkGJMt3doAJTyKEWg+62tVVmrVi1Gjx4NQKVKLlZg5cyarV6/mvvvu45dffmHLli1ERkZSuXJ1A
MqX/6eb8Lvvvkvr1q154403AKhcuTL79+9nypQp903b1zbugQce4OmnnwZg1KhRzJo1i4YNG/LII48AMGLECJo
OaUJOdDRBQUFAzpTTmTNn0qhRIyAnsRYeHs6WLVu46667LruP0aNVLp27QrkVPBdSoY98cQTlCyZ07DE3
9/fdo3rOe5mntt/ff755xgMBj755BNcXFyoVq0aZ86cYcCAAAZeNnTBhAi1atABg5MiRdOjQgfT0dFxcXHjzzTc
ZOXKkLa7y5cszbtw4hg8fbosfPEjSvSvSJiIiIhIgQoK8MfD3Y2UlFS8vTwJdsrEiBULBnuHV
qQ5WdJ51X1xoa1YKyyqZ8YzJSaebAx861mWb728OOCYiMWOa0U2CGxwy+eoVatWrvfBwcHExMQAsHPnTkqXLm1
L8v1XZGQknTp1yrWtWadOzJs2jezsbFTg4EDDhg1tY6pWrYQPjw+RkZG4dpy+YZkVfg4EDDhg1tY6pWrYqPjw+ZkZGXJ
fpSUlI4cuQI/fr1y5U8y87Oxtvb+4r+feeIPdz+68DBw5Qq1YtXFxcbNvyHvbTGMltrBwTnNRGNiYihTpgy
7du1iw4YNuSr4rDOcQcg9ZZqvKm4ubmBkCDBjf+uVCiTORERERERApUST9f/H28uRifgLeXJ45GKyWdMonOtP+C9
0XZ605f4muNs3cYRYYDVh5NOs6jSXDawZ15PmX5xc1KrKngm5g0CLr1RJ+jY+51CAOGg22KvKur6y2f/7/XMBg
MV9z276n5NyI5ORmATz75xFYBemMmlZGN+HgdXf2634mrPJTk5mTfffNNWffhv/04iuru73pegTERERERGRA
mUOGqlepQLLV/9u2xaiRN8tKWU5Sw+H3+wdRpFVOjuFNy7s5w1glXsIn3uVYKdzMtmG7Gsemx8aBze+reevVas
Wp0+f5uDBg3lW9Yhh7Nhw4Zc2zZs2EDlypWvmSi7luzsbBDO+jDx8fGEh4dfNjYwMJCQkJCEkIDQkBCOHj1Kr
1698jyfk5MTwFWXs07Um7xCV6BPJy969Thw4AAVK1bM7xCV6BPJy969e+0dwh2lRg2
1kRcRERG501QsVwbIqXAxGo2Ucs5gR7KnnaMqut51+wwnMu0dRrFwX8pZ7ks5S5zRmQU+5Vjh7shZh8Tbdr1wv
3CC3IOuPfAWtGjRgubNm90tWzfeffddKlasyF9//YXBKB9+/a8/PLLNGzYkHHjxvHoo4+yadMmZs6kceV1da6/
F0dGRF154genTp+Pg4MCgQYNo3LjxFae8vmmmwwePBhvb2/at29PbvBPhz4f/at29PRkYGW7duJS4ujpdeeomAgABBcXV356aefK
F26NC4uLnh7e1/zuPzQs2dPXnvnvtNQYOHMjIkSM5efIk77zzDvBP1d7dt71GDVqFA8++++CBlypTh4Ycfxmg0smg0smvXLvb
u3cv48eNvKcbb3nV38uTWnp70o888/j7+/Px4eHnTr508og5o5ZVc508DR1gd8nTg5uZGQEAAw4YNy
9VmWEREREREERIqucqEheHm4iUM00ywCkTRzuukVaUtbD8yV3ssncYxY6vJYMhsQf4+dReIs6cm+qNy4Wp3y
/zr2h9+b70fPy7bff0rBhQx577DGqvavG8OHDbVVY9evXZ8mSJXz99dfUqFGDUaNGMXbsWPr27XtLMbu5uTFi
D179qRp06Z4eHjw5ZdfXnF8//j//oX8//79mTNnDvPmzaNmzZq0aNGCiIgIypUrB4CDgwPTp09n9uzZhISE0KlTp
8vF7Ptq8OBjpY3gJpDXg24I4N5WWtq1a8yZctYuXIlDRs2pHHjpXHjxx4ds8HtbcxHlixf2XdFTDB6IIqJNTEYkNfs8kPYqwl2wgNSKZfH/pCaV+wY
3Tv3h0vLy/uvfdepk2bBsCzzzt7L8uXLiYiIwNvGDBmE0Gm1lomazmTp16hAUFMSKVM4d+4cfrrOYcCAAUy
c0PG6rp2YmIi3tzcJCQl4eXndrlu0+e7AudtrluO+e7AudtDcmta5Xg23ZuVfQVLFXOiYgUDwX9/ZeIFGlWq5Wx0z7m5Nkow
kJLAfBrnA9H09zsHFkRY7Ww0X04IZaz9o7kjpCOka+8w1js4cYRx0SshltPqXzT8Ruq+FXJh+gKn4iICIYOHUp
8fLy9Q7ltFilaxJNPPklCQkK+rYV4K25bRV9ycjKHDx/2vdfX17evXik08+wdfX17Y9ISGBuXPn8u6779Kqvsq16/PvHnz2
LhxI3/88QcAK1euZZP/+/SxcuJA6depw3WsBCAehVfvexmq6bVt5l/SrHCF
5ed60REm+AuSChT4JR1h8Zg+Lz8TyULIXXpabT+6U8ihVbJN8xdWCBQtYv349x4dY8mSJYwYLu3bsXiiQf3
MZE3/PPPP0+HDh1o06ZNru3btm1j0jKysr1/aqVatSpkwZNm3aBMCTmTZuoWbVmrS0z5JQ1JiYmsm/fvtsVsoiIiI
IiBSg8mVKYzIZyczKAqC0S7qm794Ab0s8zzn/a08w7lgVspKYcH4vxtIsLUNb3p7g5LaJi
oqid+/ehIeH8+KLL/sPe4dlc1uacXzxRjHRCHxheFRRDpSmUe/ZXxwTz4GR7Rk7bMwz3j7NSvbYAWoUjFPs
pX14yMjLIyMiwvU9MvPjMkiZqdzK4tYeXa4utUDDWIQz4MCrXR1M06LKcMBZZZL8P/LUNvQrPlK5GgoJHvjkadEZs
kkeSj5JjMmFCJ8wfnY3EGNKvuaxBbU+n73O7ds3X9b5K0yGDx/O8OHD7R3GRZeV7BBd
VpEmT8Pb2tr1CQ0ML7Noqel7NoiIIiIiJy49zdXKlbvSpxCf8Uamj67vWpZj5Ie+Mf9g5D/iPAnM7wi5GsPrmfWVEZN
E3zwcnqmPdY1wAaBDYo4AiluMv3RN+2bduIiYmhXXr16OtlAMDA8nMzLxsIcbo6Gi2CgnLaSQc
FBV3WhffS+0tj/uvVV18lISHB9jp2YjSRlB9jp16lR+35qIiIiIjksxpVK2IGsnM1PTdG2GIxZNhrPoraz
doTJxgU50zZLG/4V++0Bys8iMlosl+AUizle6KvdevW7Nmzh7O7d9peDRs2pEefPnS+VvbZrEcaysxBw4c40T
JkzRp0gSAJk2asGfPHmJiYmxjVq5ciZeXF9Wqqztd7ZB2DKoxXxfo8/T/T05/9Wqz7Hq6gVcbHZEB7eI2uuG1e3/eQ5r8uRWBDFHJv/UxpiEYW5I5MaYbfJTpIcvf
+wbVCH3sHU1Va2H7B2GXCcPazZPxx9i2ek9LDqdwL3xLpRO9qNTxU72Dk2Koxxfo8/T05MaNWrk2ubl7o6/Lc4PDK8v79
te79+/XjppZfw8/8/PDy8uLF154gSZNmtC4cWdxd5++23iYqKYuTSoXr2GJmtc
```

```
RERO3FzdaFejXB+/GO9IYElAQh3T+GvVHc7R1Y40Voz+J/7YlDRY5FUKzuB6XEJWDwbY/Qub+9wpBi6bV13r+a
9997jwQcfpFu3bjRv3pygoCC+++47236TycSyZcswmUw0adKE3r1706dPH8aOHWuPcEVEREREROQ2qlG1Iiajk
YzMTAD8HbMJcMyOc1SF02uOX+JnibV3GHKLjPWfsncIUkzdlq67/7VmzZpc711cXPjggw/44IMPrnhM2bJlWbF
ixW2OTERERERERERROytWsXylA4JIirmAmVLhwA5VXOx8U52jqxwCTGfo6fLr/YOQ26R1dUXQ/XO9g5DiqkCSfSJF
DUHHf3tHcIdpca1h4iIiIhIMebs7ESzhnVZ801SLBYLRqOR8q5pbE70It2iZgWXTHX/DCdU6VjUGer2BgctSya
3h12m7oqIiIiIiIj8W/1a1fDz9iI2PgHIacpRSU05b06xbKUxO+0dhtwiKwao/6S9w5BiTIk+ERERERERsbugk
v7UrRFOzIV/1p+r4ZGMEasdoyokrBYmuX+Jwd5xyK2r2gH8K9g7CinGlOgTERERERGRQqFxvZo40JhITUsHwN1
kobJbqp2jsr9njd9T2nrG3mHILbJixNDqdXuHIcWcEn0iIiIiIiJSKFSrVJ6w0NKciz5v21bbIxnDHVzV52lJZ
JDLj/YOQ/KBpVpnCAi3dxhSzCnRJyIiIiIiIoWCg4MDrZo2JCMrk8ysLAA8Hcx39Fp9k1wW4m5NtncYcossBhO
m1qrmk9tPiT4REREREREpNO6qU4NypUM4cy7atq22Z9IdWdUXbj7I/cZN9g5D8oGlxiNam08KhBJ9IiIiIiIiU
mi4urjQpllSUtMzyMrKBsDbwUyFO7Cqb6rHIkyY7R2G3CKLwQEHVfNJAVGiTOREREREREAqVRnVrUrZUMGeiYmz
b6nsmYbqDqvq6WH+jmvWQvcOQfGCt2wd8Qu0dhtwhlOgTERERERGRQsXdzZXWzRqRnJpKdnZOVZ+ng5lq7il2j
qxgOFozeMP9O3uHIfnAYnTCdO9Ie4chdxAl+kREREREREKTQaVy3JmVDgjl97p+qvrqeSbgYi/9U1v85foWf5aK
9w5B8YG3YHzwD7R2G3EEc7B2AiIiIiIiIiIyH95erjT/t6mfPJ/35KRmYmzkxNORisNPJNYn+Bj7/Bum2DzOXq6/
srtnKU86fcMvvsri78uWHB1MHB3qIm32jhTpYQJgOPxFsLez7vT71cPu/JIdcc89/Vdksb8XVm5trWrYOKn3u6
2845bm8Gvx7OJSrYS4mmgd01HXmvujJPJYBvTZ3Ea286ZqR9sYkEXV8r5/FOj9ODnqTxZx5Fu1fKOoTAxO7pja
v6KvcOQO4wSfSIiIiIiIlIo3V2/Nr9v2c6h4yepUr4cAFXcUvkr1YOLWU72De42ecd9Ic7WjNt6jbUnsnm+oRM
NQOxkW+B/v2bQdmEq+5/zwN3JQKiXgXMve+Q6Zva2LKZszOD+SldPI7SvaGJeJ1fbe+e/E3gAf12wYAE+ftCVi
n5G9saYGbAOnZQseKetCwAvrOynlJeBuQ+58/pvGbyyMp1vursB8OXeLIwGikSSD8DabhK4+9s7DLnDaOquiIi
IiIiIFErOzk5OaNOcIwaSknPW5zMY4G7vBAzFsDFHU8t2mrDztl/np97u9K3jRPUAE7WDTEROcuFkgpVt53KmR
ZuMBoI8jLlei//Kons1RzycDFc9t7Mp97G+rv+Mb1/RgXmdXGlbwYHyvkYequLIKO2c+C7ynyrAyPMWnqjtRCV
/E31rOxJ5wQJAfLqV13/L4IMHXG7DE8l/yYF34dDgCXuHIXcgJfpERERESkOKpbvQp31anBiTPnsFpzknsBT
llUL4aNOSZ7fIHRDgnMhL8LCP1c807ibTtrZmeUhX71rl1Jt+Z4NgFTkqgyM5lnl6VxMdVyjWtbc123dpCRX45
mY7FaWXnETK3AnLTFsJXpPN/QkVDvwp/GyDQ44/roXHuHIXeowv8vREREREREO5YRqORjve1wNfbi+gL/zSoa
OCViJcp246R5a9nDN8Tajld4Ne1WKOM/SmdpqEmagSY8hwzdOcm4SWM3B16rWm7Dizo4srqPm681caZtSfM3L8
oFbMl7+Tl4VgLM7Zk8nT9f6Zhv3OfC39dsFBuWjKHYi28c58L605kszPaTJ/aTnT/OpXy7yfxzLIOMs2Fs6oz5
e7hmPzK2DsMuUMpOSciIiIiIiKFWplSwbRt3oQLF+PJzMyZ5ulggOY+8cViCq+nJYEXXJbb5drPLO9nb4yZLx5
2zXN/WpaVz/dkOa/utav5etRw5KEqjtQMNNG5qiPLerrx51kLa45f3in5TKKF9gtTeKSaIwP+legr5WVkWU83T
r7oybKebpRwM/Dc8nQ+6uDK+HUZeDoZODDIgOOxFj7emnXZeeOtzqcGvvepAYfYjxJ9IiIiIiIiUuila3E31Sq
X58iJU7YpvEHOmVQrBlN4J7h8jrs17y63t9OgFWks05TNbO+4U9or7/TAN/uzSM2CPrVvvAFGeV8jJdwMHI7NP
X33bJKFe+encneoA7M7Xn3NvYm/Z9C2ggP1Q0ysOZ5Nt2oOOJoMdK3qyJoThauiM9PghFvP+fYOQ+5wSvSJiIi
IiIhIoefm6kL3B9vi7urK+dg42/aGnklFegpvVfNhOhg3Fug1rVYrg1aksfivbH7t40aY75VTA3N3ZPFQFQdKu
t94+uBOooWLqVaCPf9Zg+9MooWWEanUDzEyr5MLRsOVm3tEnjfz+d5sxt3rDIDZCll/FwdmWayYr778X4FLafQ
KzgEV7R2G3OGU6BMREREREZEioWrFMNq1vJuY87H4H/TOE1WmnlG4epiE7hneqxEBOXT229nZ5fkc7C3Vl83tUVT
2cDUckWopItpGXlfoaHYy2sO2Gmfz2nPM9TdWYyi//umJucaWXYynT+OJ3N8XgLq49m0+mLVCr6GWlXIWdtvzO
JFlrOT6WMt4F37nPhfKrVdu3/slqtDFyWznvtnHH/u9NvO1ATn2zPIvK8mQW7smgamveagvYQ51Od3/Yj7B2GC
FdfSVNEREREESkEGnfsin7Dh7h4NETVK0YYBkAJpywaeyewIcHHvsHdoM7W36huPVjg153199p2Leen5to+r5M
Lfev8k9T7dEcmpbOMtK2QdOLtwEULCRk5yUGTAXbHmJm/K4v4dCshngbaVnBg3L3OODvkJOpWHc3mcKyFw7FQ+
r3cU5Wto71yvZ+9LYtAdwMPVv5nyvCYls7O/DaNRnNSaF/RgefvyjsBWdAyDC6asiuFhsF6aXGDYiYxMRFvb28
SEhLw8vK69gG36LsD5277NSS3rlWCb9u59fdZsG7n36WIiBScgv7+SOTuXPsPHWXanIU4zkRWNLftv23OB+Op
LnZMbLr52DNZLPby/hbL157sBRaZquB2LYzKdmOt71DEQEOdVdERERERESKmGqVytOxQxTsuxseTkppm297MOwE
fh8LXiTUvrzp+pSRfMXCsfG8l+aRQUaJPREREREEipz2LZtyd/O6HDt5hmxzzhp3jkYrrX3jcDAUsi4N/xFoj
qa342p7hyG36KBzHUK7v23vMERyUaJPREREREEihxHRwce63Q/FcuFcuTYSS6tSuXrmM29PvEYCnFzjqnun+F
szbB3GHILThtC8HtiPs6uRWOquNw5lOgTERERERGRIsnf15ueXR7A3d2dE4bdvLuqbTyCvRjpFd2d2WHdzND
nuHIbcgzuqJucscSoSUs3coIpdRok9EREREREESKrGqVytOlfSsSk5NJSPqnk2sNjxRquCdf5Uj7mOzxfxgLcbW
hXF261ZGoZpMoW6upvUMRyZMSfSIiIiIiIlKktWnWiNZNG3PyzxzZG3PyzDlSO9Jt2xt5JVLOJeOqRxasgYYfKGM5be8w5
CaZrQYOVH6Wqm3UfEMKLyX6REREREREpEgzmUzO6NSOJvVqceT4KTKzcjrvGgzQOjeOAMdMOOcIHpYkBrsss3c
Ycgv2+Xeg2iNvYDAY7B2KyBUpOSciIiIiIiJFnquLC0888hPGh85hbk0NETmMO5nXcdDNDO/yL+dk72jXdZhIe18
EOlluuzO6Up5ft+gKOTk71DEbkqJfpEREREREESkWPD19uKpRztTtTnQIh46dsHXidTZaud+Oyb7K5iM8aNxol2v
```

```
LrdtirE+pJz7Gw8vH3qGIXJMSfSIiIiIiIlJslAoK4KlHO+Pn48WRE6dsyT6XS8k+h6wCj+ldj4U4kF3g15Vbt
zG7BiW7v0fJ4FB7hyJyXZToExERERERkWKlcvmyPPVoFzzc3Thy4nTuZF+JC/gVYLLvIetaalgPFNj1JP9szK6
BT5e3Cata296hiFw3JfpECoDZbOb/3n+bZ1s34rHa5XnuviZ8/eF7tm84ALpVDcnztWTuh1c994+L5vFMq7voU
SuMkd07cGj3jlz7o04e561BT/Fkkxr0rl+Zd4Y+TfyF87b9WZkZvD/8BXrXr8ygds3YtXFdru0XzP2QOeNey4e
nICIiIiJScOpUrOK/Hl1wd3Ph6Mncyb4HCmgar4M1k9Hu397260j+slrht6y6+HR5m2r1m9o7HJEbokSfSAFY8
skH/Px/8+n/xgTeX76Wx19+jSVzPmTFZ3NtY+b8vjPX6/kJ72IwGGjctsMVz7thxfdETH6T7s+/xJTvfqZslWq
M69+ThIsXAEhPTWVsv8cwGAyMifj6/9u77/iaz/6P46+TSCJ7IIsIsSokZu1NzduKW1ujRqlZm9LepahVtdpq6
24pipa2qNLWHq1RWwghdowQK4ggkeT3h7vn190gCeEr0e/n43Eezbmu63zP+5wjD3Ieji3e3YkJeX
+5sRrFs3nxMH9jFu4nJdebse0wb3NPwRdPBvN2u++oe2AYU/x3RERERERReTrKlHiBLq+2xClnTk5En/v/Yp9tC
v/KdYV8Dnee6vMPtfueXCmXn+pzSOZKTjWxKulFfFq0UZFPsqRML/SNHz+eF198EVdXV7y9v9vWnRogVHjlh0U75
z5w69e/cmV65cuLi40Kp5VKy5evGGxJjo6miZNmuDk5IS3tzeDBg3iyZMnVnrs97S9ztdDhgzh3j3YkJeX
6Wq1uTYgX3mMZ55vC1u09avomTFqvgGBD70usvnfEG91m2p0+pVAgoXpfuoD3DI6ci6xd8CcHjPDi6d080b46c
RWKw4gcWK02fCRxyPC0fAH5sBOHviGOXr1Cd/kWI0bNeJG1evc0Pa/VQC+GDmM1wb/BycX16f35oiIiIiIPEVlS
xanS5uWOOZ04MRflvHa2aRS3+sqRZ1uPZXn9Um+SAe7tU/l2vF9CfNpLg8tWMjiPyWDK90LdpOyZ
69+7NH3/8wZo1ahKSqJ+/frcuvX/f3/3337ZDZHVfkmShWpjHtm3mJ8s0LDvNSbMZTmZqQmJjIlq1bmTt3L
nPmzGHEiBHdn+ezxk2WnIygWKoSxBgdCrXVzzf0ghRrbqIc4sHp1CUwv+8SKDq1d4Y7JdFxdpjFgW6R9UDbW+S
nkXZpkfZiYXTjJAD4GRSnMGjsBPBQqqsS6hOpKIO+vcmMSqXuWcddnHt2BqGGcpFDxPFDAVEVp+1CjZf21sa
XL8GIHx3mYaJ0hdRjwWmt5sXsTwcC3xKiZPDHPSu5VDrtXpu5IcJoVUZwQeSBlYQo11rWNRUMiuF9xWT+ahnl
uHh3jnMj/W2HLzbO0BAnhY77hmQ0WKjPnKBtxvfrGmmTCTiD0w0BifutuXm40V2b3zcfKbnXZ+emKpFMaT1+
+lvPAuvXtvGFJpVPuVm4e1nJ6KHvWaJXfV7iDkGrWg9XRhWb/I23HPFRimSgnOBD2vdIfNnBmnhyYptnRMfJ6v
```

Im5SIF8/ri6OFuMOXzLiZO3XLmbaott6j3+cB5EnpRLBiWWB4myLcYvl/zJ4e5LjSavUrJCDXPRViQ7y/QZfSI
iIiIiIiJZlY2NDVXKl6Zwwfz88PNatu3ah4ODA/nz+poLRS84J5Dj3B4OJbjwht9RFfmeI3E2XqxKKMmp284UC
S1PlQZh+AYEGR1L5JlRoU9ERERERETkb7xzedG9XStCihVmycp1HIo6TuD/Zvcl3rnNjWPbqeTuQN7cxblwKz+
+96KNjmzV7pGD3aZQ1l/wILd/ARq3bEFwuarY5lDZQ6yL/sSLiIiIiIiIPICtrS3VK5alSFAgi39Zy459B7hw6
QpJl07gmJJAyVKVicvhynr31uRNPE7pW5twT7lmdGyrkoKJOzkKs+pKXhLsPClfpxYV6zbFzTOXOdFEDKFCn4i
IiIiIiMgj+ObJRc/XWvNiqRL8tGo9RyPXky9PLpyc/3/vvnP2hThvV5DCd8MpcXsHTinxBibO/pKx5aRDcbbE5
+f8lXsEFg2hYf2WBBYticlkMjqeiGFU6BMRERERERH5BzY2NlQoXZLgIkGsC3Tn7KHtREdF4OXjj6tHLkwmE6k
mG47mLMMxh1ACE4/wwu3deCXHGh09W7lHDo7lDGXPvaKcOncV91ye1AtrQqkqdbF3yPnPFxDJ5lToExEREREREE
UknF2cnmr/akWuXG7J70OoidV7GtUsXyOOfHyeX+yeOp5psOeUQzCmHYLyToil+ezf+SSfQPLPHl2hyIMqhFLs
Tgzh/7hoOjncJrViTyvVbkssnr9HxRJ4bKvSJiIiIiIiIZJBnbh/qhnUguFwVdqxfwfFDe7lOPhrPPL7mGX4As
Xb5ibXLj2vyVV64vYeCdw+Sg3sGp886bpucOJKzLLtu5yX2zDWcXO9RqnIdQivVJm/BolqmK/I3KvSJiIiIiIi
IPAaTyYR/gSIO69SPs8cjidj5OOcP7OJq7HlcPXLjmdsHG1tbAG7aerHTpR7hTlUpcieconf24Zh6y+BX8HxKw
cRFu/wctytORLw7V89cxcUtlRdrNyakYi188hVQgU/kIVToExERERERERXkCNjY25C9SgvxFSnAp5gyRe7ZxcOd
vRB87hKOzC17e/tjZOwCQaOPIQadKHHJ8Ed+kaAISj5Iv8Rg5U28b/CqMd93WixMOJTiRoxjnr8Zz8/wV3Lxsq
NKgFSEVamiJrkg6qNAnIiIiIiIikkny+AWQpOkAZau9xJHwHez/YzOXoo9jY2OLl09eHJ1dgPv7+MXYFyTGviA
7U+uR5965/xX9juJsRSf2Xrf14ox9UaLti3IhOYm4Sxe5k3Aazzy+1GzalhIvVsMjl7fRMUWyDBX6RERERERER
DKZi7sn5WoOIKRiTY4f3MOB7RuJPhZJ7Lm7OLt64OaV23xKbKrJhli7AGLtAtjtVJtc9y4QkHiUgMSjuKbEGft
CMlkyNlzL4UOMXQGi7YtyJdWd61diib9+ETsHB7zzFqBEuaoUK1ORF3dPo+OKZDkq9ImIiIiIiIg8JfYOOSlet
gpFS1Xk7PFITh7ez7GI3Vw6d5qkpEScXN1x98yNg6PT/QeYTFyx8+OKnR/7nGvgce8SPknR5Lp3Aa97F3FNuZa
lTu9NNDlwOYc/l3L4c8kuL1dy+HI3GW5eu8zNc7GYTLF45PahcqWWFAoujX+Botj+b1/D9JozZw79+/cnLi4u3
Y/p1KkTcXFx/Pjjjxl7Qc9QrVq1KF26NNOmTTMOx+O8v2IcFfpERERERERERnjJbW1sCi5YksGhJqjZsxflTR4k
+FsnRA7u4cvEciXfv4OjsgptnHnI6OZsPm4jLkYe4HHnM17FLuYPXvYvkSr5f+Mt17wLOKTeNellpxNu4cTmHP
7F2ebmUIy/XbXODyUTi3TvEX7tG/PUowIS7V27KVK1HgRdCCCxSkpxOzmmu9bBi3MaNG6lduzbXrl3Dw8ODV15
5hcaNGz+bF5jNFShQgP79+/f39zm97frEWFPhERERERERZFnyM7ewVzOq/xSCy6cOcGZ4/eLfpdjzhJ79hZ2O
R1xcnbFOdkV+5yO5sJfkk1OLtoHcpFA8/VyptzC694FvO7F4pxyA6eUm/dvyTexIylTsydhxy1bNxJs3Ii3cTN
/fcvGjXhbd+7YOJOamsqdhFskXL9OQvxhUpLvkcPOARd3T8pWbODBYiHkDXoBJxfXTMnk6OiIo6NjplzrSSQnJ
2MymbCxsTE6SqZ6Xt5fSZ/s9adPRERERERERJAvJYWdHvqBiVH6pBe36jaJNn+HUa9WJwMIlALgae57oqINERx3
kwpkTXL96icS7dOhNTTVf446NM+ftCxHhVJntLg3Y4PZvfvbozPe5+vK9Z29WuHdijdsr/ObSjO3OL7HPqRoRj
hU54FiJ/Y6VCXesyj6naux1qsEep5rscqrNTuc67HCux3bnl/jNpRm/urfnB89efJ+rL794dGKjWxi7XOoR6Vi
BEzmKcOq2I9Exl4k+epDoowe5fjWWnDmdCKlQk4avdKNN73d5/aOPqN/6dYqEvphpRT64v7TUw8PDom3MmDF4e
3vj6upK165dGTZsGKVLl07z2EmTJuHn50euXLno3bs3SUn/Xxi9e/cugwcPJm/evDg7O1OxYkU2btyY5nl/+uk
ngoODcXBwIDo6+oEZIyIiaNSoES4uLvj4+PDaa69x+fJlc/+tW7foOKEDLi4u+Pn5MXny5DTXMJlMaWY3enh4M
GfOHPP9s2fPOqZNG7y8vHB2dqZ8+fJs374dgOPHj9O8eXN8fHxwcXHhxRdfZO3atebH1qpVi9OnTzNgwABMJpO
5uPyg9/fzzz+nUKFC2NvbU6xYMebNm5cm68yZM2nZsiVOTk4UKVKEn3766YHvjWQuzegTERERERERReQ7Y2tril
78QfvkL8WLtJty9c5ursee5cvE8Vy6c49ypKK7FxnD5fDRJSYmYTCYcHJ1xyOmInbODdg45yWFnbzGjLMkmJOk
2OZ8oV2pqKsn3kkhKuEti4iWS7t4lMfEO9xLvAqmACUdnVzxz+xBQuBY++QqSxy8AL29/ctjZPdmb8hgWLFjA2
LFj+eyzz6hatSoLFy5k8uTJFCxYOGLchg0b8PPzY8OGDRw7doxXXnmF0qVL88YbbwDw5ptvcujQIRYuXIi/vz9
Lly6lYcOGHDhwgCJFigCQkJDABx98wMyZM8mVKxfe3mlPCI6Li6NOnTp07dqVqVOncvv2bYYOHcrLL7/M+vXrA
RgyZAibNm1i2bJleHt7884777Bnz54HFicfJj4+npo1a5I3b15++uknfH192bNnDykpKeb+xoObM3bsWBwcHPj
6669p2rQpR44cIX/+/CxZsoRSpUrRrVs383vwIEuXLqVfv35MmzaNevXqsWLFCjp37ky+fPmoXbu2edyoUaOYO
HEiH374IZ988gnt2rXj9OnTeHl5pfs1ScapOCciIiIiIiLyHHLI6Wgu/POpIf4mVy+e4OpsDJdjznD+9FFuxlO
lIf4GSVcvkZR493+HdZhIJRUbW1ty5LDDNocdtjnulwBSU1MhNZXUv36dmgrc/29qaiopycncSOrExP1SnmOOO
+z/VOxOdvMgX25v3HP54ObhhUduH3L7BeCRy9s8CyyzrFixAhcXF4u25OTkRz7mkO8+oUuXLnTu3BmAESNGsHr
1auLj4y3GeXp6Mn36dGxtbXnhhRdoOqQJ69at44O33iA6OprZs2cTHR2Nv78/AIMHD2blypXMnj2bcePGAZCUl
MRnn31GqVKlHppn+vTplClTxvwYgK+++oqAgACioqLw9/dn1qxZzJ8/n7p16wIdwd+5c8uXLl8536b5vvvvmGS5c
usXPnTnMxrXDhwub+UqVKWeR8//33WbpOKT/99BNvvvkmXl5e2Nra4urqiq+v7OOfZ9KkSXTq1IlevXoBMHDgQ
P744w8mTZpkUejr1KkTbdqOAWDcuHF8/PHH7Nixg4YNG2bodUnGqNAnIiIiIiIikkU4ubji5PIC+Qq9YG5LSrx
LQvwNbt28zu34G9y6eYM7CfHcvZNAwsOb3LoZROL8De4k3IL/7SFnY2ODycb2/n9NNtjY2GL6X7utrS12Djnxy
O2Di6sHzm7uOLm64+zqgYubh8WegU9b7dq1+fzzzy3atm/fTvv27R/6mCNHjpiLUH+qUKGCefbcnOqUKGFxwq+

fnx8HDhwA4MCBAyQnJ1OOaFGLx9y9e5dcuXKZ79vb2xMaGvrI1xAeHs6GDRvSFCzh/nLa27dvk5iYSMWKFc3tX
l5eFCtW7JHX/bt9+/ZRpkyZh86Yi4+PZ+TIkfz888/ExMRw7949bt++/dDlxg8TGRlJt27dLNqqVq3KRx99ZNH
21/fF2dkZNzc3YmNjM/RcknEq9ImIiIiIiIhkYXb2Drh75cHdK88/D85inJ2dLWalwf196DKD3d+WFZtMJotlr
ra2tuzevduiGAhYFOwcHf+56BkfHO/Tpk354IMPOvT5+flx7NixdOU1mUwWezMCFnsK/tOBGYMHD2bNmjVMmjS
JwoUL4+joyL///W8SExPT9fwZ9aj3V54eHcYhIiIiIiItlGsWLF2LlzpOXb3+//kzJlypCcnExsbCyFCxe2u
D1qWeuDlC1bloMHD1KgQIE013J2dqZQoULY2dmZD80AuHbtGlFRURbXyZMnDzExMeb7R48eJSEhwXw/NDSUffv
2cfXq1Qfm2LJlC5O6daJly5aEhITg6+vLqVOnLMbY29tTXDg4OfuTj5NlQoU9EREREREREs
o0+ffowa9Ys5s6dy9GjRxkzZgz79+/POHLjokWL0q5dOzp06MCSJUs4efIkO3bsYPz48fz8888Zyt07d2+uXr1
KmzZt2LlzJ8ePH2fVqlV07tyZ5ORkXFxc6NKlC0OGDH9+vVERETQqVMni0NVAOrUqcPO6dPZu3cvu3btokePH
haz5tq0aY0vry8tWrRgy5YtnDhxgsWLF7Nt2zYAihQpwpIlS9i3bx/h4eG0bds2zQy7AgUK8Ntvv3Hu3DmLU4H
/asiQIcyZM4fPP/+co0ePMmXKFJYsWcLgwYMz9L7I06FCn4iIiIiIhkG+3atePtt99m8ODBlC1blpMnT9KpU
ydy5szY6c0zZ8+mQ4c0DBo0iGLFitGiRQt27txJ/vz5M3Qdf39/tmzZQnJyMg4ODGXvZ9+OG
HVK9enaZNm1KvXj2qVatGuXLlLK4zefJkAgICqF69Om3btmXw4MFOTmZ++3t7Vm9ejXe3t40btyYkJAQJkyYY
F56PGXKFDw9PalSpQpNmzalQYMGlC1buI5Ro8ezalTpyhUqBB58jx4KXiLFi346KOPmDRpEiVKlOC///Ovs2f
PplatWhl6X+TpMKX+fYF3NnHjxg3c3d25fv06bm5uT/35lhyJ+edBkqnCivk9tWvr83n+nsbnow/n+TTcvHmT4cOHs
3TpUmJjYylTpgwfffqQRL774IklJSbz77rv88ssvnDhxAnd3d+rVq8eCRPMp3U9zLlz5xg6dCi//dCi//vorCQkJFC5
cmNmzZ10+fHng/qlVc+fOtXhMgwNWLlyJXB/Y+CuXbuybNkyfH19+eyzz6hX4Icffk/Xr5557IcffkhOlRVrVyIiIpg3bx7/+/v7M7/zf3Xqc6jQIVxcXNizZw+++euZeG9Idcw1OdDSffPJJJr8jI
iL3PeufvORERNLrpZdewtfXl3nz5hkdReSp0mEcIiIZ1LVrVyIiIpg3bx7+/v7Mnz+fevXqcejQIVxcXNizZw/
Dhw+nVKlSXLt2jX79+tGsWTN27dr10OGteu3aNqlWrUrt2bX799Vfy5MnD0aNH8fT0tBjXsGFDZs+ebt7v4Og/
vqLL75g9+7dbNu2jV9//ZW2beJFTCYTJ0+e5MsvvbBhERERGR7CAhIYEZM2bQoEEDb1t+fbbb1m7di7i1
r1qwxOprIU6dCn4hkaxEREZl6vTt37rB48WvhjvLy8uHPnDv/+97+ZOLmmXSN1PQ2y4IfhGkG5+Q7eRPLJdrrB
DBgAG3atGHNmjX4+T149uLUqVVPAuDWrVv4+/tz+/H79+7/+eTbNXxrlz54D7G+BWrlyZ1NR
UatWqxZAhQ/j999/x8vKiR48e907dm+jo6Ex9P/6uZMmj3ik3XLjjhpX7Bn3qkYB4qrxkdthtHD7lp8j7Yp8w
gWLx4s/cVqF5HsSoU+EcnWouxyZer1btNv3/6ll1Nui2un0rqwdV8EDR/wfJG3TZhMJi5i56BXLTzvWB11296XdKV6tFt8F
vc3DnNnL5+NKgTSdeermdecwNm5zs2LWRqjVr4eLmTslK1Wjb7y1cPb0A8Chelk0//UC+BZx4fLnkXZ
smKpSTmdCGg4ctEPfDZM4/KfCIiIiJiNEdHR9auXWt0DBFqNAnIpBji4uFCtdjh8+m0a+oCK4587D5D5p9/JGr
fbnzzFOgzPvHuHuHeZPGku1Ji1wcnlwkQ/g4ploVn37NU07dSOsex+OHQjnq7HDyWFnR+2WLwNQpnotKtVvhHfe/
Fw4c4pvpk5gTLf2jFu4HFtbW+OHFtbW+q0epXTUYfo36QWrp5eDJo2g/jrcSz8+D1GdN/PIrNiaK21+6IJVXQGoJqnw0ZoQe4aQyyd
r7YOoIiIiIij6ZCn4hIBvWd+AmWRZmWgr3ayUeYu6R7BLIU+eKw/CkxJXbV4yoYI
Qil3cC3AQgKDuHMOcOsXjjPXOir1qSFeXxgseIEFgum90uVObhjK6GVq5PDzo43u070t/vT+LUunIyMYMe
6lUz+cSO/zvyUWWOG89ZYnMzPh3RAREREZHnhY3RAUREshrf/AV4f/4SPgB5jH3k
pKYPKA7l86f471ZCx85mw/AI483+QoXtWjLW6gIl2POPTxHQCBunl5cOH3qgf0H/tjCmWNRNGrXmYjt2yhbow4
5nZyoOqgZB3dsS/8LzsYKFCiAyWRKc+vduzdw/4CTWrVq4ebmhslkIi4uLs3z+PLzb/2X1M4zdq1T4/NDKCg7Bui0g9QVlEJf1o/U3B8b8uH5dF3Ed9N8FqUc
uHo6EiePHlo3rw5hw8fNvdfvXXqVpk2b4uLiQpkyZdi7d6/F43v37s3kyZNPlZw+F43v37s3kyZMf6zWLiIiIiEj2pUKfiMhjyunkhZKe
3D/HX49i3eRMv1mkA/H+RL+b0Sd6bvci8h96jvFDmRc6fPG7RFnPqBHn88z70MVcunOdm3DU8vb3T9CXevM9
9+h+6gPsLW1JSUmeR79wBIvrdSqtrVz5055iYmLMtz9vx9vb2xMfH0/jxo1p2bIlkydP7nAOTfJvm3+xvOsdDf9DMKJzkVg9OX/Nt9UDeT7zD/3Wd6MjwXj2l6MG3nszg36J5BzF8b8y/uhnu61y3zJQ6KuhvEVh4bPwa8eV9rafiNMfTzn+p8M5J0e8QzAU56WN0w8ntd9PWPx4sA0TO4NWQxZk9nfwmv1PSlwz+1vvMpWcyvL0r8SWc9s4c0+fc5HSzS0RD6pCQiBzp+Ldr3aTOmxnHPuf6nb9vwJoa3XDrJaeiEeG4apZVrWBsx9sCDcerpRzT5rvj6Blq9+XPqlOXK9UtnXi16B7t9/UjS+nhk8I0ZAOFPEROv
qEBmrp7ZumyYBS0WrzpLadF8PnfYVKhyNGpbpMW+5IbEOKlD9hvrDuO5F

```
PFH4uLi6NSp0xNdZ+HChezZs4ed03c+ctxnn33GW2+9xa1btyhWrBhr1qzB3t4euD8DrGfPnhQqVIgCBQowa9Y
sjh49yty5c9m2bRs9evRg9erVlC9fni+//BJ3d/cnymwN5s6dy8CBA9m+fTvbtm2jU6dOVK1alZdeegkbGxs+/
vhjChYsyIkTJ+jVqxdvvfUWn332mfnxCQkJTJo0iXnz5mFjYOP79u0ZPHgwCxYsIE+ePPj5+bF69Wrq1avH77/
/TseOHUlKSqJnz5589dVXmoEpIiIiIs+EZvSJiIgAs2bNolGjRvj7+z/2Nc6cOUO/fv1YsGABOXPmfOTYdu3as
XfvXjZt2kTRokV5+eWXuXPnDgDu7u588803nD59mk2bNhEcHEz37t358MMPWbBgASdOnODIkSM4OTkxerRmZ6Z
HaGgo7733HkWKFKFDhw6UL1+edevWAfeXaNeuXZsCBQpQp04dxowZw3fffWfx+KSkJGbMmEH58uUpW7Ysb775p
vnxJp0J7777jvfff58SJUpQpkwZXn/9dSZMmEDt2rXJmTMnVatWpVixYkyfPv2Zv3YRERERsR6a0SciIlbv9On
TrF27liVLljzRdXbv3k1sbCxly5Y1tyUnJ/Pbb/78xffp07t69a57Z5e7ujru70OWKFKFSpUp4enqyd0lS2rRpk
+a6s2fPxsPDg+bNmxMWFkaLFi2ws7OjdevWjBgx4okyW4u/H4ri5+dHbGwsAGvXrmX8+PEcPnyYGzducZ
cuUNCQgJ0Tk4A0Dk5UahQoQc+HqBatWoWszijoqL4+uuv2bt3LzVq1KBfv340atSIkiVLUqNGjQce0iIiIiIi8
qSe60Lfp59+yocffsiFCxcoVaoUn3zyCRUqVDA6loiIGCAiIuKpXfuzzzz7Dy8uLwMDABz7PyZMnATh06BBubm4
PvY6fn1+aYuHw4cMpWLAgr7/+unnftr9LTEwkJSWF48ePp3n+q1evMnz4cObPbnUtERATXrl0j0jqaiIgIjh07x
q1bt57Ke1OyZMlMv6aR7OzsLO6bTCZLLsbTCZSUlI4deoU//rXv+jZsydjx47Fy8uZLzZs306VLFxITE82Fvgc9PjU19aH
P1717dyZPnkxKSgp79+6ldevWODk5UbNmTTZt2qRCn4iIiIg8Fc9toW/RokUMUHDiQGTNmULFiRaZNm0aDBg04c
uQI3t7eRscTEZFnLMou105bkpKCt8v+4lqLV/hhKOPRd+1S7HEXY7l+LkrAKw/GYuj8y1y++XF1eP+aboj071
MhXoNadz+dfDIBR75LZ/AyY1UL19SgysRBVw4c5qtv/xEqao1cfPy4sqFGJZ+OZ0cDo7krdM8zeuc9uEIGnbuy
fV8wVwHAspX5YcVy/GvOZi5S36iYLlKT+W9yV5lvofbvXs3KSkpTJ48GRub+ZJ/H3ZkbkbNmjULLLy8vmjVrxrV
r14D7S3///G9ycvKThRYRERReYjndo++KVOm8ZMWMGTk5OfPXVV0ZHExEExGRbGT/1t+4fP4cd
cNeTdO3euHXDG5Zn8+H3z/BeHj7lgxuWZ+d61ebx/z9BOV/Ym/vwKHd2xnbvT1vNqjKlIE9cHR2Ydy3Dy3DPldt
i7N7fN3Ih+hQN23YytzVq1xnvgECGvdyEe0mJvNx7UMZesFgoXLgwSUlJfPLJJ5w4cYJ58w4cYJ58+YxY8aMx75ebGwsY
8aM4ZNPPgHA090T4sWLM23aNLZt28a6deuoWrVqZsUXERERERbHwXM7oS0xMZPfu3bz99tvmNhsbG+rVq8
e+Ji7d+9y9+5d8/3rr168DcOPGjacx8m8+k+eR/3fjhvNTu7Y+z2dLn2X2kRU/y6KlyzFv15EHPkftZt1p2
rn7I/NM+WndI/MN+3yORX90Z2cGTvn8kdf8U7Ey5Rjx1ULuJNyyyaO81ZtIjH5cZnuZn+awlJyeTmJho8TPBvXv
3SEpKomDBgowbN44JEybw9ttvU6FKFUaMGEH37t25ceceMGNjY23L59G7D8mSIhISFNG0CvXr3o3bs3LI4u5r5PP
/2UHj168NFHH9G3b1+KFSv2zH4+edb/Ens/Pnz5M2bl61K5cmVz7+cmVmTZ/+/1ltmHPsvvsQ5/lP3N3dzc6goiIiIjVe
i4P47C3t6dcuXXKsW7fO3JaSksK6desIDL5FChWYNm0at27donP
nzkZHExEREREREREREe48t4W+V155hUxLJFixAguXLhA6dKlWblyZXQ0ix7n2X2X2o
c8y+9BnmX3osxQRERGRrOC5PHVXREREREREREREMua53KNPREREREREREREMkaFPhEREREREREREkWxAhT4RERE
REREREZFsQIU+EREREREREREGRbECFPhERA927d4/Ro0dz9uxZo6NI7pg9ezYJCQlGx5ezYJCQlGx5BMdOzYMVatWsXt27cBO
BlmIiIiIvI806m7VmDgwIHhpHjtlypSnmEQYw4dOhiUSjLK1dWVAwcOUKBAAa0jyBP
y8fHh9u3btG7dmi5duHlyoClShWjI8ljunLCq+88grr16/HO09GTy5MlGRgoKCeP311/H/HO09GTy5MlGRxQRERSSOHO
QHk6du7d2+6xp1MpqecRDLT8uXLadu2LTY2mlqPi+Dt5LheS4uVl8fiaTSYW+LKROnTps2rRJhb5s4Ny5cyxfvpvpw5c+ZQq1Y
tgoKC6Ny5Mx07dsTX19foeCIiIvKMqOgnIiIiIplm0qRJ5ObmvvB3rwBn2BZmCmCl2TfgPSS/xyOv+ae1X1z4C1VbNHmDKbp9v
zcOJycno+PIE5gxaJRo2iXbt2lCtXnqzJ3ZGXDmdnZ4v+C1wmcfqipWmyp9BzJeRT9sFiGoMeBLBj1Su7ZkHvBWX1TkQBHXuXOnBw
Oc8bzztfX11WrVlGVClcXV0JDw0nS7h8BoaSnx8vNERRURESDDkcMGaVPRXgG72+ET9Iw+kSzq3LIz9O3bVOW+bKBXr17Ag
5fOm0wmkpORTn3UkyQQ+Pj5UaqaK1aNqKiOHDgAB07tsTTO5PZPZs2dTq1Ytyo1YtoyPKI9y6deuBf79evXXoVBXcHAxK
JiIiIPwzFfqsOK5du/juM5qmMTHRom/JkiUGpmCgpZKMatCgq6/omRtFy9eevXOBf79evXoVBwcHAxK
ZsTJO7QokULVqxYQb169uzD592uio8gjVqfn66+/5v333332KY0YOHEitWvXinXNjidiL
IiMiDaemulVm4cCEd0nSgQYMGx2fPNjqipNOTt3JiQkBDs7m7t+L
ffMmu7cuUPOnDmNjiGPqWnTpTogJeXl8WY2NhYH19VeB9zkVERFC3bl3Kli3L+vXradq0KRcuXMDBwYEc2PcNNi+r
sGQcPHuTq1ats2bKFQoUKGR1RRERERCQNFfqsTGhXHR61B5fW
u6ZtSQnJzNu3DhmzJBxYsXiYqKIigoiHDhiOghQAG6dOlidERJRpy5dutC1a1qVc6780DGpqalERQcTGBj4DJP
J47h+/TqffPIJTz/72+nrJly9K7d2/8/8/PyMiYihMjZZBaR2hKcMfNV0fr5u2EcbBhXlhbfd7l4b
tSpQ0xMjNERRaz6gNr6/gM1NqeA9gbhqaxLPm09i2BvZESUdkSaNiwITNzmMkBIkSJGxE
REREDESukY/+sjKenJzdv3gQgNSLSEQVdzv3gQgb968REAEABXF0dCQOKUSs1tdff80XX3MjcPG5hMM
sL0zo79+/cbHUMy0e+/0779u2pUqUK586dA2DevHls2bLlH2Zmngi9kgmCRsvSJzjDfgeGdc8rsgkAGg1p1Xq2MeVoS
```

eeIM2bdpQt25dg9NJRm3atImmTZtSuHBhChcuTLNmzfj999+NjiUZdO7cOQoXLpymPSUlhaSkJAMSyeNq3749s
2bNMjqGZILFixfToEEDHBOd2bNnD3fv3gXuL+cdN26cwelERERERB5Mp+5amenTp3PnzhOA/vOf/2BnZ8fWrVt
p1aoV7777rsHpJCPmz59P586dCQsLo2/fvgBs2bKFunXrMmfOHNq2bWtwQkmv4OBgfv/99zR7tv3www+UKVPGo
FTyOO7du8dXX33F2rVrKVeuHM7Ozhb9U6ZMMSiZZNSYMWOYMWMGHTpOYOHCheb2qlWrMmbMGAOTiYiIiIg8nPb
oE8miihcvTrdu3RgwYIBF+5QpU/jyyy+JJly+jYsSNvv/O2oOePZtSoURw5coSvv/6aFStW8NJLL
xkdUdKpdu3aD+OzmUysX7/+GaaRJ+Hk5MShQ4coUKCA+fCqoKAgpw4QXBwsPkfzURERERRniea0WcFbty4gZu
bm/nrR/lznDz/Tpw4QdOmTdOON2vWjHfeeceARPK4mjdvzvLlyxk9ejTOzs6MGDGCsmXLsnz5chX5spgNGzYYH
UEyia+vL8eOHaNagQIW7Zs3byYoKMiYUCIiIiIi/OCFPivg6elJTEwM3t7eeHh4YDKZ0oxJTU3FZDKRnJxsQEJ
5HAEBAaxbty7N3m5r164lICDAoFTyuKpXr27eP1NEjPfGG2/Qr18/vvrqK0wmE+fPn2fbtmOMHjyY4cOHGx1PR
EREROSBVOizAuvXr8fLywvQbJPsZNCgQfTt25d9+/ZRpUoV4P4efXPmzOGjjz4yOJ08jsTERGJjYOlJSbFoz58
/vOGJ5HHs2rWL7777jujoaBITEy36lixZYlAqyahhw4aRkpJC3bp1SUhIoEaNNGjg4ODB48GD69OljdDwRERERk
QfSHnOiWdjSpUuZPHmyeT++4sWLLM2TIEJo3b25wNCsmIo0eP8vrrr7N161aLds2OzXoWLlxIhw4daNCgAatXr6Z
+/fpERUVx8eJFWrZsyezZs42OmQnJzMli1bCAONxcnJiWPHjhEfHO9wcDAuLi5GxxMRERERReSgV+qzM7Nmzc
XFxoXXr1hbt33//PQkJCXTs2NGgZCLWq2rVquTIkYNhw4bh5+eXZnl9qVKlDEomGRUaGkr37t3p3bu3uHQCHggU
L0r17d/z8/Bg1apTRESWdcubMSWRkJAULFjQ6ioiIiIhIuqnQZ2WKFi3/KK//3bpx5MgyGYPgVMnKleuDMC2bduYO3cu48PPi9bpgiLLtQnJ6BxjwvXrXgGh1
FRERERCTdbIwOIM9WdHT0A2cnBAYGEh0dbUAiyQgvL68HzmDdt2+fVtrUAiyQgvLy8uX74M3D9kxcvL66E3yTqCg4PNn6tkbZ6enty8eROAv
HnzEhERAUBcXBwJCQlGRpMMGjx0ErmYNZ7vNn48hzw4bRqVMnKleuDMC2bduYO3cu48ePp2PHjkbFlAxat
mwZQ4cOZciQIVSoUAGAHTt2MHnyZN577z3u3bvHsGHDeOWVV/jwww8NjSoiZ0hPSpyI9DyIGTy90FZJK2r1ZKvBKrp8NeusWJd7EVHVH1fmC3aDjGpfBnM7a0bW6lqtlnaQVlggT9oUmABPTnqD/GPm6DVMHrghM0QvJeHGLbaYcTsSIYIkiQKw5NC5ggRyy2WaEppOQ9eNpzJbh3+GkHDX7Q
y5Qre3t4qDj3n/qmI8FcqKIiIiIiEh6qNBnpaKioggaPD8fR0ZGQkBACAwNjiQZZGNjw4ULF9IU+s6fP0+OH0/+D/wQ4OgkhWvn9emwn3xGNvZEwJ3DhaVBkbFNlPmaGskKzzdB8idNtSpB9MH0gp/WFL8KAA+zLHE2xJ8A8I9CDy4XKHBUp2ZZC6F1HkhFjbgmemkNPE9/rkBQ
+/fpERUVx8eJFWrZsyezZs42OKOmQnJzMli1bCA0NxcnJiWPHjhEfHO9wcDAuLi5GxxMREREReSgV+qzM7Nmzc
oW4ffu2QclErEtG9mvTMuysY//+Q9sN5lM5MyZk/z58+Pg4PCMU4/z7/UEv5Q/4E7ULXXOP8rpqqcbsFXaDdqb3N/9WEVZHn5Ecn74rV9ep6ugYiPhE6sH+qTjW9DukgAOuO2h9ztPTZBHBIMTrEw0O8EcoYgX0R/KjQdhoGKbmtrQ8i5aaCfM8uHGNoOwqFoTH80DT/KudKjpSgZJLBVWMEzTOEIcBQWhEqjW3dKmBrmmn3GsgyVD0mWBi05nb6zLNDlKKzy/eXMIUzA/mzVQsGbPqZHYBm2Gh6O4mcMJsqjOm07l4SVAx7GNdAJbxpeZX6kTbHuVShL6TiGhKtTrWVNBRLMS6ZgZZXs2OLGwKBZWJ9TbMFT/HMjuv+Ifd+WS3tCbIdNlO/wDSgjLW/8j62lsXI5NPxvUGACDXsK2CgHCMH5bUBjyN0i8ytKdtPMqJI0x4cI4T5SHv5CrGGZ5wRT+tb3WeUX0TyaZlABTvZ/HNz0kl2HbvDZxnXFECAtTd07ZLS9jZy/zFzxUHFthRqa6z2pj6z7OMVfAKCX4wvJJeZ82j/o2ZWLtcxQlcbDAmg8rQkxIaVHZeqi/vqUhwlXTVchrn0pnbv/uEQpfFzyRBLRVXwWg5y7c07Z4Ud64uu1J4/cEKtm/E4bJ2PZ0UVXN4kXEEGgbQnEWCjHD97UQFqm/RKXpbWmXLEvJ8hXkQKjZgz/QCuFDWU7vRN+wQhA1/Qq4J6pOuS6FbgPrKBF3Hj6KhSuaHOJhe7/WEsvpnkVi8LC5nXJDlydJhRUnegEAQRR6YQnMfmF3hL
FTyOO7du8dXX33F2rVrKVeuHM7Ozhb9U6ZMMSiZZNSYMWOYMWMGHTpOYOHCheb2qlWrMmbMGAOTiYiIiIg8nPb
oE8miihcvTrdu3RgwYIBF+5QpU/jyyy+JJly+jYsSNvv/O2oOePZtSoURw5coSvv/6aFStW8NJLL
xkdUdKpdu3aD+OzmUysX7/+GaaRJ+Hk5MShQ4coUKCA+fCqoKAgpw4QXBwsPkfzURERERRniea0WcFbty4gZu
bm/nrR/lznDz/Tpw4QdOmTdOON2vWjHfeeceARPK4mjdvzvLlyxk9ejTOzs6MGDGCsmXLsnz5chX5spgNGzYYH
UEyia+vL8eOHaNagQIW7Zs3byYoKMiYUCIiIiIi/OCFPivg6elJTEwM3t7eeHh4YDKZ0oxJTU3FZDKRnJxsQEJ
5HAEBAaxbty7N3m5r164lICDAoFTyuKpXr27eP1NEjPfGG2/Qr18/vvrqK0wmE+fPn2fbtmOMHjyY4cOHGx1PR
EREROSBVOizAuvXr8fLywvQbJPsZNCgQfTt25d9+/ZRpUoV4P4efXPmzOGjjz4yOJ08jsTERGJjYOlJSbFoz58

dVXBqWSjGrSpAlDhgzh0KFDhISEYGdnZ9Gv/RazjoEDB9K7d2/u3LlDamoqO3bs4Ntvv2X8+PHMnDnT6HgiIiI
iIg+kpbtW5pdffuGtt95i5MiR5gMb/vjjD0aPHs2ECROoVq2aeaybm5tRMSUdXFxc2L9/f5olgidPniQ0NJSbN
28alEwyatSoUYwePZry5cvj5+dncYADwNKlSw1KJhml/RazlwULFjBy5Ei0Hz8OgL+/P6NGjaJLly4GJxMRERE
ReTAV+qzMX38J/bOY8Ocfgb/e1y+kz79cuXKxYsUKKleubNG+detWmjRpwrVr1wxKJhnl5+fHxIkTee2114yOI
iIPkJCQQHx8PN7e3kZHERERERF5JC3dtTIbNmww0oJkkvr16/P222+zbNky3N3dAYiLi+0dd97hpZdeMjidZER
iYiJVqlQQx0oaIPIST5NOURYRERGRLEEz+kSyqHPnzlGlGjRg2uXL1CmTJlgPuHrfj4+LBmzRoCAgIMTijpNXToU
FxcXBg+fLjRUSQTbNq0iUmTJhEZGQl0aAcHAwQ4YMoXr16gYnk39SpkyZNEvnH2bPnj1POY2IiIiISMap0GdlVq5
ciYuLi3kvvk8//ZQvv/5JC3dtTIbNmww0oJkkvr16/P222+zbNky3N3dAYiLi+0dd97hpZdeMjidZER
iYiJVqlQx0oaIPISTk5NOURYRERGRLEEz+kSyqHPnzlGjRg2uXLlCmTJlgPuHrfj4+LBmzRoCAgIMTijpNXToU
FxcXBg+fLjRUSQTbNq0iUmTJhEZGQlAcHAwQ4YMoXr16gYnk39SpkyZNEvnH2bPnj1POY2IiIiISMap0GdlVq5
ciYuLi3kvvk8//ZQvv/yS40BgPv30U4vTeOX5d+vWLRYsWEB4eDi0jo6EhobSpk2bNAcAyPNn4MCB5q9TUlKYO
3cuoaGhhIaGpvn8pkyZZ8qzjyW0aP38+nTt3JiwsjKpVqwKwZcsWli5dypw5c2jbtq3BCeVRRo0aZf76zp07fPb
ZZwQHB5u3SPjjjz84ePAgvXr1Yvz48UbFFBERERF5KBX6rExISAgffPABjRs35sCBA5QvX55BgwaxYcMGXnjhB
WbPnm10RBGrULt27XXSNM5lMrF+//imnkcxSvHhxunXrxoABAyzap0yZfmme5SfPv65du+Ln58f7779v0f7
ee+9x5swZnYYtIiIiIs8lFfqsjIuLCxERERQoUICROIcSERHBDz/8wJ49z2jcuDEXLlwwOqKk09dff/3I/g4dO
jyjJCLyJwcHBw4ePEjhwoUt2o8dO0Obki5Ws5+eOQckko9zd3dm1axdFihSxaD969C3jly5fn+vXrBiUTEREREXk
4HcZhZezt7Ul5mKQl5cXN27cMDKaZFC/v0s7iclJZGQkIC9vT1O0KQl5cXN27cMDKaZFC/fv0s7iclJZGQkIC9vT1O0Tk4q9IkYICAggHXr1qUp9K1du
4HcZhZezt7UlISABg7dq15mKQl5cXN27cMDKaZFC/fv0s7iclJZGQkIC9vT1OTk4q9IkYICAggHXr1qUp9K1du
1b7ZmYxjo60bNmyJU2hb8uWLeTMmdOgVCIiIiIij6ZCn5WpVq0aAwcOpGrVquzYsNFixYBEBUVRb58+QxOJxl
x7dq1NG1Hjx6lZ8+eDBkyxIBEIjJoOCD69u3Lvn37zCcpb9myhTlz5vDRRx8ZnE4yon///vTs2ZM9e/ZQoUIFA
LZv385XX32lg3NERERE5LmlpbtWJjo6ml69enH27Fn69OlDt1atWgAAgcRHVUW2Uc+MVwGtgD2zDG7ZM21fT5P
fwAo8Tf9XDgGJRGGAkKJgZ7VSrxnN8fQNhpdx+ppN2UDMhvTPFeXrSxZQRhUVBVCLLEf7Js5ZcWtKvljbNnKtq
GjWq0ihOzFPl+uDf5+/N42ZxZavf+zn3vP1lgOk+YfffTX2DUjxfcbqsEFkzcGDtIc9g+w4lv+FkCw9PqX/JYSk4
ZmWpKCUlXA6XD4q9Pb7RXrjRRvOaEeNIxKWSV+XXCTqdo0zBZkd5wqnjsL1G0y+4WrnuhLtMGfDhFbWpWWH8FtV
Ln7VbRGScp2jl0WF+6W47/hzmL0u5GmS9sL6dWqoeGrQrMDhQRqkGVKJwo9zLfKgz0VjoylLj54tRCqT0jGq7O
S6h+hOZknY07rvOpj4ncc0ykQt9DYJCdAuDHmtW+9NHtXzgcHE99NhGNJdfvazmyYzVk1V4nhZ6O/hsZtFWIzv6
aJq2gmEMnPeT7UeHbW4GpGfRkWmdh6c4pnvVqzRq8JCnB2DPuftqX5BSdPBzPE6+y/fxGAmvvfrnv7pxDDMn
DZUwLyOLdGZbXhgyGOJP6MWR7x4vjFTuGYcprNWsD1XYQqxPkBmyGdLGsqLYLmj3sp/Rfe6VXN5q95ypPQ6W+Dn
2kz1YO8FhNL6kNp8ccPr0ouZfeDC5WhaY5V5uymVpQ/Tx9iyxY35U0flbmk5+2WZPVBrL2WcNsW0Qx5bRqJhN
gw8D6B0kW2D7jpMyQQhUzRiSWiqsrj9CiWxpEtEGPx5ZlSWxTq8amcRIR1bK+pqeeFbrdMPsvu8x6hvNeFtwKl
GeJjtsq9f0rmRzVFoq0Xt5xVmfO2Wj3A8zy50sLXjpL2xtffyZfvdqXaq3MFpSPT3E+LxNQgr6gPDtA+f3FQtp
+Z+zJhXrY7D7x+cXsSD6pGkzxqeLvTadR2VPtvEO4BhhVX0FGy86c+UMVmS6YmEtZFjzIrbpuWf/8u4O9z/vTqo
CnXGz4zEF4Sh0MkNERERERDq0+dwF+Rk7j23N/J4a9eGGZkJlmXb+uwpn0GmdyPa3ORVf8+9s/H5bpwfM5sx9Sce
ctV5Ea/7eNOrZ+h/o0v//gm6dMT8WW5PPbMtAyzaT7tX//1J/r6Jvf5gtRCcI/9/K9ov9jlymgcD2GxKKRbO5ObN
8+9dHn7Y/3jOTgb4EmfdWA4XHNgnz//Ae7xZz8o8O9PR3ecUHuAdW4nJ6G7dKQKr2cmZ9eGmqRVCLfWXJnWcVmwH
5c/rNgGuCxBVvFAAB19JbkjUZCkJb/Dy24pHN1ZLOxJiTBf7a7UtWyQv8nCa/P76hhsRu7qE6VVKe08ArnJ1a0d
bVjGeb2sPo7wKgH+U2vjR7SXqWKJmG5A09RvqC2NyvmvtyV2/rMYs2WmXD1+tqvnZgCqoDdaxZb9d7YX9smujP
/aMBw4ihsZGvBKp2HLmTq+7QdNyp3XZ6Rj+mxOw1t9wDHZqybHAb7nXcX1DSMT5Hto6uDFhDyW2XuHoWTEU0rm/
+a2P4NmN3cDXXGDhNUvJb8uMvdR7NE8H0BjhfzKgc6rGLH+q9xDE2/37n+6x+xSixRT/uHy0rwmRc9u5P3U7un
tHoEfDZTW9NjxTenSAgHzWrSJDW6uxgnX1DU+eMR9ufx6smunFo1FVNVPsuWNsxxXtSNtqb7kbDmcL3SMMbSRgi
yA1e+EHDczD/jXB0tCn9PwKEFTRzBCVnoj6Szi+Lkm2WYX6OzQZHT60k92XFvvqtvB8IPJN0EeKTR1pPIdgDS3
o8yxnyc1SF/Wp++EE4QzsVf5WD0RvD/8eFdXQTSkVTrsA7g42dXhx+/v2jVJxKhBrWHNXW9ml2u+O+PTTGZ5MLD
zoFXzX9kCxMr5Ju4kd7PXdqq8pDH7n96DH1BJlsPLyRoG3ebT4ImN5tAXYLgqhiBMT+/ecvLfGPX7OmKTPvrHlG
T4sfbrR4rfXK1k04fyF+Cnt+dmPjL1EN4TNymdw+JmCnddQqzqhc2+z85JCu2jJmzwH8fh/6mnxVS4ppz5sjWmY
8F7rcCRiUl92Srhnki2B/2PnjdPjUqEDHdyGg8HhK9u3nuaGIfiQVJsd0HTSF8L+e1mfWaMftgTCN+Fn9XrNAg3+

```
j/fffx9XVlf379xMYGMirr75KSkoKP/zwg9ERJZ2Cg4MZN24cLVq0sJidGRERQa1atbh8+bLRESWd6tevT1hYG
D169CAuLo5ixYphb2/P5cuXmTJlCj179jQ6oqTT7t276dixI5GRkfz9RyWTyWQuyIuIiIiIPE80o08ki5o4cSJ
169Zl165dJCYm8tZbb3Hw4EGuXr3Kli1bjI4nGXDy5EnKlCmTpt3BwYFbt24ZkEge1549e5g6dSoAP/zwA76+v
uzdu5fFixczYsQIFfqykNdff52iRYsya9YsfHx8NOtdRERERLIEFfqs0K5du/juu++Ijo4mMTHom/JkiUGpZK
MKlmyJFFRUUyfPh1XV1fi4+MJCwujd+/e+Pn5GR1PMqBgwYLs27fPfDjHn1auXGlx0Io8/xISEsyHOaxevZqws
DBsbGyoVKkSp0+fNjidZMSJEydYvHgxhQsXNjqKiIiIiEi6qdBnZRYuxEiHDh1o0KABq1evpn79+kRFRXHx4kV
atmxpdDzJIHd3d/7zn/8YHUOe0MCBA+ndzuzd37twhNTWVHTt280233zJ+/HhmzpxpdDzJgMKFC/Pjjz/SsmVLV
q1axYABA4D7+7y5ubkZnE4yom7duoSHh6vQQJyIiIiJZivboszKhoaF0796d3r17m/cCK1iwINz7d8fPz49RoOY
ZHVHEKi1YsICRIOdy/PhxAPz9/Rk1ahRdunQx0JlkxA8//EDbtm1jTk6mbt26rF69GoDx48ezJ22+/8euvvxqcU
NLr8uXLdOzYkQoVKlCyZEns7Ows+ps1a2ZQMhERERGRh1Ohz8o40ztz8OBBChQoQK5cudi4cSMhISFERkZSp04
dYmJijI4oYtUSEhKIj4/H29vb6f/H29vb6CjymC5cuEBMTAyJr+XL1/Paa69x48aNN
H06jENEREREnlc2RgeQZ8vT050bN28CkDdvXiIiIgCIi4sjISHByGgiAjg5OanI8X5+vpSpkwZc5EPoEKFCir
yZTF9+vShffv2xMTEkJKSYnFTkU9EREREnlfao8/K1/K1hhSEkLr1q3p168fzf69evZZ82aNdStWtW9foeJJOq
ampnDlzBm9vb3LmzGlOHHkMZcqqUSfcpnnv27HnKaUTk765cucKAAQPw8fExOoqIiIISLqp0GdlpK+fzp07dwD
4z3/+g52dHVu3buXVu3bqVVq1a8++67BqeT9EpNTaVw4cIcPHiQokWLGmY2NWR0HHKOLFi2MjiAijxWFsaGDRsoVKiQ0VFER
ERERNJNe/SJELlSpRg1k2bRpNmjgjCNpjY26d4LTBuMZx2enp4YPJxWiZiQkFQkzWEcfv2v2NSCr0WZmi
DuLm5PcUkkpmWL1/Ox4eniiQUWuVHrJTyMMzsenrYWtvTvDhw+nYMGCRseeRJ/SozKzPEf9r9/SquQQ1f2UkpmWL1/OxIkT+fzzzzylZsqTRcSQT7Nq1i8jISACCg4MpV66cnjkKkuvjevTvDhw+nYMGCRseRJ
/Soz9BkMnHixIlnmEZEREREJH1U6LMCNjY26d4LTBuMZx2enp4kJCRw7949703tcXR0tOi/evWqQckko86ePUu
bNm3YsmULHh4ewPODcqpUqcLChQvJly+fsQEl3dzd3dm3b58KfSIiIiIYgjt0WcFNmzYYP761KlTDBs2jE6d0
lG5cmUAtm3bxty5cxk/frxREeUxTJs2zegIkkm6du1KUlISkZGRFCtWtWDIAjR47QuXNnunbtysqVKw1OKOnVokU
LfvzxRwYMGGB0FMkkiYmJnDx5kkKFCpEjh35sEhEZHnm2b0WZm6devStWtX2rRpY9H+zTff0MUXX7BBx40Zjg
olYMUdHR7Zu3UqZMmUs2nfv3k316tVJSEgwKJlk1JgxY5g0aRJ169alXLlyaWR4aU5qjYqKonTp0iooZDHHjx9n9uzZHD9+nI8++ghvb29WrlyJ7zI+aQIqbmYlFqe62v6pNb2ZUVxyeYjhJ5qNZUZdZNmYVqNp0jEpe1wWZzcj33N7wUTXb
AoqKiCAoKok+fPuTNm5dhw4YZnFBEREREJCOV+qyMk5MT4eHHhaU5qjYqKonTp0iooZDHHjx9n9uzZHD9+nI8++
ghvb29+/fVX8ufPT4kSJYy0J+lUtGhR5s+fT4UKFSzad+zYQbTl27JhBySSjtK9b9tGvXz+2bNnCtGnTaNi
wIfv37ycoKIhly5YxcuRI9u7t3da3REEREREREZE0bIw0IM9WQEAAX375ZezbN2dd06vXOTLEqL5iK36O1fPE
2TJEuLj4wEIDw/nvffeMzidHH35Inz592LVrl7l65d9OvXjOmTJhmYTDLq5MmTD72pyJe1/PjjjOyfPp1
q1apZ7HNbokQJjh8/bmAyyEaEAyEREREZGH04w+K/PLL7cSWLady4scEJJJbQqV65MM6
9atGThwIK6uroSHSHxMUFMSOHTsICwj7NmzRkeUdPrrwSp/7gH259d/X/qpQ1ayBu3rlvU5OTkRERFBUFCxd+
x4eHh1KhRg+vXrxsdUUREREQkDf32YWUaN27MH7M0aNH+eyzzh8+DAAHprR8UcOHCAb775k27t7c3l
y9fNiCRPC4drJJ9aF+37KN8+fL8/PPPvDss8+afzk/adIk8mLS9u/vXLyofj2qt9+4kf1x1K2LM4e0Gp9ea69x48aDwICY
mJs2eYHv37iVv3rwGpZLH0bFjF+jR6MjSCZ5++N23CQ8Pz/dDcXq9ePUaOHKlCXxYbtw4GjVqxKFDh7h37
x4fffQRhw4dYuXKFWL+fh/FvL/MKfq7ZXsB+YF6c/N2fcqlNEE3GLM9GR+YWDlKmSb9k36r/VCBEx7oU+rKg
JREpKCLu2bGHw4MH6HL0o2NjYB35fhoaGGpRIMurHH39wZXFqAsWQrqIqpUKvR9Z6KuzHXy
mXZtm0bISEhRscTEREREXkg7dH6kO2ZNqTt0n0V0vYbcmNd2ILV0V2HXRkI+Z7eJ4cW3uXB0NrYxvVYrKaG0kxV
n07ZtW+bMmY0tra3RESWddu/eTceOHYmMjOl0vfyWbTCaSk5MNSiYZpX3dRERERERETESCr0WZmiRYvSuHFjxm
50Tk9FxJBNER0cTERFBfHw8ZcqUoUiRIkZHEhEREREJL0SiwEUfkHH8fDdJGgcGr1VOfkZ6r7Dl0ZwEpA6rUiVyjFjzkm7YhArbE0vq8YK0aA02fQPc5BSGdgQJS4a3MW1JLWpbzd95rjZT01sOLzKFbPnlG
wdXVl//79FCxYkD59+nD06FFWrlxpdER5hBs3bqR7Jub2GFhoteZUkuUHrN2Ig1Qq7NGhFtN43bVo1Kwxe
rqyd+9eChcubHQUeUKbN2+mUaNNGtG/fnjlz5tC9e3eLfd3KlStndER5BBsbmzhYbjRbNLOf7RFOH7hcsFbjOuRZkq0pVPvVRxJCjrGZSqgDGnmgZjIzK6vmbzhPqyc0Iyay4q65ReLs
BA3bt2qVCXzaQnJzMnDlzWLdu3QP32vQ3dVu/fr1ByShj6tatS3h4uAp92YD2dcvaNmzYYP761KlTDBs2jE6d0
d22bRtz585l/PjxRkWUx3Tp06a+5NBefvsZLfh4ItDBUX8yMvRSPQwdZHGnZ+1s4rO6B
80sxCmTp1qkHJJKMuX75Mx44dqVChAiVLlsTOzs6if9iwYQYlk4yyt7cnPDzcvECUiIiIiMg/m8qiJmzKOWGbWkpIKYMPJD8JNRnY5CIDN1wfddBwvLx3Fk8+tUFqbWG+h/Un8h5RVZjc+VIfiNPdUDv0vwuaNC4lXFC0f
EKflbGxsXlonzb7zb9ln587N119/TePGMXTlX1Op17Vr8Lr3
wNPueRkVFUbp0aRISEgxKJiIiIiIiLycA+v+ki2lJKS8tCbiglZi729vZZ6ZhN9+vShffv2xMTE6Psyi7O3t2f8+
PEULlyYgIAA2rdvz8yZMzl69KjR0UREREREREjIUKeRThLuzDl+++1O3Nm3axKZNm4iKisLPzgZ+TUp5PWTxaU8G2Wsz8bgNgLZDP9t
6N4+X55OrqyrzZXZA+vFvW57eJxtWi9kiLycmk+F/MSpsV4+pYKvp8fKOWDZ5ZhXK/L9Vt/vw+3JyhH3eXGi6Nxu5C2Fh+KJzs4+vE8+t8yyDd3ZtwIhVXMVsHTaoHOgw4+VWsbanFnaU4etmTAJgkQ8wRbSk85
rBjxw6OHj3K4sWLNZtaRERERERERERERE
oUSLNvm5Llix5ltHkCYSFhbFhwwYV+rqY+rIRTO9PcuXKhaenJx4eZ0azbcGfIYHUsyoHHjxhw9epTPP//c/P/Lp
k2b0qNHD83oExEREZH4Kgh7Kb4j7htcUB6lc+f06R47e/bsp5h
EMtPYsWOZNm0aS7R57+B1l+9z2TGmuDWKHmOSNvHrY3ejU2NRhcSBEREWfnXr6n+Fo5HumMbUUa8kmONa/PsYsMSTFIlE72rMRX6U
```

LMyxYsXp1u3bgwYMMCifcqUKXz55ZfmWQsi8uwULFjwoXOmk4kTJO48wzTyJGxsbMiTJw8DBgwgLCyMokWLGh1
JnlBCQgLROdEkJiZatIeGhhqUSEREREETk4VToszIODg4cPHgwzSEOx44do2TJkty5c8egZCIiWV94eDibNm1i4
8aN/P7779jb25tn9dWqVUuFvyzk0qVLdO7cmV9//fWB/TooROREESeR9qjz8oEBASwbt26NIW+tWvXas+hLKZ
MmTIPPITDZDKRM2dOChcuTKdOnahdu7YB6eRx/flvLzpgJWsqVaoUpUqVMi+3Dg8PZ+rUqfTu3VunKGcx/fv3J
y4uju3bt1OrVi2WLl3KxYsXGTNmDJMnTzY6noiIiIjIA6nQZ2UGDRpE37592bdvH1WqVAHu79E3Z84cPvroI4P
TSUY0bNiQzz//nJCQECpUqADAzp072b9/P506deLQoUPUq1ePJUuWOLx5c4PTyj/5+uuv+fDDDDzl69CgARYsWZ
ciQIbz22msGJ5OMSE1NZe/evWzcuJGNGzeyefNmbty4QWhoKDVr1jQ6nmTA+vXrWbZsGeXLl8fGxobAwEBeeuk
l3NzcGD9+PE2aNDE6ooiIiIhIGir0WZmePXvi6+vL5MmT+e6774D7a+/YtWrRIxaAs5vLLywwaNIjhw4dbtI8ZM
4bTp0+zevVq3nvvPd5//319ts+5KVOmMHz4cN58803zIms8zemvL88vLyIj4+nlKlSlGzZk3
eeOMNqlevjoeHh9HRJINu3bqFt7c3cP8U5UuXLlG0aFFCQkLYs2ePwelERERERB5Me/SJZFHu7u7s3r37gfstl
itXjuvXr3P48GFefPFFbt68aVBKSY+CBQsyatQoOnToYNE+d+5cRo4cycmTJw1KJhn1888/U716ddzdzc3IyOIk/
oxRdfZMyYMTRoOIBmzZrh4eHB+PHj+fjjj/nhhx84fvy40RFFFRERERNLQjD4rs3PnTTlJSUqhYsaJF+/bt27G1t
aV8+fIGJZOMypkzJ1u3bk1T6Nu6dK9+iYmLMS+n/qkqVKsTExBiQSB7XX5dzj17FoB8+fI
ZFUeeQL9+/czff++99x4vGF2NvbM2fOHGPDiYiIiIg8hI3RAeTZ6t27N2fOnEnTfvbsWXr37m1AIsmlAInlcf
frOoUePHvTr14/58+czf/58+vXrR8+ePc0HAaxatYrSpUsbG1T+UeHChclL6f9q0aJFFClSxIBE8rhSUlIYPXo
O7u7uBAYGEhgYiIeHB++//mKSlGx5MMaN++PZ06dQKgXLLynD59mp07d3LmzBleeeeUVY8OJiIiIiDyElu5aG
RcXF/bv30+3QUJBF+8mTJwkNDdUSzyxmwYIFTJo0iaJ0Rh4ceHExQURER
EBLVq1eLy5ctGR5R/sGnTJvPXqampNG7cmJkzZ5I3b16LcZpxknVcv36dhg0bsmvXLvLlywfcnwFWvXp1lixZg
oeHh7EB5YmFh4dTtmxZ8z+uSNZz8+ZNvv32W2bOnMnu3bv1WYqIiIjIc0lLd63MyZMnzaft/pWDg4P2Acsi/l7
As7W1pVKlSgQFBRmUSJ6Uu7s7W7duZc2aNeZDckJDQ6lRo4bROUSs3m+//cbMmTNZsmQJ/v7+hIWF8emnnxodS
ORERETkgVToszIFCxZk3759BAYGWrSvXLmS4sWLG5RKREwmE/Xr16d+/fpGRxGxehcuXGDOnDnMmjWLGzdu8PL
LL3P37l1+/PFHnbgrIiIiIs81FfqszMCBA+nduzd37twhNTWVHTt28O280twhNTWVHTt28O233zJ+/HhmzpxpdDwRq7Vu3TrWrVtHb
GwsKSkpFn1fffWVQakkvcLCwh7ZHxcX92yCyBNr2rQpv/32G02aNGH69Ok0bNgQW1tbpn//+0ZHk0ZNgQW1tbZsyYYXQ0EREREZF/pEK
flenatSuOjo68++67JCQk0LZtW/z9/fnoo4949dVXjY4nj8lkMhkdQZ7AqFGjGD16NOXL18fPz0+fZxbk7u7+j
/0dOnR4RmnkSfz666/07duXnj17UqhQIaPjyIEaPjiIiIiIhkiA7jsGIJCQnEx8fj7e1tdBTJgL/PHFq+fDl16tTB2dn
Zon3JkiXPMpY8AT8/PyZOnMhrr71mdBQq/fHH38wa9YsFi1aRPHixXnttdd49dVX8fPzY8S7Y+RSPiJjpHBcXR7FixbC3t+fy5ctMm
mNOAHm2bt++TUJCAgBOTk7cvn2badOsXr1aoOTSXq5u7tb3Nq3b4+/v3+adsk6EhMTqVKlitExRASoVKkSX37
5JTExMXTv3p2FCxfi7+9PSkoKa9as4ebNm0YHExEREZERERERF5KM3oszL169cnLCyMHj16EBcXR7FixbC3t+fy5ctMm
TKFnj17Gh1RxOoMHToUFxcXhg8fbnQUERET4TSUKHPyuTOnZt
NmzZRokQJZs6cySeffMLevXtZvHgxI0aMaMIDIy0uiIIlanX79+fP3114SGhhIaGoqdnZ1F/5QpUwxKJiJ/lZycz
PLly/nqq69U4S6cmJw4cPkz9/fl5++WVKlCjBe++9WVKlCjBe++9x5kzZyhWrJh5Wa+IPDu1a9d+aJ/JZGL9+vXP
MI2hdM6TFo1y/k4ZFfIpqYdvBQOmB+bqJl+YwT8EHpmNBWBBgAQGxsLG5ubganE7FOGzZqCiIiIiIiIZAM6j
MPKjBgxgxgsGDB1OgQgAEqVqxI5cqVAVi9ejVlypQx8OJ2IiIiIiIiIDwuLd21lQhcuXCAmJoZSpUpE5SpUphY30/1rtjxw7
c3Nx44YUXDE4nYj3CwsLSSsLSNW7JkiVPMjDmIiIiIhkB1q6a4V8fX3x9x9fW1aKtQoYJBaSU7u7u7u9ERRERERERJ
BvRjD4REREREREREREZFsQHv0iYiIiIiIiIiIiZAMq9ImIiIiIiIiIiGQRZAMq9ImIiIiIiIiIiGQRrB7926jY4iIiIiIiIiIiIiIyFO

```
iQp+VOHv2LIOaNSJfvnz07NmTX3/9lcTERKNjiYiIiIiIiIhIJtFhHFYkJSWFLVu2sHz5cpYtWOZMTAwvvvfQSz
Zs351//+hdeXl5GRxQRERERERERkcekQp8Vi4yMNBf9du/eTYUKFWjWrBlt2rQhb968RscTERERERERERZEMUKF
PALhO6RI//fQTP/3OE9WrV2fw4MFGRxIRERERERERkQxQoU9ERERERERERCQbOGEcIiIiIiIiIiIi2YAKfSIiI
iIiIiIiItmACnOiIiIiIiIiIiLZgAp9VubWrVtGRxARERERERERkadAhT4r4+Pjw+uvv87mzZuNjiIiIiIiIiI
iIplIhT4rM3/+fK5evUqdOnUoWrQoEyZM4Pz58ObHEhERERERERGRJ2RKTU1NNTqEPHuXL1i3rx5zJkzh8jIS
BoOaMDrr79Os2bNyJEjh9Hx9xRERERERERERQkg1ToEz755BOGDBlCYmIiuXPnpkePHgwbnwNgwnJyejo4mIiIiIi
ISDqp0GelLl68yNy5c5kzZw6nT5+mZZ3WdOnShbNnz/LBBx/g7+/PBBBxo7+/P6tWrjY4pIiIiIiIiLppEKflVmyZAmzZ
89m1apVBAcHO07VrV9q3b4+Hh4d5zPHjxyevDiJiYnGBRURERERERERRkQzRZmxWpnPzrrRp04Yt7bw4osvPnC
Mv78///nPf55xMhERERERERReRKa0WdF7t27xxdffEGrVq3w8fEx0o6IiIiIiIiGQiFfqsjJ0TE5GRkQQGB
hodRURERERERERERMpGN0QHk2apQoQJ79+41OoaIiIiIiIiGQy7dFnZXr16sWgQYM4e/Ys5cqVw9nZ2aI/NDT
UoGQiIiIiIiIvIktHTXytjYpJ3EaTKZSE1NxWQykZycbEAqERERERERERF5UprRZ2VOnjxpdAQRERERERERE
XkKNKNPRERERERERERQkG9CMPit16NAhoqOjSUxMtGhv1qyZGQYlEREREREERORJqNBnZU6cOEHL5li05cOEHL1i05cOCAeW8
+uL9PH6A9+kRERERERERERsqi0JzNIttavXz8KFixIbGwsTk50HDx4kN9++++43y5cuzceNGo+OJiIiIiIiMhj0
h59ViZ37tysX7+e0NBQ3N3d2bF/WKFWP9+vUMGjSIvXv3Gh1RREREREREREQeg2bOWZnk5GRcXV2B+OW/8W+OW/8+f
PAxAYGMiRI0eMjC7BggWqSIMHEiEydOxN7eni+++IkoCCj44mIiIiIiIiIiIk9Ae/RZmZIlSxIeHk7BgwWYjdoyOKiIiIiIiIiMhjUKFPuHr1Kp6enuaT
ERERERERREJOtRoU9ERERERERERCQb0B59ViAsLCzdY5WcsWfIUk4iIiIiIiIiIyNOiU3etgLu7/nm5ubGunXr2
LVrl71/9+7drFu3Dnd3dwNToioiIiIiIjIk9DSXSszd0hQrl69yowZM7C1tQUgOTmZXr164ebmxs
RERERERERERReRwq9FmZPHnysHnzZooWLWrRfuTIESUaPUqcKVK1cMSiYiIiIiIiIiIk9CS3etzL179zh8+HCa9SHD
5OSkmJAIhEREREREERyQw6jMPKd07cmS5dunD8+HEqVKgwaw5kwYQbPt25kwYQKd03c20J2IiIiIiIiDwuLd21Mik
pKUyaNImPNPvqImJgYAPz8/OjXrx+DBg1yREENHkJQERERERERESelAp9I
iIiIiIiIiIi2YAO47AyFy9e5LXXXsPf3x87OzvsbW0tLiIiIiIiIiIiIkFKQ/JZARVLFS4t2trcRMRERERERERkaxJh3FYmU6dHEdHc3w4cPx8
5OSkmJAIhEREREREERyQw6jMPKd07cmS5dunD8+HEqVKgwaw5kwYQKd03c2OJ2IiIiIiIiIiDwuLd21Mik
pKUyaNImPNPvqImJgYAPz8/OjXrx+DBg0y79snIiIiIiIiJZiwp9VuzGjRsAuLm5GZxERERERERESelAp9I
iIiIiIiIiIi2YAO47AyFy9e5LXXXsPf358cOXJga2trcRMRERERERERkaxJh3FYmU6dHEdHc3w4cPx8/PDZDI
ZHUlERERERERERDKBlu5aGVdXV37//XdKly5tdBQRERERERERElEWrprZQICAlBtVOREREREREQk+1Ghz8pMm
zaNYcOGcerUqerUKaOjiIiIiIiIiIhIJtLSXSvj6el5dwSjgeeKQkIC9+7dw8jdw8nJCTs7O4sxV69eNSid
iIiIiIiIiIiIi
IiIiiIiIiIiIiIiDwFmtEnIiIiIiIiIiKSDWhGnxWYNm0aPXr0wMHBgWnTpj1ybN++fZ9RKhERERERERERyUya0WcF
ChYsyK5du8iVKxcFCxZ86DiTycSJEyeeYTIREREREREREcksKvSJiIiIiIiIiIhkAzqMwwqkpKQQFRVF3bp1KVmy
JKGhoTRt2pRvv/0W1XlFRERERERERLIHzejL5lJTU2nSpAm//PILJUuWpKgRYuSmppKZGQk+/fvp1mzZixfvtz
omCIiIiIiIiIi8oS0R182FxUVxaZNm1i3bh21atWy6Fu/fj3NmzdnxowZdOzY0aCEIiIiIiIiIiKSGTSjL5urX78+tWvX
ZtiwYQ/sHzduHBs3bmT16tXPOJmIiIiIiIiIiGQm7dGXze3bt48GDRo8tL9hw4bEx8c/w0QiIiIiIiIiIvI0qNCXzV29
ehUvL6+H9nt5eXHt2rVnmEhERERERERERJ4GFfqyueTkZHLkePhWjLa2tty7d+8ZJhIRERERERERkadBh3Fkc6mpqXTq
1Al7e/sH9ickJDzjRCIiIiIiIiIi8jSo0JfNhYWF/eUYnbgrIiIiIiIiIpL16dRdERERERERERGRbEB79ImIiIiIiIiIiGQD
KKhPIiIiIiIiIiIiIiIiIiLZgAp9VubWrVtGRxARERERERERkadAhT4r4+Pjw+uvv87mzZuNjiIiIiIiIiIiIiIiIiIiIiZEB6HCn0ihMS9a
2X/qvzWelw/qvs6ZsLuSEk/ssLiH3U3bGFdLJ+dYWYG1/h6ZKL9G2zSIuuI4q3BhSMvW72oiaCwdRVaiTVCoOJ1clVQcW
5SRx/KdU1gr8Cn8w/SyNAS9X8T5BEW17R11VDXUR//WVq7SmxqTsgFC3LpzLmxz6uZfE1UiOJ+K5AsV41VWkNEIGXoC
2h6PsW6rZnaj19aSM4JWlUOGdxT7EijSrCivM9gQQEmyKZKOhqgA0A/zHeoNP0CMWEROOFdX8DRoGEcNJ3cn8AOV/g7lyg
IiIiIjZMxlPSyv/3ad2wC4KLCIiIiIiIiIiIiOGQ0eQPSEvjEWnQCA7UcvyLRrSvTiMBaw7qvH28xkNq9njcGxWDfRVpROAD
GZEY6WSNXqUPt/yCgY3gcTQaEqWM+NAUyM98J3ftbhPxJ17erOpXRi8L5bo/7xk49ARpnKtVo5Ilw7k9FpHpSG9
```

```
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "Most people have a secondary/secondary special education (66.6%)"
      ],
      "metadata": {
        "id": "_ZkX5B3CDYXW"
      }
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "492sURnty3NO"
      },
      "source": [
        "#### Housing_type"
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "analyse_categorical_column(\"Housing_type\",90)"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 506
        },
        "id": "SksW3yJe4Ln1",
        "outputId": "fccb3dba-2023-42c1-9db2-0911cdb910ee"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 1400x500 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAABRkAAAHpCAYAAAAGWzuMAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAAD8dEl
EQVR4nOzdeZzN1R/H8de9d/Z9MWZhGNlliZQQ2YpIKYWIlGhDWumnhIpUlKUoZStCkVJRlpCdse/b2Gcx+z535
t77+2Nyc7NkGS7j/Xw87q05330+5/v5fo3M/cznnGOw2Ww2RERERERERERERERC6T0dkBiIiIiIiIiIyI1NSUY
RERERERERG5IkoyioiIiIiIiIiIyBVRklFERERERERESuiJKMIiIiIiIiIickWUZBQRERERERERERERZEro
iSjiIiIiIiIiIiIXBEXZwdwtVitVk6ePImvry8Gg8HZ4YiIiIgUGzabjfT0dEJDQzEa9Tfv5cpsNnPy5EmTE
```

```
SGRnp7DBEREREiq1jx45RunRpZ4chIiIiIteBYptk9PX1BQp/+PXz83NyNCIiIiLFR3p6OpGRkfaftORERERERi
m2S8fQUaT8/PyUZRURERK4CLUkjIiIiIqdpERORERERERERERERG5IkoyioiIiIiIiIiIyBVRklFERERERERERES
uSLFdk1FEREQKWa1WzGazs8OQG4ybmxtGo34fLSIiIiIX55KTjCtWrOCjjz4iOjqa2NhYfvzxR9q1a3fOvs899
xxffPEFn3zyCf369bMfTO5Opk+fPsyfPx+jOUj79uOZPXoOPj4+9j7btm3jxRdfZMOGDYSEhNCnTx/eeONS75
BERGRm5nZbCYmJgar1ersUOQGYzQaKVeuHG5ubs4ORURERERuAJecZMzKyqJWrVo8/fTTPPLII+ft9+OPP7J27
VoiIiLOauvSpQuxsbEsWrSI/Px8nnrqKXr16sWMGTMASE9P57777qNFixZMmDCB7du38/TTxMQEECvXr0uNWQ
REZGbks1mIzY2FpPJRGRkpKrS5KJZrVZOnjxJbGwsZcqUOS7SIiIiIvKfLjnJeP/993P//fdfsM+JEyfo06cPv
//+O23atHFo2717rh3HLmTDhg3UrVsXgLFjx9K6K1bGDVql1bGDVqlJK
MIiIiF6mgoIDs7GwiIiLw8vJydjhygwkJCeHkyZMUFBTg6urq7HBERERE5DpX5CUNVquVrl278vrr3Prrbee1
b5mzRoCAgLsCUaAFi1aYDQaWbdunb1P48aNHabntGzZkr1795KSknL06+bl5ZGenu7wEhEhERuZlZLBYATXeVy3L
6++b095GIiIiIyIUUeZJxxIgRuLi4OLdv3302x8XFUbJkSYdjLi4uBAUFERcxTGhrqOOf0+9N9/m348OH4+
/vbX5GRkVd6KyIiIsWCprrK5dD3jYiIihciiJNMkZHRzN69GimTJlyzX8wffPNN0lSu7O/jh07dk2vLyIiIiI
iIiIicrMq0iTjX3/9RUJCAmXKXFxxeOHDnCq6++SlRUFABhYWEkJCQ4nFdQUEBycnJ4YWH2PvHx8Q59Cr9u
r8/3eff3N3d8fPzc3iJiIiI3CiaNGlCv379nB2GiIiIMhlueSNXy6ka9eutGjRwuFYy5Yt6dq1K0899RQA9ev
XJzU1lejoaG6//vY+AwcOJD8//377Q+KJFi6hcuTKBgYYFFGbKIiMhNZ+7e2Gt6vUcqh1/We
WvWrOHuu++mVatW/PrrrOUclYiIiIiIFKVLTjJmZmZy4MAB+/uYmBi2bNlCUFAQZcqUIAg42KG/q6srYVK5
cGYCqVavSqlUrevbsyQYE8jPz6d379506tTJnuXs3EM0PtSWJSWLeLQoUNO7NgRgEqvVqS
4ezQ7ihVK9e3dkhiIjIde7rr7+mT58+fP3edL+c8LVYjabtUm0iIiIiMhluuTp0hs3bqR27drUrUrl0bgFd
eeYXatWszaNCgix5j+vTpVKlShebNm9O6dWsaNmzIxx9/bG/39/fnjzz+IiYmhttvv51/HzBg1u4nXPaAgm3nh5k8w
CIiInIDyszMZNasWTzTz//7+rwaxhP2EE5OZyWvw/2RFfXURYisFttC8Phk4MDAwz2DOoNAWhRY5W2Y4POGfgG+ihWrIi
HhwctW7ZOWLN58ODB3HbbbXz11lVeUK1cODw8PAI4ePcpDDz2DGd9ev64+zAYDOzZs8ch3k8++++ZHfCHzvtAY
YTy5cvb3+/YsPfXHx8fQkND6dq1K4mJieHtW2akEJPqHtrMVDa2a+3D9rpkvYrVB8z82bVPhpBmNvdVX/x6+
Zznqd+cP/mQ4fPnzW+kJBQUHMmDGDjIwXb9ata+UjE6sd9pG3VPkR5OzAJPEInY7+eiBIwCdTrfjFKvggHVAmwq9vNu4RNgZP5vbEh5vfIq/XUqmtqbJyH3
/2jFvTy7SqXWMtZyhl6eAtXlOyXXLhtGzN5FxQ4/xxLxCeT0z4Vvw/xUqyTDXw8GDL6/ePlHGpDyBwNqk8KjLbM+
+T2Zt7/NfYz/cXsArMIs8jVkYyxKEZGtXZuXrUAemcQYXlHKo+I1LV1sA4KtXgb2B9HDQo9pFRaC4t2r2Btd
VI+z3lvI7KRwNHbDpB4o8vkUTnUl6JeeY7nN1T0Bi3xGQzzRrQ1oVoHnwddf8hVN3BLVDBF+8G5o8Lxew+bxlbn
8RYNmtn/6qPRV9Pt4r0WZv7vsp4KvSfvs27HqoAPxwv+OctYdTfqk+zf36RHNvoAuMp+K3vdeCoJtjGtlPEj
Hv9hZc0gwvnepCf1fbZY3u8X6gXtnHsivC9hMbjirg6V9vtaRDT/9/QnUJMnKbvHjQHwexiHR4yQFETK1Zq5z1Z
XtfjPbRm4HSa+FfeUZmBXZ7ZGmwGYYs4RVuaGbW2edwahZWe3c3Ebd02i41upe8MvREHHo8pMRLcHeIbSu/5
hBYRpVQ8rSvkp1u9kGmfZ7C9+U4cWIMLS9xYqCjPr4KoN91YtEhrlYXDGHBWsDy5BbcFGthXQEKSZ9H0A7YFRsW8
31qn1ORjD9rRCpeu0x8/wKbhjBDfXeo6GKeNfWrPeNNSZ/pMOSCmGD39nrFFP2ujf//f8Bm3YmxbDz6OOQJJd6V0
WB1RwECAeUAzT8K83YP+Bc9jbzlncaLyevtZgfLQxHtcyXsNjJtJkGaj4XZkzDJQ71IsSuNedW9PjxBHF0G2rqR
hXUx1Ixl6mkLaM5fDX4x3qWuVOMPefqmP8XX2g+zh1TEnjhHl2yDVH90yh2M1YFJ7GBvGvI6w9F1Cfk4MqqtRqG5Y
```

g8//DA228VPFVMSTUREerqVTGXn8tf8Uf+1PZOWBRE5l5Dk7pCKXV2Bl2/E0th1P47v1xwAoH+LN3RVK0LBCCe4
qH4yfh6uToxQRZ1CSUURERK47Hh4ejBgxgmeffZbAwEBnh3PVmM1m3NzcnB2GiIhcptx8CxsOJ/PX/kRW7DvF3
vgMLuF3YcXGwVNZHDyVxdQ1RzAZDdQo5U/DCsE0rFCC28sGam1HkZuE0dkBiIiIiPxbixYtCAsLY/jw4RfsN2f
OHG699Vbc3d2Jiopi5MiR9rYmTZpw5MgRXn75ZQwGAwbD+dePGjVqFDVq1MDb25vIyEheeOEFMjMz7e1TpkwhI
CCAefPmUbFiRTw8PGjZsiXHjh2z9xk8eDC33XXYbX3zxBZGRkXh5edGhQwfS0tLsfbp37067du0Od+fTuDB
lygBs376dZs2a4enpSXBwML169Vf/48//sDDw4PU1FSHmF966SWaNWmf79y5UoaNWqEp6cnkZGR9O3bl6ysL
Ht7QkICbdu2xdPTk3LlyjF9+vQLPlsRETm32LQcvvvrEF2/XsdtQ/+g69fr+XLFIfbE3ZwJxn+zWG1sOZbK38
epPPEddQa8gdPTlrPD9HHycorcHZ4InIVKckoIiIi1x2TycSwYcMYO3Ysx48fP2ef60hoOnToQKdOndi+fTuDB
w/m7bffZsqUKQDMnTuX0qVLM3ToUGJjY4mNjT3v9v2fatGm
sWrWK1NRUOnXq5NDnwIEDzJ49m/nz57Nw4UI2b97MCy+84NBnyZIl7N27l0WLFvHLL7+QlZVFy5YtCQwMZMOGD
Xz//fcsXryY3r17A9C8eXMCAgKYM2eOqULAAcPHqRVq1a0b9+ebdu2MWvWLFauXGkfAwoTnMe
OHePPP//khx9+4PPPPychIeE//4PPPPychIeE//iRERAQgx2zhx83HeeKrdTT8YCnv/bqbv/Ynkpt/Y6yv6Ey5+VaW7zvFa99vp
e57i+nz3WaW7omn4AZZm1JELp6mS4uIiMh16eGHH+a2227jnXfe4euvvvz6rfdSoUTRv3py3334bgEqVKrFr1y4
++ugjunfvTlBQECaTCV9fX8LCwi54rTPXbIyKiuK9997jueee4/PPP7cfz8/P55NPPuH999/uu1ZOnnQxn
8+dd94JQG5uLtOmTaNUqVIAjB07ljZt2jBy5Eh7DN7e3nz1Vf2adITJO6On+ft7Q3AuHHjaNu2LSNGjCA0NJR
OnToxY8YMevToARQmKlNTU2nfvj0Aw4cPpp0uXLvb7qFixImPGjOGee+5h/PjxHDl6lAULFrB+/XruuOMAL7++
muqVq16kX8hIgI3H5vNxrqYXOZEH2fbBjjgyVYV3xXLyLczfepL5WO9SwseNB2pG0K6K52aKW6LDHB2aCJSBJRkFBE
RkevWiBEjaNasGa+99+99tpZbbt37+ahhhhx5yONGrU6Pr6HfihHg8zOWEB0oLPGHqHGbDhMEDc3FGqxFMgC5ZdX0pdvFMF
ehP+Yz23l76+2fg/EDkzNHoHnwfw+Lf/qxfvpHQVPi3RTEGaJBKsXbhoyw4+VH
x0uTEnsJ3RlBKVPtfWpRR9DPAIxXyPLlEZWbfcz5NHUJYfxp0JtWKBVG/JtBTiPtPuDkPuoaRt0zqrRv5jSMtjzFHRyx0qhzGIbvdYdWppCa/PW9Hv+j/Hc8Tv0tbZCVe9V7lK6mqgSXLeuqVGEwZdoDyOsmKhSeQg5DjVBl0TfmbtIh0VnGzBiNyzgrw
cqxXvuZVM0JERI7Gi0MHNWdZjDv5s9lrh0bJP/MzhWDfwiFYMpdaS9IJQlDPgXJbgkP8DUk+
ZXJN3wOKwYK9NHNM3K/UuPC/71qYRfd37EuhxHQ4IrZd3p56hsUq4fJ2sF+RFwGPUvy6lqjEaW0DE1k7B4IHABF9upvS/
zqXnXwvnHDB06ZkHwk0CfUyC9Q+UrP5ZP3SzCh1sM6y4dePFnhJycE0Y8YucZSGVNJSRKGbtWjdt/7y8xgNkkWL92
z8H7TCBt3IeWVFzvAeVWFpjVgbRd7aNcXFZnjibSzqcGymp6tyqX3bd4gSEdh1kMTI0iZfUSi7+0X2dbOc2VjbzK4
Vh8IfREFBcl/wm0tCBoJLLDU3LdUnnVKG2r1dfNdrTGmq+kMWM5tLtHN/wnl+ZLrhKhgnhUQRIsnf2Xs3Y0V/
aBmU/FYn2hF0iEnHnhvgFL5fQBnNcFUijVSJDCEXl4aFJd3UfAP42b29NncR9Fr/TGXCjNXL47Eoc4KDJqPt
tp8gIzjHZV25TfPnr8lB8Bhng4Oq/92e6CgHdZQJOdqTqU0aMoi5wtJPuWeJC8kzpd/MWxxakg/2xO92yDTvC0Hh4qpG0Lq
cdyHv3P9Pd3T2ws72PbWZlDPFThP0Ktoh3IFoYAVSAhmvwmRvUVpTjmEAEZLbG1cdnrHtSbQd0PqKXuMtI40TV3fLw8PXcH

```
s/dsmF69OmQA+6XgbZYO9nR2KSLF11X81nZaWhsFgICAgAIA1a9YQEBBgTzACtGjRAqPRyLp163j44YdZs2YNj
Rs3dvihtmXLlowYMYKUlBQCAwPPus7w4cMZMmTI1b4dERGRG4qbmxsVK1bUlGm5ZG5ubqp+FZHrisVqY/CcDUy
PTsCKKtLk0mw6mkrr0X/x1gPVePzOMs4OR6RYuqpJxtzcXPr378/jjz9u/y13XFwcJUuWdAzCxYWgoCDi4uLsf
cqVK+fQJzQ01N52riTjm2++ySuvvGJ/f7qSUURE5GZnNBrx8NAOiyIicuM6EJdGz8mriUmzghKMcpmyzBbenLu
dJbvj+aB9TUr4uDs7JJFi5ar9ejo/P58OHTpgs9kYP3781bqMnbu7035+fg4vERERERubF9tXQXrUev+DvBK
HLlFu9OoOUnK/hjZ5yzQxEpVq5KkvF0gvHIkSMsWrTIIeEXFhZGQkKCQ/+CggKSk5MJCwuz94mPj3foc/r96T4
iIiIiIiJSfKVm5dFp3BLe+yMGs03LN0jRSsoy0+ubaN6cuw2ztiYXKRJF/n/q0wnG/fv3s3jxYoKDgx3a69evT
2pqKtHR0fZjS5cuxWq1Uq9ePPXufFStWkJ+fb++zaNEiKleufM6p0iIiIiIlJ8rN8fyz0fLGLt8VxnhyLF3Hf
rj9F54lqSMvP+u7OIXNAlJxkzMzPZsmULW7ZsASAmJoYtW7Zw9OhR8vPzefTRR9m4cSPTp0/HYrEQbycf
cH5qlWr0qpVK3r27Mn69etZtWoVvXv3plOnTkRERADQuXNn3Nzzc6NGjBzt37mTWrFmMHj3aYc1FERERERKV5
sNhuTF2+hy6SNpOVr7UW5NjYeSeHBcavYE5fu7FBEbmgGm81mu5QTl86/uSTOmTc86/uSTTzJ48OCzNmw57c8//
6RJkyYAJCcn07t3b+bPn4/RaKR9+/aMGHx8fe/9t27bx4osvsmHDBkqXLk2fPn3o37+/wz1nz57NwIEDOUV
pl2R9xrl7Y6/6NYqTRyqHOzsEEERERuUzX+ucsEbk55Obl8ea0Zcw7kI/NoASjXHvebiZGd6pNi2qhzg5F5IZ0y
UnGG4WSjNc3JRlFRERuXEoyikhRiz2VRK8v/2R7hqezQ5GbnNEA/VtV4tjs7FJEbjlbPFRERERERafZvv8
w7ccsVYJRrgtWGwxfsIfXvt+qDWFELpGSjCIiIiIuIUC9dso8ukaE7mezk7FBEHP0Qfp8tXa0nOMjs7FJEbh
pKMIiIiIick1ZrVa+/Pkv+s47SLrNw9nhhJzThsMpDPhuJQdPZTo7FBEgpKMIiIiIics2YzfkMmbqAD1a
lYDa4OTsckQs6npJDpy/xciTQkz88fsNGDv7Br9VlbKoNTPJZNIeRM/IGCLciojj8cnruVAQoazQ
xG5rinJKCIiIiIlfdyfhTDJgwlzlH3bAZ9FFUbiynMvLo9OU6JRpFLkD/ZxcREREREZGr6tCR4wz88kcWJgZ
gUwWj3KASMwsTjfvjlWgUORclGUVERERERERERERERO8SqiTl6gqFTfmF5ZpimSMsNLzGzcOr0PiUaRc6iJKOIiIhcF
YePneTdyfNZkR2hBKMUG4mmZZp5/Mu17I1TRD0OSRA8X6S96bMZ1l2uBKMUuwkZpp5/Mu17I1TolHkNCU
ZRUREREREpEgdPRHLu5Hs5N/5s/MMKwGF2eHI3JVJGUVVjQeOpXp7FBErgtKMoqIiIiIiEiROXYyjg+mzuPPPzHAsS
jBKMZeUZeapKRtIysxzdigiTqcko4iIiIiIiBSJ47HxfDfDptLktSSyrBKDeNIOnZPDNtI7n5FmeHIuJUSjKKiIi
IiIjIFTsRl8C4b96g96RA8owezg5H5JrafDSVl2dtwWazOTsUESuSEJSMhO++Z4/4r3JNPk5O
xwRp1iwI47hC/Y4OwwRp1GSUURERERERC5bZlY2U2b/zNKTBpJdQ5wdjohTfbniEO0ijzs7DBGnUJJRRRERER
ELkt+fgEz5i3gj90JnHAv6+xwRK4Lb/64nbpySjiIiIiIiIXDKbzcZPfzpZPf/zJr+t2ctirqrPDEblum
AusPPtNNPHpuc4OReSaUpJRRRERERERELtnytdH8i2rUKCPliIOEjLy6Udp+Umo38JRERERERE5JLs3n+Q6T+uYcupE6wFhsIxJRE6hT15/iTh7+xwRK5rU9ccZsW+U84OQ
+SaUJJRRRERERERELkp0bi7fzPmFHceTOeYW5exwRK57Nhu89v1WUrLMzg5F5KpTklFERERERETK9VqZc6vS1i
7ZSfH/GtiweDskERuCAkZefzvx+bS0OkqlOSUURERERERCP7T+i07WLRyDenB1UixuDs7HJEbyoIdcczbdNzZ
YhcVUoyioiIiIiIIyAXFJybxw6+LyDT5sy8/2NnhiNyQhs7fxbHHkbGeHIXLVKMkoIiIiIi52WxWPjhlOUcjj3
FfveK2DRNWuSyZ0YV8PKsLVisNmeHInJVKMkoIiIiIi57Vi3SbWbNpGaomapFtcnR20yA1t45EUPv/zgLPDE
LkqlGQUERERERGRczoeG8+PKAb0/s8MRKRZGL9nP1m1PmpzgSiIiIiJnMZvzmTTX/d2KTM9h
jKufscESKjQKrjf/9uB2rpk1LMaMko4iIiiiIiiJxl8cq1bNy2iIKXXPn/OF6ZhaHXKOOm7TIVZK
IiIiIMDh48yf9FyXP1C2J/n5+xwRIqlj3/fS3pvrPDEC  kySjKKiIiIiIiIiIXXZOLrPn/OF6ZhaHXKOOm7TIVZK
UZWb04v3ODkOkyFxyknHFihWObduWiIgIDAYD8+bNc2i32WwMGjSI8PBwPD09adGiBfv3O/6lSU5OpkuXj5+
REQEECPHj3IzMxO6LNt2zYaNWqEh4cHkZGRfPjhh5d+dyIiIiInJJ/ly9nu179uMeXpHjeR7ODkekWJu25jA
HEjL/u6PIDeCSk4xZWVnUqlWLWLzz777ZJrb29Pt/IWPGjChAmsW7cOb29vWLVqRW5urr1Ply5d2L7dFNkzE5fbly3aWdMNzV4GjrsWz5Wvrj
35hxYoV9OrVy96enp7OfffdR9myZYm6SCjIiIiInIx4hOT+H35avz8/NiUHeTscES
KvXyLjXd/2eXsMESKhMulnnD//fdz//33n7PNZrMx6fdeD7Oj7/LhIg8FVFa6dbeG32pmZFLsjd/MQkzPs7qnwOKsiE/FxcpoC2lGQFpHkUV6/k6p5PnaG1Px0UcOB7fpcdDvx4h7wJ4kljBCO1V
A0bNlC3bl0Axo4dS+vWfn444+JiIhg+vTpmM1mvJyl5Vs1ab8BY2yPeWR5SonEwheyQLDa2vqwI7uy4/6Qe6n8ja8SUvfMdRCIwsT6DPNkVUGmtHVWVc3RK7ITmHgq6yKBGJKnoJknvu3NXOa36ajaH3Pn1s8bfpuz7RFvXGm4ZvL9u20D3O98JGhYzTxgeeDBiSWGyydMyR27+qzuuzDVNXzRB
mITEFQ+kapKa7OjskkZvC8n2nWLonnmZVQp0disgVKdI1GWNiYoiLi6NFixb2fgICAqhXrx6r1q3Ny51RV8MekKT3/IfwoxR/DQbqo9+5ddXPXI7Ng9h3j7w/E0kEBsZ6R6Z8zv1KJQZPdRrMN81pB
uwJRoAWLVpgNBpZt26dvU/jxo1xc3Oz92nVqhV7917ll5fl+84u4pKuGH1KUZ7OxKVrMNtdsFE8Ax3Bmez/UM3D7kxgDGBC2Hzx7xqf0g06YHkFxnRiQnffbaHtH1/eTG6ZXHUmrLIoYH/TgUj9RXhZc5xV8Hm6SvKtO7k6fL9WtXsj1WvOhSpiFZV9Lh7x4jh+wXSZSHg1dUSRCUsb1F8HBpUNzN63jjQMhbLx4dX+XvJhW+9aGI5nS4
sztRmLyLX0nu/7CbfYnV2GCJXpEiTjHFxcQCEhjpm30NDQ+1tcXFxhFRydwdL+zqqo5ORK2h0BY30XDfgwf8/pEMgqYFDh2vjZPQ7uFFhHOHQlUVhm7z+l9pODBrGwz84OVMLTAMrm8kcmfSLuH/RiBEKlJSpE99MBhG9YP2+pNa2pm
ff3t78iIyOv/IZERERERuAhaLhV+WrCAr05cjpghyrSZnhyRyUzmUmMXkVTHODkPkihSb3aXffPNN0tLS7K9jx445OyQREREREZEbQvS2Xzv3aSV2wDzgZb1HWpCS+KFpjz9zxyRm9LYJQdIzTY7OwyRy1akScawsDAA4uPjHY7Hx8fb28LCwkhISHBoLygoIDk52aHPua5x5j3+zd3dHT8/P4eXiIiIiIiIiMjFmc35/LpkFQGBQUyIc90ZkRQIQNtB6SURERERERELiw7J5dflqzAZDKyM78kVgzODknppSRV8CU1Ye
dHYbIZSvSJGO5cuUICwwjyZIl9mPp6emsW7eO+vXrA1C/fn1SU101Jjo6291m6dClWq5V69erZ+6x6xYsYL8/Hx7n
OWLFlG5cmUCAwOLMmQREREREZGb2op10eyPOUpAaCSHcz2CHcz2cHY7ITW3q6sPkmC3ODkPkslxykjEzM5tW7awZcsWoHCzly1btnD06FEMBgN9+/blnXfe4aeffuKvv/6iW7duRERE0LZtWwCqVq1Kq1at6NmzJ+vXr2fVqlX07t2bTp06ERERAUDnzp1xc3OjR48e7Ny5k1mzZjF69Gj69etXZDcuIiIiIiIicrM7mZDAbyvWEhgcwqmEBA1kk/XFcXnPT8ZeiWaZhwk/M5SXexQWHz/zM6t0a21x69ViYjp6aeYvzOFN5DRoRL+4Of0Fw6JTI+dEw3Uy7ktvYP1y+8yVWA7s1bIFO55KBvSfGWPN37Gq5KuM/jbwm2iNtcWnwiOTdrjemY8/03ArEe1X+vAEB+0uOQFnLH4X1eF4z+++DNyeIyUtd3FGSdyvSZwwBVAzgpC
```

```
VMnIiIiAOjcuTNubm7O6NGDnTt3MmvWLEaPHsOrr7xSZDcuIiIiIiJyszuVlMLvy1fj5+vDnrwAbKpiFHGqlOx
8Zm446uwwRC6Ly6WesHHjRpo2bWp/fzrx9+STTzJlyhTeeOMNsrKy6NWrF6mpqdx999OsXLgQD49/fiM2ffp0e
vfuTfPmzTEajbRv354xY8bY2/39/fnjjz948cUXuf322ylRogSDBg2iV69eV3KvIiIiIiIicoZFf6OhNu4U5Sp
UZPEpL2eHIyLAV3/FOPWusriYisO2GnKTMNhsNpuzg7ga0tPT8ff3Jy0t7Zqszzzzzzzzh3b+xVvOZx8kjlcGeHICIiI
pfpWv+cJSJXR9ypJIZ+MgEXFxeOuZVlc6avsOMSkb+NfKwW7W8v7ewwRC6JOuIiIiIiIiI3oTXRWOlOTSMwKIh
d2apiFLmefLHiIMWOJkyKMSUZRURERERERbjJpGZmsWBdNYIA/+3K8yLOanB2SiJxhX3wmS3YnODsMkUuiJKOIi
IiIyA2ue/fu9oOWz7Rs2TIMBgOpqanXPKZLlZOTg7e3NwcOHHB2KGe5Hp+jwWBg3rx5l33++iO7iEtIpGSJYHZ
k+hRdYCJSZMYvP+jsEEQuiZKMIiIiIiLidIsWLaJs2bJUqFDB2aE4yM/Pd3YIRS43L48/V63Hy8uTY/neZFgue
T9QEbkGoo+ksD4m2dlhiFwOJRlFREREERG4ic+bM4dZbb8Xd3Z2oqChGjhzpOH6uCrmAgACmTJkCgNlspnfv3oS
Hh+Ph4UHZsmUZPny4vW9qairPPPMISEh+Pn5OaxZM7Zu3fqfcf300088+OCD523v378/lSpVwsvLi1tuuYW33
37bIQE4ePBgrbvrNr744gsiIyPx8vKiQ4cOpKWl2fts2LCBe++9lxIlSuDv788988999zDpk2bzrr/8ePH8+CDD+L
t7U3Pnj1p2rQpAIGBgRgMBrp37w5BjEN7w5AkyZN6NOnD/369SMwMJDQ0FAmTpxIVlYWTz31FL6+vlSoUIEFCxY4XGPHj
h3cf//9+Pj4EBoaSteuXUlMTLS3N2N2nShL59+/LGG28QFBREWFgYwcPtrdHRUUB8PPDDD2MwGOzvL9am7bs5ciK
WUqEl2aYqRpHr2pcrVMOoNw4lGUVEREREbhLROdFO6NCBTpO6sX37dgYPHszbbb79tTyBejFjxzzjzz8ze/Zs9
u7dy/Tp0x2SXI899hjxvDzzz8ze/Zs9
u7dy/TpOx2SXI899hgJCQksWLCA6Oho6tSpQ/PmzUlOlOPn81jtVq5ZdffuGhhx46bx9fX1+mTJnCrl27GD16NBM
nTuSTTz5x6HPgwAFmz57N/PnzWbhwIZs3b+aFF16wndujUZGRkO4wwePJiHH
36Y7du3M2TIEObMmQPA3r17iY2N2NZfTo0fa+U6dOpUSJEqxfv54+ffrw/PPP89hjj9GgQQQM2bdrEfffdR9euXcn
OzgYKk7DNmjWjdu3abNy4kYULFxIfHO+HDh0cYpg6dSre3t6sW7eODz/8kKFDh7JoOSKgMFkKMHnyZGJjY+3vL
OZBQQFLV2/AxcWFJJs3ifluF32uiFx7f+49RUJ6rrPDELkoqosXERERESkGfvnlF3x8HKvTvSLBaLw/tRo0bRvHl
z3n77bQAqVarErl27+Oijj+zVjBIjk0hJCTg7u4OwMcff8y8efP44Yq
Ycf6NWr1znHXLt2LQD16tU773Xfeust/XeuDwUDZsmXtbStXrmT9+vTmK9+vUkJCTg7u4OwMcff8y8efP44Yq
Ycf6NWr1znHXLt2LQD16tU773Xfeust+9dRUVG89tprzJw5kzfeeMN+PDc3l3/tPM+PDc3l2nTplGqVCkAxo4dS5s2bRg5ciR
hYWEOa9bMYcwvv/ySgIAA/3rlzZ5566in7+5iYGABKlixJQECAwxi1atVirx9fuXKknUCSnvk8w6TyDe8VAwqVIibEmi1ttd
Xr27AnAoEGDGD9+PNu2beOuu+5i3Lhx1K5TF6BQVx4QkSqUZRNBbYK4tExQf3tvvR7vR9bG8Rmr5vR2y4YxS8w3pkthwcW3
yZAn33nsvISEhQGGFaVhY2Hmf2bns2HuQfYcOUzo8jA3aUVrkumex2vhhO3FeaHJ9LSUhci6qZBQRERERKQaaN
m3Kli1bHF5fffWVQ5/du3fTsGFDh2MNGzZk//79VwtAA7dvW8+cKQbS2LyFypUrmUbZlRiDVgFV8wcRzbtGWZGDcAakCRR5PUZ91BLdvfbp5BahSSwmPxGnETXlbgMfkBqZe/QpM1wiDc8AP3Z2oq3PSRXOXhZ/M4Xd1Mjcw==
```

jiIiIiIhIMWSz2di4bRce7m4YDAZilWQUKRZWHkh0dggi56Qko4iIiIiISDEUm5BIzNHjlAgOBFAloOgxsfLAK
WeHIHJOSjKKiIiIiIgUQ4ePnSA9Kxs/H2/SCkxkWU3ODklEisCGwynk5lucHYbIWZRkFBERERERKYYOHjmOOWD
AaDSqilGkGDEXWFkfk+zsMETOoiSjiIiIiIhIMVNQUMCOvQfw9fEGNFVapLjRuoxyPXJxdgAiIiIiIiJStE7Gn
yIxOZWQ4EBsNog1u12T69qsFtJWziBz1zKsWSmYfILwrt4c/wadMBgMAFiyUkhZNoXcw5ux5mbhHnkrQS2exTW
o1HnHNZ86QtrK6eTFHcCSnkBgs5743fGQQ5/UldNJW/WdwzGXoNKU6jnB/j55yUSydizB4OpBwD1P4nNrU3tb1
p6VZO1YQslH3ymKRyFyVa3crySjXH+UZBQRERERESlmYo6dIDsnB2+vCJILXMi9Rusxpq+bQ8aWBQS3eRm3EmX
Ii91POoLRGN298av7IDabjYS572EwuhDyyFsY3bxI3zCP+FlvEdFjPEY3j3OOayvIwyUgDK/KDUlZ+tV5r+9ao
gyhHd//54Dxn8l72QfWkbV7OSU7vEtBykmSFozGs1wdTF7+WPOySF0xjdBO7xXZsxC5mnbHpZOYmUcJH1Upy/V
DO6VFRERERESKmQMxxzAajRgMhms6VTrvxG48K9TDq/wduPiH4l3lbjyjamOO3QdAQcpJzCf3EnTfC7iHV8I1u
DRBLV/AVmAma/fy847rHl6JwKZP413tHjC5nj8AowmTT+A/Ly9/e1N+0jE8ImvgHl4R72r3YHDzoiAtHoCUPyf
jW7s1Ln4li+ZBiFxlNhusOZjk7DBEHCjJKCIiIiIiUoyYzfns2n8Ifz9f4NpNlQZwL1WV3CNbyU8+URhLwiFyj
+/C45bbAbBZ8gEwuPwTk8FgxGByJe/4riu+fkHKSY5/1oOTE3pwav5HFKQn2NvcQsphjjuAJTeTvLgDhdWRgRH
kHt+JOf4gvre3veLri1xLOO+mOzsEEQeaLi0iIiIiIlKMHIuNIzktjfCSIQAk5V+g8q+I+d31KNa8bE5OfK5wq
rLVSkDjrva1D12DSmPyCyF1+VSCWvXG6OpO+oafsGQkYsm8st1y3cMrE9z6ZVyDSmHJTCZt1XfETe9PxNOfYXT
3wvOW2/G+tQlxU1/G4OJGiTYvY3R1J/n3zwlu8zIZm38jY9MvmDz9CGrZG7eQskXxSESumr1xSjLK9UVJRhERE
RERkWLk8LGT5Obm4enhjtlqIMty7T72Ze/+i6xdyyjR9jVcQ8pijj9EypKJmHyC8anRHIPJhZCHB5KOYDTHR3c
CgxGPqNsKKx1tV3Ztz/J1/3lTshzuEZU5Pv5psvasxLfWfQAE3N2FgLu72LulrpyBR9RtGIwm0tbMIuLpz8g5s
J6kXOcR3n30lQUkcpXtjctwdggiDpRkFBERERERKUb2HTqCi4sLBoOB1Pxr+5EvZdlk/O96tHDtRMAtJIqC9AT
S1n6PT43mALiHVSDiqbFY87KwWQowefkTO+OV3MIqFmksRg8fXINKUZB68pzt+UnHyNr1J+Hdx5C5bREepatj8
vLHqOojkhaMxpqXjdHdqOhjEilKJ9NyScvJx9/z2lUri1yI1mQUERERERERpJgoKCth/+Ch+Pt4ApFzDqdIAtvw
8MDh+zDQYjGCzntXX6O6Nycuf/OQTmOMO4FWxXpHGYjXnUJAai8k76Ow4bTaSfv+MwGbPYHTzBJsVm7Xg7xP//
u85Yha53qiaUa4nqmQUERERERERpJhJTOsjKyrZv+pJScGO/8nlWuJOO1bMw+YXgVqIM5viDpG+YhO/Ne+19sva
sxOTlh8mvJPmnDpO8+Eu8Kt6FZ7k6/9zHLyMx+QYTeE93oHDDmPzEY4WN1gIsmUmY4w9hcPPANTACgJSlX+NZ4
U5c/EtSkJFM2srpYDDaqyrPlLn1dOyefnhVKExsupeqSurKGeSd2EPOoWhcg8tg9PC5Sk9JpOjsjUvnznJnJ9J
FnEFJRhERERERkWIiMTmFrJxcwkMLN3251knGoBbPkvrXtyT/8TnW7DRMPkH43HY/AQ072ftYMpNJWfoVlqxUT
D6B+NzaDP8z2gEKOk85VERaMpOJndLX/j59/VzS18/FPbI6YZO/KDwnI5HE+R9hyUnH5OmPe+lqhHUdicnL32F
sS1YKaWtmE/bER/Zj7hGV8bvzYRJ+GILRy58SbV4uOucicrXsUSWjXEcMNpvtCpfXdWSxWBg8eDDffvstcXFxR
EREOL17d9566y0MBgNQWJr+zjvvvMHHiHx4+nYsV/1uBITk6mT58+zJ8/H6PRSPv27/Rk9ejQ+Phf
326T09HT8/f1JSOvDz8+vKG/xnObujb3q1yhOHqkc7uwQRERE5DJd65+zROTi/bl6AxNnzKFapSXpbV5
OSoRORqub1sIHOeb+DsMESAq7Am44gRIxg/fjzjxo1j9+7djBgxgg8//JCxY8fa+3z44MGT7dMmsG7dOry
9vWnZsiW5ubn2Pl26dHnzpOsWrSIX375hRUrVtCrV6+iDldERERERKTYSEhKtn9tthqUYBQp5vapklGuIOVeO
7969WoeeugheugBUVBTfffcd69evB4fVVBTfffcd69evBwqrGD/99FPeeustHnroIQCmTZtGaGgo8+bNo1OnTuzevZuFCxeyYcM
G6tatC8DYsWNNp3bo1H3/8MREREREUUdtoiIiIiIyA3v6IlYPNzdgWs/hH3/8MREREUUdtoiIiIiIyA3v6IlYPNzdgWs/VVpErr2MvAKOp2RTO0lA7oYvzFXklY4MGD
ViyZAn79uOVWraxcuZL7778fgJiYGOLi4ow/f+rVq8eJq3oHxMHnLxNmjt04SYpTdW+m/3fb0QEcT8m
unNfNy8sjPT3d4ftfJKHbz3OIqPe5DmPT3d4SUiIiIiInKzyM8vIEGLHKC4/AIVzqOnv
7//Pl126dAEgcCBueUQEtcy7J+YT4ReIkNKYNFkFgHIAJchUrG2bNnM336dGdM336dGbMmMGTmTZuY0nUqH3/8MVOnTi3qW3tqSzl48803SUtLs7+7SUtLs7+8803SUtLs7+OHTt2V
a8nIiIiIiJyPUlMTiE7JyeH+t8deyZeyjLfKPeyJyHUrMUpJRrg9F/qut119/nQEDBtCpUycAoWowZEjxDhPPvk
kYWFhAMTHxdx2220OAhIWFkZkZI7CQ4DBuQUEBycnJ9vP/zd3d3fe/1x4RERERERG52SQmp
fyYl6sko8hNITHD7OwQRCrUMmwYnZ2N0eg4rMlkwmq1AhAdHU1YWBiLFi2yJ709PZ17qir
ROdH2PkuXLsVqtVKvXr2iDllERERERXOSGF5+YhMHwz3slGUVuDkmqZJ25b333+fMmXKcOutt7J58
2ZGjRrF008/DYDBY6NOnD++//z7lypUjOjqat99+m4iICNq0aQNApUqV+Kp/5gXi3EnuZKL
Os7SIiIiIiMg5JCQm+vJKPIzSExxNixvxNixvP3227zwwkjCzaA5kunbWIrwA8zmBn1RvfWMMgwEDBuRpBgqGeb
bKysujVqxepqancfffdLFy4EA8PD3uf+fPn8e677+Lh4UFISAjt27fnqaeeKuZ3LkgEIR4UBM1A5i5PnXN86d2K
OuZwmXRYPA2JubVNEZQbTx23W3mn+1jwvlSjGsrOuZd/zQ37ONHtXz0Bu2O/bp4Az2ISHXD/7uEheu9c
9ZIvLfbDYbbbwcjR09ztKCzhTAlTj9zi9wMSvi4sfGte5d1hkjRr8koIiIiIiiV23irgD7ffgwbKDQ8L95O+x8eMjIoz5XoyqIx3ZKzScnOx2otlvV
jcoPRvzzwiIiIiIiiI3u0zcXPLzC3DVzte3fmIP89BXV4SWDqRrgf6Lkj+QdwRNyyLS1XI+zCdAt4Qus/P3DvbLZoLMoR2vC/iw3DtYjZ0yvuIoaNmyx9ZgzcJFd+Yx5YxCXmF4zYQ0tLr8smQHBz54g3CEk0OTMVYKZfM7oCYrpRe7bfA
0dEREREROQGl50bS35BAa6e6BA6aLi1yU0r0UpJRnE//8oiIiIiiNzgsnNyKSj4Z7pOnpKMIjeVfIV2SGI4OLsAER
EREREROTK50TmYbPZMBoXThHTM5oLk4tKN0hxlrHpF9LWzcWSYXJbyXIEtXgW94W94jK5+ybvXc1aWtnk58SC9YCXAIj8LvjY

```
XyqNwPAZikg9a9vyDm4kYK0OIzu3niUrUXAPd1x8Q0u7FOQT9LCMWTvX4vJO5Cg+17AM+o2+zXS1s3Bkn6KoHu
fu+r3fj6WK9j4JS4ujvfff59ff/2VEydOULJkSW677Tb69etH8+bNizDK4mHw4MHMmzePLVu2ODsUAJYtW0bTp
k1JSUkhICDAqbEoySgiIiIiInKDy87JBYPB/l77zEpxlbV7BclLvyL4vhdxi6hMxafSJg9m5a1BJvSCKKnw6d2k
TB//6HXANigSTCzkH15P026eYvPzxvOV2bAV5mOOM04t+gE24ly2HNzSR5yZecmvsu4U9+CkDG1oWY4w4Q9sTH5
ByKJnH+R5Tu/SOGg4H81Dgyt/5u7+ssBZeZZDx8+DANGzYkICCAjz76iBo1apCfn8/vv//Oiy++yJ49e4o40hu
XzWbDYrE4O4zrmmroRURERERERebnDZObkYbEotSvGXvmEe7hGVSV31HbFTXiIv/hAYTWRuX3TO/gVpCZz6cRgnv
36ek1/2JGXJRLBZyTu+CwCjuzchjwwk7/gO4mcOJH7m/7Bkp2KOOBBegIA+UnHMLi6Ez+jP+kbf8KanYY1Jx2
A5D8+x7NCPRLnf3RtHsB5WKyXV7/8wgsvYDAYWL9+Pe3bt6dSpUrceuuuutvPLKK6xdu9be7+jRozz0OEP4+Pjg5
+dHhw4diI+P/8/xt2/fTrNmzfD09CQ4OhevXqRmZ1pb+/evTvt2rVjjJAhhISE4Ofnx3PPPPYfZfP41JpOSknj
88ccpVaoUXl5e1KhRg++++86hT5MmTe/fG39+fEiVK8Pbbbb2M74/+T33zzDXXr1sXX15ewsDA6d+5MQ
kKCvX3ZsmUYDAYWLFjA7bffjru7099++y1Dhgxh69atGAwGDAYDU6ZMAcAcBgMPDFF1/wwAMP4OX1RdWqVVmzZg0
HDhygSZMmeHt706BBBAw4ePOgQ608//USdOnXgl1tuYciQRUFNjbDQYDX331FQ8//DBeXl15UrFiRn3/+G
ShMEjdt2hSAwMBADAYD3bt3/88/lGtFSUYREREREZEbnNlsdqhetNkM5+OrcqOyOqYeOjIl35iHVPzPNwNz78AjO0bi
H3oI1L5vco9vPe57BzYvSL35DqRemUeLBARhc3HCPrG5vTn6FTmHNlGi7aiPTmHNlGi7atEPDMer4r1Acg9/k8VnzkhhpBH3
sa7SqPCWGw2Mnf+icFgJOfAOoLue/4q3fXFuZZwlGZOTk1m4cm4cCEvvgi3t7eZ7WfnprtVp56KGHSExM5M4778R
isfD999/Tvn17sr0zad++PX5+fhgmMBlJTU4mKiuLTTz81lKyuLli1bEhgYYyIYNG/j+++Vv3drj0kiVL2
L17N8uWLe07775+bmcvvtt/Prr7+yY8cOevXqvXqRdeuXVm/fr/r1Dv6lTp+Li4eZXXm/fr1Dv6lTp+Li4sL69es
Ti7N8uWLeO7775j7ty5DBky5Lxx5+bmcvvtt/Prr7+yY8cOevXqvXqRdeuXVm/fr1Dv6lTp+Li4sL69esZPXoOo0a
N4quvvvrK35+fn8+6777J161bmzZvH4cOHz5mgGzBgAB988AG7d+/m3nvv5dVXXX+XWW281NjaW2NhYOnbsa0/77
rvv0q1bN7Zs2UKVKlXo3Lkzzzz77LG+++SYbN27EZrM53P9ff/1/1Ft27deOmll9i1axfPEFUU6ZM4f3333eIYWq
QIXToIFt27bRunVrunTpQnJyMgRGRkcyZMweAvXv3v3Ehsby+jRo8/aSjHmnfH
83dG3vVr1GcPFI53NkhiIiIyGW61j9nich/m7tgCT/8uohqlcoDsDXDhw0Z+vt5M3rUfSnhHCCuwAOfbHf8s93
wzXXFI8+IzehCgcmNAhdXCkyuWIOmLEYXbAYjVoMRGwashtO1SBawWbBhpXCVz7+/thV+jrK9j+/f4cX9ts/
3r/z3HDmX1s1vPO90/LyeW9+YupUSqMbg3rAmC12Xj7x4V4urryUjYYP23egQ0osFgxGgw8XKc6d95
Sxt7n44XLqRUZzr23ViLfYuGcPzQRaTRLBAWT1mknOz
ibY25sOd9TE38uzaP4wL8iGwWCOr5LQus9rVGnQ+JJGWL9+PfXq1ePLL79kzZo1LFy4kMTERMLDw2nXrh2DBg0
iODiYRYsWcf/99zNkyBBGjx7NkOkmTJrzxxhtMnjyPqJECUJDQ0lMTMTb25vp6vp06fTvV+//jh07Z
k9i/vbbb7Rt25aTJ08aTJO8SGhpK9+7dmT9/PseOHPLywuAcPPLywumACRMm8POWlmZfZ/a/PPDAA1SpUlKxTMH
IYOf0nRj+fkgDBgzg559/ZteeuXeccY+PGjdxxxx1kkZGTg4+NjX+tw3PsRGy2Zcl08ZwPvAMcHcshAmHzAYe
Outt3j33XcBWLt2LfXr1+frr7/m6aefBmd5QhMmzmAO7sXTpNTx1N4kRNdHsSYcDGcPhN3
rKwsfHx8WLFhAq1atGDBgACNGjNCajCIiIiIiiiJSdH7Ia0YHd2/6u0/ixxKlme/jwWHHmkQ2le1Em2
Y3wZAPhKQUEpebin5aJa36+fQyLwUiepx+5nn7kevhidvfF7OZNvps3+a5eFLj4UeDiicXkgcjtXohtXghs3
gCjYXsJnAasLA5VXUWm0WMBRgwwKGvxOcBguGtDhgMWVLN8alxP2AFRsWAn03kZ6VirHEfTgkRLFgSNyAuWAbP
l7euFlteHt68vPWPQSH3Or5iNKAlbIRh9mVkECdmsH8tHY5uZbCNQ4rlquCzackpcLNbDh8kpfa3k9yRjpf/L6
Y9UdiqRoZwZ7jsbiYjLxyfzP+3LWfeVt28mSD2y/zzi+Rzcrpsj
Gdh2HzpupOXv99dftU47LlSvHzp07Ef311
1mwYAFr16519+7dREZGkpKSQtWqvalevbAKNCAggG3btlG1alUG6duzIkSNHAGjUqBguNkHDNjqvL8Vm8
tsVgYNmwYs2fP5sVhzGALjrrrvsC
cbT444cORKLxYLJZCI6OprBgwezdetWU1JSsP495fzo0aNUq1bNvc3f17dunXtzvH0pWbOm/evT91ejRg2HY7m
5uaSnp+Pn58+fWrVtWqVQ+WixWIhNzeX7Oxs+z2dn4K0LE4u7uvfOoyoiiIiII3OBsNpvDWmNGQ7Gcs
CYXaXZePXYbSjLRMpKeackcNPrytU8Zlvta0ReVzLqos88pmepFmURPIhNdCE8xUDLVQom0VErEn8TtjPXhLkW
eqwc5Hj7kuvuR5+ZNnrs3ZlcfzG6eFLh4UeDiSYHJA4vJA6vJHZvIH3ZvRHZvBFZvRDQMuGGxuGGx/JyttYLIVrtNnt
PnhavmnEtGIG2DC3XLrWTGU8vXiiaYVKBV0CznmLJZsnU1i2kYWb9xJ9QefAeDxu1swffnHfPDdVABMBhc6N3m
V6pH3AVC7HJxMmsq4Xxfj6uJ0i1pPsP3IauLSzJQLu4uMnBRG/7EN5NeS1ILNVoDNYAFbY
eLUdkYStDAhWsDpxKjt7wrS0wnUf5Km/xwHK7l5uZQKC8HfzwtsVtw8gy/5z6dixYpA4f8//vjjDzj6/lHc99JTa
pQ+3atSlfvjwDBw6katWq+4hMz0ThQsX2se78847mT0k2aNOHTTz+1HO9NTaV///7m5sq+2gRyMuH+gzdvyv36
zePlNRUFavXk3jxoxoWVl2lpaTRq1IiNGzdSokQ5kKFSrg7e3t3t7gvGPWDAAMaGOHhwdms5kFSrg7e3tsI7j2rVrSU9
P5/HHH+fnn38mICN1v34+CCAAxB4wN6elZVF48aNcXlJtc3l4CAA0644w5++eUXl7jmz2n358uW6899577Nu3jyeeeK
7fanwPAn3/+SVRUFFBYHTpmzBg2btetxof7779u3j+eefZ8+ePp8bkvemZmZmJmAtGHz5s0cP36cnK0VK1ybN55
44gk8PDw4fPw4fPgwUFh1+fnnn7Nu3Tq2bdvGwokTHAwNDaVKlSpjxo2jEHjw4sVj77DgmMYwHAAqKvYesR
L9XwTIpyXjT224rx30ZQxhnHkzjEkkMS98J6fBTrjdTXX05UNoVm8c/VVkJAWYSAsxsrHD2WIEJJsKPGwiPh/A
```

```
kG2HpEJZto2S+FY8LxOCen4t7fi5kJF72fVgwkO3iQZarJ4szMgBYsGESP6//imdrtaRqeFWSOO/i4eZJVmION
pMHuHiAyYMjiYdZED2DhLQT5BfkEeQbSv3K97M/ditJGXH2ayyInsamA8swYMBoMOLt6c/Mvz7F3yuYKqVvZ/H
W2fy1s3CjjQZV7mf9vkWUD6+Jm4s7cSlH2HtiM+8/MYuVu39h4aZvATAaTICp8AKnH/PFljdeRD9fLzCnw6nOv
O8xhVzk4Odm/dfGMWFhYTz66KPMmjWLmTNnkp+fz+OPP87hw4eZO3cuhw4domHDhrRr1474+Hjmzp2Lm5sbQUF
BuLgUppqqVq3KlClTuO+++8jOzubbb7/l6NGj9OrVizJlCpPEGRkZbNqOiREjRjBpOiROnTpFpO6dcHFxITIy8
pyxrl+/nrp16/LFF1/g7u7O1KlTGTZsGPfee69Dv61bt9KhQweGDBnC77//zksvvURkZCQmk4k9e/aQnZ3NzJk
zqVevHocOHaJLly7nvN7o0aP56quvCA4OJjw8nOjoaA4fPsz+/fsBCAoKsk9t/u677/2/3B
z5h9OjReHl5OSQ6ASIjI/ntt9/49ttvqV27Nps3b6Znz56UK1eOChX++cs4ffpOvvjiCypWrMjAgQPZvnO7Fou
FBgOa8NJLLzF69Gh27txJUFAQPj4+F/VnfjUoySgiIiIiIlLMaIdPAUh3C+ZJ1O8YFPsBT4UXrkP3kEcWD5FF5
nEDE/MD+cXPm4QwgzOfdi4pJQ2klITCEc7MmBkJiodSJwyExxsIT7YRmm4jNNtGSbMF9yK4BxM2fAty8C3IISK
nMMlotRUmxIJPbmbV4XVYLflUN7jSdsckBsSepKSLC6+ElGRXbi6Hc5J4KsCbUm7BROcXMHH9RApsNkq7exCwb
zInzQX8sXkmLkYTj9dpR3BABEnZmUQf3cwfm6bi7mLj1w2T6dVyMAaDic9/e5PqZe5i34nNvP7I57wz4wlcTC5
k5qaSlBGHzWYl15yNh5vXhW6ryLm4Xvrf+tNJMpPJxJ133snQoUOpWbMmBQUFLFqOiIULF5KSkkKtWrWoUaMGy
5cvJywsjKNHj9K3b1/uueceypYtSOpKCmFhYWeN36VLFwYMGMCGDRuYP38+Li4uDBs2jK5du9oTcNu3b8fV1ZU
tW7bwwAMPkJqaSlZWFgUFBZjNZjw8zk5j33777fzwww+kp6cTGBhIfHw8BoOB2FjHvTIMBgMJCQnYbDZKlCiBy
WTC19cXKKzWdHNzIzo6mjvuuIPMzMzzToX++OOPadCggf19YGAgu3fvJi4ujtKSzv8kqddu3aObNkSgKeffpq
+ffvSq1cvGjZsCECbNm2YOHGivX9mZiZms5nt27dTuXJlqlatSvPmzRk4cCBPPvmkw7ht2rQBCjeB+eGHHOhIS
MDNzcO+pXzDhg2Obt36nPdwrSjJKCIiIiIicoMzGoOOH5A1XVpOsxlcGBL4FvvSvmewzzzcTYXfGz4mGy+bknn
ZnMyhQ658aQjmrxIepAdc2jbFyaGQHGpju8NuLQbAheAEI6VPmohIMBKRDKFpVkpmWwnOy8fddunbITfz9aWGh
wduBgPROTmMOJVANQ8PfEOm7vQqXPcvNj/fnmSv5uFBGVc3vk5OIr6gAHeDAW+DgVzgbleoc3Ijc2MLK9AKrBa
+if7R4Xq3ursTsX40VV2NvHhwGnvy8vgSGymxm3ghshJbl/QnOs2N5AIrn/zYmxAvfwBy41di8wjE+ndVpcHki
cHkjtHkgcnkjovR7ZLv/b+4uFOgS/wfPvroIzZv3syrr75KbGwsISEh3H777Tz66KN8/vnnGAwGfvrpJ5o2bcq
WLVtoOaIFrVq1YuzYsQwfPvy843p5edGtWzfGjx/PY489hpeXF+3bt2fUqFH2PsnJyRqPUFPDDDz/w3XffAYVJT
4CYmBiqVq161rj9+vXj559/plGjRthsNlxcXLBarfbNVE6rU6c00Tk53HnnnZhMJpo2bcq+ffsACAkJ4dVXX+W
TTz7ho48+wmg02q/773F7ZTlw4ABNmzYYlNTWVyZMn06RJE3vbaSVKlABwqEg8PaUZCqdsx8bG4ubmx
tChQ+1TnAOGw1nViGeOGx5euJFuenq6fUx3d3cGDBjAU089Rbdu3Zgy+1aUJJJRREREREREETkBvfvHVg1XVr
+7Tv3xzhiq8bnxrEEWNMd2m5xyecD4iAF/oj3YpqbHzvDXCnwvLLtS5JKWkkqaWXrWS1GSiS4UPqE8e8p2BCab
qNktoUSZgvu59teGugeGMSbcYUVa/1DSrLfbOaI2czD/oUJvnBXVOq4/JPqcDUYeDsOjDKurmzNyeGt+DisNhv
t/QMA2JKbS6iLC5lWKOYgxMWFmh4eLMjIoL1/AJXc3TlsNnMyPx9foxEjEOlipL41kylxxxgSFkbvEylMLh3Ji
qxM/nBz45GDP13wuRQYTOS4e5Pj6kOuxhdvHE7OJFvsmDAhcPLEZ3LC4ehWtVGt3B5I7B6I7R6Ib
J6Pb3lOy/79P90pOMFSpUwGAwEB8fz7hx4xg3bpxDe69evQgy5Ytu+hr3HLLLYSGh
to3hfm3/Px8oqKi+OOPP85qOz2l+t+GDRuGZWbjhx9+oEKFCnh6evLoo4/aE32nmUwmxo8fz/jx44HCac+nqzc
PHz7MqFGjeP755+nYsSNBQUGsXLmSHj16b162BObVapUwd/f3yExeHrcO++k3nz5tmPnV478cwp26eTgXXq1LEfq
1q1Kv7+/gQEBBAfHw/ApEmTqFev3lnkX0NOZ457+hdLpikkADw+Psyo5nUFJRhERERERkRucyWTkzIXcPIyXXiU
mxd/qglu5L+8dpnh8RDVT3Dn73OeWzX1kkxNrYJophHlenhwPsl1wOvXlSCxpJbGklS1ntZgIT3ahbLwrpROMh
CXZCEm1EJxRQEBOPvf7+ZFssfB+Qry9kvGLOpH2xOKZ1YwA6VYLzx4/xum/n52Oz2Sj
n5k6ipYCjZjMxZjP3+/rSMSAAg8FAv5AQnjl2DIvNhrvBwLDwcAbExvJqSElSLBbcDAa6HD1CqIsLnOSU+s97d
7FZ8M1Nxzc3/T/7XkiBhzcEhmD1DsThE24tLX4goODuffee/n88888+eWn6dev2nzsqX0j
NmjU5fvw4+/bto5KlKyNHjrT/k
mb27NkXdXO3NzcsFstFx3s+oaGhFkQklFERERERQGZzKaODMP4KUko5xHgimcB/OG8UHB+zz
qd/C8/TwNNp61JvBsJhzP9WaqfxkWedpIcs2+6jHGBhUQG1QAZ82UNVI61ZPyiUHwXjy9776VVt6BBKblYUnPx
mSxMLVMWYczBpQM5YmAQLJtVrbm5DLqVAIxZjPV/l7rz2qzUd3DkxlnTEd9Pz6eHbk4uBk59uRap4BAOgX8U802Ly0
Nb6OR2zw9aRNziFllo4gvyOeNk7FUcS+Kl7WnbG5gkjtuFsRmAYdx8bm8NSDHjRtHgwYNaNmyJe+99x7lypVj586dv
P7665QqVYr333333//imK85557aNzzzyDt7c3u3btYtGiRWdOYzN7zs5euFtfvnYVefjbyvlnwfUzAaxyzDOPaDyPk47c+Ou9kwkJ
VVgIMHjyYY8eOMXfuXNq2btYvBYODtt98+a+MaKEwyfvjjh8mfff1PHd9+wyd8ofMF+hafvKbePSGsybv5y4yrQvrY+sHDCwBrL0BmuvVHtGB2QtNjRmbtDr8Dfz2bJ65MLKX3qzNb0g3Czt+Q8nbu/+OasiE2evTjPhhrtmaznU+kkLd/Utx3tR+1IZTmYwrHFjNPvdnRfbLVTjT2C5w6JGN1
mrVKiZMmHBR9xQcXvv//03r17CQ40xv/vKtrLMWTIEPr27Yu/vz+tWrUiLy+PLRs2UOmPUvk27YuNmnJGvsE
kyRJq1aqFl5cXXl7Xdk3Q07QesIiIiIiIyA3OZDLCGZsPeJquj6oWT4VGD14ze1d3k9tRv5F5KNLF2QxMGk3y
47v4cu4XO7J8cfDWvTrCl6M4wF5LK9QWAH4Y+0MXn881lWeey+Hk+HxNwy88rw3nz0WzPxmoUTXDuVIVAnSA30J9/K
kkrsHjwUE8GRQEJ8l/rPbdYiLC+lXvJsbsbsQUF57x+SkEBnyc1MrBkKNtyc4hycyPKzY16Xt4UYONwvvnq3
```

fz5mEwYvbOv+bTu3btTqVIlEhMTWblyJU2bNiUqKoqePXvStGlT1qxZQ1BQOBWHd+edd3LixAkef/xxqlWrxht
vvGGvvPvpp59Yu3Yt+/bto1GjRtSuXZtBgwYRERFx3vEiIiIIDAykQYMGtG3blpYtWzpMSYbC3bEHDhzIxoObq
V27Nu+99x6jRo2yTzGuVasWoOaNYsSIEVSvXp3pO6dfcH3JM/Xs2ZPKlStTt25dQkJCWLVqlb3t559/JiAg4KK
fzTPPPMNXX33F5MmTqVGjBvfccw9TpkyhXLlyFz1GgwYNeO655+jYsSMhISF8+OGH9rbDhw9jMBjYsmXLRY93J
VTJKCIiIiIicoNzc3XF5jBd2oYJGxaubEO9Kd4mejzDOfzqfGgaj79L/kWdUz8ngfo5CeRhZI5fWeb5+rLXNR2
rwfnVs8cD8jgekAf/nnlrhdLp3pRP9GDf7zlk7szgaFQJAtLyuM3bmxizY2LwcL6ZCBfXc17jg1MJdAsMJMzVl
R25ueSfkdy32GxYnLAcqunvad2Xo1WrVkyePJn8/Hyio6N58skn6dy5MyNGjDir76effnpRx/49ddnTO5MyZcq
cN9F1xx13nHNNxvNxc3Nj6dKlDsdefPHFs/r5+fldcArOyy+/zMsvv+xwrGvXrvavu3fvTvfu3c86LyQk5Jzxm
s1mpk+fbn/fpEkTh52nzzdm586d7btt/1tUVNRZYwQEBJx17My1J51JlYwiIiIiIiI3OG8vT8Dm8MFT1YxyMX6
33cVjtg+INYReOnnuWOmcHsPsE9tYeCyeJ9K9CbX4XqUoC1lyLeQcySHnSOHuv+ZEMzlHcjAnFSYJ476P4/iXx
+39kxYnkb45nbwEMwdzO/nxRAxLth/DONyP1/6ugNzbN5jNeTk8FZ7N5Nv9GBsIs9LTaFc6HIuL4OKUq7OyOGw
2O/nvqdPVPTyIMZtZkZnJ7NRUjAYD5dyufYWny9+7GF8Od3d3wsLCiIyMpF27drRoOYJFixbZ261WK8OHD6dcu
XJ4enpSq1Ytfvfv jhB3v7smXLMBgMLFmyhLp16+Ll5UWDBg3Yu3cvAFOmTGHIkCFs3boVg8GAwWCw73ycmprKM88
8QOhICH5+fjRrloytWx23CfrggwIDQ3F19eXHj16kJube8H7sVgs9OjRg+PHj/P6669TuXJlRo8e7dCne/fut
GvXjiFDhtiv/dxzz2E9m8cOFC7r77bgICAggODuaBBx7g4MF/lhc4XSE4a9Ys7rnnHjw8PPfTpPPfUUaWl
p9ns9vWNOVFQU7733Ht26dcPHx4eyZcvy888/c+rUKR566CF8fHyoWbMmGzdudIh15cqU9PTyIjI+nbt
y9ZWVn29qioKIYNG8bTTz+Nr68vZcqU4csvv7S3n66IrF27NgaD4azNcD4azNcYqakowiIiIiIiI3OB9vLOwmEwVnLP7
vqXUZ5SLts4Rzb+5wNptqXtb54ZYc+iftZvHRnXwdmOOzbH+8rEW/NmFOTA4H3znIwXcKkz1x38Vx8J2DJMxNA
KAgtcCecATABvE/xHNgOAEODjlI8tJkQjuEUvLhkvYuXuW9iOxbhiOH4vl49gYmHYkhuGsYP7zlxeOvYp+CPbd
xCd5KS+S52lXIDPLDYjIR5urKwJKhDIyL5YukRIaHheNhvPZplitJMp5px44drF69GrczEqXDhw9n2rRpTJgwg
ZO7d/Lyyy/zxBNPsHz5codwzBw4cyMiRI9m4cSMuLi48/fTTAHTs2JFXX32VW2+9ldjYWGGJjY+nYsSMAjz32GAk
JCSxYsIDo6Gjq1KlD8+bNSU5OBgo3Yhk8eDDDhg8aNhIeH8/nnn1/wHqxWK6VLl+aXX35h586dDBgzg9ddf56effnJ
e+sisYlS5awe/dulilbxnfffcfcuXMZMmSIvT0rK4t/Es3smTJEoxGIw8//PBZaz/8OGDCAl156id27d9O
OaVM+/fRT/Pz87Pf62muv2ft/9KH0mTRu6du1Kt27deOKJJ9iaRPly5enW7duLy5enW7du9l8WHTx4kFatW
tG+fXu2bdvGrFmzWLlyJb179y4+r7Hr2pf4ayn1793aIYmdStW5fNmzfzwgsv8P6zzvwy8Pzzzz+PqpNEG/vpNEU4Ly1wMhlTT+Tr0
7xrKYSE9Px9/fn7SONPz8/K769ebudf5W4TeSRyqHOzsEERERuUzX+ucsEflvx07uxx07KLvxvdwtKyR9mB9nT+X
ZLr+R9nijga4TWdx6y/YeTKUgX5Gjjr5Vn7scrs+plMXKauNOukeVEjOoEySC+EpEJJiISAtD++/N6G5Vvw
feYSIYZe+QUv37t359ttv8fDwoKCggLy8PIxGIz/99BPt27cnLy+PoqIiNnq2+Kb5+/p69J23ZvWWkdnoneUYDTBs3eZdV4buNv
cto2rQpixcvpnnz5gD89tv2ttvtGnnThpycHDw8PDBw8PBg8eDDz5s1zmC69cuVK2rRpQ0JCAu5nbJZ3TXr16kb
DBg2oXbs2n332mb39rrruIjc395LWGOzduzdxcXH2Cszu3bszc3xmIuNt1CjeOoDsEtBeD9nKtv
iEhIWzfvp3qu1atz+PBhypUrx6effLuTCC6euY6zbJzF
HAoWPX9evX5/Y2FjCjCwsJ45p1nMJlMfPHFFw7P7P7Z577iErKwsPD4+zxzExRUbzyxKBsqJTvCtt6+E2fbFf0pCVt9l0y+66tLvhcSz

+hndrM0JqMTwSwWZo8byYqf55CaeIrAkqEOfbgDjz7fz76eQmrikb75+H22rlpOVkYa+33iMiMi6paLusb
KX+fxyasvcEfzlgz4bDIN1JMkzg4bDIABfn5fDd6BJuWLyX++BG2fx+VuvsWHJ7wSUKEnPd4ZRq
OFj+7jzvv6cxJMneObtK9tpSkRERERErpy3lydurq7kmvvg77yidpiWy9E/juio2OGYf
pmAEJJg+m+Ufxh7eJWJeMSx4raWkSyUuTyU8sXDfSvZQ7JR8qiW/N/56enbo2leMTjuNb25eyLxXuIGOrsBE/N
56MbRmYE8yYvEz4VPMh9LFQXAML12KO5ls5MekEGZszcPF3IaJbBD63+tjHHPbUwkZNJ+RzvGnHONSAj/14D8lo
kIF3DLm2q+5maNm3K+PHjycrrK4pNPPsHFxYYX27dsDcODAAbKzs2nZsiUsdUBzDPDAAbKzs7n33nsdzdzjGbzdSuXdvhWM2a/1TBhocXFhUlJ
CRQpkyZc15369atZGEhwc7HA8JyfHPi159+Hd33TPPfccevZE8ZI3TQxU9t27kPxZfkP4HhH8ISWnXo3rGTuyvEmMuPx9PvkZ9dGCDg5W4rdVg+MFhJS
Bdn0X5jPHzczM5NixY5QtW5b9+/czaNg1q1bR2c+ejRo0ec8cnP68r5jSvXt1+Xt26dS8Yy5nAfEahoYv2/jvq1DjrWp
EJCAmFhYWzdupVt27Y5rPNs9mwWqgEA5yEhYWrlJkBwOXEVJSUZnWjexM/4/bup9PlgNJEVKnnN
wx1bG/e9lvHx8adPtGGWw2GyNefBgqTqwsDPp+Mp7cP86d8yZCnDd8yZCnOzL6l+V4/MeW5AnHjzH1w3epWreew/G83Bw07
drOoy/OI6pyNbLSO5g0bBAfvNCdD+csBGDRrG85tHMbw2bOZ/OKpXz62kotMWrUNg8FA/PGjLLhqz0
ERERERC6ej5cXbm6umM0kKzzZqe15DdLqX4ym0kQdakKx6vpCWX15L38FoybHYP5puACFZ75JFlvPDP6e
qe5BroS9lgYbqGFO3hTV6ZydPRRyg8tjOcpj/OeZz5lJm5WHF6VHD87W81Wco7kUPLBknhEemdJmDJshA7I5Yjo49
QYXBhxjBlWQq5R3K55e1byNyWybEJgypgoGgwHzKTMpy1MoP7j8ufWVhqVeWgHQJC
7vkcO7z9vamQoXCICdNmkStWrX4+uuv6dGjB5mZhX++ZP369ERERDic93fFuaenp/3Y30knm8123nOsWdPJS3ZoIxjwL3rwm0O49DTT/7O4XZdkiy+yfkL1jxfZn/Pf12muhFtnKMk4VfCvHNxZdUGUk4vJ+hPDQiDUV7RQtvQ9UXCY+jVIY+Hw6HzJKVNDtvZnMOYdF5nJNpebf5gX8+tOodiOKZ1v/X9H07hmX6u3wSx1ZjEBsTESsDGtmHN4xg++YgE29edOo/42Z7mxwqCjZGfHCpX9lRadtQJBb4QQLcPxhZl9jMmtGw5ZtPhF+XwYt7cE4XrZYbaznrszhi7RJPK8iWeL0ERJPYrHXJkfe4IxFxe0nYa75k5njtWUoI36y8vd+fP/oQ+W5RL0NnuLZ7PvE2bRbvLR9JIQsJ4U/M3zdYbzPJ5gWC1kYf9znE7lk

9fIh9W6P5ZP6flKlYGYBegz+gx921WPnrj7R4rMt5x7ZYLHz6+ot07PMquzeuJysjzd7m7evHO5NmOfR/5u336
f9YaO6dPE5IRGmOHzpA3Wb3UaZiZUIjyzDto3dJTOnGPyiYLwcPoOtrA/HyubqL+oqIiIiIyMVxc3PFx9uLpJR
/fu7XmoxyJTYVRNHCMpzvvD6msmVfkY1bOy+J2vFJFGDgZ98yzPH1Z6dbBhbD+ZNrfrUdl+YIfTSU5D+TyT6Qf
d4ko81q4/gXxynZriTZ+7KxZP8zvsnLRLnXyzn0D38inENDD2FOMuMW7EZebB6+t/niUcoDtxA34mbFYcmw4OL
nwsmpJwnrEIbJO/Tvy/63/OpApvlQPtGdMskuRCRDiYtIQLqdp1rwkkMzGvnf//7HK6+8QufOnalWrRru7u4cP
XqUe+6557LHdXNzw/Kv2OvUqUNcXBwuLi5ERUWd87yqVauyrgYUHHcYSFRvxvUKXtP7BUjywYehtV4Nha9ah42AW+fV
NWO7bunUrOTk5eP69vMTatWvx8fEhMjKSpKQk9u7dy8SJE2nUqBFQOE35cu/1ctWpU4ddu3bZk8CX4/T6mkUVO
3/Rxi9OVLl2XbavWcnJmMJv+MN7drJnO3pqN24GQP7fGXK3M35DYDQacXVzY3fOhguO/f1no/APLkGLR8+9Dfq
/ZWWkYzAY8PbzByCqcjX2RK8nLzeHLSuXERgSil9gECvmz8XV3Z16995/yfcrIiIiIiJXT6CfH2Zzvv29tyoZ5
Qol23y4P+ttfjM2K/KxXbDxSMYRpp/cxuKjJ+mR5kmpgv9e59dmtZG6NhVrnhWvCuef3ZfwUwImPxNB9wRdVDz
WHCsYChOQAB6RHmTvz8ZqtpK5PROXABdMviZSV6dicDXgd/tVWJPYaOBYYC7LKqYxrV4SH9yfZN8F+/HX4NXnf
Pj8OWDm3OXL9nplyGlQE24pi2tkZJGF8Nhjj2Eymfjss8/w9fXltdde4+WXX2bq1KkcPHiQTZs2MXbsWKZOnXr
RYOZFRRETE8OWLVtITEwkLy+PFi1aUL9+fdq1a8cff/zB4cOHWb16NQMHDrTvsPzSSy8y8xadIkJk+ezL59+3jnn
XfYuXPnBa9VsWJFNm7cy0+//86+fft4++232bDh7PyJ2WymR48e7Nq1i99++4133nmH3r17YzQaCQwMJDg4mC+
//JIDBw6wdOlSXnnllYu+18zMT/Zz+rX///qxevZrevXuzzZcsW9u/fz08//XTWxi8XUrJkS
Tw9PVm4cCHx8fGkpaX99O1XQJWMTvRwr95kZ2XQt3VjjCYTVouFzv0OGOLjtIwCUuqUCJSJK8e2o4Tw3ZATunl7
8MvVLkuJiSTkVf95xd0evY8mcmYyc98dFxWHOy+Xbj9/n7jbt7NWJzdp34si+XfRr0wTfwCBe/XQCmWmpzBzzE
UOn/cCMTOew6refCIOsy4vDRhEcqo1cRERERESScKTjQH3P+POnGAJcCJOYjxYUVEy9kP8PzHlG8wjRcKfrvqxL
WPPol76VfMmxzD+Ib/whWeuaTacyx98k9lsuh9w5hzbdidDdSpk+Z81YxZu3LImVFChWGXlwFmNVsJW52HP71/
O3ViYGNAsk9lsv+/+3HxdeFyBcisWRZiP8xnnIDyhE/J560dWm4lXSjVI9S9rUcr5q/E5DHAnOhIkAOcBJ3kzv
rI4ru87iLiwu9e/fmww8/5Pnnn+fdd98lJCSE4cOHc+jQIQICAqhTpw7/+9//LnrM9u3bM3fuXXJo2bUpqaiqTJ
O+me/fu/PbbbbwwcOJCnnnqKU6dOERYWWRuPGje1rE3bs2JGDBw/yxhtvkJubS/v27+efffffz3utZ599l99ls2
bN9OxYOcMBgOPP/44L7zwAgsWOC731rx5cypWrEjjjxo3Jy8vj8ccfZ/DgwUBhcdfMmTPp27y1atXp3LlyowmZM
4YmTZr85702aNCA5557jo4d05KUlMQ777xjH/dS1axZxXLlzNw4EAaNWqEzWajfPny9t25/L4L4aLiwtjxoxh6NC
hDBoOiEaNGp1zinpROe7SReRydpde+es8pn30LtlfpWJ2bOTycPeofuAd2j6cAcADu7YxudvvvcLhPbswm
kzUrN8Ig9ENhtvTZx+1pg5mZm88lBzer4znDp/V0SOHdCPrIw0+8YvZyrIz+ejvs+QFB/Lyetyb/Yi
qWp3Q0pFM/+QDPpj1K/O++oyj+/fyxtivLunetbu0OiIjIjUu7S4tcn35csITvf11EtUr/rBM3LTYMsO0T2KRoN
HbdwziX0fjZrm41FIAVmO9Thjl+AWx3y8BsySc/KR9rjpW0DWmkrEih3IByZyUaLTkWDrx9gIhuEfaNYY5PPI4
l22Lf+0VMtgIbR8cdJT8ln3IDyl1wCvTxr47jUaZw+nT8D/GUH1SeU7+dIu94HmX6FM2U5UtVObAyPzz4g1Ouf
aPq3r07qampzJs3z9mhFDuqZHSiaR+9y1K1clcSTx5n75v/LVz8Q8KZC8/bkD7ZH3Kxl4JcXx4NBuLNDo4ytoVs5n75Vh7krF89ZqMnLeYrIx0CvLz8QsK5q0OD7B/OmBCvLz8Q8KZkC
HNpSvXvOcY8Yd0OzCiWMMf/5J+zHb3wt+PnZrJGMX/EVYmSigMME48uVnOXXyBEOmzL5ggnH721UcO7CP598598by
bQP36VO42Z4eHnR4P4HWTD9kSJ4GiIiIiIiciW8vc+eOhrkmk+c2f0cvUUu3Yr8KrS2vMd0j48paz1yVa9lBB7
KPMpDmUdJMbrzbUBZFgS7ccyUjmeUJzkx0SQtSqJUd8czny6Rnx/V1atePpHVT8oCLuJQv/TtgKb
Bz9/Cj5SflE9Y+6YIIxc3cmeSfyKPVOKeJmxeFb0xejuxH/O/2JWRxT1Ld/OW4JuMVp1xb5NyUZnSgvJ7ewKvE
MRqMJm/Xs4lJv38IqgZOHD3Fwx1Y69X39nGOWuqUCn/y81OHYjNEjyM3K4un/vDfw/
OtUmPNy+erd//HSR+MwmUxYrRYYoKIzRUpBf+F5ERERERJwqwNcHAJvNZt+lVEElGKWrHrcGOyH6Xr70+p7H1wpt
wFJVAax59kvfRJxl2ugXwjX9pxluN2PLP/uzsHu5OhfccpOnHz4nHmmslvEs4rkGF05pPJxjN8WbK9S+Hi8/50
yNWs5XYb2Ip/WxpDEYDWAv/np0e51yf4a+VW/yVZJTrh5KMTlS36b3MmTCGsPRmxbqjZ6yfvP+bWwe53mZm6B4yxGwWiCDTuuUhNlFjAmwePeHFf1l0ZVQ66tj+o0d9stGM2GQePnDVTZvJTUKCB
z8QsMpkREKY7u282k9wdxR/NW3HZ3E3ufMf37ElQyjCde/R9u7h6UqVTF4TrevoWbuZw+XpCfz8cv9eTQru38b
8IOrBYLKacSAPDxD8D1792HTvv+80+p07gZgZt1SrAUCVOncw7aN3afpIRxZMn0yVOncU+bMRERERERZFLU7JEMB7
u7uTk5uHlWTiFNNg1/z/OErlO+bjQLbsvAzAaubuHJ+Dk/fGHjPvPlOGA7kB/
DRtJ6ZAE2GPhWFOM+JR2nEKtXOzl7+P2wpsHP3sKDlHcijbryw2q4381MK/KyYfEOYXx2KgUz+fwqemD55lC3c
i9qroRdysOAIbBZK8JBmviuffgOZqU5LxOk2ZMsXZIRRbSjI6OTNvvcd8TNvvcd3Yz7ky6Fvkp6URGDJUO7t2JXHXnjZ3
iclIZ4pHwwmLSmRgJCSNHnoMR59vp/DOIknT2AwXPw6K8nxcWxYWrgpwrpzVvt7nVoGzL1B6rXa2B/f3TfHlYvnM/
IHxfZj9Vv+QA716/h7S4PE1GuPP0+/uxSbltERERERK6CkiWC8PbyJDDPyJDM7+58kozZ/kavog5yH2O5Wlo9Mn+Fly
7pq1OnIstHtxxxiM234uxuoGWrk9ye8uLdUKsSmcvfJbAry/QnJ9+eISzo2w4UrC/NT8snYnAHAwUEHHdqi+kf
hU9XH/j73eC5pG9IcNpHxq+tH1p4sDg07hHuYO6WfK11ON3uJlGSU68lV2fjlxIkT9O/fnwULFpCdnU2FhRWYPn06devWBcBmszFo0CAmTzxdqqpqjRo1mDBhAtWqVbV3keHBq0+RKpLH/oH1/45DmN6NbMX1/1YE3OmSH2DsWmq2MPIjgmJQGpfqxOuUb44PtgWKq1QuPXM+DGbr0sW0WuKOetotXqm7gF09SNd3l1ey4ZGOGd7pkNs/JaTH9ys1L9r6Mq+d9RqfOq+NW2pa2uDiTmWY+KYXNIUbQVDG9uC2WX9hb2pwrw+uW9PpY2jp1+jBBIqCJsTBkeSRc3gIxHGzZfYXnt9ZDvzhiKWhqDHh48cUAgXNQ+ybwzRvP4/3kZzlVN5j4UT2rOZ8rpOudE8XltL0EYRhyKQcg7dY82rHHqmy8HHi4P1X++u1bbDwKobNYe3G5dgtQ1fHA2GJ6Qf9QEqHa1Of4E6Bfx5xUTyXF5gMEagu6eXAbIvqNoFZmpuz1hgUB8Gi/1ZwFbGC7IXQhB4a8nrmeUUkX4b8bKXhJTg3fyA9M2kn7tj5vEBh8vqcFc2fYDt+OiGNHuT0YDHvLIkq5QEvqXA7cAqLuVwkz1q+Izh3IW4MJfgRD2dnmp4mZK9oQ8zR0ijX3fDtUvKoWffljHqCRXZCkx8u4/BLrtmI+EzSLhyc/bm5PzK7WlYqJySdr7Yk7hTJaMiiwXMmavhoQg+u9OP4Xq/OY/5nyWMNkPdHa5nE4n0SJM1OuHlfdi1FqV9Qb0XgN0Mz+Vt9N5X/MkKnVwpp+X56oQEg+WMGCrgvkv+UbKO76sa/TDaYz4qP4p3hdsYi5SyB1p6+hz0LzbJNxeNCjknpijcHUkOe1BWtT/1T51gKhUxPeGJLIbuuYhtr8Bk2HCjmw+8wYdKFpijn5P4xy0kdzGRUOj6wiyYqOQ4nzQxwczTuLm94ETQFJbGfaWXaAswpRb+2ImnexHX5Ql66e6r7UzCQT3s4NWPAYMDCPoxTkFR5I5ETVSJGlGlwNpuzMz5q+cKDxQwoY5WVcXVSQJ/W4wgjmJ4tH/OPKLv3NDvo8XBu4v7R9XMBGT2snKNlOcn+yuKmtfYR9HtO1V61dPNqSP62zmojFQXz0dYlWJDh3qNbtwq9I/tsEX7HASDXjBYyO3d9OU6VF+3KLYhcvNhPnWDpo+MF8Us0XCbsoIVnJWlKp6Ou0nHvmz+F/TSy0ErGH/Ja99Y6HVlNBxS/hw0Vm/MFUTQT2B/0f3ij/vlN8bBQ7UnxBZsvh7NrIBmUPU56T3RfwsFqj31W3tdnTvY1zuDBrwhkd9l5nLlGo9WzcH8aZHdYcNjkVtlTPJc9XtTdV4VpLbZSKgBjtmL+uOKfpVQK+t+LUm7G8g0WuLYeIVmoNSK9mx6bzUvtmj8qCp5zHJRRvGQtd37Grd58zBxtAc0b1/nR2eL4N2BfuGkUjrq/n0nLBm7U2BvMj74aqGNaxnLIkeXbyVW1ESbY3nVGdRDfoT1rZ3UtwRIGQnTUqkDk6tqKj/WG5mCaWp8H1g+i/OhhpeKd5Z51wt+rb8o5Ldk3JUgGSUW3H9oVy2LEHs1WhTB2R1+AWSb8nQ+OtM6FWgaTKvWhp/+5OXLLdbSeLJk

7azw7mqXIwurO+8HlfTVd7ZWuQiFfk2YykpKTRs2BBXV1cWLFjArl27GDlyJIGBgfY+H374IWPGjGHChAmsW7c
Ob29vWrZsSW5urr1Ply5d2LlzJ4sWLeKXX35hxYoV9OrVq6jDFRERERERKTbKRZYi54zPVSYDBKiaUa6y/ZYwm
me/T7TpNmeHQuX8NN4/tYOVR/bwcYKFO3MDcLUVzyRcteBqSjDKdaXIpOuPGDGCyMhIJk+ebD9Wrlw5+9c2m41
PP/2Ut956i4ceegiAadOmERoayrx58+jUqR07d+9m4cKFbNiwwV79OHbsWFq3bs3HH39MRETEWdfNy8sjLy/P/
j49Pb2ob01ERERER0S6Fh4awr8nqwW55pNSoESEXF3ZeNA+6w2GeX1HJ+svGHHeZiintcw6QcusE2QaXJjhX45
ffTyIuYjp1DeKOiXrODsEEQdFXsn4888/U7duXR577DFKlixJ7dq1mThxor09JiaGuLg4WrRoYT/m7+9PvXr1W
LNmDQBr1qwhICDAnmAEaNGiBUajkXXr1p3zusOHD8ff39/+ioyML0pbExERERua6FlgjC1dUFs/mfDV+O+Yt
cS//LfpyBtt7kGa6fXc19bAXOSt3PT8e38+OJZB709CXQ4u3ssK6YkoxyvSnyJOOhQ4fs6yv+/vvvPP/88/Tt2
5epU6cCEBcXBOBoaKjDeaGhofa2uLg4SpYs6dDu4uJCUFCQvc+/vfnmm6Slpdlfx44dK+pbExERERua6FhgT
j7elJVk6O/VgJJRnlGvsurz4dzINJMpZwdihnKZ+fwdBTO1lxdDej4wuonxuA2wO4ndqAgdolazs7jIuybNkyD
AYDqampF9W/SZMm9OvXrOhjmDJlCgEBAUU6ppytyJOMVquVOnXqMGzYMGrXrk2vXr3o2bMnEyZMKOpLOXB3d8f
Pz8/hJSIiIiIicjMJ9PcjON+PzKx/Nrwo6WbGZLA6MSq5GW21lOXenGHsNlVxdijn1Sz7JF/GbuOvw4d5OcWNC
vn+GGw3xiZJ5fzLEeARcMnnde/eHYPBwHPPPXdW24svvojBYKB79+5XHuAZGjRoQGxsLP7+/hfVf+7cubz77mt
FGsP171lITsdeCwWBg3rx5l3ROkScZw8PDqVatmsOxqlWrcvToUQDCwsIAiI+Pd+gTHx9vbwsLCyMhIcGhvaCgg
OTkZHsfERERERERcWQwGLilbGmysv+pZHQxQLib2YlRyc0q2eZD66y3mG9sfh2sOHh+Xlh4OvUAPx7fzvzjSTy
a4UPwdT6duk7o5U+VjoyMZObMmeScUfGcm5vLjBkzKFOmTFGE58DNzY2wsDAMhotL4AYFBeHr61vkcVyv8vOLT
7V5kScZGzZsyN69ex207du3j7Jlyw2KFm8CEhYWxZMkSe3t6ejrr1q2jfv36ANSvX5/U1FSio6PtfZYuXYrVaqV
evXpFHbKIiIiIiEixUTo8FKvFsXKxtHveeXqLXFO2jPTJ7sEH9CCf639actmCTN5J3MWyo7sZF2fm7pwA3G1uz
g7rLHVD6/53p/OoU6cOkZGRzJO7135s7ty5l5Cl1ThtqlHadgROVF8emnnzocu+222xg8eLD9vcFg4KuvvvuLhhx/
Gy8uLihUr8vPPP9vbz1Wlt2rVKh1zrrVKpo0aYKXlxeBgYGObNmSlJQU4OzpOlFRUbz77rs8/vjjjeHt7U6pUKT777DOHm
EaNGkWNGjXw9vYmMjKSF154eXnT
oOIG0tDR7nw0bNnDvvfdSokQJ/P39ueekbnjl7OrRpUwACAwMdqkmbNGlCn
z596NevH4GBgYSGhjJx4kSysrJ46qmn8PX1pUKFCixYsMDhGjt27KFVq1b4+PgQGhpK165dSSoqsrJ46qmn8PX1pUKFCixYsMDhGjt27OD+++/Hx8eHONBQunbtSmJior29SZMm9O3
blzfeeIOgoCDCwsIc/kyjoqIAePjhhzEEYDPb3/6XIk4wvv/wya9euZdiwYRw4cIAZM2bw5Zdf8uKLLwKFD7Nfv
3689957/Pzzz2zfvp1u3brRo/+6tGKlWtlRpvp1u3boRERFBu3btgMLKg1tWtGzZO/Wr1/PqlWr6N27/C5fjS
ElNpVmzZtSuXZuNGzeycOFC4uPj6dChg9P/GPHChg0MMU6dOxdvm3Xr1vHhhx8ydQ+zPdsGEDAJMnTyY2Ntb+/r+4XFS
vS3DHHXfw448/8uabbzJO6FDKlSvHp59+SpcuXex93njjjjDbKysujVqxepqancfffdLFy4EA8PD3uf6dOn07t3b
5o3b47RaKR9+/aMHT4ed+/aMGTOmqMMVEREREpVspFliLA34/UtAxKlggCINC1AB9TAZmWIv8IKHLRVhZUppX1fWZ4fkQ
5y2H78cHLchmy3HFKf+VgI3t6+5xznJOJFgYtyyP6pIUjaTY+ael0v7scd7OO+jSDI2lnT9J+oa4rn7XxBOCV3
3OZssWMt5uBD5p7OKXmP5WW3+/MZ9q2fOY/7oUnFrqlHaRbGhx38WZqQBkWexlINF1apVxRqRJUhSCPoCsa44k
nnuDNN9/kyJEjQGFl4cyZM1m2bNlljde9e3cef/xx26dfn888888888/tx2699
dYLjt+wYUMGDBgAQKVKlVi1ahWffPIp9957LmN5TJs2dZLY/vvfcezz33nM1/stbb73lMMZrr73GzJkzHZKVubm5TJs
2jVKlSgEwduxrRpw8iRIwkLC6NZs2Y0Y3755ZcEBASwfPlyHnjgAfvxwp7ZsuRZG9TUqlXLH
tubb77JBx98QIkSJejzsycAgwYNYnz48WzdupU7bLzQM2aNXn
nnXcAqFixIuPGjSPKNi3UYSu4mVRe1gpHzSdBk2JezaphgYzhNS1A6kDGZqV5s8I5yNzcpGI2GT3QWx0SLFzGDM+Tw
EEWAxN3MxBY7RnKdP+SrHfPIdd47dY9vdIqRoCQkBDatGnDl0ClTsN1stGnThhIlSlyQXTPM4LPDRM+6J3ZlXNnxBdCd8V+
ViumtdlWgWBDZb0aDv2ys+3NAMJNG3NjPw8zG8qnJ0i7dNs4PNcRQGdlrDDLSBYpSpPmfh5aQ5xAcCRm0ixEYYuj
kyqL/a+7aSLDNumd9Fa27SN0Ra7WdhudvGH6XEN/+s5e1g9TG6gc+/CEM5g6gvD8zLG2eWQoC4O1emhd2qCRBzBa
88pzzIPVsB+XxaMbrRMbU6/eozU6qMlijXw2+sWV4I2XAF4kKegDc/5Hg7/2Di++OyweMSyjGO0Pe4eOv+lF7t8
+8MaWBcv1e/ZAkaDffWqgl83m3aBo7B9gQQVXAFWvDLjrXTAHGUt0kOdQpkv2mJQpNZuS2/xQwr6ZuUb26Lfyv+ck
e+o/7mGfvcWxWpm71Dlg0d0zUqTmZtbbl5JU8/LePrrufavR3x9/ZLY18jMyWZEB8bRdrX9Pf+dYoXN/Ks/9OfD
jX9DFYh/Ucx02tU9C2K0HmQnU9sMAw4UvRe/pbpvJC9Nr8MvqbswqJl4OW8aG/FeMU4pKI3h+j1p43fzH9JTbNa
8ks5YrJPH67tBZQfbYmHt7BXKdWpUeDmXmq5KvjtwWjY8QvmE8YyRw4jULz1cTFPB+A4ko5jnx2pZYsFiurrfqZ3sjB
L6oDa0R7UdvOlDzamSJUadfhKmQY3dpvP9sRMnW3y3tTRjovHFdNfOTNSpRIBxAbLmeZSSuE+6N/LjtNJnA1dZ1
ck57+IV65hZbp0JlmQp6+5LFxtzQVJU47h/KP7oLexo5Fp+3enWRtt4A8pbmQVuH/IZNmM5xNlMOM+3oWHeJ0w7
cJqMxUgKjb/sgbyxvxF9+6WaqfG0QDdzZ3j7l/dPhhU93/3xJmIL9P1pgfWWqejPv8qtu3VRYvvSy3u5/Wmffr1X
6XvfR1v6K/RrzuGovGnFC8nnYVgPUuSCV78wp9i

```
SoOcPzGbzRSUZ16xZQ5cuXRgyZAgtW7bE39+fmTNnMnLkyEuK5cknnyQpKYnRo//f3n2HN1W2YQC/M5oO3Xvv0
pbVlkIZBRmFsocIArKXi7OFP2UjYJUhQ4ayRRAZiogiq8jelD1bKKMtOL1Hku8PJBJbdttTTu7fdeWyec/JyZ3
XY6QP7/gGnp6eUCqVCAsLQ36+fhHa1PTFv4uL69sn2x5/xz/u78zMTLRt2xZffvllkWs5Ozs/87pP+3f2olhkJ
CIiIiIiEhlfT3eYqIyRlZODM9NHv2C7KvMggRZaCL+mHBEAfJXTFucVHrDGDFxLzofLrAwYy4EwdxlmNDF+bpH
vReWrtfjhbAFGhil0BZlgRxmWnsxHSo4WMSka5BRoUcFGigNxhTiVoMa3rY2fc9V/KaFB9/QYdE8H4mUmWGXli
Z2mEtwvhenUKrkK1R1efWfpJ7VoOQL5+fmQSCRo3rx5sefY29sjPj5e9zw9PV03tfdVBQUFYffu3Zg8efIL
v+b
IkSNFnleqVAkAcPLkSWgOGsyaNQtS6aN7ZsOGDS+V6dChQ/DO9MRnn32ma3s8lfxJcXFxuHfvnm6/kCNHjkAql
SIgIADAo2nn3377LVq1agUAuH37tt6GKO+jUDzaXOhZ61K+qOrVq2PTpk3w8vKCXP7qZT8jI60XiG78QERE
RERGRsNycHWBva40UtHRdmOKqhYOi7KZOEr2IP/KDsd3xfcx+xwV/9jDBotYqxKZoUX9FFjLyiq6p+Cp+uVyI1
Fwt+lT7d+RW8wpy9AgyQs3vMtHn1xysaq+CqQIY8HsuFrdWYdGJAgQsyES95Vm4cP/FCy306myMS7qE3XEXsSw
+B+HZllBplM9/4QsKdQyFkaxkdumWyWS4dOkSLl68CJlMVuw5jRs3xpo1a7B//36cO3cOvXv3fuq5L+rTTz/F8
ePHMXDgQJw9exaXL1/GokWLnlmMO3jwICIjI3H16lUsXLgQP//8M4YNGwYAqFChAgoKCjB//nzExMRgzZo1WLx
48Utl8vPzQ1Q1xcHNavX48bN25g3rx52LJ1S5HzjI2NObt3bORHR2P//vOYOnQoOnfurFu30M/PD2vPpt5dw9

```

```
1AVCH5P0AxEJYFFRiIiIiIiIgPlaGeLhnVq4EFSst500zIJEGaZJmAyorKxu6AK2uRNwx2pm2AZpK2/BiSSUn+
fqKgoSCQSpKamPvM8Ly8vzJ07t9TzkPiwyEhERERERGTAmjYIg4erM+LuJei1uxvnwV2ZK1AqorJzU2OPiOxpO
CwLLfP31lR5F/AMe6nXLF68GObm5igs/He0cWZmJoyMjNCoUSO9cx8XFm/cuIG6desiPj4elpaWAICVK1fCysr
qdT/CG2HSpEmoVq2aODFEjOVGIiIiIiIiA2ZtaYHWTeojNzcPObn6RcUwyzTIoBUoGVHZyYUCXbNGYoWkIzRlV
CrRyFWQNp/2Oq8LDw9HZmYmTpw4oWvbv38/nJyccPToUeQ+8d/x3r174eHhAV9fXygUCjg50UFSBqMmS4JWq9U
rpFL5xyIjERERERGRgatboxqCKvnh5u17eu0WcjUCuQkMGZDJOR0xUjMcOZIy2ISl4VjAwvmlXxYQEABnZ2dER
UXp2qKiovD222/D29sbR44c0WsPDw/X/fx4unRUVBT69u2LtLQ0SCQSSCQSTJo0Sfe670xs90vXD+bm5vDw8MD
SpUufmalRo0YYPHgwBg8eDEtLS9jZ2WH8+PHQav/9S4o1a9YgNDQU5ubmcHJyQrdu3XD//n29rBKJBH/88Qdq1
KgBpVKJAwcOQKPRYMaMGfD29oZKpUJwcDA2btxY5HW7d+9GaGgoTExMULduXVy5cgXAoxGbkydPRnROtO6zrly
5ElqtFpMmTYKHhweUSiVcXFwwd0jQl/73Qf8q9SLjzJkzIZFIMHz4cF1bbm4uBg0aBFtbW5iZmaFjx45ITEzUe
11cXBxat24NExMTODg4YMyYMaxgExERERERlQIjIznaNW0ElbExklL012KsZpYJU24CQwbkl/xQdMifgvtSx1J
7jwIbf0jDBr3y68PDw7F3717d871796JRo0Zo2LChrj0nJwdHjx7VFRmfVLduXcyd0xcWFhaIj49HfHw8Ro8er
Ts+a9YshIaG4vTp0xg4cCAGDBigK9o9zapVqyCXy3Hs2DF88803mD17Nr7//vt/P3NBAaZOnYro6Gj88ssvuHn
zJvr06VPk0uPGjcPMmTNx6dIlBAUFYcaMGVi9ejUWL16MCxcuYMSIEejRowf27dun97rPPvsMs2bNwokTJyCXy
9GvXz8AQJcuXTBq1ChUqVJF91m7d0mCTZs2Yc6c0ViyZAmuXbuGX375BYGBgc/vfHqqUt2r/fjx4liyZAmCgoL
O2keMGIHff/8dP//8MywLTF48GBO6NABBw8eBACo1Wq0bt0aTk50OHToEOLj49GrVy8YGRlh+vTppRmZiIiIi
IjIIFWs4I0Gtatj+579sLY0h1T6aEyKXKpFbYt07EmxETghUdm5pHZFk5wvsNZkDoLUF0r02hqJHEadVwByxSt
fIzw8HMOHD0dhYSFycnJw+vRpNGzYEAUFBVi8eDEA4PDhw8jLyyu2yKhQKGBpaQmJRAInJ6cix1u1aoWBAwcCA
MaOHYs5c+Zg7969CAgIeGomd3d3zJkxBxKJBAEBATh37hzZk1kZmbCzMx
Md2zKlClo2rQpACAvLw/Tp0/Hrl27EBYWpnvtgQMHsGTJEjRs2FD3ui+++EL3fNy4cWjdujVyc30hUqlgZmYGu
Vyu91nj4uLg50SEiIgIGBkZwcPDA7Vq1Xp0z90zlNpIxszMTHTv3h3fffcdrK2tde1paWlYtmwZZs+ejcaNG6N
GjRpYsWIFDh06pBvS+9dff+HixYv44YcfUK1aNbRs2RJTp07FwoULkZ+fX1qRiYiIiIiIDJZEIkHL8Lfg7GiPO
/H39Y75qHLhaZwjUDIiYWRoTdAu6zNslLaEBiW3jmHhW6MAp6qvdY1GjRohKysLx48fx/79+/79++Hv7w9e3s0bNh
Qty5jVFQUfHx84OHh8dLXf3Kw2ONC5JNTm4tTp04dvfUew8SdPnkTbtm3h4eEBc3NzzXUEwLi507
zqhof9uwHP9+nVkZ2ejad0mMDMz0z1Wr16NGzduPDWzs/0jaejPytypUyfk5OTAx8cHH3zwAbZs2cIZtK+p1Iq
MgwYNQuvWrREREaeHXfvLkSRQUF0i1V6xYER4eHjh8+DCAR9X2wMBA0Dr+0zS5efPmSE9Px4ULxf8tFNQl5eHtLTO
/UeRERERERE9OLsba3RqnF9ZGRlITdPf4BHfatUmEjVAiUjEs7o7J6YhI+Rj1cfefhYtnVFKMLHvvZ1KlSoADc
3N+zduxd79+7VFexcXFzg7u6OQ4c0OYe/evWjcuPErXd/c4cWW7VlZaN68OSwsLLB27Voc+34cW34cW
7ZsAYAig8lMTU11P2dmPlo7oT9vfff8eZM2d0j4sXL+qty/KL3mHhUJZE2Fnn3CH2cLEy+3rFNSm8EFTQhHbLSn5
7ZsAYAig8lMTU11P2dmPloT9vfff8eZM2d0j4sXL+qty/jfzI+Lnc/K707ujitXruDbb7+FSqXCwIED0aBBAxQ
UFLzy5zR0pVJkXL9+PU6d0oUZM2YU0ZaQkACFQlFkm3Z2YU0ZaQkACFQlFkm3RHR0ckJCToznmywPj4+ONjxZkxYwYsLS11D3d39xL4J
ERERERERIalQa3qqOLvi5u37+q1G0u1aGidAnC3aTJAq3Pro1vhBKRIX33ZgEKJEZRd1wBSWYlkCg8PR1RUFKK
iotCoUSNde4MGDfDHH3/g2LFjxU6VfkyhUOhhGGZaEo0eP6j0/cuQI/Pz8IJPJcPnyZSQlJWHmzJmoX78+Klas+
NyRkQBQuXJlKJVKxMXFoUKFCnqP4kYpPo+xsTHUanWRdplMhmPHjmHixInYt28fpkyZAmdnZ6xbt+6Fr01Pvz6
vFKwSNKQDr+kqz79aRWzJ5zNTeuJgNGp76qnYY4mTXgYHaWS7uVOrfmoStx/DvE5+h9ivmoZhOv/xZDsnMZvncC4aR2tCh8AQE6LLimJyamI/7EZEN6GuoiUoRGp4+nuqqq9+FHkKbeknwpHJo
```

sxOIfNmL/sVOo7OcLmezfsSq5Ggm23HdAlqZkdsslelMpkY8Vpt+irvqYri3VxAuWI49BIueMTDIMXECDiIiIi
IiInkoul6NL2+bwdHVG7O27eseMpVoOtkmGDKIcuOLOwvKgQLes4fhO8i7UWgmyJSZQ9vqZBUYyKCwyEhERERE
ROTPZ21qjc9vmkEokSEpJ1TvmqCjAW1apxb6OyNB8kdMBQ/MHIyX8K6ic/IWOQ1SmWGQkIiIiIiKi56oRWAnNG
tRBfOJD5OXn6x3zM8lBsFmGQMmIyhf3ymFwbdBD6BhEZY5FRiIiIiIiInouiUSCtkOboWrFCrgeexsajUbveKh
5Bjy5EQwZuCb2mRjTrYXQMYgEwSIjERERERERvRBTExV6v9sWLo72uHHrDp7cR1QiARpZpcLWKP8ZVyASrwBlK
r75sCVkMm6ERIaJRUYiIiIiIiJ6Ye4uTuj1blsYKxW4l/hA75iRVIvmNskwkxUKlI5IGGaFqZjfoxbMzC2EjkI
kGBYZiYiIiIiI6KUEV/ZHp9ZNkZmVjeTUNL1jJjINWtgkQynRPOXVROKiOubhmy7V4O/nJ3QUIkGxyEhERERER
EQvrXG9WmgZXg/3Eh8gMztb75iVUSGa2iRDBu1TXkOkDkpNHiY1dkSTmlWEjkIkOBYZiYiIiIiI6KVJpVJObBm
B+rWq42bc3SI7Tjsp8xFunQIJC4OkUkaaPAyurkSnpmFCRyEqF1hkJCIiIiIioleiVCrQq2MbhFSthGsxt1BYq
L8Wo5cqFO2sUyBloZFExkiTj56+BfjwnaaQSllaIQJYZCQiIiIiIqLXYG5min5d2sPfxxNXbtyCRqO/FqOXKhc
RnDpNIiLXFKCTRzZGdG8DpVIhdByicoNFRiIiIiIiInot9rbWeL9rB7g6OeBabByOWv2CoodxHguNJAoybQHec
c3E2N5vw9zMVOg4ROUKi4xERERERETO2jxcndH/vfawtbbC1ZhbRQqN7sZ5aGabBDl3naY3lFRbiHaO6fhfn3a
wNDcTOg5RucMiIxERERERERZWISn4++Kh7R9jZWOHKjZtFpk67KvPR3CYZRiwOOhtGqlWjlVOqPu/7NqwtLYSOQ
1QuschIREREREREJaaSnw8+7tEJjna2xRYanVlopDeMkSYPreOeYGK/drC1thQ6DlG5xSIjERERERERlagAXy8
M6NUJrk4OxRYanZT5aGmbBAULjVTOqdRZaGf3EJ/1aQ97W2uh4xCVaywyEhERERERUYmr4OWBj3tOgpuTIy5fv
wm1Wr+g6KAoQGu7hzCVFQqUkOjZLAtTONbuAUb1fgdO9rZCxyEq91hkJCIiIiIiolLh4+mGAbO6w8PVCVduxEK
tVusdtzUqRHu7h3BU5AmUkKh49gUJaGmXjKG93oWLo73QcYjeCCwyEhERERERUanxcnfBwF6d4eXugsvXb6LwP
4VGlUyDVrZJ8DfJEighkT633FiO99RiWL9ucHN2FDoOORuDRUYiIiIiIiIqVR6uzhjYqwt8vdxw+Xos8vLz9Y7
LJEADqzTUsUiDBFqBUpKhkOID35zL6Bxkg8F934Ozg53QkYjeKCwyEhERERERUalzc3bEoN7vIbiSP67G3EJGZ
tGRi1XNstDCNglKbghDZcwIhaiSex59GgTgw+4dYWluJnQkojcOi4xERERERERUJpwd7DC4z3toUq827sQn4v7
DpCLnuCrzOc7+AazkBQIkJENkoc1C9YLz+KhNXXRr3xLGSqXQkYjeSCwyEhERERERUZkxNzNFvy5voOvb5sjMy
kFs3B1otfpTpC3larSzewh3Za5AKckwaOFWeBdhOmsY+l4rtG7SADKZTOhQRG8sFhmJiIiIiIioTMnlcrRt2hA
f9XgXpiYqXL4ei8LCQr1zFFItmtkkI9Q8HVKuOOglTCVVo3LOedSxzMCQPl1Qr2YIJBKJOLGI3mhyoQMQERERE
RGR4ZFIJKgdEgg7Gyus+nkrLl2Pha+nOOxUxk+cA1Qzz4S7cS72pVgjudBIwMQkFs7yLDgnRyPQ1w3932sPD1d
noSMRiQJHMhIREREREZFgfD3dMax/D4RVDOLMrTtISUsvco6tUSHetn+AYLMM7j5NrOwGLSriNnzSz6BZWDCG9
e/GAiNRCeJIRiIiIiIiIhKUrbUlPu7ZCXY21tix7xDSMjLh4eIEqfTfcTEyCVDTIgMe/4xqTFfz1ll6cZayAnh
lnIebuRQdOr+NRmGhXH+RqITxW5mIiIiIiIgEZ6xUolv7lvBwdcLmP/bg4tUYeHu4wtREpXeeo6IAHewf4FiGO
S5mmQLgOnrOLFp4y1Nhl3wewRV9OO3tVvDxdBM6FJEoschIREREREREE5YJUKkWD2jVQwcsDP2/7C8fOnIe5mSl
cnRzONuWQS7Woa5kOT+Nc/J1qhSyOaqRiWMvz4Z17DRb5GWjatD7aN29cpGhNRCWHazISERERERFRueLiaI+Bv
bqgT+e3YSSX4eLVG8jJzS1ynqsyHx3tH6CiSRbXaiQduUSDaqok+CcfgY+lDAN7dka39q1YYCQqZfzrHiIiIiI
iIip3jIzkaFq/DgJ8PLHht79w8vwl2FhawMnBTm9UoOKqxVtWaahomoUjaZZIyFcKmJqE5qnMgU9BLHIfJKF29
UBOadcCzg52QsciMggcyUhERERERETlloerM4b264aeHVpDo9HgOrUY5OXnFznPzqgQbeySEG6dDDNZoQBJSUj
W8gIOMrOLl6TjUCEfPd5phYG9urDASFSGOJKRiIiIiIiIyjWFwgitGtdHgK8Xftq6A2cvXYOtjRWc7G31RjUCg
K8qF57GuTiXaYazmWYoOHJsjZgpJRqEmKVBlXINhekFqBMShPbNw+Hp5iJONCKDU+LftjNmzEDNmjVhbm4OBwc
HtG/fHleuXNE7Jzc3F4MGDYKtrS3MzMzQsWNHJCYm6pOTFxeH1q1bw8TEBA4ODhgzZgwKC/m3UUREREREIbK1
9MdIz7oga7tWOCr1eLClRtIS88ocp5cAoSYZ6KLw31UMc2ElOs1io6RRIMgOwwOVlwF7p6Fg4O1BvTqjMF93mO
BkUggJV5k3LdvHwYNGoQjR45g586dKCgoQLNmzZCVlaU7Z8SIEfjtt9/w888b846dOig065Wq9G6d
Wvk5+fjOKFDWLVqFVauXIkJEyaUdFwiIiIiIiJ6g6iMjfF2s3D8b3B/hNcNxcPkVFy5HovcvKJTqI1lGoRZpuN
dh/vwUWUDLDa+8ZQSDaqbp6Ot+U2YJp6BOjcL7Zo1wrhB/VA3tBpkMpnQEYkMlkSr1Zbqt+yDBw/g4OCAffv2
UGDBkhLS4O9vT1+/PFHvPvuuuwCAy5crKSjh8+DDqVFQcM+aeEd3d0daWlpsLCwKOwLCwKM2PCADYfCW+1N9DTDoEOAsdgYiIiF5Reno6LCty+zPWURET6PVa
nH2OjX8tmsfLl67AZWxdHSGXF78yWFqhDOcyzXAtRwU1p1G/UYylagSaZsFPkYZ78XehVmtQI7AS2jZtCF9
Pd6HjERHKYOOXtLQ0AICNjQ0A4OTJkygoKEBERITunIoVK8LDwwOHDx8GABw+fBiBgYG6AiMANG/eHOnp6bhw4
UKx7zNjxgxYWlrqHu7u/u/JIhIiIiISM4lEguDK/hjzUW980KBJpqBD/j7KytcPn6TdxNuA+NRlPkfEu5Gm9ZpaGrw31
UNO+HsVQtQGp6GSZSNepYpKGj7T1YpV1F7M1YuDs7YUif9zCkb1cWGInKkVLd+EWjOWD48OGoV68eqlatCgBISRERERRGRuCmVCoTXrYkagZXw99GT2Ln
/CC5euQEHe1vY21oX2RzGWKZBdfNMBJll4nq2Cc5lmSKtOEig9FQcM1khgsOy4WWUhoSE+4jJyYG7ixM6tWmGe
qHVYGqiEjoiEf1HqRYZBwOahPPnz+PAgQMOREREREROWThbkZ2kQORO2QIOw6cAR/Hz2
FC5evw9raEk72tkXW65NLgIqm2QgwyUZcnhLnMs2QkM/fK4UigRZuyjwEmGTDUZqOewmJiM3Lh5ebCyLq1OGta

lVhZmoidEwieopSKzIOHjwY27Ztw99//w03Nzddu5OTE/Lz85Gamqo3mjExMRFOTk66c44dO6Z3vce7Tz8+h4i
IiIiIiKg49rbW6Pp2SzSsE4ojp85i/7FTuHw9FiYmKrg42kP5n3X+JRLA0zgPnsZ5eJBvhMvZJriZoOIe12OsE
1byAvibZKOCKgfSwhzcvpeIm4WF8PFwQ9MGdVAzuApUxsZCxySi5yjxIqNWq8WQIUOwZcsWREVFwdvbW+94jRo
1YGRkhN27d6Njx44AgCtXriAuLg5hYWEAgLCwMHzxxRe4f/8+HBwcAAA7d+6EhYUFKleuXNKRiYiIiIiISIRcH
O3RoWUTNK5XC8ejL2DvoWOIuXUHMpkMrk4OxU65tVcUwF6RhrqWabiTa4wbOSrE5SlRyIJjiTKSaOCjykGASTY
cFAXIzsnFnVuJOGg08PP2QNP6dVAjsDKUyqIbvxJR+VTiu0sPHDgQP/74I3799VcEBATo2iOtLaFSPoCHzBgA
LZv346VK1fCwsICQ4YMAQAcOnQIAKBWq1GtWjW4uLggMjISCQkj6NmzJ95//31Mnz79XB3aSI
iojcXd5cmojdVbl4eTp+/jKjDJ3D5xk0UFhbCycEOVhbmRdZtfFKBRoJb/xQc7+QpocXTz6Vn0cJZkQ9/k2x4G
+dColUjKSUVD5NTIJfLEeDjhYj6tRFSpSIUCq6RSfSmKfEi49O+mFesWIE+ffoAAHJzczFq1CisW7cOeXl5aN6
8Ob799lu9qdC3bt3CgAEDEBUVBVNTU/Tu3RszZ86EXP5igy9ZZCzfWGQkIiJ6c7HISERvOrVajYvXYvD30VM4c
+EyMjKzYW1lATsbqyJTqf8rVyNBbI4KN3JUSMhXACw4PpNCooGrMg9uyjy4GefCRKpGZlY2Eh4kIb+gADZWFqh
RtRKqB1ZG1QDfF/6dn4jKnxIvMpYXLDKWbywyEhERvblYZCQisdBqtbh5+x6OnTmH49EXkfjgIdQaDWysLGFrY
wWj5xS8stVS3MtTIj5fgfg8JdLVLJABWtgaFcBNmQd3ZR4cFPmQSoD8ggLcf5iM1PQMmKqMUchLA3WqByGokj9
srS2FDk1EJYDfgERERERGSQJBIJvD1c4e3hirebh+Py9ZuIvngFp85ewXOACArYOVbK2sIJMVXZPRRKZBB
ZMcVDDJAQBkFspw75+CY3y+ApkGUnRUPh6taJwLN2UeTGQaAIBGoOFKajruPOyGBICTgzOi3qqNkKoV4e3uCqm
U61wSiQlHMpYQjmR8ORzJSERE9ObiSEYiEruMzCxcuhaDMxevIPrSVaSkpkMmk8LayhLmpsUL2Wn1Qr3Ol3W0Mwx+ZezPDGZnsrGiJm93Q0d4
/kK3M9XIEMtw5s+vVoKLWyMCmBnVAB7RT7sjApgLS+E9J+PlZuXh5TUdKSmZ0CjlcLK3AyBFf1Qs1pVVPH3hYm
KuOQTiZVh/LUKERERERERERQsyNzNFrZBA1AoJRHJqGi5ejcHJcxdx+XosLj14CMk/51hamMPc1OSpexNYyNWwk
GcjwDQbAJCvkSClUI7kAiMkFxghpVCO1EI5cjWyMvx0L04lVcNaXggbowLY/FNMtDEqgOyJj6vRaJCWnomk/DSS
k5OZBYSSHrbUVmtSrhQBfL1Tw9oCjna1wH4KIygyLjERERERERERERPYWNlibdqhaBezWq4n5SMmFt3cOPWHVy8e
gMPklNw514CAMDC3AyW5mYwe0bRUSHVwlFRAEdFgV57vkaCCtEK57pGlliFXKOGeRopcjRR5/zxKbldrLYylGph
INTCRqR89/vnZ9J9/msnUUPOz7VnvlVotcnIfj1bMBKCFhbkZKni5I6iyP3w93ODt4QpjpbKEshLRm4JFRiIiiI
iIiIqLnkEgkcLSzhaOdLcJqBEOtViA+kP+/kPcunPvUdHxWgweJqcg7l4CJAAsLcxhZmoCU5Xxc3dMVki1sFcUwP4
/xccnabVA/n8Kj7kaKQq1EkgBSCSPSpB6P/+nTS7R6oqJshesV+bl5yMzKxsZWdnIzs6BFoBSYQQ7ayvUqhaIA
F9P+Hi6wcHW5qnFVSIyDCwyEhERERERERERERERERERERERERERERERERERERERERERERERERERbOkmUwGN2dHuDk7ol7NEBQWFuJe4gPcuhOP6zdv49L1GKSlZyA+8QH
UGg0kAFQqY5iqjKEyNobKWPnc4uOTJBJAdAuoS/zxqtQbZOTnIyslBdnYucnLzAGhhZCSHuakp3JOd4
eftATdnRzg72MPDxQlKpaLEcxDRm4tFRiIiIiIiIqLXJJfL4eHqDA9XZ9SvXR1RqtRoPU1Jx/2Ey7j9MRuLDJMT
evovEB8lITk1DTm6ervgol8thJJfDyEgOuVwGI7kccvm/PxvJ5a88SlCj0SC/oBD5BQXIz89Hfn4B8goKkJ9fA
LX632KlRCKBicoYZiYqVKzgDU83539GbtrA2dEeluZmHKlIRM/EIiMRERERFRCZPJZLrp1U/Ky8tHUmoakh8
/UtKQ8CAJqRkZyMjMQlZ2DgoKCpGdk4vCwkIUFKpRqFIOdHzuQnWXJhgegKFEAkfVzUYKRI6VQEIP9mFfjHbBF8qmFV8ZyM+MolEElOxCNkDuojm6Wwlh
QwkYDw3QjktaR48SvVbHO4+sWsMKKMZr1IWAZJNCZcZE/ScdRYNXd0o0JHuSaUDq0jb3r8i2YNsYMK/vVGGyJdbLoLwL77Nrnsz7ycW1GmuUaJu74Plbk+aN0tVWT5cs1qXFm4NN/5OPlBG/E1xLLzJThtuMEjNb3ANNg5HwbHhoTdR1eLmUKEzGFNCzqy2g6r0+hFkpb2YI5ZzGPiQLbLZy/Z1OdXSt2Rz9+eeBNMFBNHCImkmO5XY5PvMvJUa5AKYiUHN5rSbzPBVPDd9TU8nZHhX/Lxz60s2zUshwjAVx0ZCxqNJxp2kEdr4JX4JmbNiqxKJVm/rcdTqeDlDllqW8t9tiPoS26s7gLLpu8RkNtTKQ1sbjkb/Fty2DfBRUN3pzyIZJD12lRMd9LmgJ0kx2WXMjwuC7p0Tg9ztVw9CxkyUGMJYCnOs0i4yH+jNeh8FPJgXbIKaRA6tDvnhcRHpkqKjJjDhpcIJWQz09tWZ5jxGfKyoNdyeBzpYo3jLEPKf7LTYsxhsMyHcrsWDxrkz4h6gLOc3+oBHXRDLJn1R9aNKJcCMnfz4sRahvE1ojrBAZiOMSO62V+1jqbZ2e0Z4qvH0rRvFcsxFYoKe0Rsfbwn/+Ai4WYsyZSM7hZDxhOB8ziG/bnvBI

7cgCT+nTGwr8OwcnDq9hzJvbuhNa9+uPB3Ts4GbUL/1uyBovGj4G5lTX6jJv4Up+7JPv9ZTfcSUtLQ+fOnVGzZ
k1O6dIF1tbWiIuLg7u7O9zd3aHVatGjRw/I5XKMGTMGpqamWL16NQ4ePIhffvkFJiYmxV73999/x4QJEzBlyhR
Uq1YNt27dwueff44WLVrgkO8+AQDs378fpO+fRpUqVTB8+HDMnTsXTZoOV3jypUr6NGjBxYsWACtVovBgwfjx
x9/hL+/PwoLC9G1a1dMnDjxtTbO4aY7RCQmZf3nLCIiIiIq/ziSkQySk4cXpv6wGWtPXcfSvSfw5c/bUVhYAEd
3T9O5xiYmcPbOhn+1GhjOxWxI5XLs3rjuude+f/cOzh3ej4h03Z57rl9QdQBA/K2bxR7fs2k9TCOsUKtJC1w4f
hi1IlpAbmSEui3a4MKxQy/2YcuJ5cuXw8nJCdOmTUNgYCDc3NxQt25duLu7AwBu3bqFs2fPYvz48ahatSq8vbO
xfvx45OXl4Y8//njqdc+cOYQkBCObtOarq6uqFu3Llq2bKlXBK1fvz6GDh2qV1h8UmxsLPz9/VG7dm3UqVMH/
v7+iI2NBQCsXLkSNWrUeKOLhGq1GuPHj4e3tzdUKhV8fXOxdepUPPl3TFqtFhMmTICzszNUKhUiIiKeO4oOIyM
Dw4cPh6enJ1QqFerWrYvjx4/rndOnTx9IJBK9R4sWLLXTH8/Ly0OXNnT1hYWMDf3x+7du3Se/1XX32FIUOGgIiIi
IiIiMo3FhnJoBmbmMDawRGZaak4c2AfajZu/tRztRoNCvLznnvNvZw59tq33Ju8XHxXCrBOcihxLSO7
Cz9/OQf/PpwEANGq1brpuYWEBNBp1kdeUZ1FRUahcuTJGjhyJhg0bol0nTti4caPueH5+PgBA+cRamVKpFEZGR
jh16tRTr1utWjVcvHgR586dAwDcvn0b+/fvR/369V84m7+/P27evIn4+Hjcu3cPN2/ehJ+fH27fvo1ffvnljS9
yffnlll1i0aBEWLFiAS5cu4csvv0RkZCTmz5+v0ycyMhLz5s3D4sWLcfToUZiamqJ58+Z6087/6/3/3338fOnTuxZ
s0anDt3Ds2aNUNERATu3r2rd16LFiOQHx+ve6xbx92+xfunSpTh58iQOHz6MDz/8EN26ddMVP2NjY/Hdd9/hiy+
+KOEeKTteXl5FiqwSiQSDBg3CzZs3izOmkUjw888/P/WaT3vNV199pTvn6tWretWrePvtt2FnZwcwLCwu89dZb2Lt3r
+54cnIy2rZtCzMzM4SEhOD06dN67zF4YQERERERE9C4uMZJBO74/C6f17kXgnDtEH92Fi73fh6lMBjTt
OQW52NtbOnoGrZO7i/tO7uHH+LBb+bwSSExMQ1qKt7hqT+nTG9h+J8jkcr1jCXE38fO3c
3Dj/Fncv3Mbx/fswLyxw1A5tA68AopuKLNi+gSO7fsRbBOfTXGuWLOm/v51I+7cuIadG9aiYvWaJd8xpej0nTv
YsGEDPDO9sXjxYnTu3BkzzZ87Er7/+CgDw9vaGs7Mz5s6di7di7SONBQUFGDZsmVITEzEw4cPn3rd1q1bY9CgQejVq
xdCQkLQqlUr1KxZEx988MELZ/Px8cgWYcPw4YcPw4CPh4+ODyZMnR8ICDeeecddOrUSW8NyTf
FoUOH8Pbbb6N169bw8vLCu+++i2bNmuHYsUcbHWm1WsydOxcjf/w23+9esetXAkBOTg42b
dqEyMhINGjQABUqVMCkSZNQoUIFLFq0OSO9cpVIJJycn3cPa2lp37NKlS2jXrh2qVKmCQYMG4cGDB7p/3wMC
XX375Rk/FPH78uF6Bdef0nQCATp06wd3dXe9YfHw8Jk+eDDMzM7Rs2fKp1/zfzmnTpgOKC
wuxZ88enDx5EsHBwWjTpg0SEhIAAF988QUyMjJw6tQpNGrUCCNHjgCMhJjI4XwpJ3EA7iB5Xg6Xk4ZZ3iXVJrVH4mBw3JD7INAAjNUnVMIyEuDNmo+lVKRBUmbNwEudJfl/9MVp1Pa2XVPPJmqyUP3sG5lPMQUMLMAGJZNPzrWLKL+dmIZB2qgUM
Q52+qnRiFb+C1TSBZ4Hj+dKCQQmjpvpy0ZJ+Yst5G+ZTydAqzLfGRr4j+RILUzmVIPjbEqKWX3cTUOfEjo1q6SKfzfGoqQxe5NjSKfpq6C7RhOHJ5rfDyRqu+RJdcWz+pZqLbMsBGgR2TNsEmtQt0c0R3Hvwh+qbM5BjdYbhFTCb4SwQT0SLG2FJpTt/NTdCQqgwvSFEjwhCbSZsgDUozIV4Qa7IRr0YbXwu4N8rmC9ZGlUHjCQDzc4/c2snXJh4hHKvsbfW2hFVqWrAQqrgFNdHGzkAqhwFakmnVD66yaakqKKhnowpC2nEfDt9I0+A94+p7c6ZmT4BNmvJh+U0rQsbbeucdMxsbtWH34NPd4G7+5eqWAYvGBACnAwHKYzZTv8dDM4s+hBnHDXwR0l8bSrTeUu3qRZhIZzg73HSpd2g/rQrYBk9TJYeHGqCrwxGGBlw8GYxdbKvaAtd7qBKwVmLkgJlyMGHVr3lrqyFqEc1dR+5I0zkcMiA7v9JGtr/+0H8tOSHAhfdBsExZTm5zfzqvFJ5QjqNKJzKwcjK08sSC8TVUCsW4i0lqJ1uoyq1X+LoJ3URtmMcd1RA5qVpv0W/8r3sxN9BqwfJ1O6lm8+X3/yYjDZXL1a+6AeOUBpk+BfB26+Q5TfSyrk7ReezIFnm5z/OW5s3c0ZB6EPkF8t8aO9pdH0qzq7GtICKdpeYZfk1xYpfx5T5pbJfNsfVgjhRILJsE0DmnyOAbAR1dlMNpvaQcw4xV/bbnbRgGpP9Kd6XvPFRbGdXNVmfPd5J4b4TQ7LmHR6+ejZUZ4L6cb2XQrAe6YUW1N1jUd8K8VMVlt/MqIjh8jmgbUUOoTPMzx3O6HLTA9CHuPkPKTvVqIdpmTIhIRXYqzZ/ZzsIFH6TvFHr7zUcLOFnjBvm9KhOSHIEr1BWxqyOVhI2e9HvPJmLSnnaQ5Fb/kvfL2fGGkfHDKvkyW
bhw8fxnvvFb2fCwsLoVarYWxsrNeuUqlw4MABvbaoqCg4ODjA2toajRs3xrRp02Br+6j/g4ODsWbNGuTk5GDHj
h1wdnaGnZ0d1q5dC2NjY7zzzjsl3R1lyt7eXu/5zJkzvri4YNG0IikcDJyUnv+JYtW9C5c2eYmZk99Zr/fc2
vv/6K8PBw+Pj4AAAAePnyIa9euYdmyZQgKCtK977fffovz58vvvQd/f398+OGWLp0KQCgoKAAH
3/8Mb7//nvIZLLX/vzlRUpKCurVq4fw8HD88ccfsLe3x7Vr1/QK1dhD7rxnJlj9YxTxJnzFd+h/+0xT2jGsjTV2yV/Y2X1Ig7oEYIx4Bh7+UCfX2V1Z1E0nnPAyLVXIk6b3OPFJlXzXjzkk9PP
GjhyJxYsXo3bt2pg7dy6aN2+OK1euwMHBAb/99ht+/PFH/PXXX7h27Rr+++239969/zfj5/PdKC5Xl6GORiRG7ZzFaVZqA8xlEsH1dfQr16ZrLTdNoMmd8+DfKYfO8pVHyRvK2Z3LGW5AA9euDTIZxBsfyHX+tP0mXgUg12mB2lsjEGHENxPKCMaXKlZMP9Sm5aCBu8aKxO3SwdgjzX4BFi1lk4FwtYOfJjS2+Gf2wV8ejjy7j6Z5SyBvmS3C8ZiO5MBU0mcDRnslhbTO7V
ERERPSmYpGRDFK9lu1Qr2W7Yo8plMb4ZP6y5l5j8Z5jRdqqvdUImy7fK/Z8O2dXTP1h8wvlC6nfCCH1G+m1KVU
mGD13yQu9vjyysneAWwV/vTZXXz8c+Wu77rlv1SDM+mUXsjLSUVhQAEsbW4zr3Bq+VYOeet318yLRoF1HRHTqD
gDwDKiE3JxsLJ4wBhO/Hgap9OUHbKenJGHDwtmY9sNmXDt7Ci5ePnDx9IBy7/bRYzpUR7ZZ+TYJfv2GJt3sWeK7yrVPZ6OsHfOA3QoAdXmLfXO4lfutOz/CHxeD5LlSmXxOCDkH4X5GH4C6YA5xd9Ah4m8sA3ZK85PDCK9gTxyKf9KOMPyHRLXWpfA3jGj0dF0FmIQwNgGLdoNUQ+BnJCTymS9HmQyAs09LKADaFqz1unmLWfjTIjLOEsRdR8Zz0bTvbPMmIHoG5MmIKsxkpH7TE7vDq9LkP8zNEGn6rovqkZWmtjcMtI6cFkBgeejrPHkd1Ya7Hxd7
gDwDKiE3JxsLJ4wBhO/Hgap9OUHbKenJGHDwtmY9sNmXDt7Ci5ePrqHurAA92Jj4BlQ6aWvK5Rx48YhPTOdFSt
WhEwmg1qtxhdffIHu3R/12ePRbY6Ojnqvc3RO1B37L3Nzc4SFhWHq1KmoVKkSHBOdsW7dOhw+fBgVKlTQndeiR
QtO6NAB3t7euHHjBv73v/+hZcuWOHz4MGQyrfr164ezbKsLOzw4YNG5CSkoIJEyYgKioKn3/+OdavX
fX18sX74crq6upaPvRLpS8/Px8//PADRo4cLp+CYlEUuT4yZZMncebMGGSxcuPCFr5mYmIjff/8dq1at0rXZ2toiICAA
1evRvXq1aFUKrFkyRI4ODigRo0aAB4Vd/fs2s2YP3338f03bs0BUjIyMjOahRI4SGhr7mpy7fvvzS7i7u2PFiFihW
6Nm9vb93P/x3NCwCrV6+Go6MjbnlXz8XhL3vbhVVJ3gs9XOgh3yjUFeJyrMpOb1rxkzGDM9wHzd5vMKqKFbzFf3oWLmLOKxWGIOWDJfL0Ky3EEQoK9yI1Lnl5ODRkm3V1IgdO2zW6eB6SwmrXBIMiYrA5sG4zLrDd5ymBSMYn5zbYx5avFxOuEoC1vzhrwHpQl6w4vZWHyQmmXDjphBHV5k1MzCgDjRYXZ8N7TnhEpXm3M+hT8CGXLJpfKCrRSxGeU2ZTrwqZbUgY9eMs3FHGGVBtGEGhMxdNVkCmYlW+4dL6vGkSG
6Nm9vb93P/x3NCwCrV6+Go6MjfvnllI2IL7QAwe/ZsfPDBB+jbty8AYPHixfj999+xfPlyjBs3DpcuXdL1Z2hoK
IYPH47Y2FjY2dnhk08+wYABA+Dh4VGKn5yIiIiIqOxwujQRlYmKITVxL/aGXGXlv8zRjYuxQtGpmaW8DSxhb3bsb
gxvnoZ66VmZeTU6SQ+Pj5kxubvIwVMyahbe8PYOvkAo1aA3Vhge6YWq1+49bD3LBhA9auXYsff/wRpO6dwqpVq
/D1ll/rFaZexZo1a6DVauHqlUol58+aha9euev8+3nvvPb0L3n6atlmG1mFo0lURL1DSiBIvHLHHcN/6j6HaMKSJsf8D7MrXbiT0pGzqRCbe4dHKgMlkD0sQcW9o/yV0mVvs4DYXtlF5t7nzRbbYSNbW5iPHRFyV94YZv6lnfOBB6hxjxPtl5nz5P
LFyI2NhbHjx/HW2+9hVGjRmHoOKE4ffo0fvnlF0RHR6NOnToYaO+UV2i+//ILU1FT06d2On2OPLli1DpUqVU
Ldu3Re+5qpVq2Bubo4uXbrgwoUL2DtUSCXbt2oXTp0/D3NwcxsbGmD17l7Nv78800/dyL1x48ZBLpfD19cwW7ZswbJly3D
t2jWsWrUK48ePx8cffwfHx907twZaW1pr/W5y4OtW7ciNDQUnTp1gooODA0AJCQvDdd9/pjj9vNG9x8vPzcfLkS

```
b3XSKVSRERE6F4THByMEydOICUlBSdPnkROTg4qVKiAAwcO4NSpU2/8PU1ERERE9CSOZCSiMtG2z4f4X9d22LR
4Huq2bIvrZO9j54Yf8PGUfzerOPTnb7CwtoWdiyvirl7C8i8moGaTFqj2ViPdOfPGDoWNgxN6jPofACAOvCl+W
7kU3pWqwi+4OhJuxWL9vK8QGt5UN9OzJysLCXGxumvcv3MbsZfOw8zSCvYubno5ow/uQ/zNGAyZ+QOAoEJgMO7
G3MCpv/fgYfw9SKVSuHj7llY3vZAnd85+EcOHDOf//v1lO2SHhISgW7dumDRpEmrUqKErIhO8eBAVK1bUvS4mJ
gYVK1Z85vstXLgQ2dnZyMrKgr29PUaPHg17e/tnvsba2hr79uOrMnISAI4dO4YTJO5g+PDhmDVrFmrVqoXY2Fi
EhITgm2++eenPXp52BV+2bBlatmwJFxeXIsdycnLw448/Yvz48S91zeXLl6N79+5603m1WiOGDRoEBwcH7N+/H
yqVCt9//z3atm2L48ePw9nZGZaWlvjxxx/1rtW4cWN89dVXLWt2LWJiYnDlyhV88MEHmDJlyhu/CUxMTAwWLVq
EkSNH4n//+x+OHz+OoUOHQqFQKoHfv3q8Omvfhw4dQq9XFvuby5csAgObNm6NHjx6oWbHmIL9SbMm6NHjx6oWbMmVCoVVq1aBVNTUwwYM
AArV67EokWLMH/+fNjZ2WHpOqWoUqVKKXx6IiIiIqKywSIjEZWJCoHV8Mn8ZVg7ewZ+/nYOHNzcOffTKWjQ9t8
RWCn3E7Fy5iSkJT2ElbODGr3dqcialQ/v3YVE8u9IuXcHDIdEIsG6byKRnJgACxsbhIY3Rbfh43Tn3DgfjYm93
9U9XzlzEgCgUfvOGDJzrq49LzcH3O/9DCPnLNaNxrN1ckH/z6di4f9GQK5QYMjMb6AOVpVgz7y8l1OPMys3Hw+
M9NehTDEyR55WiqtGttB628DK3gHbjp+DNLAeACA7MwNnz51Hg279n/9+lraAJXA7LRX7Dx1Gz9GfP/U1SQn3k
JqaigJnnyLn5OflYuLOmRj21QLcMHZAEhSQah5ljEU8CjRPX5vzacpLifHWrVvYtWsXNm8ufsmEjRs3Ijs7G71
69Xrha+7fvx9XrlzBTz/9pNe+Z88ebNu2DSkpKbpNc7799lvs3LkTq1atwrhx44pca8WKyssbb7+NDh06o
H379jAyMkKnTpOwYcKEl/ik5ZNGoOFoaCimT58O4FGh/fz58li8eDF69+5dqu89adJfd88uTJoOSfd88uTJiIiIgJG
REaZNm4Zz585h27Zt6NWrFO6ePFmqWYiIiIiISlO5LjIuXLgQX331FRISEhAcHIz58+ejUGt4Uo
eFNn3q8da/3ObrX+8+8xpQ1m/Sey+RydB48CpOHP3OjnKq16z51rcwnKY1VmP/ngSLtEZ2669Z8fBOFhjfFpsX
zYO/sCvcKAYi9dB6/rVyCxhOfrTMnkUjQptf72Lj4Gzh7ecPB1QPr5kXC2sERtSJa6K4zqU9n1IpogVZ9+gF4t
EERoIWLty8SbsVi9VdTdbu0A49GkG5YOAthzVrDys4BCbdvYs1XO+Dk4a03OvxWxn7+di+oNGsOnciCARzuqr/5
qKsI7dMEfa1e8cTuqP2nFihVwcHBA69atiz2+bNkytXrt2dnZAFDsMgIajabIB9xvP8FXwDlsMgIajabId
R48eIApU6boNuxRq9UoKHiOREBBQQHU6jdreYDiODs7o3LlnptlSpVwqZNzyAQCC2oL8xiE2jaBFVKvjBBXeX88
tWrHXtLOzgOwmQ2Jiol57YmJikc2pZp07o1+/fsjIyIC5ufkrmrfkwiI
iIiIkGV2yLj83ZsJCK8iF/P+59Owbl4JSbB2cETT/8j31t9u9u7V/f//5ufwMQ3HQnLPrRtmOCi5sfbvkpTYMmkSHrNuwd0hxsyNIidi25Dt9u//6gfzbM+QRZ6emoWKMmxn+3Fgrlv9N
w/7uj+rN2aQcAqUyKW1cuIeqXn5GdkQ5re0cAcE12uIrsM+KbI7d9zVyzjO52+YtWWnri2seRtcOHYY47u/AxdvX
wz/+sU3RCkNLztV+zGNRoOlS5eiVatWummOT4qLi+Pff/+t2/+Pff/+t2/35v9q2bYvhw4ejSZMmurbMzExs2LABoOePLvI
aKysrWFhY4J133sHHH38MpVKJTZs2ISYmBgEBAUXOHzt2LLp27YqUlBSkpKTA19cXS5cuhbe3N2bNmNvXcKfNPU
56mqterVw9XrlzRa7t69So8PTODNoExsnJCbt379YVFdPT03H48GH06dPHmGHJ3E3OEGDDgv5cDACgUCtSoUQ07d+9G+/btAz
6d7179240H2C7K53a4LparRYfffQRZs+eGDTSzsyLFXACiKOgSERERkeGSaF91Z4RSVrt2bUFvgSb0Z7u77z72BEiryWQtaAK7ffWRMLFiwA8OgP7u7u7
hgyZEixU73y8vKQl5ene56WlobatWuDyMjOsaiWIEyYhqHBm/UJg2yEx7E6HSrEz4hHm+Mz0DnRfrwW1TnvgJXKaoy7UFhjNW0mZAcVxP8bSgtlpWJ0Ak8mk7R3VBLM1kJvxXgZKVwoGrpEe1PrVHrI1hyx3bVhB/ebAZCTOzC2hlsnHmDEqqWeZ
9EbnpTzh7ehc5vmHhbBzavhWzf9tT7G7oPUMD8MHEGXpLC+zZ/BPWzpq0+TsOwMSs6Mi3mIvnsPHbuYi9dB6Fh
QVw8/FD+/cHIrheheQ73zzh7ej02L52Hiip0752Xm4Olk8bh70KShAHTZsHS5uWmqQMl1/clcb/HXDiLKf2
6osNHQ1C7aUvcuHAWy6aNR7/PpqBey3YAgG0rl+K3K+3Vd/hoOkzYe/qho2LvsHt61cwc8N2KJSPiuIzBvRGaK0ma
NqlBwDgyF/bsXTSPT93xT4VAnCjh9X4eiuPxC58Q9Y2trpZdi7ZQPOHOHT6AoZHzAAA3zp/F14pMFyxB96G8
c37ODMzf8/tqftSTv+WdJT0+Hu7s7UlNTYWlpWbSvUEPUBSvURERETI5W7ksMUbn58PexERGAgtbsV9H58
vprkddMmjQJkydLPs49HmV69Ivs2PhfnN376KuaOhFoksJycDFtbW0g
kklLNW149HmVQVqM56RH2u3DY98Jgvwmu6HfS8M9vuj6d8ZGRnF7pZOrErRIapXBYZX4VSqYRSqb/Bhy
hkLCwuD/SVISOx34bDvhcF+Fw77XHiG+sO6a6+cJk3tT4u5KPLViewAF999RUSEhj0ZoMHiZoXf6+FfaMebtjwmRAFbXlmwvpgLtByn+2HmQuaOZ05LI0+tF1OZEyjQ0zP0DH0n9eS
jWFiYgMmIiIiIiIjov8rtdOmRIOeid+/eCAONRa1atTB37lxkZWXhb9++Qkd7oXOHINHIqXex34
bDvhcF+Fw77XhjsdyIiIiKiosrl7tKPLViwAF999RUSEhjJQrVo1zJs3D7Vr1X6FhERERERERERET2hXBcZiYi
IiIiIiIiiIiqPwrl2syEhERERERERERR0ZuDRUYiIiIiIiIiiiJ6LSwyEhERERERERERER0WthjUWMhjDg7tyr
4uMIuLj440kpKQi7ampqfDx8REgkWFSq9U4c++YMUlJShI4ierznhdG4cWOkpqpqbYWaU9PT0fjxo0LPpABmTJ1Cr
zs4u05+TkYMqUKQIkMgz9+vVDRkZGkfasrCz069dPgEQfinWGtiHrgfasrCzo69dPgERERERERERRUPD5cWEaLUioSEBDg40Oi1JyYmwsPDA3lMc
QIlE7fhw4cjMDAQ/fv3h1qtRq94n+Hq6SrCgoKBEomfjZzDPH
x8UX6PikpCQ40DlCr1QIlE7en9fvDhw/h5SEwsJCgZaIEREREEREZUfcqED00vbunWr7ucdO3bAO0tJS91ytVmP37t3w8vISIFnZNhdHVOr1di+fXuRaarRZkZRwRYZGRkwNjYGDRkwNjbWHAo2NjQCJxcNjbWHVOr1di+fXuRaarRZkZRarRYZGRkwNjbWHVOr1di+fXuRaarRZkZRarRYZGRkwNjbWH
t3w8vISIJlh2LhxI3r06AEA+0233xAbG4vLy9y9jzZo1+Oyzz4GBE4oP73lhnD17Vvz
x559/wtXVVYhoBkr1UIikRRpj46OHo2NjQCJxcNjbWHVOr1di+fXuRaarRZkZRarRYZGRkwNjbWHVOr1di+fXuRaarRZkZRarRYZGRkw
EYRkEofzXqXSCT4779OIyMjeHl5YdasWWjTpo1i5fh5ubGz788EOYmJhg7ty5ilI2NRXBwMNLT4W
OKDq854UhlUp1Ba7i/tehUqkwf/58Th8tBdbW1pBIJjEhLS40FhYVeoVGtViMzMxMff/wxFi5cKEGBK8Xnyni+OR
CLB5MmT8dlnn5VhKiIiIki8okjGUVAo9EAALy9vXH16lWj/6JRYsWAQCO2dkJnMiwODo64ufOuLFi3B2dsaff/7J
```

hk8kETidOvOeFERsbC61WCx8fHxw7dgz29va6YwqFAg4ODrznS8ncuXOh1WrRr18/TJ48WW/OrkKhgJeXF8LCw
gRMKE579+6FVqtF48aNsWnTJr3RogqFAp6ennBxcREwIRERERFR+cGRjESvadKkSZg7dy6cnZ2RnZ2Nq1evQql
UYvny5fjuu+9w+PBhoSMSkUjs27cPdevWhZGRkdBRDMqtW7fg7u6uGOVNREREREREFscgoMrt378bu3btx//593
Wivx5YvXy5QKvHbuHEjbt++jU6dOsHNzQ0AsGrVKlhZWeHtt98WOJ248Z4XxrVr17B3795i+33ChAkCpTIMGoO
G169fL7bvGzRoIFAq8UtNTcWxY8eK7fdevXoJlIqIiIiIqPzgdGkRmTx5MqZMMqYLQ0FA4Ozs/cx0pKjmrV69Gl
y5doFQq9dq7du2K9evXC5TKMPCeF8Z3332HAQMGwM7ODk5OTnr9LpFIWGQsRUeOHEG3bt1w69atIutiSiQS7i5
dSn777Td0794dmZmRdbElEgkLDISEREREYEjGUXF2dkZkZGR6Nmzp9BRDIpMJkN8fHyRHUaTkpLg40DAX/pLE
e95YXh6emLgwIEYO3asOFEMTrVq1eDv74/JkycXW1h/cq1GKjn+/v5o1aoVpk+fDhMTE6HjEBERERGVSxzJKCL
5+fmoW7euODEMjlarLXYE3Z07d/gLfynjPS+MlJQUdOrUSegYBunatWvYuHEjKlSoIHQUg3L37lOMHTqUBUYiI
iIiomfgCuYi8v777+PHH38UOobBCAkJQfXq1SGRSNCkSRNUr15d9wgODkb9+vUREREhdExR4zOvjE6dOuGvv/4
SOoZBql27Nq5fvy50DIPTvHlznDhxQugYREREETlGkcyikhubi6WLl2KXbt2ISgoqMjuo7NnzxYomTi1b98eA
HDmzBk0b94cZmZmumMKhQJeXl7o2LGjQOkMA+95YVSoUAHjx4/HkSNHEBgYWKTfhw4dKI4/HkSNHEBgYWKTfhw4dKlAy8RsyZAhGjRqFhIS
EYvs+KChIoGTi1rp1a4wZMwYXL14stt/btWsnUDIiIiIiovKDazKKSHh4+FOPSSQS7zNmzpwbzTGI5Vq1ahS5cuM
DY2FjqKweE9Lwxvb++nHpNIJLh68+XQhNIJIiJiSnDNIZFKi06AUEikeiWbeAasKWjuH5/jP1OREREREREREROW0RUYSuX7+OGzduoEGDBlCpVE/dmIRKxrZa1bt5553NPTs4ySEBERERERERER6WORUYSuX7+OGzduoEGDBlCpVE/dmIRKxrZa1bt5553NPTs4ySEBERERERERER6WORUYSuX7+OGzduoEGDBlCpVE/dmIRKxrZa1bt5553NPTs4ySEBERERERERER6WORUYSuX7+OGzduoEGDBlCpVE/dmIRKxrZa1bt5553NPTs4ySEBERERERERER6WORUYSuX7+OGzduoEGDBlCpVE/dmIRKxrZa1bt5553NPTs4ySEBERERERERER6WORUYSuX7+OGzduoEGDBlCpVE/dmIRKxrZa1bt5553NPTs4ySEBERERERERER6WORUYSuX7+OGzduoEGDBlCpVE/dmIRKxrZa1bt5553NPTs4ySEBERERERERER6WORUYSuX7+OGzduoEGDBlCpVE/dmIRKxrZa1bt5553NPTs4ySEBERERER6WORUYSuX7+OGzduoEGDBlCpVE/dmIRKxrZa1bt5553NPTs4ySEBERERER6WORUYSuX7+OGzduoEGDBlCpVE/dmIRKxrZa1bt5553NPTs4ySEBERERER6WORUYSuX7+OGzduoEGDBlCpVE/dmIRKxrZa1bt5553NPTs4ySEBERERER6WORUYSuX7+OGzduoEGDBlCpVE

UWkX79+yMjIKNKelZWFfv36CZDIsGRnZ+Py5cs4e/as3oNKD+95YUyZMgXZ2dlF2nNycjBlyhQBEhmOW7duwdX
VFZ6ennoPV1fX5xYg6dWtXr0aeXl5Rdrz8/OxevVqARIRERERERERZU/HMkoIjKZDPHx8XBwcNBrf/jwIZycnFBYW
ChQMnF78OAB+vbtiz/++KPY45xGV3p4zwvjaf2elJQEBwcH3vOliHOvDPY7EREREdHzceMXEUhPT4dWq4VWq0V
GRgaMjY11x9RqNbZv317kFyMqOcOHDOdqaiqOHj2KRo0aYcuWLUhMTMS0adMwa9YsoeOJEu95YT1e/++/oqOjY
WNjIOAiw/GOvk9KSoKpqakAiQzDO/r9zp07sLS0FCAREREREVH5wyKjCFhZWUEikUAikcDf37/IcYlEgsmTJwu
QzDDs2bMHv/76KOJDQyGVSuHp6YmmmTZvCwsICM2bMQ0OvWrYWOKDq854VhbW2t1+9PFl3UajUyY2Fi1atBAwIRERERFR+cE
iowjs3bsXWq0WjRs3xqZNm/RGEikUCnh6esLFxUAhOKWlZWlGzVnbW2NBw8ewN/fH4GBgTh16pTA6cSJ97ww5
s6dC61Wi379+mHy5Ml6I7gUCgW8vLwQFhYmYELxetzXWq0W5ubmUKlUumMKhQJ16tTBBx98IFQ8OWrfvj0A4My
ZM2jevDnMzMx0xx7f8x07dhQoHRERERFR+cIiowg0bNgQhYWF6N27N0JDQ+Hu7i50JIMSEBCAK1euwMvLC8HBw
ViyZAm8vLywePFiODs7Cx1PlHjPC6N3794oLCyERCJB48aN2e9laMWKFXi8hPL8+fP1il1UeiZOnAi1Wg0vLy8
Oa9aM3+lERERERM/AjV9ExNzcHOfOnYOXl5fQUQzKDz/8gMLCQvTpOwcnT55EixYtkJycDIVCgZUrV6JLly5CR
xQt3vPCMDExwaVLl+Dp6SlOFIOiOWhgbGyMCxcuwM/PT+g4BsXY2BiXLl2Ct7e3OFGIiIiIiMotjmQUkcaNG2P
fvnOsuJSxHj166H6uUaMGbt26hcuXL8PDwwN2dnYCJhM/3vPCqFWrFk6fPsOiYxmTSqXw8/NDUlISi4xlrGrVq
oiJiWGRkYiIiIjoGVhkFJGWLVVti3LhxOHfuHGrUqFFkp9F27doJlMxwaLVaqFQqVK9eXegoBoH3vDAGDhyIUaN
G4c6dO8X2e1BQkEDJxG/mzJkYM2YMFi1ahKpVqwodx2BMmzYNoOePxtSpU4u95y0sLARKRkRERERUfnC6tIhIp
dKnHpNIJFCr1WWYxrAsW7YMc+bYMc+bMwbVr1wAAfn5+GD58OON5//32Bk4kb73lhFNfvEokEWq2W/V7KrK2tkZ2djcL
CQigUCr0NYAAgOTlZoGTi9uQ9/+Su6rzniYiIj+xZGMIqLRaISOYJAmTJiA2bNnY8iQIbqddQ8fPowJRIOYgL
i4OU6ZMETihePGeFOZsbKzQEQzW3LlzhY5gkPbu3StOBCIiIiKico8jGYlek729Pebfb0PebNm4euXbvqta9btw5Dhgz
Bw4cPBUpGREREREREFQ2OJJRZLKysrBv3z7ExcUhPz9f9f79jQoUMFSiVuBQUFCA0NLdJeo0YYNFBYWCpDIsPCeF
87FixeL7XeuhVk2cnNzi/Q91wYsXdnZ2cXe81yPHlIIIIxlF5fTp02jVqhwys7ORlZUFGxsbsbPHz4ECYmJnB
wcEBMTIzQEUVpyJAhMDIywuzZs/XaR48ejZychHCxcuFCgZOLHe14YMTExe0edd3Du3DndWozAv2vVcX26OpOVl
YWxY8diw4YNSEpKKnKcfV86Hjx4gL59++KPP/4o9jj7nYiIIgIePquCfTGGTFiBNq2bYuUlBSoVCocOXIEt27
dQo0aNfD1118LHU/Uli1bhqpVq+L999/H+++/j8DAQHz33XeQSqUYOXKk7kEli/e8MIYNGwZvb2/cv38fJiYmu
HDhAv7++2+EhoYiKipK6Hii9sknn2DPnjlLi5YvXq1OPFEa/jw4UhNTcXRo0ehuQn
w559/YtWqVfDz88PWrVuFjkdERRERERVC5wJKOIWF4Z4ejRowgICCV1RUOHz6MSpUq4ejRo+jduzcuX74dsdERRC
g8Pf6HzJBIJ9uzZU8PpDAvveWHY2dlhz549CAoKgqWlJY4ePQo/Pz/s2bMHI0eOxITdTQ8GJJGLU8/KFsvhSbyh
qBAsLC5w6dQoVKlTAmjVrsG7dOmzfvl3oaiKLk70yMX3/9FbVq1YKFhQVOnDgBf39/bN26FZGRkThw4IDQEYm
IiIBMclGUXEyMgIUumjwakODg6Ii4tDpUqVYGlpidu3bwucTvucTry466hweM8LQ61Ww9o9e/cQQ4Wv9zcHMCjguO9e/cQEBAATO9
PXLlyReB04acnAwfHx8Aj9ZfTE5OBgC89dZbGDBBggDJDRRC0rKwsOD4AAGtrazx48AD+/v4IDAxEUV0nBE5HR
ERERFQ+sMgoIiEhh+/Dj8/PZsGoIiEhITh+/VKhA1DJycjQIM+ePHq6oqGDRuiYJcjQgZi2YQcWxEl5MIKHSx3teGIWFhRg
/fjwsLS3h5eUFLy8vWFpa4vPPP0dBQYHQ8URv9+7daZ4mDXx9feHr64sbSzWXLlqFq1aowNjaGs
bExqlatiu+//17oWERERERE5QY3fhGR999H6dPn8a2bdsQFhYGADADh8+DGDRuf+DCGDRuGjz76COOvXrxc4oXhdu3YNe/f
uxf3796HRaPSOTZgwQaBkhnJgjzBvk5s2bERkZqdfvkyZZNQlJSEhYtWvMQvH69ttvvMWzYMLz77ru69R+PHTuG
DmCVq1aYc6cORg0aJDACcVpwoQQJmD17NoYMGjJiIiIiTH6dIiYmpqih07duCtt97
Sa9+/fz9atGiBrKwsgZKJ23fffTk5OelPoJBIJTp06JWA6ceM9LwxLSOusX78eLVu21GvfvfvnO7u
nbtylGkpcjNzQ3jxo3D4MGD9doXL1yI6d0On4+7duwIlEzd7e3vMmzcPXbt21Wtft26dhgwZgrocPHwqUjIiIi
o/OBIRhGxtbWFpaVlkXZLS0tYW1sLkMgwTJgwAde+fff+/fvBOgl+fWOEMLEpVbGEqlEl5eXkXavb29oVAoyj6QNT
UWLFi2KtDdr1ozfQaWooKAArIRCARIRERERERZU/XJNRRD7//HOHOMHDkSCQkJuraEhAR8+sknGGaiMGT48ecC4rRHr
PFzCZuKWkpKBTp05Cxz5CxzBIvOeFMXjwYYEydOhV5eXm6Xtry8PEybNg1DhgwRMBk5PN2R3NRJ3XX39FmzZtBEhkG
Hr27FnsMgBLly5G5F9+7dBUhERERERFT+cLqW0iH0OxD69vV/fP3116uhI2ud0OfP3116hXr55ubcAjR
47g4MGDGDVqVFCsLHTnDh06VKiVKiCsWnL7u7uqFONDgDg6NGiIuQLQ69evWBBkZZKQ7d/bs2ULIIInIGAI

```
SFIuMIjJ58uQXPnfixImlmET85s2bp/s5KysLs2fPRuvWrREYGKj3yybAX/RLE+95YfTt2/eFz12xYkUpJjE83
t7eL3SeRCJBTExMKacxHOHh4S90nkQiwZ49e0o5DRERERFR+cQiI9Er4C/6RERERERERET/YpGRiIiIiIiIiIi
IXgt3lxYRtVqNOXPmYMOGDYiLiON+fr7e8eTkZIGSiduUKVMwevRomJiY6LXn5OTgq6++woQJEwRKJn6854Wzc
ePGp/Y7178sXXfu3MHWrVuL7XuuB1h6Tpw48dR7nmuPEhERERFxd2lRmTx5MmbPno0uXbogLS0OeORIcOHSC
VSjFp0iSh44nW5MmTkZmZWaQ9Ozv7pdYMpJfHe14Y8+bNQ9++feHo6IjTp0+jVq1asLW1RUxMDFq2bCl0PFHbv
Xs3AgICsGjRIsyaNQt79+7FihUrsHz5cpw5cpw5c0boeKK1fv161K1bF5cuXcKWLVtQUFCACxcuYM+ePbC0tBQ6HhE
RERFRucDp0iLi6+uLefPmoXXr1jA3N8eZM2d0bUeOHBGGPP/4odERRkkqlSExMhL29vV77nj17OKVLFzx48ECgZ
OLHe14YFStWxMSJE9G1a1eYm5jO0joaPj4+mDBhApKTk7FgwQKhI4pWrVq10LJlS0OyePFnX9w4ODujevTtatGi
BAQMGCB1RlIKCgvDRRx9hOKBBun739vbGRx99BGdnZ/6FEhERERO0JJRVBISEhAYGAgAMDMzQ1paGgCgTZs2+
P3334WMJkrW1tawsbGBRCKBv78/bGxsdA9LS0s0bdoUnTt3FjqmqPGeF0ZcXBzq1q0LAFCpVMjIyAAA90zzE+v
WrRMymuhdunQJvXr1AgDI5XLk50TAzMwU6ZMwZdffilw0vG6ceMGWrduDQBQKBTIysHTpUoHTE
RERERGVD1yTUUTc3NwQHx8PDw8P8P+Pr64q+//kL16tVx/PhxKJVoe0Jzty5c6HVatGnTh1MjxZb8qcQqGAl5c
XwsLCBEwofrznheHk5ITk5GT5GR4enrCw8MDR44cQXBwMGJjY8HB8aXL1NRUtx6ms7MzoiIi4OPjA0AU
bO2ttYVO11dXXH+/HkEBgYiNTUV2dnZAqcjIIofWGQUkXfeeQc3b97ExwcHeHDh+WEf+mGEFHKzfeQe7d+9G7dq1MWTFXwcRow
YIXQ80enduzMzMBAUFISsrC6NGjcLXX3/Ps7
g4cOIBKlSqhVatWGDVqFM6dOzxEp06dMGzYzYMOzZswc7d+5EkyZNhI5HRER
ERFQucE1GETty5AgOHToEPz8/tG3bVug4opKeng4LCwvdz4YUXkZFz8H+HNbW1kJ
HJCIiIISHIuMRK9AJpMhPj4eDg4OkEqlkEgkRc7RarWQSCRQq9UCJCQiIiIiIiKjucLk30Cvbs2QMbGxvdz
8UVGYmIiIiIiIiIDAVHMhK9otjYWHh7ewsdg4iIiIiIhIcCwyEr0iqVQKT09PhIeHo3HjxmjUqBHc3NyEjkV
ERERERERERVOZYZBSBmJgY+Pj4CB3D4ERFRekeR48eRX5+Pnx8fNC4cWOEh4cjPDwcjo6OQsckIiIiIip1L
DKKgJmZGby8vNCuXTu8/fbbqF27ttCRDE5ubi4OHTqEJVPl0e5ubi4OHTqkKzoeO3YMBQUFFqFixIi5cuCBOPCISmfz8fMTGxsLX11e
3yzeVjdu3bwMA3N3dBU5CRERERFS+sMgoArm5udixYwd+//33rGsuLp7GR4TGLbnzbTj2sGlEG/TqpNQoAcPXqVfj
4+GDIkCFwdXXFuHHjBE4oTWhFhZg8eTLmzZuHzNMMAI/+gm/IkCGYOHEiJyMBE4IRERERCQ8Dn8QAWNjY7Rt2
xZt27aFVqvF4cOHsXXrVowd3u3ZFREQE2rVrh7Zt28Le3l7ouKSn5+PI0eOvbpp0+7u7mjQoAEWLFi
AhgObCh1RdObOnStOBIPUvn173c+5ubn49ttvZYSFhQEAjhw5wggsXLmDwIECJTQMn376KoxEVFYUWL
Vro2iMiIjBp0iQWGUvJkCFDRuru+cOHD2PSpelFHoPW3tzcaNmyI+vXrAxh2dhY6ev+Hs7OBG3b
9/G8uXLBUomfp59QpAd5+nvW59Qp04dmJubo59GkJavX4+WLVvvfvh1du3Z
FWlqaQMmIiIiMoPFhkNSFJSEpK3+Hn5ydOFEwMjKCs7vjOaNGqFhwHaw4wBgaEBOdDTy8/PRpEkTd
OjQQXfu5s2bhYpjjJRERERERERCQoTpcmKiEipEaj4Rx3cmxrzyRHmFbPEFqLY+hXS/Ws
pVkhQAWvWrMG6deuwffft2o2SK1oYNG/DNN9/gQVLAIBKlSph2LBh6Ny5s8DJxO/GjtQ
2Ps2LEIDAwU0ppo9e3b94XPXbFiRSkmISIiIqv1hkJHpFGo0GJ06cQFRUFRUFPbu3YuDBw8iKysLwMnaMAqn
8PT2FjipaZmZmuHjxIjw8POlmOdm5obNmzejVq1aiI2NRWBgIDIzM4W0SERERERGQQOF1aZFJTU7Fx40bcuHEDY
8aMgY2NDU6dOgVHROe4uroKHU9UrKyskJWVBScnJ42PonDlolKgRfH19hY5mMHx8fBAbGwsPDw/4Dw8/f2
sQK1atfDbb7/BysPK6Hii9vi7JiYmBqNHj+Z3TRnZmjdvrte+Y8cOaDSaIrsfUl68OABrly5Ag
AiAvb29wImIiIiMoPjmQUkbNnzyIIgKWlpa4efMZmrly5Ah8fH3z++edJKLE7xrhBAUFYebMmWjVqpVe+59//
FjmKw5syZA5lMhqFDh2LXrl14o27ZttFotCgwCoKMHv2bAwcObeBerrNqqeWDTdtLnAAb+kbrXTkoiKLE7xrhBAUFYebMmWjVqpVe+59//
```

```
Co5SqUS6enpRdqvXr3KdaNItHbv3o02bdrA19cXvr6+aNOmDXbt2iVOLFHjd41w3n77bQwfPhw3btzQtV2/fh2
jRo1Cu3btBEwmbtnZ2XBOdCzS7uDggOzsbAESERERERGVPywyiki7du0wZcoUFBQUAAAkEgni4uIwduxYdOzYU
eBORCXv22+/RYsWLWBubo5hw4Zh2LBhsLCwQKtWrbBw4UKh44kWv2uEExkZCVNTU1SsWBHe3t7w9vZGpUqVYGt
ri6+//lroeKIVFhaGiRMnIjc3V9eWk5ODyZMnIywsTMBkREREREETlB9dkFJGOtDS8++67OHHiBDIyMuDi4oKEh
ASEhYVh+/btnEpHouPm5oZx48Zh8ODBeu0LFy7E9On+TcffuXYGSiRu/a4Sl1Wqxc+dOREdHQ6VSISgoCA0aNBA
6lqidP38ezZs3R15eHoKDgwEA0dHRMDY2xo4dO1ClShWBExIRERERCY9FRhE6ePAgoq0jkZmZierVqyMiIkLoS
ESlwszMDGfOnEGFChX02q9du4aQkBBkZmYK1Mww8LuGDEl2djbWrl2Ly5cvAwAqVaqE7t27Q6VSCZyMiIiIiKh
8YJFR5FJTU2FlZSV0DKJS0a1bN6SbN4SEhGDMmDF67V9//TV//V9//TVOnDiB9evXC5RM3FavXo0ouXbpAqVTqtefn52P9+vXo1
auXQMnEad68ef8jwwwwwwhbGyMefPmPfPcoUOHllEqIiIiIifSwyisiXX34JLy8JLy8vdOnSBQDQuXNnbNq0CU5OTti
+fbtuiheRWEybNg1ff/016tWrp1sX7ciRIzh48CBGjRoFCwsL3bkUnJSXBwcEBarVao
GTi5O3tjRMnTsDW1hbe3t5PPU8ikSAmJqYMkxmm1q1b4/vvv4ezs7PQUYiIiIiIyhUWGUXE29sba9euRd26dbF
z50507twZP/30EzZs2IC4uDj89ddfQkckKlHPKrg8icWXkiWVSpGYmFhkJ+no6GiEh4cjOTlZoGTilJaWBktLS
6FjOD/MzcORHRONHx8foaMQEREREZUrcqEDUMlJSEiAu7s7AGDbtm3o3o3LkzmjVrBi8vL9SuXVvgdEQlLzY2Vug
IBiUkJAQSiQQSiQRNmjSBXP7v/0LUajViY2PRokULAROKk42NjW7kaOPGjbF582YugOFERERER0UOi4wiYm1tj
du3b8Pd3R1//vknpk2bBuDRTqvckhEr6t9+/YAgDNnzqB58+YwMzPTHVMoFDy8kLHjhOFSideZmZmuqnoUVF
RKCgoEDqSQfPO9ISRkZHQMYiIiIiIyhOWGUWkQ4cO6NatG/z/8/NatG/z/8/JCUlISWLVsCAE6fPl1k912iN9XIkSMxdepUm
JqaYuTIkc88d/bs2WWUyjBMnDgRAHRRvxobGwucwDBEREQgPDwelSpVAgC0b88Pz/bs2WWUyjBMnDgRAHRRvxobGwucwDBEREQgPDwelSpVAgC88847UCgUxZ67Z8+esoxmkM6fPy9
OBCIiIiKicolFRhGZM2cOvLy8cPv2bXx+PgwCIECpyMqGadPn9aN5Dp9+vRTz5NIJGUVyeDO7t0bw
KPdp0/fvw+NRqN33MPDQ4hYovXDDz9g1apVuHHjBvbt24cqVarAxMRE6FgG6eTJk7h06RIAoHL1yqhevbrAiYi
IiIiIyg9u/EJERC/l2rVr6NevHw4dQqTXrtVqIZFIuDxCUtPT9ftlB4eHo4tW7ZwTcYydv/+fbz33nuIiorS9
X1qairCw8m8E7n2ZVuXRY8jGM4E2xbW9zfUnUHqhjmCTC0gVWXv2RLrTLmqo68WjYpJ3nuRyua4wQyWrT58
+kMvl2LZtG5ydnTlqtJRZW1vrNn5hXwtjyJAhyIjwIULF3TT1i9evIjevXtj6NChWLLduncAJiYiIiEx5GMI
mJtba33vKCgANnZ2VAoFDAxMUFycrJAyYhKR8uWLdG2bdsiywbW7duxfbt2wVKJm6mpqY4efIkKlasDuHQ
Ug2BpaYkjR46gUqVKkMlkSEhI4Mi5MmZpaYldu3ahZs2aeu3HHjtZs2aeu3Hjh1Ds2bNkJqaKkwwIiIiIqJyhCMZRSQl2
7Vr1zBgwACMCGTNGgERepevo0aPFbu7Sqf7SqFEjfPbZk7wcPhY5hMJ7c+EWr1XLjFwFoNJpid5Q2MjI
qsiYpEREREZGhYpFR5Pz8/DBDhw5Fu0uOBkMgwfPnll/jkk08wfp0B
AYGFim+cJp6yeLGL8Jr3LgxhgObnXr1sHFxQUAcPfuXYwMQJNmjQRODROREREUfnA6dIG4MyZM2jQoAHS090O
FjkJUosLDw1G1alXMnz9fr33Qo+EE4e/Ys9u/fL1AycZNKpQaqCK7uDNjV9KHzd+Ecbt27fRrl07XLhwAe7u7u
lWrYuvWrXBzcxM4IRERERGR8FhkFJGtW7fqfa3VanH37l1MmDABDrerXrw8/Pr7/NVZZjTh8+bt
v3o3jx4/jr7/+Qv369VOKE779u175vGGGDRuWURTfXi6up2dncBJDIdWq8WWqVEiiIgIgVMRERERE
ZUfLDKKyOPPPPRPRRY9JJBLYR29ujcePGmDnVrFpydnQVKR1R6zpw+gpnzyJxahEPv30U/jbqdzpSQVCFUAQPv3OU/j5+Qk
dahDEpaa
m4rPPPsNPP/2kW4PX2toa7733HCgWGYmI6KXt378fS5YQUXMDH7++We4urpizZoZ18j6YPb2x
ltvvSVOPNFJTk5GWFgY7t69i+7du6NSpUoAgIsXL2LdunWIioqClZWV0DEMwqRJk3DmzBlZWRg+fLl2LDz+9wAAAAACkfzy3+cpJCcfYdnjMnDsHB9TTyDEspDWSBtiWjHrmjWaNGcbzvNjO5GdQlpGWnbDsCgS+ZVK
1F66Uzi5y9KpNkBhdnwAeSUHRPvn2v9Qvd3XF+W2eyvhnBWh1yDnDDhKB0vhUnVajYqWsTgWjbjKfAbBbjYPrq9
Zfhct+Du/oI2v/AvOYb2+OLP7R9AAMGvRjh8mhkd3oPbM8LNAtXWyTyj2JIVaXJmaF1/9Qe/+5ZKyL/IuvD1Q8
```

```
fvw6tVgs/Pz/uYE9ERERE9B8sMhLRG6tDhw5477330LlzZ6GjGIS8vDxMmjRJty7gJ598gvbt22PFihX47LPPd
FNKx44dK3RUIiIiIiIiKmPc+EUE0tPTX+g8CwuLUk5CVLZat26NMWPG40LFiwgMDISRkZHe8Xbt2gmUTJwmTJi
AJUuWICIiAocOHUKnTp3Qt29fHDlyBLNnz0anTp0gk8mEjklEREREREQC4EhGEZBKpXobXvyXVquFRCKBWq0uw
1REpU8qlT71GO/5kufj4405c+eiXbt2OH/+PIKCgtCnTx8sW7bsmd9BREREREREJH4sMorAvn37dD9rtVq0atU
K33//PVxdXfXOa9iwYRVIR2NSgUiI2N1X23qFQqHDt2DIGBgQInIyIiIiIqFxurQI/Ld4KJPJUKdOHW6GQ
UQlSq1WQ6FQ6J7L5XKYmZkJmIiIiIiIjKCxYZieiNNWXKlGcenzBhQhklMQxarRZ9+vSBUqkEAOTm5uLjjz+
Gqamp3nmbN28WIh4REREREJiNOlRcjc3BzR0dEcyUiiFxISove8oKAAsbGxkkMvl8PX1xalTpwRKJk59+/Z9o
fNWrFhRykmIiIiIiiovOFIRpHiJgxkCE6fPl2kLT09HX369DHf5HX369ME777wjQCJxY/GQiIiIiInoYjGUWgQ4cOes9
/++O3NG7cmFMYyWCdO3cObdu2xc2bN4WOQkRERERGQQOJJRBCwtLfWe9+jRQ6AkROVDWloa0tLShI5BRERER
EREZDA4kpGI3ljz5s3Te67VahgYcOG+PHHHwVKRkRERERGRYWGQkojeWt7e33nOpVAp7e3s0btw
Yn376KczNzQVKRkRERERGRYWGQkIiIiIiIiIiKi1yIVOgAREREREREREREREREREG92bjxCxG9cfr16/dC5y1vfvryUk
xAREREREREREREwOnSRPQGkkql8PTOREhICJ71FbZ1y5YyTEVEREREREREREERkuDiSkYje0AMGDMC6desQGxuLvn37
PHrCxsRE6FhERERERZHB4khGInoj5eX1YfPmzVi+fffDk0HTqE1q1bo3///ZGYiIiIiIiIiIIgMCouMR
PTGu3XrFlauXInXVq1ejsLAQ5FLAQFy5cgJmZmdCxiIiIiIiIiIiAwGd5cmojeeVCqFqFRCKBVquFWqOWO4RERERERERGRwWG
RkYjeSHl5eVi3bh1bh2aNm0Kf39/nDt3XsWLEBcxHMRIEREREREGVMW78QkRvnIEDB2L9+vWoXLkyDB06FD
tjZ2Qkdi4iIiIiIiIMhgcU1GInrjSKVSeHh4FPDlAz4ICQk5JmbvGzevLkMUxEREREREREREREEZLo5kJKI3Tq9evSiRNBEREREREREREVE5wpGMRERERERERERERERE9Fq48QsRERERERERERERERERERG9FhYZiIiIiIiI6LWwyEhEREREREREREREREREREREREESvhUVGIiIiIiIiIiIi
iIiIiIieiOsMhIREREREREREREREdFrYZGRiIiIiIiIXguLjEREREREREREREREREREPRaWGQkIiIiIiIiKi1/J/1a5
JDqNXvk4AAAAASUVORK5CYII=\n"
          },
          "metadata": {}
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "Most people live is house/apartment (89.15%) and have an approval rate of
89.42%"
      ],
      "metadata": {
        "id": "D0T47rbNC82l"
      }
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "PeutE5WKG-bc"
      },
      "source": [
        "#### Children"
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "analyse_categorical_column(\"CHILDREN\")"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 507
        },
        "id": "GoM4IHzK4Uyg",
        "outputId": "4442c278-f574-4db0-ad88-1ebea510ce2e"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 1400x500 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAABPwAAAHqCAYAAACQilDXAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAACs8Ul
EQVR4nOzdd3hW5eHG8fvd2QvIggCRvZUhIIgoCCoOBAeKs47W4sD252rVunHUUfcoFVRwVUVRS0UUr0whe88EM
sne7zq/P9BXYxJkJDnJm+/nunLVnPO859yJMU3uPM95LIZhGAIAAAAAAAAQFKxmBwAAAAAAABQfyj8AAAAAA
AgCBC4QcAAAAAAAAAEEQo/AAAAAAAAAIIhQ+AEAAAAAAAABBhMIPAAAAAAAAACCIUfgAAAAAAAEAQofADAAAAAAAAg
ojd7AANxe/3KyMjQ5GRkbJYLGbHAQAAQAtgGIZKSkqUnJwsq5W/rQMAAHMEbeGXkZGhlJQUs2MAAACgBUpPT1e
7du3MjgEAAFqooC38IiMjjJR36YSsqKsrkNAAAAGgJiouLlZKSEvhZFAAAwAxBW/j9Iw3Kiqw8AAACNikfKA
AAM/FgEQAAAAAAACCIUPgBAAAAAAAAQYTCDwAAAAAAAgiQfsMPwAAgN/i9/v1drvNjoFmxmuyPL0Oymrl7+YAAKD
povADAAAtktvtl49e+T3+82OgmbGUrUqNTVTCgAAC1ovADAAAtjmEYyzMlM1mUOpKCr01cMT8fr8yUm+HQw2/XWedpS+++KKeUwEAAKC5Y4YfAABMzN9+nTdcsst+u6
775SRkdHg93073Q1+DwAAANQfC7j8AAIBmpLSOVO+//75uukkmjRs7TjNmzmAicAzAZHPEZtDdvlmzXZFrxfMFHTuzN48PoPqAFJXWBFp0dhw8lssSjhPKb4tqAAPUPjkR3v3kS5nqcGIFctlK3dswoahT1lLdewhwGY8YqUbytT3k69EHGEZGKu+jLkkyEtmJapxDVa9Z4FKf2aKqJYYOr2u5KqzSs+MbsBsszAHDDvdvdp7QdASFwe/rNpCuypDqXTR8xxZpnafeERqqEpFbPxYn59MhJbStG1rnixaTGhoX6OHJTCtIpPDrqzbX2gamYl9jr2WN2F3xaZDntSfAqq9aaXRBeQgRSnh8Q9tlvvXtI0/TV6yW3utaX8GaCrhHmPvgDV0PNkJxCPqPQpmbwxCM6UxUNRTaDrRzLtjuyQL6nmVlezW+RNhiC+jJjBBhoXqxQPOzT/mRKyZNH5+Dk

AAJqJ6d0ny+v1Kjk50XDMMAy5XC69+OKL1qZvXYPfU2bb72lp59+WmPmPGWGjKlDl2bvz48Xr33NmzzddNNwXOl5aW6Rzzr
tMTTzxR47VJSUmBXDVbjLX09Tc0RByGi4gT/ZVVcCKkPdZNk+YTKxgaooPmrVvUyD2ttSfGSJWlJxEgq2IDX4Ps
Ds+WWLVum9u3bS5IKCgq0ft29ejRo9q1iI1PAfoXXt74utY4XXRrPPnm6Ntc41PsOAAYgMmPTbcyHoAcDRiTtLVeNqVAOLLJV78mNWFI1xvGxjDA










































































[illegible]
```

DHq06ePpk6dqpiYmECp99RTT+nUU0/Veeedp9GjR2v48OEaMGDAMX3OAAAAWiqLYRiG2SEaQnFxsaKjo1VUVKS
oqKhGvTdLelHfWNILAPWrsrJSe/bsUWpqqkJCQsyOU28WLlyo008/XQUFBYqJial1zIwZMzR16lQVFhY2arZgc
rivHzN/BgUAAPgJM/wAAAAAAAACAIELhBwAAAAAAAAAQRCj8AAIAgMXLkSBmGUedyXkm65pprWM4LAAAQ5Cj8AAA
AAAAAgCBC4QcAAAAAAAAAEEQo/AAAAAAAAIIhQ+AEAAAAAAAABBhMIPAAAAAAAAACCIUfgAAAMCvdOzYUc8995zZM
QAAAI4JhR8AAEAzcc0118hisejxxx+vdnzOnDmyWCxHdSOKLQAAgOBlNzsAAABAU7Fx48ZGvV/v3r2P+jUhSF
64okn9Pvf/16xsbENkKppcLvdcjqdZscAAABolpjhBwAA0IyMHj1aiYmJmjZt2mHHHffTRR+rVq5dcLpc6duyop
59+OnBu5MiR2rdvn26//XZZLbJbDzg585pln1KdPH4WHhysJV//OMfVVpaGjg/Y8YMxcTEaM6cOerSpYtCQkI
OduxYpaenB8Y88MADOvHEE/Xaa68/pJSVFYWhuuSSS1RUVBQYc8012j8+PF69NFHlZycrG7dukmSNmzYoDPO0
EOhoaFq1aqVbrzxxxsD9v/rqK4WEhKiwsLBa5ttuu01nnHFG4P3vv/9ep556qlwbfv9bjJDQ5WSkqJbb71VZWVlgfM50Tk
677zzFBoaqtTUVM2aNeuwn1sAAICmjsIPAACgbHZbHrsscf0wgsvaP/+/bWO2bNmjUaN4AMPPPS
KD77rtPM2bMkCR9/PHHateuneunR566CFlZmYqtzzvt5ZrVY9//PHHateuneunR566CFlZmYqtzzvt5ZrVY9/
WW29p8eLFKiws1KRJk6qN2blzp/z8PyHDGb3yfmrhQ1uaL7XHW+/hbddHXMr1KIRnUbqdr31S2cOrRZD
WxsrFauXKkPP/xQX3/9tW6++WZJ0qhrRoxQTE6POPvooz8/P0LXBsLUVOpkU9TEW4DvH7EOoHxXcS/yD07/SlHoNtqYqctw3KlzGFbSx4LAjCUaGKHyRPhXUgTDIBZwNFyJGKtFxOb1/T1x/
f1/fffx+4hnSobExPT9e3336rf//733rF55ZeVk5PzG/8mAAAAmi6W9AIAADQzF154oQmPWyH3LtW3
+aZZzRq1Cjdd999dqKqSuXbtq8+bNeuqpp3TNNJNdcoLi5ONptNkZGRSkxMDDoe0cx9k6dGvFHHtEf/vAHvfz
yy4HjHo9HL774774ogYPHixJmjlzpnr06DGEtQxQdzNlO6I1lbaRqsbZbV7Ok2Klt6zsKGrdTK2KJU6zvRhPCE7nIhu5jPiQkEmo2irsbN2MW7wT1NUkL
Oc/AOt533jjjcDrwsPDJUkuVvvviiizjvPvD3xxBNKSEJQpEmTNHv2bF133XWSDpWGhYWFmjrA6aLsmoeH7iBGOTnKKnG6nJ5R53VuD
j6NKli55//nmddtppeuWVVV5SWlqb//Oc/WrFihQYNGiRJmj59unr06eVe8iHXqAGKQZDBwqYmpN5zSZY0ax/Ez/ckyqqsK+xkpsPbh52kkMxLyebjbqwFAHlPFv0HEZnTQm8gmVUXydjTtzj1hyWiFtKoPJVeh+fLxPQHIcXOY/tw5z2qo9e+Yj8jtdxdb1YsVWrzOubzcW
S1bttQ4t2XLFg0bNqzasHHhmnHjh3y+Xd+vv/5ZeQ/bqU9ZvCvnn/DSbDRn/SejNbgWN5j0csdHuGy5+XSXlt1aKaOlsEKE9dzYm63K8Sq2eM6PtqGayx+9ii5sq2ix4aSeX9ZNu9F4yx2+2BskySunf
vrpiYmGrZ2rdvHyj7JGno0KHY+/3atm1ehFnPi7OwiqnIIm2FkmsSD4Aj5yJ10Mhx+qPXh9XrwT1WbRPuXhWuHqPiHBh2pUUQcGvWHKmIIQ3fyOY7Glu
IObN04xMTGSpHXr1mnGjBmKiIgIvI0d01Z+v1979uzR1i1bZLfNWDAgBr5AQAAmisKPwAAgGZoxIgRGjt2r06
5554Gu8fevXt17rnnqm/fvvroo4+0O+vrQTSS5IObapR335Z7B2pQYMGqVOnTvdUUVGhTz75JLCcV5JKS
Ov1+9//XmvXg28rVu3TjtJkyCC9PT0Uo86Ksbjs4iL6ql+3+evPTm8bHiqzbjbjUlyhivoJXP9u2nhH
L6/Vq9erVgVmKP+UHAAB0rpjhBwAA0Ez16dNd1/t56ZdcBJbaNxaDbXqRs5BR2nxKN+hHuHjGjy9H/9tesDBeGe
aMmRIoACUpJCQEF199dat6d26ppiM5yvtlYNhpl9pVMNjlhhbeMhPe3LzCWsNenyJpcDSIY5Yw48D+l7jtR
0aPPvqvqoLrroIrlcrsC5uWKEt0LUXYNeOuNRFe29Nl37RsQjAyI+jUzbUqXr9YsfUNaBbqh0ToBzSsqlXTdAO4TBzEN
OYWQDHvt/Wb05hQ4eptSWCF/Kq7y9Zz7XSpHWgN5BYgZhm8l8Rl8++o5r9gSVjmO+mN7nf6MecLq89KNQLuv+mK1t
oyYNSRBOl2M1ab6UqCEVK2UCqcMxz7TLA4MZU8Y6Fpbgc6uuTC+Os0mVOTmiMqQ8tKhbZmxrAyJlWZ8b+ZV2DQi4AvLd
WdoomDZsKwVHIh0MpuuXWNHE1ZZkvLHdnoCZJKcN8fBoZBRR1cVxQtZh1QzzEKw7L7E+nWfsj2yHDpwzbg5DGZVAYNjFK
eWOAXNq9S8Gx+xFLE8KJG0ziOXqrOmrTzs9mEwVwmg5Ee6TYrX5+ZeR2oP23pbvgfqOFVJtUmB8DMSNtetPOnhx
oJMdCGRpEkrb/eBaQ3Ax5x2n7yQebI6a3MeXz7nT9bUAEVLLm3fV3I5T4JAZmpw0WwYrfDwZz0sm0bVaRYfWS0sfTYLkJ5v3AEK+iO6Tjs3gfptm4Ig3kPpGsR+6NfuPP/90GSqmVdSX5ivsPNUbBnP3+B7D4P+jhz3yOLWCfLh4IJv/d86s/0vwjgzU5NDQ9u4qY0eQmTHLcgt3
V9Z/l9zMDNnsqFmOQWnUeIiq8cxCK6yYumDtV7a3V0mkD7S3/Uym5+JdX+5JOfrxMgsZhgQrndqJ1UcNiuuBg/KuVMl
XSiEcCRrz0+kL3UIMEuOzfZHxNLg1Hdsv9AD0wcdaTIoLDJG/eZxIxvGyVKUYM6mj3VA5FQhlRa27CiVszfA4DyVJfBpEe0uDb6x/HzpjLI8r0
Y5pMlYiu4PDERQ9j4RSHXFZkzHczvZtPmFtKnXIX21FbYjsuZCM+Th8SzmDDxNkd3AbJoXsHwGR2mYcU9Hin6RyIc
j/wd7dd0W9HzhrvtVz0n1bTzNl9+hBCq7qsLLbYKbLyIV++ThTmC/RZOQK6EY8C6zZDU0dbOF3Obz86z0J3lbUbdoJ
nU1TH7hmAMtYr9kN3tDKQLy2XROgULxvBb6tlv4Fp9BkS9Tcrc7yg89GGAJMt3g8qXBM4t8d0Q3k9zAu5ecfJiIJvc
5vBJ6O4O/CcUzyAm9lq2PM5zUKl1ZhjbnUc8UNgU8dfwGZNWBcxCqQ2Ff4wPbZHv8rz

q9PjNjoTDyCmp0syl+zRz6T4lRoXo7D6JOrdvkvq3p/xD82UxjuMhJBaLpdoMP8MwlJycrD//+c/6v//7P0lSU
VGREhISNGPGDE2aNElbtmxRz549tXLlSg0cOFCSNG/ePJ1zzjnav3+/kpOT9corr+ivf/2rsrKy5HQ6JUl3332
35syZo61btx5RtuLiYkVHR6uoqEhRUVHH+iEek40bNzbq/RD8evfubXYEAAhKfr8/sDwTOFJOp7POPWaFm/gwKw
FxF5R7NWXtAn649oLXphWKvjeYvOTpEZ/dJOrl9k3RS+1iz4wBHpV7XIezZs0dZWVkaPXp04Fh0dLQGDx6spUu
XatKkSVq6dKliYmICZZ8kjR49WlarVcuXLtSlKSIEYGyT5LGjh2rJ554QgUFBYqN5T80AABw/KxWa
2AFAgAAR8vvN7R410G9vzJdX23OltvLTL5gklFUqenf79H07/eobUyoxvVN0rg+SeqXEmN2NOA31Wvhl5WVJUl
KSEiodjwhISFwLisrS/Hx8dVD2O2Ki4urNiY1NbXGNX46V1vhV1VVpaqqqsD7xcXFx/nRAAAAAABQU3p+uT5cv
V8frd7PzrotxIHCCr3+3W69/t1udY6P0JVDOmhC/7aKDHGYHQ2oVdA8aXjatGl68MEHzY4BAAAAAhClR6f/rs
pSx+sSteSXXk69odjobnbmVOqv322SU/O26rxJ7XVVUM7qltipNmxgGrqtfBLTEyUJGVnZyspKSlwPDs7Wyeee
GJgTE5OTrXXeb1e5efnB16fmJio7OzsamN+ev+nMb92zz336E9/+lPg/eLiYqWkpBzfBwQAAAAAaNE2ZxTr3RV
p+nTtARVXes2OgyakzO3TrOVpmrU8TSd3jNMVQzvo7N6Jcthqf84r0JjqtfBLTU1VYmKiFixYECj4j4iouLtXz5c
t1OOO2SpKFDh6qwsFCrV6/WgAEDJEnffPON/H6/H6/Bg8eHBjz17/+VR6PRw5H7S/Xvv32W3Xr1q3W5bwk5ar
PDwcAAAAA0EIt3nlQry7apf/tOGh2FDQDK/bma8XefLWJdOmvZ7TTpq4MdEg5aRNLNprFNPMC0P3WRi42+TV7sKZ7MLPm7OfAzgjFk7hjTvlVJNUWGo0+5RDdozGWRXWVHd/XrfvOlR9HsnEdXlyeI1wt1eyt/tM2T1pfkjPTIIV1HEHTm+OQbB1JJgAeA6V9ZHx5KEm0URTdkTwOzQTWL6iVdVQ8MbH5IvDezBmoxvMv9t7QsEbn5WSH1V/cZd8ub/kE8HzaUncbitbyEkrKDDFf3WZYPbvkT0B6vM9V4F4MZ4HVsxvvdmTx6Ir5T9YTgVQ/GLrrqtZZRgd89Ybz9ubUw3v7o7IcnQMiAJZQ8cp0m3Um03NzNh5i1VkLz1RZkc6I0zdPUzOA8bT6v9RHWq58cbzm
6xLmwxTyA9HWbHtb24yX5Ukrhu2TWgzdZgEDZ7tDa9aBNf6KJ1xKAgDbafZ/IFDV7f9ZSn9VOHjjOq/zlXrb/AlzDc5ZepOa46ZiEiTpVg9VnyeHMCl4h+bUjGtRgVTgn+3gs9rqKfUVmyRetDAhVcb9rwnE/9JrFxN4uSRp1EbZxJn3AYqg2YdVDydf5DEW0GWrQu+PPZcyyjAgrJh4pWHHoFq/rvWQZmcq7PdqQEJzB1UKa4AAmU9DkhzkPZRaPrZvqZt1+TAAQUyUxHQVdEFPGHp23DvoK1vnXZoa6E/fNp2NSzd3GV8WyrnLwy+S6ZIg9CJ7M6Ybd41RMK4sR5JzcQWj9TQB6hW4N6Uqy3m5pq6X9jkg2rZsMnsvNOl6gXOOtB9gDfNqXu/Wi8/7ubWRMRLKWP0Ap67Eakl9Ud9NNNq+ILd2H9z5OEzXXt+h9hRsH9xUFrnO5AGa9HVpJqeDcl9Tu/P4ebpv4+ZbUbbUEpU/Ai3JlIX6a7+KPS8pI7M5yN2v03qtGUz7cp0wC7OsOm8vdB0YKMn+C/O/wJc/AzR3kpFjr2fwtOXo7qldlW92TD9V9rDXJ/vPcTM1ZSoCJ7fwr1fsfRZf1oE2HFCefl+/o5r5pkQyFSlMdLSFovuMgZBPJFlEoZyf+qcLG1uLqQRBfp5sBqD5nxPP8sbOu9/Wr8Hfnr1mIbMrKHM8rP+gOzQeGf8H+MwdgyRbJlbvbQd3lhqgSHJEb5uBHpJbF2uMQa8iERqG8T8VIaWz0AuhJHkmIpUt8PA/XmW3dVsD4V6WWVYtDbPfHRKvGxp/vTUV3/l41T9tlOfU0ilbqzWeOXmm1G8zc8dYRlrT2zrltDxSsRiUCVXc8d5LSV1vvVuR/kH4TtLW9mBX1eF/XuPd3bmn0vRCkHXrO9vmrTHv8m3CbHVGf9SndYa7Kd8V5ZvLOOT7bnfo8XF5MLjyGEE90JnvjC3YbBrldW7yC6ML3SN/VZY8xPgEuVEL2T9LS2YGt5hsvYD7pGNdYlCeUKl/zGsr8l3DsOxW+kh0x7W3ZcW2l1FZ9S0xfmDndPrgmt6qsDJMSeqtzHf4mm7DbiJSRYRyzMN8e5RwfZMN8y8J+99Mf+zrqv5YDpCRvgtzfeDRpXbXsvaZMDJQL/7DRSfb0aV8sN60kPvfd1QnWa4evXRXBTgNaezx3iVgeu9t/oXrqR0Hxn7xUe9A4gvW7ZRmajl5zSwAyRQOVo1qz9tVxXDtHc3mzQMkbBgmgYTP//zDPVW0QMfX8tLNcRHN0TdkWuAAAAXrK9bXOVx6OCnWFJUXQVyV2o/H0kSn3Hvpi2FHn5WtIkUhlDbVZvCuFyS13vHYBO5HlMnCyZ8DiSxu1TwdrBPgSFaiSVb3jZ3VTNMxmqaXK8WODGx1QnSfQPDFwuoSZvfD26+zzuYQkYHM8JZ2wmLZkpWiNWmyDKfzB0b4FBiFjkK9LnVyIZlGcRlXeFq3sVHqcPCP8f1pkk8a9e/vzBztGQhF3r6QAA9F8k+nWeWuzNXi3oZOX6W8kD5WGbP6cHQ+Pt7lnPw2QdTxt5RHu53jNxQTlu/KsqD+ckSYTzUzbNTtrjM5Rp5+ppwHDvDIHYF5q7VK4R0rUCVMFmDIPKF5qTqV+6zlm8z8ldTIuK4p8FkiezG1NOOcBnp3v0FEFTPVJqGtz9izsxU1h2nmzZAZDZtxr+LCPWNB3PEBV/9sHp/KpWx8OnOdbT6qjCfyY1o/wC7SQXOfvBVyL9p7HX7vwnZ1yJhjXtWZGQc9pe1HPT3Lvlhr2lz97aMeR7cqSXo0k4V+V1tGcZ64GOJmTuNvh1O9pWpLk1ahsxsJKWTKtl9sZSEdu29P30XHhMEvd5KP05Q/xH5e5J9kjkxNDKvGPtmpo9lDqjzZgo8/1+2YWg3j1PDPQuT9rWmDm6dTnpvnxmrqsbmCaVwKAZGb5Jds2dHUHKqZQWvtBQaBgE13Ryqk4/BEbwaNfwjpwrvUjAHhCv/x1ROymlc/bGHfdHh3sr+GrS5TG6vF1XrptVgH+73HwH1H9EXWwSYX4LxNoNzyrq3B5CvDOZs0Ez4ok7Ez6+M2tAdBpHqZPfqrhnJ3cfhI7MZiSTc9B9F8sFdAkD7zxyn1wHQVwM6MVsK7kl2eNCyq4Xp+f/FiQ2PlAvEajaRUsRSMHbVZeGBKYh7kfCTAZxnb9/KQXq5F7l4wh5xqK/1RtqZjcDJPFrbb2YqVU3K0u8bSsXkcUAlKc+2HQVXDyfi16SNpTn3SB4DHhU8T8/rA8VrTd4sU+RoiIdFONBSvWS5NXOL9ip8SGG5CQQH3B9W5JANCpP3mKwi5vZfoe9YuDPUlCYUSVcV90rXOUvpOF6b5q+TRtgBvk5SaTiUKpqlyh6akAMS2vFXUbv6kX1eJMwIcSdPPQO8KvQtR4PzVGxpz7twhP3+mDXvS7DP8Eua4f0aUbB+2IO2iQgyfEfXOAjvWDHZfeH4y5sW30qiobQAEjQqh+7bJ1sNpKOxYALdHx5DXBv/qcKpG6t1d1vRYgZTn5XG5tIG2bGRCjQrIGupVB7ZY1fh6Gtel8wFdh0jzQL1OY8JGsY/V1TJAsBkQ3Yb/7Pk9eBZOc4Az4mgJlI68MwCeVyFgBYdW/7jaqq9e4uDCrbkULk5uqU7o95Fn4xPPsYfYl3JABoT08ZBMR80IvFXljArW0G4BlJK58Z5d9l2G3v71Oj+HZ5fNQzfJ4+1cHbT03tC1yAHEuP9GWY3mvuuBBlqjuuKBjC9b5AmQLGlRKsmq5eh7S5oPcTKHmOhtq+09cq1bJTfmkLg52aAYZLn3G0aRdlm7UOCcIxAKeXqF6ApUxSQ8ykWWZ2Ub2dM9tpMHk9TK3mXhXJB8h4HMNEEYONVHwVT7aKjUmyvXm3ZFxzjWf1DqLqE0HKgsLnBgnYOlJOHCqo6G7bJjKuTRfNQWTV6VaNUGjPVs7sM0ipGu0wQvlP8ufVvOqTPkFGlD6lcKqLxLdKmz4tCCrEN4Gp9tR5UOoKJrkuScvoDDO5nfqoLn0NW1pj4CECzbQ0ZxuHf5qzRGx7SFfwF1y6mjLsVvZTXXLZ6R6BHvHO3UvPqTUG9RDzhkszeEKaDL9AY7aAUrXFMoWRpozFnUgBjXaXBtWMjvDYAH05x8xCNUNFhLqPQBsPjYIvSrBFNzcCeI7e5kaQzi7UxNHUHXcBqoqqJ0DAqzm3H/OqQBmRGSckmxQEulzzVaS3nzBTZYpCjgYbgj3NUYWXLCmFoMUNUKtlumKbAo+nBemsFzrg/BvCGlQEF8GeqxgIjkpdqzQCMTKdAW8j5xeUwCsV3vnyzEhmo7TVCnaupEpAWmAkZ/+dd62xFCewAjBzMfnwBm3ZIsqUQRpHBXOaHxjoNoVnOH3Nd4ohwVFkLhmU2NK1Cc5SOLhWfAoi3vfYMZkgEC6vuaFk9dlzaBjHNJEzsMTLkZ

1D2AS1AUYVHV05frl25pWZHwRGg8AMAAAAAHJW1m7bpr+8s1AZ/W8o+oAU5WOrWFf9crvT8crOj4DdQ+AEAAAA
AjtgPm7bqr7MWaaO/rdlRAJggs6hSk/+5XNnFlWZHwWFQ+AEAAAAAjsgPm7bqr+98pO2UfUCLlpZfrsn/XK680
iqzo6AOFH4AAAAAgN+OZuMWPfDOt9psUPYBkHbmlOrK6StUVOExOwpqQeEHAAAAADis1Ru26PHZ87Xe387sKAC
akM2ZxbrmzRUqq/KaHQW/QuEHAAAAAKjTqvWb9dzsL7XG1OGGhV8hAVT3Q1qhrpu5UpUen9lR8At8twYAAAAA1
Grluk166d25WuFPlddiNzsOgCZq2e583fTOavn8htlR8CMKPwAAAABADWs2btHr732mlf5UVVlcZscBOMR9uy1
XD3++2ewY+BGFHwAAAACgmh17OvTmB59pla+9Si3hZscBOEzMWLJX765IMzsGROEHAAAAAPiFjOxcTX/vEyOva
6OCa4zZcQAOM/d/ulHLdueZHaPFo/ADAAAAAEiSCoqK9a/3P9H3OXblOBLNjgOgGfL4DN3Ozmql55ebHaVFo/A
DAAAAAKiislIzP/xM3+4qOX5XR7PjAGjGCso9um7mSpVWec2OOmJR+AEAAABAC+f1evXup/MOf91e7QntZnYcA
EFge3apbn33B/nZudcUFH4AAAAAOIIZhqFPv1qo/3y/Srsjessvi9mRAASJb7bm6PF5W82OOSJR+AEAAABAC/b
N4hX6dP5C7YvsrQrDYXYcAEHm9e9269+r95sdo8Wh8AMAAACAFmrluk16/7P/KjOko3L8kWbHARCk/vLJBq3el
292jBaFwg8AAAAAWqAde9LO9kdzleOPOO6DHXkBNBy3168/zlqjgjK32VFaDAo/AAAAAGhhCoqK9da/5yqjsEL
bnF1l8Nw+AAOsu7hKd3603uwYLQaFHwAAAACOID/tyLt9zz6lR/dVpd9mdiQALcT8zdl6e9k+s2OOCBR+AAAAA
NCCfPXdUn2/co2KWvdVtifE7DgAWphHv9isHdklZscIehR+AAAAANBCbNi6Q3P++63cke2O1R1jdhwALVClx69
b3v1BVV6f2VGCGoUfAAAAALQAuXkFmj3nSxW6pfVGB4nn9gEwydasEk37cqvZMYIahR8AAAAABDm326PZc77U7
rQD2hfRS26DXwUBmGvGkr36dmuO2TGCFt/lAQAAACCIGYahL775n5b9sEHehJ48tw9Ak3HHv9cpt6TK7BhBicI
PAAAAAILYD5u26osF3yk8Nl7rK1uZHQcAAg6WuvXnD9fJMAyzowQdCj8AAAAACFKZOQf13qfz5PX5tdnSUR6W8
gJoYr7bnqvp3+8xOObQ4bs9AAAAAAShn57bl3YgU57WXZXhdpkdCQBq9eR/t2l3bqnZMYIKhR8AAAAABKEFi1d
o1frNSmrfUStKos2OAwB1cnv9uufjDSztrUcUfgAAAAAQZNIOZOrzrxcpJipSqyoSWMoLoMlbvidf769MNztGO
OC7PgAAAAAEEY/Hqw+/mK/8wiKVRbZXehW78gJoHqb9Zyu79tYTCj8AAAAACCIL163S6g2bldiug5YXx5gdBwC
OWFGFRw/O3WR2jKBA4QcAAAAAQWJ/Zrbmzl+oqIgIramKVxVLeQEOM5+vz9TCbTlmx2j26v27v8/nO3333afU1
FSFhoaqU6dOevjhh6s9eNEwDN1///1KSkpSaGioRo8erRO7dlS7Tn5+viZPnqyoqCjFxMTouuuuU2kpO7YAAAA
AQG28Xq/+/cV85eYVyBvTXvsqQ82OBADH5IHPNqnK6zM7Rr8Nmr+8LPvHEE3rllVc0c+ZM9erVS6tWrdK1116r6
Oho3XrrrZKkJ598rXlOs8//7xmyMjZMkrdyqJBDz5eYPHmyD6cYbNXvO8J/2a6yrd8J8PnUWhqf8WNu
27PqODAAAAADN3v9W/KCV6zapfbu2+k9Vd/8rv5OvuOYMoYiTxqnVmJtkeN3K/2a6yrd8J8PnUWhqf8WNu
Um28Ngjun7ef19U6dp5ij3jBkUNuiBw3JN/QAXf/ktVB7bI8HnkbJOqmFOvUEiHvpIkXOWJ8r54RpVpG2SPTVb
rc26TM6HTz9f96hU5YhIUdfKE4/wMAM3b3rxyvbZot24d1cXsKM1Wvc/wW7JkiS644AKNGzdOHTt21EUXXaQxY
8ZoxYoVkg7N7N7nvuued077336oILL1Dfvn3111ltvKSMjQ3PmzJEkbdmyRfPmzdM///lPDR48WMOHD9cLL7yg995
7TxkZGfUdGQAAAACatYzsXH361bcKDwvVXq01in31PrejWUm6+lm+lm1m/J24C3+0kckSeHdhOmS8he8oYqdK9R6/
N1KuPxxeUvzlPvJY0d07fLtS1SVsU22iLga53L+/aDk9ylh0qNKuvo5OeJTlfPRg/KVFkiSipa+L7+7QknX/EM
h7fsob94LgddWHdgqd+Y2RQ68oMZ1gZbo5YU7lZ5fbnaMZqveC79TTz1V8+fP1/bt2yVJ69ev15IlS3TWWWdJknbv3q2srCyNHj3afCc6OlqDBw/WsmXLJEnLly9XTExMVdEnSaNHj5bVatVPP/1U6+9WVlaquLi42gIAAACAlsgwDP37i/nKyy9UZKu22lHZyuxIaGI8hqEXF2w3O0azZa/vFe+9914VFxerR48estls8nq9evTRRzV58mRJUlbWoffu4uPjq30XHx/v/y4rK0tt27atHtRuV2xsrH+dQ82YMUMPPvhgfacDAAAAAE3O+k079O2PP6lDSrJWeTvKwwxeVtTBR2v36daJ3dUuinf96lu9/y/yww8/1Ntvv6133nlHa9as0ZtvvqmnnnpKb775Zn3/qmruu+8+FRUV+Zf09PSA/j4AAAAAaAxud5Xe+3SeXC63yiO7ap8nyuxIaKIqPT69vHCn2TGapXpv8bv77rt17733atKkSZKkPn36KC0tTTNmzNCUKVOUkJAgScrOzlZiYqL/e9nZ2erfv78kKSEhQTk5OdV+r8fjUX5+vv/9hwoODlZwcHB93w4AAAAANCnf/bhGazZsVkL79vrGw5i9OD4frEzXLRO6q3VYiOYOK13FY8BiZm0wYIlS7nH4dPPHOJgMkmYO6eW+3X+Gxt6z5JVtuaC/6cZxMBAyQcTWY2lQMhEyS0bxaYCprilEqHzAvbe6ccR1/KoS9tQkZUhtR2Xtp2dKO77cHN1vyQ6y6/LxpS/91L7+5j0tL36/Rx+sjpAJN6//jD/w76Vr0T+bnJTP7frl6tb76eBsVqIa0z7p8zT7m8Xa9P4tq/SCCy44/SMeO41F3o95/KT9WZuTdEyq4HwPX/w7ebj3eUvk9nb6qqdS/32EL3yQJTTq+5tu1Mp5Of/77R81t+p8cNO5C8fZ1nLvcTwvt6eYp0zXm6QTtmdTJ0/t0DX4qS3fgg/P/6dvT2SVjL/h7Vb8v3df8/07P/7JlWa1ft0ee53WfZvfHxxsfz0C3N33rvnr/vfL2jHdJOTUmPNTeM77HPm2svGLzFz91cVf1xevyGk+vrq3l/6vf1EvZyHcfX+p/tp7/0NVQ+E//h9f7vKvk0tdnafLb66hVePrPD/+0j0rj7d8/9SUf2+U/9RNPf+zlOU/0dJzyfxQn/mFVOV/eH/yl3wn//8W6/+3W1VSnofv9L7DVZp/dW//9/ePT+fHfI2feN9OKzv2KEX9xn9V4t+sl6m3/2Yp0KXvuN6f/o8z/q1X7+5/AQbp/8A79//cdLPffRpd//f8+fp+ff+GT/z/Pq+s+4/8XufnD3u+Pf/IP/Mz+nl+/qF/v/+Iv/ff8LP/Pf/Vf9fn8/L8//YvP1P/Y3c8HNF/+7H4n/h+f//FPJ+/9X3/+c//iP//E//5X/nN/XPb/f+SnT/1//+D+lT/8/X2zv//9Z/1/+73//ev/z//v/+/f//5/f/+//X9/v///ZZ/9P9v/Xef/8v//f/6//+f//3Ov3/v/fP97P3M//ff8/o/+/X/f/X/z//Y/zy9+/+///7/v4X/X9//99a/7f93/Fvu/n/P/73//r//8/+/r//T8/P/f++f7/f2P39//9f//X/vv//1v2/f+//////Xv//X/o//e//3f/P/zP///v//7n/9///1v+fX/2/f+P/7v7//3//X////o/u/8//t/+//+/x/fmf/vf/W/3//r7//5/nP//3/+///9/X/P/+///7/t/+v2/Pf//z//b/1v8n+f///7/7//8+//X7f//59+fv/3f+/7/89f/9//n///P/P//8/+nef/8/9v/+/P/9//73+/3/+/3/8+f///////f//+/f3+/z/f//n//33///zfv2/3/75+v/+9///9//X/33//X/7//H//1///9//f+3+/3f///P////9//X/3//H/+///5///x/f//f///f///+///X//3//3///v///9//f/f/////////////////////+//v///1/+/9/////9////////f//3//

```
3ZAAAAABoVgzDOJfffK/8wmL17NpJn+Yx4aE2pRvmyxYWrdBOgwLHXImdJatdFfvWKbzboV17PXn75SvOlSu59
uf3hfc+XSEd+1U7lvPB/QrvdYYi+hzakNLw/vhM+V/PALRYJaPmckRfeZEKl7ynxMlPHDrg98vwe3886ZVhUG4
AP9mXV673V6briiE1n5+J2tV74ffCCy/ovvvu0x//+Efl50QoOTlZv//973X//fcHxtx5550qKyvTjTfeqMLCQ
g0fPlzz5s1TSEhIYMysWbNO8803a9SoUbJarZo4caKef/75+o4LAAAAAM3O5h27tWzNerVLStDOyjDleZxmR2p
yDMOvOg1fK7z3qGo731pd4Yroe6YKvvmnbCGRsrjCVDD/VbmSu1fbsOPAG39Q7GlXKazrKbKFRskW+qtS1WqXL
TxWjlbtJEmu5O6yhkQo74tnFT1skix2l0rX/VfewmyFdhpYI1/+gtcVNWi87JGtD72+XQ+VbfpWoR37q2TdPLn
a9myAzwrQfL3wzQ5dNKCdQhzsZHOk6r3wi4yM1HPPPafnnnuuzjEWiOUPPfSQHnrooTrHxMXFafbs2fUdDwAAA
ACaNY/Hq8+//k5VbrfCI6OOKofZfbWp3LtWvuJcRfQ9s8a5uFE3KN9iVe6cx2T4PApJ7a9WZ/6x2hhv/n75q8q
P+H62sGjFX/ygCr97S9nv/lWG3ytH6/aKn3CvnPEnVBtbsXu1vAWZan3unwPHIvufK3fmTmW+/Se5kroqZthlR
/kRA8Etu7hKM5fs1e9P62R2lGbBYhi1zCOOAsXFxYqOjlZRUVGjP8/v422ZjXo/BL8J3ZLMjgAAAI6AmT+DouV
YsmqtXn7rA7Vvm6Qd3tZaWcLXGoCWISbMof/deboiQxxmR2ny6n3TDgAAAABAwygtK9eX33wvh8MuZOiINpaFm
xOJABpNYblHb3y32+wYzQKFHwAAAAAOE4uWrdKufelq3zZJ28vDVOHnWVYAWpbp3+/RwdIqs2MOeRR+AAAAAANA
MZB/MO1ffLVVMdJRsNrs2lEaYHQkAGl2Z26eXvt1pdowmj8IPAAAAAJqB+YuWKSs3TOnxrbWrIlQlvnrfgxEAm
oVZy9OUWVRhdowmjcIPAAAAAJq4A1k5WrzqByW2aS2Lxar1zO4DOIK5vX7NXLLP7BhNGoUfAAAAADRx369Yo/y
iYrWOi1FaZYgKvOxQCaBle3dFmsrdXrNjNFkUfgAAAADQhOXk5et/K35Qm1axslgsWsvsPgBQUYVHH63eb3aMJ
ovCDwAAAACasKWr1ulgfoHiW8XpQJVTuR6n2ZEAoEl4c/FeGYZhdowmicIPAAAAAJqogqJifbfqB1WrW
uJNLsSADQZOw+WKZvt+WYHaNJovADAAAAgCZq2Zr1yso5qMQ2rZXjdijD7TI7EgAOKdO/32N2hCaJwg8AAAAAm
qDSsnJ9u2SlSloiLDZbNZtYFn9wFADYt35mlLZrHZMZocCj8AAAAAaIJWrN2oAxlZSe0UbnPqn2VIWZHAoAm6V/
M8quBwg8AAAAAmpiKykotWLbxLxcYaGhstvt2l4eJr8sZscCgCbp03UZOlhaZXaXaMJoXCDwAAAACamNXrt2
sbLMKSt5WFmRwKAJsvt9eudZfvMjtGkUPgBAAAAQBPi8XiVFyOp+AIRRrFOZSUbOEGSqPRZPqQVKYp8c8NrX+Ru
Gkt9qOTvOS8KNOPkgRRsfRGNLgHsTgmCWVLSZfIAh29ufAVgABHKAHGoBR8Ek9fU4cU5DpOBZd6JGaTH9A/32N2Q
8iI6+SpQsjGJRzPxL3gHUAsLQ+NcfGUcOHP8SD4PU/WODvzmTAbMIDjfOBOaLaq2KEMF9QJZPSEIWUGbGxTDHDRSA
ThOjCf4ajg8AiN5PaiFnZ2mp9R4EeDQwvKVAOmG4D+gNEH5DwTQJhHwrrwkwTI0XIvTHJYmD4KGqgKXDWkUOu+I
9Ag9e9XKfMWEPBTcJhJCBIUgH1MhJA==
```

3/+ub777jvdeOON9ROXAAAAABpMXkGhnA5HtWM8ww8AmobC8uCd4Wf/7SFH54knnlBKSorefPPNwLHU1NTAPxu
Goeeee0733nuvLrjgAknSW2+9pYSEBM2ZM0eTJk3Sli1bNG/ePK1cuVIDBw6UJL3wwgs655xz9Pe//13Jycn1H
RsAAAAA6l1BUXGNwo8ZfgDQNARz4Vfvf1r67LPPNHDgQF188cWKj4/XSSedpDfeeCNwfs+ePcrKytLoOaMDx6K
jozV48GAtXbpUkrRO6VLFxMQEyj5JGj16tKxWq5YvX17fkQEAAACg3vl8PhWVlMrprD7Pgmf4AUDTUMgz/I7c7
t279corr6hLly7673//q5tuukm33nqrZs6cKUnKysqSJCUkJFR7XUCQuBcVlaW4uPjq5232+2Ki4sLjPm1qqo
qFRcXV3sDAAAAALOUllfI7fHWmOHnpvADgCYhmDftqPclvX6/XwMHDtRjjzOmSTrppJOOceNGvfrqq7r66+qvr+
3YBO6ZNO4MPPthg1wcAAACAo1FSWia326OoiPBqx/2GpY5XAAAaU1E5m3YcsaSkJPXs2bPasR49eigtLU2SljJi
YKEnKzs6uNiY7OztwLjExUTk5OdXOe71e5efnB8b82j333DKOioqLAW3p6er18PAAAAABwLMrKK+T2eOROVp/h5
zMpDwCgOpbOHoVhw4Zp27Zt1Y5t375dHTp0kHRoA4/ExEQtWLAgcL64uFjLly/XOKFDJUlDhw5VYWGhVq9eHRj
zzTffyO/3a/DgwbXe1+VyKSoqqtobtobAAAAAJilpKxcXo9XDnv1hVU+ZvgBQJMQzJt21PuS3ttvv12nnnHKKHnvsM
V1yySVasWKFXn/9db3++uuSJIvFottuu02PP68yRqqmCUw0Lz/JyckaP368ooF/88pEPf/UdFwAAAAA
aRJXbU63sk5jdBwBNSaUneB+yUO+FnySde+65O8vfcc+s8b7FY9NBDD+mhhx6qc0xcXJxmz57dEPEAAAAAoMF5v
TV/kQzeXyOBoPnx+w2zIzQY9oMHAAAAgAbg8/tkGNV/mWSGHwA0OU9zoG
2uM81eWquC7t1WxfYl8lSWyR8UrbtQNCuOOSJJUuulbFS6aKcNdofA+oxU36oaf71GUrez371PS1c/J6gqr/w/
aJMG8pJfCDwAAAAAagMfrrXGMZ/gB+C2Gu1KO+BMUOfdM5X7y2G+/wOeRLSxKOadcquKVn9Z+TZ9H2e/fJ1tYt
FqPvOf2yFbyFuXIGhJ+6BLlRcqf94JanTNV9phE5fz7QYVO6KewzidLkvK+elmxp1OTVGWfJPko/AAAAAAAR8P
t9sjKLrOIUg9GvaLPwvxyVvoVXW5XVIVVkZUWRZZLEZWGIip8Cq3OKazSo9BKt+xer/jqPOr792mIpP/b8KFO2
7/g8GOtknJydVNFrrruX6Hbv8mpdvrjnGzNKsjQ+237yb794xov31Raqjtk6PPM76RM6a8um3osn6ErOr7SV3k
HNT8/TO8d+EY68E39fXyNzOPxKiI8TCEuZ+CYrWOqpFPNC9WAKPwAAAAAoAF4vF5ZflX4Be9cErQkKdY8XeX+X
qN94bo1oasWJx34zdeEeKyKLXcptsKhmHKHoitsiqqwKqrcoohKQ+EVhsIrfQqr8Cqk0qOQSrdcVVWy+v2N8BE
1baHlZYqyHdkWDHavVO53laKKC6sdX3owRyc5XfrHzm36prRUsTabxkVF6fq4VrJZLOrp86nK79P+nCwlOxzaV
lqiS8PDZBTk6Y3OfZqROr7GNZulitJq34dtUZH1evlpO6bp448/1tatWxUaGqpTTjlFTzzxhLp161bna2bMmKFu
rr7222jGXy6XKysrjykLhBwAAAAANwOPxymqt/ku63ULlh+ZvvHO1LDLU1leqjzLW6OnYPno7ukw+S8l17D+pd
PiVGV2hzOiKo7iTVVGVLsWWOxVT7lBshV1R5TZFVVoVWSFFVBgKr/ArrNKnsEqvQirdCql0y1FVxWzCX9nv8Wi
5p1znRkXp1XYpSnO79VB2lryGNKV1a0XbbJqWmKR7MjNVafh1flSUhodH6N6sTE2OidUBj0dTDuyX1zAOpXVrj
Y2MMvtDqh/1/IWyaNEiTZkyRYMGDZLX69Vf/vIXjRkzRps3b1Z4eHix3bHidr4uKitK2bdt+jmU5/mAUfgAAAADQANx
eCj8Ep9Osa6VfTLz7c8EGnV4WrzsSEpVjL6zXexWHeFQc4tG+uCN/jdVnV2zFoaIwtsKu6Aq7oitsiqyQIissP
xaFPoVV+hTy45JjV2WV7LU8dzNY+A1DcTabHkxIlM1iUa+QEGV7vfpXfp6mtG4tSRodGanRkT/PeFtZXq7tVVX
6a3yCztq9W39PTlZru02Xpu3X4X7tungaFhamVv/pWSxWqr1+vNmzev2vszZsxsQfHy8Vq9erREjRtSdw2JRYmJivWZp/
v92AAAAAKAJ8tTyDD8KPzR3dn+Veng2SL/qSfq7c/T5vlzd4krV8rbeep85dTT8Nikvokp5EVVH
Fl1gVVWpVZJlFUWWHSsLICkPhlYYiqgyFuw2Fu3OK8/oV6vPJbjT9/67b2O2yWyyy/eJ70gl0pw76fHIbhpy/+
l7l9vv1UHaWnkhKVprbLZ8MDQo7tGFHR6dT6ysrdHpE/S6HNYPF6fztQcehqKhIkhQXd/jGurS0VB06dJDf71lf
//v312GOPqVevXsd1bwbo9HFiuFH4LLsLJvFR5W+37ToVZD//Ts1rtbIvRUSqw8kc1rYa3bJR1s4
9PBNkezn7ZN4SVSTLFFUcVSVMmPJWGZftzARL8oCQ2FeQ2F+QyIeIfR632tjsNco
+SXo1P0/Dw8PVMyREmysr5f1FqekxDPmC5FuZxdVwhZ/f79fUqVM1bNgw9e7du85x3bp107/+9S/17dtXRUVF+
vvf/65TTjlFmzZtUrt27Y75/hR+AAAAANAQavmFmIPzd3ZEet+c8xloaUamVOhG0qTtC/pyDabaM7KIqWySEM
H2kqH35rHop+mPlp9Uqtyp1qV2xVbbldsuU0xFVZFVUj2Yp9KDpYrxG1I26SdDptaWwy19vnVzmrTM7k5yvF69
XhScuDKW37c4KHcbyjf69OWyko5LBZ1drkkSZNiYjS7sXOEP5WTritgg47XO79XpenibHxtZIub0qSv8pLtZHHVM
lHZoJaLVY9FFhoVrb7drjdqtPSEh9fOpMZ3W6GuzaU6ZMOcaNG/X9998rJSXlqNbDi8qNpGIz/L8ZFM
0OPP/zwMd+fwg8AAAAAGoDL6ZThhrT2fdtzQoUM1dOjRaNG/X9998rJSXl/AGoDL5ZTP769x3G4x5DEo/
ND89LGlq6Oj8qheM6Vgu0aWxepPCW2VYS9uoGTBJaJHhHrPqHOJaIX8ipuSJElaqKLA8d6Dax9/pXySrIqucKp
VWaT6VyRrVLlN0eVWRVdYtCz/hVR69XpUhBxVbhluT+BPEyMjIjQyonN9fphNgrWeCz/DMHTLLbfok08+0cJXv4fQ
O8+0cKFC5WamnrU1/D5fNqwYYN+97vfHde9KfwAAAAAoAGEhYbI4/VXO2a1WuS0O+SuZbs+gKYgovWLr+/qCYmQSTCS5Pl11WRWuk+YQHqUkOEFBbYEJ3mKuIYLAGQb/5+Xdfh5KHPpd/YTAWy2eF6XxqFPNpnEl8bJzz

C3deml74ML5HBOvZGVxTqVVGoV7uP4jVWn73OZccRFRZFVPgVVmkotNKnkEqvnFUeOavcsnmP5hmI5qrvwm/Kl
CmaPXu2Pv30UOVGRiorKOuSFBOdrdAf73XVVVepbdu2mjZtmiTpoYce0pAhQ9S5c2cVFhbqqaee0r59+3T99dc
fVxYKPwAAAABoACEup/x1zPADmqPh1vXSMXY5Dhl6PHejRpUn64HWkSq2VtRvONS7wy07rptFoR6nWpU5FFduV
1y5PbDs+FBRKIVXGgqr8CuOOidXlUfOSo8cbo+stXy/bGjWyPrdafiVV16RJIOcObLa8TfffFPXXHONJCktLU1
W68/PtiwoKNANN9ygrKwsxcbGasCAAVqyZIl69ux5XFko/AAAAACgAYS4XLVOhqLwQ3MUaSlXR9/e477OmWUZG
lDhOuOJ3bQmpPC4r4emp8Lh1/6YKu2PqTqKV/207NihuIofZxOWWxVVblFkZc1lxyGVXjmq3LL/YtnxsbBFRR3
Hq2syjN/+/r5w4cJq7z/77LN69tln6zWHROEHAAAAAA3CbrfX+osohR+aowmuVbKrfjbeiPNXaWbmev0rprNei
jHktnjq5bpo3up12XH5odmEEZXGj8uOvbIWlyvOGSZ7RYWMikPPorRF12/h15RQ+AEAAAA3DYa/91y2Ft/GV
rwPEaY1tzzMt56/K7wp0aURal2xLaK83Bhh44eke/7LhEkhRhhCnJG6m7TmqtwQ0Xz1TW3x4CAAAAADhaDkfth
V8ohR+aob7+rQ1y3c6eYn26f5MmlEbIalBRoHGUWtza4ciTKzLG7CgNhv+aAAAAAKABOB12qZZFveG25rODJSB
Jw+zbFWkO3Aw8uww9mLtZL2RXKtYX3mD3AX4tNiTW7AgNhsIPAAAAABqAw26vre+j8EOzc4FzRaPcZ0RFlubu3
6khlTGNcj8gLiTO7AgNhsIPAAAAABqA3W6XatmxkcIPzc1QbWiOeOX7PXojc73uzrMrxHA22n3R8jisDkU6I82
OOWAo/AAAAACgATgdDkmS8avSL4LCD81IgqVIbf37G/2+k4t366P9WTrBE93o9ObLEMzLeSUKPwAAAABoEBHho
XI6HKpye6odD2fTDjQjF7mWy6qaM1UbQ3tvqT7ZvOGXFOfIxoYeqGfxofFmR2hQ/BcDAAAAAAOgKjJCLqdDbre
72vEQml82kwoU4GidYVtr6v2tku7J26zXs8rU2hdhahYEl5SoFLMjNCgKPwAAAABoAFER4XI6nTVm+Ek8xw/Ng
1U+9fRvMzuGJOnkylx9kbZNp1VEi74c9aFDVAezIzQoCj8AAAAAaAAup1MR4WGq+tUMP4nCD83DGMdGhRoVZsc
ICJNPL2Zt0IN5FoX5XWbHQTPXPrK92REaFIUfAAAAADQAi8Wi1rExcjPDD83UOMcqsyPUakLJPs3Zf0Dd3TFmR
0EzlhLJkl4AAAAAwDFo0yqWGX5otk7WJrMj1CnJV64PD6zXdUWhshs2s+OgGWJJL1okn8+nd//xpG4aNViX9Tt
BfzxzqD58+VkZxs8PS6goK9MbD/1FN5w2QJf100G3jTtN/33vrSO+x/dfzNHE7sl6fMq11Y6/cPdUTeyeXO3t4
esvD5z3uKv0jztv0RUDuurmsc01bsl31V4/Z/rL+ufDfz3GjxwAAACoPzHRUdV+hv4JhR+auk7WbCX4s8yO8Zu
m5m/TjMwiJfgizY6CZiTSGanYkFizYzQou9kBODTNeeMl/ffdmbrl8X8opXM37dq4Ti/+5XaFRURq3FXXS5JmP
P6ANi5frNuefEHxbVOOdvEivfHQPYqLT9CgM8Ye9vo5+9M188mH1WPg4FrPn3Tq6Zry2LOB9x1OZ+Cf57//jnZ
vWq/H3purH777Rs/93xT9a/F6WSwWZe9PO9cfzNaTH/2nHj4LAAAAwPGJigiv9Xi03dvISYCjc5FrebPZHKNfV
b4+TyvUX+N7aX5YsQxLMwkOOwT7cl6JGX6ow7YfVmnQqLEaMHKO4tulaOhZ56rfsNOOc8Pan8esXaWR4y9W78G
nKL5disZceoU6duupHevX1nld6dDswefumKJLb/mzEtrVPoXW7nQqtk184COiOiZwbv/unRp4xhi179JNZO2+R
sX5eSouyJckvf7A3bry//6qsAj+ugMAAADzRUUeKvx+PcsvlsIPTdwI6zqzIxyVEPn1dM4GPZ7rV4Q/1Ow4aOI
6RAb3cl6Jwg916HbSQG1Y+rOy9uySJO3duklb16zQSSPO+HnMiQO18puvlJedKcMwtGHZYmXs3a1+w0477LU/f
OkZRbdqrdEXXV7nmE0rluraU/rolrOG67UH7lbJj4WeJHXs1lNbV69/ULFtEhQVG6fv5n4sh8ulwWe
efZwfPQAAAFA/oiLC5XA45PZU37gjzOaXy+I3KRVweCFyq6vp9ZZemam75PGdEdCIy5/r5H9Op9d+rGOwbIZrfLYrHqpoefUq9BQ+q87pbVy7Xgo/f
O9Jyv6hxzOqkjNWTM2Ypv215Z6Xs1+9nH9ciNV+ix9+bKZrPpjImTtG/7Zk0dN1KRsXH0GwbIZrfLYrHqpoefUq9BQ+q87pbVy7Xgo/f

```
XtAY5voXH7oyzIInVGeoZPS8/SnhG5aFVJoapbQzqEqWFigqqwquRJdqkirUNmOMiVNqvsZ9CVrSxSaGqqDXx5
U4ZJCWV1WRZ4UqYQJCbI6DO3GKV5brLDOYcp4OOPFPxTLHmlX9JBotRnXRharRa4El/xuvyr2VcjRyqGKPRWKP
TVWvjKfcj7OUce7OjbSZ8A8vVv3NjtCo6DwQ62uv/cRvfv8k3r9oXtUnJen2PgEnXnplbr4j7cHxtz+zCua9cx
j+scdN6u0qFCtk9vqsql3aeykqwJjDmYckMVy5JtBW21W7du2RQvnfKjykmLFtklQv2Gn6bLb7pTDWf35Jmnbt
2rJvLl6+pP5gWNDx56rTSuW6r7JFyo5tZOm/v2l4/gsAAAAAMevTatDz9by+/2y/mKFjNNqKMrmU7GPX8vQdAy
zrA/awk+SYv1VejNzvWZGd9ILsVKVxZyl9W3GtZG/wq8d9+yQrJL8UsLEBMWcElPna9w5bpVvL5fVYVX7W9vLV
+JTxlsZ8pX61076doExnoMeRQ+NVsc/dZQ7262MtzIknxQ/Pl62cJva3dBO+9/YL8NtKOaUGEX2idT+6fsVNyp
OnoMepf0jTYbPUPz4eEUPCr6ZcH3b9DU7QqOwGL9c7B1EiouLFROdraKiIkVFRTXqvT/eltmo9OPwm9CNnYYBA
GgOzPwZFE3XvvOZevDZl5QQ3OrhoaHVzi3Ij9WeytA6XgkOrmhLmda4/iBbMOzYcQR2OSI1NaGj9jqKGv3ehcs
KlfVBlhIvSVRI2xBVpFUoa3aWEi9LVOzw2Fpfs+epPSrfXq7u/+guW9ihZ+wVrSpS+kvp6vlaT1mdVm2/a7sMj
6Guf+8qi/XQ7OGD8w7q4H8Oqvs/utd63bKtZcp6POupd6dq+13blfKHFNmj7dr1OC51faKr7FHB9UeJd855R/3
a9DM7RoM78qlXAAAAAICj1jouVuFhoSovr6x5jo07OIRMcK1qMWWfJHXylOjT/RtOcUmErEbj1iNZH2SpzTltF
DMkRiEpIYodFqtWY1sp9/PcOl/jiHHIEesIlH2S5Ep2SYbkyT/OvcQeY5czORko+34a4y3yyu+t+WgBv8evjLc
ylHx1stw5bhk+Q+Hdw+VKcsmV6FL5LvOfd1if7Fa7esT1MDtGo6DwAwAAAIAGFB4WqvjWcSorr6hxrhWFH5qQM
bY1ZkdodFZJ9x/crJeyKxTnC2+O+xpVRo1GxmK1HHY5dViXMHkKPfJV/lzKurPckkVyxDkCY9zZ7sCGm5JUlVU
le4xdVnvNCij3s1xF9IlQaMfQQ6/5RSdoeKu/Hwy6xXaTO+b87YFBgMIPAAAAABpYpw7tVF5Rs/Br43TLEswPT
EOzOse/1ewIphleka3PO3folIqYRrlf5ImRyp2bq5K1JXLnulW8ulgH/3tQUQN+fhxE1odZ2v/6zzt8Rw+Jli3
CpgP/PKDKA5Uq23ZoKW7sqbGBTTviTo+Tr8ynzFmZqsqqUsnaEuV+nqu4M+JqZKg8UKmiFUVKmJAgSXIluSSLl
L8oXyVrS1SVWaXQE4LrkQMtZcMOiUO7AAAAAKDBJSfEq7bHp7ushuIcHuV5WsaMEzRdIxxbFWGUmB3DVJGGV69
lrde7Ual6JtamSqu7we6VdEWScj7OUcbbGfIWe2WPsStuZJzaXNAmMMZb6JU77+cMthCbUv8vVRmzMrTrwV2yR
dgUPShaCRMTAmOcrZzq+H8dlTk7Uzvv3Sl7rF2tzmylNuPaVLu/YRjKmJGhxMsSZXUdKgutTqvaXt9WmW9nyvA
YSroySY5YR4N9DszQp3UfsyMOGgo/AAAAAGhgiW1ayeFwqMrtlstZvdxLdrop/GC68xOrg2755rG6rHiPhpeH6
9aEE7TT2TAbethCbUqanKSkyXVvONjuhnY1jrmSXUq9I/Ww1w7rHKZO93c67BiLxaIT/npCjeNRJOYp6sTg3XS
qJRV+LOkFAAAAgAaWGN9aEWGhKi2ruaw32VVlQiKguiHaYHaEJiXFW6ZPDmzQlcXhshm2334BmrxIR6RSow9fl
gYTCj8AAAAAaGDRkRFqHRerOvKaO14m8Bw/mCzJWqBk/wGzYzRJd+ZtOfTMErXxRZgdBcepZ+ueslgsvzOwSFD
4AQAAAEADs1gs6npCB5WV1Sz8nFZDrdmtFya62LlcVkrnOg2oOqgvOrbp9PLow+6ii6ZtQMIAsyMOKgo/AAAAA
GgEJ7RvKOOS31/zQWks64WZTretNTtCkxcqn57P3qCH86Rwf4jZcXAMTm17qtkRGhWFHwAAAAAOgg7tkn98jl/
NWX5JzobbDRQ4HLt86uHfbnaMZmN8SZo+2Z+unu4Ys6PgKMSFxKlXq15mx2hUFH4AAAAAOAiS4lsrvnWcCotLa
5xLcLpZUglTjHVuUIhRaXaMZiXJV6H3D6zXDUWhsht2s+PgCAxNHtqint8nUfgBAAAAQKOwWq3q3bWzSsvKapx
zWA21YZYfTDDOvsrsCM3Wrfnb9FZmgZK8kWZHwW8Y3na42REaHYUfAAAAADSS1PZtJdX+HD+W9cIMA7XJ7AjNW
p+qAs1N36Kzy6JkMVrWDLLmwmqxaljyMLNjNDoKPwAAAABoJB3aJSsyPEwlpTVn+bFxBxpbF1uW4v3ZZsdo9lz
y68mcjXoyl6tIf6jZcfArPeN6KjYk1uwYjY7CDwAAAAAaSULrOMW3bqXCktqf4+ew1Jz5BzSUi5zLzY4QVM4qO
6DPOveqX1WM2VHwC8PbtbzlvBKFHwAAAAAOGqvVqt7dO9e6U6/NIrUPYfMENJ4R1nVmRwg6rf1VeidjvW4pcMl
hOMyOA7XM5/dJFH4AAAAAOKhS2yXLIsnnqzmbL5XCD4OkTFXq7NtldoygdWPhDr174KDaeaPMjtKixbhi1Kd1H
7NjmILCDwAAAAAaUYeUZEVFhNf6HL92IZUs60WjON+1Rg55zI4R1Lp5ijQ3fZPGlObKalC/mGFo8lBZLS3zc98
yP2oAAAAAMEl8qzgltGmtwuKSGufsFimFzTvQCMbaV5sdoUWwy9DDuZvOj+wqxfjDzI7T4pza9lSzI5iGwg8AA
AAAGpHFYtGJvbqqrLzmc/wkKTWOopEToSU6ydhsdoQWZWRFpj5P362TK2PMjtJiOKwOnZZymtkxTEPhBwAAAAC
NrEfnE+ROOlVRWfOZfSmuKtlZ1osG1N++VzH+QrNjtDjRfremZ67Xnfl2hRhOs+MEveFthyvK2XKfoUjhBwAAA
ACNrFOHFCXFt9bB/MIa5+xWg2W9aFAXOpabHaFFu7Jotz44kK1UT7TZUYLa2alnmx3BVBR+AAAAANDIHA67Bvb
tqaLiOlrPs6wXDWmYZb3ZEVq8VE+J5uzfoEkl4WzoOQBC7aEamTLS7Bim4qsKAAAAAEzQs2snuZwOVVTWnM2X4
qqSjWW9aABxllJ18O8zOwZOqJD568Etei27XK184WbHCSojUOYq1B5qdgxTUfgBAAAAgAk6dOhRYnxrHcwvqHH
OYTXUjmW9aAATXCtlE2VyUzKkIkefp+/QqRUxkmF2muBwTuo5ZkcwHYUfAAAAAJjA6XRoQN+eKi6pfVnvCSE1N
/QAjtdo2w9mROAtIgyvXs5ar/vzrAr1u8yOO6zFhcRpeNvh9XKtadOmadCgQYqMjFR8fLzGjx+vbdu2/ebrPvz
wQ3Xv3lOhISHq06ePvvzyy3rJczQo/AAAAADAJD27nCCHw67Kqpqz+TqEVsjFsl7UI4v86uPfanYMHMbFJXv18
YEMdXXHmB2l2Ton9RzZrfZ6udaiRYsOZcoULVu2TPPnz5fH49GYMWNUlZW52uWLFmiyy67TNdddd51ODOEHjR8
/XuPHj9fGjRvrJdORavKzs6u9LiOtTePGjVNYWJavVKERmjhxorKzs6u9Li0tTeeee67CwsKUkJCg//u//5PX6
+N1xx13yOv1NnRcAAAAAGgOXTq2V2KbOnbntrtUhdwsobPxSC1mmOr0Qo3ap9RiqajnbdMHx1Yr6uLwmQzbGbHaXY
u6HxBvV1r3rx5uuaa9SrVy/169dfdPM2bMUFpamlatWiVJKigo0AsvvKB//OMfOtPOO00DBw7Un/70Jy1dulTL
li1riE9xSI2+8Ad7qZH0uzVr1mjr1q36+uuvZYxRWVmZXcvXk9Xqr/1ar1artu3eq9fnz+i67Q7rr7jj+NerpVXrKl7gRq4qMPs6PRqe7bsn4fBrXmIa18TTXJiNsTKHF+Ke6sn0QQlHGHpsHIPLIiUyhusHa8Cp2uoEdKurTq+2HemfuE0l+lpBbfe6Dlaj/hfrBl6Q16pV6b01c0lC9FqYZBG0JX9xuH2fHj06aPbF+59aJAZlcZW9eCXTXHA4LzLyHjJhSZsxnvKQhkm6C/5hDV9CPoMgLFzuGvPKqZ5fFo+hozcRvJkzfh7mqD0sKQyW7gpm/BZhpgs+TpwHHoyBL5yTfSN8OHSBS1aTkCGmMfmMzkjsjr5NIGyaRKrepSXpeEAl4nnvAbf+4Fr2j6g/zaS28A2NcaAmxGHnNeZCfvUeT6hURUahfYWXoqkakq/B7X12SDnbfMf+Sp51bivWzdH4ijn5w7vDPsZtzKIy/o+JeXrFyH0l5aUEJ8yCcYhEZwhRm5dAnK8psmA7QjvC0Uqm3dgVOIbqmR+Wr2HP7bzKp3NLc3ebtbiDGy5WVCDAPTkIPpMwOXK8C6LdCu9nYnX7uNexIyAyGV4N2ICRFkEVdS3uHpgOTYVEZt/8BGSqVy1B7z3SrNmbJB13iyq1NXrH58dpMkfrzZynmkrU8wi46y1zR5Ktwjv3YtdHhJyZhrHw+ix7AAAAANjb5bOjWHuPGHBIOwcwhEk7IxqOyAAAAAFp1yQJ4xMR87hYsNRVr+dLj0Kg/A58kwUQ6t1ZqLiT4t46f/vvHJzuj3YoDVlDSWaeVXmP0ZZHXe2xTuLQPZoHyMxe64/3WLhAHOkYa9kO2OWR+0hvkZKNAw5BcDyjn4kxjg33yoUoAZPFxMKSfPawYpV1bH6l8s91PQhh3zvzxaeNg1eyuP91u3hX3FEaAPMOwbiaxTYKjbpIZmbeo6D9FZBq/3rhwsEM1YsN7LPrx2u2Uv5ihpxLMcbNmhLAcLGt/FMP7Wsxi4t7WuTyhsbnsL4dlzm7TCYVs3wBAAAALgkqK9rp7tbRWvAOGPFa9IMPLtk9SzrTN6t/wpbyy27xd/vR+AabA6iMyKnu2/r7YRjaf8I7d5Vr5hpJdiAyljCOgejR2VuIpC3xVcgZ2tcF1lPQBHnjnH5fxjzaYF8aNk/9T//6SW/+l7a6xaxZnWfp+fp27ydn8jfzT+nOhBtbxCO/7Ad3n/j+nvY0/qc/jJA1qsdJ2v3rgT3+q+3LW+7+u7l63KFvprPmjCjVKrJZdU6fy7V9Vt9arBw3+jDp5FzdtdpEdf47B6bGl7FRQ++/5NS6oaJeRjAn7VdaHpM4jSTpZKiyUHxPbpp4KfMOd9dYX8MZPfNfaaNuM2ekmt+8Gfwj+8/xa9aVSpmqoWrLxK+ilEJJ/Ij3yaq+k9NrHgBsSIn6F9dF79qNB0yTXN0PG40HZQ9gZpcM2s9mf
```

```
Q/fff35BxAQAAAKBRuVxODejTo87deruH1z2LBDha5ztWmR0BR+H/8rfqzcxixfsizI7SbHSL7abucd0b7PpFR
UWSpLi4uDrHLF26VKNHj652bOzYsVq6dGmD5apNgxV+paWlmjx5st544w3FxsYGjhcVFWn690l65plndMYZZ2j
AgAF68803tWTJEi1btkyS9NVXX2nz5s165513dOKJJ+rss8/Www8/rJdeeklut7uhIgMAAABAo+vZtZPsdpsqq
2r+rhNj9ynJyeYdqB+DtcHsCDhKJ1Xl6Yu0rRpVHi2LYTE7TpN3fqfzG+zafr9fU6dO1bBhw9S7d+86x2VlZSk
hIaHasYSEBGVlZTVYtto0WOE3ZcoUjRs3rkaruXr1ank+nz8nmrHu3fvrvbt2wfazqVLl6pPnz7VPkFjx45VcXGxN
m3aVOv9qqqqVFxcXc0NAAAAAJq6LqntldC6Va279UpSD2b5oR60teYpyZ9hdgwcgxD59Vz2Bj160FC4P8TsOE2
WO+rUuZ3ObbDrT5kyRRs3btR7773XYPeowS1+L+L333tas2ZN1Rp61t8ERk4ZAzmiYFpmzrhy9Xfh9kqDoHpenfpu1nhW6r5BWhNYlv0n5ueklqbvSCsMd8T0nM1ta2d84F8yeuMweqMcwvW9WA72I2GR+vqWelV5ErXLlj4oY9Fd3mU9jpdMCMcEGN0r8/M
Js2TdHR0YG3lJSUevhIAAAAAKBhhbhcGtK/jwqLimUYRo3zHUMqFWr1mZAMweRi5wpZVfPrC83HeaVp+mx/mnp
XxZgdpUkad8I4xYXUvdT2eNx88836/PPP9e2336pdu3aHHZuYmFhjn4bs7GwlJiYGjCuaYo5/CTUhJq8j/Rw
s1SSEjjNc/33HOPioqKAm/p6emNdm8AAAAA0B4D+/ZSZES4ikpqPsvPapG6snkHjtPptrVmR0A9iPdV6t2M9fp
DYYgcRv3sRBssruh5Rb1f0zAM3zzzfrkkO/OzTffKDU19TdM3ToUC1YsKDasfnz52voKHasfnz52voOKH1nu9w6r3wW716t
XJyctS/f3/Z7XbZ7XYtWrRIzz//v0x2uxISEuR2u1VYWFjtdb9sO+tqQ386VxuxXy6WoqKhqbwAAAADQHHRol6Q
eXU5Qds7BWs93DyuXhdlZOEZ2+dTdv8PsGKhHUwq26+2MAiV76T4kaXDiYHWN7Vrv15yZYreeecdzZ49W5Rk
crKylJWVpYqKioCY6666irddNNNc9c7/kOp94Lv1GjRmnDhg1
au3Zt4G3gwIGaPHly4J8dDke1tnPbtm1KS0sLtJ1Dhw70bl0JOTExgzf/58RUVFqWfPnvUdGQAAAABMZbFYN
GzgiTJkyO0321DgfafepnYvNO3Bsznauk8uoNDsG6lkvd4E+T9+kcaVRLX5Dj4aY3SdJr7zyioqKijRy5EglJSU
F3t5///KmLS0NGVmZgbeP+WUzR79my9/vrr6tevnn/79739zrpw5h93ooyHU+/zPyMjIGh8i8Kxt7WT6fvtTFp
XXX6U9/+pPi4uIUFRWl1W265RU0HDtWQIUMkSWPFHPnj115ZVX6sknn1RWVpbuvfdeTZkyRS6Xq74jAwAAAAID
p+vboouSEeGUfzFNKcs2VTT3Cy5RexQP7cfTG2VdJfrNToCE4Z0jx3IOaVZ6sB1pHqtha8dsvCjIdojrotHanN
ci1a3uu6q8tXLiwxrGLL75YF198cQMkOnOnINtkvv4Tz77LM699xzNXHiRI0YMUKJiYn6+OOPAP+OOPA+dtNps+//xz2Ww
2DRO6VFdccYWuuuoqPfTQ2bEBQAAAIAGFxoSouGDTl2JRcUmtv2SmuKoUYf0akAzN3UBtMjsCGtiZZRmam75X/
Vvghh6Xd79cFkvnLnuFYG4txJHVl1M1RcKzo6GgVFRU1+vP8GK4WrHgzVFRU1+duDgKMwoVuS2REAAMARMPNnUASH/ZnZeuQ
freyssLFRxMdE1zq8vDdeK4prHgbp0tWXqK8efzY6BRjQ9prNejjHkttR8PECwiXRG6uuLvlaYI8zsKE2OKTP8A
AAAAAA1tU2MV58eXZWdm1fr+R5h5XJZfY2cCs3Zxc5lZkdAI7uucKfePBTP5Cr9p7g/8PThM4TKPvqQ0EHAAAAAE2
ExWLR0AF9ZbVYVFnlrnHeYTXU07zMhGRorkZY1pkdAS607CnWp/s3aUJppVn/NotN1/e430YTVZw/lsHA
AAAgGaqd9f0at8uSVk5d93yu8TC4L0zDgt4WpUif4d5sdAyaxy9CDuZv0QnaIE1clmx2i
yKPwAAAAAoAlxOhOaPugklZaVye+vWew1+vWew5rYZ6RZSakAzNzXjXGjnERi8t3YiKLM3dv1NDKmPMjlKvru11rdkRm
jQKPwAAAABoYgb07alWcTHKzSuo9Xyv8DI5mOWH3zDWttrsCGgiov0evZGdG5Xnfn2RViOM2Oc9xGthupPm36mB2
jSaPwAwAAAIAmJr5VnEac3F+5+5efkyDKPGeZfVUC+e5YfkKKxxxewIaGImF+/WR/uzdIKn+e72bZFFt/S/xewYT
R6FHwAAAAA0QS0GDFRcbIxy8/JrPd+vsEPO/nOlnvaP9m/Q5cURs/jXL3XW07mh2
jyWt+/2YBAAAAoAVIim+t4YNOVPbB2mf5hVgN9WCWH4kHfWiv/QpzC
sdoFij8AAAAAKCJGjF4gOKio5R8SbX/uz/PqEl8nOLD/U4hTLerMjoBk4uTJXX6Rt02kV0VLNvys0ORd0vkAdojq
YHaNZoPADAAAAgCaqXVKChg08UZk5ubXO8ouwGTohkv3yK4U6Z/4kzqlycpkBxapYEQhQSl28Oj1v4
a1O/aHfH8yO0WxQ+AEAAABAEzbylIGKiY5UXn5hrefpHhZ/ZnhchGiDmDBpLMpIEdyPPN25tZrvNSGEkdxW/2/VpGW
wcr
```

```
Olh+uH34eoQbdXot8tV5q45e/UnH18apsw/RwTeNt4ULptFurjnz4Xdk4vden65W6+OC9Hy68MV7rRo7DvlqvQ
euu7rqz1aneHTOuvCdeMAhy7/qCIwY3ZPgV9vrPHoOVE1M6Nu1xTtOvsHctTRc2zLca/seaW6xHap51QtE4UfA
AAAADRDFotFZ4OcpsT4NjqQlVPrGKfVOKkxhZLqLk7Q/PWOHVAr/OGzY9Tp7C4OPXJGiC7s4ahxbke+X8v2+/T
KuBANamtTt9Y2vXJuiCo8Orsb654lFhdqUWKENfA2f7dXYQ7p4p6H7mEYhp5b7ta9I1y6oLtDfRNsemt8qDJKD
M3ZeqgE33LQp/O72dUr3qYpg5zKLTdOsPzQfys3fVGhJOa7FOViptnR6uQpOaf7N+iikghZjSOvnRLCEnRTv5s
aMFnLQuEHAAAAAM1Um1axOueM4SouKVWV213rmLYut3qElTdyMjSmic7lZkc4ZlU/TkANsf9crFktFrnsOvdpN
Zfs1mX6Dx5N6u1QuPPQdfYUGsoqNaotOYOOsWhwO5uWph+6br8Em75P86nCY+i/u7xKirCodZhFs9Z7FGK31Fp
Q4shYJf3t4Ga9lF2hON+RPVbgL4P/ojBH/W7+OZJR+AEAAABAM3bqySepe+dU7d2fUeeYk6OKFcnS3qB1qnWd2
RGOWffWVrWPtuieBZUqqDDk9hl64vsq7S82lFl6ZA8lXHHAp405fl3f3xk4lvXjaxPCq8/QSwi3KKvsOLnfneR
QvwSrer5cqkf/V6UPLg5VQaVO/8JKvXB2iO79plKdny/R2HfKdKC4iT4gsYkbXpGtz9N36JSKmMOOu6DTBTqj/
RmNE6qFoPADAAAAgGYsNCREF4w5XQ6bXXkFte/M67AaGsHS3qAUYanUCb49Zsc4Zg6bRR9fEqbteX7FPVmisEd
L9O1er87ubJf1CFfTTl/jVp94q05uW/cmH3Xd+6VxodpzW6RW3hCh4e3t+vNXlbr1ZKd+yPJpzlav1vOhQkPa2
nTrvMpj+OggSZGGV69lrddf8mwK8TtrnE8KT9LdJ99tQrLgRuEHAAAAAM1cv55dNWr4ycrIzpHHU/tMviSXW73
Dyxo5GRrahc5Vsqt5z94ckGzT2j9EqPCuSGX+OULzrghXXoVfJ8T8dmVR5jb03iaPrjup+vLbxIhDr8Ouq15yZ
5cZSgyv/brf7vFqU45PN5/s1MK9Pp3Txa5wpOWX9HJo4d4jX16M2l1WvEcfH8hUZ/fPG3pYZNFjwx9ThDPCxGT
BicIPAAAAAJo5i8Wi888cqW6dOmp32v46xw2MKlYOS3uDyhjbGrMj1JvoEIvahFu1I8+nVRl+XdD9t5+h9+Fmj
6q8OhV9q49NjbEoMcKiBbt//novrjKOfL9PQ1NqzgSs9Bqa8mWlXjs3VDarRT6/5Pmx4/P4JZ+f2bH1IcVbpk8
ObNDEbMkuu67qeZUGJg4OO1ZQovADAAAAgCAQGRGui8edKZfTqdz8glrH2C3SabEFsrCON2jOM7aYHeE3lboNr
c3yaW3WoQZtT4Ffa7N8Sis69Fy8Dzd5tHCvV7sL/PpOqOdnvl2u8d3tGtPp5wO3rvqkQvd8XXNZ7fQfPBrf3a5
WYdXrDYvFoqmDnXrkf1X6bJtHG7J9uuqTCiVHWjS+u73GdR5eVKVzuth1UtKhMnBYe5s+3urR+myfXlzh1rD2N
V+DY/dAeZretaXq1v63mhOlaPEVC+CYbNy4OewICCK9e/c2OwIAAEGhd7fOGjNiiD76coGiIyPkdNScIRXv9Kh
PRKnWlOaakBD1aYh9p6KM2p/b2JSsyvDp9Jk/7xT9p6+qJFXp6n4OzRgfqsxSv/7OlVvZpYaSIi26qq9D953mq
naNtCK/rJbqpd62gz59n+bTV1fUvrPrncOcKvMYunFupQorDQ1vb9O8K8Kq7QgsSRtzfPpgs1drf//zbrIX9bR
r4V67Tn2zTN1aWTV7IrvH1iefzaXuYx+XbDWf6Yf6YTEMIyj/tFNcXKzo6GgVFRUpKiqqUe/98bbMRrOfgt+Eb
klmR6iBwg/1icIPQLAw82dQ4Cdl5RX6+2sztWNPmrp16iiLpebOBz5DmnuwtQ56+GW7OXs8bLYm+T83OwZw1Pz
nPS/rgKvNjhHUWNILAAAAAEEkPCxUl5w7RmEhIcrNq31pr8OijY4tkMvKRgTN2VBtMDsCcNSqup5L2dcIKPwAA
AAAIMjO6HKCxo48RTkH81dtc6JsLuO+kxhTzPr5lqYYlSij/d7BjAUakKS5Rrwstmx2gRKPwAAAAAIAidffp
w9e7eWbv37VddT3JqF1KlAZEljZwM9eEi1OpZ5Tc7BnDEfLLKednbUkiO2VFaBAo/AAAAAAhCYaEhuuTcMYoMD
1N2bl6d4/pFlKpDSEUjJkN9GGX9wewIwFHxj35IlpSTzY7RYlD4AQAAAECQ6npCB4ObNUL5hUUqLS+vdYzFIpO
WU6gom7eRO+FYWeRXT2Ob2TGAI1bW/SI5ht9idowWhcIPAAAAAILYWSOHadigk7Q37YC83tpLPafVOOi4fNktL
BFtDkY7NivMqL3ABZqakri+Cr/4NbNjtDgUfgAAAAAQxBwOuy674Cx1PaGDdu5Jr/N5fnEOrO6NLmrkdDgW5zl
WmhOBOCKljtYK+9OcyWY3OOqLQ+EHAAAAAEEuLiZaVO48T9FREUrPyKpzXKewCvUKL23EZDgWJ2uj2RGA31Qll
yyXvy9bRCuzo7RIFH4AAAAAOAJOSW2vS84bo8oqtwqKiuscNziqWAnOqkZMhqPROZqrBH+m2TGAw/IbFhWd8YT
CUweaHaXFovADAAAAgBbi1JP7a8yIodqfma3KqtpLPatFGh1bwCYeTdRFruWymBOC+A2ZPa9X/IhrzY7RolH4A
QAAAEALYbVaNfGcURrYt6d27kmvc9pkQQxSLlY6rfvn9ja6eOUJqf1T2oYQAZ3kuSB2sW4Q+nxIDIX15dOQc1
ACxIaEqIrJpyr9mOTtWvf/jrHRdt9OjMuXzZ27mOyXHKrq2+n2TGAOhOM7aT469+XxcI8VLOxTQqAY7LdwYNXU
X96mxOAAIAWJim+tSZfOE4vzXhPWTkHlRjfutZxCU6PTo8p1IKCWBksJDXdOOc6OeU2OwZQqxJrjMJ+94nsrlC
zoODM8AMAAACAFqlfz64af9bpKigqVnFpWZ3jOoZWakhOUSMmQ13Otq8yOwJQqOq55L3kHYW16WB2FPyIwg8AA
AAAWqgxI4bqjFNOVtr+TFVUVtY5rld4uU6KKGnEZKjNAGO2OwJQQ5XhUMHYlxTb/VSzo+AXKPwAAAAAoIWy2+2
6bPzZGtK/j3btTZfb46lz7ICoEvUIq3smIBpWH1u64vx5ZscAqqky7Eof+qiShl5sdhT8CoUfAAAAALRgYaEhu
vri89WrW2ft2L1PPl/dm3ScEl2kEOLLGzEdfjLBudzsCEA1bsOmLd1vV6exN5odBbWg8AMAAACAFi42OkrXTbp
QHVOStX33Xhmes4iOUaGVOodq66l/+iYQy3rjc7AhDgMaxa3fYa9b7oLnbkbaIo/AAAAAAASk5oo99deqFax
8Vol77OOks/qOUaHVugZGdVIydsuSIt5Ur17TU7BiBJ8hpWLY6eoH6XPyC7w2F2HNSBwg8AAAAAIEnqktpe114
yXqEhIdq3P7POcXaroTGt8pjp1OgmuFbJLq/ZMQB5DYsWhY1TvysfVVhElNlxcBgUfgAAAACAgBN7ddPVF5Oni
OU6kJVT5zi7RTozLl8dQioaMV3LNMa2xuwIgHyGRd+FjtWJ1zyp2DaJZsfBb6DwAwAAAABUM6R/XX112wdmqqKh
Udm7dO8PaLNKo2AKdQOnXoPr6t5odAS2c37DoO9co9b3mabVKSDY7Do4AhR8AAAAAoIbTTxmki8aNVmFRsQ7mF
9Y5zmqRRsYWqAu79zaIYfbtijSKzY6BFsxvSN85T1evq59R68R2ZsfBEbKbHQAAAAAAOPRYLBadc8apqqisOqf
//VaGDLWJi611rNUijYgplM1iaGt5eCMnDW4XOFdIfrNToKXyGxb9z36qul/1jOLbdjA7Do4ChR8AAAAAoFZWq
1UTzh4h4lq9V6qPTzG4pvHVfrWItFGh5TJJvFOKayiEZOGryGaKPZEdBCuQ2bvrGdob5XPa7ElFSz4+AoUfgBAAA
```

AAOpks9l04VlnyGG3699ffi2/4Vdim9Z1jh8aXSybxdD60shGTBmcEixFaudPNzsGWqBSv1Nf2cdq8JX3K7lDZ
7Pj4BjwDD8AAAAAwGFZrVadd+ZpmnT+WJWWlisjO/ew40+OKlH/SJ47d7wuci2XVYbZMdDC5PnC9al1nAZN/qv
apnY1Ow6OEYUfAAAAAOA3WSwWnX36cF1+4TmqrKzS/szsw47vH1mqETEFFFbH4QzbWrMjoIXZ743THNs4Db/8/
5TSqYfZcXAcKPwAAAAAAEfEYrHozFOH6MqJ58rr9SrtQOZhx3cNq9DZrfLksvoaKWHwsMqnnv5tZsdAC7LVnag
vbGN1xqSb1aFrb7Pj4DhR+AEAAAAAjpjFYtHIoQN19UXnS4a0Nz1DhlH3LL4kl1sXtD6oGLunEVM2f2c6NinUq
DA7BlqIFRUd9H34OJ19xS1K7d7P7DioBxR+AAAAAICjYrFYdOrg/rr20gtkt9u0J+3AYUu/KLtP57c+qLauykZ
M2byd61hpdgS0AH5ZNL+Om7YmXKALrp2q9l16mROJ9YTCDwAAAABwTIYO6KffXTpeIS6nduxJk9/vr3Os02pob
Fy+eoaVNWLC5utkbTI7AoKcVzZ9UtRHhZOnaPw1t6lNcnuzI6EeUfgBAAAAAI7ZySf21h+uvFht4mK1ZeceeTz
eOsdaLdIpMUU6JbpQFjbzqFMna7YS/FlmxOAQK1eIZhUOkGvAZTr3ypsVFdfa7EioZxR+AAAAAIDjOrdHV9123
eXq3qmjtu3ao/KKwy/d7RlerrFx+XJa6p4R2JJNdKOwOwKC2AEjXv8qPEUpI6/S2EuuV2h4hNmROAAo/AAAAAA
Ax6192yTd+rvLNXRAP+3et18FRcWHHd8upErntT6oKFvdMwJbqtOsa82OgCBkSFru6a73y4ZqOHm/O8jzJ8vhd
JodCw2k3gu/adOmadCgQYqMjFR8fLzGjx+vbduqbyVeWVmpKVOmqFWrVoqIiNDEiROVnZ1dbUxaWprGjRunsLA
wxcfH64477pDXy/8RAAAAAEBTFRsdpd9fcZHGjTpVObl5yszOPfx4h1fj2+Sqc2h5IyVs+kLkVlffTrNjIMhUW
kL177LBWqF+GnPpDTr5jHNltTIHLJjV+7/dRYsWacqUKVq2bJnmz58vj8ej0aNHq7y8vLbNpkz5wez3n777Zo7d64+/PB
DLVqOSBkZGZowYULgvM/n07hx4+R2u7VkyRL+8OfrkzBTLTNKz+fLJPLMmx27ZHf/X+fVIScq42Xqx8yA9Vp7yd1Sk
Yfdw0Ib0Om48W8Ev8h4CMRZRUjTijF1/o+XVIzYVuN94hxjQlbyLkbZxkltUTMXGJ84tMyqrtI5uzkdSkD2i1Xwv
Hx2vRokUaMWKEioqK1KZNG82ePVvhCHRTgxVNhLDEmVvpiE0nXwsHNavbB7YKjQ8hIsfXkIB6Mxk/F3tIlGqRQl9
+5Sbm6unEcw5bS4uFjR0dEqKipSVFRUQ36INXy8LbNR74fgN6FbktkRauDrHPWpKX6NA8CxMPNnUOD/27vv6Kq
q9P/j79vTE9ITUiCUQOidgIMIKCrjiDpjYxAR8YcKO04Ncy4sygYsGC4lhHB8cyIzpWRBAQpAYiPUAIPQkkk
N7vPb8/gGgEAoQkN7l8XmvdlWSfffZ5zmavKS5KHffF5zmavs5KHffZuitZu2MK/PvmSQzm5tE+Ix2q11lo/v8rCoqMtyK28cF8
zfNP3FYY6l7k7DPEABrDB2oOvskKJatW0y28YT2RsgrvDkkbS4PM38/PzAQgODbgYJShVl2nQ4cOx
MXFsWLFCgBWhBly5dqpN9eDAD6egoIDVDWrrUzUYm6N6ZDdYWUVtHMTU10eGRlJR
2mGh77bMCPcm19lXuDkGasROv8L6V1RF7WAJX33o3Ay67RjvxXqBqX0DhPEyaNIkvvvvviCpUuXEhMTU10eGRlJR
UUFeXl5NWb5ZWdnExcXV1Kzeu3bw1FbqaDR8XYJAy1dgrqdx8k9PJYtzmsNotrwlPQFtGj5VjsTG4xo3nfLk
VfTW6RRHJS5Z9wacmSO1v20hLtWsNXydaeYvmJcxwOvs1sZsvBR3rKgqmGR
cgSQKE0GbGxtpkSFGTJYrK6LxJTWRoJL4FgR5EYHqjp8hT+Rz9A0l8hE4ePRlfP5AxAX1J+4Mq2tb9
mAUrRIIaPYGqSOk8B0FCqnSv8PeWa/pNYmUjR9CKdFJdHVVm1YLF4UrSRU0G1szfPnn3fsBY1lFRbRvZDuJ9YrjNsWQGfvm2XZ9LrN7wP
5RDfXr7m1cpvrcG4xKc4B2Xs43SQJ6X1W6XUP9k+FYgxQr3VEXKJrP4aeK3ZEtZnfjBPg5u4TD9i3f9B7jgUBFHs
YXqbq6+oJUXY0SsCVCBcnr4Vmj8RD7+qu7+e4xXGKXVP4EhnmsS3/nSi6dB8ZOgmlUaPBZvdMGHFA6Qvzl/1PM6uD1X
ZmkSn0GdDAhLqc8GH8pHdmU92kPvUF3K2qUvFVMGZlWSaMvHj8KdV8SK3VoaMEnRSS1Cwi/ejaW2fAtP8pPpv6JMKtTZzZQm+Yu1
hSkGrHKWrMv7o+fXbrwMdSBxDIz9RmJLpyqqtiTlMTb3vAw4kbMmkcnaCb/j7PMw3D2XyTtvc0J1qW7E+XKdqmWwd5cPXe3dPNJ8eS
DFkmeEHOGkbSjoPPRTfRy6amP/2lqB9bxc2FRdsKQvfYyKUWpNvR+AmQ2uk3b+7dW7m+/S0WGGe+ZvbOO4ZN6AvSE5BRNF04Pdbe1
xhAuR7TRRT2s6Ll1V8nG9EfsJDwqoTAjKHcQJQHmsM09mDdi+1b+R6+tI8/IS+4e3rhQdpDXEH9SRkN7c1EtP6b2J6iQgWCvg2HSWCvj
2Fb3CmX5xSrb1U3UHRVfcFX6MsvZ6fSoQ7Ajrb/t+82+pBUa/xpMmHjEJ2+q0M/4U8mJ7h4u6e5SvLCtUuUvjqQfzYqtqkWqi+
xUbpJGYlvD5mIu4+q1C+bkEdPJmVNjMy1E9f9VTSkt7NZ+ZzNbk5/6DV5X2sj0fzvoxSXemJmLm9Frfn6VLkHd4GPfg3ePqnUo
Hdi8hs3CqIR2BmcS0S+x6aUjpK/ESh6kUdS5yUdy7Zl9PERc/eFGKUdrSq/C3Hb1mWubeu0CqsJWMGR+5+k06OO55Dmv7dZgP

CSBwWHuDlGaMSX8REREREREpFkwmUx0aNua++64he9XrOGrRT+wOS2diPCQM77mCxBur2REaC77yxysKQggt8r
WSJGf2mX2jXgZZW6NQRqfEzPpji5s8kmm2HCQc3A/ZSWFRMe3Y8Dwa2nbudcZx7LImSjhJyIiIiIiIs2Kw2Hn8
sED6ZaUyMIfVvHDmnVs3p5OTGQEQYH+Zzw/xquclo7DZJR5saHIj5zK2l8Nbii/ta4Fl1suLW7gwsReeyIbfAZ
SYArg6KFMCvNyCW8Zz8W/vZFOvS/C4e3j7jDFQyjhJyJynNPp5KOXn2Xp//5LXs5hWoRHcMk11/P7OydX/w9ba
XEx/3r2b6xeOJ+ivKOEx8Ry5ehxDL/xltO2u+CjuSz57GP27kgDIKFTF0bdO412XXtU1/nszVf59M1XALjm9on
87rYJ1ce2/7SO1x+fxpMffan1OOREREREER+ISo8lD9eN4KL+vZg/pLlrE7dxMHsQ8S1jMLPt/bEickECd5lJHiXk
VluZ2ORH3vLHdCIm3vOZnOjXUvcp8JkZ5ejC2lePSgyB5Cfe4i8nEOEhUUy5JrRdOk3GF//QHeHKR5GfzmKiBz
36euzmf/vf3LXky8Q2zaR9EO/8fJD9+Lj58+IW24H4JOn/8KmVcu55+mXCG8ZS+ryJbw+fRrB4RHOGTL8lO1uX
v0jF40YSWKP3tgcDj59fTbTx93ErC++JyQiit1pW/jgpZk8NOddDMNgxoQxdBt4MfGJHXFWVfGPvzzIhOkzlew
TEREROY1WsdHcMer3DOrXi68XLyd1cxqGYRDXMgpvrzOv1xflqCDKcYS8KgubivzYUeKDs4ETf+OsWYS7shvOG
uJeheYgOrx6sMurM5XYKMw7wpHsTfgHtWDA5dfRY+AwrdMnDUZ/PYqIHJe2fi19hg6n1+BhAITHxPLDl5+yc2P
qz3VS1zJ45B/o3G8AAJfd8EcWfPgeOzaknjbhN/mZ2TV+vvOvz7Ly26/YuGIZgOf+gQO7dhKfmESX/hcBEJ/Yk
QMZO4lP7Mhnb75Kx979adule/3fsIiIiIgHMZlMdGyXQGKbVqRuSePrRcvZunMXVquVuJaR2G1nXq8vyOrkoqB
8evkXsrXEhy3FvpS5LAOS7+/tq8BokKbFzbKscaR59+CArQ2YTBQX5pOTuR1vX396XXw5vQYNJzQyxt1hiodTw
k9E5LjEHr1Z8NG/OJiRTnTrNuzetplt61Zz69S//Fyne2/WLPqWIdfdSHB4JJtW/cjB3bu4ddrjZ32ditJSnFV
V+AUGARDfviOZu3dx+OB+DAMO7t5FXLsOZO3dzaJPPmTmf7+p5zsVERER8Vxms5menTvSJbEda37axNffL2dHx
l58fbxpGRm07SzemvC2uOjpXORXvyJ2lviwsdiX/Hre4OM35p/AWa9NihtVYWWPowPbvHqSbz02a6+spJjDB/d
itdno3Oc39Lr4CqLi2mhDDmkUSviJiBx3zR2TKCku5O4rB2G2WHA5ndw8eSqDrrq2us7tj/yVOY/8mTsu7oXFa
sVkMnPnEzPp1Kf/WV/nvWf/RovwCLoO+A0AMW3acfO9U5l+24OAjJoyjZg27fjL2OsZ/cD/kbpsMR/Ofhar1cr
Yh544p2uJiIiIXKhsNisDenenR+cO/Lj2J+Yv+ZHtu/bgsNtpGRmGt5fXGduwmqCDbwmJPiXsK3eQVuLDvjIvX
Of5uq8P5bRzpp9XG9IOlJp82eHVjR1e3Sg3+2AYBiWF+RzJPojZbCYhqTt9Bl9JfPvOSvRJo1LCTOTkuB+//h8
/fP4Jk5+ZTWzbRDK2bebtvz9WvXkHwFfvvcX2n1KY+so7hLWMYcualbw+/SFahEfQtb/NmJ6jk3+8xPKvPuPxd
/+D3fHzL5nDb7ylxsYf38/7CG9fPxK79+auK37DX97+gOjEP8vjdNWRvnC/sRdUx9qZ2bNfF0k1ly3szX1w1LXD9dZ5DR9aHk8/3qrL6lscU9wnHZuPsmAUwbdzfebtTMM9vcP3Trof1tTwvf5P1oQGehmyXGcaKn1I9QbMzKuP7lVrr79fkxNIu/1kPw04WDnYYeFpzJhe0CKVlRs7VPcnFvnqj9Fbp8MvW0DyRmd+UFEQrEfm7pBnEHtYDB1dFYgmR2CzyHQXmx9w6vVl2Ei3zk1tHIn+ZjzZFk0P2WkzGFUo1GNdcJBrvWOnXRLeFAiCgrrbn8tMbsQ4PDVRDxkmI+tiRM7eUKTKAJMyNr4YnJBeEg5NyTtgHL7k2ed8L3Ih8c3CsH9WqB0sAxTsXX+nNt2Ek9o9AjWh4yZktabB5g9EI+AQmSL+hDiy/YsmZ2W5U1IsY9crnLnvEbXU3L3uM2pWPaBGXWkDRHoNHdGvbz0P+yxRSZ3JFWQ8LBqbuLfTtBaUhb6h4WAuz+1kW+i+81rmN9jqcNQl0e6/WFpg1o3l4R9WDlj5Vcs5i0YZVWAOGRUnVVvdVbVmgyZX7b17RLomiE/1LhP8OWL7zl7kRRf1ZWWvkfP7RsI5yRGuBnP2FI+ODN2OFcKRgaFoOFIbg1CRt/9DYwt9+TMxXo48GDnxG0hHM4B8BUtg15o4UGAEFPjKU6MaP0lqCfzjzRtw+xgGaDbHlTeIPUeuOWcXMObAY4wlWY/EbdWN2hP+7+UCqcIDnNLzEHRBqHKKBqn0xpdI0EBlCBZeFUeVjvrQ0d/ueC7gVfpNqXT7nWpgAVGZ/iw1Ikq4wndSc/DD5ufmsD4iBt4GjvN3HRFg9jBYSe/D67pL/9QOQJ2Nxt+aWbnvn9VhStMgkQu9Z6yM7lpdBq4V3Qm1R0MDRsQ1g6QbBfKV7CKbhr+XcXhvuK5yGEs/BaDfb6QyaWh/XWXhDD5vmZ3SwjwmY3M=

```
UzWnsz8ymvKICu8mFqayA8MS6reFaZvblgL0NB+xtqsv8nHmEVGUSUpVNcFUmwVWHsFJVX7fVLDixUGAJJs8Sx
lFrGHmWUI5awyk3175LblVVJcUFeRQX5FFZXobDx/cXSb4kwlu2wnyGxKxIc6KEn4iIyClst5386o3I+dCWHSI
insnXx5veXZPo3TWJ3195KRn7DrB91x5WrlrJ4axAXOVF7Nu5Fb/AIPwCg7HZHXW+VpEliCJLEHscHQEwGS4Cn
Ln4u/Lwc+bj68rHz1lw/Gt+sO4GGkCpyY98awhHLWHVs/cKLMEYpjPvjOtyOikuyqc4P4/yOhLMFgu+AYGOTux
KXPtOxLbpoCSfeDQl/EREREREETqgcNhpOPb1nRo25rfDhvEoYM3cmj/bvbv2sbeHZvJ3r8bZ1UFdocPvgFB+
Pj5Y7XZ63w9w2Qm3xpGPmGnPO7lKv5FIvDnhKCfKw9vVxEWXHW+9vmoMDkoMftRaval1OxHqcmPUrNfzTKz71k
l9k44seFGcUEepcWFmDDh4x9AREwrWnfoSkRsayJjE/D1D2zAOxNpOpTwExEREREalnZrOZyJhWRMaOomv/w
ZSXlZK9bxcH96STseOncrP2k7XvMM6qKsxmM16+/nj7+uPt64fFUj9/qpeZfSkz+5JD9CmPm4OqbEYFVqMC2/G
P1ajEZpT/4vsKbEZ59fcmXLiw4jSZcWHBZbLgPP7V9YuvTpMFF2ZcJitOLJSbvY8n9PxwmmznfW8up5PyOhKKi
/IpKSzAMFx4+/gRFBpB9wFDiYpvQ2RsAgEtQrUen1yQlPATEREREERaWAOL2/i2nUirlOn+g29iuLCfI5kHyA
3+yCHDu7lQMZ2CvNyOZp9EJfhxGK14eObgI+vPw4f3wZ59dRlslJuslJO7evfuZNhGFRVVlBaXERZaTHlpcW4n
E5MJhMOb1/8AoJI7NaPlq3bExnbmuDwaL2mK4ISfiIiIiIiKNymQy4RcQhF9AEHHtOgHgcrkoOJpDbvYBjmQ
fJHPvLrL2pVOQlOvZwT1ggM3hwO7ljd3hhd3hjc3hhcVy9q+9NnUnZu2VlRZRWlJMZfmxXYltNjtePr4EBYcRF
defkMiWBIWEExQaQVBIBFbb+c8YrKulS5cyc+ZMUlJSyMzMZN68eYwcObLWcxYvXsyUKVPYvHkzsbGxPPzww9x
6662NEq9cOJTwExEREREXEzs9l8LIkVEk6bpB4AOKuqOJqTRe7xmYBZe9M5cjiLspIiSooKqCgrw8AFxrHzb
Y4TycBjH6vdOWRmu52YqVdZUX7864nvy3FWOTFhwPFZe96+frRq35mImFaOCIuqTu75+gc2uddzi4uL6datG7f
ddhvXXnvtGetnZGQwYsQIJkyYwNy5c1m4cCG33347UVFRDB8+vBEEilguFEn4iIiIiIiTZDFaiUOMobQyJjqM
sMwKCOuorjgKEUFeRQX5FFUcJT8IzkcOZRJwZHDlJUWU3gOl4qKMkwc2/EWjiUFzRYLZrMVs8WCxWLBbLEeL7c
e//lEmQWTyYRhuDAMA8PlOva9y8D167Jm6+qqjyWzKuqxGSAcTwCm82O1e44NmPP24fQyBgCW4QQOCIUb78Ag
kLCCAqNIKBFGDZ73TczaUxXXHEFV1xxxVnXnzNnDq1bt+bZZ58FoGPHjixbtoznn39eCT+pVOr4iYiIiIiIiDQ
TJpMJHz9/fPz8CYuOO+m4O+mkpDCfooKjx5OBx3atrawop7y05NinrITy0lLKyOqoKC/DWVVJVU5FS4nTqcTl
9OJy1mFgYHJZMZsMmymzCZzJhMJkzmml/NZvOxemYrZruFIL8IAlqE4N8iFF+/ALx9j8Xr7fvzx+Ht0+Rm6zW
GFStWMGzYsBplw4cPZ/Lkye4+4JSDyWEn4iIiIiIiHsJiseAffFIx/UPBZ1T/2qu3xGXkV5VSeeO22ohwDA7PZg
sV6bMbfsdmBZ/q5abxC3FRlZWUREREFRoywiIoKCggJKSOvx9vZ2U2TiaZTwExEREREblAmUwmbHb78Vdo/d0
djojUE6XeRUREREREQaQWRkJNnZ2TXKsrOzCQgI0Ow+qVdK+ImIiIiIiIiINILk5GQWLlxYo2zBggUkJye7K
SLxVEr4iYiIiIiIjUQVFREampqaSmpgKQkZFBamoqe/fuBWDatGncncc/8y2bdt45ZV/fUnTTJjArl27+POf/8y2bdt45ZV
X+Oijj7j33nvdEb54MCX8RERERERERETqYO3atfToOYYMePXoAMGXKFHr06MH06dMBOMrMzpxXyD6B169Z8+eWXLL
FiwgG7duvHss8NKiIv794t0s+u5rCnJzad2xE7f93x0O7dIdgJXffsW3H7x
L+uaNFOUf5Zl539K6Y+cztvvvjN5/z7xee5vCB/UTFt+aP9/8fvS4eesq6rz32IN9++B5jpz3Ob8eMB6CyopxXH
r6fNQvnExQzazvjH/k63AYOqz/n0zVfIOXiA2x/x/52/l3goiIiEgDGTx4MGDMjMIZhnPb4O++8c8pz1q1fpn+4i
IiEd75ZH7+OnHpdz91Es897+EwdBt4MY+PvYf6c29y9h13OO++is29y2bg3P3/cnhv7+Jp6Z9y19h13OO
5NuY+/2bSfVXbXga7b/lEJweGSN8gUf/otdmzfw9w8+59LdwD4cqiIgg5+b5ecWWwVW+v98DrBUfbmEzNmCVzJmbNTzTiZF1Fj/4pfHqQuV3v2ebcHg//X7+8dwr9ccQA
p4SciIuKhystKWfntV9xy/8N03o6PjW3HDX/UTGtWL+v98FYPDVv+f6iVPomjzoDK397Mv33qDHRZcwctyfi
GnTjpvu+TOtk7rw9dy3a9TLzc7kjb8+zD0ozZ20x1nypYP+unfQechlx7RK5fNStFBzJpeDoEQD8Zep7aepD/oEQD+8ZepjL7///D
x8z/PHhAREREBuTAp4SciIuKhXFVOXE4nNoejRrmXtzfb1q46+bHrSdBS6nFd8Kt0nrcx8D/DvhNjbLuAy8mLTXl52u7XLz457u5e
tydxLVLPKmNVolJbEtZTXlZKanLFtMiLIKAFsEs/fwTbA4H/S69os7xiYiIiIhc6LSGn4iIiIIfy9vMjsXsv/vP
KLGIS2hEYGsayLz9le2oKkXGt6txuXs5hAkNCa5QFhYaRl3O+xNxWJhxOhxp2xjyHU3smf7FiaPGIx/i
2DumzWHOvw8PnhxJtPf/Q/vz3qK5V99RkRsPBP//hwhEVF1jlJ9OBTGX9wTs8V
CQlIXLhoxkvTNGxrsmumbNvDle28w87/zMZlMp6xjtdkY/+iMGmUvT5vMlE6sXfs0zn37Hp2/M5s2/P
sKfX3qjweIVEREREfE0eqVXRETEg0XGXfNGteKJf33C3HU7+cf/5L/fx6+cf3a3nq46+oqqokIja+zmOGhYaRn5tToywbBoWq/p0uYTk7LX3kr1LQVM
AbE1ZRX5uDv9v9vSB/+0CmWP3SK5fDB/fzzqceZMKTvKdvuHI5+3Zu54gVf8js2rV9Q5V
hEREGRC5Fm+ImIiFwAvHx8YHjx88sPBvkvNvXWqlXPn5ZF0yuAlbLqRSrVnSrvw6pkvTUxm+Km49czT4oa
4KWcAdtN9xuvQPN7z9NSN/0VV4cmIMYcuKbUhv+9c6rlf2sBK0ZvcEvt2t+UcY/QGsqk1sCMKM3W/R/zDcAqB
vI8AbD9g33Rb5Ss4MN6MNbsKcvERW/HKXl7oHjKs2f3w0Yw5NYg/F8uqm8Z6fsIl/hxm0tO1xOqMdQVxfbfcAe+
3x003M+jq6xhyzQ0ntVdRXsYbTzzEPTNfxmKx4HI5oerYr30NivSDrf9hMWAQ3boNW
XsyeHfmE7RMaMuQa48l3grzjpKTeYAjh7IBOJiRDkBQaDgtwo7N2HvxwbsJDo/kj/c9BMCIObfz6C3X8b+35tB
z8FCWf/kZ6Zs3MLOj7GGgTGOTGh7UowfvzKLnoOGkJDUBYAOPfvw7w7swnu0TaG/h67tt06
NmnvrtFRERERMSjKeEnIiLiwUqCpj73AxysWfQtsx+6AHabACswfQtsx+6t7r+c1PuBOD6iVO44a7
7Acg5eACT6edVQDr07MPkZ2bz71lPMff5J4JlqlZo/v/wWce07nHN8e7dv48DvPufZeQuy5KH/5bNq1fwyKhri
G7dhsnPzK7TvYuIiIiIXKhMhmEY7g6iRQUFBAYGwh+fj4ZBAQGNeu1PjXbiPqjSiIcL28ZhVjLAmAPdQlHXDuXE7Aj8d7Xfr9cT
c51L/3Pk7qIiIiMgJ2rRDRERERERERERETEgyjhJyIiIiIiIiIi4kFU8sNTIp1iEXBMRERERERERERERERERfEgSviJiIiIiIiIh4E
```

CX8REREREREREREPIgSfiIiIiIiIiIiIh5ECT8REREREREREREPooSfiIiIiIiIiIiIB1HCTOREREREREREExIM
o4SciIiIizVphYSGTJO8mPj4eb29vBgwYwJo1a05Zd8KECZhMJmbNmnXW7T/55JOYTCYmT558yuOGYXDFFVdgM
pn44osvqsuPHDnCVVddhZ+fHz169GD9+vU1zps4cSLPPvvsWcchIiIicraU8BMRERGRZu32229nwYIFvPfee2z
cuJHLLruMYcOGceDAgRr15s2bx8qVK4m0jj7rttesWcNrr71G165dT1tn1qxZmEymk8r/9re/UVhYyLp16xg8e
DDjx4+vPrZy5UpVp12iSiiIiIyPlQwk9EREREmq3S0lL++9//8vTTTzNo0CDatm3LX/7yWchPebSXvXTrRSyqQ
AgQPcddddzJ07F5vNdlZtFxUVMWrUKF5//XVatGhxxjqpqak8k8++yzvPXWWycd27p1KzfeeCPt27fnjjjvuYOvWr
QBUVlYyYcIE5syZg8ViqcNdi4iIiNROCT8RERERabaqqqpwOp14eXnVKPf29ub777+ub/Q9E9Pl4FAuFwuRo8ezQMPPECnTp3
Ouu2JEycyYsQIhg0bdsrjxJSUlsI3HzzzycyePZvIyMiTjnfr+1o1FixZRVVXF/Pnzq2cJPv300wwePJjevXufdSwI
iIi5OIJPxERERFptvz9/U1OTuaJJ57gMGDOJ10/vWvf7FixQo7vWbardx9991n3e4tzS9N9+dmBM9dsrjYIDYu3tVyJTmrRbuUvHRwPMhz/xcHdNKLsmNDF
jxmnr3HvvvQwYMICrr776lMenTp2K1WqlTZs2zJs3jzfffJMd03bwz3/+k0ceeYQJEyaQkJDA9ddfT35+/rndu
IiIiEgtrO4OQEREETkfLz33nvcdttttGzZEovFQs+ePbnpppptISUkhJSWFF154Ydnb1T2bdvH/fccw8
LFiw4aebgCf/73/9YtGjRSRtx/FJgYCDvDvv+jbIhQ4Ywc+ZM5s6dy65du0dQhLS2P8+PFMnz5dG3iIiIhIvVHCT
ORERJqE9pW57g6hQW3a5Nn3526zZ8+mpKSE4uJiwsLCuP/++Ol0ISOCHH37g0KFDxMXFVdd1Op3cd999hZzJoli92
7d5/UVkpKCocOHaJnz541zlm6dCkvv/wy5eXl1LFq0iPTOdIKCgmqcO3r06NPG+Pbbbbx+8M9+PMUFMTVV1/Ntddey8iRI
7HZbPzhD3/g0UcfPe8+EBERETlBCT8RERFpErbbQtwdgjR3gSEQCPvy8/8xMrrvuupPW4Bs+fDijR49
m7Nixp2xm6xm6NChbNy4sUbZ2LFjPW2b69nNV6wo6n
U4qKyuBY5t4OJ3OOt+2iIiIyK8p4SciIiIizdr6HxYDBtGt25C1J4N3Zz5By4S2jBO7FpvNRkhIzWSyzWYjMjK
SxMTE6rKhQ4dyzTXXMGnSJPz9/encuX0Nc3x9fQkJCakuj4yMPOVGHTExExMaeMcfLkydx3332ObNkSgIEDB/Lee
+9x2WWX8Y9//IOBAwfWvQNEREREfqVJb9oxe/ZsWrVqhZeXF/369tNP16tXuDklEREREmpiSogJen/4Qd18xiBe
n3kOHnn155I33sdlsZ91Geno6OTk5DRLf/Pnz2blzJ3/605+qyyZNCmRCQgL9+vWjoqKCxx57rEGuLSIEL9+vWjoqKCxx57rEGuLSIiIhcmk
2EYhruDOJUPP/yQW265hTlz5tCvXz9mzZrFxx9/TFpaGuHh4Wc8v6CggMDAQPLz8wkICGiiEiH/2SVpmo15PPN+
1iVHuDuEkGudSnzTGRaQhuOPZ4s7fQUVEREROaLIz/J577jnGjx/P2LFjSUpKYs6cOf0PDWW2+5OzQRERERE
REREZEmqOmu4VdRUUFKSgrTpk2rLjObzQwbPpzbSwCbZQAodbzQwbNowVK1ac8pzy8nLKy8urfWdPCr
IJ9E4l/qkMS4iDcEdz5YTv3s20ZdoRERE5ALRJBN+OTk50OTk5OJ1OIiapRHRESwbdu2U56TlJTEvz0FwAdhXq
yRGEREREZHTKSwsJDAw0NlhiIiIyAWqSby2fyCIy2PZt+f1lt
pJ0rzxqX+bnzq88anPm986vPGpf5ufOfa54ZhUFhYSFhYGNCNEJyIiInqRLFxoaisViSVlnTY2kZGRp
zzH4XDgcDqrnfblqOFhFB6qlAUFBZ3ztvPVV541OPvV5410fNy71d+M7lz7z7XzD4RERFxtxtzcr1
sXLiwuszlcrFw4UKSk5PdGJmIiIiIiIiIiIiiEjT14lnRBPjZmnFmTtTwTMnpVUzzuccLPjWnUFs8Lt3b/r27cusNk5m7Nix3
ERERERERERERKR+NMk1/rKSEySmqcCb8bbriBw4nMzExdRvf5vuj/
+blzq78anPhcREZHmyGQYhuEERERERERERERERHxIEr4iYiYiIiIiIeBAl/EREREREREREREREERDzIB1EAV0v
IiIiIeBAlERERERERERERERERERERDzIBZ3wO3LkCjNGSIgICgoCDjGjRtHlIZVFRrecMHjwwkYzTxNhwoRGir
7NqlatcLLy4t+/fqxevXq095zkUUUKUcQxOTlklOaqCmtQKatdPtWHVs0q56HEnqZ9lJZXeoRhmH
iI6OhqTycSnn356xfVhRGh4oRxIn5RPPQWcnXWyPmMm0wmsrKyGXxiGgjIxGz6lDv78
/4eHhjBw5krS0tNOO9a5PE3b8ue7nMhbsT/bDu/QQuv/fqPjWrmKfyqKtJG3hF6d5xZS+qQjEh6xKs9cpoQSyLnhJj7tbEzGdqnOdlzdPdsTYU8vtZLoBCtdzjaeP91ibpvpjXeeaqcw5s
/4eHhjBw5krS0tDOep2d53dSlv/UsPz+vvvoqXbt2JSAgoDr4eXl5nXauGz8vVtT+sw5ttUHwyZTGLGTxeWOhB39iMGB

jjz9y0003MW7cONavX/vRTn8bRfpN9ZA0EYuW7VfKSP1WRcbm4r+Gk4NBBb/FQw8RG3syeuUq6e6W7ABSHDiB69HMGUn1xFLm4ZZG4ejqi/G0uuJuM+UkP4MtC+KNHSNEJzXTEFNNYgEhZd2GXZTP4L7j+aXs78DLPdc4AFEtYZXxf6uNS7o
IgRXHLJJaSmpjJ58murv/125s+f38CREuN4JDHA0fUpPZHaThpXUzymdzdEc/72fN9b8SlDCdjsfdlxKvD8
FxcWnPUfP8rqrS3+DnuXnIyYmhieffJKUfJKU1BTWrl3DS0qVFHdlH2l5bLQ1wQfdPaPZRZ5FuPzKNkTVtVLXxgmrzSzO6nZTTc16t8i4iISHNhMgzDC7rB161ak
pJYs2YNvXv3Bh5BuC1bb77hyiuvZP/+/vZt/+/URHR5/yvMGDB909OpyYoavnQmqwOGVjp+8e9rdWmasR//dNPz68/PLLAL
YOnXqSfVvuOEiouL+eKLL6rvr3/6/vTt25c5c+YQnstVqLmJjY6Z06RUSqbOnf7npP8mQ6LNmc2i9GWhPczgj2fk0pkqDvQTpLjWR2iYPh7Yo1ZZv3x9K7mIrxFWbKJ7oeohj6NhbaRFiDrUknu5VyslxqWm
8EHH+TLL7+s8YfhTfeSF5eHt98KjkxqNBm25vcGr7zpjpdlQbwtjFejBDDE3Lc3kwLgqJtHsLacH0ISCUbHGHWVaztSdmw0Ff4bKitlKn/nWkHoXz0KHRiNd5g1i6B1T1

8EHH+TLL7+s8YfhjTfeSF5eHt98800jROk5zqa/Fy9ezCWXXMLRo0cJCgpqtNg82eHDwkPD2fJkiUMGjTolOHDZW+Gg+bs24qh+zvrMZIQ9BVr8aUjNBhg1MmFgQMDcVnyK0+GVUXP1jYNFTLpwJ+t8cPdLmJdpHgBqY9djQcjOGqzwjrtPFLEXYcKhRFEFxwMDRB5B+1lUvlqcX9sUYsTeW2mmQVHy8Q
0LK8/Z9PfepbXv+DgYB0UGrsF9T5crTRRNvlqAQqPtFxPDLNnOPqkhQESqVOfBm8lakKFnwqBBZJBE5PQFHRkr8b+dgbzDiGq3Wyx5Fo0RD
dy5M9OmTaOkpKShw212KioqSElDJYdiwYdVlJrOZYcOGsWLFilOeU1ZWRmRkpMZtNLK5Dc5L2WsWwVFJElFqB2mkS2l0Oyzh4Lm8x08QRoEBk6hhJHIEr5+5jqxrdwlsNpbNm8WOPeSoVRtOzOLG8uSwPFjZOT0mTgbQ48XfzHWLxM6DQg5LcxQBZxHoWe

EBpaSne3t5uiswzRUVFMWfOHHr37yW7PS3HvvSeHpTQfUMu5sQXvGIB7v8DHpJMG7fZE3vtYyeAlxp0Tz88YB+LbJ2zQNkw/POo0LzS5kWRvhZjX6P24/ioeiOi5Fz38UwcZvmNqxL+p1mtFM2+ik6ZeJxkX
z7W89y8/fxo0bSU50pqysDD8/P+bNm0dSUtIp62bD8/QBNQiPl0Ttg4TDKEuarhDfPGLSvhjxK/xC3p/L+HiXrJ7fkUOtBx2gcuGzjTW1y7KFTjFkwfPWRq6uePB7e9Gc8TX/W5WH7s1j0M4Y+6+z0BvQiqZ4+hrXHDjJ0F6g3nNzPvD1mpYwzUx+cFWpPdjrVDwXMRvs2TZPuRI0jtETTwU2PwGyadeQWy1pS1mXnM4Xv2yM5dRHj5SK7+rq0dpD+mJZ6s3WNR8tyR5zsa5WSmq5O6Ww5X6lebCtpzhYq

```
777kxsuZp4sSJbNq0iWXLlrk7lAvC2fa3nuXnLzExkdTUVPLz8/nPf/7DmDFjWLJkyWmTfiIiIiLNgccl/O677
z5uvfXWWuskJCQQGRl5OmYGVVVVHDlyhMjIyLO+Xr9+/QDYuXOnEn6/EBoaisViITs7uOZ5dnb2afs3MjLynOr
Lz+rS379ms9no0aMHO3fubIgQhdOP8YCAAM3uayR9+/ZVwqoOJk2aVL251ZlmjelZfv7Opb9/Tc/yc2e322nbt
iOAvXr1Ys2aNbzwwgu89tprJ9XV+BYREZHmwuPW8AsLC6NDhw61fux08nJyeTl5ZGSklJ97qJFi3C5XNVJvLO
RmpoKHHt1TH5mt9vvp1asXCxcurC5zuVwsXLjwtOviJCcn16gPPsGDBgtPWl5/Vpb9/zel0snHjRo3lBqQx7n6pq
aka4+fAMAwmTZrEvHnzWLRoEa1btz7jORrndVeX/v41PcvPn8vlory8/JTHNL5FRESk2TAuYJdffrnRoOcPY9W
qVcayZcuMdu3aGTfddFP18cOHDxvt27c379/2LQ1ZHMe1x4G4+8f5RGZ7aNY5JPcmqUrs20nmElJLxSVOxh/MWE
dlg2nWfWwNFy4aE4Dhxe/OQYpdBuwnBdu/gF/ZltEJWlOgfXfMWjlFPHWe1hDqWxk0B2H1UPnGqN58afPqmSRUdN
fDBB4bD4TDeeecdY8uWLcYdd9xhBAUFGVlZWYZhGMbo0aONqVOnVtdfvny5Ybmd6yI9S9NsG1ZnfEBnNv8tjZRaxP
YuHGju26hWTnX/n788ceN+fPnG+np6Uk9f3NThb4y0Dp50YiZ1JMR4L4MDR9b0x5xdWlZzZRC0+LZHs3ZOWoJbhh
409e/YYhmEYU6dONUaPHl1df9euXYaPj4/xwAMPGFu3bjVmz55tWCwW45tvnHXLTQr59rfzz//vPHpp58a03b
sMDZu3Gjcc889htlsNr777jt33UKzc+eddxqBgYHG4sWLjczMzOpPSUlJdRO9y+tPXfpbz/LzM3XqVGPJkiVGR
kaGsWHDBmPq1KmGyWQyvv32W8MwNL5FRESk+bqgE365ubnGqNGjDI/Hg+DQFFjjcLCwurjGRkZBmB8//3
3hmEYxt69e41BgwYZwcHBhsPhMq2bWs88MADRn5+vpvuo0l76aWXjLi4OMNutxu9e+82bWjN/nbdjzZ+pb7/4nTvv6hk+kyQ3s7pTwOF6iYyeH/tc1aNvb+urskJRBhVkd6k6q/1W9z
```

```
          },
          "metadata": {}
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "1. Most applicants have no children (70.48%)\n",
        "2. It is observed that approval rate increases with number of children"
      ],
      "metadata": {
        "id": "eJdaRIgDElaX"
      }
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "tLw-IjAuNLWs",
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 313
        },
        "outputId": "5ca379ac-b433-44ef-df93-1b9d3e28ea8a"
      },
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            ″text/plain″: [
              ″  GENDER Car_Owner Propert_Owner  CHILDREN  Annual_income  \\\n″,
              ″0      M         Y             Y         0       180000.0   \n″,
              ″1      F         Y             N         0       315000.0   \n″,
              ″2      F         Y             N         0       315000.0   \n″,
              ″3      F         Y             N         0       166500.0   \n″,
              ″4      F         Y             N         0       315000.0   \n″,
              ″\n″,
              ″            Type_Income         EDUCATION Marital_status
Housing_type  \\\n″,
              ″0            Pensioner  Higher education        Married  House /
apartment   \n″,
              ″1  Commercial associate  Higher education        Married  House /
apartment   \n″,
              ″2  Commercial associate  Higher education        Married  House /
apartment   \n″,
              ″3  Commercial associate  Higher education        Married  House /
apartment   \n″,
              ″4  Commercial associate  Higher education        Married  House /
apartment   \n″,
              ″\n″,
              ″   Work_Phone  Phone  EMAIL_ID Type_Occupation  Family_Members  label
\\\n″,
              ″0           0      0         0      Unemployed               2      1
\n″,
              ″1           1      1         0         Unknown               2      1
\n″,
              ″2           1      1         0         Unknown               2      1
\n″,
              ″3           1      1         0         Unknown               2      1
\n″,
              ″4           1      1         0         Unknown               2      1
\n″,
              ″\n″,
              ″   Employed_years   Age  \n″,
              ″0            0.00  52.0  \n″,
              ″1            1.63  38.0  \n″,
              ″2            1.63  44.0  \n″,
              ″3            1.63  38.0  \n″,
              ″4            1.63  38.0  ″
            ],
            ″text/html″: [
              ″\n″,
              ″  <div id=\″df-f37b5432-6ecf-43de-8be4-fdd41fd92943\″ class=\″colab-df-
container\″>\n″,
              ″    <div>\n″,
              ″<style scoped>\n″,
              ″    .dataframe tbody tr th:only-of-type {\n″,
              ″        vertical-align: middle;\n″,
"    }\n",
"\n",
"    .dataframe tbody tr th {\n",
"        vertical-align: top;\n",
"    }\n",
"\n",
"    .dataframe thead th {\n",
"        text-align: right;\n",
"    }\n",
"</style>\n",
"<table border=\"1\" class=\"dataframe\">\n",
"  <thead>\n",
"    <tr style=\"text-align: right;\">\n",
"      <th></th>\n",
"      <th>GENDER</th>\n",
"      <th>Car_Owner</th>\n",
"      <th>Propert_Owner</th>\n",
"      <th>CHILDREN</th>\n",
"      <th>Annual_income</th>\n",
"      <th>Type_Income</th>\n",
"      <th>EDUCATION</th>\n",
"      <th>Marital_status</th>\n",
"      <th>Housing_type</th>\n",
"      <th>Work_Phone</th>\n",
"      <th>Phone</th>\n",
"      <th>EMAIL_ID</th>\n",
"      <th>Type_Occupation</th>\n",
"      <th>Family_Members</th>\n",
"      <th>label</th>\n",
"      <th>Employed_years</th>\n",
"      <th>Age</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>0</th>\n",
"      <td>M</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>0</td>\n",
"      <td>180000.0</td>\n",
"      <td>Pensioner</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>Unemployed</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>0.00</td>\n",
"      <td>52.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>1</th>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>N</td>\n",
"      <td>0</td>\n",
"      <td>315000.0</td>\n",
"      <td>Commercial associate</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>Unknown</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>1.63</td>\n",
"      <td>38.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>2</th>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>N</td>\n",
"      <td>0</td>\n",
"      <td>315000.0</td>\n",
"      <td>Commercial associate</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>Unknown</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>1.63</td>\n",
"      <td>44.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>3</th>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
              "      <td>N</td>\n",
              "      <td>0</td>\n",
              "      <td>166500.0</td>\n",
              "      <td>Commercial associate</td>\n",
              "      <td>Higher education</td>\n",
              "      <td>Married</td>\n",
              "      <td>House / apartment</td>\n",
              "      <td>1</td>\n",
              "      <td>1</td>\n",
              "      <td>0</td>\n",
              "      <td>Unknown</td>\n",
              "      <td>2</td>\n",
              "      <td>1</td>\n",
              "      <td>1.63</td>\n",
              "      <td>38.0</td>\n",
              "    </tr>\n",
              "    <tr>\n",
              "      <th>4</th>\n",
              "      <td>F</td>\n",
              "      <td>Y</td>\n",
              "      <td>N</td>\n",
              "      <td>0</td>\n",
              "      <td>315000.0</td>\n",
              "      <td>Commercial associate</td>\n",
              "      <td>Higher education</td>\n",
              "      <td>Married</td>\n",
              "      <td>House / apartment</td>\n",
              "      <td>1</td>\n",
              "      <td>1</td>\n",
              "      <td>0</td>\n",
              "      <td>Unknown</td>\n",
              "      <td>2</td>\n",
              "      <td>1</td>\n",
              "      <td>1.63</td>\n",
              "      <td>38.0</td>\n",
              "    </tr>\n",
              "  </tbody>\n",
              "</table>\n",
              "</div>\n",
              "    <div class=\"colab-df-buttons\">\n",
              "\n",
              "  <div class=\"colab-df-container\">\n",
              "    <button class=\"colab-df-convert\"
onclick=\"convertToInteractive('df-f37b5432-6ecf-43de-8be4-fdd41fd92943')\"\n",
              "            title=\"Convert this dataframe to an interactive
table.\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\" viewBox=\"0
-960 960 960\">\n",
              "    <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220
220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-
160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",
              "  </svg>\n",
              "    </button>\n",
              "\n",
              "  <style>\n",
              "    .colab-df-container {\n",
              "      display:flex;\n",
              "      gap: 12px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert {\n",
              "      background-color: #E8F0FE;\n",
              "      border: none;\n",
              "      border-radius: 50%;\n",
              "      cursor: pointer;\n",
              "      display: none;\n",
              "      fill: #1967D2;\n",
              "      height: 32px;\n",
              "      padding: 0 0 0 0;\n",
              "      width: 32px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert:hover {\n",
              "      background-color: #E2EBFA;\n",
              "      box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n",
              "      fill: #174EA6;\n",
              "    }\n",
              "\n",
              "    .colab-df-buttons div {\n",
              "      margin-bottom: 4px;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert {\n",
              "      background-color: #3B4455;\n",
              "      fill: #D2E3FC;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert:hover {\n",
              "      background-color: #434B5C;\n",
              "      box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",
              "      filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",
              "      fill: #FFFFFF;\n",
              "    }\n",
              "  </style>\n",
              "\n",
              "    <script>\n",
              "      const buttonEl =\n",
              "        document.querySelector('#df-f37b5432-6ecf-43de-8be4-
fdd41fd92943 button.colab-df-convert');\n",
              "      buttonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "\n",
              "      async function convertToInteractive(key) {\n",
              "        const element = document.querySelector('#df-f37b5432-6ecf-43de-
8be4-fdd41fd92943');\n",
              "        const dataTable =\n",
              "          await
google.colab.kernel.invokeFunction('convertToInteractive',\n",
              "                                                     [key], {});\n",
              "        if (!dataTable) return;\n",
              "\n",
              "        const docLinkHtml = 'Like what you see? Visit the ' +\n",
              "          '<a target=\"_blank\"
href=https://colab.research.google.com/notebooks/data_table.ipynb>data table
notebook</a>'\n",
              "          + ' to learn more about interactive tables.';\n",
              "        element.innerHTML = '';\n",
              "        dataTable['output_type'] = 'display_data';\n",
              "        await google.colab.output.renderOutput(dataTable, element);\n",
              "        const docLink = document.createElement('div');\n",
              "        docLink.innerHTML = docLinkHtml;\n",
              "        element.appendChild(docLink);\n",
              "      }\n",
              "    </script>\n",
              "  </div>\n",
              "\n",
              "\n",
              "<div id=\"df-8b9238ec-a6c0-4a0d-8aca-dcf84bf2a15a\">\n",
              "  <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-
8b9238ec-a6c0-4a0d-8aca-dcf84bf2a15a')\"\n",
              "            title=\"Suggest charts\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "<svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0 0
24 24\"\n",
              "     width=\"24px\">\n",
              "    <g>\n",
              "        <path d=\"M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-
.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\"/>\n",
              "    </g>\n",
              "</svg>\n",
              "  </button>\n",
              "\n",
              ″<style>\n″,
              ″  .colab-df-quickchart {\n″,
              ″      --bg-color: #E8F0FE;\n″,
              ″      --fill-color: #1967D2;\n″,
              ″      --hover-bg-color: #E2EBFA;\n″,
              ″      --hover-fill-color: #174EA6;\n″,
              ″      --disabled-fill-color: #AAA;\n″,
              ″      --disabled-bg-color: #DDD;\n″,
              ″  }\n″,
              ″\n″,
              ″  [theme=dark] .colab-df-quickchart {\n″,
              ″      --bg-color: #3B4455;\n″,
              ″      --fill-color: #D2E3FC;\n″,
              ″      --hover-bg-color: #434B5C;\n″,
              ″      --hover-fill-color: #FFFFFF;\n″,
              ″      --disabled-bg-color: #3B4455;\n″,
              ″      --disabled-fill-color: #666;\n″,
              ″  }\n″,
              ″\n″,
              ″  .colab-df-quickchart {\n″,
              ″    background-color: var(--bg-color);\n″,
              ″    border: none;\n″,
              ″    border-radius: 50%;\n″,
              ″    cursor: pointer;\n″,
              ″    display: none;\n″,
              ″    fill: var(--fill-color);\n″,
              ″    height: 32px;\n″,
              ″    padding: 0;\n″,
              ″    width: 32px;\n″,
              ″  }\n″,
              ″\n″,
              ″  .colab-df-quickchart:hover {\n″,
              ″    background-color: var(--hover-bg-color);\n″,
              ″    box-shadow: 0 1px 2px rgba(60, 64, 67, 0.3), 0 1px 3px 1px rgba(60,
64, 67, 0.15);\n″,
              ″    fill: var(--button-hover-fill-color);\n″,
              ″  }\n″,
              ″\n″,
              ″  .colab-df-quickchart-complete:disabled,\n″,
              ″  .colab-df-quickchart-complete:disabled:hover {\n″,
              ″    background-color: var(--disabled-bg-color);\n″,
              ″    fill: var(--disabled-fill-color);\n″,
              ″    box-shadow: none;\n″,
              ″  }\n″,
              ″\n″,
              ″  .colab-df-spinner {\n″,
              ″    border: 2px solid var(--fill-color);\n″,
              ″    border-color: transparent;\n″,
              ″    border-bottom-color: var(--fill-color);\n″,
              ″    animation:\n″,
              ″      spin 1s steps(1) infinite;\n″,
              ″  }\n″,
              ″\n″,
              ″  @keyframes spin {\n″,
              ″    0% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-bottom-color: var(--fill-color);\n″,
              ″      border-left-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    20% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-left-color: var(--fill-color);\n″,
              ″      border-top-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    30% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-left-color: var(--fill-color);\n″,
              ″      border-top-color: var(--fill-color);\n″,
              ″      border-right-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    40% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-right-color: var(--fill-color);\n″,
              ″      border-top-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    60% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-right-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    80% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-right-color: var(--fill-color);\n″,
              ″      border-bottom-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    90% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-bottom-color: var(--fill-color);\n″,
              ″    }\n″,
              ″  }\n″,
              ″</style>\n″,
              ″\n″,
              ″  <script>\n″,
              ″    async function quickchart(key) {\n″,
              ″      const quickchartButtonEl =\n″,
              ″        document.querySelector(’#’ + key + ’ button’);\n″,
              ″      quickchartButtonEl.disabled = true;  // To prevent multiple
clicks.\n″,
              ″      quickchartButtonEl.classList.add(’colab-df-spinner’);\n″,
              "      try {\n",
              "        const charts = await google.colab.kernel.invokeFunction(\n",
              "            'suggestCharts', [key], {});\n",
              "      } catch (error) {\n",
              "        console.error('Error during call to suggestCharts:', error);\n",
              "      }\n",
              "      quickchartButtonEl.classList.remove('colab-df-spinner');\n",
              "      quickchartButtonEl.classList.add('colab-df-quickchart-
complete');\n",
              "    }\n",
              "    (() => {\n",
              "      let quickchartButtonEl =\n",
              "        document.querySelector('#df-8b9238ec-a6c0-4a0d-8aca-
dcf84bf2a15a button');\n",
              "      quickchartButtonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "    })();\n",
              "  </script>\n",
              "</div>\n",
              "\n",
              "    </div>\n",
              "  </div>\n"
            ]
          },
          "metadata": {},
          "execution_count": 1315
        }
      ],
      "source": [
        "merged_data.head()"
      ]
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "ufiIONbhKS2p"
      },
      "source": [
        "#### Email ID\n"
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "analyse_categorical_column(\"EMAIL_ID\")"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
        "height": 507
      },
      "id": "G-zvteaJ4ZlE",
      "outputId": "7b2fadf1-4552-4af4-dd89-a3eb06456753"
    },
    "execution_count": null,
    "outputs": [
      {
        "output_type": "display_data",
        "data": {
          "text/plain": [
            "<Figure size 1400x500 with 2 Axes>"
          ],
          "image/png":
"iVBORw0KGgoAAAANSUhEUgAABPgAAAHqCAYAAACZYfCtAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAACVn0l
EQVR4nOzdd3xV5eHH8e8dSW72giwIQzayh4AMESJTFMGB4iwVq6JFras/pVarKO5VrVYLVnBVRLEVRVBQNkGmE
PYmCRCyc/f5/YFejQkIkuQkN5/365WXcs5zz/negJh885AMESJTFMGB4iwVq6JFras/pVarKO5VrVYLVnBVRLEVRVBQNkGmE
AAAAAAKjDKPgAAAAAAACAOoyCDwAAAAAAKjDKPgAAAAAAACAOoyCDwAAAAAAKjDKPgAAAAAAACAOoyCDwAAA
AAAAKjD7GYHqC5+v18HDx5UdHS0LBaL2XEAAAAAASZJhGCoqKlJaWpqsVn7eDgAAzlzQFnwHDx5Uenq62TEAAC
ASu3bt0+NGzc20OwYAAAgCQVvwRUdHSzr+hVNMTIzJaQAAAIDjCgsLlZ6eHvh6BgAAoHoK2QyhvExtv3EfAFbcH342O5MTExFHwAA
ACodVhGBgAAVBUW/QAAAAAAAAAADqMAo+AAAAAAAAAoA6j4AMAAAAAAAADqsKBdgw8AAKAq+Hw+eTwes2OgDkJCZH
NZjM7BgAAqEco+AAAAACphGIays70Vn59vdhTUQXFxcWrevLnZUQAAAIoAADwCz6fL1DuJSYmRXzUGG1dWvVhoBSAIAAhqcEsMwvV
FpaqtzcXElSamqqYkAAEB9QMEHAADwCz6fL1DuJSYmmh0OdUh4eLgkKTc3V0lJSXzcFgAAVADUUx4eLgkKTc3V0lJSTyuCwAAqh2bbAAAAPzCj2v
uRUREmJwEdWPf3ZYvxEAANQECj4AAIAT4LFc/b82QEADWJgg8AAAAAAAAAAACowyj4AAAAUOcNHDhQt956a2XVcZqky
```

cWKFcrMzNT111+v3r17Bwq/ymRkZKhjx44aP3681qxZo5UrV+raa6/VeeedV+6Hu2PGjFFRUZFuvvlmnX/++eV
mG956663Ky8vTlVdeqVWrVmnHjh36/PPPdcMNN8jn8ykqKkoTJkzQ3XffrYULFwbyW63sHQcAAOqvKv1KKDMzU
88//7ymT58ui8VSlZf+VVOnTlVsbGzg4+drvwAAANQHb7zxhjIyMhQbG1vh3NixY7V69WqtX79ekvT444/rj3/
8o7p3767s7GzNnTu33NMTEREReuvfee3XVVVepb9++ioqKOnvvvvXfS+1ssFn388ceKj4/XgAEDlJGRobPOqvC6
6KjozVq1CitW7cusLnGj9LSOrRkyRL5fD4NGTJEHTt21OTJkxUXxco8Z588kn1799fo0aNUkZGhvr166fu3bv
/ps8ZAABAMLAYhmH85hdbLProo48CP+197rnndO+dedd0edd5b7CarP55PValV6erp2796tN998U3fddZeOHTsWGOP1e
uVwOPTBBx/okksu0bXXXqvCwsJy0/R/R+9dVXGjRokPLy8ip9RNflcsnlcgV+XVhYqPTOdBUUFCgmJua3vsFd
OAVQFHtEFzOFOOrVr1y41b95cDofD7DhV6uuvv9b6B8P+vY/TJk5Wfn1+j2YLJyf4MFRYWK
jY21rSvUwEAQPCp0jX4rrnnmGmVkZJQ7NnToUK9T/Sw7/0P6y9/zqA/gQV3Cm/u/Q0U8TuXLcjpZsLXzEsm1P5IIPb17yXyRJAcljzQgxC/+4AAcNHfR6WvnuMtMaKEH/xDGYqCCsI6WlZuXwv8yCN8/cqLIOf/19ZzQZ0Zrp+K6Yhf1wxGYWe4
qn8R40vQSKJDMz5Q0nLAOCjaRKkvnoGRG92/IRiPdKCNrRGmFX6fKXyEhBFSKb0ZNpOYk39wQiFgD9kx9wd4ol3QlG7FrP5RMw7KUVnbZYOrRdVXSaXeJYJT8BxN1BBtqW4+ZJYYZH/6Z3j3j7ozG0vyrAYT/bYpz0oC/HGCH3Qxs31JyDlPH+LwK+sN7XtDeWF+o0H+W/5rqrrqEM3rr2ap+wnwl7UtR90r57k6Ch8Hu/hp1nm7xKSd9rDkb4FNb2QuwAkDeaKEBowZRkN+dTOEqrW7uCyNbxQexPNXNRNHV4GUU1WtIevPtS8uknMJx61jkaSI0s1I3ABWX3FwT/ijGjfpYb0O9FTCNmE+s+0AQ/uy/B/1Rl3qv/0RZo2ZKbqLn3pwMrcVnNTqU8sW8WCIrDrybXNWPnD8WLPdOIyQ/2d7s/tGXIOeukv4vxs+o62jL4FfuCp7C/K0CItLKuBW8GiN8SPtHxKGW3jHIX6XmpAnsiLw6NZ/aonW+J8NeCyWhHkUKZ2sxUzTtJyi2uFtTR8VJ/sq1NkDBE5X/SYHlSt9fJgTZEvfqTSu46Fs6uqb/fSBMOXz3ueu6+cKJ81ZP+DdRX2KfW/MA52iTqKhKzfmFRKxF/fWqehz6L+4JUUGEn5QpWkiC/Qu8=
88sx7RsvnG+uNlGQGnnp1iwZmdS5NzVFvgBgvoX0Zz4Pkv2xLOb5q2KJhXdIbnWGqwXLsenEHvoGnVhpFRFtwK4rnoIYfEqZo3yFavlTYa0f5LpQ/D3ve6Jyj5yjJ0f8bNI+Hz78WIpYfDOqhZ88vn/XPHIdbyBb+b+VJ2FrGCzVDnSXw+Wpp5qFX/5s/sESTBa9s7t6/YeCj9MzamdeqVd1kPaNEyZR9j/pCjWTe9dHUJzUzbthWDI7h5qP8u3c9yHLnlD1BszoBtxUiO3rtAYBONmKftMOoSzeWiIU5bAbC4LXwBUeaDhFX3J9vSOP+zRRuXtm5LEB5Yf5aq2ykdMPVt/8/ux8CrY2UtqO9t6uQoT+Pi0kRg3pbHO0UoO6CIfRCnp9NXcHDf8=

```
G1DrM4AMAAAAAwGSGYWhrTrEy9xzThgP52nCgQFuzi+X21Y8yrzLZhU5lFzr17fafdnIPsVnUoVGszmmWoJ4/f
MRGMNMPoOADAAAAAKCGlbl9+m7fMWXuPqbVe47pu73HVOg89Udb6yuPz9B3e/P13d58/WPxTlksUuukaPVsHq9
zmifqnGYJSoll9j3qHwo+AACAEwjSlUxQA/izA+CX/H5DmXuP6cvvc7R851FtOlgor5+/K86UYUhZOUXKyinS2
8v3SpLSE8LVs1nC8Vl+zRPUomGUySmB6kfBBwAA8AshIccf9SktLVV4eLjJaVAXlZaWSvrpzxKA+snp8embbUc
O//tsLdySqyPFbrMj1Qv78sqOL++AZq85IElqEBWqHkOTdF6bhhp6dooSIkNNTghUPQo+AACAX7DZbIqLi1Nub
q4kKSIiQhaLxeRUqAsMw1Bpaalyc3MVFxcnm42F4IH65liJW19uztH873POzbYjpu1oi58cKXZr3qZszduUrQf
nbFSfFoka2TFVwzqkKC6Csg/BgYIPAACgEikpKZIUKPmAOxEXFxf4MwQg+OO9Wqovvs/WF9/nKHPPMfl49LbW8
voNfbPtiL7ZdkQP/FD2XdgpVUPPpuxD3WYxgnSBkMLCQsXGxqqgoEAxMTE1fv+NGzfW+DOBBJ8OHTqYHQGo93w
+nzwej9kxUIeEhIScdOae2V+nAjhzhmFow4ECfbHp+Ey9rJwisyPhDIXYLDq3RQON7Hi87GNnXtQ1zOADAAA4C
ZvNxmOWAABJOuEil95btVfvrNynA/llZsdBFfL4DC3aeliLth7W/83ZoL4tj5d9Q85OUWw4ZR9qPwo+AAAAAAB
OYtmOo3p7xR59sSlbHl9QPgSHn/H4DH2ddVhfZx3W/32OUX1bJurS7uka1iFFNitr8qJ2ouADAAAAAOAXCpOez
c7cr5kr9mpbbrHZcWASt8+vr7IO66usw2oUF65r+jTVlT2b8AgvahOKPgAAAAAAfrDxQIHeXr5Hn6w7qFI3O+D
iJwfyy/T4Z1vOwoJtGtOtkW7o21wtGkaZHQuQRMEHAAAAAKjnnB6f5q47qLdX7NW6fflmxOEtV+r26e3lezVzx
V6d17qhfte3uQaObmh2LNRzFHwAAAAAgHpp5+FizVyxV//J3K+CMnZMx+kxDAXW6mudHKXrz22uMdOayRHC5ly
oeRR8AAAAAIB65fuDhXpm/lYt2JIjgz0zUAW25hTrzx9tOJOfb9GV5zTRtX2aKSXWYXYs1CMUfAAAAACAemHH4
WI9M3+r/rfhEMUeqsWxUo/+/vUOvf7NTg3vkKoJ/Zqrc3qc2bFQD1DwAQAAAACC2r68Uj2/YJs++u6AfH6aPVQ
/j8/QJ+sO6pN1BzWobZLuHtpG7VJjzI6FIEbBBwAAAAAISrmFTr24cLveW7VPbp/f7DiopxZuydXXWbka1TlNd
13QRk0SI8yOhCBEwQcAAAAACCp5JW698vV2/Xv5Hjk9FHswn9+QPl57UP/bcEjjejbRbYNbKimaNfpQdSj4AAA
AAABBodDp0T8X79SbS3ar2OU1Ow5Qgcdn6N/L9+g/mft1Q99munlgCOU7QsyOhSBAwQcAAAAAqNPK3D79a+kuv
bZ4p/JLPWbHAX5Vmcenv3+9Q++v3qc7L2ijK3qmy2a1mBOLdRgFHwAAAACgTvL7Db2zaq+enb9NR4pdZscBTtu
RYrf+/NEGvbVstx4Y2V79WjUwOxLqKAq+arI1JNHsCACCQAezAwAAANRSmw4W6P8+2qi1+/LNjgKcsS3ZRbr6j
RXKaJekP49op7MaRpkdCXUMBR8AAAAAoM4odnn19BdZemvZHvn8htlxgCr15eZcLdp6WNf2aaa7hrRWCi1DU4
Nf1IAAAAAAHXC/zYc0sNzv1d2odPsKEC18fgMvfHtLs3/PkfTLu2k3mfxhCB+HQUfAAAAAKBW23+sVA/O2aivs
g6bHQWoMXvzSnXl68tZ9mundYW4WH2syOhFrManYAAAAAAAAqYxiG/r1st4Y8u5hyD/WSYUjTl+7W8OcXa9X
uPLPjoBaj4AMAAAAAA1Dp7jpboyteX68GPN6nU7TM7DmCq3UdLdcU/lunhud/L6eG/B1REwQcAAAAAqDX8/uPrj
wl9drGW72TGEvAjvyG9uWSXRjz/jTL38N8GyqPgAwAAAADUCjsOF+vSV5fqkU+/l9PrNzsOUCvtPFKiy15dpkf
/y2w+/ISCDwAAAABguv9k7teI5xdrzd58s6MAtZ7fkF7/ZpdGvvCNvtt7ZOw4qAUo+AAAAAAApnF5fbr7g7X60
wfr5PIaZscB6pQdh0t06avLNPWzzXJ5mc1Xn512wbd48WKNGjVKaWlpslgsmjNnTuCcx+PRX79f+5oyM1lJ
paWm69tprdfDgwXLXyMvL0/jx4UTE6O4uDhNmDBBxcXF5casX79e/fv3l8vtbFbal
leqUc8vOgeZB8yOAtRZPr+hfyzaqQtf+FbbcorMjgOTnHbBV1JSos6dO+vll1+ucK60tFRr1qzRgw8+qDVr1mj
27NnKysrSRRddVG7c+PHjtWnTJs2fP1+ffvqpFi9erIkTJwbOFxYWaujQoWrWrJkyMzP1zDMaiz
94iAAAAAKC2mbd+v4Y+85W2Hi4zOwoQFLblFuSvy/V/O9zzI4CE1gMw/gJNc6AtFos++ugjjRkz5qRjVq1a
pZ5N5+G2xKa1s3XgbxFHdLi0rvOIbDN6eV9+76wGYJu7+KiC4zrFBmeIUX9Xi/j5vKNSSGyMaZjpVJVy/dgPk9Sm1
OOUd79uxRkyZNtHnzZrVv7rVv316rVq1Sjx49JEnz5s3TiBEjtH//fqWlpenVjz/Wn/uuEavBJQAAAIAjNwQr
szRli1bTilbYWGhYmNjVVBQoJiYmNv/6Fn+z2mxfQZkcAAFQxs790Bczm9xt68D+rNXNNjiSL2XG
AoGOxSJMHt9btg1vKYuGcKlotRCQ7knSOk1YExgwYMCBQ7knSO
KFDlZWVpWPHKl88OuVyqbCwsNwHAHD2Ors3A0qh2FIz365VX94O1Mlq/ZcVBDqrXgcz
duvfee3XllVcGfjqZe9zmq/ZnZ2tpKScuPsdrsSEhKK90nJoB2d0lWxsbGBj/T09q/T09Kp+OwAAAACA3
2jZthwNmvalNh12mxOFqBc+35SjMX9fqfjcP8LsMw9Morr1TXbQLuv/9+FRQUBD727dtX5Ltr7dt
X7fcEAAAAAPy65z7boFPFvrFIB3R5Qo7JyinTRS0vOzbbDZkdBNbNXx0V/LPf27NmjhXCUFBm5ueXGe
71e5eXXlKSJ1JTAmJ6f8opA//vrHMb8UFhamsLCwqnkDAAAAAIAzUOz06L4fvYtw+WhW11W6r1hX6m+BdgmGX6P1dxC
bTRzQwuw4qQCZVPoPvx3Jv27Zt+v/sLrt5WYmFn+DuVK2Gc+6oA6fu6BOLoW8IWcRvZSf1zLT5
X7fcEAAAAAPy65z7boFPFvrFIB3R5Qo7JyinTRSOvOzbbDZkdBNbNXxOV/LPf27NmjhQsXlltbJCUlRbm5ueXGe
tqOaaP4+PiqjgwAAAAAqGJZBws0eNr9AGby+Q099r8tmvzudLhYa9eu1dqlayVJu3b
t0tq1a7V3797AR5PzW4c4Z6fs7Gx4T4+F7tdu3YaNmyYbrrpJq1cuVJLly7V5MmTNcUVYwsYrT1AW3Tz1anGMe1fulhKL3aUFpbPy54FSaw+QuhHcd3XnmjVa6gTHKAn7I9OwvmoTN1tC
kbXmMw4B2TJCqVyfWwPGQ/5lpO0DkhOKcMgmbMhgHZa7kzAuvVg1pBLgkuFbomrTVe2Z+mKzTKWgMo6FuGH7tnOFHD+YUt5WRUDPmNTGFKmEYcMAgbNmwKlCewyL9H8A/ZE/FmDoG4JhOuvXmSzQ
ZDuaXmROFVcxiGMZp/Y379ddf6/zzz69w/brrrtPf/3Jn8cvT+ZT7jmbL+LOlyz3Q/YgGqX7eSwrQEo3/y2/+WkbiqJE0fOSK3yUPGGmv6MvSbtxDKZHZ59KQfm3GRNCbP4c7Zle1OPQjK9kK6BVekCW7sHsUnzTRUOtP7vZpbvPYuDebeqoHeK1HGdPdSDVVeGKvkWBvC+lC
1Q3iZVmnNaewvXPsaffBsomk1g6Z7Rw7amBQGYfOHHe1VA7IsZQ+4mbFhg4QKc8Ndr/eszuq1J3/zKqYk0aQBcfuO7Bae3wwOw8AowPtp0sqYRhRHBZ1wx3l/UEeFb+Yr7OlwRYFTb3oSOVAe
tFULbKPIjZ3ztpR2GsfGBgD1iFr5KL5j3sQzXqFk/Gl9sNt0OGLsyazbJ2lCs2PUE7Dx0jD/PPcHQbG41LI
DOHUlzOC3/TqYDtYU93fcnSPnIUzZb7UhcpFx/hjqP9zsFn1BpBYnqMLCJmDmmD08iKO5X9e4yo3e4wFp+/1l8sfi0W2cc3I17
```

oPZ+sPrC7WpLNbsKADOwMOffi+n16dbBrYOOwp+Iwo+AAAAAMBpy9q5R3/411Lt8lDuAcFg2rwsOT1+3XlBa7O
j4Deg4AMAAAAAnJbvvt+qW97O1CE/G8UAweSFBdvk9vp13/C2ZkfBaWL1UwAAAADAKVu8eoN+/+/vKPeAIPXqo
h3669xNZsfAaWIGHwAAAADgVxmGoc++ydR9/9upQkWZHQdANfrXkt1yef16dHQHWSzsnlMXMIMPAAAAAHBShmH
og/lL9Kf/7lahIs2OA6AGzFqxV3/6YL38fsPsKDgFzOADAAAAAJyQz+fT+599rb8tPqpSK+UeUJ98uGa/3D6/n
r28s+w25ojVZhR8AAAAAIBKeTxevTP3cz29LF8ltjiz4wAwwdx1B+X2+vTild0Uaqfkq634nQEAAAAAVOBOuTT
jg7l6fslhFVDuAfXa55tydPPbmfL6/GZHwQlQ8AEAAAAAyikuKdUb787R66sP62hIQ7PjAKgFFmzJ1f2zN5gdA
ydAwQcAAAAACChzOvXme3POzneHlRPayOw4AGqRDzL36+kvssyOgUpQ8AEAAAAAJB1fc2/WR5/po3XZ2u9obnY
cALXQiwu3a+aKPWbHwC9Q8AEAAAAA5Pf79eH/5uv95du0O7y12XEA1GJTPt6k+d/nmBODPOPBBwAAAAD1nGEY+
t/CbzTrq7XaFt5OhixmRwJQi/n8hm57Z40y9xwzOwp+QMEHAAAAAPXcouWZmv7ZEm12tJefbxMBnAKnx6/fz1i
lXUdKzI4CUfABAAAAQL22atOmvTFnvjaEtJNHdrPjAKhDjpV6NGHGKhWUecyOUu9R8AEAAABAPfX9tp365wf/V
aZayaVQs+MAqIN2Hi7RrTPXyOvzmx2lXqPgAwAAAIB6aNfeA/rnOx9pmStdpZZws+MAqMO+3X5ED83dZHaMeo2
CDwAAAADqmUO5R/TGux9paX6MCmOxZscBEATeXr5XO5fsMjtGvUXBBwAAAAD1SF5+gd54d7aWH3AqOzTN7DgAg
sgj/92sRVsPmx2jXqLgAwAAAIB6orikVP96/2Ot2HpIu8PbmBOHQJDx+Q1NmrVGOw8Xmx2l3qHgAwAAAIB6wOV
y66OPP9XStZu1O6azvHw7CKAaFDm9uv3d7+T2sulGTeJvdAAAAAAIcoZh6JP5X2vxikxlJ3RRkT/E7EgAgtjGA
4V66osss2PUKxR8AAAAABDkVny3Qf9b+KOK49vogDfK7DgA6oHXv9mpb7cdMTtGvUHBBwAAAABBbM/+g3r3k3k
qskZrszfJ7DgA6gnDkO58f63yStxmR6kXKPgAAAAAIEgVFZfo37M/1cEjBdoS1lqGLGZHAlCP5Ba5dM9/1pkdo
16g4AMAAACAIOTz+fT+3C+OYfN25SROVrHPbnYkAPXQl5tz9day3WbHCHoUfAAAAAAAQhBYsWamvlq2SN6mt9rg
jzY4DoB579L+btTWnyOwYQY2CDwAAAACCzOZtOzX7swWyRCZqrbuh2XEA1HMur1+3zfpOTo/P7ChBi4IPAAAAA
IJIfmGR3vn4MxWWurTR1kI+g2/7AJgvK6dIj/1vs9kxghZ/OwMAAABAkPD5fPrg0y+OdeceFSa21zFviNmRACD
grWV7tGBzjtkxghIFHwAAAAAEicUr1mjx8kzFpDTVptJos+MAQAV3/2e9cgudZscIOhR8AAAAAABAEdu87qNmff
SlHeLjWetJkyGJ2JACoIK/Erbs+WCfDMMyOElQo+AAAAACgjispLdPbH/1XR48VqCT2LOV6Qs2OBAAn9M22I5q
+dLfZMYIKBR8AAAAA1GGGYejjL77Sxi3blNakmVYXxZgdCQB+1TNfbNXhIpfZMYIGBR8AAAAA1GGbtu7Qgm9XK
jW5oTJLG8jNrrkA6oAil1ePf7bF7BhBg7/5AQAAAKCOKnM6NefzhXK53SoNT9YuZ7jZkQDglM3+br8y9xwzOOZ
QoOADAAAAgDrqq6WrtClrh9LTG2tpQazZcQDgtBiG9JdPNsrvZ8ONMOXBBwAAAAB1OP5DOfrvwm8UHxuj71OJK
vLZzY4EAKdt44FCvbNqr9kx6jwKPgAAAACoY3w+nz6at1BwoUlpCqDcVRZkcCgN/sqc+zlF/qNjtGnUbBBwA
AAAB1zMq1G7Xyuw1qOihNSwri5ZfF7EgA8JsdK/XoqS+yzI5Rp1HwAQAAAEAdcqygUHM+/OohISE6YGmgXE+o2
ZEA4IzNWrFXGw8UmB2jzqLgAwAAAIA6wjAM/W/hN9qz/6CSUhtpVWGM2ZEAoEr4DemhTzaZHaPOouADAAAAgDp
i8/Zd+mrpaqUmN1RmSbxcBt/SAQgeq/cc0+w1+82OUSfxfwMAAAAAqAOcLpdmf7ZATpdL1qgG2l4WbnYkAKhyU
z/bomKX1+wYdQ4FHwAAAADUAV8tXa1NWdvVvEkrjrSmKlsHGGgCCOOEil57/cqvZMeocCj4AAAAAqOUOZOfqwu
/UVxsjEosEdrldJgdCQCqzfSlu7U9t8jsGHXKaRd8ixcv1qhRo5ZMmaLU1FSFh4crI
yND27ZtKzcmLy9P48ePV0xMjJOLi4jRhwgQVFxeXG7N+/Xr1799dDdD6enpmjZt2um/OOwAAAACo4/x+v+bMW6j
DR/OUltxQXbvvAQhiHp+hafO5HHzT9d9Vj5Rp5+V/rvT8GnT9MILT+mbb75R+/bt1dbWVvfu3a0bN24o
DR/OUltxQa4qiJWbvAQhiHp+hafOyzI5Rp5x2wVdSUqLON+Tvr5ZdfrvT8tGnT9MILL+jVV1/VihUrFBkZqaFDh
8rpdAbWjBkzRsnbm5o4cKEX//+q3/3ud4Hz+UXF2nIlpxlGDWmzPfmKR3HdVLM0mqHtBjJ/b5sphM5wuPkDPSlafDbBJ0ScpDBgMXMGDRlExV3wOjuKPRtJQHdSS1qzv1PAQmdKzyXbvSmBAcIVrxyzYsUsrSPScfvtevDzdXHTh4UU3EqBkBnnvMdQyRdOdKWfthrlqs1tBx7X8z7t/39K9WJKPfYJWXarS/fJDRJxXGLgp4w5xm6YvrF6E4gMwtUdwDWqd3YLmPJkQjrAaJX2oBidz69Re8H7T3Kq3kEvO3YHjeCPbU8Y2zLDd3ZNSkj+VHSMt6H8f3ZxEYOZXz/5prcomJgk4Sx9+XJL7a3IGjEC2fL+IzxrMDgVBYZptj3zT4Ygbz5Eu/HzPHNhJlg2W9G2Um5+UMLAeRNBkxj0+9TCf1c0uUbfvuYKZ0kJFyk0oqHDw4DgB4bqtPvYjx7zK8qwuQFPo+U9oIPdJydx4u9UR6P/cxNwy3KtCKHz7/C/12YVXZFYubHgXYdpnrHB9XGlW07hn6KXx//1S6TM2phtEiIXP3Fa1e8lMlzFAp4ZRbG8gXhQ+NrFCDVsWi9fiCXfKHH8eUv+rPLmM4pRqpxBA+lOdmWdGLaA5d6kMzkMjnZ3w6Thbyx/4U9u9y7eB3tjGdmMpw0xm3oJAAMJPjwC1DO9uuuq6vsWPQLBZt5/wzxtBcWlKkR4jx8Ba5Vga6W3Z/NdsawH8L7iBbQC8R8kS1ac3ReutF4ALFrVpvl6Pi3Vk0YU8uWm5XSBgAJMmXnZADNA7eXZSZbNkTDzeodVq/QAOlk+J+N6Ikd6t9Nkjrk1QuW3NdGN+OLDP6LxZ0c8wNN73b6v/t+DJJP2Rr3kMzUQ3tq4Y0kd/gZc4AlWGxpMbsbC2dN+HO/mNLZ3jYDwgFp9vhSF4lZDNwDAAAAAANBCsrDKe7Vvb2XW3rR5q6uPF9zAGB2hJsqHAH3VxSc7bGXTYhZHBovy1WqaaEYYMEgBACFhBqXprVj5RqaRgzQLnXJuJ2NmlALVGRVkHF+WH3B2x5xB6oSEGeNB63aFAGcBiEgdBf4XWqb1DL9ptnjqYaabpx0mUGvWKXxsW4uH6eYMdffTXH28PPuGrzBYtAMU+pp+5qdlx8X58NcbpU4dSiN88GWJSU8HHY8uf0AAf6J4Fo6XJxx1P85/BAX0t6npXGAYbV/NJbTQVUMd6HNwjUhTlgAm1+KKNOmEJAEGxzBjhW0ArC2hlrnAkZb3ul86Wl0YJsH5GRKqDK8LixyU/oIJhtm5+9gOn8Mhbta4D4iVtllrGImZYrZOHWMTxl4dTzIZvuKzlIvS0o/6VLqBoQ0DKKqnY2SUVnWfVjtm0/jmE6YOUJFNMUrmOeOYKkwY7ACSgDEvDsXyw5MIu8SjaU5kdUn2RVPIRm3TQDn6UdBsdVP0dWWxdA3ys3gIJzz7BNmCT8bGR/Bq8BtUlpuHlJLYlOVCUNBHDtVH/zu62SL0yVbHdQQMACoFzDmYdTNVRWVV3Dj6mddqp1WQeXYVyj0BlQ2hBVFwqB/ZK1l+Tos46fkPJnSR4vGEkgMVq+ZWQFyvaUU9vDGI2rl5xKfWl1KH2+LwBD6+LjfIyTGXCIu+7w5vKz4SLLRB6yeF+KXwdOPbWbc8/Gd7+VQxUcHmVv+W7W8Bs3GmkcGzUtKCafqybh+8Ud/zcfGZ62mmX+/D2J/Cfr+MYuvpDkOR6EgA7OzbS2FwLQBnCAM7G0wlnvz5sH3u02C1J2jEgUx1Q0NMaLdP2NjL7G6+a8H7X0D3dy2NcFz6iuD5/4Mz2n1Ndq6Hl8GlTBXJ+ktn8Gpfe8G9BfU6Wsbz5Q0/4z3+JtRL4VKbE+rf9VOjdGhK0n4Jg9h8dTxDa1evgfKSCy8cymfuYHfBkWEugJv5NHW67Wvc+ssq5+3Qnq/FMvv93KWB58a8OwfzhT7Dkw+Oi63RPNw/zBw9A/xmJcWp1FuVQQHXO5Yn8HAh7JEmJpFeBz8p4G3XPfa8tOo3ZWZWMsKrBL9KTaNUKaq9wFFa1S1VfxkGL7hwaNFqTK9gwnJc9QnEWjqyV+tQ8W7Uk6+ksKK9MqD6rEl1TvYZeG3yAX67lJ1RuNCbFIxQvzPG1n09Wc3sfLqUb4nj16qnqCPRhmGkjfl8Bl5PQ8iMdT7b2k+/Sfx51ymjZb1nBqEyB3D7n52OSw1N5EH5yQrqpoUGc6ywvHu1+5ee3mu5AAUyzjXHm7dSWsqhp4vBqjsXGXjg+/b0NJGKfySS+JTlKqr3bIHpzm7XxHk7nR5/3UvDe0gBH/5iM6jVDFd2nAGM1+/XHqJHUIbeJFZ4cVf09aZJjzq5RCtmkJqBO4sQDMdr8djOaO02HvHO9L2fO4/8fBg4K1AMAMUUo6B5KhNpxWMEA82jnsG8HmGCvZGFUi6oIT4d3Gfz1dUhgO7y1BMn1+l4OGSF3+vO5Nt3KAp3dVnzyjXzBvFSGoQmmBbAvAlPX/VG6+tIYGxUDOkG3Q8hTovYqMOSfcL8WBFFJg9rPTPRY8RTKqg5ePYxijc+1oXMPgmJZj+RWpmjAXFSAv0xqUcR8YnMmo9pRr3jZ6W1uXNmXEvqH9wZzhQ9BFYnp7PE0vUUdf+bqyg1MCy1hT/F9rDLvFVdyRvkR6aMTqqX3RVrT2OHqfn9J5g2wK4U+t3OVjbmUqZV3P4dTeKZ5uvT6BqlKTg8Z+kAqwPw+c+CLJ1CBKRyWpCN8tCuqJ0XjF05lv4kdfNKwPX/uMsaTq5gvX7VpAAAAAAElFTkSuQmCC

```
Ttu3ao/SOFOV7bZR7APAzb3y7SO6Pz+wYtRIFHwAAAADUAi6XW18sXia7zS5HWKi2lESaHQkAapXDRS69v3qf2
TFqJQo+AAAAAKgFVq3bpG279iq9UYp8hrStLNzsSABQ6/xjOU55fX6zY9Q6FHwAAAAAYDKny6Uvvlmmk JAQhYW
GandZuFx+m9mxAKDWOZBfprnrD5odo9ah4AMAAAAAk63ZsFk7du9TelqyJGlLaYTJiQCg9npnJY/p/hIFHwAAA
ACYyDAMfbtqrWw2m8JCQ1XgtemQO8zsWABQa63claddROrMjlGrUPABAAAAgIl27T2grTt3K6VhoiSxuQYAnIJ
3V+O1OOKtQsEHAAAAACbK3PC9iktKFROVyeYaAHCKPsw8wGYbPOPBBwAAAAAmKS4p1bLM9YqPi5XFYtFupONON
tcAgF91pNilBVtyzY5Ra1DwAQAAAIBJ1m/equwjR5TUIEGSlMXjuQBwyt5bxWYbP6LgAwAAAAATGIahpavXyWa
zKcRuV4HXpoPuULNjAUCdsWjrYWUXOM2OUStQ8AEAAADASbz88stq1qyZHA6HevXqpZUrV1bJdXftPaAtO3Yrp
WEDST9urmGpkmsDQH3g8xv6eOOBs2PUChR8AAAAAHAC7733nu6880795S9/OZo1a9S5c2cNHTpUublnvu7T8cO
1ShTD5hoA8JvNWXvQ7Ai1AgUfAAAAAJzAM888oxtvvFE33HCD2rdvr1dffVURF68803z+i6v9xcYw+bawDAb
7L5UKG25hSZHcNOFHwAAAAAUAm3263MzExlZGQEjlmtVmVndZGjZsmVndO1fbq6xtTTijK4HAPXZnO94TJeCDwA
AAAAqceTIEfl8PiUnJ5c7npycrpycrOzs7N98XcMwtDRzvWzW45truPwWHXSFnWlcAKi3Pl57UIZhmB3DVBR8AAAAA
FCDdu87qC3bdymlYaIkaa/TIT+bawDAb3Ygv0yr9xwzO4apKPgAAAAAoBINGjSQzWZTTk5OueM5OTlKSUn5zdf
N3PC9ikpKFBMdJUna43ScUU4AAI/pUvABAAAAQCVCQOPVvXt3LViwIHDM7/drwYIF6tOnz2+6Zklp2fHNNWJjZ
LFY5DWk/TyeCwBn7LON2fL76+9juhR8AAAAAHACd955p15//XXNmDFDzdv1s0336ySkhLdcMMNv+l6m7buUPb
hIOpucPzx3AOuMHkNviODgDOVV+LWhgMFZscwjd3sAAAAAABQW11xxRU6fPiwpwkyZouzsbHXpOkXz5s2rsPHGq
dqUtUOGYSgk5Pi3YnvKwqsyLgDUa99sO6zO6XFmxzAFPyoCAAAAgJOYNGmS9uzZI5fLpRUrVqhXr16/6Tql
t35yl+LhYSZLfkPbyeC4AVJnF246YHcEOFHwAAAAAUAO27dqrI3n5Svyh4Mt1h8rpt5mcCgCCx3d7j6nY5TU7h
iko+AAAAACgBmzetkNev0+hoSGS2FwDAKqax2do2Y6jZscwBQUfAAAAAFQzt9uj7zZmKTY6OnCMgg8Aqt432w6
bHcEUFHwAAAAAUM127t2v7CNHlRh//PHcMp9VRzwhJqcCgODzTT1dh4+CDwAAAACqWdaO3XK53Qp3OCT9OHvPY
m4oAAhCu46UaF9eqdkxahwFHwAAAABUI7/fr+82bFgGI/nAkD1qY+z+Cj4AAAAAKAaHco9ooPZh5Xww+6
5hiEdoOADgGpTH9fho+ADAAAAgGq0c89+FZWWKiYqUpJOxBMip99mcioACF5LdxyVz2+YHaNGUfABAAAAQDXas
mOXbBaLLJbja+7luENNTgQAwa2gzKN1+/PNjlGjKPgAAAAAoJqUO9OUT8POmt/vUnCSb7zfQHqH4uGj4AMAAAACArJjz355PF6FhR5R/LNftt6jQazc5FQAEv7X78lXs8
podo8ZQ8AEAAABANTAMQ+s3b1V4uCNw7KgnRIYsPBasJqYCgPrB6zf0/cFCs2PUGAo+AAAAAKgGRcUl0OpRzWDFRUYF
jPJ4LADVnS3b9KfiYGw4AAAAA1eBQ7hEVlZSqaePUwLEjdXwHXb+rVPnfvK3SbcvkLy1QaNJZis+YqLDU1pKOz
1os+Hamitd9Lr+rRGGN2ilhyCOKSWhOwmvuf+V38hXmVjge1XWkEofcHPil68BmHVv8b7kPZUkWqOKTzlLS5Q/
LGhImw+vROXkvqHTbctki45Uw5BaFN+sSeG3Big/lKzyshAv+UHWfDAC13uZDFHwAAAAAgDNwKPdIufX3pLq/g
+7ReS/Kc3iPGlx4l2xRCSrZ9JVy3n1Aab//u+zRDVS44kMVZs5Vg5F3yB6brPxv/xPrPxv3lbu+1OU9vtXZLFXXm6mXve
s5PcHfuO+ske57z2gyLZ9A8dcBzYr5/2/KLbPZUrIuEkWqO3u3F2yWI4/lFaObp7c2duVcvVTKtuZqSNzn1TjS
W/LYrHIk5+t4nWfK/W656r1cwOg9tl8qMjsCDWGR3QBAAAAoBrsP5Qjq5/WWn9fZcfosKfXXV3joXf41Jp1hLFnX+
DHOkdFBKfprh+4xUSn6qi7z6TYRgqWv2xYvtcoYhWvRWa1FwNLrxT3uI8lW5ddsLr2iJiZYuKD3yUbV8pe1yqw
tI7BsbkLfinYrqPUmzvyxTasKlCEhsrsl1/WezHC1PPOXOKb9lLoQ2bKrrbSPlLC+Qv0z5zJ++Lvyt+4PWyhkV
U7ycIQK2zNadIhmGYHaNGUPABAAAAQBUzDEPbdu1u1RRER44Fhdn7OnvO8y/LLYyr8Piz1Mrv2b5C3Ika/kWLlHY
61hkQpLayPXwS2ndAvD51HJ918rqtMFsli0l6O+kny5D2XJGhmn7H//SftevFrZs+6Tc/+mwOtCk5rLtf97+TO
uOXetkSOqQdbwGBVv+koWe6giWp975u8fQJ1T6vZpz9FSs2PUCAo+AAAAAKhi+YVF7ndi5o1driOr79nD
YtQWFpbFSx9V96iozL8PhVv+kquglvkKzkmX/Gx4+Mi48q9zhYRJ19J/indo3TrcvmdxYrsMDhwzJufLUkq+Ha
WojoPVfLLf1VocgvlvPt/8uQdkCRFW1Sw7HO1uPhe+Z3FKvh2phIbtKxxf/WgX/cqJz3HpS36
MiZfzIA1Bn1ZaONujs/HAAAAABqqR832GiQGB84Vudn8ElKvPAuHf3seR34+Y34+3XHN7pIaaHIdgPkyt5eJdcvXv+
Fws/qLnt0YuDYj4/XRXUZpqhOF0iSEpJbyLlnnYo3zFF3zFf8edfLYrOX25BDko789zlFd8ldd85Ol1W1bptQbXlThi
g917MvX1PCSP1dJXgC13+ZDRRrWIfXXB9ZxzOADAAAAgCp2KOewvD+Kv0WCoeALiU9VylWPyu/PK/2O/6jRLdO
Veu2zMvw+hcSlyBZ1vMzO/2K2nq80X7ZfzOqrjLcgV8496xTVeWi54z9eN6RBk/JZEtPlLbLc3SPo
rtdKOfe9Qo/q4esoQ5FtOOn594Np/huAQSD+jKDj4IPAAAAAKrYvoM5+2mCjzGdVcR3eYOOXrKEO2aMS5HM
Wq2zXGoW36i17bLJskfFy7lkbGOd3lcp1MEthaW1/9ZrFG+bLFhGr8BY9yx23xybLFpUg79H95Y578g7IHpNU4
TqG1628+a8ocegkWaw2yfDL8Pt+COSTYfgrvAZA8NqSXT92Og2e/8MAAAAAQC1gGIZ27NmryS1t8Lhtl7klS
2M1OSZE9oJO+xQzr29ZsKSWisqI4Zslgsiu5xsQqWvid7fCPZ45KV/83bskclKKJ1n8A1ct79s8Jb9VFM91GBY
4bhV/GGLxXZYfDxUu5nLBaLYs4Zq/xvZyokqF8utV9To+ytkzF/6rsLP6qHQ5BaSpLBG7XXs6zc
V1TFDRWs+laNRu+r41ACopfbmlarE5VVkWHBXYMH97gAAAACghuXlF6jM6VRMVFTgWH2d+CSl+gL/pIXr6io34q+PKCpODzu0qVv3iGvEVHZ
HNEK6LNuYobcKOstuPfWsbOGivD49TRz1+U31kiR+P2Srr8YVnsP2Ow4jmWrbCy8o/MOXevla/wcGCNvV+K6Xm
xDJ9bxxb+U35nkUIbNlfSFY8oJL78ulruw7tVuuUbpV7/YuBYRNu+cu7boOyZ9yoksZEajLq7qj4dAOoAwzg+i
```

```
6970/hfH1yHWYwfVywNMoWFhYqNjVVBQYFiYmJq/P6zsw7V+DOBBJ8xbYJ/MVgAqG/M/joV1W9j1nY9/vKbatE
sXSH248XXkvxYbS6NNDkZANRPj17SQeN7NTU7RrViDT4AAAAAqEKHco/I5/cHyj1JKvHZTvIKAEB12nIo+Nfho
+ADAAAAgCq078Ah2azlv9Wi4AMA89SHnXQp+AAAAACgivj9fm3bvU9RkRHljhf7+dYLAMyyPbfY7AjVrsr/L+P
z+fTggw+qefPmCg8PV4sWLfTII4/o50v9GYahKVOmKDU1VeHh4crIyNC2bdvKXScvL0/jjx49XTEyM4uLiNGHCB
BUXB/9vCAAAAIC6q6CoWPmFRYqMCA8c8xqSy88MPgAwS36ZRx6f3+wY1arKC74nnnhCzil566nSVt3rxZTzz
xhKZNm6YXX/xpF6Np06bphRde0KuvvvqoVK1YoMjJSQ4cOpDIwZP368Nm3apPnz5+vTTz/V4sWLNXHixKq0C
wAAAABVprCoWC6XW2GhP+0ay+O5AGAuw5CO0FrvNjlGt7L8+5PQsXbpUF198sUa0Ch2aOlJatasmd555555x2tXL1S0vH
Ze88995weeOABXXzxxxZKkt956S8nJyZozZZs3a968eVq1apV690ghSXrxxRc1YsQIPfXUU0pLS6vq2
AAAAABwxgqLS+R2U/ABQG1zpNillFiH2TGqTZXP4Dv33HbtO1YMECbd26VZK0ObtO6ffvttxo+fLgkadeuXcr0zlZ
GRkbgNbGxserVq5eWLVsmSVq2bJni4uIC5Z4kZWRkyGq1asWKFSd836ZGjli4uIC5Z4kZWRkyGq1asWKFZXe1+VyqbCwsNwHAAAAANSkwqJi+Q1DNttP3
2oVU/ABgOkOF7vMjlCtqnwG33333afCwkK1bdtWNptNPp9Pjz76qMaPHy9Jys7OliQlJyeXe11ycnLgXHZ2tpK
SksoHtduVkJAQGPNLU6d01V//+teqfjsAAAAAcMoKi0tksVjKHWMGHwCYL9gf0a3yGXzvv/++/++Zs6cqVmzZmnNm
jWaMWOnnrqKc2YMaOqb1XO/fffr4KCgsDHvvn37qvV+AAAAAPBLBYVF5TYYlCj4AJ4AKA2OMIMvtNz991367777777tO
4ceMkSR07dtSePXs0depUXXfddUpJSZEk5eTkKDU1NfC6nJwcdenSRRZKUkpKi3Nzcctf1er3Ky8sLvP6XHA6HHI7gmV0JpjFBCwvrC
BYWVtVvBwAAAAB02eG8fIWEhJQ7RsEHAOY7UhTcz1FYg0Vz+Az+ArLS2V1Vr+sjabTX7/8e2ImzdvrpSUFC1YsCBwvrC
wUCtWrFCfPn0kSX369FF+fr4yMzMDYxuXCi/369evXpVdWQAAAAAqBJH8o6V22BDYg0+AKgNmMF3mkaNGqVHH
31UTZo00dlnn63Vu7vvvvvtOzzzjH73vvvt2GjZsmG6/e79qRFHwAAAAAcIZ8Pp/8qYl8ogehADAAAAgDPk9/tlGH5m8Kk1/oMYM
P9dkuf5KGl0zRk7peZZZZws+MAUjgz+AAAAAAAJxAWGiqb1Sqvz1fuOAUfIL3sHKILXNOOztbR7Cio75jBBwAAA
AA4kbAfZvD5fjGDL4xHdAFJ0n5/oi4uuV+PaKJKLFFmx0F9FBYj2dhFFwAAAABwAqEhIbLZrPrPL5y8y8/gc1AwAeW
84RyoQc4ntcrazewoqG+CePaeRMEHAAAAAGfMarUqIjxMHq+v3HFm8AG1u9Y/+tFDY7qFxFG8G2xIFHwAAAAAUCUiI8Ll9VLw
j4AAAAAwK85XvD5Kh5nnFh9QqVmuc3We8yl9a+tFDY7qFxG8G2xIFHwAAAAAUCUiHBR8wOnKM6JOdckfdYf/TuV
b48yOg2DGDD4AAAAAwK+JioyQx0fBBwC1hiVlzmOwU0+gIBuIoyQx0vBB/wWc9w91L/sKS2w9ZNfFrPjIBiFBiFM4MPAAAAAPAr4mnNj5K1kBl+4eIxA
BUVGRGaUHKLbvXfozxrotlxEgyE2ygwYwQcAAAAA+DUREeGGSUXElMWbwAafnM3dn9St9Sp9Zz5ef2gJVJbKB2QmqFf+
lAAAAAEAViAx3VHo8ihl8wGkrVZhuLr1Rv/P9WbnWZLPjIBjENzc7QbWi4AMAAACAKhBxgoIvzu6t4SRA8Pja0
179S5/QR9Yh8slmdhzUZYktzE5QrSj4AAAAAKAKhP9Q8Pn95R/R/JpeADzoxLobqj9Hpd7Z2iQ9YOs+OgLrLapbi
mZqeoVhR8AAAAAFAFIsPDFWK3y/uLnXS jbD7ZLKzDB5ypZd5W6l/6hN61Xiiv7GbHQV0S11SyBfefGQo+AAAAA
KgCEeEOhYTY5fZ6yh23WKRY1uEDqoRXNt1XepUu9z6sfdZ0s+Ogrgjyx3MlCj6gUj6fT+88P003D+6lKzufpVs
u6KMP/v6sjJ/tilZWUqLXH/6BZjyvu67sfJb+OPI8ff/OW7967U9Y/TXlZ3P0sSB3fWvqX+R2+UMnJ/3z
gzdcdFgXd29ta7u3lr3XzFKaxYvLHHeNf019SNf1aq+JA7tr8dzZ5c4vnTtbU4XFDEJqf
TXfEcj+kCVWqNt5nOK31MOy1j5FGI2XFOS5HLOOR5TjV9rkjhjxbq3DdKtOrAiX8IMHuzRxf8uOQNnyxSzNRC9
XmjRJ9vP/HfKY9/65Llr4WaPM9Z7vidnzuV8ESh0p8t0sz15X8Q8cEmj0a9U3pmb6w2S2xpdoJqF9zzE4HfaM7
rL+vzd2botsefV3rLNtqxcZ1e+vMdioiK1shQshrfy9Jmv74Q9q4Yon+OO1FJTVK9/l/crIS1lZPQcNrfS63
8ydrbeffky3Pvq02nTtqYO7d+il+++3QZNEN9z8kSUpMTtXd/1ZqU2bS4ahr+Z8oCduvUFPzv5CTVq10aqFX+j
```

b/36kB//5jg7t2am//99d6tLvPMXEJ6qkqFCznn1Cf/nXuzX0mQIAAMCPoiIjFBcTrcN5xyqcYx0+oOr5ZdNDZ
ZfqA1tvvRj2is7y7zI70in5/dwybcz169+XhCst2qq317uV8e8SfX9LlBrFVJyHtXiPTxecZddjg+yKc0j/Wnu
8jFvx+Oh1TS2/8ciqAz79I90tTsnlrzM3y6NZGzz64ppIbTvq1+8+KdPQljY1iLCqwGno/xa69OW1EdX6vk2Vc
JbZCaodM/iASmR9t1o9Bw9V94EZSmqcrj7DLlTnvudp+4a1P41Zu1oDR1+mDr3OVVLjdA254mo1a9Ne29avPeF
1t3y3Wm279VT/UWOU1DhdXfoNVL+Ro7V9w3eBMTOHDVH38wYrrdlZSmveQuPvuE+OiEhtXZcpSTqwc5vOPqePW
nbsrP4XXqLwqCjl7t8nSfr3k3/TOCuvVcOOxtXyeQEAAMCJWSwWNU5NVmmZs8K5uBBPJa8AUBU2+RprUOmjetV
yhdwKNTvOSZV5DH34vVfTMsIOoKldLROsemigQyOTrHpldcXZv5LO3DCH7ukbpp6NbGqVaNNjgx1qlWjV3K3lf
3BQ7DYOfnaZXh8VrniHpdy5zUf8GtjMph5pNl3ZMUQxYRbtOnb8CbV75jt1c48QNYkN4oqoHszgC+LfPeC3a9O
1hzYs+1YHd+2QJO3esklb1qxU1wGDfhrTpYdWLfxCR3MOyTAMbVi+RAd371Tnvued8Lptu/bQjk3rtW398UIve
98erVm8QNOGDK50vM/n07f/nSNnaanadOkhSWra5mzt2LhexQX52rFxvdxOp1KaNNPmzBXa+fOGjbhmQlV9GgA
AAHCaUpMbyuer+KgdM/iA6vd42cUa6XlcWbbWZkc5Ia9f8hmSw16+gAu3W/Tt3lNbq9NvGCpyGUoIL3+NW//n1
MhWdmWcVfFhzc7JNq0+6NOxMkOZB30q8xhqmWDVt3u9WpPt0+29ancxesbqwRp8PKILVOKSiZNUWlKk20cMkNV
mk9/n01WT79OAUWMCY37/4N/O6oP3aOJ53WWz22WxWHXzI0/q7J69T3jd/qPGqPBYnh4YP1qGYcjn9WrIuGs19
g+3lxu3J2uz/nzlKLldLjkiInXPS28oveXx/0l17T9QA0aN0b2XjVBomEO3Pf68wsIj9NpD92vS1Of0+Tsz9Nn
bbyo6PkF/ePhJNWnVpno+SQAAAKigYUKcJMkwDFksP33zfXwNPkOSpdLXAaga23wpGlrykCY7PtMfLB/IYVScU
Wum6DCL+jS26ZHFLrVraFVypEXvbPRo2X6fWiac2hysp5a6Vew2dPnZP1U67270aMOhn1bdGFnpa4a2tOvqTiH
q+XqxwkMsmjE6XJGh0s3/dWr6xeF6ZbVHL650q0GERa9d6NDZSbZKr1Mn2R1STPA/5UbBB1Ri6Wef6Ju5szX5q
ZeV3rKNdm3ZpH899hfFJyXr/EsulyT9799vauu6TN339+lq2Kixvl+1XK8//GfFJyWr87kDKr3uxhVLNfu1F3X
jlMfUqlM3Ze/drTcfe1Af/P1ZXXbLHYFxac1b6KmP5qu0qEjLPv9UL933Rz3879mBku+K2/6kK277U2D8+y89r
U7n9pctxK4PX31ez3yyUJlfzdeL996uJ2d/Xo2fKQAAAPxcYnyc7Ha7PB6vQkN/WvjfbpGibD4V+/gWDKgJzzm
H6yNrT70c/qo6+L430045/74kXL/7pEyNnimWzSJ1S7Xqyg4hyjz06zP4Zm3w6K+LXPp4XISSIo8XgvsK/PrjP
KfmXxNRYWbgzz000KGHBjoCv/7r1y5lNLcrxCb9bbFLG26O1Kdbvbp2TpkyJ0ad+RutLeKbS9bgf4CV/7sAlXj
ryUd0yY2T1G/kaElS0zbtd0Tgfs1+7UWdf8nlcjnLNOu5x3XPi2+o+8AMSVKzNu21e8smffLmqycs+N59YZoGX
DRWGZeND1zXWVaqV6fcrbF/+KOsP/ylExIaenyTDUkt0nTS9o1r9d+3/qk/PDytwjX379ymRXNn66nZX2jhh++
oXY/eik1I1LnDL9L9LL/3enyoqLFR4VRH85AwAA1GKJ8XGKCHeo1OksV/BJxx/TpeADas4efwNdWPKAJjoWaLL1H
UUYtWOX2BYJVi26PlIlbkO0FLk0p0VZd8Z9SnRV/8hLq3Y0e/f6TMn1wWXi5x3AzD/mUW2Ko2z9KAsd8xvHNOV5
a6ZbrgWjZrOWLvy1HfHp7g0ff3RSpN79za0BTmxpGWnX52SH63SdOFbkMRYcFyYzjevB4rkTBB1TKVeaU5RcNv
9Vqk+E/vgipz+uV1+M5wRj/Sa5bFijxfnrN8V8bhnHC1xl+Qx53xQVXDcPQP6bcq+vv/YvCIyPl9/vl83p+yHj
8n37/qa3jAAAAgDMXHxutyIhwlZU5FRcTXe5crN2r/S6TggH12GvOwZpr7aaXw19TN986s+MERIZaFBlqObEyQ
59v92raBY4Tjn1ng0e/+6RM744N18jW5X94MLi5XRtuLv9o7g0fl61tA5vu7RtaodwzDEM3ferUMOPCFBVqkc8
veX74NvbHf/pO/O1p3dOwfixbRcEHVKLH+Rfow1dfUMPURscf0d28UXOn/ODDxo6TJEVERevsnn30O1pOPKDTMo
YaNGmvTymVa9PF/dN19fwlc54V7b1dCUoquvuvPgev0nf6amfroFaduyl7zy69+8KT6nH+BbLZjq9x8PbTj6n
rgEFqmNpIZSXF+ubTECsxSQkquegIZKktt166v2XntbWtZlas3ihGrdsrci+dAEAA
OAHNptNackNtTFre4VzbLQBmOeQP15jSu7VtY5vdI/lbUUZRaZl+Xy7V4akNolWbc/z6+75TrVtYNMNXY4Xd/d
/6dSBIkNvXRIu6fhjudfNKdPzFzjUu7FNOSXH19mLuTkE+EbAsmusAvc5dk1A2LxQzqnmPeugDRN4TjVELt87
75N7HpokUvL93v12Tav2je0Ks4RJLP3JKlRD7MT1AgKPqASv3/gb3rnhWl67eH7VXj0qOKTknFXF6aBYA+AjE
e0cxnHtPzd09ScUG+GqQ10pWT79XQcdcGxhw5eEAWy08z9i69ebIsFoveeX6a8nKyFZ0QoB7nX6CrJt8XGFOQd
0Qv3nu7jh3OVUR0tJq2aacH/zmrwu68+Uc068NXn9djk73wSONaqU1eNuuEmPXrTtYpNTNRjz9fHZ8eAAAAnET
j1GRlrq+45lcsBR9gurec/fU/S2e9FPP9fatNiVDgcvQ/Quc2l94fCfcse3senSQyG246XaoWJDewt+ejLst
Uy3vP7ju+Te+r+frnNd5xBNHx1+WvfOKfbr0W9cWjrhpxl/5zSy6a4+YRo5q0xJkcc34Agq6eeYnaBGWIyTPRd
YhxUWFio2NlYFBQWKiYmp8fvPzjpU4/cmnUDsg1XcpN3JqmWl9Zc0CG9tkdqXPPdlbl/dKeHyY3v8f5anRrd6MmjBc6c19DCZH+5anG1qHMeO8Ps0u1jqNyT5fDzEoQuXqI7xvOjoxCKxgkKbrh2xrpGIX0NhIAs1tKiUBhuNKbFRq+mqz2bvuWGBWQuUxhHP10hMMgT1FQvQ1f7i3vEMIiRr0qKsY/pmZ
L90edgKPWCdrhijw0woqC7xzaU/rjU7RY0I/m1EAAAAAKAGJcbHyWKxy0crvzZzuM2vGBuz+IDa4n1XLw1wPqV
F1j4KyplPkBr3NDtBjamWgu/AgQ06+uqrlZiYqPDwcHXs2FGrV/8dUwDE2ZMkWpqakKDw9XRkaGtm3bVu4ae
Xl5Gj9+vGJiYhQXF6cbbrhBxcXFNfE4AAAAQJVJjI9VuMOhMqhezwrnUMHbZAGqTfCNS15Xeptv9f9Ixa4LZcVD
V6snjuVI1FHzHjh1T3759FRIS+88ff/f+76776Sj/G+NyffGTJ2TK2TS+88IRsf/4PJCmzsArGRlcpFbG5Q7r8AG
12gUhGzU15HU18B82OwrOREiEdN8+yVY/9pet8hl8n3zyiXr06KHLLrtMSUlJ6t27t1599dXA9zt37lROTo4GDBgQ+C42NlaDBw/W0qVLJUlLly5Vv/VrWmq9h7n+gHa/ceAv/FmVJKTJq0zPVbJ7bx9HtVEbwAAAAAAAAAACJD6K+NdixyIIUlWX1xZkuIAtcmBF3lszOO0UO/pJmaH3o0fHOA84PxrGh6kz1tmnu4N9BevW/QpakZ+G4O1VFwixyqr5VLXuJ9Qnf+JKx4V5mdJmlsvxv8ObVPdiq8W9fz1jH5fmn6+0ArCqBsbfzqp+Gs0W4cPZymr8ZB57zpQzFxEwKbZAZxyqfv5HpSvWnpAJnWqNGyTsqPgKJ3iyHR+0fU6BVu4wnB0tK54JCYVJKfX/bHahszdB8YGFpQ1EYJdRLsGkZUrdI3PtHGV8zNQpTMQDAcj9Rmq22u/3hRqHnqe1HXCdM+nstNpWxIMyXcxEZljmxmHxHtV2lpXNvxbKGyGSXWmRFtkwbcWq1rdJQSR75dwrSH2xZu7Hbv/hOgKSQF+cl8nvIXsmV10p7f7+/QSaKh/b8gIMSRgZ7zmHMDoW0BkFY5bhFhplMx4cDtQRQjLT9GjPrdnqQ2bMw7KWaELjF+a4VNYeXVlzx5OFgAtc0ETHRaTkoN+VhB1eRVbpNhSd+wqFnfvStdQ4H7ZOmAOWzwH88UqTjwuJQEXtXaHnoC38Gs7Yb8hCQEkUL3LXCs9vE6uNyQvIgrSu1XjwTHfvj1FAP1wUR7mFIcX5/J1h1SvKcLZYl2y5hxkKNObhH9c1T0GUtwvVPCNUr3Ko0YnGcIxxpCRzZV9sf0hSeTeSbJOsYIm+pEt5RTYLuBXv6h7eqQJnTc5Si2Ls6dVU2QgNfsLH8Tbd2gC3BBQKygUcqP3DMR0Zpm1oQFG8Xo8BNPK5dSr0SB53yg7XpTXUOjIGLNDuH+tkzDvgKydxRW8ST1GmFoH8fVJupkrC3e+qwDpRkSk9Eb1fEKEWaztWBhUtfTaOSWj9OB1Z2Z2bBB9oJ57aPZ0uxB0XRr4BMlXDHCNcNLkwh9SHMxqyKmsNu3xWbC8uvBBTF6AKtoO81NZV4rZO8v5AYnhZMIUsLXz6nYqvTGn8/3bdTYSSvTl9b8Ga3gzYh6VKSxOzuYNMLZSbdgaa1hWhvf2bhWkUaVkhAaF0t9ri+zDezl/YXWZldhpMLWiptmqUkGtnvQuidQ95V7cHp9xGrm1dH4Xbd6FRPBGptvb3y1edDGnd5PBkJwUQ8KiGpOPs6DXdU8QBVHmEjTaJEpNVNGDEaU5SuL35eFmnt6cHIGGvUOyX3mNT3BljEsBo0Hm4fhWrm3NmuI6aSwN5G0p3D3Vm1bhaDuS4M3NSDozyWm8XgdzaCk1yYIy0fjhgXXfMb8UH9pd/s9o8uTJTHRh0pCK0sqb9kkH57c4OHNvbxaEZZNgP2W8mc1H6Q8ZuHbm8FnA3RwXXZJBvr4ez5Zf5R+xEaRb4JKpfE4Kq7MOBrjs0nY1NLfG9FevMX8NVHzBQQjHbSZFAVYVo/dSNFNvz1gqdwG3aXF3NzXDDhcqlQmp2Ej5B1UuQUD5OA5tqMlZaHrUNd2E5A4dTsdX1JhbiXB9qtsi6i0NhvKKJzjbfYvtAWEmS4qkIpyMtNhzVqydD/cuVq9fmnLQcmXeGdJKiOH27o2KqXJAgY6t/xnbgjfyjh4kNcTfaSvbUzDgjrGQjzkw0KQbD9XmIiEBTkFJBfqtqRrQ5P4DlpyaW3jyhmWG4XJ9Thk1uSTIdTq9HvJ9qgU5D8XzI5QiWlG9LpvJ4iDSh6bP6EmX2PUBwDjJnVOHPLkrWgvcdDZDZjeTfoGsVJbQ4k3aW2xVK6qI9kgIvtt0pOnTYfw5vBBWWeJKjLf5nBcnqjszBsdI4TxqMx0LVsnD+7aq3gAAAABJRU5ErkJggg==

RmBY7GxserVq5eWLVsmSVq2bJni4uIC5Z4kZWRkyGq1asWKFVUdGQAAAACqVHpaisrKKs7gi2QdPqDWm+/poH6
lT2qudbD8bF1Qd6V1rTflnlQNBd/OnTv1yiuvqFWrVvr8889188036/bbb9eMGTMkSdnZ2ZKk5OTkcq9LTk4On
MvOzlZSUlK583a7XQkJCYExv+RyuVRYWFjuAwAAAADMOLxJI/n9flX2wBTr8AG1n1Ohuq1Ogq73PaAcK7Nu66T
GPX59TBCp8oLP7/erW7dueuyxx9S1a1dNnDhRN954o1599dWqvlU5U6dOVWxsbOAjPT29Wu8HAAAAACfSJC1F4
Q4H6/ABddxiT1v1K52mD6zD5ZPN7Dg4HY3rzwYbUjUUfKmpqWrfvn25Y+3atdPevXslSSkpx5vvnJyccmNycnIcnI
C51JSUpSbm1vuvNfrVV5eXmDML91///OqKCgIfOzbt69K3g8AAAAAnK7GqcmKj41RQWFxhXPM4APqFo/survOG
o3zPqQD1kZmx8Gpqkc76ErVUPD17dtXWVlZ5Y5t3bpVTZs21Q1b95cKSkpWrBgQeB8YWGhVqxYoT59+kiS+vT
po/z8fGVmZgbGLFy4UH6/X7169ar0vmFhYYqJiSn3AQAAAABmCA0NUduWzVRYVLHgYx0+oG5a5W2h/qWP623Lx
fKq/qztVifFNZGikn59XBCp8oLvjjvu0PLly/XYY49p+/btmjVrll577TVNmjRJkmSxWHT77bfrb3/7mz755BNN
t2LBB1157rdLS0jR69GhJx2f8DRs2TDfeeKNWrlypJUuaNKkSRo3btwp7aALAAAAAGZr2ayJfD5fpevwpTCLD
6iT/LLpgbIrNNbziPZYm5odByfSfIDZCWpclRd8PPXv21EcffaQHH3xEkydPVrp06fTss8/q6quvDox55513
tt2nixxInq2bOniouLNW/ePDkcjsCYmTNn6NmUdPiBIrfM11cSR/VPy6VyKO9TsOPillheYnaDGWjihfv366bXXXqvquAAAAAABQLZo2SmUdPiBIrfM11
cDSR/VPy6VyK9TsOPillheYnaDGWYzKfpwUBAoLCxUbG6uCggJTHtednXWoxu8JIPiMaZNqdgQAAQBUz++tU1By
326P7H39BZS6XOpIbljtX4rPqnRx25gSCQVvbQb3keFUtfdvNjgJJstqle3ZKjlizk9SoKp/BBwAAAABgHT6gv
tjiS1NGycN6yXKVXJYws+Og8Tn1rtyTKPgAAAAAoNqwDh9QfzxVdqGGuZ7QZ7QZltbs6PUb6OyzE5gCgo+AAAAAKg
mrMMH1C+7/EkaXjjJFT+l6lVnCzY5TP1Xh+nuLFy/WqFGjlJaWJovFojlz5lTZ5lr166VJO3VnZhuLr1Rv/f9WbnWJLP
j1B1xTapt99yBAwfKMIwKH9OnT6+W+50JCj4AAAAAqAGtmzdVWlJDDHT6aV+Gc1SI1YxYfAEkLPe3Vv3Sa5liHy
Edt8+s6XyVZWMOQPykAAAAAUAPCwkLVo3N7FRYVV3qex3QB/Mil1UE0uvV7XeqfokDXV7Di1mEXqcpXZIWoFCj4
AAAAAqCEd27ZSWGioSsqbdqbqqRGupWhNVnQioAtdUSb2v1L52md6OXyiu72XFqn+b9pfimZqeoFSj4AAAAAKCGt
GyWrsapyZU+pmuxSM2ZxQfgF7yy6b7Sq3S592Hts6abHad26XK12QlqDQo+AAAAAKghdrtNvbt1VLHTKd8v1udDAo
8j+kCOJE13mY6r/QxTbdcIo9CzI5jvrAYqf1FZqeoNSj4AAAAAKAGdWjbSg0TGyg3L9/sKAAkqqk7ajGHAUfk5Z5SaEexds9JqQCUBf4Z
dNDZZdptOdR7bI1NzuOuc6+RAoJNztFrUHBBwAAAAA1qFnjNDVpnJrK7Pm5RIIZHaVWHZkSQOnAlDXbPI11vklj+pVyxV
yK9TsOOboyu05POfBBwAAAAA1yGaz6ZzOHVRS6qzOMd2WEaUKs/hNSAagrnm87GKN9DyurbZWQ1aS+nnm
J2iVqHgAwAAAIAa1qFNCOVHRaiwuOJsPbtFasMsPgCnaJsvRUNK/qrXrndI2cFofZcWpG12vMTl1DrUPABAAAAQA1
LTOvRWUOa6/CRyh/TbR9Z9Iosqzu4DgBN5zjlcQ13TtNHW2VuwuolSssRup+ndkpahOKPgAAAACoYRaLRd07tpPf
ZbX56twPsrmVzOHO4RkAOqyPf4GurDkAU3VBJVaIsyOUz163CA5Ys1OUetQ8AEAAACoYRaLRd07tpPf
stgHgt/qHc7AGu57UGltns6NULVuY1PsWs1PUShR8AAAAAGCC2Ogo9e3RRUfzCcyrdbCMlzKOGIW4TkgEIBof88
RpTcq+m6GYVW6LNjlM1Ol8hRaeYnaJWouADAAAAAJP06tpPO6tpRcdFRFR0lZQWOn5TlHFNZwIQLB5y9lfA51Parmth9l
RzozFKp37R7NT1FoUfAAAAABgksapyYLxh1MxNm8NpwIQbI4YMRpXcqfuTUo
KXZKWotCj4AAAAAMInFYlG/nl1kt9tVtvZxUw2LhVl8AKrO+67egGuB8SoutferePt197zA7Qa1GwQcAAAAAJmr
f6iy1bJaugzmHKzm3fKqJUEdaKO+OCwG+Rb0Tq2tLbdLv/TzpmTTA7zqlp1l9q3N3sFLUaBR8AAAAAmMhut2ndgn
x5yu93yeCo+jmuzsKMugKo3191N/cue0hfWAfLLYnack+s32ewEtR4FHwAAAACYrHvH9kpPS9HBnKNKz7eLLFG
oxV/DqQAEu2LDoYmlf9BNvvt1xNrQ7DiVS+kotcwwOWtR8EHAAAAACaLLjtqjXAjOLD
0A1me/poH6lT2qudbD8ta0q6jvZ7AR1Qi37XQMAAACA+qlX145KTW6g7MNHKz3fKapYDtbiA1BNnArVbaUTdL3
vAeVYU8yOc1xiK+nsS8xOUSdQ8AEAAABALRAXE63zenHiXsfwC+f0VH8cNtRrqFl1kQjIA9cliT1v1K52m/1iHy
yebuWEueFiympyhjqDgAwAAAIBaok+PkzPzz9ZohV8QrNcHz/0OwWxy7lLr2S9eSGSw3Z0K
2I8y5dx1EwQcAAAAAtURyg0T17dFFR0qfN0kRt4rjdlwsr+w1eGeLNOSRGrxf3
UfBBwAAAAC1SP9zuik+NkZ/ucKlRmLNmQwGot/yy6YGyKzTW84j2WJvWzEO7XS6lda2ZewUJCj4AAAA
AqEXSO1IOoFc35eQeqXQtPun4LD6LKs7wA4Dqss7XVANLH9UblkvlUUj13cgeLg2eUn3XD1IUfAAAAABQy1zQv
49SkhvqYM7hSs9bRhd5HlM220tqucmvW+WYhtXz7WDGAUfAAAAANQyDRPjNaB3d+UXFMrr9VU6pnOM5LDWcDAA
QPfSfqVqgF2x7jpHiyYSq8Plw5i9yBDpE5SjOoE2ZP5XwexVDGAUfAAAAANQyDRPjNaB3d+UXFMrr9VU6pnOM5LDWcDAA

```
oLJLH461OTPfoIoVYKp/hBwDVabOvkTJKHtHLlivlsoRV3YUjGOr976y669UjFHwAAAAAUAsN6NVdLZqla8+BQ
5Wej7D51TmquIZTAcBPniwbpWGuJ7TZ1rZqLjjwPiksumquVc9Q8AEAAABALRQZEa6Rg/rL5/WqtKzyTTU6RBU
rylb5DD8AqAm7/EkaXjJFT+l6lVnCf/uFGrSRul1fZbnqGwo+AAAAAKilenY+W53bt9Ge/QcrPW+3SD2ii2o4F
QBU9JJziC5wTdM6W4ffdoELHpZs9qoNVY9Q8AEAAABALWW32zVicH85wkJ1rKCw0jEtwsvUMMRdw8kAoKL9/kR
dXPJnPaIbVWKJPPUXNj9PajOs+oLVAxR8AAAAAFCLtWvZXH26ddaBQ7kyDKPCeYtF6hVTefkHAGZ4w3m+Bjmf0
ipr1l8da9gdOoXP1kCq4EbBBwAAAAC1mMViObDz+yohPlY5R45WOiYlzKOW4aU1nAwATizHiNVlpXfrz8YkFVl
iTjjOct69UmKLGkwWnCj4AAAAAKCWa5yarIx+vXT46DH5fL5Kx/SJLVCEtfJzAGCWWa5zdZ7zKS2xnaMKc5BTO
krn3m5GrKBDwQcAAAAAdcCgvueoaaNU7TuUU+l5h9VQ/7j8mgOFAKcgz4jS+JLJust/h/L8UZIkw2KTRr3Axhp
VhIIPAAAAAOqAuJhoDT+/n0rLyuRyV76pRrrDpTYRJTWcDABOzWx3Tw1wPaPlMcO13rdIjbqZHSloVHvB9/jjj
8tisWjy5MmBYO6nU7feeqsSExMVFRWlsWPHKien/E+h9u7dq5EjRyoiIkJJSUm6++675fV6qzsuAAAAANRa53b
vrLYtmmv3/oMnHNM7plDRNr53AlA7FVuidK93opzn/9XsKEGlWgu+VatW6R//+Ic6depU7vgdd9yhuXPn6oMPP
tCiRYt08OBBBjRkzJnDe5/Np5MiRcrvdWrpOqWbMmKHpO6drypQp1RkXAAAAAAGq1sLBQXZgxQDarVfkFRZWOCbE
aGhCXLOvF1a4AwHRWi/TMFVOUHmozOOpQqbaCr7i4WOPHj9frr7+u+Pj4wPGCggK98cYbeuaZZZzRoOCB1795d/
/rXv7RO6VItX75ckvTFF1/o+++/19tvv6Ouxbpo+PDheuSRR/Tyyy/LfYKp6AAAAABQH3Q9u6OG9u6p/YeyT/i
UU2qYWx0ieVQXQO1zY/+z1L1pgtkxgk61FXy33nqrRo4cqYyMjJHLHMzMz5fF4yh1v27atmjRpomXLkmSli1bp
o4dOyo/5OTkwZujQoSosLNSmTZsqvZ/mXWG5DwAAAAANhaLRaOHna+WzZpo594DJxzXI6ZQ8XZPDSYDgJN
rOSBCdw5pbXaMoFQtBd+7776rdWnKzQ0VHFxceWOJycnKzs7OzDmt+XekfNHzlVmxowZio2NDXyaNGlSBU8AAAAA
AGqfuJhoXTryAoXXTrySoXXYbTqSl1/pGJtFOi8pYgcqqKp7M9MkpdOKyr9mwp+bj3/bj3/
8o2bOnCmHw1lHVlz+h+++/XwUFBYGffv21di9AQAAAAKCmdW7fhn7fWhn9eys797Dcnspn6jUI9ahLdOVr9QFATbrl/
Jbq2DjW7BhBq8oLvszMTOXm5qpbt26y2+2y2+1atGiRXnjhBdntdiUnJ8vt0dntdiUnJ8vtdis/P7/c63JycpSSkiJJSklJqbC
r7o+//nHML4WFhSkmJqbcBwAAAAEK4vFolEZ56ldq7O0c89+GUblM/W6RBWrYQhrmQMwT7uUKNO+uJXZMYJal
Rd8gwcP1oYNG7R27dpySs/Fr/b0/yA/AlX8vnAbOK5+2aPT7d+Um5W+KbVdjTdm8fGzQd5qB4EBMn7HoPIj06HKLJ5Qb1jgMQt7N8Xw5Sb7MQ2iHS0mBmuRD3UXS0iDRRVnvrxYTTQ4K0hjjm+kCxgdzzY8ZF9l82i4/m8pk5XZxodZh7OVXg3PGG8vrfLq4ujC5P+lfA6qKeS7cOhmczPLgHjj+bzYqWhqMwLjO7wE5sC60EN5Ka0aiTEYfWM=
ykBnC6b/GpZ+r0uah+vm66+TPGxMWZHwhliSxQAAAAAwAnFxUTrxvFjZbFatGrtRp3VNF0R4Y4Tju8UVaJYm09
f5cfJazCnBKhtQuVVq9JNuqhXa1l9yUgeyw0S/G0LAAAAADip2Ogo3XjlGPXp3lk79+xXfkHRScc3DXfqwgZHF
MEOu0CtEi63Opat19WDu+qGyy6m3AsiFHwAAAAAgF8VEx2lm66+VCMG9VN27mFlHz5y0vENQry6uOFhNQg58dp
9AGpOtFGi7p6NmnDxQF0xaqhCQ0PMjoQqRMEHAAAAADgljrAwjb9khK66ZITKylzatXe/DOPE6+1F2vy6MPGom
jrKajAlgF+K9x1Tb+sO3XLlKA0/v5+sVuqgYMPvKAAAAADglNlsNg0/v58mjh+riPBwZe3YLZ/vxI/i2q2GMuK
PqVPUyR/rBVA90tz7NTAqW7dfd6n07dFFFovF7EioBmyyAQAAAAA4LRaLRb26dlR8bIymv/+xNm/bqVZnNVVYa
OgJxkvnxBQp1u7Vkvw4+UXBAFS3UItPzUo265z0SE244ko1b9LI7EioRszgAwAAAAD8Jq3PaqrbJ4xX5/ZttHX
HHhWXlJ5OfJuIMg1PPMrmG0A1S7A51b5wlc5v3UC33UC5Vx9Q8AEAAAAAfrOUhomadP04nX9uD+05cEhH8vJPO
j41zKOxSblqzrp8QLVoEZKvlsdWaVD3drrluiuU3CDR7EioATyiCwAAAAA4I9FRkZowbowS4+P06YLFcrpcapS
SdMK1vhxWQ4MTjmlrqVPLCmLlMZh7ApypEItfnWz75cjfqwsGnqtxFw+VIyzM7FioIfwtCgAAAAA4YyEhdl068
gJdf9nFkmFo++598vv9J31N64gyjWl4WMmhrhpKCQSnhiEudXevV5wzW5eOvEBXjxlBuVfPMIMPAAAAAFAlLBa
Lzj+3pxLiYvXWf+Zqy/ZdatW8qUJCTvytZ7Tdp5GJR7WuOEpriqJlsAEHcMosMnS2I1+RuRvUODVZV44eo24d2
rJTbj3EDD4AAAAAQJXq3L61/jjhKrVpOUxZO3bpWEHhScdbLVLX6GJd1OCIYm3eGkoJ1G1RNq/6he5RVO4G9ez
UXnfddK26d2xHuVdPUfABAAAAAKpck0apuuP3V2vYwH46cvSYdu7d/6uP7DYM9Wh0w8NqG1FSQymBuuksR6m6u
dbJXpyjMcMHadIN45Sa1MDsWDARj+gCAAAAAKpFTHSUrrtslNq1aq4PPv1C32/doeZNGisyIvyErwmxGuoXV6B
Oh1Pf5MfJ6bfVYGKgdguz+NU98qj8hzYrMamBrrxomHp26cCsPVDwAQAAAACqj8ViUa+uHdUsPU3vffK5Vny3Q
THRUUpLbnjSUqKpw6WGDQ/rm/w47XM5ajAxUBsZahNRquaevco/dFjdOrTTVaOHq3FqstnBUEvwiC4AAAAAoNo
lN0jULdderusuu0gWiOWbt+2Sy+O+6WsibH4NTcxT39h8hVpO/ngvEKwSQ9waEZ+j5KPr5HOV6IpRQ3X7766k3
EM5zOADAAAAANQIu92uIQP6qGXTdL37yTyt27xVKUkN1DAh/qSvaxdZqmYOpzKLopVVGsFOu6gXQi1+dY8uUiN
/jvbvOaSWzZvoilFD1bFtK7OjoRai4AMAAAAA1KizmjbW5N9frf8uWKx5Xy/VsfxCndWOsey2E6+3F27zq19cg
dpFlmhZQayy3WE1mBioWS3DS3VOdIEOZx9QjsutjH69NXZkhuJjY8yOhlqKgg8AAAAAUOMiwh26dOQFan1WM70
3d562bN2pJo1TFRMdddLXJYZ4dWGDo9pZ5tDKwhgV+/i2FsEj3u7RubEFinTnaef2bKUkJerasaPUt2cXWa2ss
oYT429CAAAAAIApLBaLOrdvrSaNUvThf7/UohWZyisoVJOOlF8tM84Kd6qJw6n1xVFaXxwlrOH5gborxOJX1+g
itQ7N1979B1VssWhgn+66aMj5SktuaHY81AEUfAAAAAAAU8XHxuh34OarTYtmmj1vgTZt3aHGqcm/+jii3SJ1i
y5Wm4hSrSyMOY6yiBpKDFSd5o4y9YrJV9HRXG3fX6DWLZrq4iHnq+vZbZi1h1NGwQcAAAAAMJ3ValX/Xt3U6qy
m+t+CxVqyep1yco+qaXqqwh2Ok742OubX+fH5ahdRqmWFMTrqCa2h1MBvF2v3qE9MoWK9edq9/ZAS42M1/pIRG
tT3HEVGhJsdD3UMBR8AAAAAoNZIaZioG64Yrd7dO2vu/K+1Ycs2hYWGKr1R6kk34ZCklDC3Rjc4oq2lEVpdFKO
y/8nHA2aIs3vUJapY6SFF2nfgkEr9fvU7p6suHjJQ6WkpZsdDHVXlcz2nTp2qnj17Kjo6WklJSRo9erSysrLKj
XE6nbr1l1uVmJioqKgojRO7Vjk5OeXG7N27VyNHjlRERISSkpJO9913y+v1VnVcAAAAAEAtY7FY1L7VWbrzxmt
10/jL1DAxQVu27VTO4aMyDONXXiu1iSzVZUm56hxVpBCLv4ZSAycXb/doUHyexjTIVVTJPm3dvkuNU5I06bord
Mu1l1Pu4YxUecG3aNEi3XrrrVq+fLnmzz58vj8eqjIUOGqKSkJPz/P/P
5NHLkSLndbi1dulQzZsQ90nTNWXKlKqOCwAAAACopUJC7Orfq5vunRB1GuoZU
6QrknPUNapIoRR9MEliiFsZ8Xka0/Cwkvx52rxth3xen664aKjuveUGnd01I2vt4YxZjF/78ccZOnz4sJKSkrR
o0SINGDBABQUFatiwoWbnNmqVLL71UkrRlyxa1a9d0y5YtU+/evfXZZ5/pwgsv1MGDB5WcnCxJevXXVV3Xvvffq8
OHDCg399fUUCgsLFRsbq4KCAsXEnHxh0uow+0+tQPAZ0ybV7AgAgCpm9fQF2272C25n65WCu/2yC/36+
m6WkK04XvDyXJ7bfo+5JIbSyJlJNHd1EDGoa41TW6SE0cLnl9Pu3Zf1Bej1fd07XX6KGD1Cw9zeyICCLVXhEXF
BRIkhISEiRJmZmZ8ng8ng8ysjICIxp27atjRpomXLlkmSli1bps4dOgaTUka0nSoCgsLtWnTppq8HgAAAAAChdL
TUnTzNZdp8u+vVquurmmr77n3ae+CQfL5fn50XajXUJbpY45Jy1TumQBFWXw0kRn2UF0LWsISjurjhETUKKdOB7
Fxt3rZTyQ0Sdf0l1+u2G66k3EOVq9ZNNvx+vyZPnqy+ffuqQ4c0kqTs7GyFhoYqLi6u3Njk5GRlZ2cHxvy83Pv
x/I/nKuNyueRyuQK/Lioq+qK3AQAAAACoJSwWi7qc3UbtW7fUNq1GsWPvYsVY8uVbZMCRcC0BEp4TcmpHjsfVvKNqDWP6VTAwATPTAZ8GrRQ+EnAbEERxu00KxYwl8Fp8SpDcDdjZaHmeGzwNa81rRaLLBtKlcFBOsdPwVVC0WEa1vs2iTmetWEq3sEd+T9MR7n3BTu6
6gStYssOY6ycG0sjlKeN6SG0i0YpYS61C26SGlhbvl8Pu07eFj5hUVKTWqo8aNH6Lw+PRQbHWV2TASpai34br3
1Vm3cuFHffvttdd5G0vHNPf76179W+30AAAAAOYLCwtVRv/e6tqhnT5ftESLl2dqU9YOJSclnlLRZ7NIrSPK1
DqiTPudYdpYEqn9rjBJJ38dUJ6hRmEudYkqVmqYW16fT3PRRecp3N7dFF8LEsyoHpVW8E3adI
kffrpp1q8eLGaNGkS+NyUlBS53W7l5+eXe8vy0KFDSklJCYxZuXJlufke2nXp0s7A8Xem9993r7774LOHz9JSU
lfV2wEAAAAA1EKJ8bG6avG6avQI9e3RRV8vZrU9Z2JTVIUMPEhF8t+iSpscOlxg6Xjnns2lASqR2lEfJR9OE
kIqw+tYooVZuIUsXYffJ4vdqzP1dFxaVqnJqsS4adr3N7dGHGHmpMlRd8hmotttu00cffUff/IKVvLtaaxmq1aa9d0y5YtU+/cvePe
/fuCgkJ0YIFCzR27FhJUlZWlvbu3as+ffpIkvr06aNHH31Uubm5SkpKkiTNnz9fMTExateuXaX3DQsLU1hYWFXfDgAAAAAglkIhWdkgYoeaNPyBQe7YgVV2Q3Y4t7/c10DT7M8bKSz0BwZ7mE2LwB5LtVBRDmeB8W9+/KzgNIeSpSBj+4IInb/LSD6Ar1XHtPpTqY9kD5hG/J7o0q5Cu+EdS+UYdM2OOZoJP2VR0NN9/RA1gPWQkKhYsPHBDPQzYN+xBuf8kgoTMU0Lsux2RisXLDWrLGnp3ZngXgnKmLt8Hjh2GN
/HQAAAABAHdCOcZquu+wiDe7XS4tXZ0OrbVWu1act2JSbGK7lBwintUhof4tWAuAL1jC7StrJwbS+N4PFdBFhlq
InDqTYRpWoU5pLVInk8Xu3al6PSOjKlp6XosguHqfXjoqh2EMNq/JddG+55RbNmjVLH3/8sc4+RjRjb
```

```
wcEnSzTffrP/973+aPn26YmJidNttt0mSli5dKkny+Xzq0qWL0tLSNG3aNGVnZ+uaa67R73//ez322GOnlMPs3
cnYRRdAVWAXXQAIPmZ/nQrUF9mHj+rblWu0aHmmDh/NU1xsjFKSGshu070ddPM8dm3/oewrZffdeine7lGbiFK
1DC+Tw3Z8Qxe326MD2TkqdbrUtFGqMvr1Uu9unRQVGWFyWtRXVV7wnWj687/+9S9df/31kiSn06m77rpL77zzj
lwul4YOHaq///3v5R6/3bNnj26++WZ9/fXXioyM1HXXXafHH39cdvupTTo0+wsnCj4AVYGCDwCCj9lfpwL1zeG
jx7R09VotXpGpgzmHFe5wKC2loRyn+QYYUiH3KHaXhqhXU6HPMavzwhE3RVq8eus8DK1iShVw1BP4LjL7daBQ
7lyul1q1jhNF/Tvo306dFBkRLiJaYFqKPhqC70/cKLgA1AVKPgAIPiY/XUqUF8VFhUrc8NmfbVslXbt2S9DUq0
UJEVHRZ72tbyGtMfp0PbSC013hclgvb4gYSg11K3WEaVqHl4m+w+/rYZhqKCwWDlHjsrv9+usJo2V0b+XzunSQ
eEOh7mRgR9U6y66AAAAAADUBjHRUTr/3J7q26OL1m3eqsUrju+6u/fAITVskKDEuDjZbKc2K89ukVqEO9Ui3Kk
yn1U7fyj7DntCq/ldoDok2D1q6nCq1Q8bZvzI6XIr98hRFRYVKyYqSl07tFXvrh3VrW07054BClQ3Cj4AAAAAQ
L0RGhqinp3PVveO7bRt114tWfWdMjduVta0XbLb7UpKTFBsk2EoJhCUsQZEfS00J
GWJR6xfqowFV/lIdof62tl6JtsfcKIn0oClWrxdKqQH1oQbQIWgVFlFIVoWW5EjYBAxIggSRkTybJzPf+ERkdw
paQbZLX8/E4j5n5ns8553s++c4J+XCWiEt6+q4khdp86h9eof7hFSqutelgRZi+8jhVUGOXOLOvTbLJqLPTo66
uKnV1ehTxraKez+dTwelinSoolM1qVZf0ifr+6GsOuH+aUpKTLnlcAC2NAh8AAAAAoMOxWq1K69ldaT27daLRD
craf1Bbd+7WF18eVs6JPIWHuZQQH6uIsEt/aEJUiFfD3KUaplJV+Sw67nH6pxIvf363plCrV1d8XdS7wumR3Rp
4t7KyigrlnSpQVZVHMdFRun7k1Ro+qL/69UqV08mZmWj7OMIAAAAAADq02OgoXZM+TKNGn2n9Qn23/X
EeOndDhquOKdkcqIS5GTselF3pcVqPUOCqlhlZJkspqbTpW7fAX/Cp5Im+zCrH4lOSoVhenR8lOj2JDanX2yXe
1tbU6WXBap4uK5XI51SOli0YOG6RB/dKUEBfTOhOHGokCHwAAAAAAkiwWi6onQnHPSWsJxhTUh/mLfiWoHT+W9TBYZxdtrlOz0qIvToRRHtWznuJr
WGKOiklKdzC+U1+dVYnycbhl7nYZcdaV6dU+RrYE/V6CtoMAHAAAAAAAMBZbZDab+qR2U2qnZD0g9KOdRIUa79kBmzk6MIRQtjt0fa77FaZHTRqevL6KvRnT
ZmlwOePy+v1Ki4n5ljUVpbaPhn7FTSUe5sv7MzzU5lOorVr+kV9MsNbdaJcvCleunyQe/xhVXE9vbAXapBcmxk7Li2VlMNrvGrbsL1d/gBjaUMSa7Sl/OiyXI2cvhm3NbSGU/Z57GuDeb3YSP0I5SXr5
ZmlwOePy+v1Ki4mWnGx0QqxNaxAFxHiVVpIpdLCKiVJhTUh/mLfiWoHT+W9TBYZxdtrlOz0qIvToORHtWznuJr
WGKOiklKdzC+U1+dVYnycbhl7nYZcdaV6dU+RrYE/V6CtoMAHAAAAAAMBZbDab+qR2U5/Ubvp+5jXad/Cwdu7er
+1Ze/TFl4dls9qUEBejaHekrNaGF+di7bWKtdfqohy+x6ncqsd0l1jVwVn+J1XiMWn2JBaxdtrFGu
vUby9RjH2mnMW9KS6M/UKi0t0uqhEtbW1clr8R3hlVfZZUxdKCuSuvVqCcpA20NBT4AAAAAAC7A6XCoS3o
ltvvEG79x/Uvz/fo70Hvtk4fNoXLRNs7jaaF0AQ2fp4e8FbDVzk8yWVXZNOjg/vZHgb+9jgXhx3ZeJFexvmqWnNV1G3UU3f0ZN2ZQUUSTRBlbVLeVijFJdP6MjQZnuRvOerGyzVFm3tTMMXUB7ViMO4Ldda2y+vmpNy9jUi31FbRrSGgKP17oeI6RqiTyE9fOEW5zYbUUGuXbtPTRKjtIhx
vQx4nVVntHGOkmUMgTzgtM3l98aUkg/mLa5jdpJprCL0kwz8o/FzEzqz3Sjr3omE0zTjMu0zfZFSybYOewtGRFhxaYJBrt18SlFGgP2mr5gTtqWDjxvvJjXBm/MQA8pjj/0WmhixC6o5aZnxx6UZZ8FGmfX8dB5hr/+EvFdOyNqmLAzw
cTlzLH9Ujw/1E8wV3nR9/FpCrqH8AsdHvbNTAALhABxrVzlE9VTbKxQ06vlyeWz5F5KAzCs+9Jn0sZQVaz1wRe
ydn2ruzM1+0FMBrHLYPDjhX6yaq2LqlpLtsXiLfvG9U6pTsJ8GM1nXJ3W7xy0XG/3uDiJh1j6T4JaZx3o/Tg6Iw6sFIkI0i3bx2YLeJg7wdbvRPyDgz0aPT/pvXS6EfEdI2I/7M1xILyZg3/L6yp4T8cqnRWE3HdGkGe0xWYJRgbhLJWoOEbUrePDs/BrWQ8D3DvK4RR2V7iQo1i1ZuYDq3cpmpoFlTR1LlXqFpV
```

N0FAAAAmh2X6AIAgBaVNmS4dm3+WMezD0mSDu/brX3bt2rINTecM74o/5S2/20DRt82+aLrfvHR32jYdaM1aOQ
1jepb97R+2rdtqzxV1dr58UbFJCTKHR0rTW+vkt3pVPqYGxu1XgAAAKA5cQYfAABoUbfe+zNVlJfq/puukdVmk
8/r1X/0mKVrbv7BOeM3rl6p0PAIpY+96YLr/fid1fpyzy498ca7je7bDbdN1pEv9mjGh0sUGR0rB55ZrLLiIq1
4doEeffkN/fWZJ/TJu2uUmNJN9z32l0ISOzd6WwAAAEBTocAHAABa1Kdr39I/316lGQsXKaVXmrL37dbSx+Yop
l0irr/19nrxG/62QqO+f6scTtd515l/4piWPDZbs5esuGDcxYTY7bpnduC9+f7w0AzddNfdyt6bpa0b1ul3qz/
Q6hcX6aXfPqxfPfdio7cFAAAANBUKfAAAoEW9vGCebr3nZ/rehImSpG5pVyr/eI5W/fm5egW+Pf/eouPZh/TA0
4svuM5Duz9XcUG+fvmDcf42n9erPf/+TGtfXaoVnx+WzWZrcF93ffaJjh78Qj/57e/08pPzNPSaG+QKC9PIG/9
Da1899xmHAAAAQEujwAcAAFqUp7JKFmvgbYCtVpuMz9SL3fDGcvXsP1Dd+/a/4DoHfmeUnn7rw4C2P/zmF+qS2
ku3Tr+vUcW9ak+VXpz3G/3Xgj/IZrPJ5/NKtXV99NbW1H0GAAAA2gAesgEAAFrU8OvH6G+Ln9W2jR/oZM5RbVm
/Vm8v+5PSx4wPiKsoK9Xm997W6B/+5znX88iPbte7ryyRJIVGRKhrn74BkysOTJHRMerap69/md0nTip7b5Zyv
8qWJB35Yp+y92aptOh0vfW//vwzGnrNDUrtN0CS1Hfo1fps/bs6vH+P1r66VH2HXtOk+QAAAAAuF2fwAQCAFjX
9f36r5c8+qT8/+pBKCgoU0ylRYybdpR/+9BcBcR+/s0bGGP+1vGfL/eqwSk8XNmjb7694WSsXPeX//PCdt0qS7
nvsad3wg0n+9q++2KdP172t37253t+WMe772r11sx6+41Yl9+ipGQsXNWjbAAAAQH0xGGPqXw/TDpSUlCgqKkr
FxcVyu90tvv1V+0+0+DYBtD8/SOMJaHy7RBQAAAAAAAIIYBT4AAAAAAAAAgiFHgAwAAAAAAAIIYBT4AAAAAAAAAAgiFHgAwAAAAAAAIIYBT4AAAAAAAAAAgiFHgAwA
AAAAAAIIYBT4AAAAAAAAAgiFHgAwAAAAAAAIIYBT4AAAAAAAAAAgiFHgAwAAAAAAAIIYBT4AAAAAAAAAAgiLXpAt+iR
YvUvXt3uVwupaena+vWra3dJQAAAAAAAAKBNabMFvtdee00zZ87UnDlzH37dg0aNNEjjxo3TyZMnW7W7trAAAAAAA
AQJvRZgt8Tz31l0655x5x5NmzZN/fr1O+LFixUWFqYl5a0dtcAAAAAAACANiOktTtwLtXV1dq2bZseughf5vVa
lVmZqY2b958zmU8bhs0Ho88//HtSh3oc3h+AqgbTWMfbMv0+NMa2yfQA
AOP60yQJffn6+vF6vEhMTA9oTExO1b9++s7+OzDXqU1Jv+4c+QcjGvzIK9
Nxxij0tJSJScnt3ZXAABAO9EmC3zx8fGy2WzKy8sLaM/Ly1L1NSUtI5l3E6nXI6nQFt0dHRzdXFDsntdvMP+mZAX
psHeW0e5LV5kNfmQV6bB3ltGpy5BwAAmlKbfMiGw+HQDtGHDBn+bz+fThg0bGJGR09AwAAAAAAANqWNNnk
GnyTNnDlTU6d01fDhwzVixAg9/WTs8kNnJWZy7/wzKi8v17Rp01q7awAAAAAAAECb0WYLfJMmTdLp06f1yO
l29B2+g+TmdTs2ZM6feJdC4POS1eZDX5kFemwd5bR7ktXmQV
1VwAAgLbLYowxrd0JAAAAAAAAAAAI3TJu/BBwAAAAAAAAAAAAAghgFPgAAAAAAAACCIUeADAAAAAAAAAAAAAghgFPqiwsFB33HGH3G63oq
0jdffdd6usrKzGZUaPHi2LxRLw9+Mf/zgg5tixYxo/frzCwsLUqVMn/eIXv1BNTU2AzYbNG3T11VfL6XSqd+/eWrp0aVNvXtChwAcAAIBmc8cdd2jPnj1at26d3nnnHX3wwQe69957z7vc9OnTlZOT4/+bP3++f53X69X48eNVXV2tjz/+WK+88oqWLl2qRx991B9z+PBhjR8/XjfeeKN27NihGTNm6J577tH777/fLNsZLCzGGNPanQAAAADan+zsbPXr10///Oc/NXz4cEnS2rVrdfPNN+v48eNKTk4+63KjR4/W4MGD9fTTT5/1/ffee0+33nqrTp48qcTEREnS4sWL9eCDD+r06dNyOBx68MEHtWbNGu3evdu/3JQpU1RUVKS1a9c27YYGEUbwAQAAoFls3rxZ0dHR/uKeJGVmZspqtWrLli01Lrt8+XLFx8drwIABeuihh1RRURGw3oEDB/qLe5I0btw4lZSUaM+ePf6YzMzMgHWOGzdOmzdvbopNC1ohrd0BAAAAtE+5ubnq1KlTQFtISIhiY2OVm5t7zuVuv/12de/eXcnJydq1a5cefPBB7d+/X6tWrfKv99vFPUn+12fWe66YkpISVVZWyuVyNXr7ghUFPgAAANTL7Nmz9cQTT9QYk52d3eD1f/sevYEDByopKUljxozRF198oV69ejV4vTg/CnwAAACol1mzZumuu+6qMSY1NVVut1unTp0KaK+pqVFhYaHcbnetP+9b7/lWhw4dUq9eveR2u7V169aAmLy8PEnyr9ftdvvbvh3jdrs79Og9iQIfAAAA6ikhIUEJCQnnjcvIyFBRUZG2bdumYcOGSZI2btwon8/nL9rVxo4dOyRJSUlJ/vU+/vjjOnXqlP8W4HXr1sntdqtfv37+mHfffTdgPevWrVNGRkatP7e94im6AAAAaDY33XST8vLytHjxYnk8Hk2bNk3Dhw/XihUrJEknTpzQmDFjtGzZMo0YMUJffPGFVqxYoZtvvllxcXHatWuX/uu//ktdu3ZVVlaWJMnr9Wrw4MFKTk7W/PnzlZubqx/96Ee655579Nvf/laSdPjwYQ0YMEA/+clP9K//+q/auHGjfv7zn2vNmjUaN25cq+UjGFDgAwAAQLMpLCzUfffdp7fffltWq1UTJ07Us88+q4iICEnSkSNH1LNnT/3tb3/T6NGj9dVXX+nOO+/U7t27VV5erm7duukHP/iBHn74YbndbknSkSNH1LNnT9ntdoWGhio8PFxTp07VvHnzFBLyzc2qWVlZ+q//+i/t3btXXbt21SOPPHLeW4s7Agp8AAAACEq1eYiHy+XSrl27avUQDzSMMfUrLvqQDQAAAAStPXv2KC0tTWvWrNHixYtrvE13wYIFeu655/TWW28pLy9PDodDjzzyiCZNmlRjPGqHAh8AAACCUvZbKY9PV06NChunzzz1v+N1+KAKDcePKG5/2kk6ffq04uLigvnnNngcBcMAAAAACvYFgMG4pAkZWrjHCtPnXyC6kI0pMhQZAAAgiGuNxSnpsXqH1GfpMCAAAF3gCWrYSgAAHw5nyxStRBEHvoAAAbgyQ0ZN68k8gAAAAAAADTEK
Bugd01f5g8Hh7i/ZN61XCJOoUbI1GXsTIUubgMyeBz0N0W0dNxsGkLgEiLQsXdOHVXxrl1

```
ZtDV//etfNXfuXK1ZsybgPwlvvPFG//uBAwcqPT1d3bp108qVK895n14AAAA0Hc7ga6cee0AB7d2794JTamqqk
pKS6t1UvLa2VoWFhUpKSrrgNkpLSzV+/HhFRkbqzTfflN1uv2B8enq6cnJy5PF4Lnv/2pL4+HjZbDbl5eUFt0f
l5Z03h0lJSReMP/PakHW2N43J6xkLFy7U/Pnz9f7772vgwIEXjE1NTVV8fLw0Hjx42X0OBpeT1zPsdruGDBniz
xnjtc7l5La8vFwrVqy4pCJIRxuzDXW+46vb7VZoaGiTfAc6shUrVmj69OlauXJlvUuhzxYdHa0+ffowVgEAAFo
ABb52KiEhQX379r3g5HA4lJGRoaKiIm3bts2/7Icffiifz6f09PTzrr+kpERjx46Vw+HQW2+9FXCG1Pns3LlTM
TExcjqdTbKPbYXD4dCwYcO0YcMGf5vP590OGDRsCznr6toyMjIB4SVq/fr0/vkePHkpKSgqIKSkp0ZYtW867zva
mMXmV6p7m0m/ePK1bty7g3pLnk5OTo4KCAnXu3LlJ+t3WNTav3+b1erVr1y5/zhivdS4nt6+//ro8Ho/uvPP0i
26no43ZhrrY8bUpvgMd1fLlyzVt2jQtX75cEyZMuGh8WVmZDh06xFgFAABoCa39lA+0vvHjx5shQ4aYLVu2mI8
//tj07t3bTJkyxT8/JyfH9/JyfHpKWlmS1bthhjjCkuLjbp6elmwIAB5uDBg+bEiRP/5PFhrbFtvvWVeeOEFs2vXL
nPgwAHz/PPPm7CwMDN79uxW2cfmtmLFCuN0Os2yZcvMnj17zL333muaYu+66y8yaNcsf/8knn5i
QkBCzcOFCs3fvXjNnzhxjt9vNrl27/DHz58830dHRs2aNebzzz83t9xyi+nRo4eprKxs8f1rLQ3N6/z5843D4
TBvvPFGwLgsLS01xhhTWlpqHnzwQbN582aTnZ1tPvjgAzN06FDTu3dvU1V1Sr72Boamte5c+ea9957zxw6dMh
s27bNTJ482bhcLrN7925/DOO1TkNze8b3vvc9M2nSpHrtjNm6HOzYscPSDJPPfWU2bFjhzly5IgxxphZs
2aZu+66yx//5ZdfmrCwNmjYbDazbt06f8zFfk4dQUPz+uqr5qQkBCzaNGigONrUVGRP+aBBx4wmWz27YwAAAKCjocAHU1BQYKZMmWIiIiKM2+0206ZN8xdEjDEmOzvbSDIbN2684uo6Zyi
aBBx4wGzduNNnZ2eaTTz4xmZmZJj4+3pw8ebLF9w8AAKCjocAHU1BQYKZMmWIiIiKM2+0206ZN8xdEjDEmOzvbS
DIfffSRMcaYjz76yEg655SdnW2MMWbt2rVm80DBJiIiwoSHh5tBgwaZxYsXG6/X2wp72DKee+4507Vr/+NwOMy
IESPMZ5995p937737bXXmqlTpwbEr1y50vTp08c4HA7Tv7/Tv39/88477wM9/l85uGHHzaJiYnG6XSa0aNHm/3797fEr
rQpDclrt27dzjku58yZyZY4wxpqKiwowd09YkJCQYu91unuunXrZu65554409Uf9GQ3J64wZM/yxiYqK5qabbjLbt28
PWB/j9RsNPPRbs27fPSDLvv/9+vX/+9+vXUxZs//0+dMHqdOnWquvfbaess6nqapaYuXVpvvRf60XUEDc3rt
ddee8F4Y4yyZNGm0qRJ5uDBgy27YwAAAB2UxRhjWuJqW6MQQQQAAAAAABNj3vwAQAAAAAAEG
MAh8AAAAAAAQxCjwAQAAAAAAEGMAh8AAAAAAAQxCjwAQAAAAAAEGMAh8AAAAAAAQxCjwAQAAAAAAEGMA
h8AAAAAAAQxCjwAQAAAAAAEGMAh8AAAAAAAQxCjwAQAAAAAAEGMAh8AAAAAAAQxP4P5z2kxfBunI4AAAA
ASUVORK5CYII=\n"
        },
        "metadata": {}
      }
    ]
  },
  {
    "cell_type": "markdown",
    "source": [
      "1. Most people do not have an email id (90.76%)\n",
      "2. Both categories have almost same approval rates"
    ],
    "metadata": {
      "id": "nqMu8wCjNrDN"
    }
  },
  {
    "cell_type": "markdown",
    "metadata": {
      "id": "SvwP101BKd2a"
    },
    "source": [
      "#### Phone"
    ]
  },
  {
    "cell_type": "code",
      "source": [
        "analyse_categorical_column(\"Phone\")"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 507
        },
        "id": "M4813tZe4mjH",
        "outputId": "a3026785-917d-44c0-9bb7-0a684fbc1770"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 1400x500 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAABPgAAAHqCAYAAACZYfCtAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAACRHkl
EQVR4nOzdd3xV5eHH8e9duTd7Twh7D9kCgoCAgCIVoVUUFdxVsUX7q6Ot1uLWahVnHRWt4GhVFFQQAUHZQ/beI
wuyd+44vz+oV2MCiNzkZHzer1deNec895zvTTEmX57nPBbDMAwBAAAAAAAAqJesZgcAAAAAAAAA8MtR8AEAAAAA
AAAD1GAUfAAAAAAAAUI9R8AEAAAAAAAD1GAUfAAAAAAAAUI9R8AEAAAAAAAD1GAUfAAAAAAAAUI9R8AEAAAAAA
AD1mN3sADXF5/MpLS1N4eHhslgsZscBAAAAJEmGYaiwsFApKSmyWvn7dgAAcPYGMGXlpam1NRUs2MAAAAA1Tp
8+LCaNm1qdgwAANAANNiCLzw8XNKJH3nwiIiJMTgMAAAAcUFBQoNTU1PqhPPxwAAAGerwRZ83y/LjYiIoOADAABAn
cNjZAAAQKDw0A8AAAAAAACgHqPgAwAAAAAAOoxCj4AAAAAAACgHmuwz+ADAAAIBK/XK7fbbbbXYXYM1Cm83
sGAAAoBGh4AMAAKiGYRjKyMhQX16e2VFQD0VFRSkpKYnFrAAAQdGOAAAKjG9+VeQkKCQkJCKGrwsxiGoZKSE
mVlZUmSkp0TTU4EAAAaAwo+AACn/B6vf5yLzY21Y2luw4qGeCg4MlSyes4+LgAKS+bH2sgEAAPAT3z9zLyQ
kxOQkqK++/+3JM5gAAAAAAAAAFAb14pfizw4AAKhNFHkAAAAAAAAAAAAAAAT
MEmGwAAAGfgo53ptXq/ce1/2S6sK1as0MCBAzVq1Ch99tlnAU4FAACAuoQZfAAAAAA3QG2+8oTvuuENLly5VWlp
ajd+voqKixu8BAACA6lHwAQANDBFRUV6//33deutt+tt2r06eunL
Vu2+MfMmDFDUVFRmj17tq2bSuXy6WRIOfq8OHD/jEPPvigunfvrtW7aUy+WSJB06dEiXXnqpwsLCFBE
RocsvvlyZmZmSpF27sdslisWjHjh2V8v/tdvt27aqJm/n+W3k3H/fdeFEE7L6BHBIZ2GtVOqeupgkcJ3PjQQNEOqQZUtU/hJ6D4Ew
7CwMCUnJ+vpp58O06NcPAACgvqHgvZG6du3aVuzcvZs7v/2yt2bugwf9btceVKg3tRCdqsMrwtJUxYqcuTeZWunD3bqxtdhsCz6UMIDQ8Knv7HeYH2YavGV1O2sk8HOVGP4Z3KlYzvv7mn9ERK5uVCm2XL7b68




































































uVLNmzSRJubm52rzVrlzp27FjpWmvXrUvU3925c6enTp2+cdNjFuSTbg++8NqyqPWo6Z1Q80XIAe5fWWnuVx5UB49


IuXPnKjc3VzfclcO6Ol0dOlS6WP8+PGVlulOmzOFDFl6mcjm5pBKi6MzR+3TWSjE+jsoU6ZAY9Pi0hhLhMZQ7QdXyH5
```

Mnq16+fv/CrzvDhw9W1a1dNnDhR69ev1+rVq3Xttddq8ODB6t27t3/cuHHjVFhYqFtvvVUXXHBBpdmGt99+u3J
ycnTllVdqzZo12rt3r+bPn6/rrrtOXq9XYWFhuuGGG/THP/5RixYt8ue3WvmxFgAANF78JAQAANBAvPHGGxo+f
LgiIyOrnBs/frzWrl2rTZs2SZIef/xx/f73vevXr2UkZGhOXPmKCgoyD8+JCRE99xzj6666ioNGDBAYWFhev/
99095f4vFok8++UTROdEaNGiQhg8frlatWZ1V5XXh4uMaMGaONGzdq4sSJ1c61pKRo2bJ18nq9GjFihLp27aqpU
6cqKirKX+I99dRTOv/88zVmzBgNHz5cAwcVK9evX7R1wwAAKAhsBiGYZgdoiYUFBQoMjJS+fn5ioiqPX7sOQ
XQCCwRBcwR1lZmfbv36+WLVvvK5XKZHSegvv76a11wwQXKzc1VVFRUtWNmzJihNyWNSSn+jNk9s+pA
ACg4WEGHwAAAAAAAFCPUfABAAAAAAAA9RgFHwAAQCMyZMgQGYZx5kjjR58mSW5wIAANQjFHwAAAAAAABAPUb
BBwAAAAAAANRjFHwAAAAAAABAPUbBBwAAAAAAANRjFHwAAAAAAABAPUbBBwAAAPwCLVq00LPPPmt2DAAAAAo+A
ACAhmTy5MmyWCx6/PHHHKx2fPXu2LBbLGV2LAgsAAKB+sJsdAAAAoD7ZsmvLrd6vS5Z/wal8ulJ554Qrfccou
io6NrIFXdUFFRoaCgILNjAAAAAmI4ZfAAAAA3M8OHDlZSUpMcee+yU47766iuddtppCkhoJAAAA+eyU4z788EN17txZTqdLVqO0NNPP+0/N2TIE
B08eFB33nmnLBaLLLbbtCQCQ00OVmpqq2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q2267TUVFRf7zx9bbMaWdbnKXUhxv+uO/b0Pb4C4C1Nrv/JHNX00Vu/t+tnq//7q

roAVfRkaGJFXasez7z78/l5GRUeWBxHa7XTExMZXGtGzZsso1vj9XXcFXXl6u8vJy/+cFBQVn+W4AAAAAAAKgdX
p+hPVlFlcq87ekFKiz3mB3NFD5DOppXqqN5pfpm9w87uYcG2dS1aaR6NItWj9Qo9WgWrfhw5ymuBDQOAS34zPT
YY4/pb3/7m9kxAAAAAAA4La/POIbDeVq257iW7TmujUfyVOb2nf6FjVxxhVcr9+Vo5b4c/7GmOcH+wq97syh1S
YlUkN1qYkqg9gWO4EtKSpIkZWZmKjk52X88MzNT3bt394/Jysqq9DqPx6OcnBz/65OSkpSZmVlpzPeffz/mp+6
77z7ddddd/s8LCgqUmpp6dm8IAAAOaoZh8kOMUG/xZwdAdfZkFerb3cf17Z5srdqfrcKyxjk7L9C+fw7hnI1pk
qQgu1V9WkTrgvYJGtI+QWOSwkxOCNS8gBZ8LVu2VFJSkhYuXOgv9AoKCrRq1SrdeuutkqT+/fsrLy9P69atU69
evSRJixYtks/nU9++ff1j/vznP9vtdsvhOLGjzoIFC9S+ffuTPn/P6XTK6WRaLgAAOHvf//xRUlKi4OBgk9OgP
iopKZHOw58lAI1TVkGZvt1zXN/u0a7le7KVBdmdqRGocLjO7I92Vq2J1sPf7ZdzWNDNKRdvIZOSFD/VrFyOdi
8Aw3PGRd8RUVF2rNnj//z/fv3a8OGDYqJiVGzZsOdepUPfzww2rbtq1atmyp+++/XykpKRo7dqwkqWPHjho1a
pRuuukmvfLKK3K73ZoyZYYomTJiglJQUSdJVV12lv/3tb7rhhht0zz33aMuWLXruuef0j3/8IzDvGgAA4BRsNpu
ioqL8qw5CQkJksfBAb5yeYRgqKSlRVlaWoqKiZLPxSyTQ2OzKLNTcjWmavzSzTOzMLzY4DSQezS/TWioN6a8VBu
RxWndc6The0j9cFHRLUNDrE7HhAQFiMM1w/8PXXX+uCCy6ocnzSpEmaMWOGDMPQX//V7366qvKy8vTwIED9dJ
LL6ldu3b+sTk5OZoyZYrmzJkjq9Wq8ePHa/r/QoL+2Ha7KZNm3T77bdrzZo1iouNZoliouL0x133KF77rnnZ+csKChQZ
GSk8vPzFRERcSZvMSC2bNlS6/cEOPB06dLF7AhAo2UYhjIyMpSXl2d2FDQwUVFRSkpK4gtgoJHYf6xYczemae6mNO3K
LKo2Omj61H7bJkuDuMpMbXg09yXRzIvHz2/uMv9vs4+SlOLH+/Jq3b4RF9ccQRGjbjFgFNlHsFcvAfZxpGfzWFCzcqZe
2b/nAogMA7nlGjOpjTN3ZiubekFZsdBAHRMjtC4Hk10aY8UJYS7zI4DnJEGs4suAABATbDZbCyzBIAz9OKLL+q
pp55SRkaGunXrpuefff17nnnuu2bHOWlpeqT773Oy9jUfyzY6DANueXqBHOgv0+LwdGtAmTuN7NtGkKDuLnA
NR9FHwAAAAAgIB5//33Rd5ddddd+mVV15R37599eyzz2rkyJHauxjJHauXOnEhISzI53xjxen77clql/rzioIfuz1TDXwOH
HvD5DS3cdO9JdxxTmtGtUlySN69FE/VvH8kxe1Fks0a0hLNEFEAgsOQWAhsfsn1OBmta3b1/16dNHL7zwgiTJ5
/MpNTVVd9xxh+69916T0/18WYVlenfVwXp5QmjhSoWx/LmVn9Tis1cEF6hxl8AAAAAICAqKio0OLp163T
ffff5j1lmtVg0bNkwrVqw6hkiUiYFJ2LTEc/nvL/rriolfuz1tTDXwOHHvD5DS3cd09JdxxTmtGtUlySN69FE/VvH8kxe1Fks0a0hLNEFEAgs0QWAhsfsn1OBmta3b1/16dNHL7zwgiTJ5
ffff5j1mtVg0fPlwrVqwwMdnPt2pftt5eeVBfbs2Q29sg58PgF0rLL9NLX+/VP5fu06jOSbrx/Jbq0Sza7FiAJ
Ao+AAAAAECAHD9+XF6vV4mjJiZWOJyYmaseOHSal0r3ico8+/u6o3ll5UDsyz2zP02eZ0fbY5XT2bRen
G81tpZOck2aws34V5KPgAAAAAAI3S/uPFemv5AX247ogKyM3zz2mx0E9tP5Qnm6buV5No4M1+bwWmBuM4U5qVpQ+
/hTBwAAAAAIiLi4ONlsNmVmZlY6nZmQakJJNSVbX3WJGmL9ytORvT5GMVLgLgSG6pHv5su577areu6JOqyQN
a8Jw+1Cqr2QEAAAAAAA1DUFCQevXqpYULF/qP+Xw+LVy4UP379zcjwvuzXvcWdSoV1xvNPffwj8OWqBC2E2cbbWRA4pXTMl3RA7+Qd43QRecxnA2QEKnx/eruWBmfZfs1s+/v19fjP+Ko9fuktNEZfE5w7Gxb2aCxf8oYHn2gJE0pY4dxEbyIiYuR8Yt/4r4+1kW+M7R1K6njPfc8MB72+hBfNaaqnACN6aP+Nq8kPXd0tGkD5CS1EOUKOJw6a+W2+dpRBX1wY6YhgG2Yn5d2JfS8A2ZuvDGsG5DCrhaF3dtBx4Rf+VwWAHJ9AqZIvLHn8EqXFtQ/6RLE1XVUs8/I0mUSfL8pJVMDmQOuR0bu1R5A9cNs7TAWUbglNAp4xOKkPbAyIDIDJJD9yHw4B5Z55Glbd+Y0IU2mWY8CzC8YgJCkQZAj5xHlaZrHcAD5Cl2N0ZnDyAnH5xHnDBdu3hj3WN6Wyp3PCMlDlcyUs5pqWsUVp0aOR2IkjlIGRmOYkRfpAbWNjo9XWQhFrkvFOEfcznOJFjkFSWeBXkS8FrJhZQmzo+4+DzVKYcV4sNmk1RaYBDQEYjx79X4KyuM0DwBAa6eJwVAWvnTFbz8bc5rTDbGm8PjP7eQ1WKjXtd5V+7pHS5a9BoajYxyMSOmTvDGjydUOf97LAQDTPVTdd1znjzGP5+fe17YoP+6zOHtuX3fnPH17hpu38cHcQUKd5pV+fEQfQTQ6pHm0/2bUxA1eOKgSzl3HxJGmWxWt8PZc7xqfcQcVNN7LMwmbtSn6Z5ZXRrj7ZXdzxkcuDn4KN7S+VZRPLLeu9bHF/kkGA9LZhLT7Km/zmYbl7cZQlzoAM8bBaZVhaU0QO8hNqRlRNplcYQx/dwxoVFytkk2DXJT5oNAz7Bw0oVN4o8aYlVoMhDq3TrbaZ7w8wYI5rDLqF4Gyr3n6yBXTb7G/1u+aNnFZsPGeW8Sg2TXkqAt7DB3ERyEgkXbPmlVqfBvW2eBoDEEH/sqdSKLamF6pWmW0O/+8d9NhDHvY5Sn9I/OyhUUGNbQhZt5gwPK6eNoPYb0rjpv5/60qI0W6Yu/Z5FrhuTzr3E/vhP4gi4tWftd2pX7fpL/yOANYK8ypo6cezWX8kBejvFaaNpW8/OTN6RLqi4ZbSddcZF2FmW4P0YWaXV3iTMb/pqoghmxhjLYP9oaKzBPlqCd0wI3Ff+H2YwDM4BXV9g45lJ+a9E8Hdg8Y8SyMP1E3knfuZ9e5Nkc8lzVh1W0r7kSN2cdXUyPZ3hQCwHr2LuWnKfZ1Ydt2fcVpDvxmNTd0I+BjdJ4+ytUHH6O4AfWx8VuJWk6ntlvx+UyKHeGFpAtI/pOX7Uv9bGGAG0rgt3u1c1OHt4e5AT9WfX4U6fnB+eOiczHe7EaOI8dTXcgHsFjvPAhCoMDvqXWf7+4sxpyOmYtUqV7nG9MxvMhfY4CKf3NOSS8enf3qsh/6oDNNKg4g4iU3T1iW63YUrEy0bpt5YbPB1jC7S3T5eb/4vM6uKE7tw8J4Z3qbpMFg8uMFsAf7BPyVbU3XxZ0D0dmD5cy0j9hkYzAXFjCt7S8Zc0lNPAYTj0wHaeEmnJfNqZwb1YLL+kxNnVNy0dDa8YcUmx8wdT2s3f5hgZ9x/ElOEJ2XnV4rF9yl/k8TeyCYbKZ0nx4pOLMDEa1jjNqNk7jmFqD5hrmh6V8XGoEqTT2VNEpVCOX0aIjXBpAQuMLWtm1gm4Dbmq9fPfv9sZOTcqzNwrcZdu+MhxMKkqgwaMlKNXcUbg0v5f+JDYKN6qGkGj3d0kgJ/Q5rgKqS2j9fbQJbeSpr4+a6wMRcH+I3NU8+yJkoYmeYOMWV4cSH+0Uv20dSHAVcmFNr3p2cpSyn9mjhJn/30NFP7Ch8ImSjdsnBCzy5s0J/rPsTcLfvJqOd4T7UU2Vcd0TaO7E8dV5lI2V6zQ2bhbxUUmrePs+2Ls2Er8+IfcITRY5MGDEShbyPp9fVZ6t07yw33GITTWP5p5Dgz1HA9d+o57UgBNUNW8HYZDqPbXVY12n8HedZPbWObhKDS0Rs/PdK1WqcoUXOypHiMzk6o+2lWnmlNd6QBHPrlZkz0rK8eCgaeR9j5OeHm1Kc1vQ95VZqZD9DvOQkz4O5qHuCfqmk0K5zJq+J62Jyxdz3A0bJ12PYq70uAcc70yVnB7fEAdF76J6eLFL0A8h3kHvPo5dNm/ZqjAAAAAAAAAAAAA

```
9fsKPiZKPgAAAAAAKfk8Xj07pwFemD+IRVYeOYeOBgs3JGl62esoeSrJyj4AAAAAAAnVV5eoTf/+7me+CZLBbZ
Is+MAqEXL92br1nfWycNy3TqPgg8AAAAAUK2y8nK9/t5svbgmT/m2aLPjADDB4p3H9If/bJRhsPFGXUbBBwAAA
ACoory8QjP+M1evbypRniPW7DgATPTJhjQ9+0lWs2PgFCj4AAAAAACVVFS49c7Hn+u17wqU64g3Ow6AOuCtFQf
1jwW7zI6Bk6DgAwAAAAD4udOevfvJPL2yJlvZjgSz4wCoQ55buFszlu03OwaqQcEHAAAAAJAkeb1efTB3vl5ek
aFjjiSz4wCog/42d5s+2XDU7Bj4CQo+AAAAAIC8Xq8+/Pwrvb50nzKCUsyOA6COMgzpDx9s1OIdWWZHwY9Q8AE
AAABAI+fz+TR7/mLNWLhJB5wtzY4DoI7z+AzdOnOd1hzIMTsK/oeCDwAAAAAaMcMwNGfBEs36cpV2udrLkMXsS
ADqgTK3T9fPWKPdmYVmR4Eo+AAAAACg0TIMQ58v+kbvfrFU21yd5JHN7EgA6pHCMo9uenut8kvdZkdp9Cj4AAA
AAKCR+mb1er332VfaEdxRpQoyOw6AeuhAdol+9+538vkMs6M0ahR8AAAAANAIbdm5R+9+Mk+7HW2VZ4SaHQdAP
bZk1zE9OX+n2TEaNQo+AAAAAGhkjqRn6u3/ztHW8lhlWGLMjgOgAXhlyV7N2ZhmdoxGi4IPAAAAABqRvIJCvfn
BJ/ouy6tDjlSz4wBoQO7+7yZtSysw00ajRMEHAAAAAI1EeXmF/v3hXK3anal9we3NjgOggSl1e3Xzv9cqt7jC7
CiNDgUfAAAAADQCPp9P//18gZas3az9EV3l5ddBADXgSG6pbp+1Xl423ahVfEcHAAAAgEbgq29Xaf7Xy5UW1VX
FPofZcQA0YMv3ZuuRz7abHaNRoeADAAAAgAZu3ebt+u/cBco0aaY0b4TZcQA0Av9atl8frjtidoxGg4IPAAAAA
Bqw/YeO6t8fzlWuN0g7jCZmxwHQiPxl9hbt01ZkdoxGgYIPAAAAABqoEzvmzlb68VztDu4gryxmRwLQiJS6vbr
z/Q3yeH1mR2nwKPgAAAAAoAHyer36YM6X2rXvoHLiuinPw3P3ANS+jUfyNX3RHrNjNHgUfAAAAADQAC1ZuU7fr
FonJbbT7rIws+MAaMReXLxH6w/lmh2jQaPgAwAAAIAGZt/BI/rwi6+kkEitK0s0Ow6ARs7rM3TX+xtUUuEx00q
DRcEHAAAAAAA1IUXGJZn3yubLzirTd0U5ug1/7AJjvQHaJps3ZZnaMBovv9AAAAADQQBiGoY/nLdKWHXtUFN9Fx
91BZkcCAL/31hzWl1szzI7RIFHwAQAAAEADsXL9Jn31zUqFJjTT1tIIs+MAQBX3fbRZxwrLzY7R4AS84PN6vbr
//vvVsmVLBQcHq3Xr1nrooYdkGIZ/jGEYeuCBB5ScnKzg4GANHz5cu3fvrnSdnJwcTZw4UREREYqKitINN9ygo
qKiQMcFAAAAgAbhSHqmPpj7pWx2uzZ6m8iQxexIAFBFdnGF7vlwk9kxGpyAF3xPPPGEXn75Zb3wwgvavn27nnj
iCT355JN6/vnn/WOefPJ8+Xa+88opWrVql0NBQjRw5UmVlZf4xEyd01NatW7VgwQLNnTtXS5cu1c033xzou
AAAAABQ75WVl+vdT75QRtZxFUW3ZWkugDpt0Y4szVx10wYDYrF+PHUugC45JJLlJiYqDfeeMN/bPz48QoDtY
777wjwzCUkpKiP/zhD/q///SVJ+fr4SExM1Y8YMTZgwQdu3b1enTp20Zs0a9e7dW5I0b948XXzxxxTpy5IhSU
lJOm6OgoECRkZHKz89XRETtT03/aGd6rd8TQMMzrn2y2REAAAFm9s+paHgMw9CHn3/lDz//SompLTU3v4k8bKw
BoI4Ld9r11R8GKzHCZXaUBiHg3/XPO+88G27ZIkbdmuEtkbdy4VvXSNOqQRcS8Ie9rkE
n37asWKFZKkFStWqyl/uSdLw4cNltVq1atWqau9bXl6ugoKCSn8AAAAAAA0NBt3bVX875eroS4WK0tjafcA1A
vFJZ79Lc5W82OOWDYA33Be++9VwUFBerQoYNsNpt8Pp/+/OMfa2JEiWMgsk8ee3Lucf2Ghf7Xtzc8EL+xczUEEoKw/zS0v+6zh2jNxcALqw2o1o3EFKp2ClGa
AAAgHrF7fbooy++Un5BoXwRydpVGmJ2pDqjYOV/ZY+IU9zoqXKmtJcjKknkQSJQ9LVW6q4L8v5+7yx3Xme0xIlCZivVlGzD9IXlx7bj8yxBvTwUTM4L
yyusW7/hWed+8I09+phzRKYoeMlnBrftUGuM+fli5S95U2aEtkuGVI7aZ4i+7T/aIEwVGzsLXVLxloSw0l6IGT
1JY5wsqXb94y0Il/PqvAf6KAHXXA59s0YI7B8vlsJkdpV4LeMH3xz/+Uffee68mmTJSUpIkTMzU8nJPzxbKjMzU927d1Ssrq9J1PR6Pjh49qujoaCcbinTJKSsrq9J+jzjfPcKcKwuJdjQNjf2SyDAFIiaPrwsLZW46XXxKWk9tcGNddy4qxjm8F4avTSC8cVjOmrc+lXbYisdRygSDDiwq8kpD+44PPm+NrjEnMPB2GPGeuDwK9CGLHPXUdEhqqrcNTLvLEGaH2nwXKmfbsp2hMcDvmQqajRUmLqmEIUcCNetRtwp8Jdpb5lrAhDhn+K+BfsUmTNXx/IE4MZNjRHJO9z9u37Kk7rMiFXdNqW7T5hFMBFs6X9rKGW1/t75HBmRtYOtGkMxZBamfzmZhq4oO9IdJq3iVEydO/ppcyoKwtZT0gQ3IPbXZKqNuJfRiPvNbR8P/74jY1DVCWiiAd+oZ5TkSxY73zxz3uJhCPqrqHrHSVmGmdtqG0SUVmwb9MT0Mac9MXsL0VY+lqKIKIbwQIFMJbU6wVbcT6iUZoROYeG+YIkgWa9SX6K4zXAj8hmbeg6l+iSA9K7y9vPmD72YvHrM9ulmK5qWC9uGnbkXgG2eS3zl+qPAYyvVoRaU1evdSyWQQAjiUkC9p1/ZklZi0O09PSqx2o4gqzVQ2bX1jCBYSwcwz9ZjpaLqoeqr6mTxvsrFLfGIoT0T+Ta6SRZH/fmgOS8VdzZO7/mF1e+4DpF
AAAgHrF7fbooy++Un5BoXwRydpVGmJ2pDqjYOV/ZY+IU9zoqXKmtJcjKknBLXvKEX1igoknN00VaTsVM+I20ZP
```

2tz3xWKrgNv0U0Wfs/+7dSuVHt6twwxdyNesqSYoaOFFRAyf6X5f37Sy5WnSXxWpT/or3lXL9iyrds1rZnz2j5
MnP/fJAQD300GfbdH670DntPFP0TAV8iS4AAAAAoGas3bRNy9dsULMmydpcEqFyNtaowhYWLUdcs0rHHLGp8hY
c83/uTGqjl0ueV+rU99V0yr+VePk0+UoLZY+qflNHSbKFRvtn633PW5znn9VnC4mQrDY54lJPee8fc2cfVvG2x
Yo6/2qVHdosV9MusoVEKqTD+arI3CtfeckZvHOg/juYXaLXv9lvdox6iYIPAAAAAOqB/MIizZ6/SBarRfaQCGO
pDjU7Up3kbNJJ7pwjlY65c45Wu3mF1RkqW0ik3DlHVZGxRyFtT76po7NJB5Ud3FDpWNmB7+RscmLnT4vNIWdSW
3lyjla5t62aexuGoez5Lyp66I2yBgVLhk+Gz3Pi5Pf/+79ZgUBj8uLiPTpeVG52jHqHgg8AAAAA6oEvlyzX3oN
H1KJpitYVhstr8OtcdSL6XKrytJ3KX/GB3LlpKt72tYo2zlNYz9H+McU7vlXZoU1y52WoZPdKZb5/v0La9lNwy
57+McfnPq3cJTP8n4f3+pVK969XweqP5M4+rLxvZ6o8Y4/Ce17yw737jlPx9m9UuGGe3LlpKlg3R6V7Viu858V
VchZtnC9bcIRC2pwoFZ1NOqrs4CaVH92hgjWfyBHbTFYXS7DR+JRUePXy13vNjlHv8DRWAAAAAKjjjqRnavHyN
UqIjVGh4dLukhCzI9VZzuR2ir/sz8pb8pbylr0re2SioofepLDOF/jHeItylLvo9RNLbMOiFdZ5qCIHTKhOHU/
BMcnyQ4nqatpRcWP+qLxv/q3cpW/LEZ2ihHF/VlB8C/+YkHbnKXbkbcpf+R/lLnxV9pgmir/sT3I17Vzp2t7iX
OWv+EBJVz/1Q+6U9oo49zJl/fdvsoZEKm7OnQH+ygD1xzsrD+rmQa2UGOE6/WBIkiyGYZzFYOTrroKCAkVGRio
/P18RERG1fv+PdqbX+jOBNDzj2iebHQEAEGBm/5yK+scwDL35/mx9uXSFOrdvowU5sTpUzi+9ABq2a/o11ONju
5gdo95gTjcAAAAA1GG79x/SivWblJKUoMwKJ+UegEbh/TWHdTSv1OwY9QYFHwAAAADUUT6fT18s/lbFJaWKjoz
Q6gJmfQJoHCq8Pk3/arfZMeoNCj4AAAAAqKM2bt+l9Vt2KDUlSQfKgpXlDjI7EgDUmg/XH9HB7GKzY9QLFHwAA
AAAUAe53R59sehb+bxehYWGam1huNmRAKBWeXyGnmMW389CwQcAAAAAddDqDZu1bfc+NWuarANlLuV7HGZHAoB
aN3vDUe3JKjQ7Rp1HwQcAAAAAdUxJaZm++HqZHHa7gl0ubSwKMzsSAJjCZ0j/YBbfaVHwAQAAAEAd8+2a77T3w
GGlNknSkTKnjvPsPQCN2Oeb07U9vcDsGHUaBR8AAAAA1CG5+QWa//VyhYeFKsjhYPYegEbPMKSXv95rdow6jYI
PAAAAAOqQ5Ws36GhGppokJ3Sirwq0CqfZkQDAdF9sSVdWQZnZMeosCj4AAAAAqCMKi4q1ePkaUaEy2azaTOz9
wBAkuT2Gnpn5UGzY9RZFHwAAAAAUEes+m6zjmZkKTkxXoUemw6UucyOBAB1xqzVh1Tu8Zodo06i4AMAAACAOqC
OrEyLlq1WSEiw7DabthaHypDF7FgAUGccL6rQnI3pZseokyj4AAAAAKAOWLtxmw4cSVOTpARV+CzaWRJidiQA
HNmLN9vdoQ6iYIPAAAAAEzmdnu0cPlqBQUFKcjh0I6SELkNfl0DgJ/acrRAaw/kmB2jzuG/GAAAAABgsk07dmv
v/kNqkpQgw5C2F4eaHQkA6qw3lx8wOOKdQ8EHAAAAACYyDENLVq6VYRgKdjmVXhGkQq/d7FgAUGfN35Kh9PxSs
2PUKRR8AAAAAGCiPQc0a+vOPUpJSpAk7eLZewBwSh6foX+vOGh2jDqFgg8AAAAATLRszQaVlJYpIjxMFT6LDpS
5zI4EAHXee2sOq8ztNTtGnUHBBwAAAAmyTiWrVUbNishLlaStK8OWB421wCA08oprtD8rRlmx6gz+C8HAAAAA
Jhk3aZtys3LV1xMlCSW5wLAmZj93VGzI9QZFHwAAAAAYAK326MV4RSqsvL6Fo8tFoR6bstxBZscCgHrjm93HlVN
cYXaMOoGCDwAAAABMsGPvfh1Ky1BifIwkaTez9wDgjHh8hj7blGZ2jDqBgg8AAAAATPDdlh3yeDwKdrnkMyj4A
OCX+GQDBZ9EwQcAAAAAtS6voFBrN21VbHSUJOlouVM1Ppu5oQCgHlp3KFdHckvMjmE6Cj4AAAAAqGWbt+/WsZw
8NtcAgLNkGNKnG5nFR8EHAAAAALXIMAyt/G6THa7bDabyn0WHSpzmRO0LAOqtT76j4KPgAwAAAIBad0hounbt0
6jEuB0baxwuc8kri8mpAKD+2plZqB0ZBWbHMBUFHwAAAADUoo3bdmpwZBWbHMBUFHwAAAADUoo3bdqmwqFgR4WGSxOw9AAiAxr7ZBgUfAAAAANS
S8vIKLV+3URHhYbJYLPIZ0pFyp9mxAKDe+3RDmgzDMDuGaSj4AAAAAKCWbN+zX0czspQQFytJSq8IUoXBr2UAc
La05pVq/aFcs20Yhv+SAAAAAEAt+W7LDnm9OFH+7zmNmRzANBR8AAAAAA1ILSsjJt2rFLOVGR/mM
UfAAQ0Et3UfABAAAAAGrQngOHdTwnTzFREZKkHLddhV67yakAoOHYfDRfOcUVZscwBQUfAAAAANSCnXsPy0Pxy
BnE8lwAqAk+Q/pmd+OcxUfBBwAAAAA1zOv1asPWnQoLC/Ufo+ADgMBbuuu42RFMQcEHAAAAADXsONEMpWcdU+z
/nr9X6rXqmNthcioAaHiYwQcAAAAAqBG79h9UcWmZQkOCJUmHyp0yZDE5FQA0PFmF5dqpGjFpHwQcAAAAAN
cgwDG3ctkuuoCBZLCdKvSMszwWAGtMYd9Ol4AMAAACAGnQsJ1f7Dx1RTHSk/1hmRZCJiQCgYVvaCJfpUvABAAA
AQA3ave+g8gqLFBURLkkq9NhU4r0ZnAoAGq41B3JVUuEx0OatouADAAAAgBq0Oddde2SwWWa0nfv1i9h4A1KwKj
O8r92WbHaNWUfABAAAAQA0pKi7R117FR3F8lwAqE0r9+WYHaFWUfABAAAAQA05cCRNOXkFio6K8B+j4AOAmrf
xcJ7ZEWoVBR8AAAAA1JAj6ZnyGDwDwKcjgkSRU+i3I9dpNTAUDDt+Vovnw+w+wYtYaCDwAAAABqyJ4Dh+X4knS
VkVQTJkMTERADQ0xRVe7T1WZHaMWkPBBwAAAAA1oLy8QnsPH1ZEWKj/GMtzAaD2bDySb3aEWkPBBwAAAAA14Gh
GlvILihQRHuY/RsEHALVn05E8syPUGgo+AAAAAKgBRzIyVVVpermCXU5LkM6RjbsdpXgUACBRm8AEAAAAAzsqBw
2myyCKL5cQz93LcDrkNfgUDgNqyPb1Abq/P7Bi1gv+6AAAAAECA+Xw+7di7X+FhI5jx5m9BwC1qsLj086MQrN
j1AoKPgAAAAAIsMzjOcr0yau0wUaex25iIgBonDY2kufwUfABAAAAQIAdSc9UUmpwkIp+ADATJsON47n8FHwA
QAAAECAHUnPkM9nyGb74VcuCj4AqH3M4AMAAAAA/CI79hxQcLDT/7nHZ1GR12ZiIgBonPYdK5bS8gdo8ZR8AE
AAABAAJWVlys965jCQn7YYPDPa5NkMS8UADRSFV6f0vJLJzY5R4yj4AAAAACCAjufmqbikVKFnfS8hAHGOGiAJRoNeEYSCGvSrXE1DcpZkRKuMcu5sdMGqbLOWLZNFARHcjFPQnTYELEYs9UJJjfd9lKA6AOtc0ycUPgdSVBTbN+AC/SX6ZRwaENSf72KDZqGTqG9mApZ2wd+3i2DxIyJuwVG3FjpLQoKXlP7eV1VYyfnEQy5b1i+0NDbFbw9r2V6LzgBUzwS/vh0B8u9VLwTnErRmyCT0bt6ZVLeh9rvPmQb4HRLUlRTMYA0BWk6j7kdNmvA9tOmTfexo/E8Ul94vf6

k6cqIiICEVFRemGG25QUVFRTcQFAAAAgIA4npMrn88nh/2HWXsUfABgvkM5DXuZbsD/S5Obm6sBAwbI4XDoiy+
+OLZt2/TOOO8rOjraP+bJJ5/U9OnT9corr2jVqlUKDQ3VyJEjVVZW5h8zceJEbd26VQsWLNDcuXO1dOlS3Xzzz
YGOCwAAAAABczwnT8ZPjpV6baZkAQD8oKFvtBHwhOE88cQTSk1N1Ztvvuk/1rJlS/8/G4ahZ599Vn/5y1906aW
XSpLefvttJSYmavbs2ZowYYK2b9+uefPfxmac2aNerdu7ck6fnnn9fFF1+sv//970pJSQlObAAAAAA4aOczsuROO
CodYwYfAJgvs6Ds9IPqsYD/l+bTTz9V79699Zvf/EYJCQnqOaOHXnvtNf/5/fv3KyMjQ8OHD/cfi4yMVN++fbV
ixQpJOooVKxQVFeUv9yRp+PDhslqtWrVqVaAjAwAAAEBAHMvOkdMZVOlYCTP4AMBO+aVusyPUqIAXfPv27dPLL
7+stm3bav78+brlv1v/3m1/9Tm+99ZYkKSMjQ5KUmJhY6XWJiYn+cxkZGUpISKh03m63KyYmxjSMkymHhUmxj/mp8rLy1VQUFD
pAwAAAABqi8fjUWFRiYKYwQcAdU5DL/gCvkTX5/Opd+/eevTRRyVJPXr00/TKK69oQRJgb6d32OPPaa//
e1vNXZ9AAAAADiV4tIyVXg8cv1kBh8FHwCYr6EXfAH/LO1ycrI6Idepu6I6dNAhSVJSUpIkKTMzs9KY9KYzMx
M/7mkpCRlZWVVOu/xeJSTk+Mf81P33Xef8vPz/R+HDx8OyPwBaQOnySij4z
siAAQOOc+fOSsd27dql5s2bSzqx4UZSUpIWLlzoP19QUKBVq1apf//+kqT+/fsrLy9P69at849ZtGiRfD6f+vb
tW+19nU6nWIIiKnOAAAAAQGOpLilVRYVbDscPC6XKfFYZspiYCgAgNfwZfAFfonnvnzXPPeOPPPqrLL79cq
1ev1quuvvvqpXX31VkmSxWDR16lQ9/PDDatu2rVq2bKn7779fKSkpuHbzGGjRqlm266Sa+88orcbremTJm
iCRMmsIMuAAAAgDqpuLRUbo+n0gw+lucCQN1QWOaWz2fIam4Yf+kSIam2fIam2Yf+kS8IKvT58++vjjj3Xfffdp2rRpatmypZ599
1lNnDjRP+buu9Wc/XGxbr75ZuXl5WngwgI2N2XWo/P+s8XRRs2TFarVePHj9fO6dMDHRcAAAAAAqK
4uFTSiUkN3ytlB1OAqBN8hlRYRY5lFkiOP0g+uhgBd8knTJJJZfokksuAqBN8hlRY5lFkiOP0g+uhgBd8knTJJZfokksuOel5i8WiadOmadqOaScdExMToomZtVEP
AAAAAAIuKKSkirHKoyGOVMEAOqjvNKKBlvwMV8cAAAAAAKgpLSsyjEfBR8A1BkN+T18FHwAAAAAEAB5+QWyGivBDTYkycsMPgCoM/KYwMcqAAAAAOJX8wkI5HJUc84SXQCo08oqvGZZ
DEUfAAAAAAAQAKVl5bLbKu+a23B/rdSdrGGZHqDEUfAAAAABwlnw+n7xenyzmGeNGxSvGZXHqDEUfAAAAABwlnw+n7xenyzWyjP2mMEHAHWH1OfBBwAAAAA
4CZ/PJ59hyGr5SclnmhzuAQFU+ZvABAAAAAE7G6/XJGzW4yiy4z+ACgzmAGHwAAAADgpLw+n+eXCIdhdzE+Q5afzOBjF
1OAqDsacL8n++mHAAAAAABOxev1yjAMWX/2W37zcvxiyjAMWX/6DD6T8gCBNNCxS8/aXzA7BvCz+QyfDJ+hqMhwWa0/zG1zO+6S1NK
8YDWIgg8AAAAAzpL3f8/g++kMPpbooiG4zvGV4nzHzY4BnBmrpMLcSoccRok5WWoBS3QBAAAA4CydeAZf1U02L
JYGvB4MjUK4pUQDfGvMjgEEhsVmdoIaQ8EHAAAAAGfJ6/PK8PlksVb+FctOwYd6bpLzW7lUbnYMIDAsDXdWNQU
fAAAAAJwlr/fEEt2fzuBzUPChnhtd9pnZEYDAsTKDDwAAAABwEt9vsvHTZ/Axgw/12TkVm9zVRdczsGEDgWBvuV
hQUfAAAAABwlnw+n3y+qgUfM/hQn/O++huzIwCBFRRmdoIaQ8EHAAAAAGfJYrHIYrHIUOVCjxl8qK9cqtAA3zq
zYwCB5YoOOGNoeADAAAAgLPkcDhks1rl8+mVNWuJXEap2TGAwKLgAwCcTJDDLpvK9PCj671
XeSVwB126+tS8yOAAQeBR8AAAAA4GQcdrvsNluVGXyGNNpLHpWEoHHaaHPZZ/NBPBwAAAABnKeh/S3S3
9Pm+14zyDD/XRrc75spx+GFD/MIMPAAAAAHAydodDNrtNXm/VGXwepSAwAAAAASEM8hRZZONIKtPEiUf6o8rbYvNjgDU4jAa8PFe
i4AMAAACAgHA5nVWW6EpSFBtZmh+hLHyoJ25zfSmr+HCwAAAAICAcLmcVZ7BJOkhFHyoJ25zfSmr+
POKBio42uwENYqCDwAAAACwBVU9Rl8khRi81YzGqhbLPLpQi03OwZQcyKbmp2gRlHwAQAAAEAABLtc1Rd8zOB
DPTAuaK2ifHlmxwBqTlQzsxPUKAo+AAAAAAiAsNBgGczgQz11tWOR2RGAmkXBBwAAAAA4nbDQkGqPh1LwoY5Ls
uTpHO9Ws2MANSsy1ewENYqCDwAAAACIPwkBV8YBR/quFtdX8om/pyigWMGHwAAAADgdMJCQ2WxWOT7yTLdcAo
+1HEXW9hcAw2c1SGFJ5udokZR8AEAAABAAISHhijIYVdFhVs222/LhtsTdvC/GNvd8GxSvC/L7BhAzYpIkawNtw
O8OAAAAAGpJWFiIgoIcVRgoIccrsrF3x2ixRsZRYf6qbrgr4y0wJQ8xr48lyJgg+oltfr1bvPPalbh/XVld1a6bYL++s
/L/1DhmH4x5QWF+u1aX/STYN76cpurfT7OYM1/723T3vt4oJ8vTbtPt1wfnddWFpowcqHVLFp7RvT9542Vdd
15XXXdeV336r1cqXX/XxvX647iR8nr4m2IAAIDaFBEWKmeQQ2UVFVXO8Rw+1EvRlmL18K9xwQ8xpBwWc3OBwc3Ow
QF81+7UXNf/ct3fH4cOpt0157t2zUC3+6UyFh4Rp97Y2ShIS1Wfoy
Gqv666oON+un6DI2Dj98blXFZOQrGNpRxQaEfGz731g5za99/xT+tMrb8swDD3220nqNmCwmrfvKK/Ho1cfvEe
/nfaUbHb+9QYAAKhNLqdTkeFhysrOrXIu7RMXeQCamAk7vFtVAOw336gUB918B30JUo+IBq7fxurfoMG6leQ
4ZLkhKapuqbz2Zrz+YNP4zZsFZDxv5GXfqeJOkaccXVWvD+v7V704aTFnyLPnpPRf5evTdT2V3OPnpPZXPpN7H92
3R83bd1LXfgMlSc3bd9TR/XvUvOO0eqC8FAAAAzkBCXKJMkwpOpFSy1LJO
PO4oN6LbW12ghrHEl2gGu179NbmFd8qbQ9BHVu1Y/1q9q9Rg9Icx3XtrzaIvlZq14ZGMdmwKZSXsp97fN/gk4o/1zZyPNPXvLyq1TXvt37FVdMWwtWHnlMqUd2KduAwa
f9LprFn2p9t176bVpf9L1A87RQHT6Uf2KdjaUeUdfSIOg7sU702HZRx6IAWffS+r
vr9PTXxJQEAAMDPkBgXqwp31RlgDH2XUnxpZscAakdiF7MT1DhmzmKSopLtTvLh4kq8Omn9e
rq6beq0FjxvnH3Hj/w3rl/rt18+BestntslisuvWM5+5THP/5TH/+5zvKOLRfr/7tT/J63
Lp8yh9+1r2btm6rq+68V9OunyBJmnjXfWauq0ev05yXfPHHP2vDt1/r/r/ReflT1u13V/euiUeQAAABBYUZHhslR
zPMbBMkjULTc6F0gOhKNgc0pxbs1PUKAo+ADqrH8i0/1zZyPNPXvLyq1TXvt37FVdMWwtWHnlMqUd2KduAwa+7
39p18Z1uvelGYpv0TWO/Tb1qzUa9P+p0iERHU7b1C11zV8hiJjY088I89mU+su5yg7MOOf/Otlf8H3c+49sK1Gjn

hWv91F3/8gYJDw9S+e2/dcdH5euI/nys7I13/uOtWvbxwpRxBzhr+igEAAECSosLDZBiGDMOQxfJD1Rdi88l19
arMZzMxHXBCiMrV37fO7BhA7YhvL9kafv3V8N8h8Au8/dRDuuymKRo4eqykE8+5O552RB+9+rwuuOxylZeVata
zj+vu59/wPyuvRftOOrBjqz791ysnLfii4xNkc9hls/3wg13T1m2VdyxL7ooKOYKCTnvvnyrIzdYHLz6jh9/5S
Ls3rVdKi1b+D6/HrbT9+9S8fcfAfoEAAABQreioSAUFBam8wi2Xs/KmGrEOt46WU/DBfNe7vpbLKDM7BlA7ks4
xOOGt4Bl8QDXKS8tksVb+18NqtcnwnXgCrdfjkcftPskY3Omv26FnH2UcPCDfj8akHdin6PhEOYKCfta9f+rNx
x7UmEk3KTYpRT6vT17PD8s/vF6vfD7m3QMAANSWhLgYhYYEq7ik6o4aPIcPdcV42xKzIwC1J6nhP39PYgYfUK3
eF1yoD1+ZrvjkJieWyW7fojkz/qmh40889y4kLFyd+/TX2089pCCnS/FNmmrr6hVa8sl/Nenev/qvM/2e3ykmI
UlX/+FPkqSRV16rL2a+qX89cr8uvvp6pR/cr4/+OVOXX3PDz773j21ctkTpB/bpjsefkyS16dpNR/ft1fqli3Q
8PU1Wq1UpLRv+bkFoXLZs2WJ2BAANRJcujeMHftSuiLBQxUZFKvN4jmKjIyud4zl8qAvOsR1SC+8Bs2MAtacRb
LAhUfAB1brxLw/r3elP6tVp96kgO1vRCYm68Ipr9Jvb7vSPufOZlzXzmUf13B+nqCg/T3EpTXTl1HsqPRvveNp
RWSw/zMaLS26i+1+fpTcff1B3XTpcMYlJGn3NjRp70+1ndG9JKi8r1esP/Vl3/eMVWf834y82KUU3/OUhvfinO
2UPCtIdjz8npyu4pr5MAAAA+AmLxaJWzZpo/+GjVc5R8KEuuNU5X5aTLzoCGp5GMoPPYhhG9ev+6rmCggJFRkY
qPz9fERERtX7/j3am1/o9ATQ849onmx2hzmEGH4BAMWsGn9k/p6LmzV+yXG998Kk6ta+8ksJnSDPSk+Wrdp9do
OY55NGG4NsUahSZHQWoHZGpOp2N4/cHnsEHAAAAAAGUEBcji9Uir7fyNCmrRYriOXwwOdWuZZR7aFwayfJciYI
PAAAAAAIqMS5WIcEulZRW3WgjlmW6MNEE62KzIwC1q5Esz5Uo+AAAAAAgoOJjohUWGlLtTrpxQRUmJAKkNtZMt
fXtNjsGULtS+5qdoNZQ8AEAAABAADkcdqUmJ6qomoIvmYIPJrnVNV9WNchH8APVs9ikZv3MTlFrKPgAAAAAIMB
apDZRRUXVMi/a7pHL6jUhERozi3wabqwOOwZQu5K7Sc5ws1PUGgo+AAAAAAiwxLgYSZJhVJ4xZbEwiw+17zfO1
YoO8syOAdSuFgPNTlCrKPgAAAAAIMAS42MV5HCovJpZfMnOchMSoTGbaF9kdgSg9lHwAQAAAADORlJ8nMLDQlV
QVFzlXAoz+FCLUqw56uLdbnYMoHZZbFKz/manqFUUfAAAAAAQYGGhIWrZrInyCwqrnItyeBTCc/hQS25zLpBN/
HlDI5PUVXJFmJ2iVlHwAQAAAEAN6NimpSoq3NWeY5kuastFlmVmRwBqXyNbnitR8AEAAABAjWjeNEUOh7365/C
xTBe14OKgjYr1HTc7BlD7KPgAAAAAAIHQvEmyoiLCq12myww+1IbJjgVmRwBqn8UqNT/P7BS1joIPAAAAAGpAa
EiwWjdPVV5BUZVzkXavQnkOH2pQjKVIPbybzY4B1L7k7pIrOuwUtY6CDwAAAABqSIc2LeV2e6o9xyw+1KTfur6
SQ9U/AxJoODpcbHYCU1DwAQAAAEANad4kWUEOu8rKq5Z5zVxlJiRCY/EryzdmRwDMOeESsxOYgoIPAAAAAGpI8
6bJio6KUH41y3RTneWyWXwmpEJDN8ixQOm+dLNjALUvprWUONHsFKag4AMAAACAGhLscqlNi2bKq2ajDYfVUFO
W6aIG3MjmGmisOow2O4FpKPgAAAAAoAa1b91CHo9HhmFUOdeSZboIsDBLmfoa682OAZijkS7PlSS72QEAADgTu
xyxZkcAOEBOMTsAGo3mTZLldAaprLxcwS5XpXPNXGWyypBPFpPSoaG5OblYToOZoWiEwhKlpn3MTmEaZvABAAA
AQA1qOTRFCbExys7Nr3IuyGqoCctOEUCXWZeaHQEwR7tRkrXx1lyN95ODAAAAQCOICnKoV9eO1W6OIUktgOtrO
REaqp72A2ruO2h2DMAcjXh5rkTBBwAAAAA1rlO71rLbbSqvqKhyrvn/lukCZ+u3QfPNjgCYIyhcajXY7BSmouA
DAAAAgBrWtmUzJcbF6HhOXpVzTquhFJbp4iw5VaGBxlqzYwDmaDtcsjvNTmEqCj4AAAAAqGEup1M9u3ZUfkFht
edbsJsuztK1rmUKMYrNjgGYo9uVZicwXYOXfI8//rgsFoumTp3qP1ZWVqbbb79dsbGxCgsLO/jx45WZmVnpdYc
OHdLoOaMVEhKihIQE/fGPf5TH46npuAAAAABQIzq3ayOb1VrtMtOWrjJZWKaLs3C5bbHZEQBzhCdLbYabncJON
VrwrVmzRv/85z91zjnnVDp+fDp5513as6c0frPf/6jJUuWKCOtTePGjfO/f93q9Gj16tCoqKrR8+XK9/e67+
44IGajAsAAAAANaZdq2aKP8luui6bj2W6+MXa2dLV2rvX7Bi0ObpNkKw2s1OYrsYKvqIk2cOFGvvfaaoqOj/
cfz8/P1r3xvfPWPsPVPp0qA3xxxRtbfHe6fUVcR3B5hwYqbPCx19ZdZzd2vfPOOmmuugiPfTQQ3rxxRwvsxDRc
fdgEAAABAXRfscqlHlw7Kyy+7Kyy+o9nyHkJJaToSG4jbnfDZqQePV/WqzE9QJNVbw3X777Ro9erSGD688TXLdunVyu
92VjnfoEHNmjXAzBHaj8pro3ZKeqEGin43nvvPa1fv16pqa/Rj2ncSKUpeOAUFBZUpeOJi
YnKyMjwj/lxxfLxbu7n89s+cUVdCwnj4mSiqL3I/lXbmemm6DM8YeqNbtY4J3EsBgk//cO7E1JPlS27hmp0dO5TQ+oWu
qB7P3vhfwgu/w4cP6/e9/r5kTZ8rlcgVq3pMaO3as8vPz/dvWrVsDNndVePLJJ/Xiyy/rnXfeUVhYWKf62HnDRv+1QEJDg49d66vCc/mOy8zPN3+NttT6SKr8lrTYalvcWPc3DpK/0Njd/bzBsObPAZ0Wzl3nwgr+z6UOzBsKBvGcq09P+9XmxLv/fIsPEvIlmhmZS/v5DTRSO4hU4b4ZXrFHhLqXmEFX5PbNWtgNgWNp/5J9Bfz6RKhg2F7YvTAp+zdt2sVxE6Hdov1/UfebWtzw46Mt3XtrBLEyeGoFq8k5qxypAzMm8e+4rP8Cf8KdWFfdaypF/VobqBPwHUWFv7b1+DGlZdjvOFudZsrQ33g9mWGnu3ZP7Ih2P1tqYWJMMkJcwBc2KdBIvTOmLJW1ULKKTItQMiz6iVSqZpVcw5QTqt99YW9gJeBRh9fcDVySIzsGg6jXoi5XeRSWOfv7vm2kDqJIivmL1H2g2U1TllX6Zw9YhgEqaaO8ZtK8yWx1v1U1PXHmYcF3z3dUbZqJEqn3Wp83ivX0H7Wz2vXZ4Y/ZryJx+wTKzeVs3qf8X6ul3IU8Fqkjk5KoKoYV/RU5mdJu6hqjlP6jrtwyaP8QTEjP8+r07u2Abuvfo6aH9AQHmBg5v8N6Zo7n/qftIKTJ0jo8rTo3oowhcPfT5+0b0F7Zd3UoVv9dd7p0Pk+/5tbvPEsCsOcVtHhfLhu1jHm5VPlKnW/u/0Dv1Ud5Ctwbxp2cN3BQp+a0uQ7rxp7Zp9nZdN65Ct+e8jUAQPBcvbBuAcHlN3+0vV96Gm9rq3CQTfjFhw5r5c1EeX+dNwBVkl7Ia1JuzR+hntbqCltAH5dWtMCcp/DfFC+9t4r9lhgb1p8tF0A7Qc1n5yWFLBtLwcuR8u8TuT6z/8Hxvc8bM4dDdH5Ytmz5NfrYhh/e9g8CjyTzv7+1/Obx9kk+VfRuq1mKh

Ha82vMdQ0pqORHqi1tdX8kuj9kxgNpnc0rnsjy3OhR8AAAAAGACpzNIA8/tqaKiYhmGUeV8U1e5Yh1VZ/cBl1i
+NTsCYI6uv5HCE81OUSdR8AEAAACASXp06aDoyAhl5+ZXe757WFEtJ0Jdd4FjmxJ9GWbHAExgkc67w+wQdRYFH
wAAAACYJDkhTt07t1fW8exqz7dwlSna7q7lVKjLbnAsMDsCYI42w6WEDmanqLMo+AAAAADARH17nCObzabSsvI
q5ywWqRuz+PA/4ZYSnev7zuwYgDmYvXdKFHwAAAAAYKL07VqrZWqK0rOOVXu+VXCpImxsqADpZtdiBYnnMqIRS
u4mtRpsdoo6jYIPAAAAAEzkcNg1dEBflZWVy+2uWuRZLVK38EITkqGuGWtZanYEwBzn/c7sBHUeBR8AAAAAmOz
c7l3UvGmKjmZkVXu+bXCpwpjF16j1se9Vqu+w2TGA2hfXXuo8zuwUdR4FHwAAAACYLCTYpWEDzlVRSYk8nupn8
Z3Ds/gatVuCvjQ7AmCOC+6TrNRXp8NXCAAAAADqgH49z1GzlCSlZVb/LL72ISUKsXprORXqApcqNMBYZ3YMoPY
ldpU6jTU7Rb1AwQcAAAAAdUBYaIiGDjhXhUXF8nqrFnk2i9SVWXyN0iTXUgUbJWbHAGrfBX86sZ04TouCDwAAA
ADqiP69uiklMV7pWcerPd+RWXyN0uXWr82OUKe8vKZC57xcpIjHChTxWIH6v1GsL3a7/efLPIZu/6xUsU8WKuz
RAo3/oESZRb5TXjOzyKfJs0uV8nShQh4p0Kh3irU7u/p/1wzD0EUzi2X5W4Fm7/jhvjmlhsa8W6KwRwvU459F+
i698utv/6xUTy8vP4t33sg06S11uNjsFPUGBR8AAAAA1BGR4WG64Lw+yssvkNdbtZCwWw31jigwIRnM0sl2VK1
8+8yOUac0jbDo8eFOrbs5VGtvDtXQFjZd+l6ptmadKNTunFem0bs8+s9vgrVkcqjSCg2N+6D0pNczDENj3y/Vv
lyfPpkQou9uCVXzSKuG/7tExRVGlfHPrqyQRVVnlT2ytFyF5YbW3xKqIc3tumnOD/dcecSjVUe9mtovKABfgUb
iwr+ZnaBeoeADAAAAgDrkvN7dlRgfq8xj1c/iaxtcqlhHRS2ngll+65xfTZXUuI1p79DFbR1qG2tTu1ibHhnmU
liQtPKIV/llht74zq1nRro0tKVdvVJsevNSl5Yf9mrlkep3ot6d49PKI169PNqlPk1sah9n08uXuFTqlt7d4q4
0dk0GV0+vqNC/LnVVuc724z5N60JQu1ibbu7l0PbjJ0p6t9fQb+eW6ZVLgmWz8v/mz9J2hNRioNkp6hUKPgAAA
ACoQ2KiInVB/z7KycuXz1d1Fp/FIvVnFl+jYJdXQ41VZseo07w+Q+9tcavYLfVPtWldu1dunzS8ld0/pk0cTc0
iLVpxuPolt+X/6/1c9h/KN6vFIqdd+vbQD68pcRu66sNSvXixS0lhVeuUbolWLdrvkcdnaP5ej85JtEmSnlxWo
SEt7OqdYgvEW274LFZp+INmp6h3KPgAAAAAoI4Z0KeH4mNjlHU8p9rzSc4KtXSdfMkhGoYrnSsUZhSaHaNO2pz
pVdijBXI+XKjfzi3Vx1cEq1O8TRlFhoJsUpSr8ky5xFCLMoqqLreVpA5xVjWLtOi+hWXKLTVU4TX0xLfl0lJgK
P1Hz+67c16Zzku16dIOjmqvc+9Ap+xWqfX0In28w6M3fuXS7myv3tro1v2DgvTbuaVq9VyhLv9PifLLqs8CSd2
ulBI7m52i3qHgAwAAAIA6Jj42WoP79dLxnNxqn8UnSedGFMgmSoKG7Er7YrMj1Fnt46za8NswrboxVLf2DtKk2
WXaduyXbUDjsFn0OeUh2pXtU8yThQp5pFCLD3h0URu7vl9R++l0txYd80rZUVWX5n4v0mXRrPEh0jg1XEsmh6p
TvE23zC3TUxc6NX0zW/tyfdo5JUwhDoumLWGzjWo5QqUL/mx2inrJfvohAAAAAIDaNnTAuVqxbp0OZGSqeZPkK
ufD7V51CSvSxqJwE9KhprWwHlMH706zY9RZQTaL2sScaN96pdi0Js2r51ZW6IouDlV4pbwyo9IsvsxiQ0lhJ3/
+Xa8Umzb8Nkz5ZSdm8MWHWtX39SL1Tj6xrHbRfq/25vgU9XjlGZXjPyjV+c0q9PXk0CrXfP07CkW5LLq0g0Pj3
i/R2A40OWwW/aaTXQ98TcFXrcF3S5FNzE5RLzGDDwAAAADqoJioSF08dKCKiktUXlH9phrdw4oUYv1ls5ZQt93
m+lJWVT97E1X5DKncK/VKtslhlRbu+2FDjZ3HvTqUb6h/6umfgRfpsig+1Krd2V6tTfP5l+PeOzBIm24N1Ybf/
vAhSf8Y6dSblwZXuc6xYp+mLS3X8xedmPHnNU5stiFJbp90kom5jVt8R6n/7WanqLeYwQcAAAAAddSAPt21bOO
G7TlwSO1atahy3mE11DuiQEvzoms/HGrUCC03OOKddd9XZbqorV3NIq0qLDc0a7NbXx/wav7VTkW6LLqhh0N3f
VmmmGCLIpwW3fFFmfo3talf0x9tvPFCkR4b5tRlHU8UeP/Z6lZ8qEXNIq3an0nV7+eVaWwHu0a0PvGapDCrksK
qZmkWaVXL6Kpzp6bOL9Mf+jvVJOLEuQGpNv17k1sjWtv16jq3BvyMsrHRGf20ZKv++YY4PQo+AAAAAKijXE6nL
hk+SM+9PlOFRcUKD6u6DLBtcKm2Focq2x1kQkLUhLFBaxXlyzU7Rp2VVWzo2o9LlV5kKNJp0TmJVs2/0kQX/q+
M+8col6zzyzT+gxKVe6WRre16aXTlZ+ftzPYpv/yHZ1imF/l015cVyiwylBxu0bXnOHT/YOcvyjd/j0d7cnz69
2U/lFVTzg3S2jSv+r5erHOb2PTXISd/ll+jdM4EqcUAs1PUaxbDMBrkU1kLCgoUGRmp/Px8RURE1Pr9P9qZXuv
3BNDwjGtf9Xk7jR3fXwEEilnfY83+ORX1j8/n00tvf6Bv13ynzu1ay2Kp+hyxjPIgzc20MyEdasLHoU+oh3ej2
TGAWmG4ImWZsk4Kizc7Sr3GM/gAAAAAoA6zWq26ZNggRUWE61h09b06kpwVahdSXMvJUBMSLfnq6t1qdgyg1li
G3k+5FwAUfAAAAABQx7VITdEF/Xsr61i2vCd5On+/iAKF2jzVnkP98VvXAtnF/49oJFJ6SL1vMDtFg0DBBwAAA
AD1wIWD+qtJUqK0ZmRWez7IamhgZH4tp0KgjbZ8a3YEoFYYFuuJjTWsVF0BwFcRAAAAAOqBmKhIXXTBABUWl6i
8oqLaMamucpbq1mMX0rYowZdldgygVlh6Xy816WV2jAaDgg8AAAAA6omB5/ZQxzYtdeBw2knH9IsoUKjVW4upE
CjXBS0wOwJQK4yYVtKF08y00aBQ8AEAAABAPeFy0nXpiAsU5LAr07f65bhBVkMDo/JqNxj0WqSlWL3Z0ReNgGG
xyXLZq1JQqNlRGhQKPgAAAACoR87p2FZDB5yrtMwsuT3Vb8aQ6ipXu+CSWk6Gs3GLa5GCVP3Sa6AhsZx/l5Tax
+wYDQ4FHwAAAADUIxaLRWOGD1bbls20/9DRk47rF5mvEJbq1huXWpaaHQGocb6kbtLge82OOSBR8AEAAABAPRM
RHqZfX3yh7DbbKZfqns9S3Xqhn32PmvhOXtYCDYHP5pR1/OuSzW52lAaJgg8AAAAA6qFundr5l+p6TrFUty1Ld
eu8m53zzY4A1LwLp0nx7cxOOWBR8AEAAABAPWSxWDTmwhNLdfedYqlu/8h8RdiqLwBhvhCV6zzferNjADXK03y
QrH1vMTtGg0bBBwAAAAD1VGR4mMZfPFx2m005eSdfqjs0Olc2GbWcDj/HZNcSuYxSs2MANcYbFCH7+Fcli8XsK
A0aBR8AAAAAA1GPd07XX0PPOVVrGyZfqxgW51Tey+gIQ5vq1dYnZEYAaZb30eSki2ewYDR4FHwAAAADUYxaLRWN
GDFab0yzV7RRaopYuZorVJV1th9XSt9/sGECN8fT5rSydx5odo1Gg4AMAAACAeu7nLNWVpPOj8ngeXx3yW+d8s
WgRDVV5Um/ZL3rU7BiNBgUfAAAAADQA3Tu11/Dz+yot45jKKyqqHRNkNTQsJkd2i6+WO+Gn7PJqiLHa7BhAjSh

```
3xsp5zfuS1WZ2lEaDgg8AAAAAGgCLxaKxI4eqW6d22rP/kAyj+k01Yh0eDeR5fKa72rVMoUaR2TGAgPPKJtuV7
0qhcWZHaVQo+AAAAACggQgJdunqca0VGB+nA0fSTjquTUip0oVSLplpgm2x2RGAGlEy5EHZW/Q1O0ajQ8EHAAA
AAA1IakqSrhgzQh6P95TP4+sXUaDEoPJaTIbvtbZmqp13l9kxgIDLbX6Rwof8zuwYjRIFHwAAAAA0MP16nqMRg
/opLe0Yysqrfx6f1SINi85ViNVby+lwm+tLWVX9EmqgvioIa62oa94y00ajRcEHAAAAAAA2MxWLRZa0GqWfXjtp
74JB8vuo31Qix+TQyNls0Nt2oNRb5NFwrzI4BBFSZNUzB130si91pdpRGi4IPAAAAABqg75/Hl5qSpL0Hj5x0X
KzDo2HRucwoqyXjnGsV6csz0wYQMG7Z5b3833LENjc7SqNGwQcAAAAADVRKYryuGnuxghx2ZWQdP+m4pq5ynR+
VV3vBGrFr7AvNjgAEjM+wqGDY3xXaYajZURo9Cj4AAAAAaMC6d26vy0YNVW5+gQqLik86rm1IqXqFF9RissYn2
Zqrrt5tZscAAib9nCmKPf86s2NAFHwAAAAA0OCNGNRfg/v11sEjaSqvqH7TDUnqEV6kDiEnLwFxdm51LpBNbGq
ChmF/0iVKGfeQ2THwPxR8AAAAANDA2e12XXnpKPXo0kG79x2Ux+M56djzIv0V6iyrxXSNx8WWZWZHAAJif2hPN
bnuDVksFr0j4H8o+AAAAACgEQgPC9V1V4xVu1bNtWvfwZPurGu1SE0jcxXn0PlMP5y5UY6NivMdMzsGcNYO2Vs
qevIsBTldZkfBj1DwAQAAAEAjkRAboxsmXKaUxHjt2X9IhlH9zrk0q6GRMTkKt518ph/0zHVBX5kdAThrmYqX8
+r3FBWfbHYU/AQFHwAAAAA0Is2bpui6K8YqPCxUB46knXRcsM2nUbHZcll5ZtzZirEUqad3k9kxgLOSa0TI/eu
3ldiig9lRUA0KPgAAAABoZDq3a61rxl8ii6T0zJMvG420e3UxJd9Zu8W1UA65zY4B/GJ5vlBlj3xRTbucZ3YUn
AQFHwAAAAA0Qn17dNVvLhmhgqJiZefmnXRcjM0j0ZR8Z2WM5RuzIwC/WK43VEcG/0NtzvuV2VFwChR8AAAAANA
IWSwWXXh+P40ZPkiZx7JVWFR80rHR/yv5gin5zthA+06l+E6+FBqoy3K8IdrT5yF1Hnq52VFwGhR8AAAAANBIW
a1WXTZqmIaed640HklTaVnZScdS8v0yNwV9aXYE4BfJ9oRoS6d71XP0ZFksFrPj4DQo+AAAAACgEXM47Lrqsot
1bo+u2rP/sMrKK046Nsrh0ei44wqh5PtZQlSmfsZ3ZscAzthxT6jWtbpD/cf/Vjabzew4+Bko+AAAAACgkQsJd
un6K8aq1zmdtGffwV0XfHYvJd/PdKPrazmNk8+KBOqi455QrWx6kwZefoccQU6z4+BnspsdAAAANC5er1cfvPC
0ln76ofKOH1N0QqIuu0xy/frWqf7lH6XFxXrn6Ue0euF8FeXlKqFpqi6+5gaNnHDtSa/7wDXjtXXNiirHew4ep
j//89+SpPef/7u+/fwTZWekye4IUqv0XXXV1HvVrltPSZK7olwv/eX/tGbhfEXFJeimvz6qbucN8l9r9hsv6Xj
aUd14/yOB/JIAQJ0QFRGumye01+sWi9Zs3KK2rZrL5az+l/vI/5V8nx+PU7GP2T0nM862VKIHRT1yzBOq1ak3a
+jEuxQSFm52HJwBCj4AAFCrZr/2oua/+5buePw5pbZpr71bNuqFP92pkLBwjb72RknSjMcf1JZVy/T7J59XQpN
UbVi2RK9Nu08xCYnqM3Rktdf94/0vy+N2+z8vzMvVH8YOV/+Rl/iPpbRopRvvf0SJqc1VUVamuW+9qoduuFIvf
LlckTGxWvD+O9q3dZMefW+0vlu6SM/+3+3617JNslgsyjxySF99MEtPfvhFzX6BAMBEURHhumnieMkirdmwVW1
aNlOw6zQlX3asirz8avlT3WOH1cJ7wOwYwM+W5QnTmtRbdMHEOyn36iGW6AIAgFq187u16jNspHoNGa6EpqnqP
+oSdRswWHs2b/hhzIa1GjL2N+rS9zwlNE3ViCuuVov2nbR704aTXjc8KlrR8Qn+j03Ll8rpCtZ5o8b4x5w/Zpy
6nTdISanN1axte02+90GVFBXq4M5tkqQj+/ao99ARata2vUZNnKyCnGwV50ZIkl598F5d839/5gdeAA1eZHiYb
rpqvM7t0UV7Dxw65cYbEXavRsdmK8ruPumYxuq3zvlmRwB+tgMVMVrZ/HbKvXqMgg8AANSq9j16a/0Kb5W2f68
k6cC0rdqxfrV6DBr6w5juvbVm0ZfKzkyXYRjavHKZ0g7sU7cBg3/2fRb+910NuPhSuUJCqj3vrqjQgvffUUh4h
Fp06CRJatG+k3asW63yZJt+PZrRccnKiI6RkvnfCSH06m+F150Fu8cA0qPyPAw3XTl0P/GG6cq+cLtXo2J067
koPJaTFi30eTR+cYas2MAP8vmsmRtaH27Lrzqd5R79RjzqAEAQK2670YpKiku108uHiSrzSaf16urpt6rQWPG+
cfceP/DeuX+u3Xz4F6y2e2yWKy69aGn1LlPv591j92bvt0h3Tt02yNPVzm3dvEC/eMPt6q8tFTR8Yn667/eU0R
0rCRp6PgJ0rhrm6a0HqLw6Bj94dlXVJSfp/emP6Vpb/9Xs559Qss+/0SJqc11+6PPKDYx0TBfFACogyL+V/JZL
RatWLdJrVukKiTYVe1Yp9XQRbHZ+iYvSrtLq/+LlcbkWtc3CjWKzY4BnNay4pY61nGSRl5xk4JDw8yOg7NAwQc
AAGrV8i8+1TdzPtLUv7+o1DbttX/HVr384L2KiEvQ0PMv+y+aofgmTbVtzUq9Nu1pIrLTpxcks/
0+7atauo9qe06PKuS59B+jvHy9QYW60Fvxnpp6eeose/+AzRcbGye5w6KYHHqs0/oX7puria27Q/u1btHrhPD0
9+yvNfv1FvfHw/br7+dcD80UBgDoqPCxUN0y4TJK0Yv0mtW5+8pLGapEGR+cp307R+sKI2oxZ51xh+5rNNVCn+
WTVvML2qugyQa0uuEmukFCzI+EssUQPVfP2HTXk019rz0Sb9NGrz0uSystLZ1xuTXS0AePVfP2HTXk0l9rz0Sb9NGrz0uSystKNevZxzX
53gfVZ+gItWjfSRdffb0GXPwrffqvV057/bKSEi37/BMN+/WV1Z53hYQouXlLteveS7c/8oysdrsW/vfdasduX
rlMh/fs0kUTr9OWVSvUc9BQuUJCdN5Fv9LW1VV37AWAhig8LFQ3Xjl05/Xqpr0HD6u4tPSU43gF2lwVK6sMmo
pYd3S1pahNt49ZscATsotuz7I7yZP16so9xoQZvABAIBaVV5aJou18t8XWq02+XyGJOnAm79Eh2Jq9xZQm0x3f
pL5
82Ru6JCg3+05PdUDJ9P7oqqz42qKC/T6w/9Sb9/6gXZbDb5fF7J831G94nPAaCRCAsN0Y1XjpPNZt03q79T0+R
ERUWe/FldbUNKFWbz6qucGJUbjWteyW30ebL6Gme5ibqvWMH6IL+bks4dp6Fjr5EzmCX1DQUFHwAAqFW9L7hQH
74yXfHJTU4s0d2+RXNm/FNDx0+QJIWEhatzn/56+6mHFOR0Kb5JU21dvUJLPvmvJt37V/91pt/z08UkJ0nqP/y
p0vUXffiuzh0+UuHRMZW0l5WU6MNXnl0OfoSMUFZ+owtwczZv1pnIyM9T/Rzvtfu8/Lz2rno0GqlWnrpKkDj376
O2nHtIF467QFzPfVIeefQL9pQGA0i0JFg3XnmZIsJC9eWS5SqvqZGOFBif0xJxyc7KzQm/pi+zI5Vgbdx/0pplVf
DjFVmxwCqlWNE6IPC3uo47AoNGDledofD7EgIoMbxXRYAANQZN/7lYb07/Um90OuO+FWRnKzohURdecY1+c9ud/
jF3Pv0yZj7zqJ774xQV5ecpLqWJrpx6j0Z0uNY/5njaUVkslWeFHN23R9vXrdYDb1Rdcmu1WXV0/x59/bv/qCA
3R+FR0WrTtZsenvmxmrVtX2nsoV07tHzeHD398QL/sf4jL9HW1St0/8TLlNKytab+/cVAfUkAoN5w0Z2aeNnFi
```

```
omK0EdfLNKBI2lq3iRZFoul2vFR/9thd0F0jLLcQbWctvb9xrlGEUa+2TGAKnZ5UvRlRS/1H3uNegy8UFZr45p
Z2xhYDMMI6Nzhxx57TB999JF27Nih40BgnXfeeXriiSfUvv0PPziXlZXpD3/4g9577z2Vl5dr5MiReumll5SYm
0gfc+jQId1666lavHixwsLCNGnSJD322G0y239eJ1lQUKDIyEjl5+crIqL2H/D60c70Wr8ngIZnXHt26Pwpvr8
CCBSzvsea/XMqEAiGYWj52g2a0fsLFRUVq03LZqcsDDyGtCI/UjtLGvazvj4NfUTneLeaHQPw88mqJaXttc3ZQ
8Muu1YdevQ3OxJqSMAr2yVLluj222/XypUrtWDBArndbo0YMULFxT9sEX7nnXdqzpw5+s9//qMlS5YoLS1N48b
98Jwcr9er0aNHq6KiQsuXL9dbb72lGTNm6IEHHgh0XAAAAAADAGbJYLBrQp4dun3SFEuPjtH33frk9np0Ot1uk8
6PyNSQqVw7L6Z+nWh81tWars3eH2TEAvxJLiGbl99LeyAEac+0dlHsNXMBn8P3UsWPHlJCQoCVL1mjQoEHKz89
XfHy8Zs2apV//+teSpB07dqhjx45asWKF+vXrpy+EKXXHKJ0tLS/LP6Xnn1Fd1zzz06duyYgoJ0P7Xb7L8ZZ
YYJgEBgBl9VfH8FECjM4AMC43Baht54f7a2796nNi1SFexynXJ8nsemhTkxyvU0r0d/PRLynib6PjU7BiBJyrA
m6/1jbRXTqrtGXXGT4l0amR0JNazGF13n5594/kBMzIkHXa9bt05ut1vDhw/3j+nQoYaNWumFStWSJJWrFihr
l27VlqyO3LkSBUUFGjrVqY7AwAAAEBdkZqSpN9dd6X69uiqPfsPq6Cw6JTjo+xeXRp/TO1Cik85rr65SMvMjgB
IkjZZu+jNzE5K6TJQl076PeVeI1Gjm2z4fD5NnTpVAwYMUJcuXSRJGRkZCGoKUlRUVKWxiYmsjI8I/5cbn3/
fnvz1WnvLxc5eXl/s8LCgoC9TYAAAAAAKcQExWp3179G0WGhnh3wlWKK69QQlzMSScfLbLdKgqHwl1B1VoWX6kPEb
9fuD/JUHfKcaXbXYMNHJuBWmBu7c2F0Wp56BhGnTJBAWHhpkdC7WkRr+L3n777dqyZYvee++9Ovee++9mryNpBObe0RGR
vo/UlNTa/yeAAAAAIATQoJdmvSbMfrN6BEqLinVngOH0H5f0d+nl7bUNKNTbuuKLt7lpKWTMm2ReaHQGNXI41Tm/
l99MepWrEb67X8F9fR7nXyNRYwTdlyhTNnTtXixcvVtOmTf3Hk5KSVFFFRoby8ZSU5B+TmZl5fz35
6pz3333KT8/3/9x+PDhAL4bAAAAAMDp2012jR15gW6bdIUiXiY6K1bdc+lZVnfI1UQ6PLo07rnbBJbWUMrDiLAX
q7ttkdgw0Uj5ZtMneXf/M7CZ7SheNvW6qegy8UDabzexoqGUBL/gMw9CUKVP08ccjavMmHC8ccfa9GiRWrZsmW187ZLD4
dDChT/8DcfOnTt16NAh9e9/YkeX/9e9/YkeX/v37a/PmzcrKyvKPWbBggSIiItSpU6eaPdMoLhUREVHpBQAAAAAAAABQuywWi3p27aQ7Jo9Xs+QEbd25R3n5BTV6+dq9SdbbZEhbFHaA5JExJuXxNuQRiSTKAGyUaHB4O7TVPYeg7cy0fOhwJ6YKTS7pXfQiTE8oZMUP6uUkWNVAA6EWIBjWXDJz0gGAIhCKrq
rWpvjNBwxfwZ/Ddfvvtmj Vrlj755BOFh4f7n5kXGRmp40BgRUZG6oYbbtBdd92lmJgYRURE6I477lD//v3Vr18
/SdKIESPUqVMnXXPNNXryySeVkZGhv/zlL7r99tvldDoDHRkAAAAAEGBJ8bGaMmmC5ny1RZ8eF8s/lYFRUVqmdpEV
uvJ55m0CSlVirNcy/MjdaAsuBbT/nJjLN9IhtkpONjsDeqkz3NayG0J0vmjf6W+Q8fIERRkdiyYK0AF38svvvy
JGjJkSKXjb775D77775piZPnixJ+sc//iGr1arx48ervLxcI0eO1KutZXjb775piZPnixJ+sc//iGr1arx48ervLxcI0e01EsvveQfa7PZNHfuXN16663q37+/QkNDNWnSJE2bN
i3QcQEAAAAANcTpDNL4i4erRWoTvffWW6Qq20U66WtCbD4Nj8nV/tJSLc+PVHF1iKRkkJjiRp56VVq27WPLBaL2dFgMothGA3y7xoKCgoUGRmp/Px8
ewIdQ7fXwEEilnfY83+ORUwS3rWcc2a/bnWbNyqxPhYxcdEn/Y15T6LVOkPHP/Dpmu830rzY6BRmK
fs7MWl3dV5rFsterUQ8PGXav4ZDYYxQkBn8EHAAAAAMBPJSfE6fZJV2jO0giX1EtTpB/ZdVoNDY70U
6vgUi3Lj1SRt+78ChtmKd05xndmx0AjUGwN1yrXEK1NN+RwVGjARb9W9Wv2G/kj04bhbfMEfd+e4IAAAAAGjQXE6
nfj36QrVIbaL3P52vrbv2qlmTZEWEh55yvNvq0/DoPGVV2JVdUXOfgrnxclFuYubXB1LFpbleNwBrm95BVIFIn5uyPwYVHMPb2VQQF
j+YrpQWbTXokivUskM3WYKa8Ovs5o3TdbseYu1b0l3Op6dqxapKbLZTv68PYfV0HlRXXX/Ms0rsKHH/aMpbxd/3Dyt0R+W
o1VXKpv8qKUb/JsvsusS6X6teEv6pE0RwutcQ3WnvR8+by56jlopAZe9BuFRUSZHQl1FAUfAAAAAADWJcTG6ML
103Tu308bxF2r57n5okJyo68tTPpkxyVmhcQpa2FodqQ2G4KoyTL/GtKb3s+9XMd6jW74uGr8gaofWhQ7SzPEH
H9h1UQpPmGjBqvNp373fK5ewABR8AAAAAAwBRWq1V9e3tVu1bN9cn8r/XN6vWqcLuVnBB/2l3VgEo6N/X5o
UJKtL4wXNuLuM2LDQI4jZKjWPZpkq6BGoMKtSvIjrmrR1kjrYqUPhjCtRj8gUXxgpcEsFzV6yykWAzcrYZHdLg
YVZZDKmBhpp7L7nt5LpdLBeDi58z4uhPh/ogt6o1N1zqWyTXauYKN3A2nEUU0RqzEXVrM/4YfqVk31GU2bI
JovLlI6cjqjbUMHU5qD4CkSJQJEq/L+dqvx00FEqnTgB9ppeQHVFv7VvTAXrBBUG8w2VQomfTJhMwtr+DKxkc
fvWc4VPnqS7q8GVprjtbaPrhBoaMZbb8Y+4OZ1XN0nX4+k7+oKpSCzcXvqU1O6mJpjoq4eoGoqVarlJb/VbNlaS4
zGovZDJPiYp2bGvkQLeoLJZIYuUb0JsG2LKJlCyeJy5SL4PGs0MEGpcBFSmYpdpLOPGXDrGLzmtyHbq+7K7B5GjUQ
Z70vVEEv5x8ZUSHxpDLJIGOmC6prGCCTNFCSDZVLPWzaXvyB8d51fpQDP82QHNS59Dq0Wq2trBwrvxu2B6+CGjsJ
ZTdiwpGRSg2+UedXD3j5iJcXz9e1TQjpVwLevYYfE5T85O+HJmq1RrBR57Vx3vKbOP8M7pFhRbcqD+xGPc1FA/
lo7ZQUEt0Vf9tPSyrDHytc1ZG7DNf/sPdQBhjZh5iohRbpRJ8s3Rbo/ZjuOTGY5p3ykUWDLv2B5Dz8oHk6Xx2ZFbG
```

s4mOiT1vOWSxSy+AytQwu09HyIGOsDFdahbPasc2tx9XRu6Mm3gbqOa9sOuDsqB2unsq3x8vn8yk3KO2FudmKT
UxR/+GXqnOf8xUaHml2VDRwFHwAAAAAgHorOjJC4y8err49umrx8jVavm6jtu7co9iYaCXGxchqPfOOuk2cFWr
izNZxtl07ikO1p8Qlj35Y8nur60vZfL6afBuoZOotIdrj6qZdru4qt4bI5/MpLytdBTnHFBWfpAvGXq2u5w5WW
GSO2VElSUVFRdqzZ4//8/3792vDhg2KiYlRs2bNTEyGQKHgAwAAAADUeO2TE3XN+EsObGBfLVvznb5ZvV7bdu1
VVGSEkhLiZD/NM/okKc7hOcCofIVvn618W6QKWgxVhidEI7W8Ft4B6oPj9mTtcXbVAWdH+Sx2+Xw+5R9LV372M
UXExGnQJRNOTt8hioiJMztqJWvXrtUFF1zg//yuu+6SJE2aNEkzZswwKRUCiYIPAAAAANBgpCTG6zeXjNAF552
rFes2avHyNdq5Z79CQOPUJDFBDsepfw2uqCiTN3ufeqUmqWvidmVXWJVdOURhFUVyGBW19C5QlxRbw3TA2Un7n
J1VaIuRJHncFcrJOqiSwnxFxsbrvJHj1K3/UEXHJ5mctnpDhgyRYRhmxOANouADAAAAADQ4cTFRGnPhYA3q1Ou
rvtusRctWa/f+g3I6g9QkKVEuZ/Uba6Tv3yWnr1wtW7eVJMUG+bQ6aITWGRcotWK3WpVtUaLnsE79hD/Udx7Zd
TiorfY5OyvTOezEQxsllZUUKzvzqLwet+JTmmnAqPFqf865dW7GHhofCj4AAAAAQIMVGR6mEYP6a2CfHlqzcYs
WfrtK+w4dldVqVXJCnMJCQyptyJF7eKcSoyIUFh5e6TpeiOMHnJ1OwNlJod58Na/YodTy3Yr1Ztb2WOINMSRl2
Ztqv7OzDjnbyWM5UQIbhqGivBzlHkuX3e5QaqsOOqffELXuOktOV7C5oYH/oeADAAAAADR4IcEuDe7XW/16nKP
vtu7QkpVrtWvfQRO8kq7Y6EjFx8WoKOeYVJKt1DZdT3mtYluktgX31bbgvgrxFii1YrdSK3YrzpMmq1gGWZ/4Z
NFxe7KOBrXWoaD2Krb9sNutz+tVXnaWCnKPKzQ8Ul36DFLnPuerWdvOsv2MZzoCtYmCDwAAAADQaDidQerX8xy
d272L9h48onWbtmrF+k3auWe/StN3KcIuNUn9+buKltgitDO4l3YG95LLV6ym/yv7Et1HZBU779ZFbjmUHtRCR
x2tlRbUUuXWEP85wzBUUpivvOwsedwVioxNOIBR49WxR3/FpzSrNNsTqEso+AAAAAAAjY7ValXbls3UtmUzXTT
OfG3YukPzPpopa46hw3u2KywqWlGxCbI7qn9WX3XKrKHa4+quPa7uCvKVqmnFXjVx71XC/7d378FR1ff/x197z
2aTzSYkJATCVSBCuSkl8xp8O9WcELOPAb/qtwFCHMqCd1qnjYC/4hOTKH2JlbKcOLR1HoJ22IHaOOK2ilJIy2hQ
6gF8BwXKJSIAkJCTZ3Vz38vn9kbC6hgCBhOwmz8dwZvd89rNnP+edT4nmZrms47xqnSO
U7VjgJFLfFxSHtri+prq9US9Cs1zavRE6eocPo9GlM4VWkZmfOOauDGEfABAAAAAAa1jPQOzb5npu6fdZcunj2
l08cO68Thcl08eOoykndIjtIzsmTtwdsy261unUn5ms6kfEOyRpmRGuWFPldu6HPlhM7LoVAf7hFCcqjWMUw19
gJdcI5RvT23S59IJKzGuksKNNTJZndoaH6Bih9aoLF3TudsPSQdAj4AAAAAANRxVt/wMRMOfMwEFT34iCo+/V
//eiAzp46pnOnj8tqtSk9c9c4jSfVmy2Xrw57TFonp7rurutru/rosJqoh4YvKDZ1TbuhzZYcvyq5w3+3YINBqc
euSfbguOYarxjFC9bahMhZrl37RSERBf4P8ly8pGo3Im5Wj7nDOw+iBw+iBwAAAADAV7j
cqSqcfo8Kp98j9/VanH/3tU2XQCBh1jjr8lvZx/5ggPuSdEqAbvO9+RyuEcdOx1VEI3DdG0e/+vRo2
YVrlBmuUXq0Xpw/1rOma7CM6AJ37ClTX8f3jji
w/3sJt2ZXaGfVmRjtuMSN2g++COkBxqtGerwda52LPVaMuO+2CMrzLGqK21WYH6OjUH/bLZbEr3DdGOe/+vRo2
frOFjJsibmc1bcDGgEPABAAAAAHCDOrw+Fc4oVuGMYjUHAzp/5Azp/5oQ+++8xnfnkkKrPnVEkEpLD5Vaa16fU9Iwen
913Rdji1CXHCF1yjIi1WU1YvkitfOFLSos2Ki3SOHnbqBTTO1u72C/Csitoy1Djl4K8BluOmqzebs/M+7JoJKL
moF+Bhjq1tbbIleJWZk6eZvyfhzRi3ETljxqvlFTPbdgToH8Q8AEAAAAAcBNS09I1I9k1urfurXW8qgunD2
vNndfbkMdVUfqaLn51UNBqVKyVVHm+GUtN9svXggzq+Kmqx67I9T5fteVOes5t2pUUa5ekM/NKiDUqL+JUaDch
pWuWMtvbbB3tEZFOL1dO5pKnFmqZma7qCVq+abBlqsnqveUbe1YTDITUHGtUc8KutpVkWq0VuuT7ryR92h0752t
4aPmaDcEWNksxN7YHBgpgMAAAAAcItc7lSSWVyNTVfqfpVqfpLVTLGKCXVI7f
HK7cnTQ6nq1deP2xxqsGeowbldNvHYiJydYZ9TtMad99pWmUzYVlkvliMkb68ri/WI7IrZHEqbHEq1Ll0d7/dk
nJDZ+FdS6i9TU2BRjUHGhVqb5XVapMnPUDRo1Twbg7lTt8lLLzR8o3ZChvvcWgRMAHAAAAAAEAvS/Z6PeUFsZ
NmqHiOf9Pgfo6VTZ//TFXnKvTZ0fJUa5k3DKWGxqtXjXa
1VrS5Nam5vU1tKsaDQiu82qu90hjzdDYZO1YhxhRo6fKRyho2Ux+sj0ANEwAcAAAAAAAOhJ+ySWCzyN6r6
H/OdNgUbVX7qoyzVVulxzQRfOnljDbZVqqyoVamP1LQV6TwqGOtxA7n
SlypaYqa+gw5Y4Yo+y8EcrJL1DOsJFye9L6edsJFye9L6edRAYiLgAwAAAADgNrJBn+Hl9Khh3Z6y9pSmo+ksXVV9brcs
1F3Tx7GnVVZ9Xc6BRDZeqFA1tSnU1qr29lqr29laF2tpkTFGG
NnsdqW4PUpk18g35ChyhgyVL4huXJ70gZUoAn0JQI+AAAAAASgNuTJrdnvPJHj4+1hdrbFFqFXq6m
xXsFAg5oaG+RvvKz1Sllu agWoIBtbe1KBqdNdlzqzlhkjJHV
9L5zyJZLBOhYmewZrFYFI1GEOu53tnWsRxSNdtw3V8YgyWq1yWq1qbGhQWq1+50yuFOyZXiVnZ+gTKz85SRlS1
Puk8ZQ3LkyxqqNF9W53gA3CwCPgAAAAAAEpTD6VRmdq9qy9vU3trc1Gaftrc1qb2tRR63zT8lqb21RKGHkrc
1NAzUG/WpoCamkKqK21ReFQu9pammWiUZmOjcjIqIacHNFpvO245+RJFksvctlsVN1mvLFZbZ2iXIacrVS63W66n
cHnm8GXJ70uX2pCnVk95xP82r1DQvIR7Qhwj4AAAAAABIQaLpfOsvzTl1DCu4bv9IJNR/rW1qa1xDd7yKqKg
Olc71gko2g0Khhjq80uh9Mlh9Mlu8PZeTuwrgcIJDsCPgAAAAABgbAAAAAABgGbzRYLBAEMLJwfCwAAAAAACQxAj4AAAAA
AAAAgiRHwAQAAAAAAAAAAAAAAEmMgA8AAAAAAAAAAAAAAABIYgR8AAAAAAAAAAAAAAACQxAj4AAAAAAAAAAAAAAAAkMQI+AAAAAAAA

IAkRsAHAAAAAAAAAJDECPgAAAAAAACCJEfABAAAAAAAAASYyADwAAAAAAAEhiBHwAAAAAAABAEiPgAwAAAAAAAJI
YAR8AAAAAAAACQxAj4AAAAAAAAgCRGwAcAAAAAAAAAkMQI+AAAAAAAAIIkR8AEAAAAAAAABJjIAPAAAAAAAAASGIEf
AAAAAAAAEASI+ADAAAAAAAAkhgBHwAAAAAAAJDECPgAAAAAAACAJEbABwAAAAAAACQxAj4AAAAAAAAgiRHwAQA
AAAAAAEmMgA8AAAAAAABIYgR8AAAAAAAAQBIj4AMAAAAAAACSGAEfAAAAAAAAkMQI+AAAAAAAAIAkRsAHAAAAA
AAAJLGEDvg2btyo0aNHKyUlRUVFRTpw4EB/DwkAAAAAAABIKAkb8L3++utatWqVSktLdejQIU2bNk1z585VTU1
Nfw8NAAAAAAAAASBgJG/C9/PLLevzxx7V8+XJNmjRJmzZtUmpqqjZv3tzfQwMAAAAAAAAShr2/B3A17e3t0onjwo
J599tlYm9VqVUlJicrLy6/6nLa2W1W1tcXWGxsbJU1+v79B9uN5mCgX14XwMDi93v6ewgjh+MrgN7SX8fYK7+
fGmP65fUBAMDAk5ABX2ltrSKRiHJzc+Pac3NzeLEias+54UXXtWu7tBcUFPTJGAEAAIBbEQgEl0JGR0d/DA
AAAAOBCBnw349lnn9WqVati69FoVJcvX9aQIUNksVj6cWTJz+/3q6Cg0OfOnZPX6+3v4QwY1LVvUNe+QVT37BnX
tG9S1b1DX3mOMUSAQUH5+fn8PBQADBAJGfBlZ2fLZrOpuro6rr26ulp5eXlXfY7L5ZYr8nr9Ypr8/l8fTXEQcnr9
fILfR+grn2DuvYN6to3qGvfoK59g7r2Ds7cAwAAvSkhP2TD6XTq7rvv1p49e2Jt0WhUe/bsUXFxcT+ODAAAAAA
AAEgsCXkGnyStWrVKy5Yt08yZMzZMzVr1iz98pe/VFNTk5YvX97fQwMAAAAAAAASRsIGfIsWLdK0ZsOaVVVVa
frO6dq1a1eXD95A33O5XCotLe1yPi9LHGmrUlu/qfYS3r+8NN0U+QUqRQVmmWFWt/rldRzAQPPtsV91P/lvLR4
LyjiGNIxx8+0sXaGaY8/RN/TkCoEKiR
7fX+NrI6kt+sLyOb5LNTfdsa9q6Ww8aQP/7T8RAt26u7/D+g9HYpNh6Nm7a2C7V2UNxY+GqaLNmbN7xW5cDZ8
xzM0lbGeOcOSx6+5/NsaYhnOUWZQyYl5Ww/cD6WhUDGZVFLkvMC7cavGl7fr1zNe18frhKM7wUJB1TD7ifb6/
7+2ptpmhQ66X3V14+tn4w7kdsIdMK6ecxXpyYG8xpEG94F72Cfo3GQmjHy7HWf8LOfB4T2Ne8MvHx21/ZPPZhKh
7tuW+QX/Ym2Nz1m2qN7KFNqJ5jVyxXvqV+xs+H8UKaX4D2pNMMhrjkmpzEmB4wtwkUEq99b2YhoPTN8gYI5TU
Q/gbTeTYkeCx0yZ5PfYFrZGBAl+O22Jm6dj8MBoUEiMXXW3O1d/nGMGt+5+08Hw5Gl8ty/xchyIRSe/eSgA0
/wH2mhmK81ua5r3ZzltMqGbIuxDn+1PzhWMMgjPp9mjK43jb/a/zz17rp28xjUsduZ+o3uAhgOkSsfKuIhIkEx
wQ4Oz6pHEpqXgpSLAYz6OEoS6zT0EX1H8Xpju80fiQLPn2evl/z3JP4Y8l9+j9SPvt5B0rsjLv3MBdoz2b6x
6rQzpzDI+uRxkLugGHN/LKkHJGxwdQt3JUdkvKMa1GyI3sLNhHYV5iUbEqFSLnt2WoNUwi0fNwUK66+gB0oG
k8CMf5LeVS0l/kwZ9ZAqFKkxvu48uSXpSZhvA5THAAf3bANeg4l/vBe0Y25b5l+V/lGZZHSPY16CO9+I8CeH
PRGmnFjAv0smMUlm7cwz+LKiQyFtfjnXiTnV9w5Frnug69Yrd+2V56g8qMTXY16WN6H9pBppg9S3dcqo+yR0U
6nLKXqGnsI4l6kC3TL/3EZxHEqZxO08ZdJDNxl1/FB0O8uFW+2F8wNLSdHPMQcH2ZkxfZ1BJEQQVg9wYQWc
1zcZAwZODEYpn+t/R6tSjQOpbeTqX4iFq3aafOOORJBFIvZTNtZgSOmKNKT2bKxNDfpfAjVT4wQz6rZHT0F
bXIiZuNd6/9aBQ/5Yhl6dLdHQEMPDnf9mlzFC+SuVkRSMf8xWbmh7RVrEhCu+XOp4mGj6Pq/mJ9/EcfSBk3o
3e7+vgQRv6L8uG8pX/+IH7xY88bC7iZvwpSKFXrYUjD9xR+n6uJ7o3nRhdFE8HVC84/xXWhbwUJrcSTYc/rS
JGxtM8vjcsJAvvP6Bcj1Nfa2Nh/y4+mXPG+b9Ah7hk91mb0SDNCd+k9q1ZRgl7bG5PfpQQ+Qw2ZgwX8pu1wtYS

```
mvffeM6dPnzYHDx40ixcvNikpKebYsWOxPszXDj2t7RX33XefWbRoUZd25mxHDQ4fPmwOHz5sJJmXX37ZHD582
Jw9e9YYY8zq1avNY489Fut/5swZk5qaan784x+b48ePm40bNxqbzWZ27doV6309r9Ng0NO6/vGPfzR2u91s3Lg
x7vja0NAQ6/PMM8+YsrIyU1FRYT788ENTUlJisrOzTU1NzW3fPwAAgMGGgA+mrq7OLFmyxKSlpRmv12uWL18eC
0SMMaaiosJIMnv37jXGGLN3714j6apLRUWFMcaYd991O0yfPt2kpaUZj8djpk2bZjZt2mQikUg/70Ht8corr5i
RI0cap9NpZs2aZf7973/HHps9e3aYCRMmGKfTaSZPnmz+9re/xT0ejUbUbNc889Z3Jzc43L5TIPP
vig+fTTT2/HriSUntR11KhRV52Xpa1xhhjmpubJ5//33Bxm1KhR5vHHHx9Uf9RfOZO6Pv3007G+ubm
55pvf/KY5d0hQ3PaYr1/o6bHgxIkTRpJ5//33u2yLOdv9z5wrdVy2bJmZPXt21+dMz7d0J10M3bsWLNly5Yu2
73W12kw6GldZ8+efc3+xhizaNEiM2zYMON0Os3w4cPNokWLzKlTp27vjgEAAAxSFmOMuR1nCgIAAAAAAADofVy
DDwAAAAAAAEhiBHwAAAAAAAABAEiPgAwAAAAAAAJIYAR8AAAAAAAACQxAj4AAAAAAAAgCRGwAcAAAAAAAAAkMQI+A
AAAAAAAIIkR8AEAAAAAAAABJjIAPAAAAAAAASGIEfAAAAAAAAEASI+ADAAAAAAAAkhgBHwAAAAAAAJDE/j8rekA
X1RB1NAAAAABJRU5ErkJggg==\n"
```

```
          },
          "metadata": {}
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "1. Most people do not have a phone (69.06%)\n",
        "2. Both categories have almost same approval rates"
      ],
      "metadata": {
        "id": "1-wZTlo7OGcv"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "### Treating Outliers in Numerical Columns"
      ],
      "metadata": {
        "id": "aAZy6iNkENRQ"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "# log transformation of annual income column\n",
        "import numpy as np\n",
        "merged_data[\"Annual_income\"] = 
merged_data[\"Annual_income\"].apply(np.log)"
      ],
      "metadata": {
        "id": "MJIzEDf3fPnd"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "# plotting outliers using box plot\n",
        "numerical_columns =
[\"CHILDREN\",\"Annual_income\",\"Employed_years\",\"Family_Members\",\"Age\"]\n",
        "fig,ax = plt.subplots(1,5,figsize=(15,5))\n",
        "for i in range(0,len(numerical_columns)):\n",
        "  ax[i].boxplot(merged_data[numerical_columns[i]])\n",
        "  ax[i].set_title(numerical_columns[i])\n",
        "plt.show()"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 468
        },
        "id": "FZl-4HmhY1Nh",
        "outputId": "e69ca6d0-ca7b-4d54-f61e-77ce997a9779"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 1500x500 with 5 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAABLkAAAHDCAYAAADIusjNAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAACUZkl
EQVR4nOzdeVhU5f//8dcAsriASqSC+QGueIKGoW5ZWISqOWe9SnLJRU1o19atkiappW731JLzdKQzD7ax8yFF
EtRSss9EVPArARRBBvm94cXUyOgosAM8nxc17ni3Pd9znmPzZnlPfdiMJlMJgEAAAAAAABlmJ21AwAAAAAAAB
uF0kuAAAAAAAAlHkkuQAAAAAAAFDmkeQCAAAAAABAmUeSCwAAAAAAGUeSS4AAAAAAACUeSS5AAAAAAAAAUOaR5
AIAAAAAAECZR5ILAAAAAAAAZR5JLgAoo5YuXSqDwaCkpKQSPQaAtHXVrVhkMBm3dutXaoRTq1VdflcFgsHYYgE0
wGAx69dVXzfvl/f0ovJCREDiSHLZkOPHj2vYsGG6555zsLFdXV3Xo0EHvvvvuusrKyJEk+Pj4KD
Q0t8Pi8D+Br1qwxl+W9oe/Zs8dclvch+Ny5c4XGknneuvM3JyUkeHh4KCQnR1K1lT9fvvvv+c7Ju9aeZuDg4Puvvvt
uPfHEEzp9+nS+9iEhIbt/735+fnl06+zs30h5+FNGU1b948GQwGtWvXztqhAC
jAte8p1267du2ydogACnC9e
/fFF1+0dnglIu+ztZ2dnU6dOpWvdVvPiWvPiMjQy4uLjIYDBo5cmSh41Vuf
r8i8dnglIu+ztZ2dnU6d0pWvPiMjQy4uLjIYDBo5cqQVIgSQh+8Adz4HaweAq7766iv16dNHTk50Gjx4sJo
0aaKcnBx99913/uzk5p13
999m5ZjfAqbrf+Yx1u3t0Ix1tiWT+8Wa8cXJo53Dzc0tX3IeqUx
p0vr68uX76sXbt2lbt68PLb+uPNVeIBY8cP0/eeZVT91MY6HP9Ub6YXgiMhFf13a9f39Ixa9FbiVO/ihlnLpi

eQTVfDCCxb1MTExpRoHgMKV+8A5Ql1JLhtw4sQJzk/u+Qe8fx4hqxKPSobsaY8ImTs1ej/gZDYn2rN1OXQv6qbc
du7dF2Y8//qiuXbsqIiJCv/zyAGuSMSyOPbbqNOTsWdiWnbLVbDWiqTPvvqsaAX/z7exLYzkfabfOUG6sX85Dlvi9PTHcb1fjiqTp9Ua




```

bytnZWffcc48++uijfMefP39eY8e0lY+Pj5ycnFSrVi0NHjzYYvjz2bNn9dRTT8nDw0POzs5q3ry5li1bVuDjn
jFjhub0nat77rlHFStWVNeuXXXq1CmZTCa9/vrrqlWrllxcXNSrVy/9+eef+eLZsGGDgoODValSJVWpUkU9evT
Qzz//fNv/TigbiuN5lPf8/9///qcWLVrI2dlZ99577033vFi9erVatWolFxcX3XXXXRo4cKDF0PolS5bIYDBo3
759+Y6d0nWq703tLdp///33euihh+Tm5qaKFSvqgQce0I4d0/Id+913361NmzZydnZWvXr1tHDhwpv9ZyvR2E6
ePKnhw4erUaNGcnFxkbu7u/r06ZNvHqS816Jt27Zp+PDhqlmzpmrVqiVJunDhgsaMGWN+nalZs6a6d0mivXv3F
vkxwrYU9fnx3Xff6fnnn1eNGjVUtWpVDRs2TDk50Tp//rwGDx6satWqqVq1anrhhRdkMpksznHtnFzXGjJki06
66y5duXIlX13Xrl3VqFGjIj++/v37KzExUYcOHTKXpaam6ttvv1X//v0LPCY7O1uvvPKK6tevLycnJ9WuXVsvv
PCCsr0z8z2ekSNHavXq1br33nl4uKioKAg7d+/X5K0c0FC1a9fX870zgoJCSl07rGEhAS1b99eLi4u8vX11YI
FC247phUrVqhx48ZycnLSxo0bJUmrVq1SqatVKVKFbm6uqpp06Z69913b/rfEigJN/oOIEl//PGHBg0aJFdXV
1WtWlVDhgzRjz/+m0/zuiQd0nRIvXv3VvXq1eXs7KzWrVtr3bp1pfRoUBiSXDbgyy+/1D333KP27dvfvfVPsrV67
o3Llz+bb09PQSjvQfvXv3louLi/73v//dsG3em2y1atXy1RmNxgIfy8WLF/019fX11eDBg7V48XKd0XPmth8Dy
q8VK1YoPDxcjo6O6tevn44ePZoveSVd/eX17bff1rBhw/TGG28oKSlJ4eHh+T4QG41GdevWTe7u7u7poxY4YeeOA
BzZw585aHGG/atEm//vqrhg4dqvff1+PP/64Vq1apYcffjjh/jicuzYMfXu3VtdunTRzJkzVa1aNT3xxBMWC
aPMzEwFBwfr/fffV9euXXfXuu+/q2Wef1aFDh/Tbb79Jujo8JCQkRB9//LEGDBigt99+W25ubnriiScK/HC7YsU
KzZs3T6NGdcO0bds29e3bVy+//+kK4ceO0bds29e3bVy+//LI2btyoiRMn6plnntGXX36p8ePcfq0ePHqpcubKmTZumSZMm6
ZdfftF9991XbicWvh0lp6fne4+4NuF88Kgee+wxde/eXdHR0XJwcXJwcFCfPn20adOm68a2d0lS9e3bV/b
29oq0jtbTTz+tmJgY3Xffftp//rykf94vC/ogvWLFCCoWEh0juu++WJH377be6//77DDDFDD//r
eCDD+qHH34wH7d//3517dpVZ8+e1auvq4a4v0b1sSse3evVs7d+7U448/rvfee0PPvusNm/
erJCQEF26dCnfdYYPH65ffvlFkydPNs/X90yzz2r+/PmKiIjQvHnzNH78eLm4uOjgwYNFenywYNFenywroLu3aI+POaNG
qWjR49qypQpeuSRR7RoOSJNmjRJPXv2lNFo1NSpU3Xffffp7bff1scff1yk+AYNGqQ//vhDX3/9tUV5XlLqZkc
a/Nv999+vWrVqaeXKleayTz/9VoJUrV1aPHj3ytc/NzdUjjzyiGTNmNmqGfPnnr//fcVFhamWbNm6bHHsvXPi4uT
uPGjdOQIUPO6quv6uDBgwoNDdXcuXP13nvvafjw4ZowYYLi4+P15JNP5jv+r7/+0sMPP6xWrpVpp+vTpql5
77jl9+OGHtxzTt99+q7Fjx+qxxx7Tu+++Kx8fH23atEn9+vVtWrVNG3aNL3118sKCQkpMGEPlKYbfQIzc1Vz
5499cknn2jIkCF68803031ZKSoiFDhuQ7188//6zAwEAdPHhQL774ombom0KlKlSopLCysy0/HKGYmWFV6erpJkql
Xr1431b5u3bomSdfVq9ebW6/ZMkSyKSZLp999/L/Q6W7ZsyXeuazVtvT7zW3ZE99LPb54NJiHBZw7tDGwgDN7
L///rvp1KlTpjVr1hqhKhhcnJyMp06dcri+AceeNXm/PS+KFr0zbgrl+UJhAxeesjEbz+HnV/MY/gFYbrW9NY6pqaPF
9JkmmTZs2mUwmkyk3N9dUq1Yt0+jRo8NPWiYAcne3x8dWenS7hW4sZpyXcvsb+rWxXQAW5Q2HKOmPM++Nm6sb8T/
hMQEGBq1aqVeT/vntqyZYtFu7xrLVmyxFx26A0Di4x8njHlZqlJ58g8Lm0BUhUNd4O4ap4ZNEyZyc3+8E2u3KLgTJ+y15MbmnKKUGrnrxHLmSu6EF9JS1
mJycn07hx48xlkydPNkkyxcTE5Dtvbm6uyWQymZxnm2SZFq+fLm5LicnxxxQUFGSqXLmyKSMjw+Jx16hRw3T+/
Hlz26ioKJMkU/PmzzU1Xrlwxl/fr18/k6Ohounz5sslkMpkuXLhgqlq1qunpp5+2iCM1NdXk5uawrxlT95ztaD
NycnJZDLd/vPIZPrn+f/555+by9LT001eXl6mgIAAc9m193BOTo6pZs2apiZNmpiysrLM7davX/eebLFd
b29vU1Go9FctnfvXot7Pzc319SgQQNTt27dzPePTyXT19cDX19fUpUsXc1lYWJjJJ2dnZdPLkSXPZL7/8YrK3tzc
V9aNdccdW00tXfHy8SZLpo48++Mpfl/f+97777TH///bdFezc3N90IES0K9Dhg06537xbl+XHtcy4oKMhkMBhMz
z77rLns77//NtWqVcv0wAMPVJV17Jd8689z+j0WiqVauW6bHHHrM47p133j5KBGnt/+et0P+d+frce
PH2+qX7++ua5NmzamoUOHmmP693448/NtnZ2Zni4uIszrdgwQKTJNOOHTssHo+Tk5PF+/fChQtQtNkkyenp7m9
1eT6Z/XwH+3zfvcPXPmTHNZdna2qUWLFqaaNWuacnjybikm0zs7088//2zRdvToSZXV9d89zZgTTfzHeDzzz8
3STLNnj3bXGYOGk0PPvhgvs/rnTp1MjVubkd1MjVt2tTiMkTWMHk3PQx7dq10
6ZNm/JtM2bMKKkwC1S5cmmVduHAhX3kjBaz3hj8tG
gQYO0aNEipaSkFPdDQjmwYsUKeXh4KEP4BQjKhgdmE1RBE7vPvv0Sd9Ke3XcP+ZJzeL0ol73ZJj3xPEJVEQ17xhKiWPwLEHJuQYDfHR3nMsMk3XCucF+CJEtpzzw9IC+MWTpgnPNNhiKFnX0p8LAI9c+mH0ZoSNDVIHe23dKU5cr/IWVu2duL++xB+L5t9YbzbWyaTw/n
xwcXGC7m+Hi4mL++/Llyzp37p37pwCAwMlqcSG69x7773mxydJNWrUUKNG9x+eff+h2t1cPGj9WiUTPa+A84+Hd3xjzyN/Ap3WDkz93Dkh1O/KvsLdaucZtpR0GWfV0/y4XT92ahxtZvLtVWRzr3U0f2QGQNEb35Rh7jnbUepUF3bp
3v/+Vp6en+vXrZ66rUKGCnn/+eWVmZmrbtm0W7fv27StXV1fz5aCCnnj+rZO9j4UbDZKPczyEYgKu4bgC7m4+eWVmZmrbtm0Wx/Xp08dioYm8lWzFvE+mN4y0FcKGJ0iXdjDrA+Y3NOu0YbmJ/WmTY8Vrb2PqDHX2OjDdhqZ7a8V2KbJzTUg0XeSGdprfD+/fu3eTTL1Z8oQVZR17d8SAs
vWz6CVgb2+vdu3aacuWLbfzzwvEbMnfu3HzvERs2bLBoc6vPXTnsAzs0Dgdb8ilXxwXS04rUpO9nMQ8XdQvUq2NhY2TxfwxjbZ8RxnwQ5UHEwSNU6AXmyPwrG/ge7NTUKp5+0/cm9vqmaoaXq0zLWojKX26JtG/XDncUnAqpvQ5Rq
PntXw4cMt5p/p0aOH/Pz8LObSHDx4SM6cOWPxvFyxYxYoVcXFwUEREhSUpMTNTRo0fVv39//fHHHHxY9mzt16qTt2
7crNzdXRqNRX3/tcLCwlSnTh3z+fz9/dWtW7fcN7cb/4Neozhjkyxfv65cuaI//vhXVX9WqVQt8/Xr66afzRN
YtWpVff/99ATaLuMKuneLIyzwmnz208v0tV72GQy6amnnjKKX2dvbq3Xr6kV+z7Wzs90AAQ00btb06i8+zK1asU
Pv27Qtc80Jm90/fX8e0HdPu3bvN/y/eyvIWdrn0v69Spk3xfMz7ea91ERERFt8lq8sq
v/TdxcHDQsGHDzPu0ojo4aNmyYzp49q4SEhFuK6YEHHHtC9995rUValalVdvHjxhr1hgdJ0M98BNm7cqAoVKujpp
582H2dnZ6cRI0ZYnOe3337T5s2bNXjwYHd3R/HRqdDV6cRIOZYn0vPP//+//Ut99+q759+9eumoChQsWPc27deumgDFxPNW5urqKkkFJosKc9ddd5Dbt28vj/yaN
5ykt7BZnMzMwCk3Nz585Vw4YN1Z6erg8//8//DFbt28vdCW3SpUqFFhYruf111/Wxx9/rLfeeeoux/SgSo9GoVatW

WPHjjpx4oS5vF27dpo5c6Y2b96srl27msv//QVS+mfI7V9//WVR7uzsrBo1auRre227m/Xnn39qypQpWrVqlc6
ePWtRV1LDkq99rFL+x3D8+HHzl97CnDx5Ug0aNJCdneVvKP7+/ub66103L1FRu3btAsvz4jl69KgkFbjCq/TPa
yvKvrZt295w8upbfR7lqV+/vsUXaUlq2LChpKtD7j09PfNdM++5XXC8PX5+fvruu+/M+126dJGXl5dWrFihTp0
6KTc3V5988ol69eplfh/NeO04XNCQiT3p6urKzs5WVlaUGDRrkq2/vRO0dFas
mSJTp8+bTHEuqDXr4ISCd0nT9eQIUNUu3ZttWrVSg8//LAGDx6se+65p0iPPDdZV0L1b1O0dHUe7tW3nPHTx4sKZ
Nm6a1a9dq8ODB0nz4sBISEgqcp+pmBQQEyM/PTytXrlTVqlXl6elZ6HvV0aNHdfDgwXyfIfJc/jXJc+xngdl/rvL298
y3u8O/XusDAwCLHVNA9PHz4cH322Wfq3r277r77bnXt2lV9+/bVQw89VO0A5gZJ2s98BTp48KS8vL1WsWNHi+Gt
XYDx27JhMJpMmTZqkSZMmFXjPmof8o3SR5LIyV1dXeXt768CBA9YO0pUiuXLmiI0eOFLgU9L8/1ISFhem++
+5T//79dfjwYVWuXPm2r33PPfo4MCBWrRokXn+DuBmfPvttoOpJSdGqVau0atWqfPUrqywvWqfPUrVqywSHIVtgrhvz+UX6/
dv137BTrPtb3HJKlv377auXOnJkyYoBYtWqhy5crKzc3VQw89Z04tUdxu9rGW1nVvFE/ev8PHH39cYAxp3kXv3WtBTp7
K5bfR6VFnt7e/Xv31+LFy/WvHnztGPHDp05c8Zizp+85/Tbb7+tFi1aFHjPypUrW+y7uLgU6BEuyQHIytU+zr/NZDAa8ZNmn
GQUFBcnNzk8Fg0OOPP17g69e/e/bk6du3r4KDg7V27Vr973//O0DBWr58uRwdHf0tBl5U/fv31/z581W1l5hU99thj+X74yZ0bm6umTZvqnXfeKbD+2uR
dzX9957r1q1aqXly5dr8ODBWr58uRwdHf0tBl5U/fv31/z581W1lShU99thj+X74yZ0bm6umTZvqnXfeKbD+2uR
VabzWFTWmgu7hmjVrKjExUV9//bU2bNigDRs2aMmSJRo8eHC+hWiA0lDU7wA3kvd6NX78+Ep7UV+bGEPp4VuaD
QgNDdWiRYsUHx+voKAga4dzU9asWa0OsrKwbDo3Im4y3Y8e0mjNnTrElpV5++WUtX75tata
sqblz5+ari4mJ0dq1a2/r19vryesFljcZdZ5rezb99ddf2rx5x5s6ZMmaLJkyeby/N6Ulhtvb1bpiQr1u3rn766
Sfl5uZafKjPjPW2mqbt26xRaLdPWDdFF7gwLXyvtF9t/J6CNHjkiSxdcgJFfDhfL43Dhw/ne64PuZYMiQr5i3c5a57PLly/le+
27Ey8tLw4cP1/Dhw3X27FvSb775JkmuEmuqMFRsdMntqUnk3V4nUaVHHyb4lxbxXbjaDnBS7ERNqrPGvr6Jlh2yTkpvZP9+bnrn0r97
7fc6RS
...

```
8uXPnzZun8+fPy9PTU4899pheffXVYvtxArhZRfk0cP78eX311VcaPXq0li1bJjs7Oz366KN65ZVX1KFDB4vVl
O+9917t2bNHU6ZM0dKlS/XHH3+oZs2aCggIKPSzC0qHwVTas68CAADYCB8fHzVp0kTr168v8Wud03dOXl5emjx
5cqGrMVmLLccGlLQvvvhCYWFh2r59u4KDg60dDgAbExsbq0cffVTfffedOnToYO1wcA0k0gEAAErB0qVLZTQaC
+3BbE22HBtQ0hYvXqx77rlH9913n7VDAWBlWVlZFvtGo1Hvv/++XF1d1bJlSytFhaJguCIA3AEyMzMLnBD632r
UqFHoMuMASs63336rX375RW+++abCwsIKXbGxOKWnp+f7oaH4tT09Pq8QG2IpVq1bpp59+0ldffaV3330332qCN
3sfAbhzjBo1SllZWQoKClJ2drZiYmK0c+dOTZ06VS4uLtYODzeB4YoAcAd49dVXNWXKl0u2OXHiBF9ggWuUxnD
FkJAQ7dy5Ux06dNDy5ct1991319i18jzxxBM3nLPSZDJZJTbAVhgMBlWuXFmPPfaYFixYYLGqqHTz9xGAO8fKl
Ss1c+ZMHTt2TJcvX1b9+vX13HPPaeTIkdYODTeJJBcA3AF+/fVX/frrr9r9dtc99991lMmAngzvXLL7/ozJkz123
TuXPnUooGKJu4jwCg7CHJBQAAAAAAgDKPiecBAAAAABQ5tncxPO5ubk6c+aMqlSpkm/yRwBX5364c0OGCvL29Z
Wdne3lq7mGgcNy/QNnGPQyUXdy/QNl2s/ewzSW5zpw5o9q1a1s7DMDmnTp1SrVq1bJ2GP1wDwM3xv0LlG3cw0D
Zxf0LlG03uodtLslVpUoVSVcDd3V1tXI0g03JyMhQ7dq1zfeKreEeBgrH/QuUbdzDQNnF/QuUbTd7D9tckiuva
6arqys3N3AdttqNmXsYuDHuX6Bs4x4Gyi7uX6Bsu9E9bHuDkQEAAAAAAIAiIskFAAAAAACAMo8kFwAAAAAAAMo
8klwAAAAAABTCx8dHBoMh3zZixAhJ0uXLXLlzVixAi5u7urcuXKioiIUFpampWjBsonklwAAAAAABRi9+7dSklJM
W+bNm2SJPXp0eSNHbsWH355ZdavXq1tm3bpjNnzig8PNyaIQPlls2trggAAAAAgK2oaLGxf5bb72levXq6YE
HHlB6ero++OAPDrVy5Ug8++KAkacmSJfL399euXbv97gBhojZCBcoueXAAAAAAA3IScnBwtXX75sTz75TzQ75pAwGgxISE
nTlyhV17tzZ3MbPz0916tRRfHx8oefJzp5WRkaGxQbfpHkAgAAAADgJsTGxurz58/6NvvnJEmpqalydnZW1ap
VLdp5eHgoNTW10PNERETIzc3NvNWuXbsEowYKHvGF7IqE0j+xwZBGb8SyYsLasYkAiAPnEk7L9mDRuxQJL6ppLvpS
RBH95IOuECUZsPvoVkKw9aX7Px43DDJzeXnyKH1CPo6yFqgb3PUeqOyMzgtMGmhn+G5hVVdZV5TKe9kDSoVvvqbo
lJWmuLFMAvr3BjWu9jC3rIeN6bfHdlbwOMHK8Nu6wwmnVdg/F38vyaPdc7DnTFSBDBBFy6vMQgIlYExSnjwoSCJCtz
Zs9DfVvfzd3Q48nTu3H4lEngnVBaENfovRKS8sawOrRyNFJZvnt3V/3E8b2UTK+p1mqnJXfAbHQZCn0+IFGzsfhkF
```

AAAAMo8klwAAAAAAAAo8OhyAWXUjh071LNnT3l7e8tgMCg2NrbQts8++6wMBoNmz559w/POnTtXPj4+cnZ2Vrt
27fTDDz8UX9AAAJRT27dvL5H3bQAA8A+SXEAZdenSJTVv3vyGc++sXbtWu3btkre39w3P+emnnyoyMlKvvPKK9
u7dq+bNm6tbt246e/ZscYUNAEC5dPHixWJ/3wYAAJaYeB4oo7p06aKIiIjrtjl9+rRGjRqlr7/+Wj169Lj0d9
55x09/fTTGjp0qCRpwYIF+uqrr/Thhx/qxRdfLJa4AQA3x2g0Ki4uTikpKfLy8lJwcLDs7e2tHRZuUffu3dW9e
/frtinq+zYAALBETy7gDpWbm6tBgwZpwoQJaty48Q3b5+Tk6ZUffFh
sAIDbExMTo/r166tjx47q37+/OnbsqPr16ysmJsbaoaGEFPV9GwAA5EeSC7hDTZs2TQ40Dnr++edvqv25c+dkN
Brl4eFhUe7h4aHU1NRCj4uOjpabm5t5q1279m3FDQDlXUxMjHr37q2mTZsqPj5eFy5cUHx8vJo2barevXuT6Lp
DFfV9W+KHJgAArkWSC7gDJSQk6N1339XSpUtlMBhK9FpRUbGysA
gMDVblyZQUGBio2NlahoaEaP368jEajtUNFMbrV921+aAIAwBJJLuAOFBcXp7Nnz6pOnTpycHCQg4ODTp48qXH
jxsnHx6fAY+666y7Z29srLS3NojwtLU2enp6FXsvJyUmurq4WG
wDg1sTFxSkpKUkvvvfSS7OwsP6bZ2dkpKipKJ
O6cUFxcnJUiREm4lfdtiR+aAAC4FhPPA3egQYMGWcytJUndunXToEGDzJPKX8vROVGtWrXS5s2bFRYWJunq/CC
bN2/WyJEjSzpkAICklJQUSVKTJkOKrM8rz2uHO8OtvG9LV39ocnJyUnwAAAoMOhyAWVUZmamfv31V/P+iRMnl
JiYqOrVq6tOnTpyd3e3aF+hQgV5enqqUaNG5rJOnTrpOUcfNSexIiMjNWTIELVu3Vpt27bV7NmzdfHixet+wAY
AFB8vLy9JOoEDBxQYGJiv/sCBAxbtUHZkZmbq2LFj5v1bed8GAADXR5ILKKP27dunO0NBQ835kZKQkaciQIVq6d
OlNneP48eM6d+6cef+xxx7TTbWrTQxoOb8O1GDwAoGcHBwfLx8dHUqVMVGxtrMWQxNzdXOdH
R8vX1VXBwsBWjxK3Ys2ePOnbsdu1FdffaU1a9bI3t7e2qGiiEJCQm77fRsAAbEh4ervHjx2vWrFl
av369udzBwUHjx49XeHi4FaMDAACwXSS5AAAAbEhMTIxmzJihJhHj16qHv37uaeXBs2bNCGTBhIogsAAJ
LkAAABshNFo1Lhx4xQaGppvTq5nn31WYWFhGj9+vHr27MnQRQAAAAC7zG8r+GkFNx0EfBXOAQRQTnBvOmDTGfxTDbO5vPavtTtCo+xdcyYO68amUE5dI2NwtvAhG8CnlKvMiz7wAPtsP5TFD
KdHbVYNrXYNw/XCDFAkEe2QVzGxyVe8G9U/QsZw7qv0aTKX6xQDwu+3Ndg5W8VgOJGZA4/tGfnzG2
iGH3DLwaWIWf3OE9hQ2mKGX/hiiTXyKMF7c6yTEB5Lfu+jeidd1G48DNW+LtV6gF22k97q0h2s3uCHtLuPnVQdrYL
YWrmAKF9UxD8/aweq2pSyrZ
qRnOi0kUvVA9mmq0qMubu2l9tF3ePjtZ6hA+TpdUv1nd0rsz91y0JdD1xZZ3VMVlsT9oL
T1PhpWx6C2EVGhVd2KATJuOAJqHb
/ckMC5grsqeVeQxWwQZcqnCq4CaRo6h9ogFsKVjAVH+T9kOM6

```
d3VOuLi5q2rSp9uzZY643mUyaPHmyvLy85OLios6dO+voOaNWjBgonOhyAUA5cu1E1gBsy/Dhw+Xg4KCXX35Zf
//9tOXd33//rcmTJ8vBwUHDhw+3UoQAUP789ddf6tChgypUqKANGzbol19+OcyZM1WtWjVzm+nTp+u9997TggU
L9P3336tSpUrq1q2bLl++bMXIgfKHJBcA3EHOnj0rSdqxY4fCwsIshiuGhYVpx44dFu0A2BZHR0eNHTtWaWlpq
lWrlhYtWqQzZ85o0aJFqlWrltLS0jR27Fg5OjpaO1QAKDemTZum2rVra8mSJWrbtq18fX3VtWtXc69ak8mk2bN
n6+WXX1avXr3r3Ur3Ek1e1b99eBw4c0Jtvv
mnRDoDtmT59uiZMmKBz585p2LBhuvvuuzVs2DCdO3dOEyZMOPTpO6OdIgCUK+vWrVPr1q3VpO8f1axZUwEBAVq
8eLG5/sSJE0pNTVXnzp3NZW5ubmrXrp3i4+MLPGd2drYyMjIsNgC3jyQXUEbt2LFDPXv2lLe3twwGQ75fiV599
VX5+fmpUqVKqlatmjp37qzvv//e4hxXr69dxtLv2Le3twwGQ75fiV599
VX5+fmpUqVKqlatmjp37qzvv//+uud89dVXZTAYLDY/P78SfBQobsHBwfLx8dHOnTt15MgRbdmyRStXrtSWLVt
O+PBhxcfHy9fXV8HBwdYOFcB1BAYGqlatWhZltWrVUmBgoJUiAoDy69dff9X8+fPVoEEDff7553r22Wc1Yt
...
```

AB6cgHAHeTEiROSpJMnT1okuCTp1KlTOnnypEU7ALYnPDxcn3/+uerUqWNRXrduXX3++ecKDw+3UmQAAAC2jZ5
cAHAH8fX1LdZ2AKwjPDxcvXr1UlxcnFJSUuTl5aXg4GB6cAEAAFwHSS4AuIMObNjQ/HdGRoYSEhLMX5BbtWolV
1fXfOOA2CZ7e3uFhIRYOwwAAIAyg+GKAHAHee+998x/N2jQQEeOHNEDDzygIOeOqEGDBgW2AwAAAIA7AT25AOA
O8tdffOmSevbsqQObNmjYsGHmOgcHB4WGhmr9+vXmdgAAAABwp6AnFwDcQdqOaSNJOnjwoC5cuKBZs2Zp5MiRm
jVrli5cuKCDBw9atEPZMX/+fDVr1kyurq5ydXVVUFCQNmzYYK6/fPmyRowYIXd3d1WuXFkRERFKSOuzYsQAAAB
A6SLJBQB3kJkzZOqSjh07pt69e6tdu3aaOnWq2rVrp969e+v48eMW7VB21KpVS2+99ZYSEhKOZ88ePfjgg+rVq
5d+/vlnSdLYsWP15ZdfavXq1dq2bZvOnDnDKnxlXFZWlkaOHKlu3bpp5MiRysrKsnZIAAAAANo3higBwB3FxcVG
vXr3OxRdf6KuvvvtJXX32Vr02vXr3k4uJihehwO3r27Gmx/+abb2r+/PnatWuXatWqpQ8++EArV67Ugw8+KElas
mSJ/P39tWvXLgUGBlojZNyGsLAwffHF+b9//3vf5o7d6569eql2NhY6wUGAABgw+jJBQB3mNjYWPXq1avAOr4
g3xmMRqNWrVqlixcvKigoSAkJCbpy5YpCQkLMbfz8/OTt7a24uDiRZJLwWhVkdBDKzlu3bpp5MiRysrKsnZIAAAAANo3higBwB3FxcVG
vXr3OxRdf6KuvvvtJXX32Vr02vXr3k4uJihehwO3r27Gmx/+abb2r+/PnatWuXatWqpQ8++EArV67Ugw8+KElas
mSJ/P39tWvXLgUGBlojZNyGsLAwffHF+b9//3vf5o7d6569eql2NhY6wUGAABgw+jJBQB3mNjYWPXq1avAOr4
g3xmMRqNWrVqlixcvKigoSAkJCbpy5Yo6d+5sbuPn56c6deooPj7eipHiVlyb4Pq3L774Qm4vQmFhYaUbEAAAQBlR5
CTX9u3b1bNnT3l7e8tgMGHjSNrSpCQDyNdzk5XtThUzfEbaDDh6HvWbVTRgpdhFGcDSYR8dGCWQRHcQrH2MOrx9yC
lS5dIcJVx+/fvV+XKle5eXk5KRnn31Wa9eu1b333qvU1FQ5OjqatWqatWqFu09PDyUmppa6Pmys70VkZFhscG6srKyz
Amuhx9+WPHx8bpw4YLi4+P18MMPS7qa6GLoIgAAQH5FHq548eJJFNW/eXE8++WSBc31Mnz5d7733npYtWX19
NmjRJ3bp1Oy+//CJnZ+diCRoA8I9Lly7p0KFDBdb169dPSUlJ8vHxU86XxA/Pz9VrFixpEJEMWnUqJESExOVn
p6uNWvWaMiQIdq2bdstny8mJkZTp04txghRnIYNGyYvLy/p7pgm8M36i8/Pb9+LHqnI+F4Su6ZUkx/jtO8wj/79zbK4d
q/vz5Vo4WAADAthQ5ydW9e3d13769wDrDMDRnzhy9+uqr5uTK8uXL5eHhoXXr1mnAgAG3Fy0A4IZ8fHwU
rd1joSEBLVs2bKYIkJJcXR0VIOGDSVJLVq0UO7du/XOO++of//+ysnJUVpamsVoroSEBHl6ehZ6vpiYGI0bN8
66edrd4uPjUy8PAiw2Ub2Tt2cJsRF9p2gdz5a/n2lnTnVjMhwKcr+TeJoJ3VmKrWPWlvtUnXHgOKJEQdOwXumThw
6ixnCKzA0b1BY08Dw3F7ZyxpPw0H8yntTK9OfbRHeh2NiiBgA6gQm1A20wMfFbzdRXx47k57cnhk8l4+LkFtbkbrDBsNpa
bNfkaS8ZQIL6zjf9DAjQvc0Bz/BZaZr3ZNZOprMkfDd3PqPaoGEI+M5HETOaSgTUwKvtPQvqY2Bds5Wgem4KDOTXvYb
ywOZ6PqdTqLuTXM/3eS3pbyP+cLsMOZGlKpQeNA1ODKlKI3eoOKajGQrQDd+3Erx5ApP7PVUxjX81ZCOYYpy2MDLvWA
DKcvjtT/EmGj9j0UIBKLwcm+2dXbqHJKkdV1XNKrXl8P7sPnmdb1B0z9NvdSSWaXU8jdLzhn06JtLwHKdhaaTxkzpMbtN
zOWdw6T/rxbuadsfafJt1IgoEBrnSvoWQX58lcWFUEfoRQg5wZTW5v7pl1kxSW/fI2J3lS6qCkIdhcb+AzZPP5HJ
kvh/0jcHtNXUPpr5sWQcj2tQwdGsbkX9Mhj+pkdZBl/J96hsEg9S/fzRRt43DE7Zp1ZnIMQ/gWx6eFY4zSSJW0tA
9ELZYN0nFUwZqO3vubSpNNLDLOoqHHo5BymCHQhdJKJ0Zvo7i12Q8UJ3Vx+BVljd1LKo+OUhl/lBYdc4dMjprvV
hPzUF+HSvS8x/GmAadsoXE3zSOALuB5KbBwpDcm1vq1XrM9l0Xn/IqQg80tRr8Z19SaMD9/ahGoaKvGuZxrq9Ll
/e2jgGjVdFMEbK3mFwYQlM1/pi5NGj2qz5cpnYDYkk6GGXuqnoBxTCqzUXwjwh1fySz+NYVVSpSTfc97z1Jb/3F
kn3SLkQiZ3GN/h35vXqD1fSTS1f4FNcEUnHYBXNGUJUGmUjFGrL0Sx6uiMbVlJRcVYlfRpeOQZaHWbb7mjDadjZ
9sJfxH2fqSmOiUOtVIFDuAcUpYd0KLyJgjmOjZn+sGUU1iuqfbD2hsa2f/Bk6a7F7MbMO1+u96pXU9N5vO1/A5Rm
f/fdxB1QGOa99oZOVgg8/MO/qFVXO+GKmmYz1bDqL71Ri7qq4GlWrMlPOsmVuhrFzsHuW4SsJO75ddk9pA5
ZpXfw07O8dW2LnF/zlbkdG+Hp45q7SmV5f6B2c+NKt3mtHpyWpmY7p8hkUYU5Srh5XMjTHzX+z0yfDLwvvebCAQ
cFgYiwFPHvfNQdmKA/RwBKV3SlTEWGg/hgDnndo/5sIFOB6YnLc8TvGkNvGaqLj7hCbzzAHBcUqOO5BB9y7LNfcnT
LEK0Ux2Lw8jsQyWWTRQ8vGHlAYDeGmYpHlP0n8QyBQq/3Qs3L2X6OQxVXw1+GPHpMqAj/2kkvbYNq2bAjHmAa9pFiUF
Zc+NDvTbaKoFYj4Y1lBwmnpS4FUHgvAfqRfUGS8vI8Tv9rA8wAWWZJnL/Pl8SXqG/Y+MNJgrJhhaxXsu1XZG0Cjl
wX5x28TVDUqL5VDXzdVdI4s3hMGmAOp0p2xP6Sg97EdD0zK4XgwWfTsoVjAz7J0lKzf6t8cOqZ2Fv2IuZv0N8PNA
zZ1ZbNsyyNqxI4YDo4J5fsmPMuKgU5PBcOzFGrKnz4q8ECXCsNy28Az5H4kUeYhMxYUZlDPMHXvwfj4RRLAv39vfq
4VRmqa3Fi+5wKVcm7NOn+WGpAyyNxL0k+Wtp+PMyTt0LG6A2ETSWzYZsHhv2kdxNmfNsyvgkw95yzRkq2Bz8tAVd

FBARo8uTJsre316FDhxQREaGGDRuqZ8+e+uOPPxQXF6fGjRubz3H8+HGdO3fOvP/YY49pxowZmjx5slq0aKHEx
ERt3Lgx32TOAICSOa5dOOmSg4OD6tata1Hn4+MjBwcHi3YoO7Zv366ePXvK29tbBoNBsbGxFvUmkOmTJO+Wl5e
XXFxc1LlzZx09etQ6wQIAUEbRkwsoo4KDg2UymQqtj4mJueE5kpKS8pWNHDmS4YkAYCULFy6UJP39999q2rSpJ
kyYIBcXF2VlZWnjxo3m1+2FCxdqzJgx1gsURXbx4kU1b95cTz75pMLDw/PVT58+Xe+9956WLVsmX19fTZoOSd2
6ddMvv/wiZ2dnKOQMAEDZQ5ILAADARhw/flyS9H//9v9enOdr6+vFi9erKefftrcDmVH9+7d1b179wLrTCaTZs
wLrTCaTZs+erZdfflm9evWSJH3OOUfy8PBQbGysHn/88dIMFfQCAMovhigAAADaiXr16kq4Pfbv36+wsDA1lbdp
UYWFh+umnn5Sbm2vRDneGEydOKDU1VZ07dzaXubm5qV27dtaXubm5qV27doqPj7diZAAAAlC305AIAALARw4cP14QJEzRixAg9/
fTT5vL9+/erSpUqcnR0lIODg4YPH27FKFHHcUlNTJSnfHJgeHh7muoJkZ2dbdbrJqckZFRMgECAFBGOJMLAADARjg
6Oqpu3brKycmRJPn5+SkiIkJ+fn6SpJycHNWtW1eOjo7WDBM2Ijo6Wm5ubuatdu3a1g4JAACrIskFAABgI7Kys
nT8+HEZDAZJOqFDh/T555/rOKFDkiSDwaDjx48rKyvLmmGimHl6ekqSOtLSLMrTOtLMdQWJiopSenq6eTt16lS
JxgkAgKOjyQUAAGAjJkyYIOnqnFw1a9ZUSEiI7r//foWEhKhmzZrmVXXz2uHO4OvrKO9PT23evNlclpGRoe+//
15BQUGFHufk5CRXV1eLDQCA8ow5uQAAAGzE4cOHJUlVqlSRi4uLtm7daq6rW7euKlasqKpWpKXpjboezIzMzUsWP
HzPsnTpxQYmKiqlevrjp16mMmDF644031KBBA/n6+mrSpEny9vjZWWFiY9YIGAKCMoScXAACAjcgcLF27NghSfrss8/MCY7AwEB99dd2R
1IuD1+1++ZnrL1++rBEjRsjd3V2VK1dWRESE0tLSrBgxUH6R5AIALAR9vb2mjM2dVNhsv+Ov8UP6T
/2dnZXXd1PgBAyXFwcJCnp2e+8vT0dH3wwQdauXKlHnzwQUnSkiVL5O/vr127dikwMLCOQwXKNXpyAQAA2AijO
ahx48apZ8+eSk9P15YtW7Ry5Upt2bJF58+fV8+ePTV+/HgZjZjUZrhwoA5crRo0fl7e2te+65RwMGDFBycrIkKSE
hQVeuXFHnzp3Nbf38/FSnTh3r7s7GxlZGRYbABuHz25AAAAbERcXJjyS4pLO0ySefqEFgoJCbGoj4qKU
vv27RUXF5evDgBQMtq1a1a6el5eqUaNGSkBGSklJOZQpUxQcHKwDBw4oNTVVjo6OqlKlisUxHh4e
lSglHDpQ/JLkAAABsREpKiiSpe+/eKiSpSZMmysrKjOoQJE3T06FE1aNBQ8rpB7/9tpo0aWLRDesF6N36+u1mzZmrXp3q1
q2rzz77TC4uLqa6ahB/8gdI8sgpIHr1H6t8n9L8TE+RKKTkjQjq2PASknWmv/WiPjmuZJx9zofHgAJhbPCsvbYMmNcD9WaKJ0bgd/V
0ZQOddK6FxW+IzVy4hEP4Q5o+elSFOrsaGQv2uMbFaSqqKj0oAAAAbF3hR6wAE+iESCsbUS0tsbbe/PinIGGKAoKMuSm0klLLZa54kZSXYU5SDeLzvJAAHBoAbuWoC5ydXWWeC2VN8VGHWs7wxWNaNG5vKStQJpvl+hfBTk4APS1+ABnkiUECE9C1ovHE1Co5ZDBzSWQhnZlD3ZYdrDRnV7nWm0SmamUoeQOYEnDP8xyaC91o6ic6P5kzqKN4zeIcjFV1vA2cAOnbLvzLsCIEiHPYIUz17nHrj3zrz4BgS5QBbEg4EVCtMsVTlMkvQmwsKdkaKPK8wH8WRbEhlp1Qq/+LBnfTGNMGdtV9UNkAAGDbYt1DMiVbJvhjFqHvhTxnBxhy5kOuETpFw4zwbDgEb9QM25zOaeWT+NB3ZCBW6DmMTOOqBUyFEupN18bZFfWd0yaugnKc7drqkyTBuL8rftcu5GKwPmiuUltttK8cGYkkbe4qTeOUY4lnKBCsyx4lYhrj+HTqXKpz8cnbDhnPwHHxmBZkoLcv3Ct3jOQsY6Lt6fvs23XaD2/eyfh/N+/gAhzIdiJNkpMR17lUfGfr3c7VeyGE4LCzOWGgJBmSYyfuRDmrulCCD7AUkRSAKKGhwsKbLbK8DwDEHpeE8p/LNYwgA9WjnD7iutjrmVB3LxiCq8gldB5uR4JI+QqAI5BQEELIYCqaVOMYWvCYhxzR9LhaZ7FCVEoTcEJD0HLuWuc1sTkGVfhnR6SAjeUnCeaWW7v9PjHfxaI2qm4Bzwz1z14ojhH6eyohShdSDm9Xc7e1eT3cSxEoAMFbYoBFSuCYyyzPBcjP37eTsV6tDLUP/vK9gsvjoFKzUXMJBqdnTFGRAvAIsGzUPbfBf+w6s5ZP34a0GOhNV3ymNEh30Tbw7DgIgadJ8lRbWBLzzk3u3DBl9mzPnrvMIZDEA+Hw1uSB2MpdTQ/u6l4TVU+aXH4ydDYRzwqx62hp9jmY6Jp97Ye/1yF5t1CjNmkdpu9KhbmcVu8zfL5bzNU/lYXjKJtmuIyKtOJcTUW1ubDvO9QjJUjj1bb+lCMtzzN1Yqj7MqS26IQtuTpWLl8c7Vz9P1m2lTVFuCZGfZZ37r82KZbN16KxPUtUXhrH4ijRlWn9W4cB8SDxNDmCpOpL0hVcjIKb/ZJO1tuOo8P8mSLvUW92Tx6xu4EXc7hIZCkGaAlxOVrHUT5yXJ2yWWMNgkr7OOvhdSsLoaQ0QMIUtNfM3bKdQe1Y0bxAeodEEKtHEPG7jrp5a9pa2l/d5MF1f1vI5T+lFzFdSsOsOS4Ljg+X5nMuiyL6ciaGfcrt5U3l7R9BSVtuJkGzHdYu8jhRK3W1sm2n4nl8Q0Rn1FEPz4SrnA2T9yPL7Z6Sip4vPzpHn+5c1KcEnIzR0Di+FJTNdUtO0RUmaUJ2YZEzOgeO4sHOVXmYrqKAzRYZeHGwVBsZ3gjmyf8o12Un+wu6W5A9y8cdFpTrD7YeVtXGmI9FvAxuFLMUO4zoBLP8VvYmZYxmh0Fsoov0TALxLRJNdH3TdSV/8qX4bfLnR9RBuzXDeT2oR62hGw47+pgiK6jRp/+hxAy3EtbsUiW0GFvAc/Cj4UFnNUE5gEj35l9kUG9VGOw7m1jQZ2Z8W0fy+eqjWIq2zhfAdv9BjCRYXPc9lCd7WT0Z/eBh4Xa8wArDAD7Wp9VjS3j25LKP/X9+eX7VzD2PJzq/4Gr5kvflPlrRWUUu6Yd0Jw8Bl8/w0nxFQlXGhH7AazUJhpVjdrqHxGKc7TNTyCQyKprSk1KClr0GSjQEAQtDQFmCoIkIzkBuTsH+U/FVIDbSAE0TiQxvlO8cRFU8MaDG6ebugWvZzJ9Js+OEXAeWX6T6WebUKnoqjzrzexlqKPVxqVyjuSiyVY4lkwgiOc8gHMExgo3wzg5VqMZTGVEXA8sXo5EKcbK5Egm35eM6/yr6mi5/aoV9e1YehRY+dW1xdJymnxhJWb8/+OvCWmsC3zmtTfZbLe+/J0bfbJBbjLp4WyJjbiNcPv9UkrIUZzDQK5ea5N3Ek/SzYPh8wAC0QCKe1tB4GKV38qy38HSA30fDgx7bqIxbB5G/tXbt/+ekwMPNfxBvB3b8GJHAp5/2EFoTz8mgQPGm/x7DqdbQlpmPzaJQ9kOxfDTD4q/PSAK0NH5xoWtz3Mog0QgG84QY2N/3eVfTyBg7N3T19ZT/f6DB7/PObmiqg6tPAgI5SDbsAjjT1AUgcJ5IaWIXsUIYTVW5sB/yB9U3C69eFnCxjmtJPXbJSNvhVJOSQqmAKGyP0ZeYvF/MjfrlD/d3ZmZFLBzsgYhWPkRJPpz0zKYeCgLAB5V2rxYAg8X5mLaVGMezNn9BBH6s4Q9txMR4fC7vTvxq+TfBUXHyZbK0oS3qD8
VtdB27dq1O9KlS7Vx4ObNnz9fJO6cUHBwsC5cuFDoMdHROXJzczNvtWvXLomHAADlwm+//Wb++/TpO2rRooU6d

OigFi1aWNT9+28AAABcxXBF4A535coV9e3bVyaTSfPnz79u2+7du5v/btasmdq1a6e6devqs88+01NPPVXgMVF
RUYqMjDTvZ2RkkOgCgFvk5uamCxcuyNHRUTk5OUpMTLSor1Chgq5cuSI3NzfrBAgAAGDDSHIBd7C8BNfJkyf17
bffXrcXVOGqVq2qhgOb6tixY4W2cXJykpOTO+2GCgCQ9N577yk8PFw5OTmqVq2a6tSpo+zsbDk5OSk5OV1//fW
XuROAAAAskeQC7lB5Ca6jR49qy5Ytcnd3L/I5MjMzdfz4cQ0aNKgEIgQAXOuhhx4y//3XX3+Zk1rXawcAAICrm
JMLKKMyMzOVmJhoHspy4sQJJSYmKjk5WVeuXFHv3r21Z88erVixQkajUampqUpNTVVOTo75HJO6ddKcOXPM++P
Hj9e2bduUlJSknTt36tFHH5W9vb369etX2g8PAMq1hQsXFms7AACA8oSeXEAZtW/fPoWGhpr38+bFGjJkiF599
VWtW7dOktSiRQuL47766iuFhIRQuL47Zs2aKQkBBJOvHjx3Xu3Dlz3W+//aZ+/frpjz/+UI0aNXTfffdp165dqlGjRsk+GACApKs
r3BZnOwAAgPKEJBdQRgUHB8tkMhVaf726PEElJSRb7q1atut2wAAC3ITc3t1jbAQAA1CcMVwQAALARN7tASFEXE
gEAACgP6MkFAABgI5KTk81/GwwGtWzW4ZUvXr19exY8e0d+9ecy/df7cDAADVfTkAgAAsBGHDh0Oy/2OymZSQkKB
PP/1UCQkJFsPQ/9OOdwaj0ahJkybJJ19dXLi4uqlevnl5//fWbmn4AAABcRU8uAAAAG/H7778XazuUHdOmTdP8+
f01bNkyNW7cWHv27NHQoUP15uam559/3/3trhAQBQJpDkAgAAsBFubm46fffbm46ffr0TbXDmp3q1auXevXevTolIUny8fH
RJ598re9oh9++MHKkQEAUHYwXBEAAMBG+Pj43FQ7Kx4A+jHYZ4kAgAfR++/Pj4FGs7lB3t27fX5s2bdeTIEUnSjz
/+qO0+++MHKkQEAUHYwXBEAAMBG+Pj4FGs7lB3t27fX5s2bdeTIEUnSjz/+qO+++07du3e3cmQAAJQd9OQCA
ACwEfb29sXaDmXHiy++/PzvrqlEvvvmmBgwYUOgx2dnZys70Nu9nZGSURqgAANgsenIBAADYiMzMzGJth7Jjx
MzGJth7Ljs88+O4oVK7Ry5Urt3btXy5Yt04wZM7Rs2bJCj4mPj5ebm5t5yxGAVVsBxFnFZYqOHsAtSBTL4iC0t0A
xeKfNzBgwct/ltUVapUUYMGDW7pWBSv6OhoxcTE6NChQ3JxIUPuVVQlrOhsTX9hRKfJ8l
169ZN8+bNk4eHhxUjR1EkJSWZ/87rtUxHghDw6v1iaxgn1v6xwDBw685WOPHDlCossGbNu2TSNGjFCbNm30999
66WXX1LXrl3yy+/qFKlSpKksWPWH6quvvvvtLq1av15uamkSNHKjw8XDt27lByq4
s1MTo+y5dKlS7KzsxxkYW9vr9zc3EKPcpXJykpOTU0mHBgBAmUGSCwBsXF4PruXL18vf379Ix2ZlSkpKUk+Pj5
ycXEpOrEHDx7UwIEDb6HGYrfxo0bLfaXLl2qmjVrKiEhQfff/7/S09P1wQcfaO5AXKlXrwwwQclSUuWLJG/v7927
dqlwMBAa4SNG7h06IZpk3jeZT0a/f//9d61evbrA4SZLjXDTtLgmk/bu3StJ8v/XvfAMtw8FSUbh0kKKXpjvJPT/Inp6Z2zW
UuHFj7du3T++++8hJa4cGAECr40Gz5uSwabjdatj6NmvXzRo+qUZaccUkpMlRSgEq4kKpGDSUnVFRvpDcZ0A+eIh4cJLg+++9
dzGz89PderUXx8fIFJLiattr5DHw6pVatWVJjDUOC7BxP4tpbWSsGc4fxXGsx3eFFPGoiCl3IK8+XwDnZmgO7FRRQFuXGNPGSSNlX1vBwRkwdn6VFxVZvpixGyBKNEPMQsXkZuPMIYnxZyNBahCpUAlc7drMVQ+tBJbLHVOTYuFCZI2XRbgQ+Ir
6K+VatWqlChgjZv3qyIiAhJ0uHDh5WUlFSkGwL5+/srKSlJzs7OFnfBBAAAAIDyzGQy6cKFC/L19TXf+L0wJLkAwIaNHz9ePXv2VP369XXq1ClNmDBB9vb26tevn9zc3DR8+HBFR0fLw8NDrq6uGjVqlIKCgtS+ffsbnjsrK0vPPvtsKTwKAAAAAGVRSkqKnuL8Xa
uo5YLM7VixQr5l+Lw/oOHDmnAgAH64OE6pXZNXEWSCwCAMmnJzOIGaukuq0GlBrbqeh+1vLG2nMl6y/SJBrmSfmF7k0NmaRp3+CxenxPaL
556SpGRkapevbvpcXV01atQoQFbJiGeXm5qYP/xQrVqq1OocffUvKjbdH0gT8yXDeVuPaNWF/nNNdBU9nsisuwwHgixEkSwJ1TUUuWHeg3ViRmoU8W3nRo6pDbUB/ltkHX8jQ5hUIYnn1knRn5tc9sxrLNp6jCo0nVfRRWqm9E
1LzVXJgfnUrsmriLJBQBAGTB//nxJUnBwsEX5kiVLFHA9nxJUkhIiEX5kiVL9MQTT0iSZs6cKTs7O0VERCg7O1uhoaEqXLhQqbTF9o9l3HOwFBeJQVyRzOnNwYJHWmsRZpAwJJkcSoSBAdITEoYUTQDOhUfWY+BOY5HazYGNuqgwd
DpIcgEAUAaYTKYbtnF0dFR8fLzi4+NLISIAAACgbCHJBQAAAKBcMRqN+uOPP2SNGWZ/dzYTYRbqPzOSn5OqsWUZhS9FmLn7BTd12RXD+9sgGT1xI6hDB5ly+n9QAAxDr1ynqcuULx/sBZQAABVwaLSvZUS1Wy39MpbELaRWvyZW92FvUyBlDUQlUlJbXT+3j3d1aa8jA+pNT8b9OZ1eT0ZpRsjHq31lWcjUoOvqSEfEkdd1TYr5XtBhbsL9emaX0Zp3WdQJkUvtHRhTDt8+BM6XiEGrLVHMlBWcHeMO5bL21U5DsPi0/KwfNmwcw5gqVbqhJMvp6EUYfeq6hdDjW55jbJD3K8rLg8QVnb6vPoGRuJTKX5Y7Ntx3U1gxTTXtH1bRUm9TIFeRuBaMuiMf3QyYFjV9uCi8VxGPOHsx3uSmHYgsvQeX5BnIBpG6qGgH1vQc4Yn6Xw1xczVJdNXVpx3Tqb9BXZXHGjHeCz1ZGbzCsQJvLsAkxMaHD8UWqBxPHvNdH2ZKcgdDOiYOsUsmhjYAgDvy8uSmtTGvD4UXe9jmF0H4ba2SbhOrmYgvjcnuJhP4vHcMVpSJZwcFbJtHWMRV9Rf16kMYCdTKdnPTr+MMHpCYYvLCtcTI0+KNSLXfBlHhjvT/DB0DWgjRzdm06eMwSW5blCJmKaN4h4kUPMmUUsqFjcONPiZgk9rtlEoFu1EgEUlmIF8pZ7Am/ycIeH4TA8YbWnWbD11RjdfVoJkdQW2XLemwHFAuuLUDfbaBiEKGYe2HtGtYY3hg6hhd2ifz8cH5xUVbCmbCMvzTYz9qnGmxjZgRwpn8hyIcgUDO7jO7gEzhADYGvUAdGSU9BSjgF6dmjN7rvUyB9R8Q6BTnHwkPe+rCMsB5cV15UaVnxhGrOB4A0VgCz7afRwdrEIS8aFp+MhCDt4/9ZZNdtZzeXcCuRU8bOk8OYEHjxP3bhFeoTuPMabc+ANghKTt0aZA9WmbkOGyUO1HZKBQPl6mF3bsTjHFhTDnQPHLb8fjIEhw+tQD1gPZjqBKnfsxQW2nDWzeI6O7qLsnoKXQS/I4XxNKEBSWRqpcv1MZrN3TxFiOZp6Kw9Y4QzfcyCFJXaW7KLvOH3J0CDIpHwHeScGjGpj6ea2rf37Cahlx/a4K1SZ7HH6T4Yy6d4ar9q4d9iNxoyP/a6m5CKdRj8Y/EzvPm9zo+6fKWqhYbg3Rb0+a2Zd+QAAAAASUVORK5CYII=

erYULF6pZs2YW5WPHjtWXX36p1atXa9u2bTpz5ky+xaEAlDySXMAdKi/BdfToUX3zzTdyd3e/bntHROe1atVKm
zdvNpfl5uZq8+bNCgoKKulwAQAAAJuWmZmpAQMGaPHixapWrZq5PD09XR988IHeeecdPfjgg2rVqpWWLFminTt
3ateuXVaMGCh/SHIBZVRmZqYSExOVmJgoSTpx4oQSExOVnJysK1euqHfv3tqzZ49WrFgho9Go1NRUpaamKicnx
3yOTpO6ac6cOeb9yMhILV68WMuWLdPBgwf13HPP6eLFi+bVFgEAAIDyasSIEerRo4c6d+5sUZ6QkKArV65YlPv
5+alOnTqFLuCUnZ2tjIwMiw3A7WPieaCM2rdvn8UqppGRkZKkIUOG6NVXX9W6deskSS1atLA4bsuWLQoJCZEkH
T9+XOfOnTPXPfbYY/r99981efJkpaamqkWLFtq4cWO+yegBAACA8mTVqlXau3evdu/enaa8eO+yegBAACA8mTVq1Xau3evdu/ena8uNTVVjo6O+VYx9/D
wUGpqaoHnY/EmoGSQ5ALKqODgYJlMpkLrr1eXJykpKV/ZyJEjNXLkyNsNsDQAAALhjnDp1SqNHj9amTZvk7Fw8q
O5HRUWZf6SWpIyMDNWuXbtYzg2UZwxXBAAAAACgEAkJCTp79/qxatmwpBwcHOTg4aNu2bXrvvffk4OAgDw8P5eT
k6Pz58xbHXW8BJxZvAkoGPbkAAAAAAChEpO6dtH//fouyoUOHys/PTxMnTlTlTt2rVVoUIFbd68WREREZKkw4cPK
zk5mQWcgFJGkgsAAAAAgEJUqVJFTZoOsSirVKmS3N3dzeVPPfWUIiMjVb16dbl6dbm6umrUqFEKCgpSYGCgNUIGyi2
SXAAAAAAAA3IZZs2bJzs5OERERys7OVrdu3TRv3jxrhwwWUOyS5AAAAAAAogq1bt1rsOzs7a+7cuZo7d651AgIgi
YnnAQAAAAAAcAcgyQUAAAAAAIAyjjyQXAAAAAAYANOnz6tgQMHyt3dXS4uLmratKn27Nlj7bBwC4xGo7Zu3apPPvl
EW7duldFotHZIAFAuMCcXAAAAAYGV//fWXOnTooI4d2rDhg2qUaOGDh8+rGbq0bVw+ePEJLX1ZRZhMDmPgeGjU
YuyI0eOqG7duoUe4+TkjJFdXV4sN1hMcHCwfHx9NnTpVubm5YWjJ6aNkU0vvvj4/gnB0rDrDRpZQQ3A98SPJBQAAAAFhZhw4ddPjwY
27Fjt2rVLU6dO1bFjx7Ry5UotWrRIIOaMsHZouEn29vaaO9XOm1q9fr7CwMIvVFcPCwrR+/XrNmDFD9vb21g4VA
O5YJLkAAAAAK2vTpo3Wrl2rTz75RE2aNNNHrr7+u2bNm2ba8CAAdYODUUQHh6uNWvWaP/+/Wrfrfvr1cXV3Vvn17HTh
wQGvWrFF4eLi1QwSAOxoTzwMAAAA2IDQ0VKGhodYOA7cpPDxco5RQeXV6Sm5VVmJp7UXrXQBWsUVE7Qes
ADFyN7eXiEhIdYOAwDKHZJcAGDjLl26JEnau3dvkY8KSfHx85OLiXUqRjDx48WOTrAQAAAIC1kOQCABCABt
36NAhSdLTTz9tletXqVLFKtcFAAAAgKIgyQUANq5Xr16Sij4f1a0qrTm/AAAAAOBWkeQCABt333336d//

QEAJa/Y5+QCUDr27dungIAABQQESJIiIyMVEBCgyZMnS5LWrVungIAA9ejRQ5L0+OOPKyAgQAsWLDCfIzk5WSk
pKeb9v/76S08//bT8/f318MMPKyMjQzt37tS9995bio8MAAAAAICioycXUEYFBwdfd16HJ554Qk888cR1z3HtU
OJZs2Zp1qxZxRAdAAAAAACli55cAAAAAAAAKPNIcgEAAAAAAKDMI8kFAEAZsX37dvXs2VPe3t4yGAyKjY21qDe
ZTJo8ebK8vLzk4uKizp076+jRo9YJFgAAAChlJLkAACgjLl68qObNm2vu3LkF1k+fPl3l3vvfeeFixYoO+//16VK
lVSt27ddPny5VKOFAAAACh9TDwPAEAZ0b17d3Xv3r3AOpPJNmzZ+vll19Wr169JEkfffSRPDw8FBsbq8cff7w
0QwUAAABKHT25AAC4A5w4cUKpqanq3LmzuczNzU3BixK0Y3LmzuczNzU3t2rVTfHx8gcdkZ2crIyPDYgMAAADKKpJcAADcAVJTUyVJH
h4eFuUeHh7mumtFR0OfLzc3NvNWuXbvE4wQAAABKCkkuAADKqaioKKWnp5u3U6dOWTskAAAA4JaR5AIA4A7g6ek
pSUpLS7MoTOtLM9ddy8nJSa6urhYbAAAAUFaR5AIA4A7g6+srTO9Pbd682VyWkZGh77//XkFBQVaMDAAAACgdr
K4IAEAZkZmZqWPHjpn3T5w4ocTERFWvXl116tTRmDFj9MYbb6hBgwby9fXVpEmT503trbCwMOsFDQAAAJQSenL
huhITE2UwGMxbYmKitUMCgHJrz549CggIUEBAgCQpMjjJSAQEBmjx5siTphRde0KhRo/TMM8+oTZs2yszM1MaNG
+Xs7GzNsAHcgrfeeksGg0F0jxoyxdigAAJQZ9ORCoQwGQ76yvC9WJpOptMMBgHIvJCTkuq+/BoNBr732ml577bV
SjApAcdu9e7cWLlyoZs2aWTsUAADKFHpyoUD/TnDZ2dlp7NixK7Ae/AAAAxSMzM1MDBgzQ4sWLVa1aNWuHA
wBAmUKSC/n8eOjiiRMnZDZQa9c4778hoNOrEiRMFtgMAAMDtGzFihHOnTtbOxQAAMochisin7whiXZ2dvL
x8bGo8/HxkZ2dnXJzcxUQEMCwRQAAbsLRo0d14cKFIh938HBi/8WVZUqVdSgQYNbOhalb9WqVdq7d6927959
+2zs7OVnZ1t3s/IyCipOFCIS5cu6dChQ9w9ChQoJYtWxZTRABQvpYKE1PnJxcSnO0HH5+fqpYsGJhRq9OjRBZY
PGzZM8+fPL+VoAAAom44ePaqGDRtaOxQAACoxd+/ePa9ZGVcdvfP00/nKFy5caC1QsoqsiHkdXo6fNvkpnoOrJ9hVZbV
GVGl1vd+hoB9Bhd8vrecVoLBCOnH9dqq2V3HsRXflUVqzOk1Lsx6vyBN3tMvFpf9pY1XEs3v/pcFHqoKxd+tpjtPDPZsbn1kK
Qv7f6uCk0GbBVkZn9Tw5JBwNrxyzyi9CJR7f40JOBqIQIRPfSnmdwSPb+1x0GH/FtQEjRbo/FFXMtB+3fRhn1Ng4l
GBd3jl4o3vfz8+vpMIDgHKHJBfy2bdvnwICApSbm2v+YJ0nKytLdgYYLx1Hrvo9s+rTfRhbKkeo2lo1Q+LTCdfVLy
1Ov79+y3Khg4dKj8/P02cODFfgkuSnJycyc5OTkVFohogAVK1a84X19q/c+AKDoSHIhnxYtWpj/9vX1lZ2dnYYNG
6aFCxeaE1zXtgMAAMCtq1Klipo2bZpbxo2hUAafa6n6Of2YPe5baslf2UcrtaGh57y9nWVHohjQmVZhKKkvSYnMC
7awcA22UymfINSdy3bx8JLgAAgFKwdetWzZ4929TZ1ZIvbGIE3UHasGGDuf7y5saM9KF3uJhwHgAAAAAA2BI
7awcA22UymfINSdy3bx8JLgAAgFKwdetWzZ4929phAOXe/Pnz1axZM7m6usrV1VBQUHasGGDuf7y5csaMWKE3
N3dVblyZUVERCgtLc2KEQPlFOkuXFeLFi1kMpnMGOMUAQAAAJQntWrVOltvvaWEhAht2bNHDz74oD74oHr16qWff/
ZkjR27Fh9+eWXWr16tbZt26vYZ84oPDzcyx1ED5RPDFQEAAAAAKETPnj0t9t98h803Nnz9fu3btUq1atfTBx9Bx9o5
cqVevDBBByVJS5Yskb+/v3bt2aqXAwEBrhAyUW/TkAgDgDgJhiNRq1at0qlatUvXFFFu3Nncxs/
PT3Xq1FF8fhzYhH58n0zlZGRobQDCAAAAAA69u/fr8qVK8vJyUnDhw/X2rVr1bhxY6WmpsrR0VFVq1a1aO/h4
aHU1NTrnjMqKkru7u7mzdvbuwRfAQAAAHDkUnRO/fVnJjZ1x8tL/5VvYmMgPEMLnffX4Q1ao8zPnyQDAMOwQC
3Cm0W6e2TQGPvjV7OXcpCAIYCbjPxYV6xDE7N8W2TrvgN9DGLu0s5SRYUTpFmKcd+1IbAWmq3gHdAQU29r0jrqBl+o
Mpam5A4FbMv4m7J2lVjcj4f5YEJ8Om7+qzZ0cd/IbgoYSGm8BjOcOhtnbTGIAB/VyZIS9RXWX1zpUwM4c1Trv04
wo91EsU7Qr30XPUyK54sYdZtHhUpITvKHfnbNM4SWFSCNiaGwrqM72VyKkoDsNoTPkjr1GodqNvbfFnBYBXHj56VJ
dBs4AARH9wLnmwRAQJm7gHPjaRC+YWh3/dqJ3hEbyNKp6Ig6Mqn6brEoJI7u4TdERn5CdgSnf92hAJBHZEq1
bnLMk33hu8aHMdbEv5DtGohFX1yEAutnJMyvDAr4QLVmPDXS8VIDwo7SfkA1BPnHgzdmSn0NQ8GndgK9nsvI4
+xXt2nB/xCnoxXk39fnxlbHlgb1LyZqaaJDKHpYR4zfWwl/L8ihmJYB56Nw8dYcVVzXnHl69GYLsYU4IdF+s
aC5CNwRGGgbBV3AKKvoP7vWBO7/7lXrBrG6m72UgAAAAAAAARD7wAAAAAAASHcBAoAuF7Bq0j4UoCcKI/AAAAAAA
UQXWYmZaXCmIrAWB5orOYCxTjAMDOYEOMgAgAgAMAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

quv84/oGl8qCwKVGeGhgYo4WGqYphFJCONDYMDW2JJNPGOsYOxkwFO4amZnDSGq1lJw/VTidiE+fXmk6KSWNG4
9COmYZE3FiME4htScBHCBpZNA8sqLiGZX9/5Mf+SgBldZfdu7xfMzty7z3nnu+O892rX86eAwAwvDB/BwAAABD
MQvova05CqCK7j0lnx+/3i5HdxzQnIVQh/ZfHbUwAAAB/osgFAADgQxEXOtTO8BTpwMPSgfEbNONSO8NT1HKhQ
1Lu+AOMAADgJxS5AMDgLl26pNbW1mHnnU6n9uzZIOl69dVX5XQ6ZTKZRrxHenq6oqKifBkmMGFdnpKirOcu6C9
/+YsyOtPHbdyW1lb9+Mc/1v98N2XcxgQAAPAnilwAYHCtra2aO3fuVdtUVVWpqpq1OuNjY3KysrydmgAJLnCI
vSebUB9N82UkmaP27h9tgG9ZxuQKyxi3MYEAADwJ4pcAGBw6enpamxsdB+/+eabLt2rfLz87VkyRKFhYWpv79
fNTU1slqteuqpp/Ttb3972D0AAAAAwMgocgGAwUVFRblYTmdTpWWlqq4uFivvvqqDh48qM7OTiUmJQm5Qa
Wmpqqur9cgjj4z61UUAAAAAMKLx2+IHAOBzVqtV7e3tys3N1cyZM1VQUKAlS5aooKBAM2fOVE5Ojtra2mS1Wv0
dKgAAAAB4FTO5ACCIdHZ2SpIqKytVWFioNNBQffbZZ/rqV7+qGTNmaN26dUPaAQAAAECwYCYXAASRuLg4SVJYW
Jhef/11nTp1Sp999pl0nTql119/XWFhYUPaAQACg8ViUXZ2tqKjoxUXF6eSkhIdPXrU32EBAGAoFLkAIAj19/d
LkubPn6+6ujrNnz9/yHkAQGAoFLkAIAj19/d
LkubPn6+6ujrNnz9/yHkAQGCpr69XWVmZDh06pH/84x/6/PPPPPdeedd+rixYv+Dg0AAMOgyAUAQeTUqVPun4uKi
rR582Zl Z2dr8+bNKioqGrEdAMD/9u3bp+XLl+u2227TrFmztGPHDnV0dAzZPRcAAFwdRS4ACCJVVVVWSpOTkZLW
0tCg3N1cxMTHKzc1Va2urvva1rw1pBwAITHa7XZIUGxs7apv+/n45nc4hCAwRjMAxQ2XrayJU+wTYIqk6TPWqEk
WPGdfUF4Jmenh5JUlJSkqxWqw4ePKjOzk4lJiYqLy9PCxYs0ECffeRuBwAIPAMDA1qzZo3y8vKUk5k5ajuHwyG
Hw+E+5rPd+y73fqo5CaH68NBeRXYf86ivw+HQ2bNnlZSUpPDwcI/62UUPDwcI/62traNCchVCH9lz3qBwATHUWuCeL48eOaO
XPmDd1j6dKl19332LFjFLqAcZCUlKRPP/1U77zzzjkLRUhw8fdrcDAASmsrIyNTc36+2
3375qO4vFog0bNoxTVBNT1/tvq+nhKdK5LdI5z/vPlqTTnvfLkPTdh6eow/WJ55OBYAKjyDVBDM7gevHFF5WRk
eFR376+PrW3tys1NVWRkZEe9W1padHSx3JcnuxiZ8F9ZQzAUzBVbTtgpAf6nXsDnxCFeU/BrPJIeDm58Bn82HdXrX4sb
vAHAHjvkzUbYdhgPngtEvhgUoHBobHK9i6vW5kaWl3tNxkg5SeH0fXGpWE97+PqGnz7jWPKbnMkSlLPvZCm
9j2nKUGeDv4DxVOkPrunRVKSxdZJ4XubgMuhkrRj5nd+nPZxxgqZHneR8zHdDnZmaKRYZ81SfNr+cMoTM9jc8wRy
WrdkAVp6/t9cb8/x5bFjk+9GNZ+Zd+iRcMX1UNkZAAIoOZOyLTm5+Rn1g7Xg5Bkz5CpUD7+kDBBbVeTHjo9qS/e+Q
/u9Myp0/JPNGjbVzqHbnxJ5XvWm5k9SlNpCU0/2PxkmKYDvfxpNMVrbkrFNW9pTkMN05JbJ+R+I/LQdw1rncPknu
aYQBsgPgTwLfjWI8/3D0n4f87nt0PYWZ+HywVcn3HKjTTmPRRzD/y+e0ZWFLvc0Jb/Xv0fq6OFvLLmEwTZN2sZc
uRiv9K4fD11X10JOPg0PZIwsmxqnI4i/6tDBs2mXY8vX/RZB8+/Ax3mWRBnTJe5kMHFbD39rhzZ4kTBTT+0
aoK+/1DRGtMVLbKvHfZ/0aPF0bxhf/rKZLF4S1KIyF5OxKo3LLA+JJ0x+gPPnb1PrM/Ou8xUVF7wSRwzhl5ecYy
8Ul87MDc0cdrkPXlsDoDXYFGdtx7OCeRZbk3Mt7w9PdRkD/hPSjYvaj+LO/DFtLB3VXS7VDA3PmZ5Sw4aIXUxpmKWzb

CYyrxe5Pv74YzmdzhF3dWttbR3W3uFwyOFwuI97enq8HRIkXe79VHMSQvXhob2K7D425JrD4dDZs2dv6P5JSUk
KDw8fdt7W1qY5CaEK6b98Q/fHcPn5+XK5XKNeDwkJOcaNG7Vx48ZR27S3tw85fuml17wVHvzEarWqvb1dO3fuV
HZ2tqxWqzo7O5WYmKj8/HwdPnxYubm5slqtWrhwob/DhZfxDDaeS5cujfh3M6ilpWXInyNJTO9XVFSU12MDcG1
Xy+Gx5K9EDgP+wjM4OHm9yOUpi8WiDRs2+DuMoNf1/ttqeniKdG6LNMKyLLNvdIBRNtTLkPTdh6eow/XJjY4AY
AwGdO3MzMyUyWQaVsjKzMwcOg4TG89g/2ttbdXcuXOv2W7pOqWjXmtsbFRWVpY3wwIwRmPJ4avlrOQOA/7CMzg
4eb3INXXqVJlMpjHv6lZZWen+mpXOxW+Rk5OTvR3WhJd/94PavVvu3bb+my9ncknS5MmTlTLnjhu6P4Cx+e/dF
efPnz/sOrsrBjeewcaTnp6uxsbGUa/39fWpvb1dqampioyMHPUeAPzjajk8lvwdvAeA8cczODh5vcg1adIkzZO
7V3V1de4FrgcGBlRXV6fy8vJh7cPDw0ctjsB7piYm6+5VT4x6ffa4RQLAlwZ3V6yqqtKePXsUGvr/+4sMDAzIY
rGwu2IQ4xlsPFFRUdf8DXBeXt44RQPAU9fKYfIXCFw8g4OTT3ZXrKioOPbt2/XCCy+opaVFP/3pT3Xx4kU98MA
DvhgOAPB/TCaTfvvb36q2tlYlJSVqaGhQb2+vGhoaVFJSotraWj3zzDMsOh/EeAYDAOBdFotF2dnZio6OVlxcn
EpKSnTO6NEhbS5fvqyysjLdfPPNmjJlikpLS4fNrAbgez5Zk+uHP/yhzp8/r1/96ley2WyaPXu29u3bN2whXAC
A9y1evFi7du3Sz3/+c+Xm5rrPp6WladeuXVq8eLEfo4Ov8QwGAMC76uvrVVZWpuzsbPX392vdunW688479cEHH
2jy5MmSpEceUR/+9vf9Morr8hsNqu8vFyLFy/WwYMH/Rw9MLGEuK62PZsf9PTOyGw2y263KyYmxt/hAAEnOHM
kOOObSJxO57DdFZnB5V+Bnh+BHh/gb4GeI4EeH+BP3syP8+fPKy4uTvX19frWt74lu92uadOmqaamRvfcc4+kL
xY1z8jIUENDw4jrpPoyPiAYjTVH/L67IgDANObaXREAAAA3xm63S5JiY2MlfbHD3ueff65Fixa526SnpyslJWX
UIpfD4ZDD4XAf9/TO+DhqYGLwyZpcAAAAAAAEm4GBAa1ZsOZ6eXnKzMyUJNlsNk2aNEk33XTTkLbx8fGy2Wwj3
sdischsNrtf7G4MeAdFLgAAAAAAxqCsrEzNzc166aWXbug+lZWVstvt7tfpO6e9FCEwsfF1RQAAAAAArqG8vFy
1tbU6cOCAbrnlFvf5hIQEXblyRd3d3UNmc3V1dSkhIWHEe4WHhys8PNzXIQMTDjO5AAAAAAAYhcvlUnl5uXbv3
q0333xTaWlpQ67PnTtXX/nKV1RXV+c+d/ToUXVOdCgnJ2e8wwUmNGZyAQAAAAAwirKyMtXU1Oi1115TdHSOe5O
ts9msyMhImc1mPfjgg6qoqFBsbKxiYmKOevVq5eTkjGlnRQDeQ5ELAAAAAAIBRbNu2TZKG7Vr9pz/9ScuXL5ckb
dmyRaGhoSotLZXD4VBhYaGqq6vHOVIAFLkAAAAAABiFy+W6ZpuIiAht3bpVW7duHYeIAIyGNbkAAAAAABgeBS
5AAAAAAAAYHgB93XFwamgPTO9fo4ECEyDuTGWadP+QA4DoyN/AWMjhwHjIn8BYxtrDgdckau3t1eSlJyc7OdIg
MDW29srs9ns7zCGIYeBayN/AWMjhwHjIn8BY7tWDoe4AqyUPTAwoLNnzyo6OlohISH+Dgf6omKanJys06dPKyY
mxt/hTHgulOu9vb1KSkpSaGjgfeOYHA4s5G9gIX/hKXI4sJDD8AT5G1jIX3iKHA4sY83hgCtyIfDO9PTIbDbLb
reT3IDBkL+AsZHDgHGRv4CxkcPGFHglbAAAAAAAAMBDFLkAAAAAAABgeBS5cE3h4eFav369wsPD/ROKAA+Rv4C
xkcOAcZG/gLGRw8bEmlwAAAAAAAAwPGZyAQAAAAAAwPAocgEAAAAAAMDwKHIBAAAAAADA8ChyAQAAAAAAwPAoc
mFUBw4cOPe+9zOlJSUpJCREe/bs8XdIAMaI/AWMjRwGjIv8BYyNHDY2ilwY1cWLFzVr1ixt3brV36EA8BD5Cxg
bOQwYF/kLGBs5bGxh/g4AgauoqEhFRUX+DgPAdSB/AWMjhwHjIn8BYyOHjY2ZXAAAAAAAADA8ilwAAAAAAAAwP
IpcAAAAAAAAMDyKXAAAAAAAADA8ilwAAAAAAAAwPHZXxKguXLigEydOuI/b2tpO5MgRxcbGKiUlxY+RAbgW8hc
wNnIYMC7yFzA2ctjYQlwul8vfQSAw7d+/XwUBcPO33///dqxY8f4BwRgzMhfwNjIYcC4yF/A2MhhY6PIBQAAA
AAAAMNjTS4AAAAAAAAYHkUuAAAAAAAAGB5FLgAAAAAAABgeRS4AAAAAAAAYHkUuAAAAAAAAGB5FLgAAAAAAABg
eRS4AAAAAAAAYHkUuAAAAAAAAGB5FLgAAAAAAABgeRS4AAAAAAAAYHkUuAAAAAAAAGB5FLgAAAAAAABje/wIfo
5R4E69eyQAAAABJRU5ErkJggg==\n"

```
          },
          "metadata": {}
        }
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "# Calculating upper and lower fence using IQR\n",
        "for i in numerical_columns:\n",
        "  Q3 = merged_data[i].quantile(0.75)\n",
        "  Q1 = merged_data[i].quantile(0.25)\n",
        "  IQR = Q3-Q1\n",
        "  upper_fence = Q3 + 1.5 * IQR\n",
        "  lower_fence = Q1 - 1.5 * IQR\n",
        "  merged_data.loc[merged_data[i]<lower_fence,i] = lower_fence\n",
        "  merged_data.loc[merged_data[i]>upper_fence,i] = upper_fence"
      ],
      "metadata": {
        "id": "pWf6gVkxbZ4r"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "# capping outliers with IQR\n",
        "numerical_columns =
[\"CHILDREN\",\"Annual_income\",\"Employed_years\",\"Family_Members\",\"Age\"]\n",
        "fig,ax = plt.subplots(1,5,figsize=(15,5))\n",
        "for i in range(0,len(numerical_columns)):\n",
        "  ax[i].boxplot(merged_data[numerical_columns[i]])\n",
        "  ax[i].set_title(numerical_columns[i])\n",
        "plt.show()"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 468
        },
        "id": "LcEb82nBdZ1L",
        "outputId": "0fa0db9a-381e-4133-dad7-217f94b5c95a"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 1500x500 with 5 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAABL4AAAHDCAYAAAAqZtO0AAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAAB37Ul
EQVR4nO3deVyU5f7/8ffgMqACroAobrmAGwKZgpqWC5pZmFuWoWa2YV/H43Ntk0O/cqsTdszUk7hVSqUcy46alWlmo
UfFjhsdLSR3LAXTAkRlNJjfHz2YGmGQUbYZXs/H43507uu+ruv+3J3Pd1G8xms1kAAAAAAACAk3Gp6AA
AAAAAAAACAskDiCwAAAAAAAE6JxBcAAAAAAAAACcEokvAAAAAAAA0CUSXwAAAAAAAHBKJL4AAAAAAAADglEh8AQAAA
AAAwCmR+AIAAAAAAAIBTIvEFAAAAAAAAAp0TiCwCcRHx8vAwGg06cOFGmbQQBIiYmJMhgMSkxMrOhQbJo+fboMBkN
FhwFUGgaDQd0nT7fsV/X3wD59+qAYAlDkSX5XY0aNH9dhjj6lVq1ZydXWVh4eHevTooofnz5+vy5cuSp
BYtWujuu+8usn3H43PPPztmMpaCvgs1oNMrb21t9+vTRa6+9pl9++++aVj+ePj/ZfRsjZFWz
Vq1dXkyZNNG7c0P3888+F6vfp08+F6vfp08eq/l83f3//Qv26Sc7cdr4cKFMhgM6tatWOWHAqAI176nXLvt2rWro
kMEYENx4/dvf/tbRYdJGo+X+OVt0f6Lh4BSHwHqCqqV3QAKNrnn3+uESNGgGg
0KjIyUh07dlRWKlZTWK1eNsBTfyGSd+9sL0dFSYOb
dtSVK1e0ffft2Pffcc/r+1d0nSco/r//7v/9Sla1fl5eXpl19+eP+jj49+uijj
XcWajNz5ky1bNlSubm52rVrl+Lj47VjwwaB3unWbnAaujjqWnLnV9tLJGgQn8OnWjJcgHrs0jdkBeeRaOq6O8GYJHwkjgZVYOLZCLQFSjXJG0NrtAVFk
```

rV65UixYt9N///ldHjhxR69atKzqkcvHQQw/p/vvvl9ForOhQgBIpeE+5VlUZs4AjK2r8luePlZcvX1b16uX79
cdoNOpf//qXnn/+eavyNWvWlGscAIpWVb8DVDUkviqh48eP6/7771fz5s319ddfq3HjxpZjUVFROnLkiD7//PM
Kia1Xr14aPny4Vdl3332nAQMGaNiwYfrhhx+s4pWkQYMG6dZbb5UkPfLII2rYsKFef/11rV+/XiNHjrSq6+npq
TFjxpQoli5duujtt99WTEyMfH19b+KqUNUdP35c03fu1Jo1a/TYY49p5cqVmjZtWkWHVS6qVaumatWqVXQYQIn
99TOFzu3ixYuqXbt2RYeBUlTR4/faH1zLw1133VVk4ishIUGDBw/Wv//973KPqTzl5+frypUrFfJvD1xPVf4OU
NVwq2Ml9I9//EM5OTl69913CyWRpD3T5mZZmVqwYMF16/fq1UvH7dy3owXXX3x3xReXl
5mjVr1k31A6xcuVL16tXT4MGDNXz4cK1cudLq+IkTJ2QwGPTGG29o6dKluuWWW2QOGtW1a1ft3r3bqu64ceNUp
O4d/fzzz4qIiFCdOnXUqFEjPfvss8rLy7PUs7U+UMG54uPjLWX/+9//NG7cozM29gJ0uPjLWX/+9//NG7cOMttzz4+Pnr44Yd1/vz5m772otY
3KbiFevv27brtttvk6uqqVq1a6f333y/UPjMzU08//bRatGgho9Gopk2bKjIyOurW6bNnz2rChAny9vaWq6urA
gMD9d577xV53W+88Ybi4uLUqlUr1apVSwMGDNCpU6dkNpv1yiuvqGnTpnJzc909996rX3/9tVA8X3zxhXr16qX
atWvL3d1dgwcP1vfff3/T/O5wDKXx0ip4/X/55ZfqOqWLXF1d1b59+xLPzli9erVCQkLk5uamhg0basyYMVa35
S9fvlwGgOH79+8v1Pa1115TtWrVrOp/++23GjhwoDw9PVWrVi09rP+Ksx98QzW0SvJt6vMm9+fV5TN6pLt7Xj
lJf1nK9PYTp48qSeffFLt2rWTm5ubGjRooBEjRhRaU6Gngb9HWrVv15JNPysvLSO2bNpUkXbhwQV0mTLH8nfHy8
lL//v21b98+u68RlY+9r5//Ht27fr//7v/9SoUSPVrVtXjz32mK5cuaLMzExFRkaqXr16qlevnp5//nlq
d4+taY8eOVcOGDXX16tVCxwYMGKkYKB27drZfXOPPPPCAkp0TdejQIUtZenq6vv76az3wwAN+tjGZTjo2bZpat24to
9EoPz8/Pf/88zKZTIWuZ9ksrmW2d+ehYWFyc3HPFyc3
NTS1bttTixYtvOqaVK1eqQ4cOMhqN2rhxoyRp1apVCgkJkbu7w8PNS+pUyfNnz+/xP+WQGm73ncASTp//rwee
ugheXh4qG7duho7dqy+++67Qp/XJenQoUMaPny46tevL1dXV916661av3599OVONikPiqxL69NNP1apVK4WFhZW
o/tWrV3Xu3L1CW1ZWVhlH+qfhw4fLzc1NX3755XXrFrzp1qzXr9CxvLy8LLy8Iq/l4sWLheq2bN1SkZGRevvt3X69
OmbvgZUXStXrtR9992nmjVravoOTp8+HChhJb0x6+zs2fP1mOPPaZXX31VJ06c0H333Vfo
DvfHGG+rdu7fmzJlzw7cnb968WceOHdP48ePl1113Ffpu8fp7779eqVat01113FfpQX1qOHDmi4cOHq3///
auncePGWSWRcnJy1KtXL7311lsaMGCA5s+fr8cff1yHDh3STz/9TTz/9J0mP2Or690mjDz74QA8++KBmz54tT09PjRs
3rsgPuytXrtTChQv11FNP6Z1nntHP6ZlnntHWrVs1cuRIvfTSS9eeOEFPfroo/r000/17LPPWrX94IMPNHjwYNWpU
0evv/66Xn75Zf3www/q2bNnrlV242BllZWUVeo+4Ngl8M68jSTp8+LBGjRqlQYMGKTY2VtWrV9eIESlESO0efPmYm0
Lj4/XyJEjVa1aNcXGxmrixIlas2aNevnvbsqczMTEl/vl8W9eF65cqV6t0nj5o0aSJJ+vrrr3X77bcrOztbE6ZN0
2uvvvabMzEzdeeed+u9//2tpd+2+2LGLbvXu3du7cqfvvvvv1///0c
/9fjjj2vLli3q06ePLl26V0g8Tz75pH744QdNnTrVsvbT448/rkWLFmnYsGFauHChnn32WbmmuSkljJcWu60PFK
2r82vsaeeqpp3T48GHNmDFD99xzj5YuXaqXX35ZQ4YMUV5enl577TX17TX17NlTs2fP1gcffGfGBXfA899JDOnz+vTZs
2WZUXJKpKelfCX91+++1q2rSpEhISLGUffvvih6tSpo8GDBxeqn5+frrvuu0/z58/X0Nc9GeMWSgjKSFGfoHHDVsBNcGsWlqhpIFgq+Gp9ZmZK6WLzxqNxZ4CdtWWSbotRtmbFgCqi/zwwwzwnmYXXeMdZo6RWJRVv2t+Tu/
o0aVaj+f/7zHz3zzDMa03aspk+frpSUFN19992Ki4vTP//5Tz355JTUpIcffrhQ+99++++0133XWXQkJ
C9I9//ENNmzbVE088oWXLltlwTF9//bWefvppjRo1pjRo1Svpnz1eLFi2OefNmjR49WvXq1dPrr7+uWbNmqU+fPkUm8
oHycr3vAPn5+RoyZIj+9a9/aezYsfr73/+uM2f0aOzYsYsYX6+v7779W9e3eIpKTob3/7m+bMmaPatWsrIiLC7vd
jlAEzKpWsrCyzJP09995bovI5Z2zWf3l16ivPX+xPoFrQtZkFbfgy6i/IqsUe6hNSpIfOtT8cJa/kmGYq9h9EnYz1tsszzjfPfrIHZJP
prrlatX6FxfffWV+ZdffjGf0nXK/PHHH5s/lNhqN5l0nT1lm17927t81reOyxx4q8hqNHj5qrV69u/r//z/+u/+z+
rfjpO6FCifz9gz549ZknmzZs3m81mszk/P9/ctGlT8+9P+ct1l8Tl8+TJky11jh8/bpZkfbSGhgfnXX3+l1H/yySdmSeZPP/+3UU
jZ27FizJPPMmTOtzhMUFGQOCQmx7BeMemqW+++caqXsG5li9fbim7dolSobj/9a9/mSWZt23bZikrGBvfHjx8v8fU
X1abg78pf+z579qzZaDSan3nmSKdRvfn6+2Ww2m+fNm2eZxoX12JUrV8V8yhoaHmOnOnXqm
LOzs62uu1GjRubzMZLMkc2BgoPnq1auW8tGR5tr1qxpzs3NzvNxNzvOFCxfMdevWNU+cONEqtvT0dLOHh4dVeVxcnNliscN3
np2ehcjiegtdqUZvRaDSbzTf/OjKb/3z9//vf/7aUZWVnmFO7hq9cuWL28vIyJ+fu5Iyd+zY0XO6v5m6VLv
c8++8wsyTx161Sr8/r6+przAzZvf127bMa+/n5+eY2bdqYw8PDzfn5+ZbyvLw8s4+Pj7l///6WvitXrpS5fpPDLePJbP7j70HLLi3N/fv3t5RFRESYXV1dzSd
PnrSU/fDDD+Zq1ayZq7f2oZ9qxFfX3KykpySzJ/P7771vKmzBvKCv7/dmzp/n333WmOioqy6zQuRQ3fu19j
Vz7ugsNDTUbDAbz448/3P448/bin7/fffzU2bNjX37t3b3zbvNKZNGCBJCD9qatA8tdr4VtK+vA9MC8vz9y9ZboPBYD5
27FiJr/mvn6+fffZZZc+vWrS3Hunbtah4/frL+T//+Y9VO+uK+T//+Y9VOffXLzZLMu/YscPqeoxGjG7oPJhyZHBv42bz8GZ
JIlS8ySzD4+Ppb3WLP5z7+Df6lb8Nl7zpw5Nc+uzzjlTVzplupi6fjf2mm20ce59kFn5aRHKbkXZF4t47lGvNnDdb10fhRkL5xcNZZSvDRNTaXfSJVxtmTt5WqfoUbtvXyXHUoQ+V2N
4BipKSb4D/Pvf/zZLMs+bN89SlpeXZ77zzzjsLfV7v27evuuV0nTlafKfLz881hYWHmNm3alPnPMzp8hsUvqqbnsUwabFHB8cLb2/FlEPaBnM8Ww2ozJ
Unu7u41bt0tWzdt3ry50PbGG2+UVZhFq10nji5cuFCovF+mrUqJH8/Pw0Pf1a5dW+vXr7fcwvX5wf5b8IXXbt
NmTKlyHO2atVKDz3w30kJYUXaozZ86U9iWhCli5cqW8vb11l1xx13SNJNTJLZW8mvbIlqb0AatUqq1sTJWnUqFFWMxULbts9
uxYoX4ff/xxq/1evXXoVVa8k3NzcLP87NzdX586mtdU+3j+rX+Tuz0wSWpzG71ad++WnZ+veX6JK1Ro0Zq166d9+//VmBgoIY
0HVqovcFgkCRt2LBBPC6mEWBU/YGTNmjNWCxN26ddd0VK1cst

11t3rxZmZmZGj16tNVMgmrVqqlbt2765ptvbuafA5VIXFxcofeIL774wqr0jb60Cvj6+lq9rj08PBQZGan9+/c
rPT29yLj27Nmjs2fP6sknn7Ray2bw4MHy9/e3WpszMjJSp0+ftnpdrly5Um5ubho2bJgkKTk5WYcPH9YDDzyg8
+fPW82A7tu3r7Zt26b8/Hzl5eVp06ZNioiIULNmzSz9BQQEKDw8/Pr/oNcozdgk679fV69e1fnz59W6dWvVrVu
3yL9fEydOLLTuYN26dfXtt98yu9sJFDV+7X2NTJgwwfIeI/0xjs1msyZMmGApq1atmm699VVa733ddXFz04IMPa
v369VafaVeuXKmwsLAiH6xREg888ICOHDmi3bt3W/5r6zbW/5r6zbH1atXKyAgQP7+/lbvZwUPkLr2/axv375q0aKFZb/
g792wYc0svk8UlF/7b1K9enU99thjlv2aNWvqscce09zZ7V3794biq13795q3769VVndunV18eLF686cBcpLS
b4DbNy4UTVq1NDEiRMt7VxcXBQVFWXV16+//qqv/5aI0e01IULF6xmpIeHh+vw4cOFPmugfLG4fSXj4eEhSUU
mkGxp2LCh+vXrV6i8vJ9ak50TU2TCLi4uTm3btlVWVpaWLVumbdu22XyCXO3atYu8luK89NJL+uCDDzRr1izWC
YBd8vLytGrVKt1xxx06fvy4pbxbt26aM2mzZogEDBljK//qlUvrzdt3ffvvNqtzV1VWNGjUqVPfaeiX166+
/asaMGVq1apX0nj1rdaysbmm+9lqlwtdw9OhRyxdW06ePKk2bdrIxcX6d8L0W5C88PPzK7K8Ij7Dh
w9LUpFPlpX+/NsKx3fbbbddd3HsG30dFWjdurAtJatu2raQ/tmAZ+jYxHdbfcXX8Zf5x1aAZAj9JbUTxhWLtYz2ryQ/Q1hcYl/3Z6AkYC/GQDzVaGf9UP9/ag9wHn1TRJRdM6C13kRZfqsnnmzq
eDdPh+nGwBBjN8/RC8EiUPpkL3BGkKNcn8Pa8tgp7w4XCZU3DWUUDdyjJ3mcb/v+dp5Vfkjoao20bKYSN4tG
mSWtGZpPqBqr73wGFWd2dP8jlDkUKR1l9tIMoYM3+bRGQVDtbP5Xt1OOowvo+E96n/iWJHn7QOYwp2ugcldirXA
rhYz/RGUZRkaEiEZSkSRB96cCuOVTmmM4J9+0QmXMXtOB0Kx6ck/vkM6bHlcBpcuBtwYvzXr0+TYfNd2O5Qm07I
HBpZ3RwncjpHJ6q1W/3RBFIbcQf5Z+fxU1KvoQKGl2UqHAY1Qy7ZE2AnR3/D01K/JfpxgjdlT2HoHlX5Jzl0GH5N
bf8pHQ6hfO5a6zmE/Krm6wEfQ7ehzlp6tKYXGmoBc8MmoqpkZW0W2PSfoswUBI2OQLSO/zfTnBTN1yG08JPBW9Gdt
7D8Ri8RUyjAHOAKb4X+XSR6Ihy7bH+Dt1NzpWGuNsc9iRXKzZKmzSpURAtPNFfnZdTUtNz9UBvgYRDQKjmzK2n
+hUvx9xF9rtR8g+eAkMBrNUf69tWVHYBOHEtTJHj3/E+7k6Gt9VYtgsQmEJYMhK5pRsjz+ZzK8NAqavJYd9YKBkQ
bfcct0kffPmzfW///1P+fn5Vh/yC55u1bx581KLRfrjg7W9s0aBaxX8cvvXBPWPP/4oSVa3FP1VwWs5NTW1OEy
bfcct0kffPmzfW///1P+fn5Vh/yC55u1bx581KLRfrjg7W9s0aBaxX8cvvXBPWPP/4oSVa3FP1VwWs5NTW1OEy

l1apVsOp8/fOf/5TOxwf8unXrKiAgQDNmzNDEiRNtLnhZlPvuu8/ya/Ffn+DO0O8/6aGHHiqyzfUSX61bt9aYM
WNYJwAlsnLlSrm6uqp///5FHndxcdHgwYO1cuXKQr9Slpa33npLV69e1eLFi2UOGjVy5EjNnj27OFp5CQkJeuq
ppxQXFyez2awBAwboiy++kK+vb5nEVJ16tTRf/7zH02bNk1r167Ve++++9Jy8vL/Xt29fy1FY3NzclrHibqb3/7m
9577z1lZ2erXbt2Wr58eakn8B944AH5+vpq1qxZmj17tkwmk5oOaaJevXpp/PjxpXouVJy/3vL7V8uXL1efPn1
K5Rxt2rTRW2+9peeee06pqalq2bKlPvzww+vezj9dSbmWerUxPqqQ2mbMhrNRWm6cO33w+vv6Atw4whn8ea/6Vn2
EG98MILuuWWW4qcAdKnTx8lJSXplVdeOYIFC5STkyMfHx9169bN6ilsnTt31qZNmxQdHa2pU6eqadOmmjFjhs6
cOXPDia/Sim3+/PmqVq2aVq5cqdzcXPXoUNfffVViZ84WatWLT355JP68ssvtWbNGuXn56t169ZauHChnjii
Ru6NlQuN/saKSuRkZH67LPPNHLkSJsPZSoLLi4uWrdunebOnav3339fa9euVa1atdSqVSvVStNnjzZsvB8aalXr57
ee+89PfXUU3r77bfl7e2tBQsWFHqK3c3GVLAe78KFC5WZmSkfHx+NGjVKO06dPL7UfLYCSsOc7QGZmpj7//HNNn
jxZ7733nlxcXDRO6FBNmzZNPXrOsHqKc/v27bVnzx7x7NmDFD8fHxOn/+vLy8vBQUFGTzswvKj8Fc3iu6AgAAVFI
tWrRQx44d9dlnn5X5uc6dO6fGjRtr6tSpNp8CVVEqc2xAefjkk08UERGhbdu2qVevXhUdDoBKZN26dRo6dKi2b
9+uHj16VHQ4KAHS6wAAABUgPj5eeXl5Nmc6V6TKHBtQHt5++221atVv2rOhQAFSgy5cvW+3n5eXprbfekoe
Hh4KDgysoKtiLWxOBwAnl5OQUuejOXzVq1Mjm484BlJ2vv/5aP/zwg/7+978rIiLC5pMiS1NWVlahD+/X8vHxq
ZDYgMpk1apV+t///qfPP/QUw5KOJQDO4amnntLly5cVGhoqk8mkNWvWaOfOnXrttdfk5uZW0eGhhLj
VEQCcOPTpOzVjxoxi6xw6xn1+fr5q3bq0q8NuOIsR1NpTTu6bqPLN3GDZzfpDwvJOnIeBbgxU9UTxu69BthOe
2xeiJhQAgfHixXZPNJVKPpYAOIeEhATNmTNHR44cUW5urgK0m8BHmTbNwSKXZaWAPR5VWk4bPpPTDgrq4ujc
VW6dnz55Wi3ICcF4//PCDTp8+XWydfv36lVM0gONiLAGA4yHxBQ6sGb8OkfgC6ffq0/Pz8KjoMoFI7deq
UmjZtWtFhFML4BUqGMQw4LsYv4u7u6Q/LsjDw6OCowEql+zsbPn5n+VnGSWXD+AWKxxgGH
BfjF3BsjGHAcdkzfh0i8VUwrdPDw4MBD9hQQWac/M36BkmEMA46L8Qs4LhKMn4r343MAAAAAAAQCkg8QU
AAAAAAACnROILAAAAAAAATonEFwAAAAAAdmjRooUMBkOhLSoqSpKUm5urqKgoNWjQHXq1NGwYcOUkZFRwVEDV
ROJLwAAAAAA7LB7926dOXPGsm3evFmSfSNGLECEnSO08/rU8//VSrV6/W1q1bdfr0Oad13330VGTJQZTnEUx0BAAA
AAKgsGjVqZLU/a9Ys3XLLLerdu7eysrL07rvvavbs2afny5u7rrjvqcKsGpdWYXAAAAAAAw
A26cuWKVqxYoYcfflgGg0F79+7V1a9fPOsdf39/NWvWTElJSTb7MZlM8Ylf6Mys70ttoA3DwSXwAAAAAA3KB169Y
pMzNT48aNkySlp6fLZclaqlu3rlU9b29vpaen2+wnKiGt1f+Ql6tsf/Fn7NzQ9feG+Mojk7fkeVzgaWWkmcWgGz
6bYrMNnaMAxLF++XF5eXhoU8eLCLLCQkRDVq1NCWLVsampqUpLS1NoaGhFhAlUaXbN+Nq6dauioqLUtWtX/f7
773rxxxxRc1YMAA/fDDD6pdu7bNdh4eHLYJMoPBcOMRAwAAAA5q9+7dWrJkiTp37uz3Obns6b4ndkWV1uv/12m+OMBkO
xUzoBAAAAZ5eTk6MHH3xQb7/M6TTsuXlDjNmzeXn5+f7r33n3//fc3c1oAAADA4URFRWnw4MG169cfiezGRqo3AY
MGCCz2ay2bdsW0ubq6q4uDj9+uuvunja9dOy5YtU+fOn
ZWVlaU33nhDYWFh+v7779W0adMi25dMJplv+dnb2jYaJG3Dp0iUd0nTI5vHLly/rxIkTatGihdzc3Iqs4+/
vr1q1apVViACKUudwYLsn4lRjDFaFPnz46fPiw3NzcFFft9t2rWzHM/NzdUzzzjVatWWf1y7O7tbpJjWWf1y7O3tbbNP
9msadOm6e2331ZmZqZ69OmZZqU2ihRYsWbvAc7r1WrVmnfvn0j6e8KxeEf2yMVZBrXVuydRdrc0ihRYWWu2VRWMvHs8Grs
GPf744+bmzZubT5O6ZVe7K1eumG+55eRLZJ3ZJ9b9zR/fFhzYzJIKbVlZWTcaluZ/dn27t1b0ZdRZRWl
VXq8VHZ43NGjGHHUjBGFi5caD548KA50TnZfNddd5mbNWtmzsnJsnJsdR7/PHHzX5+fuYtWL7evbs5LCy
s2L5nzZpl9vTNK9bt87Xffme+55x5zyzyYtzZcvX7Y7PsZw+WD8OpaSjo+0tDSzl5eX+bvvvvrOU9e7d2zx58
uQSn4vP0I6BMexev/Hz9aVMmhUFkyYKYvt7XGWPz9mUxhlDJcfe8aHwmm6+TGytdLRbNnS3uYaMWKEqlevr/levrn/
9619FHi/q1yo/Pz9lZWXJw8Mj6pqrv8Zd9PTs9KOj8oenzMqbgyXZ
PxKjOHyVNQY+eWXX+Tl5aWtW7fVlholaqSEhAQNHz5cknckknToOCEFBAQoKSmpyIVyzWazfH1H99cwzzz+j
ZZ5+V9McSBd7e3oqPj9fWAAGxWbi4yGQyWR2zhc/QlR/vw
Y6lsr/HVfb4nE1pvAdLj0HyYs/4sOtWR7PZrkeekekpr165VYmLiDSW+JOm1114r8lHnZdGPvWXAg9P2D
oVauWgoODDr1svICCgRPUlAlK+SjGHb+V27Rqae/fu1dWrV57OHz6+u9PROqzaenp7q1q2bk
pKSSpz4QvniPdg59e3bVwcOHNCO7Z2ejZSkEc3fFsMnH9eHsw4Lh4D3ZedwiW+oqKilJJCQoE8++tSw4vqLWOExPT1fm
KTO+X9McH6YJ7XCMjI9WkSRPFxsZKsnmbOnZs3Xy5Ek98sgjpXWwpXWpAAA4vqLWOExPT1fNm
jVVt25dq7re3t6W3Wcr166tBgOaWMr5DA0AQNmwK/G1g2k8SX8swpfJqtRh2rlzp1U7SW0ZafY0bO1Mb0GJwBGCwj+vZJmz15+Skc
vl7hx4yRJaWlpcnH582GcNMPtKjO9HTVq1dPISEh39b39zkQMA4ISioqJO8OBBBd+vULOHxsbq

xkzZlTIuYGqjM/QAACUDbtvdbyexMREq/25c+dq7ty5dgUFAEBVNGnSJH322Wfatm2b1VPbfHx8dOXKFWVmZlr
N+srIyLD5aPSC8oyMDDVu3NiqTZcuXWzGEBMTo+joaMt+wRpBAErXtZ+Z+QwNAEDZcLl+FQAAUNaeffZZrV27V
l9//XWhNTRDQkJUoOYNbdmyxVKWmpqqtLQ0hYaGFtlfy5Yt5ePjY9UmOztb3377rcO2Oh9rBHl4eFhtAAAAgKM
i8QUAQCXw0UcfKSEhwbKGZnp6ui5fvizpj7U0JOyYoOjoaH3zzTfau3evxo8fr9DQUKuF7f39/bV27VpJksFgO
JQpU/Tqq69q/fr10nDggCIjI+Xr66uIiIiKuEQAAACg3Nl1qyMAACgbWVlZxa6hOXfuXLm4uGjYsGEymUwKDw/
XwoULreqnpqZanggpSc8//7xr99VJmZmerZs6c2btwoV1fXsr4cAAAAoFIg8QUAQCWQlZVV7G2Frq6ui
ouLU1xcnM06167FaTAYNHPmTM2c0bPU4gQAAAAcCbc6AgAAAAAAwCmR+AIAAAAAAIBTIvEFAAAAAAAAAp0TiCwA
AAAAAAE6JxBcAAAAAAACcEokvAAAAAAAAOCUSXwAAAAAAAHBKJL4AAAAAAAADglEh8AQAAAAAAAwCmR+AIAAAAAA
IBTIvEFAAAAAAAAAp0TiCwAAAAAAAE6JxBcAAAAAAACcEokvAAAAAAAAOCUSXwAAAAAAAHBKJL4AAAAAAAADglEh
8AQAAAAAAwCmR+AIAAAAAAIBTIvEFAAAAAAAAp0TiCwAAAAAAAE6JxBcAAAAAAACcEokvwEns2LFDQ4YMka+vr
wwGg9atW2d1fPr06fPr06fL391ft2rVVr1499evXT99+++11+42Li0OLFi3k6uqbt266b///W8ZXQEAAAAAAKWLxBf
gJC5duqTAwEDFxcUVebxt27b6ZasGCBDhw4oO3bt6tFixYaMGCAfvnlF5t9fvjhh4q0Ojta0adO0b98+BQYGKjw8X
GfPni2rywAAAAAoNSQ+AKcRP/+/fP/fXqq69q6NChRR5/4IEH1K9fP7Vq1UodOnTm2++qezsbP3vf/+z2eeb
piRMnavz48WrfvrOWL16sWrVqadmyZWV1GQAAAAAlBoSX0AVdOXdlSenp4KDAy4OWWfv3r3q16+fpczFx
UX9+vVTUlKSzb5nNJp0ys7OtNgAAAAAKgKJL6AK+eyzz1SnTh3W5d7e3kpPT7d5d5e3kpPT7d5jtjYWHl6elo2Pz+/Ur0GAAAAAABKisQXUIIXccccdSk5O1s6dSNHDmy1NfriomJUVZWl
mU7depUqfYPAAAAAEBJkfgCqpDatWurdevW6t69777bpF1GZZsqGrVqikjI8OqPCMjQz4+Pjb
PYTQa5eHhYbUBAAAAAFARSHwBVh+fr5v5MJl0RRx2rWrKmQkBBt2bLFFqv6WLsVsuUGhpaXiECAAAAAAAAHDDqld0AABKR
O5Ojo4d02bZP378feocePGOnfunOLi4tTX16d+vvv9d9VVWnTp20OoYNW8Voal7VoYNW8ROBA8UZfVWVmnPnj1aunRpuccPAIAj41ZHAAAAoIKlpaXpzJkzv2ZHAAAAAAAGAMAwCqRcQ+VTT6d1pZw0ItFZGuKgR6cKNnPaukfa2zpR6KIhiRFaFTk6KTFetLNCKB+oq6QcAQ3gKfp3cc39bG6xHdjkkpqSmO/pZVv/SZjBmPLBbBfGvt6rqXrkNAnEwnqP1g8Cm5oG
Ufe9i/CRe+MOrzSxWfrKZmYrdcZmHDRW3u3K5efdrZf0yuvXWOrxsgJUBIzmTW5Xl1y8SNu+j4rJzd/EwnJuVjYrSUr6K1vRmeyFLGkmU0dFdkvL/1P/37AAE3N3TRNM+7yuhqcZ8/mf3PxqOa7tGEzrmhwY61MbvH8CwJ1OcjO2jZ9+NqHHrhmr5M31y4JJ0z7yG8hExKuQSPJ4tCwFZfGAx6Iz3BxY3TyT0vK5/xw/2S3h3Wjm0tqNqzrwIXSyjF6LEVGdBBxbB+9OyB7/6WuBCJxqi+NmqKr6oyEHaxHXkHSXjHYbAzL13BQgxySL/pnKSdlKaoN3lr3mZn28P3MiQQhOVt+PHkX0WB6uHpV+5bZxoqYpLkpDZY46n9OvCdTLBcf8QO33kn4sNarw+X2g6n8TfPxnPZoHy+bHwxdtT3xm2Zth1rCNbLvNOE5wN+8Krd0x3yW6P8n9vZzzf5M7D2y6PWj8RaWlZfv4fTvS9fa5rJ7B8vmF1MnmZGU2mM6AA1mxr32dWhKWOE5KWzLfWQE46sGeGEJLdK+1MAl9ZvtCKbqLK9i9PzZnE70+XaaH6k6MUwk9/Ib96O2Wyu7zS1Xy0Gx6UiRHJvvqPKl12DvPLnnnkr0+/dkdD3CEOoyyGHcuCVJ3nPfXK0A4VPe9NLQLNB6I5m3CjDYzcH1ZTBv+CtHk3h0zl+vPODkuzOqjkfYV76vG5r2u1y9hPBUfHk9TF6l5Vf7aEpRgsmLKIL+Ty9RWVy1+N9M0A9X6G5W2v2s/ot+TYKMAZmrAgDEl6x9zlC5cq9kWJHf7ye0nfFflQD8mTjlXF5bthJubKUHd80HXB6fwHhQY67ajLtqmUh2ZV4X6Gz52Cwc5NbhxswHx2PnH71tlR/9Lz2OPsDYPXhl7kljVRIz2Lj7+xuY1wAsrWPsW+ddJZZk/ljsMDD4uQAxnFBWbWS70gG6lSaxBIa9nwtTkgwnrdCHSVgZyVLmQZcRbyuwvH2dGmiLUx9iLSwb6UkzxzTpV/MXm4JD3bgvYJzJK/Fu8a5KOaqwgyfDG8cvdsxJH8jz2GvTxEp3WJrTPfF8awiRNvJhAs0vfb2EkbITAUlzZOq6ZK8ljh4Dwr1l9XiT6Tlg3TC4PmtTaKJeKSNxwhEZ4eXVaPWR5Zh3XuUHHXgWXqJKR03RC7+XC57Rlm+lUe1LBHDY/hxDbSzyh3rIKkyYg5JqtmVtbBsAHtNlp8bHYL3WWZhMRCOsqpVUyRjm+gGkaYJnScUVIRlTYeGUQv7YvGPD1Y4mG0sbtyAcnN5P0y5YBCfhNsC4J2tlE92/WPGk2OE0O1xnbzVSWFB0UGqtlSz6m3/HTL8l6GSRfdOhCr4WOAKHaOLX96wIXVdPgVXSTEjbQ3GF4HoOpbtEOPt1OWp6rN7IHlhLiHiP7QHFiSQ2Rea2N9FpgQgxbkMCFzsOXSPb5YhgoXbHCPXHtFlfiTCs6nMpaPqoIM60eHF4Hhr/U3tcTYT7tP8vo0l7RoGbdTtXo4pVkfB2wHfpGpQh4a3F+Y3o87yc9tm8+F0u+vRh6gHJXMlVrQ1tl6oiMOvEq3afwzd75XfDk4jAh3SqHHr8lIhWfPzC3XmPhJz8gSXDGzBb7A0ljqjzHv9RUNG+PDFdHvs7vDfZj1LDSrh3JtaOqIdvoLzUYc2XbSxmvI96Yf2+H5jYTVOdrz0t+6zS1SGYBbRvGV7NVYD3S8AGG+lFNaWI5iLmYxV89NmWmgvgHr1d8sh3RnD5vSN3Gw8LGjPudhJtPKhqLfqaQ3lM0ZpHOTqivmzl3SlHaKAj0EqGdk0RhRKD9BhCzc7nw6ZQyMT7ddu1a1h+Nh7jvm/Zy6TLbJB+h+Ke8Mhd7A7tKTHnMg4ZLufExlKvxV0R+BDTVtTEq2J/3t89g4MJRuSvQgpx63XbL2ePPKJ4NtCp3Ax6HtjrZ8hEh4AWrXhrZm6DhRbl4TVa8VQaCSLtEn1aDJQ+gEuvDaR9sjnfaXqAYgBwKhAMQvmvg1hS4nsOHtXRwqhXVBjGj/97Q3xIevtsbUhkJqc3YGmIvd7uOwAcP3HuUkVLOjMwCr1ONqZBo6SeggqBLrhXQyTw9ORZ7u+lTZP8zLCbQqRRWNxTIzAOKqNRwkLHfnRfcO92jlh+Dm/ntm+3tYMmM6+YzFSDRV+4nd9KR2MDn8sJZdhkXHAL1wkJ+o9BUdymAvAUKKLGrG7K8zNdV+cFvB+oSQqNVOT/Pqvh9AmvwW2VIiamhjuqSZU7Ah0q5Tw4xSRh/Pm5FnCqFLBSFOCb2sWv0X+VAdF2eTKBgHXqSHJ5n2C2p78mvRNM5aN5cmtMtYJCgGkl/jWYcaD8L8Y0n7dqBBw0tdM8qgrlKrMQdKLR3Tg1XQNLXLKTv6x4K0gZkE2HrWLP5+m9DzaDzkRjNLmb94XEUJR6tQLO7zqRSr4PR1gDvQGGKj2rCWZXq5PT0IdMKOq0f5KStsXgJJXCq1V4IH3WT1mK2X5QL6YWJkzAKWQ3HA1KBzj3c2WGF/pKzKw5kUuMgeXaKBCfJx7g2tVV0oIIKy3j7cAFmAFNYFG3J9mDuXpfZyrwFpkGVj1nz/4Lh5iCiOlf7rJFnLc/tLUpLQsp2bpRPc0JnI2Uc4UB8nZtV2IU7r3F8PBUnO+7S4VHyMJwh/MEbfG0KNfw8ldQMR6mf0/eOwbODfvLFqvPmOCyvhBQEDVrJDoDE1jNqbYXS5ECkWq3rI8PrE6HyUnvK7O9aMvlEY5lq9nAxlg3ExoxRUAvD+CAk0nVYmFCjFl8BPlBAu7xJ6RVV9iHIHH2ngVV9k5AQdbNlFS6ghDFkeABSO9OlO0mqZpchxc0S3mTpVmxYM6a5pVzhq4M3RHG3qwfCXmLbEHTCgzgRI2U8qCbE6hpOhK8gp8ryGqvqvYB55ZIr5bQBSEnNyaPwqwKzvk2EALRo1EDKvIQ5gz2HgJvRAj3OycrVbmhqrcbZsAzhZx5xPsCTHwW1qZF7MGCrMvNtWglcYX7JO5B9R2tXtabpuvP8WFpEBr0vU7YCvtXPEqbAQUffGClbIdXukR6X1mmdvLsO0Sf+lMk7bxdsqJ7gWKXCMVxcgqSuRqsdsrEzdQjXtPgxubYjcwlmxZ9hWMo3pKjvbX0qRm1Xmuww8WPb+nuRpt3TnULS1DiTdGxxGYvIkFa+/e41bbOrDJE+iHwQm2lm+9T1Hns0lM/6wZBQOd5nXmb2Vv0gUlNoQxdINxcqCiDVRX+ZsEsrHz9mv4R0SXn8jxrq2mnI0iwvVZuWLs6DYi8wNe5h8Sx6Aq/8Z9K7gIVQwdbSfwsqQrVBVXb4tx1A1j35t8Hf5ZfyKQ1zbQodK2MEMi0s06u5zhsBPmyD44DWe0jC6PfFEcSL0r7xZa41y6uKNHXCzN7hPJ1C7NdyNgSPlFYlCQLeg7Ty7AaLGI7mrajwzCVSUO7Aoc5DRc1kNiJTzWAp9o3PZcXs5Uv1TPqI5GxJdMEGPMUYaYVMhGOr8Nv/2XOCZCpQF1N3Uo2TvuKEOBFmN4RI39vWHvbmQJaE9ZRQoRbzvY9lfGhKlY4s4qGsT6KbuaqUwSR0NvEPXJQMUFQWp/lDtrSPOOnSdKdWAHnI7HQ3JHz3DYaIaVmcIGnODW8UnmIqy

```
ZJykpyereZkkKDw8v9t5mAAAAAAAqox49ehRa8ufHH39U8+bNJf2x0L2Pj4+2bNliOZ6dna1vv/1WoaGh5Rorg
Jt4qmN+fr6mTJmiHj16qGPHjjbrpaen231vs8lkkslksuyzqB8AAAAAoDJ4+umnFRYWptdee00jR47Uf//7Xy1
dulRLly6VJBkMBk2ZMkWvvvvqq2rRpo5YtW+rll1+Wr6+vIiIiKjZ4oAq64cRXVFSUDh48qO3bt5dmPJL+WNRvx
owZpd4vAAAAAAA3o2vXrlq7dq1iYmIOc+ZMtWzZUvPmzdODDz5oqfP888/r4sWLevTRR5WZmamePXXXPXtq48aaNcnV
1rcDIgarphhJfkyZNOmeffaZt27apadOmxdb18f9Gx+97mmJgYRUdHW/azs7N5ogUAAAAAoFK4++67dffdd9s8b
jAYNHPmTM2cObMcowJQFLvW+DKbzZo0aZLWrl2rr7/+WilbtxbxVm7mzVp8+bNxd7bzKJ+AAAAAAAAuFl
2zfiKiopSQkKCPvnkE7m7u1vW6fL09JSbj/fL09JSbm5skKTIyUk2aNFFsbKwkafLkyerdu7fmzJmjwYMHa9WqVdqzZ4/l/
mcAAAAAAACgLNg142vRokXXspLhxY8v24YcfWuqkpaXpzJkzlv2wsDAl1JCRo6dKlCgwM1Mcff6x169Y
VuyA+AAAAAAAAcLPsmvFlNpuvWycxMbGQ2YgRIzRixh7TgUAAAAAADcFLtmfAEAAAAAAAC0gsQXAAAAAAAAAAn
BKJLwAAAAAAADglEl8AAAAAAABwSiS+AAAAAAAA4JRIfAEAAAAAAMApkfgCAAAAAACAUyLxBQAAAAAAAKdE4gs
AAAAAAABOicQXAAAAAAAAnBKJLwAAKoEdO3ZoyJAh8vX1lcFg0OLp166y0jxs3TgaDwDWobOHDgdfuNi4tTixYt5
Orqqm7duum///1vGV0BAAAAUPmQ+AIAoBK4d0mSAgMDFRcXZ7P0wIEDdebMGv2r3/9q9g+P/zwQ0VHR2vatGn
at2+fAgMDFR4errrNnz5Z2+AAAAEClVL2iAwAAAFL//v01bNiwYusUjb5+Pi8UuM8333xTEydi0Pjx4yVixcv1
ueff65ly5bpb3/7203FCwAAADgCZnwBA0AgEhMT5eXlpXbt2umJJ57Q+fPnbnda9cuWK9u7dq379+lnKXFxc1K9
fPyUlJdlsZzKZlJ2dbbJLJ2dbbUBAAAAjorEFwAADmDgwIF6//33lWr3lc/++frL29
rYq9/b2Vnp6us3zxMbGytPR03YWfWBgamp7EJ3amjTwMq/KMjIxi1wkzGo3y8PCw2
gAAAABHReILAAAH9NNPP+n8+fNq3Lhhxkcdr1qypkJAQbdmyxVKWn5+vLVu2KDQ0tLzCBAAAoUiS8AACqBnJw
cJScnKzk5WZJ0/PhxJScn6Oy0tTTk50Xruuee9cunThxQlu2bNG999yjhV2bNG9996r1qbKzw83NJH3759tWDBst+dHS03
n77bb333ntKSUnRE088oYsXL2pZNoB8YQ4lMI3/9SjDHhJKo4I6SD0/tBA7/+Ea7OLnG9CZWQfqAtxfjc3vXgvTz3
33KO2bdtqwoQJCgkJ0X/+8x8ZjUYUDOfKP0+P7OkhE0Ci0c2YbWW/iO6itLIJXXtXhX7+8Ji28e2wsKO2vOR7NkOaU/BZs5wqeb6du9cBxQgEs2LCMqaavmZzUQXcR0c+n8k+I8y9C4JMDhEMvRbWhfD
ZhhRlwCWq2w3zUrTQ/q1Rhw2jJ6Lmp2Sc9N6xPmNWCUBcs77lzXP2Kc4EMtXa1ZTd76Ox2Prfj4jDJ55wsjcaWU5ESH7+VZmNeEzutcbvhhNYGxBd3xQ9+A6n5
rZv9p4X2vaNGQcAZXZJHQNdL0s08PJ6v5CWjx1/oGrGE3lAnqtLqvfuqnk1vmGzIKNLrmdhX3xkKJbAgeq9sm76f7mr6lbtLkC8mvfp5fBG4bn+1/PqaV9LV8zbVNx6lyTIPudXPfr0hx+NF9b9mxfb5aBWnV84aHH2mdNoFXcS5ZadWPo5l4+bhZ1ub5Mlj4ytpvUG5OBqWhAVZe1YpfsUoyrxSJ0gMJOHGobDiKdUSZ9qs9HGPSKd9gNFeq5dGh6w
vXPr1l3Xq19fhVJrrhNs3m9Cf0UCeb2Txx2nc0dN2VQ2K2zXfN7qJ7m5yTAoIKx5T0qPA+4zlB8x5TnhgQAi0Q9r+KN2vVhGgwLvYFZGk0wyfY31MoZRH7o47j7NVqltqdd5f+8sSgf6HSaGe/YfXQ3KRp3tQdc+DROlTqxDd1UCWVpEAAQ/2JoAf8IBXtbdQ9hHi7PSqoDBfqs77tbVjlcrJuKdH3s+CG8z7Y
AuYtjNPGhlTKH4llBZ64yZqqnN2q5e3A9ckEkJF3ApQKgqPf8fNIBoDkTUJ5lsdqPjnBk+GSRlG1c2+rN/wUcnfb8t78e27kljuxW24CEWxVjWZkqaBu4Ln2yBOzyHeJHzyHwEcnzdmfEgi0YaPjs/CW7bSftQB1rn9lZbx8ZDZxKHfjRJh6C8FvUiSKpm9WQT92VlXn+AXVlqP2rnJnY0Og7ZnGQF7W/qSNsNm3IwYRzAdXzAIm61AuVNf9WoPuaTZ06Gk88z1/xezZLf9z2KzpHmprYhC8wjZwHvXfiDSZSuC
cQmKCZaoFwpSgUVZLM1CDGmcKEJuYZMmNe8F2qU2LnYc+tenaGNdsmcOoBvuMx2B7gY1xb4xwvYjTVKXsoPI7wvsp4flsBHCG37vf0M8Z8oj4CT7okQwmCFp9AO9KBE7AB9qZjMu7e/hkXHC+mEvSbW1XN9AgAMfEuf8vqwHc/k6jNpyQAQP9a71gKkCnkUYg9ohCO9nezTcnpRlVqKABcK+6r7Ldw9mVq7SKMNV
```

OBgCmZ6rVixQgEBAXa1vXz5sk6cOKEWLVrIzc3NrrYpKSkaM2bMDc0OQ+nLysqSwWBQ3bp1i613+PBh+fr6ytX
VVaGhoYqNjSO2UQYAAAA4ExJfAOCgAgICFBwcbHe7Hj161EEOKE+5ub164YUXNHrOaHl4eNis161bN8XHx6tdu
3Y6c+aMZsyYoV69eungwYNyd3cvso3JZJLJZLLsZ2dnl3r8AAAAQHlhcXsAABzI1atXNXLkSJnNZi1atKjYuoM
GDdKIESPUuXNnhYeHa8OGDcrMzNRHH31ks01sbKw8PTOtm5+fX2lfAgAADu96D5DJzc1VVFSUGjRooDp16mjYs
GHKyMiowIiBqsvuxNe2bdtsPnWqKImJiYX+IBgMBqWnp99ozAAAVEkFSa+TJO9q8+bNxc72KkrdunXVVtm1bHTl
yxGadmJgYZWVlWbZTp07dbNgAADilDh066MyZM5Zt+/btlmNPP/2OPv30U61evVpbt27V6d0nCz28BkD5sPtWx
4sXL9p86lRxUlNTrT6ge3l52XtqAACqrIKk1+HDh/XNN9+oQYMGdveRk5OjoOeP6qGHHrJZx2gOymgO3kyoAAB
UCbYeIJOVlaV3331XCQkJuvPOOyVJy5cvVOBAgHbt2qXu3buXd6hAlWZ34mvQoEEaNGiQ3Sfy8vK67gK8AABUV
Tk5OTp27Jhl//jx40pOTlb9+vXVuHFjDR8+XPv27dNnn32mvLw8y8zq2bNmpKkvn37aujQoZo0aZIk6dl
nn9WQIUPUvHlznT59WtOmV01atUOevTo8r9AAACcjKOHyOzdu1dXr15Vv379LHX9/f3VrFkzJSUl2Ux8sc4mU
DbKbY2vLl26qHHjxurfv7927NhRXqcFAMAh7N+/XOFBQQoKCpIkRUdHKygoSFOnTtTtXPP/+s9evX66effrK8nxZ
sO3futPRx9OhRnTt3zrL/008/afTo0WrXrp1GjhypBg0aaNeuXrUqFG5Xx8AAM6k4AEyGzdu1KJFi3T8+HH16
tVLFy5cUHp6umrWrFlo4oe3t3exzvPrL/O08/afToOWrXrp1GjhypBgOaaNeuXWrUqFG5Xx8AAM6k4AEyGzdu1KJFi3T8+HH16
tVLFy5cUHp6umrWrFlo4oe3t3exS/6wziZQNsr8qY6NGzfW4sWLdeutt8pkMumdd95Rnz59902339p8GhmZbgB
AVdOrVy+ZzWabx4s7VuDEiRNW+6tWrbrZsAAAQBH+ehdU586d1a1bNzVv3lwfffSR3NzcbqjPmJgYRUdHW/azs
7NJfgGloMwTX+3atVO7du0s+2FhYTp69rDz74oMg2sbGxmjFjRlmHBgAAAADATfvrA2T69++vK1euKDM
zO2rWVOZGRpFrghVgnU2gbJTbrY5/ddttt+FEKQAAAACAUyh4gEzjxxo0VEhKiGjVqaaMuWLZbjqampSktLU2hoa
AVGCVRNZT7jqyjJyclq3Lixze4VkugEAAAAlVVxD5Dx9PTUhAkTFB0drfr166tOnTqaMnmy0vDwONPPfWUQkNDeaIjUAHsTnz
l5ORYzdb661OnmjVrVrppiYGP388896//33JUnz5s1Ty5YtFDB+Xm5uqdd97R119/rS+//LLOrgIAAAAAgHJS8
ACZ8+fPq1GjRurZs6fVA2Tefvtt1a5dW+MpCaTSeHh4Vq4cGEFRw1UTXnvvbs2aM77rjDsl+w+N7YsWMMVHx+
vM2fOKCO0tzXL8ypUreuaZZ/Tzzz/Tzzz+rVq1a6eZKnMh9tvD9+/crIiJCo5gj+Sb+Pj4yMDBLpFAt5w2TCGaZ5ow9bEFSUl4pH8Ia8dnNQEYIcp
mb3j/jSjMDVFm+cA3CjY4yE3fb9HmuUZl4BDBijUSjIxPyB2/p+tXaJVzR0TLEAjKJKHUx4Ep+JSd3JB7IXQTg
a9UJqnlUJqmZbc2gxyMZJgQSqEnbPgZX3oy7qsmJcP3CkrdoKcMBZJYbaSSX5ilCyoAxOxj+jWKLsYyi72A
9+JKtHzcx0xGLvtrpcZp4onOwdMsYCDKbJNwSjhHGRlJglSJWuxIFmaklWqmQe+FRjhMlQAl2tV3Pmg
J/7YMtiuHp+4I9v3N+6odjNN1yn8Pzlhuk/22d+eO3Po/HAwrhpx/VTJ2XyI8Hmq3JnN0c2W+P4b+WAO
DLZwZ28ikDIOLrQEBixgK1xhq5ZyfSFHJaaOG2mhKbnDBqLtcJ+yK9zakRDXqa3bPVHnCZqn+n62pp4RUd
VkDVdRMIKtiW3iqZmpcCqs/Di4BDHqg+4ZfUt2iYdt/HPszfeOAV+lqR7LYyGAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAA

GTZkyRa+++qrWr1+vAwcOKDIyUr6+voqIiLDOObdvXy1YsMCyHx0drbffflvvvfeeUlJS9MQTT+jixYuWB1YAA
AAAzo7F7QEnkZOTo2PHjln2C740169fX40bN9bw4cO1b98+ffbZZ8rLy7Os01W/fn3VrFlTOh9fmocOHapJkyZ
J+uNL89ixY3Xrrbfqtttu07x58/jSDJSR/fv36+6777bsR0dHS5LGjh2r+Ph4Pf/887p48aIeffRRZWZmqmfPn
tq4caNcXf982MDRoOd17tw5y/6oUaPOyy+/aOrUqUpPT1eXLl2OcePGQmv3AQAAVAWG33MV5OMit8wfpdPlNw/
ILfNHBfm4yPB7brmdE38i8QU4ieK+NE+fPl3r16+XJHXpOsWq3TfffKM+ffpI4kszUJF69eols9s87jBYNDMm
TM1c+ZMm3VOnDhRqGzSpEmWZDYAAEBV5pqTpn2P1ZG2PSZtK7/zBkja91gdpeSkSQ0rvxNDEokvwGlc7Otzcc
K8KUZAAAAgLPKrdNMwUtytHLlSgX4+5fbeVMOHdKz+9qVm7nxJ9IfAEAAAAAAKdnru6q/en5uly3reTbp
dzOezk9X/vT82Wu7nr9yih1LG4PAAAAAAAAp0TiCwAAAAAAAE6JxBcAAAAAAAACcEokvAAAAAAAAAOCUSXwAAAAA
AAHBKJL4AAAAAAADglEh8AQAAAAAAwCmR+AIAAAAAAAIBTIvEFAAAAAAAAAp0TiCwAAAAAAAAE6JxBcAAABQxmJjY
9W1a1e5u7vLy8tLERERSk1NLbZNfHy8DAaD1ebq6lpOEQMA4ByqV3QAAAAAVc3hw4d14cIFu9ulpKRY/dde7u7u
uat0mzQ21xc3ZunWroqKi1LVrV/3+++9Ce8xbgQAOc9BSQKCqtSw6DerF2i+qnFa1VJ+yKnzz8XB1ZSa2jtLexEbyhTUpWRyWJb6P0iY9iF+sgMbkNAlWGXWw1G8R4NV1HmPNhTm0NVazF8/mDavGFADMY8vIAXhVegCsxO49YsfW7l8hG99JVRZY5cyi/3nzG+M8qNjbDxe6Ao9ReKK9Lzos4T5fAkF7Wiozl2WPTQL+ZtzPV5eDVjqpuyt5XIdJbNPOcUFsxMS8q3iaUrfkMr6z+nW/qPd2/GD6oBWIqjyyM0PjI6obZZ8ZJfe8bNPn1bMzRjY1YSVCJ7WhiaHCWQvs9F3vkXKBFHMQQHcbkxZmRKOMzVo6MdSdLrKGLoEVwdDZxd+1Dt2mcg/mTZmg0nPBv2Tc79d/G9e9uzdvhvj+MmrU/xlYX4hcM95gE4iJA01o3WzA1AQJVbzSeHOAnpI0+zrJsXpV+SYcdUldFkM5Rx2ZfKR5xFOXoYAHfgE7MPnS5MH8l+VBTcwPo+bDTEiC3U6CWTptSl5zC3Y4ZS2M0TuMkwBwXtiPRkpSbvj86PTxdQzRuFfukc+ZSqbPWLaf4gJ1k/SzoxB4+uGq3JZHZvWqCtJYdS3ax6NTb0w9Ts1/L6zlJc2+nlZF6IfAtWG7GBjcFxJTyP/h0ahRA6XffD+Wy/BStIr3ghnnA6eXUhYRDOmqHqD3fB+ZN6d9+t2aDeY/wOMOD+FFvOvMoi2kwo5z08ypejTNx8gUKRBWuWzJiPA1yh7UMJRMRZEEgsM3TlUPMrjB0NCnEM2VeF0IOWoBjUjwWRwXhb2ODxiWC8oerV8NYBlVfmSPhsy4Wg4GaJ+dyxL6C4j/nf8lePFGyygjMbS27IpHmLbQ6m5PU5e8tu6G3x3sSvoubKVuMD7tv4vf2JG4dtiNC3SOtbrFJZK3ClRn4CnpPM/9w+NbqHQGZtyEX53xaHVP+Q/9Jk8KV4n5rM7XBVC1eHHb7G/KmpOOJcK3+iO0JTnVkBk+nnd1UVIOCEJrf9fifVaNvKV0eKVdtnJM+ezSMB7fw/oZTDUZ1kUjx9SvAjSRr5kR7PEtMy9pPM4S9hjhH8Rf5anX43Yd5cwWVFANa0vBRX0d3rZBQ09dPmSkAwt8Ng/C0yHIIv4fPMhHoKc0Pp0nNN2uFD6r5vfn3WKwQtawdeF9VBj0xFs0qVR8I8KjddZEXCx+r/fXdV9QrH5fbuAo1Otnpyl5lyKLvK4mgGyQ5V3aGZLaD2XhfXlVPnhHiOZsa7OqdZWMxhBbcJAzvAYJxdhXCnV9Zm5KVMJ7uE4zz4bDLKgQmS6QNvHLg/0h30umXU9fOgSvuHT2fAXlcRkmGCYx9XA1uMfcmHIG0QzBn52lXvEC/uY/kF9KbchQYIsTE7iOPtvMWUz1MyjJQnoRaz1bVZ+Xz4o1vAyzPyI7h/t6FBOWDJXUXO+uPK6iaaJDbNTEfDs2ErEeQldDGrHdlZkuY6Uiir8JfHQJvo2TFhYG5A7NhB4nCZ1lWVSHljkRn6kpUTL2dxBvSUXw+NRZOxPQuWUlwLa6mM4Bh6x8xJQvqg+fsmhbhpmKkv9oc/RJQ4qMLSe1WlvECOMkRmsVPxMM9sv6PkbAoBLk1hcsYkGFRGJJkb7Y8NH5kvyhNEVfm5WJMnBFKVLXm5Ni2+oWR+lqRobmmBAOD4fDp1tWBk8uTxrvMCmRGXhgEmapWmgXRWMgx6IhoyCc8+blKc0N2tLh7s8tL3A7PkiOMW3kZpcFCVUNbgy1ENVfTqJmr/akVW5Yg9y8Od94Nd6HYMoaPx5hQ7AmwhMrFNIiLPsC0Gr9oXBhcyXyEy8O+eYmsfrXVeLZXkmjRdA2Yb9Py0mLcE7nA0AV4QoANrDUHVdG47GfmAz8BBaLmlzIqLoSJeNMnfF7Nl8S2vfvQpSgqt31rHAbnJYvYhcwFlpI3LlT8+dlvjvaZDMhEUqCbVRcn4MHT3RK0GJXgOKlmjOBn0LV5/V4Pq6UuWH+7K1qPJtu3Sa/kQZsF0BnGODjr11L0wqoFu71kTnwgVvkY8ENGy5p8lOMNtJrTfSgF7PBgmJyZuyDhbCOw12DwLHIgjKNBlbEG2sHR5B8AbNvi5eTgMHPvMkxX7ccNlH6x4x5pwbPq+BqxIhrW5ziUxPqrIaMFeIhGNU4cyY/ZKqkLOa1lRqaQg3z+1HRTcpw5aC9cZm4/r5Qvs9/s0SKLt+VsK6Xq0mbZSfPeCVA6cP6eCdlXi1aSFZDxRR5LSWXZSM3rzWpwq1cU1TaIcBG3uYRszYBqbVBWiD4SmxYhZlg7L7KJm8+OdCmjBdJq3+cYmQqhSnHeW6P+dXdB9/NbPU/WDUjG3P0lCbkDMGNXqpVAcqC9TTXxTx4fmR1W7WTpgqY2vh6rF9zBQbUBzuX2sivHBaM+eTrsNuIRPHbHd4O9pvnMGBf+RfuR6yN3hhfLvSGQPAARPQ7gYq0pAJ/5+ylUpwmfYTzwHLcqjjHR4pp7X7nj9a1ziQNKn3oNEDVBE3jyjkHDNq79G3MEbWTtiKtNLX2YvKjAW8aL6VeBQFkzE7fE+GqE6J/DJZr8YhJiEF1h9dDUNjnhX4zKbq8GkWhfVrPWUOdIKNFHgyk8kVJCM6HVKf8XLMR97i1lnCkqtkT2MOp2EhJ5eStsvaqxqnDeqwQ6dSROQwh1DGqxL6Zb2TPnHxBYBTZxuR5t5OkDJUYTNzlhG3rWvtqAyLJRJYGz3DWAeC0oJ09XjHhe16eDvtjtaLdKNQRyOXdxmCGGUL+vbzxXnbuHUp/VWW8ZJt/8Z4y4ciuC6rTTLj7X6VVmEwW5FvsLp9VAlrn1xUBYqFlmmjhbi+pVT3ZWp5lBcfYDxKgYvdSRNQyq3dPCpKIyIgTG/jqxbNvdKJ0sktpVKNq6OuRQ2PH1y2xW1Ys0YBbfU9VjGFhADkfevNhOsoZqcltZUkpWyhsTTROTKaZnrMa4UjgOsEqVNs/+4Srg9HhiwnCCi5tCdRfSNzXq7Cul62lkkdbqpZ/cfx5sAymY60pRPZbvsa/YlqUaL0QpyJ/K53dvTtYbSZdyslGbl2RjwGFcFwVXOV9dsKVW9AGLp2Rj6M7YgeJkSb82sZFs6wOFHm8qqvK99RGPKjvlKY1gKKSvR9YnK2s8AFmwmVyYmLgsiTlvKmwdUuYpU8tmxTBrasXB/vcjjZ/m05UY7QxaHyVEKTdhGU1CgsHwmqNJrD4YZe0zXhPkkBH/2q1HCEvdwL9q9nxbvDAHMLaqc0tznqrPdbeWhCFRXUkc3KsiuNGbSDc1FAqLB+dVUHvDKNGA38tr/Uy2BzxGOo9MeuGNg/ShgH/8QD0K5pdR5UgEYxjm8ULKxOi/lKnsDqzbYYz6/JEUwJpnXO2fLwfamWCFRWoNJvQRcMGqMaFz4KaHTLLEvgwjlrWuNr7z4n0kc4umPqtVGCeyWPwwNWkbyW3qMEGcbvBgZjyo43TLhwSNeaXXpS1bM2JspBZQMvNFOuKO9dyNH+EhhzSlMmqzL/xgU1Ej9ntpZ08nKmzBqrqCFRSNGfpLOC/Lm/1JhzUMqu8yUJIgfzmtEOmTjr5Z3Up8BADaBtgF3H3VRJWGfSr1jQnC5ZzgUlmG1x6G8w1kMhaNQY8Btk31Ab5skZzRNGxewl/JP8/gvHKLjwlt2mqYg3ESmaaUgUBSDSKR6ZT6I1qkmaqqmh8mmBLPz6btnedSFiQX1OCaRavLZxpanX/Y5idSnUqrmC/qcM1ROrCi0UQGaHZtpiopdJD/6ZjAvGU5W89s+lZo+1NFnePVdWQNR/gc2OMHo3pnFpWyuaFhCLNKM6K3m1OmUHMu0NnG7jdE+gHY/+C0cf2r5E8l7KD4PKL7UyGfZpIpLQa5EwO2DoCsrLvzF8vgP0Fp3j2PfC8rVm2zeM0S4l5V2kxeTfA3PJNVnhebKDNOa4I8bYaYEAbEaFKhzo2E/Rgq1gIsEjx/2e1/X/ZsqGQlYKhLN/j7RXRjrrhX4X6u3RmXMW9fU+7DjxA3DlzvtJsG6h0iV0jffR1n7R3JCE2Ouz9z/Ce+Jzzpa1L/tHNo7+K0QnrSV4rTsOWfpDLgAuuQm9WNbC+fBSJQ2cg73A1HA5DtcpDEZmGYf08YmBuuIUSqU1pq/ffmIMQhAAvWbKkBF1V0dH3dw3wM+HRgcNrHpTYNDSrnZJW8zAZWTgsyhWBNeY29aFhNv6uu3HYaBcMRfPqRA8kAQFHlNQ+u2qrw8+cZcBHkjcSW6Y3HAoLc7aMa9cXDZZDjG90J5rvPJ/Q4zdnVOZy7jXk9CBzv/Gd9yOLZ8G4D5V3+6UsCdLQIbwmpMOJyGyHRjFT/xLRHO7FJUEDK+RK/TjmRO0MEl/5a6z0dtjYQ3Yqd3jZ/jHAfSRYwK63JvmeWhLGKo4n5tmDMU4Tnsw/ZT6rJYhHlQBEy4j9ViZ4MPKa+R2wpmOYg7jO1vH6m2bQF4bNYbUeYp1nDjARmJeR9uvLOI6AlBX3NOp7W/cFJUyzZLH+5sDzDU3dbQdMhqP23Ja5sRs7rY+U6pdtEoR9jpNxrtLx/7B5uaf4vKfbSSHr8ddwTQcuH6Pp9gTROdHdUdqIUxIqrYDDugMHdfR49iHcpZqvrz4Gu8JtfV4LJ6TNUbvBkFoRlgIZ+w/2nqGgqzadcX2Nx8PM5NQ0HxWfxlUhNIBxtNSy6KZvWMqJ2C1WENfNm9OSlVNGe37upAcmeRnXLNtL2Ab2LBVGVnx5YvQyvrunU+dBmzTlspqmxUcZ2svAlAEMlYa3ASsrsuhqmVUGiQmX0lZDeKjkOeMAeu1IIWXYtOLThuJNXOJ0CBnmJGJaHlnONS6x6zAcqMVT3rIGVZ9b0NAwkv2gMVpvLmcKrHMOSrFPzr49+CBWWrw69Ql6e09DMawgAAAAAAAAA

wAAAAAAAJwSiS8AAAAAAAAA4JRJfgJPYsWOHhgwZIl9fXxkMBq1bt87q+Jo1azRgwAA1aNBABoNBycnJ1+OzPj5
eBoPBanN1dS2bCwAAAAAAoJSR+AKcxKVLlxQYGKi4uLgij1+8eFE9e/bU66+/ble/Hh4eOnPmjGU7efJkaYQLA
ECVEhsbq65du8rd3V1eXl6KiIhQamrqddutXr1a/v7+cnV1VadOnbRhw4ZyiBYAAOdRvaIDAFA6+vfvr2HDhtk
8/tBDDOmSTpw4YVe/BoNBPj4+NxMaAABV3tatWxUVFaWuXbvq999/14svvqgBAwbohx9+UO3atYtss3PnToeP
VqxsbG6++67lZCQoIiIC03bt08dO3Ys5ysAAMAxkfgCUKycnBw1b95c+fn5Cg401muvvaYOHTrYrG8ymWQymSz
72dnZ5REmAACV2saNG6324+Pj5eXlpb179+r2228vss38+fM1cOBAPffcc5KkV155RZs3b9aCBQu0ePHiMo8ZA
JzNpUuXJEn79u2zu+3ly5d14sQJtWjRQm5ubna1TUlJsft8KDOkvgDY1K5dOy1btkydO3dWVlaW3njjDYWFhen
7779X06ZNi2wTGxurGTNmlHOkAAA4lqysLE1S/fr1bdZZSkpSdHSOVVl4eHihdTwBACVz6NAhSdLEiRMr5Pzu7
u4Vct6qjsQXAJtCQOMVGhpq2Q8LC1NAQICWLFmiV155pcg2MTExVh/Ss7Oz5efnV+axAgDgKPLz8zVlyhT16NG
j2FsWO9PT5e3tbVXm7e2t9PTOIusz67rs5V74VUE+Ljq5a73cMn+Oq63JZNLpO6fl6+sro9Fo97nTjx9XkI+LD
L/n2tOWwB8iIiIkSf7+/qpVq5ZdbVNSUjRmzBitWLFCAQEBdp/b3d1dbdq0sbsdbh6JLwAlVqNGDQUFBenIkSM
26xiNxhv6MAcAQFURFRWlgwcPavv27aL7Ouy17G99u177E6Otm5Oln723eRpFM3du4ASXc9Vkdp5vM31gEAN
WzYUI888shN9REQEKDg4OBSigjlgcQXgBLLy8vTgQMHdNddd1VOKAAAOKRJkybps88+07Zt22wuG1DAx8dHGRk
ZVmUZGRk2HzrDrOuy12voBK1dK7VoOUKurq52tT1+/Lheeuklvfrqq2rZsuUNnb927dpqFtT3htoCQFVF4gtwE
jk5OTp27Jhl//jx4OpOTlb9+vXVrFkz/frrrOpLS9Pp06clyfIIdR8fH8sH6MjISDVpOkSxsbGSpJkzZ6p79+5
q3bq1MjMzNXv2bJO8efKmfyUBAKCqMZvNeuqpp7R27VolJiaWKPERGhqqLVu2aG3Wy1D8FfMui57D
Rv7aeiTO2+o7eV9+7Q//UX5BIUrgNkiAFBuSHwBTmL//v+v26++67LfsFv/iOHTtWy5cv12Wefady4cdb3+vcaPH285fv/9O0u
Spk2bpunTpOuSOtLS5OLiYqnz22+/aeLEiUpPT1e9evUUEhKiCTok08+kbu7u
2WdLk9PT8vTwa79AWry5Mnq3bu35syZo8GDB2vVqlXas2ePli5dWmHXAQCAoyHxBTiJXr16yWw22zwfERFR6B
3rtg+EhMTrfbnzp2ruXPlkJOAABUbYsWLZIZOKZIk9enTx6p8h3+fLllvfna3+ACgsLU0JCgl566SW9+OKLatO
mjdavX69u3bqV7+ABAHCyvQ7OMZs3BVVRXb9KwkGRTjjCu/C43p3a1rFWvMMc4aaqn6MtkQtKgQsf7ZagzDwWWFrbHmkpsGM2lKg
M1FtvvaXbbrutyLrx8fEaP358oe/jXn31VU2fPr2smwsAAAAAAAAAAAAAAAAAAAAAAAAAAA+Zy3xn5u+uS1HX4x0UCgyaAAQkgAAB/4Uc
4wpzR2o+L2g6e23lZhkGMNZ7p/7m9nenGxx/7Qvj6zGBwZq1+pZX4KBtYgE4wpTdpeEPRgvaEwAABpY
lm9Vo8dx1IjgEAAsxWrOhKTBjKwrMqIWeDl1b5ObAm4fVdY2qmwWXfmy5ZzD1Ef6TXrZgOqXAWUzZGkZIt
TBDB19mbDB1bhFjJMUsgdrR0Dm3+qHQMhaGz8NVb5QcPMA6aPfLsqgRA3sWVMVJDnETPqJWHzfeV6+
qNVGpKvEPBV3YVUMdh7oNQZf5eBE6lZ+B1j4f4QB05LDxAbG0uiZAzyhrhPumQNHPfKIf8Q3mSK0B7Fk4i
VqoQCNvGqJwKkRcGqMq5zt0YzUUvDMuUgCsmHKG4vASqb+v+Z9qqAYrvQxv1ad/fXZtBD9mf4zAOCQFjZeZnQKb2
zq1Hx4jpftd4hO5XaX0tANDjRuBoL/h23vJH+frJhO2nSdQrJMpUYk5B2PMkYzZcVqNZFdxGNu3TLSIo2MhgPDN7T
PvDdbAJBvbE43vf3KdO0rT3Ar7yKIEgs7pYXNfnxXRYtE1jg6iLNWfF5JTgAAAAAAAAAAAAAAAAAAAAAAAAAA
2YmHdkqCgkk5WtoxOmcxLOoR7jaLX4o2Ex+M1JkYUWz4MxTs7TI6pwk0kC7QbWEUuHsPKTIf1dGt7xHw7MmDG
p9fnX2NVXzG5Lnn3WsXM/yLbBVH8NtIH6Wp+3U1B0d19Zc6GV3KXsS/X2hbWVbxUXMLy2oKnRYKHwsv8G0TlBE

TQkJCVKNGDauHwqWmpiotLa3ED4UDUHqY8QUAAAAAgB1iYmI0aNAgNWvWTBcuXFBCQoISExO1adMmeXp6asKEC
Yq0jlb9+vXl4eGhp556SqGhoTYXtgdQdkh8AQAAAABgh7NnzyoyMlJnzpyRp6en0nfurE2bNql///6S/ljr2sX
FRcOGDZPJZFJ4eLgWLlxYwVEDVROJLwAAAAAA7PDuu+8We9zV1VVxcXGKi4srp4gA2MIaXwAAAAAAAHBKJL4AA
AAAAADglEh8AQAAAAAAwCmR+AIAAAAAAAIBTIvEFAAAAlLFt27ZpyJAh8vX1lcFg0Lp164qtn5iYKIPBUGhLTO8
vn4ABAHASJL4AAACAMnbx4kUFBgba/YS31NRUnTlzxrJ5eXmVUYQAADin6hUdAAAAA0DsBg0apEGDBtndzsvLS
3Xr1i39gAAAqCKY8QUAAABUUl26dFHjxo3Vv39/7dixo6LDAQDA4TDjCwAAAKhkGjdurMWLF+vWW2+VyWTSO++
8oz59+ujbb79VcHBwkW1MJpNMJpNlPzs7u7zCBQCg0iLxBQAAAFQy7dq1U7t27Sz7YWFhOnr0qObOnasPPvigy
DaxsbGaMWNGeYUIAIBD4FZHAAAAwAHcdtttOnLkiM3jMTExysrKsmynTp0qx+gAAKicSHwBAOAAYmNj1bVrV7m
7u8vLy0sRERFKKTU0ttk18fLwMBoPV5urqWk4RAYyhtycnJaty4sc3jRqNRHh4eVhsAAFUdtzoCAOAAtm7dqqioK
HXt2lW///67Xnzrfl1xnzxRQ0YMEA//PCDateubb0dh4eHVYLMYDCUR7gArpGTk2M1W+W//fr11laxZM8W+v48
nn3/W+++/LOmaN2+eWrZsqQ4d0ig3N1fvvP76a3355ZcVdQkAADgkEl8AADiAjRs3Wu3Hx8fLy8tLe/fuVa9evWy
2MBQDgkEl8AADiGrKwsSVL9+vWLrZeTk6Pmz5urm+2pzsfwcHBeu2119ShQ4eqChUAAKfGrY4AADig2m0UAsjcAAB
wMfn6+pkyZOh49eqhjx442v9eO/2PZYtfn1Lo2Vr4fJ4i2UQ+Z2P8dhhpy7cXoMaUWdIJjWKMYH2oZMiYF6TS1
PSdfa6CvjEdH9q/7RwkErysr2U7hjdCNrKHUHmmcqJCnB3pK3ebxfc3rRNtY4Rd3gxMXzVvSbsU9xSLuXG0Ns+m
UsX1Ca6IHaRf09pNsBFrHgzawAjYBABeLDbgDQjUl2TlXhdWBHo60yW4kXWIN70LKiUcrZcAAv0uJKTWEATZz2
nTDzd5SRPyy3SNrJuuvbGNYJB+P7RxE99MZ3xmKx5kZMFDMTImjqRqUDf5hnDTYbqN5Ah6CrBbnRCwsNrRhdzzk
Fvii/rTvc7H1z4u0i4q4mLp4g9E2uuMxyqAvLtFBg51v9YXXqBb4UzRrc1y2PbpAz0RmLb7xRuhpHKcWLTJNFdHz
Pk+JE7JNjkqWbBIA3+4HJ4JKHd/NBh5Ba+GxuHBK8EAZQyJlZbsu6MZ3JRyt/6EC1+Zc3A1vaeg2zsHzt2UqOvHj
v4Ur4rmJc1+91S92xjlJdiHr7bt0sgV5wGaNgxj7hWS1PNl2AnggLYp46IqpqqcKmu5yuwbHp44zS7EsVphR6c4/
FEKa0PR95Cf2r3x3wF2+G4C8tgx5vfCsoOQfEmnAdmlB3vQ8ytPPptmzPv3avtSmAlWtiOLe9imzTlkE0u6QW5Rw
eOcxRQh8ss+y8ed8kpI6JqoegYNrZiZNAXHAxbtRkk4RQl4+3fWqK/XPDbC5YZenF7wVOIcFafSqmUoyW0aAfNZjX
g65mdu5+vzu3ewOs4AQ2yvrrtVyoaWsV7UNYHZmVdSfv4nCCRpIu+mYEL8AKNHMhv/mpL5LJNUp5FNT9AeOcUJXk
DQcXz+JeYFwrtbiT8xlJH6yT+I3JT4FZ9ZHelzdL3Kte7zr1V6+r7vv7/p5nLP/4sZxzJ8nhJvV2mNvBuvn4Ohq
+9WSbxpFFC8ld6lZwtWUQWSUQU1n+nlqumxb9r2KjF9xsrQcHcRePLLqvBkU5dOhc/PwjA2LCsanzzj8cMRYxbwU
R/ePv2aOcK6gfJnWgO79rP7DRt8Fu0RFFIOeDfgR1n5u+3V2x1e9/SGcmN6VqWGNfvdf1czQhZTY4DwNHyZ1Ia0J
wP8SYVDxYHgqBe9KzK3GYl0Yc5NgQH+2zRlM8U0Nk9pBPU8mGnUxUafqNOJAEPZXT1Y3iS6zADhL0DUf6M8ux5K
HbtHQAE/69OnBpLgkMl3ULS0yyy5y0S/0AZHAu2tfo0i8cHKHuzbH8HxNV3o84zL3mMqwZfS9dGmC4Iq2/mxqQmF
/lpU6kp2GNO6o88pfqkj3ZP6GBmm4dTcUeztGLCBmC8Qbc+d/PrJo1A/ZC/3rDPRZNL1Q3r0kqJMrUA0/CHmrVNQ
RrdXKXm4Qxd5mb8lJe4gv2TLPNCK/DLmQwNFtHa4NkpKOTM1UILGKGNk/I3X5Uo3lpvu8dyeKAKGu60J/Xwvc0Su
5GuDD1rlMAHz33aclmnBWnTgPDWhGeBpFw3ufg6u6QiVEP/HTzsWk5sX7GgAAw7Gkdz5pcUP1L2t5smj6YujaBcr
tKHvtcvpYL4u+Kk2IqRd1Rg6s6ClqCjh/lTGGupRjiEhVr5Uxp/w1pFqKFYMKHuaV8HGuD8jqgJvsE7qvy+PdICJ
oAF9tzv7SpGhaXzn4gdJmz4xlkb/0BTLAhIgU/dCFmkBi0LBiLv+l5sYcxLfVADHvbPHAjPCxA39p6fpxcLWzNXvBP
NuAVqI5+4m6/XtzGoF6uzx7+PMtf+0qgTB/u3gHDYNhcu2CH46zAe4jYmhP4sZRYHHjkDsH4Ty8Fc3IY5TkUq3wg
mhS9kwgoN2PdSuWqBKxmkdJIeSlN5zeASqC2UxKqUuVIOjP7pOrcEKzy2Ad+rkVYVIIRUV5IcKwhqQ+tkBAsSZ2M
+0Eq0rNGVaFDgxeMAVKJwBwFxXXvaqc0AO6VDJNhiQ3nzZvOcyGYKIIRpxc3oDZkE7s9HChqrlgEWYjVWxeRtG0V
ulvWqnnrIfRCnrZx2mJdzKLvWsXGL+Hj5aUxZ8t5X6dFOKGt4K40B4GgYKGm4FRUcb7VP/fWpaRsGEw2iUkvXxYt
/oHd3B6dXAwGoQW/c2ZYG9pI4W/tQ2m8ZpSSsfh7AT7PCzpBnaqMMmQEuKzu3ynmx6dxyXMvA2PJ5cDsSmrxgNtGO0
b2Wl9oXp37wOl0EACsAfq8jY6kBknQhYe9qu9m4KGrUVcDqGpsWVPLxRSlNY7UGmVQ/5Bs7nuGuo/4NQTJXJ/OVu
ckqq6vySk2A5hNvXq3A4M6SemmVI/2nCOoqNFlU0kfIcuJz4cjOttwH6NjGsvd+MQaNcfmg0j2Mp1YK+5xvKQdbJ
fyvpBH/Q1YqElZvzVs9QBlk/KjG0iBVOpxlbwoyOLHXgDbMgnezO/zpKuvTfKaPIupgh3EiuSUAY57LzrXDJLnHlbs
JD5DHhKW+9oOcNvQ2C3YwKW6eNBKAFCDNh3I1Xz0XxCoJ2NCemcqVL3Ep/h0k64QACDgk0ptXbwdg1rVY4t7+RYPF
G3pVY4UJ3mNTBS74b4vQrvW6lfbpVjSDSz8gLF1G/SVXdYwAwnUN5WT9FmmKAl8OSKAVBmEzN2GXmcQqz6sTlvVbg
tH0NIYpqOCVVuRJN3TN2p5XpHu1c0SJMtGtRzxYJWWqUvRm9RmuBIq3h/p8K6oFl8m8QwODZKgJcTsVb7MEYg+S
kUZOaGiq8g4jI3JwvVfx7P4F6H+3gMcDUkJkXEcb4ZV+EAVdJOemmqWJVk+h8CQ0Es/f+RS8s7hBJJ0U6rrXb1+zH
6rmbRXrlB+Vp+3UDy7vJ2XVnMVNdYUfQdlWq2d5Q+4VFfmjq7LAmDcMdPYyvA8sOLIU4uwPaE1a3QN8LJ7nJdqkEl
1RHXB5RGkCiX0u8wbMy3NoAHrJ65sxNUY4e95yeQHeqx6mVqYHpIoTWQKEOLHfjr+UWOa8QVmmXHTOzrfQPQRhVR
QVNJKPeqyJXoPBI0e3Obh5QAfGQmw8/n2O2KPHUkuCh/XGyRDGxVsRR7SlFx2LDhnqrbrAVk/T0lknWKkGLmM0i5
AgWHQ0Nb9vO9AvFwOJRT81z9rcmBHA8l5dq7jxzyWnVmIeF2bcW8rhqVeyTgQ+/V7aLF7wYjAkO/g9VcXuPJv9T6
rpcWmCL8cpHFswEeyiS+6wGtj/Lz4lsI/QB1OaWw2+TpFsK/Rdxwn5qYV4rBWW9XGvywJBTWuL5nJ8hIwV2GNhpM

```
2r8+PGSpMjISMXExFjqT548WRs3btSc0XN06NAhTZ8+XXv27LH5IR0AABSN92DAceXk5Cg50VnJycmSp0PHjys
50VlpaWmS/pitFRkZaan/+00P69ixY3r++ed16NAhLVy4UB999JGefvrpiggfwDViY2PVtWtXubu7y8vLSxERE
UpNTbWqk5ubq6ioKDVo0EB16tTRsGHDCs3kBFD27E58jRo1Sm+88YamTp2qLl26KDk5WRs3brQsvpmWlqYzZ85
Y6oeFhSkhIUFLly5VYGCgPv74Y61bt04d03YsvasAAKAK4D0YcFx79uxRUFCQgoKCJP2RyA4KCtLUqVMlSWf0n
LEkwSSpZcuW+vzzz7V582YFBgZqzpw5eueddxxQeH14h8Q0wtnXrVkVFRWnXrl3avvHmzr169qgEDBujixYuW0k8
//bQ+/fRTrV69Wlu3btXp06dl3333VWDUQNVkMJvN5oo04nqys7Pl6emprKwseXh4VHQ4QKVS2cdHZY8PqGiVf
YxU9viAilTZx0dl jw+oaKU5Rn755Rd5eXlp69atuv3225WVlaVGjRopISFBw4cPl/TH4ukBAQFKSkpS9+7dyzU
+wNnYMz7snvEFAAAAAAD+lJWVJUmqX7++pD+e7Hf16lX169fPUsff31/NmjVTUlJSkX2YTCZlZ2dbbQBuHokvA
AAAAABuUH5+vqZMmaIePXpXpYlhNIT09XzZo1VbduXau63t7eSk9PL7Kf2NhYeXp6WjaeygqUDhJfAAAAAADcoKi
oKB08eFCrVq26qX5iYmKUlZVl2U6ePFlKEQIo3VKEQJVW/WKDgAAAAAAEc0adIkffbZZ9q2bZuaNm1qKffx8dGVK1eUm
ZlpNesrIyNDPj4+RfZlNBplNBrL0mSgymHGFwAAAAAdjCbzZo0aZLWrl2rr7/+Wi1btrQ6HhISOohamjLLi2
WstTUVKWlpSk0NLS8wwWqNGZ8AQAAAABgh6ioKCUkJOiTTz6Ru7u7Zd0uT09Pubm5ydOuT09Pubm5ydPTUxMmTFB0dLTq168vD
w8PPfXUUwoNDS3RExOBlB4SXwAAAAAA2GHRokWSpD59+liVL1++XOPGjZMkzZs3T05OTho2bJhMJpPCw8O1cOH
Cco4UAIkvAAAAAADsYDabr1vH1dVV8XFxio+PL4eIABSHNa4AAAAAAAAAAABwSiS+AAAAAAAAA4JQc4lbHgmmk2dnZF
RwJUPkUjIuSTLeuCIxfoHiMYcBxMX4Bx8YYBhyXPePXIRJfFy5ckCT5+flVcCRA5XXhwgV5enpWdBiFMH6BkmH8
MA46L8Qs4NsYw4LhKMn4N5sqa3v6L/Px8nT59Wu7u7jIYDUdTpWXnZ0tPz8/nTp1Sh4eHhUdTpVnNpt14cIF+
fr6ysWl8t29zPitXBi/lQ9jGPZgDFcujFeh1iMQXKpfs7Gx5enoqKyuLAQ84GMYv4Ng
Yw4DjYvwCjo0x7LgqX1obAAAAAAAAKAUkvgAAAAAAAOCUSHzBbkajUUdOmTZPRaKzoUADYifELODbGMOC4GL+AY
2MMOy7W+AIAAAAAAIBTYsYXAAAAAAAAnBKJLwAAAAAADglE18AAAAAABwSiS+AAAAAAAA4JRIfKHEtm3bpqFH
DhsjX11cGg0Hr1q2r6JAA2IExDDguxi/g2Bjg0Ni/Do+El8osYsXLyowMFDx8fEVHQqAG8AYBhwX4xdwbIxhw
HExfh1f9YoOAI5jOKBBGjRoUeWHAeAGMYYBx8X4BRwbYxhwXIxfx8eMLwAAAAAAADglE18AAAAAAABwSiS+AAAAAAAA4JQc4lbHgmmk2dnZF
AKOjTEMOC7Gr+MzmM1mcKanq1s379+mrWFkgJBjDgONi/gGNjDDgu
AAAAA4JRIfKHEtm3bpqFHD
A4JRY4wsAAAAAAABOicQXAAAAAAAAnBKJLwAAAAAADglE18AAAAAABwSiS+AAAAAAAA4JRIfAEAAAAAAAMApk
fgCAAAAAACAUyLxBQAAAAAAKdE4gsAAAAAAABOicQXAAAAAAAAnBKJLwAAAAAADglE18AAAAAABwSv8f1zZ
uZ5yOGrkAAAAASUVORK5CYII=\n"
          },
          "metadata": {}
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "## Correlation Analysis"
      ],
      "metadata": {
        "id": "BIa7u3qLf1Wn"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "numerical_columns =
[\"CHILDREN\",\"Annual_income\",\"Employed_years\",\"Family_Members\",\"Age\"]\n",
        "corr = merged_data[numerical_columns].corr()\n",
        "sns.heatmap(corr,annot=True)"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 554
        },
        "id": "jTqBCboP4rMf",
        "outputId": "c171347b-91ce-410d-deff-0003843609ea"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "<Axes: >"
            ]
          },
          "metadata": {},
          "execution_count": 1323
        },
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 640x480 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAAAnUAAAIICAYAAAAISxhLAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAACW70l
EQVR4nOzddVhUaRsG8HvoRhpRERVsTFbFDlawdVWsRbG7E7tBV1272727VdOzCw0xAMUJEQgxQllZzvDz5HZxnGG
hg03r/v0te18857znn0+WbkmbeOSCwWiOFEREREgqai7ACIiIiI6OcxqSMiIiIqBJjUERERERUCTOqIiIiICgE
mdURERESFAJM6IiIiokKASRORERFRIcCkjoiIiKgQYFJHREREVAgwqSMiIiIqBJjUEREREclx+fJltGnTnTBlZWV
hCJRDh8+PBX97l48SJq1KgBTU1N2NraYuvWrXkeJ5M6IiIiIjmSk5NRtWpVrFq16pvh4aGolWrVmjSpAn8/Pw
watQo9OvXD6dPnZ4vRu/sHXp6o2Gl6KrqlJijmqNyXRzijk4r8uMBBmUJBWkA0XY8BbxYkO2VUZ7dGgkrEpDqV08dXF5Gpx9TqQm5odnh8oaG+IJd7Q8R0uAhd+lXpnVV3a/sQWMmdyXhKwQqxrBeA2iU9eJxoZhKNj1CQp2Ih+bXhxi+XEWt9z/YNw1kpG3rcGj8I2kQSYKnbnVpU2iQlA4RBi5UjFVfNgvhM55x4wrs3YxXmcVr/EwxmSdKgg1KQGzKtVqWzI9L8RGmxQMNmBqV7a0WrN/jmYx9TKsOnSvvMfnUbabxgsDu+E7Sz1Em5YYDl3jCiNMxI+YVO7N3/EdXQNvd7JOrqpM7+CZTlphtWbmtrhw4f2aCDEAUGnhodrBYmjQN29g2dqSoHIiTZAjhHYhZOm1Vu6qUcnl07S8RmM6eSDjAK7nTyYC2hMOPHxNDgWAt6FFvTuENTTgbvp7v7vFpldmvvvDJK2Vq6ZPpKGE0hn7A0UC44UxHRfxENNjVfQtNqU09uUEx6Uae48WozdrzY
K0Xpd9UAgXpDgqa7fyKl1ejh3HxsqmN9sQoiWYNYRAVO8DrBgJ0t5AZfqzW2vXXnDl4rwPzp/ia4jz8l9iDmilnX
ntSfLrbhEI9kz0Xv4dCppsP9OCnbGUXzDCZgIYSd9FihDOSOsNnSSEjxfpz8QdodKtXI2BIoxJGGUIpIBBYZULP1
Vl7PVSzUjr/JJp5qpqQCXUk/Q6OvjFgPXxBoXdH5CFl9KIQ5X/LG0FVzI2SRJDLb8w4oPQPlsSSaU5/zVPcbYK7h
vdlM5PaOmOsh2CS0jMClpKlhkn/dSW0zdtmFVM5D8U/2qIFh5mBIXDEdq4V7nkCd5BuFM6mm27n9TwTTOIjJy6jU
a/xSyrJnF7uSKVkZEl9bUhwOgVLhtjgl0kzeb5ipPShG0hHSq4gGhfgNbL98mh5v7fhPgaSuG5GBSc/0mfTj0ehR
Ja3IVDm7+4CrSUWDpnLS0Ou/p17hwq5BOhhq4uTDH9VV0Uf1/S7gqBMt6ayxy4Hb1xuYtqiCrfQlY7lbaF8qPSUg
XmnIV2LFRxpVSldN8cNwPzOamXZcJWxzpZvP6GaNYmfXHk2FbZeN3pfP6Zl0M5l6FzOGQcqgAphuMYmkpD/e8Jgh
d0ppr8Dl/qBqDZKSLcgnHHnqWCjvdSGK7FWJ0m0U5rwhxfnc/Z5Hb5mZH1ttqFs+1/2qZcGn7hzL5v2EgJ3h6Cf+
OQwhn0pM6yAjtDsNhF2NpDkZ+JZKDvBBxrh8UYxcHBxP7u9lvHkE6KpMNYXlq0WR6hbgt6RBTKTS29NDY8pkpV+
```

YEDesL33lnExT5BXOwTXL18FC7OTSTvGxkVwdK/5+Dxo8tIjH+OkOe38feS2TAwOM/rS8l7WZmK2wqhXyKpCws
LQ82aNXH+/Hn89ddfePjwIby9vdGkSRMMHToOz8/fv39/REREIDg4GP/++y8qVqyIrl27YsCAATnqbtmyBRERE
bh79y7q1auHTp064eHDh1J1DAwMEBERIbW9ePFCqo6W1hYWLFiAd+/e5em1KcOHDx9RzrYOpowdouxQlKJaa0e
Om+qGO8sOYEkrD7wJeIEB2z2gZ2Igs75NjbL4c/kI3N57AYtbTsLDM3fRe/O4WJYtLqljYm2B4QdmITr4DVZ3m
41FLhNxdsVBZKSmSx3rxi4fzPhtoGQ75rkrT681P1RsXQe/T+2By8sOYkPrqYgKDEf3HZOgk8v9VNPWxLvwaJx
fsAeJObK/X9a1y+PO9nPYOn4Gdv7pBRV1VXTfMQnq2pp5eSn5zvvWQyza7Y2B7Rpjz6xBKFfCEoMXbcfbhCSZ9
U/eeIBl+89hUPsmOOQ5HDP7tMfp24+w/MA5SZ2dMwbCZ9l4ybZuQi8AwO+/VcqXa8pv48cNwbChfTBk2CTUrd8
GySkpOHl8JzQ1c/+svH4dgSlTPFGrTgvUdmyJCxev4eC/m1GxYlkAgJWVBaysLDBx4hxUrd4MffuNvNhrNzE2xYv
zi/LouU5JfofhOyZAhEIhFu374NXV1dSXmlSpXQpO+fPD+/jo4OLCOtAQDFixdHnTp1UL58efTpOweurq5wcnK
S1C1SpAgsLS1haWmJOXPmYNmyZbhw4QLs7eOldUQikeR4uXFycsLz58/h/h6emJhQsX5s2FKUkDx9/QwPE3ZYehN
I36tcLNPedxZ/8lAMCBKRtRsWl11HJtjPNrjuao36BPCzy55I8L67NbOryX7EO5Bvao38sZB6ZsAgCOHN8FgRf
8cNzrc5L2Njwqx7HSP6YiMSY+Ly5LaerOa4H7ey7Af/9lAMCJyZth27Qaqrk2wvU1x3LUj3gQgogHIQCAphO7y
jzm7l7S37mjY9dh7P21KGpfCuG3nyj4CpRnh/d1/NGoJto3rAEAmOreBpf9n+HwZV/ObdOwR32/oHBUsyuBlo5
VAADFzIzgUsdeqnXP2EBXap/NJ66ghLkxHMrb5N2FKNGI4fOw33MZjh07AwBw7zOSb175oV07Z+zbl/P7DADHT
5yVej1t+gIMHOCG2rVqICDgCDgGR4/fgrXLp8bDUJCXmDa9AXYvnU5VFVVkZkp4FYqdr/KVehb6uLi4uDt7Y2hQ4d
KJXSfFClSJP+DAtCrVy8YHRnJ7IYFgIyMDGzalPOHVONDD47uPr6qqivnz52PFihV49Up+dwgJh6q6KopXLoVn1
z633orFYjy79hA2NcrK3Memuh2Crkm39j657C+pLxKJUKFJdcSERmDAdg/MursOIw/PReXmDjmOVaNdfcz2XY/
xp/9Cqwldoa71/Z/NgkRFXRVF7Ush9Oqjz4ViMUKvPkLxGnYKO4+mvg4A4MN72S1YQpSekYHAsAjUqfS5619FR
QV1KpXBg+ey/82pZmeNwLAISRL3Kjo/2foUFV2fc6PSMDJ64/QPuG1QvFOJH/KlXKGkWLWsDn/FVJWUJCIm7
fvo86tWt+OzFUVFTg6toWuro6uHnrXq71DA30kZCQhK2Zbo+M7mPvlkRmfX1TQOBAHqmBtDSOObTwW1xavE+HPfah
fKNqsJ97Ris6TYHwbcCAQC+R67h3esYJES9Q9Hy1mg9qTvMSlth66AleXCl+UPHSB8qaqpI+s/9SY5NgGkZK8W
cRCRC8xluCL/zFDHPCs8PrHeJKcjMyoKJofSPZRNDXYRGxMjcp6VjFbxLTIH7vE0AxMjIzELnJr+hX5tGMuufv
/cEiSkf0bZ+dUWHXyBYWpgDAKKipO9XVHQsLC3N5e5buXJ5XL18FFpamkhKSkanzv0QQGBgks66JiRGmTB6FjZt
2KiZwKrAKfVInFou/ue748ePh7u4XF1E4vFOX59/v3333ByckJISYWKqOv
r4+fH19pcq0tbVlnmPBggVVo2rQpxo0b99V4V4UlNTkZqaKlWmkpoqd2wHCZ9IlN1g//jsPVzedBIA8CbgBWxqlIV
jDydJUndzt49kn4inL5EQ/R5Ddk+DibWFzK5aytZijjvMyxbH1k6zlR2K0t0JDMWm45cxpWdr2JcpjvCot1i48
xTWHbmIge0a56h/6PI91KtiC3Mj2eMbhaZbtw5Ys2qB5HXbdj1/+FhPnwaj5m/NYWigj04ZQxbmM2kS8MnXx7GYmprC1tZWquy/CZWiZGZmIigoCL/9JjO2zNLSE
ra2trC1tcWWLVvQsmVLBAQEwNz882FRWVHHHmpmHDhnB2doaHh0eOhPW/PDO9MWvWLKmyqeNHYPqEk92UZT
nkt8lIDMjU9LK9om+mSESY97L3Ccx5r3s3s+v9vnUp+l4DM9AxEBkm3IkUHv0Eph3K5xhLu9xwAYGoj3KQu5VOis
jIyofef+6NraoAkBYwddJndC3bNqm076xwkRsb99PEKEiN9HaiqqOBtfLJU+dv4ZJgayp5lueqgD1rXrYo/Gmd
3LdqVsMCH1HTM2XoU/dsOhIrK5xFBb2Lf49bjECwZIXvcohAdO3YGt2/fl7zW1MwevmBhYYbIyGhJuYW5Kfz8H
8s9Vnp6OoKDwwAA9+1g9Dhk6U1NHTO8XJ4zuRmJiMjp37ISMjQ4FXoySFtNtUUQr9mDpjY2M4Ozt
j1apVSE5OzvH+eyFhclbv7wlLIdJnSb3ONG3KKHxgKzeLMiLTrTQLGm2iFtL+ejRkjRNWbvtRqh19PBm/c6JeXLx0LWvCZO2V
ai1XFkgCAh0j3P3IpBUJWeiYiHobCpt4XMytFIpSqVxmvfGV3ZXOrl9m9UM7ZAf9Om4f3L2V3RwqZupoaKtgUx
a2AEElZVlYWbgWEoIptcZn7fExNz9E7oaqS/fq/fSpHrvjC2EAXDarKHisqRElJyQgODpNsAQHPEBERha1ZN6kv
q6OvroVat6nLHx8mioqIiSRI/Hcf75G6kpaWh/R/uOXphqHAq9C11ALBq1SrUq1cPtWrVanuzZZs1GlShVkZGTg7
NmzWLNmDQIDA3/q+B8+fICfn59Umb6ERo5AnJKSgsjISISGgjPRoO
ahQ4dOmDChAkoVix7zJRYLEZkZGSOuubm5iHhobCpt4XMytFIpSqVxmvfGV3ZXOrl9m9UM7ZAf9Om4f3L2V3RwqZupoaKtgUx
a2AEElZVlYWbgWEoIptcZn7fExNz9E7oaqS/fq/fSpHrvjC2EAXDarKHisqRElJyQgODpNsAQHPEBERhaZN6kv
q6OvroVat6nLHx8mioqIiSRI/Hcf75G6kpaWh/R/uOXphqHAq9C11ALBq1SrUq1cPtWrVwuzZs1GlShVkZGTg7
NmzWLNmDQIDA3/q+B8+fICfn59Umb6+PsqUyR5AnJKSgsjISISGRkZODVq1c4dOgQ/v77wwePBhRo0
ahQ4dOmDChAkoVix7zJRYLEZkZGSOuubm5lK/dD+xt7dHjx49gcHTxmj3vUxwVUUO+ISsFxUATGx9y8ZsRgbpXcUcxDOWoV5A+DPDN9U7mUmovmyeRwdn23BNzVV2fCsc3Gc2S3sJL+gPIOU/ooExkvIKoD4oK/lMRIXJi2KdDDJvl4mnoQAefDQEIb1QgW2VRyO0ZyB0mKnE9q/C

6+M+ePUP16tIDeZs1a4Zz57LXXtqwYQM2bNgADQ0NmJiYoGbNmti7dy86d0jw1WO7uLigVKlSmDdvHlavXg0AS
EhIQNGiORcyjYiIyHWpk9mzZ2Pv3r3fe2kF0qMnQegz/HMXw8IV2clJuxZOmDd1rLLCyjd+x29Az9gALqM7w8C
sCF4HvsD6Xl6Swf5GxUylxpKG+T7DPyNXoMXYLmg1vitiwiKxZcAiRH4xaP/h6Ts4MGUjmg1phw4z3REd8gZbB
y9B6N2nAIDM9AyUrV8ZDfu0gIaOJt6/eYsHp27h7MpD+XvxeSDg+E3omOijOZhOODMzRFTAC+zquQDJsQkAAAM
rE4izPt9PfQsjDDg1X/K67sDWqDuwNcJuBGBH1+xWdQe37CSm175pUuc6MnYdHhzImzG6yuBS2x7vElKw+uB5x
MYnoZy1JVaPc4OJYfZErsi4eKiofG6Z69+2EUQQYdW/Poh+lwAjfVOOql4Owzo2kzruzcchiHgbL1kqpTD7a9F
q6OrqYO3qhShSxADXrt1BqzZ/SrWslS5dEqamn4cCmZmZYsvmZSha1Bzx8Yl4+DAQLVt1xzmfKwCAGtXtUbt29
r179kR6cfYydrXx4oVwJ+yIxQKfvZvHROLvmUlAv5zO2JCvVyKZJjpMVnYIgmYkLtwtNHlp/N62yg5BsPQajlF
2CIKWkfY6T4//4eJmhR1Lu3Her1Ob3wr9mDoiIiKiX8EvOf1KREREhQCXNJGLSRORERERJAydKyMXuVyIiIqJCg
C11REREJAzsfpWLSRORERERJA7tf5WL3KxEREVEhwJY6IiIiEgZ2v8rFpI6IiIiEgd2vcrH7lYiIiKgQYFJHRER
EwpCVpbjtB6xatQo2NjbQ0tJC7dq1cfv2bbn1ly5dinLlykFbWxslSpTA6NGj8fHjxx8697dg9ysREREJgxLH1
O3duxdjxozB2rVrUbt2bSxduhTOzs54+vQpzM3Nc9TftWsXXJk2ahM2bN6Nu3bp49uwZ3N3dIRKJsGTJkjyJkS1
1REREJAxKbKlbsmQJ+vfvj969e6NixYpYu3YtdHROsHnzZpn1r1+/jnr16+bNOa1bt6+27vOMJ
nVERETOyOlNTUVCQoLUlpqaKrNuWloa7t27BycnJOmZiooKnJyccOPGDZn71K1bF/fu3ZMkcSEhITh58iRatmy
p+Iv5FFOeHZmIiIhIkcRZCts8PT1haGgotXl6eso8bWxsLDIzM2FhYSFVbmFhgcjISJn7d0O/eHbNnzOb9+vWhr
q6OMmXKoHHjxpg8ebLCb8snTOqIiIhIGBTY/erh4YH4+HPDQ2GhXrx4EfPnz8fq1avh6+uLgwcP4sSJE5g
zZ47CzvFFfnChBREREvxxNTU1oamp+U11TU1OoqqoiKipKqwqKgqWlpYy95k2bRrc3NzQr18/AIC9vT2Sk5MxY
MAATJkyBSoqim9XY0sdERERCYMCu1+/h4aGBmrWWrAkfHx9JWZnx8fOdo6Chzn5SUlByJm6qqavZliMXfeeH
fhi11REREJAxKfKLEmDFj0KtXLzg4OKBWrVpYunQpO0TObt3bwBAz549UaxYMcm4vDZt2mDJkiWoXr06ateuj
efPn2PatGloO6aNJLlTNCZ1REREREF/RpUsXxMTEYPr06YiMjES1atXg7e0tmTwRHh4u1TI3depUiEQiTJ06Fa9
fv4aZmRnatGmDefPm5VmMInFetQFSoZAeG6LsEARrokPezXD6FRiJ8+aX7K9g/N62yg5BsPQajlF2CIKWkfY6T
4//Yd9shR1L23W6wo5VULCljoiIiISB7VBycaIEERERUSHAljoiIiISBiVOlBACJnVEREQkDEzq5GJSRORERML
wnevL/Wo4po6IiIioEGBLHREREQkDu1/lYlJHREREwsAlTeRi9ysRERFRIcCWOiIiIhIGdr/KxaSO5OKjrn7cg
rvzlR2CoM1xmKbsEARrXNeDyg5BsBxM7ZQdAsnDpE4udr8SERERFQJsqSMiIiIhIGBqzz
sfiUiIiIqBNhSRORERMLAiRJyMakjIiIiYeCYO0rmY1BEREZEwcEydXBxTRORERFQIsKWOiIiIhIFj6uRiUkdER
ETCwKROLna/EhERERUCbKkjiIiIYRBzooQ8TOrIiIiIhIGDzQ8TOqIiIhIGNj9Khe7X4mIiIgKAbbUERERERkTBwnTq5mNQRERGRMPC
JEnKx+5WIiIioEGBLHREREQkDu1/lYlJHREREwsAlTeRi9ysRERFRIcCWOiIiIhIGdr/KxaSO5OKjrn7Fe5mNQRERGRMLClTi6OqSMiIiIqBNhSRORERMLA2a9yM
akjIiIiYWD3q1zsfiUiIiqBNhSRORERMLA2a9yMakjIiIiYWD3q1zsfiUiIiqBNhSRORERMLA2a9yMakjIiI
iYWD3q1zsfiUiIiqBH75pG7r1q0oUqTqTIN9WdOXMmqlWrlqfxEBERkWzirCyFbYnna/3rhxA/Xr14eLiwn
DiRl6fKF+PGjcPw4cOVHUahUc+tOZoMCQGzQCMzJZdn5bb76g3sBeZh2i97ZIqfHjX8yZUJ5zGz4BnFeT+Zzlcn
zMlZoPak7ytSuCBU1FUQFvcbWwUvw/s1bAMCQPdNhW6ei1D7Xd57FgSmbFH59BdFdv4fYsusen
IZmDesqO6wCp5bb76g3sBXOzAwRFRiOEzO24bV/iMy6ZnbFOHRMJ1jZl4JRcTOcmr0dN3WEpl4JRcTOcmrODNzZ753PE+aehW3M0G9g
GBmZF8DrwBfbP2IIXcr631VvWQauxrjApboaYOEgc9tqJgP98bz/pOq8f6vf4HQdmb8PFzSel3qvUpDpajOwIq
/IlkZGahqBbgdgwYJEiL01p+o/vjXbdWOPPQA8P7z7Cwk1L8DLOda71q9Wugj+HdEU5+7IwszTFhD5Tcdn7qlS
dfmPd4dSuKSysy3zJCeloGnD59hrddGPL4fmNeXk7fY/SpXnrbUbdq0CcOHD8fly5fx5fx5s2bvDxVvtDTO4OJiYmyw
ygUqrV2RLupbji97ACWtPLAm4AXGLDdA3omBjLr29Qoiz+Xj8DtvF73Xj4Nl2eKS0ibWFhh+YBa
ig99gdbfZWOQyEWdXHERGarrUsW7s8sGM3wZKtm0eu/L0WguSDx8+opxtaUOTZoRRYlVvXgcvUHri47CDWt
pqKyIBw9Nw+Cbq5fDbVtTXxLjwaZxfsQWLOu3yONn/Va02IDlN74tSyf7Gg1SS8DniBodsn5/q9LVWjLNyXj8C
NvRfg1XIS41lG0bIkcdas4/wab6nZ4HxmX471qLrXQ8+9huLn/IrxaTMCSjtNx98jVHPWEyG1oN7j26
YgFk5agX+vB+JDyAUt3/QUNTY1c99HW0ULzw2Asmrw0w2Asmrw01zrhIS+xeMoy9GjaBPbD0fEy0gs2/0Xihgb5sFV5KM
sseK2QijPkrqkpCTs3bsXgwcPRqtWrSTrN1WpUgXz5s3D5cuXAY/Pv/gzUfPZTs0fH68tv
vtqqdJ8+fP9p4/PDqeq/N73JLP3H3t7SHAejNjXt14w6XfVpQs3RX1ERKjd4ZuvehbIAgiBQhI/W2Gk98/cfoTRlu3qjrC8UXZoHNT9Ahj6bDdR/kx8G1LWHGOZWTZ8z8PKL
jl2L8SJuVVzhIxwl6Zm9aXb6q3F/yHnrcQZpRxC2N2eB3yVIghj3h5T0D5jgLvb4M7Rq6dQkbfUO6dw7YNtRoZrgNmKBuhQqfYLaBiJaBwdwNtw4XhAV7y0nSkQoCWL3A12iwhO7nOqBSk3g
xgaGqJr165ITEyU1LGxscHUpUulzlutWjXMnDVmNnDlT8nrJkiWwt7eHrq4uSpQogSFDhiCd5bonUHtJisOhnzpnFyEnvNFSDSNFN4K0tA9LBlHVMz8DPyPxQ06n5z3t98VBiYmzoyb4WlvkQWWwe8RePAQCIL3+
9Fi1ahKJFi8LExARDhw5FeipIPZCBnfpCS9fjQ7W3x2BhMFs3ftV0SoDWBxDJTLOSbK9J+KIdz3+c9z+ve5Zp4h0UUUq47+XMbHs4hNcxvEdt84LYz+ncWQEkfqJ/UwGIO
ZL3Gzduj0HDh2PUqFEwMjKChYUFNmzYg0TkZPTu3Rv6+vmVnIoR2NSUIWCPBZjH+QtRdbF+yOGJ9Q+bCgVpWYZ5qnnb9a2DsTWRj1lgtZ0Q2Zq
/dN15oVG/Vri55zzu7L+Eq0evcWDKRqR/SEMt18Yy6zfo0wJPLvnjwvrjiA5+A+8l+/D6cSjq93KW1Gk5vgsCL
/jhuNcuvH4chrfhUXh87h6S3iZIHSv9YyoSY+IlW2rSh7y81AKlgeNvGDGgF5wa1VN2CFUfTBbzpA7VW3X4tcG/PBdzffxk
xz1/j2JTNSP+QihqujWTWf/MgBGc8d+PRsZvISMuQWaewaNqvFa7v8cHN/RcgF+tw19kzZiLQPaXB0bSKzfuM+L
RB4yQ8+648hKvg1TizZh5ePQ9Hoi+8tABhaGKHzzN7YOnIFMj0j5LEpBGKqgo6H4fn/4ZOr0c4gOjUDk89e4f+J
mnl1nfurSrxO2LNuBK6ev4gCGaN8ISphSkautTPdZ8bF25j3cJNu0Sde2J75pAP7ly5hzfhEQh9FoalM1dBz

0APthXL5MVlUAGRZ0ndvn37UL58eZQrVw5//vknNm/eDLFY0j0eMmUKFi9ejLt370JNTQ19+vSRej840BiHDx/
G8ePHcfz4cVy6dAleXl7fFYeKigqWL1+0x48fY9u2bTh//jwmTJjw09f3yYULFxAcHIwLFy5g27Zt2Lp1q1QC2
7NnT+zevRvLly9HYGAg1q1bBz09PQDA69ev0bJlS/z222/w9/fHmjVrsGnTJsyd01fqHNu2bY0pqSlu376N4c0
HY/DgwejcuTPq1q0LX19fNG/eHG5ubkhJSQEAvH//Hk2bNkX16tVx9+5deHt7IyoqCq6urgq77p+hqq6K4pVL4
dm1h5IysViMZ9cewqZGWZn72FS3Q9AX9QHgyWV/SX2RSIQKTaojJjQCA7Z7YNbddRh5eC4qN3fIcawa7epjtu9
6jD/9F1pN6Ap1rdx/Ed0vRVVdFUUrl0LwtUeSMrFYj0Brj1C8hp0SI1M+VXVVlKhcGk//8719eu0hSuVyb0pVL
4snX9xLAAj84nsLZH93e/49DD7rjyEy6FW0Y5SoXApGRU0gFosx8YQX5t1ei8FbJ8ls7RMaK+uiMLUwwZ0r9yR
lyYnJeHw/APY1K8rZ8/uoqauh/Z9tkBifhKCA3LvKBUGcpbjtB6xatQo2NjbQ0tJC7dq1cfv2bbn1379/j6FDh
6Jo0aLQ1NRE2bJlcfLkSbn7/Iw8G103adMm/PnnnwAAFxcXxMfH49KlS2jcuLGkzrx589CoUfav30mTJqFVq1b
4+PEjtLS0AABZWVnYunUr9PX1AQBubm7w8fHBvHnzvjm0UaNGSf7bxsYGc+f0xaBBg7B69eqfvMJsRkZGWLlyJ
VRVVVG+fHm0atUKPj4+6N+/P549e4Z9+/bh7NmzcHJyAgCULl1asu/q1atRokQkQJrFy5EiKRC0XLl8ebN28wceJ
ETJ8+HSoq2Tl31apVMXXqVACAh4cHvLy8YGpqiv79+wMApk+fjjVr1uDBgweoU6c0Vq5cierVq2P+/PmSc23ev
BklSpTAs2fPULas7MQpNTUVqampUmUZ4kyoiVQVcq8+0TUygKqaKhJj46XKE2PiYV6mmMx99M2KyKyvb5rdlaB
nagAtPW00HdwWpxbvw3GvXSjfqCrc147Bmm5zEHwrexyJ75FrePc6BglR71C0vDVaT+oOs9JW2DpoiUKvkYRJx
0gfqmqqSP7PZy05JgFmZayUFFXBoJfL9zYhJh4WudwbA7MiSIx9L1WWGBMPA9PPXYC/D26HrIxMXNxyCrKYWls
AAFq07ISDc7fj7asYN0vfGiP3TMfsJq0QEp/8E1elXCbmxgCAuBjpLue4mHeS935GPSdHzFkzHVramoiNeosRX
cciPi7+6zsWZErsNt27dy/GjBmDtWvXonbt2li6dG33//iidTmJub56iflpaG33//Hebm5jhw4ACKFSuGFy9
efPPkzB+RJy11T58+xe3bt9GtWzAgJqaGrp06YPPkzAkO3aJFAQDR0dGSMhbn3MkY/Pz+oqqpKEtf/CgwMhKOjIOQikaSsXrl6SEpKwqt
Xn3+xfnmfVFVVYWJiAnt7eOmZhUX2P3qfzuvX5YkXxqBakSj7o
/z47D1c3nQSbwJe4Pyaowjw8YVjDydJvZu7ffD08gNEPH0J3yPXsGvMalRxqQWT///hIKL8U6JyKTTu3QL/jFu
Ta51P/z6eXnUIft638fJRKP4ZvwZiMVC91WN+haoQzh2ccD7olGRTU8vbpWLvXbuPnr/3Q/+2w3Dz4m3MWzcTR
iZF8vSchdmSJUvQv39/907dGxUrVsTatWuho60DzZs3y6y/efNmFmfDhw6hXrx5sbGzQqFEjVK1aNc9izJN
P1KZNm5CRkQQErq8+/3sRiMTQ1NbFy5UpJmbq6uuS/P31xs76YZvvzl+5/qfPm+iopKji7dL8ezhYWFoXXr1hg8e
DDmzZsHY2NjXL16FX379kVaWhp0dHR+8krlx6itrf3Tx8/tHLPLuaYMFCxbk0Nan5FkWDw8PjBkzRqp
sqn3fH447N8nvEpCZkSlpZftE38wQiTHvZe6TGPNedv6TGPNedv3/txokv0tAZnpGju6bAdZA3KOVQLtdYwv2eAwBMbSzwN
jzqey+FCpmUd4nIzMiE7n8+a7pmBkiMEXgLx09KyuV7a2BmiIRcvrcJMe+hb1pEqkzkzfzBAJ///elqlVAXomBph
9fZXkfVU1VfwxxQ1N+rTAjPrDEf//Y0d88d30SMvA25dRMLYS1sS1K2euSc0+VdfI/nfc2MwYb6M/t9YZmxkh6
PHznz7fxw8f8SrsNV6FvcZj3wDsv/oP2nRriQ1NTMUTctLQ337t2Dh4eHpExFRQV
0Tk64ce0GzOMfPXoUjo60GDp0KI4cOQIzMzN0794dEydOlGoMUiSFt9RlZGRg3+/hsWpKxjnJCAEjGIKwqnqp+w6XLl2S+X6FCBdy4cUM
qMb127Rr09fVRvHhxmft8ixoauDx48ewsbGBra2t1Karq5vrfpMmTYKBgYHUdj4s98T3axL9Pv2qMz9/Uy0BTIOAE/AxTL6vdLDC
hHs6lZGmO8zmfuE3Q+Sqg8AZetXkdTPTM9E+IMQmJeW7gYyK2WJd69znyBiVbEkACAh+v2PXAoVMpnpmYh4FIr
SdStJykQiEUrXrYxXvkFKjEz5MtMz8fJRCMrV/dxLIBKJULZuZuZYTmcm9C7z9Duf98b8vXt5d8b+8cvAxPlwnwa
jlRsr2PjMO59Uexqmf28JGXDOOQnpoGiy++2ypqqjAuZoY4Od/tgigl+YMkyXoV9hqhz8IQG/UWv9WvIamjo6e
DStUr4uG9AIWfX6QikjurVhAUOPtVVu+Up6enzNPGxsYiMzNT57EtGnTsHjx4x7Et6jP2iYWFBSIjI2XuExISgaggMHDiAzMxMnT57Et
GnTsHjx4hzj5hVJ4Und8ePH8e7d0/Tt2xeVK1eW2jp27Jiij/ZnNG3aFDt27MCVK1fw8OFD9OrVSyr7tbW1hW7duhg8eDDtGsjaxHz+xpfu1rXIMLdPhZ7GCHVv4A8/z9RMzDE9S0r5qbLz9ejmKnEG+iHX2iYNbhHG9LH0Jea3WHszcTlNvzIu4I3jsicuj+BCXhrsdBHt8xZInjkWA4XbRc37eV72/C1zW
60lasWIGQkBDs2LEDa9euVdj5v8bGxga9evtVCnz59cPjwYYSGhuLixYsSvNzt1hOvhx5DuJlmX9nPQJfGWfwP4f7OOVXkKmVPjPVbJFHp9+xtrGnKnJuwI4ysFwSrnf5yPJLGn9YZc
SOYMWMGxowZIxlP9yOGDh2KuLg4dOvWDXzu3EFwcDBOnz6N3r17SyV3
hoaOJ2/uzE+Bui4eg1YSukvpXNp9C+UZV0ahfK5iXsYLzqE4oYV8aV7edltS5uP4YqrV2RJ2uTWFa0gL1ezqjY
r0auLbjDIDsJU9+H/4HilfOXk+sklNNdF8yFMG3AhDxJDx/b4CSpKR8wJNnwXjyLLsb/vWbKDx5FoyIyO8b1lC
YXd94CjW7NUG1jg1gWsYKref1hoa0Jnz//9n8Y/EgOE3oIqmvqq4Ky4olYVmxJFTV1aBvYQTL
3gCdbs1Re2ODWFRphi6zOsHTR1N3Nx/EQDgtngo2k7oJql/cfMpVGxUFU37tYZFGSu0HNUJ1vZlcOn/39vk9Om
IePZSasvMyEBCTDyiQ7J/sH9M+oCr08+h5ejOK/gCsxLFOXXuf0AAL6FYAbs3oOH4D7SDQ2a1OWZ8qUwY/lkx
EbFSq07t2LvYnTq3UHyWltHG3aW6bGFXyRYAYFXCEnaVbGFRLHtcl5a2FgZN6odKjxMSrCspgFytmXxZQlE2BmaQa
fYxfz8/IKNA8PD8THx0ttX7bE/aysrCyMHz8eFhYW2MTHQHVbtwgsvZ6BI+NEf9HdN17eMp3cKy/GOkdQu6XoRGJVJNsPZ9X9rhkVQU1cRPLc9tcRAVX6GI5RCHSxwRtbNCzZz0LuCfbKl/b7vSXvGtB8btCJjTcEBDn6GXCU2jDBMcHBOWYBZmHUJNQnGycxGMHMyDRRPAOL4kGtBGvgp9kNDpbSAoYmkGYGA6ZHnjXGmKodq9Awr1vaHzsxmxhENgpqSNmkuySSHaqdzdlh69pN4gxsWQ6gqkEBJJJGEBUYEyJqABBPoTgNQBfbHUbkw3l6FHsDzhrALxG69LpvuMNjG4LSzGtIpLq3dm6Nlkw45QIyKpMc3qz1xvKpd2gbWmKr4yr9QWtzm9OMycxtLSjqjjSLRJVWjrAlC4XvmZJUslZUDGg7pC8vyrEVBrvZx7XqZXbs/4FsGwOnpH6HzcN+Rw3yedVbXi5HQuBwIKv2FUMbpvYH0HuUlQmzXcvdOgfbi5unBkfYIYfeHkx8tIeQWELtvSgiOSHA5aIUtrGllPyd7DOFsjPo5gyfEGMr+EK4TvTf+9QJAnCLfPqTnWmZP5IS58Hm+PaLe4I4hFq8Ya3PGbx4MGeQrvEHc40pYEAqrs1yRP7CYG3gCuzBxpIlVtgBnZ8mIpzOyEYu5A6ugnAD2LPnL8+Y5F8ZI3sCGdOCQcWOIGoTWdeHd9VwLOY3UP4sPcgp7NwIiUI3XJ89rSyM8oDuxYYzwkI4CJZQJiqH3HMkx8L+Ajg3j1mKr

yD0/fwYEpG9FsSDt0m0m06JA32Dp4CULvZi+Vk5megbL1K6NhnxbQONHE+zdv8eDULZxdeSh/L16JHj0JQp/hE
yWvF65YDwBo18IJ86aOVVZYBcqj4zehY6yPpqM7Qc/MEJGBL7Cj1wIkx2YvjWNYzETqs6lvYYQhJz9PSqo/sDX
qD2yN0JsB2NL12yd1CYHv/7+3rUa7Qt+sCF4HhmFVL0/JMAjjYiYQfzGrMNT3GbaOXIHWY7ugzf+/t+sH/IWIZ
y+/67yH5v+DrIxM9FwyFOpaGnjh9xzLu8/BhwThTpL4ZMeq3dDSOcKkheOgZ6CHB3ceYlSPCUhLTZPUKW5TTGp
9uQpVy2H1v0slr0fNGgYAOLHXG3NGeyErKws2ttZo2dkZRYwNEf8uAYH+TzCow3CEPgvLr0vLGwp8EkRuXa2ym
JqaQlVVFVFR0sN0oqKiYGlpKXOfokWLQl1dXaqxqUKFCoiMjERaWhoONBTfaioS/3dQGtEXxth0/XolkmnB3fl
fr0S5muMwTdkhCFYcCvd6eXnpbhrH1v6Mm28u5unxE4eOUNix9FfLnnGdm9q1a6NWrVpYsWIFgOyWOGtrawwbN
gyTJk3KUX/y5MnYtWsXQkJCJA0qy5Ytw4IFC/JsKFjBaLYhIiIi+holPlFizJgx2LBhA7Zt24bAwEAMHjxY8iA
AIHtd2i+7bwcPHoy4uDiMHDkSz549w4kTJzB//nwMHTpUYbfjv/J2PrXAVKpUCS9evJD53rp169CjR498joiIi
IgKgi5duiAmJgbTp09HZGQkqlWrBm9vb8nkifDwcKkhTiVKlMDp06cxevRoVKlSBcWKFcPIkSMxceLE3E7x05j
UfeHkyZNSS6J86b8zXoiIiCh/KXvE2LBhwzBs2DCZ7128eDFHmaOa0jI27ezL8JPUzqvlCyZE1lh0BERES5UeITJ
YSAY+qIiIiICgG21BEREZEwsKVOLiZ1REREJAiKfExYYcTuVyIiIqJCgC11REREJAxsqZOLSRORERERJg+KeElY
osfuViIiIqBBgSxORERERJAidKyMekjoiIiISBSZ1cTOqIiIhIGDimTi6OqSMiIiIqBNhSRORERILAMXXyMakjI
iIiYWD3q1zsfiUiIiIqBNhSRORERILA7lf5mNQRERGRMLD7VS52vxIREREVAmypIyIiIkEQs6VOLiZ1REREJAx
M6uRi9ysRERFRIcCWOiIiIhIEdr/Kx6SOiIiIhIFJnVxM6oiIiEgQ2FInH8fUEREREREUCbKkjIiIiQWBLnXxM6
oiIiEgQmNTJx+5XIiIiokKALXUkl5FYVdkhCNYch2nKDkHQpt2do+wQBOvvmtOVHYJgBatqKzsEkkcsUnYEBRq
TOiIiIhIEdr/Kx+5XIiIiokKALXVEREQkCOIsdr/Kw6SOiIiIBIHdr/Kx+5WIiIioEGBLHREREQmCmLNf5WJSR
ORERILA7lf5mNQRERGRIHCihHwcUOdERERUCLCljoiIiARBLFZ2BAUbkzoiIiISBHa/ysfuVyIiIqJCgC11RER
EJAhsqZOPSRORERERJAsfUycfuVyIiIqJCgC11REREJAjsfpWPLXVEREQkCGKxSGHbj1i1ahVsbGygpaWF2rVr4
/bt29+03549eyASidC+ffsf0u+3YlJHRERE9BV79+7FmDFjMGPGDDPj6+qJq1apwdnZGdHSO3P3CwsIwbtw4NGj
QIM9jZFJHREREgiDOUtz2vZYsWYL+/fujd+/eqFixItaudXQR1s3rw5130yMzPRo0cPzJo1C6VLl/6JK/82T
OqIiIhIELLEIoVtqampSEUNlnjctLQ337t2Dk50OTpExFRQVOTk64ceNGrvHOnj5ubn6Nu3r8LvhSx
M6oiIiEgQFDmmztPTE4aGhlKbp6enzPO+evUKmRkZMTMTFhYWUuUWFhaIjIyUuc/VqlexadMMbNiwQeH3ITec/UpER
ES/HA8PZM0aqTFNTUyHHTkxMhJubGbGzZs2ABTU1OFHPNbMKkjIiIiQVDkkiamprfnMSZppCVVUVUVFRUuV
RUVGwtLTMUT840BhhYWFoO6aNpCwrK3sgn5qaGp4+fYoyZcr8RPSysfuViIiIBEEsVtz2PTQ0NFCzZk34+PhIy
rKysuDj4wNHR8cc9cuXL4+HDx/Cz89Psrvt2xZNmjSBn58fSpQo8b03Qia21BERERF9xZgxY9CrVy84ODigVq1
aWLp0KZKTk9G7d28AQM+ePVGsWDF4enpCS0sLsLlStXltq/SJEiAJCjXJGY1BEREZEgKPOJEl26dEFMTAymT5+0y
MhIVKtWDd7e3pLJE+Hh4VBRUSvSH3wShKIMGzYMw4YNk/nexYsX5e67dtdetWxQf0HxxTR0RERFQ
IsKWOiIiIBOFHn9n6q2BSRORERILwvbNWfzWC6n51d3dH+/btlR0GZs6c3d3x7Zt2
3KUOzs7w9vb+6eDol+TQ8/f4TigFfTMDBEVGA7vGdvwxj9EZl0zu2JoNLYTilYuhSIlzHB61g7c3iz92as3pC3
KuzjApIwVMj6m4dW9IPh47cHbkIj8uBylq+X20+oN/Hw/T8zYhtdy7mfTMZ1gZV8KRsXNcGr2DtzYzO/yl+76P
cSWXQcQ8OQ5Yt7GYZnnNDRrWFfZYSld9Z5OqDWgFXTNDBEdGI5zM7YjMpfPmYldMdQf2xGWln2xGWlUvBsIQZfGbtwL3
Np6XqVPuzGar92QyGxc0AALFBr3B92SGEXnyQ59eSH9zGusGlmwt0DXURcCcAKyevxJuwN3L3ad2rNToN7AQjM
y0EBIZgzfQ1eOb3DABgXtwc227J7k/j2uTx9MHsMAPMGzcPVE1dRqkIpuA51RaXfKsHA2ABRL6JwYpcOje6J
EQffDLXUuLi6IiIiQ2nbv3q3I20gHicViZGRkKDuM71KxdR38Pc0ZV4Yewt0tnpZ8SlBzg
x6N8wv2IDH6ncw61rXL4872c9jSfgZ2/ukFFXVVdN8xCeraink
MTEFWuXUduEztgYvLDmJtq6mIDAhHz+2ToJv
L/VT///08K+d+/uo+fPiIcralMWXsEGWHUmCUb10bTab2wLVlh7Ct9VTEBIbDdcfEXL+36tqaiA+PwaUFe5EU/
V5mncSIOFxesBfbW0/F9jbTEH49AH9sGAMTu2J5eCX5o/Pgzmjbuy1WTF6BoCrnuk/DNg0
xYNoA7Fy6E8NbDkdoQCjm7pgLQxNDAEDsm1h0r9FdatuWnxAdSklJw98JdAIBdFTu8j32Pv0b+hUHNBmHPij1wn
+SONr3a5HregkqRz34tjH44qdPU1ISlpaXUZmRkBAAQiURYt24dWrduDR0dHVSoUAE3btzA8+fP0bhxY+jq6qJ
u3boIDg6WHO9Tl0OjowNXVFfHx8bnGkJqaihEjRsDc3BxaWlqoX78+7ty5I3k/Ju3jbG1tcWiRYuk9
vHz84NIJMLz588BAO/fv0e/fv1gZmYGAwMDNG3aFA3+bfvmRfHXgRVF
w31GjRqFBgwaS11evXkWDBg2gra2NEiVKYMSIEUhOTpYcx8fFBp1Xj6w1v1Lhzzdnzhz07NCFAMgzDmP
O6dQs2aNaGpqYmrV6/C398fLQe5Ta+E38+ptc9Og3+8Ttyk0GZmkCaMn2Scaa20Dr7Qvt8LmcHhZ2PpX
V0tRmVWDWhWenIFVdB2w/LT5c4+F/BjJ7w8/G/3s1ApmoYwOADc4XZb/o8mNNyrNDb2D9kgLW+qHjGmmMuQ62o
jrd3x5TPb58UB6HZSlMIOLRsXTEAK0i
V58Ls7ixhv9vJ9ds9twpg7jNdpOTQWNR0xIiPXEJZZHhqt9uhjT4BeSCHfcdbEH/
aKa82l8JUafU/5Tc1f4Dtxj+w8iTjPOwFlm44aIOrq1fHtsALQg11
AWm9tVEB8PYdLm0fXJfdxjY+AnwugtEy+rGa4kpGiQ54rf/dqDKTGsD/Y8DHzhQHXxMMgeZQMgjpqnaB/Be
dDXY5GKbRMwcg0RGDTG7tKSBh6J6g9XKB6Iii8hL3QdGFeG8FYRu27d/3g6fW9pC7VxW6RWbK3aT/jB+BYDZ
QiSHmMxFhnsv/ewPiSjl5qj5aRZpkuZYUJ1yeEqNXQy0t2CZDn/6GfPSfPnK3DErFV4MqKSTNh61EbAWE0T4
s3qSxZDvPIiFXsTKBjUGeORjTfm8XKzWqKx5HoWuNKOa+8E0WT5BWuf6sHXj+2CZKyeCMjBLcIdzoHFd6VWHgE
v2Az/6XxkKphgD2G4Qn/dXC+a0SEFi09dlGnu/ewvDNnQkUhj0q1qJBydzmlpXF9pY0qgMRg9JRyL7oHtTvxT
QhgZ/R3m26gb0ezQlF4v0CkddxDcFLqLqSvhK+iVwVHD/ws9zhu2upC8mFyRFfAdIg6JszoFi8kEDGaV3R4a
uO8g1fwMlC7qeEW5eHqKXYG1jaucGtBKrGvPD+CJzGC+SnOejK80J/tQLi5WHYwO3+GzSMUpXXmfxXsOH/N56
XsCHmlsLfGQ2Jb23F56UvzVV3wg/mY2B/FsJz+Phfrj9vMCnsjVwJoOj+nHv+GM4Tf6NaEiKYkjmA9vjL0r3gZ
QGWEvzi8ygb2VAM3OzM1iSlMYHWtKzXRihECV7E03nmxy0VIvp90E4pQRF38ZwMygRxqrXvQM2V9w9P+jRgvp
0PNb6lg6tD0zfymxTP+G83lU/QMJaAwTnlvgaW0T6VbeNyiCNWhUZuaIyqV4pbrZ9uRr4Xwqns7iG4q2M1tEf
zWYBsZXBxsCHmdsMNiBvK+2D4h6Cd9ZGrZknCoQB2+AQs7uFr4Ve5GS1HtNKm6eU+LJrh/I3YqSiHSqUq4f2
W0tC/iMKH4Lv2jL3bPkpNjlQH/SXYgX5u0KIPH08DFxUqIwdjZnRYwnLvWlz2pJVRyl0q7RkUlPGP9uQqqhVk
Yq9+pr0T2bUZ6bECzVQv2WTiXFjkZMnBSsOrkzVHjqv+HjnNmIiRz2F3WcwNcXZX8v9xNEyKhLsAWzxFDEdmB
76ZQ2yvcQfpeEiSu0RDsxlXWsr58KGbVc2iD5ah63Tfq0Hvi8jkMI+kLC55mmOVl0bHlNA4GxrdYy5ZJbC2F

/Pnz7Fv3z4cO3YM3t7euH//PoYMyf1X8YQJE/Dvv/9i27Zt8PX1ha2tLZydnREXFweRSIQ+ffpgy5YtUvts2bI
FDRs2hK1t9j8QnTt3RnRONE6dOoV79+6hRoOaaNasGeLi4gAA+/btw8yZMzF//nzcvXsXRYsWxerVq7/pHjRs2
BClS5fGjh07JGXp6enYuXMn+vTpAyD7+XAuLi7o2LEjHjx4gL179+Lq1atS9yY9PR1z5syBv78/Dh8+jLCwMLi
7u+c436RJk+Dl5YXAwEBUqVIFPXr0QPHixXHnzh3cu3cPkyZNgrp67r8IlUVFXRVF7Ush9Oqjz4ViMUKvPkLxG
nYKO4+mvg4A4MP7JIUdsyBSVVdFOcqlEHzt8/0Ui8UIvqbY+Om/NhV1VVjal0LY1cefC8VivLj6WGEJmEhFhPJ
t6kBdWxNvfIMUckxlsbS2hLGFMe5fuS8pS0lMwVO/pyhfo7zMfdTU1WBnbwe/q36SMrFYDL8rfqhQs4LMfWztb
VGmchmc3nNa5vuf6OrrIqmQ/1v4K/rhp0748ePQ09OT2ubPny95v3fv3nB1dUXZsmUxceJEhIWFoUePHnB2dka
FChUwcuTIHAv1ffz4Edu3b0Oe1atXQsGFDrFixAnv27MnR0gUAycnJWLNmDf766y+0aNECFStWxIYNG6CtrY1Nm
zYByB779/TpU9y+fRtAdnK0a9cuSUJ19epV3L59G/v374eDgwPs7OywaNEiC4SBAcOHAAALF26FH379kXfvn1
Rrlw5zJO7FxUrVvzm+9S3b1+pxPLYsWP4+PEjXF1dAQCenp7o0aMHRo0aBTs709O9StWxfLly/H9u3bJS2Cffr0Q
YsWLVC6dGnUqVMHy5cvx6lTp5CUJP2FnD17Nn7//XeUKVMGxsbGCA8PPh50TE8qXLw87Ozt07twZVatWzTXW1NR
UJCQkSG0Z4sxvvtYfpWOkDxU1VSTFSrfKJscmQM/MUDEnEYnQfIYbwu88RcyzV4o5ZgGlY6QPVTVVJP/3fsYkQ
F9R95N+eZ++tyk5vrfxOP3Jz5lpueIYFbARY4O2ovm83jg8cCneBskfd1bQGZl192S9i5Ue3vAu5h2MzI1k7mN
gbABVNVW8i/nPPrHvJMf7L+euzgh/Fo7Ae4G5xlKhZgUObNMQp3ad+p5LKBCyxCKFbYXRDyd1nx5K++U2aNAgy
ftVqlSR/PenR2jY29tLL38+BEJCQmSraxQr9nnchK0jI7KysvDO6dMc5w8Ouro5atWo
hMDD7w2xlZYVWrVph8+bNALITqtTUVHTu3BkA40/vj6SkJJiYmYmEglp6GhoZKu4cDAQNSuXVvq3LIe3psbd3d3P
H/+HDdv3gSQvaK0q6srdHV1JTFs3bpV6vz0s7s7IyspCaGgoAODevXto06ukNrK2toa+vj0aNsrs2wsPDpc7l4OA
g9XrMmDHo168fnJyc40XlJdXdLYunpycMDQ2ltsvxj+XuIxQt5rjDvGxxHBy2UtmhENFXxIVEYGuLKdjRbgb8/
vFBy8UDYWJnpeywvkuT9k1w8MlByaamnvcriGloaaBxu8Y4vTf3vqS5UpixqYZ2Ll0J3wv++Z5TIrGMXXy/fC
nTFdXV9KFKcuX3XwikSjXsK4Z9kv069cPLq0Sv069cPbm5u+Pvvv7FlyxZO6dIFOjrZ3XBJSUkoWrSozArSnV6+1ljt3xXqXW1ajNO+zesFE+b1bvyPqpMgFprv+WWJzgzS37qR8DNoIXKGNbWe0Zq8zzBtiI
zdHmzZtsGXLFpQqVQqnTp2SO19SUhIGDhyIESNG5NjjcaZRUs1LTtKGO0JrJ8Xa4OWsDA7LAT5+/ehvWcXd8yr+nv96UxnafchbTUQ4Y99X2WqaSFlpYxfSn6J7RqoDixAG5LZCUMUXXo2eNThVi2GtenW9yB8cBvkhCeM+R0lXpzW4p9Vpu7M8QTc5P8DmEeTqbKiX4XW8GcOIzL3a2AOQMWjhKAsGbw2YpIZJqFpDIW6D2e7/O4kdM8e+BwNiAB
/JYqfzJw5E927d8eJEydw6tQpzJgxA3v27EGHDh1kxurh4YEXZI8ZIlS2uPOAHr/zbpbxLRFZGJvRMpX/d65oaI
Ckm9zGV38pldi/YNauO7a5zkBgZ99PHK+hS3iUiMyMTuv+9n2YGSFTA/SQCPn9vdXJ8bw2R/JOfs6zOTLx/EQU
AiHoUBsuqpVGztwvOTN78U8fNTzfP3sQTvyeS1+oa2X//jEyN8O6LyUhGZkYIfiz7B3dCXAIyMzJztMoZmRrla
LODgPot6ONTWxM+B3xkHs/azhqeuz1xatcp7Fm+57uviQq+ArVOXXh4ON68+dNZuXKYwU6PYMhbTbGNgh7e0u6XgGgRo0aiIyMhJqaGmxtbaU2U1NTAECFChVw69Ytq
XN/anX7Vv369cPevl6sUaNGGgICMhxfltbW2hoaODJkyd4+/YtDh5FiPGLPfd+Ccf8gxUB+z5h2cnOdL4KXG4pK6vh3y8Hq4gqGKdH/tTp9i9p2Cu2bKwgHTVqbRhzFYQKJyFSDZnnf6g6zUGI1MGyBKZbT1/6YkZVl3d2mJl7vVjVmJVMAI0Z0TXUXuYMV2P/Y0a9k9MZWoSN2bRDWymNtRjEsTsQ4UM5eWpxPSapaWTp4RaAEVtYDZtL6TvvzWbMV2G3Dk9P1Xb21TWRgJFB1EdgzT1smQzZ9TjzHxWmnnaKv6nT
2EZ4Qo2DOCE1N8Dd7/ygf3OyOzcUFo7WJWXhZQ2Oz4r61TbZdAF4FgUMtITPALNhyc+ZcH+QIqGaQ2EfH8l
XQ8FzMbPtBZaMT2xMY2+i6pWgwaLTDkLsWzQL2DDYdmO+Fjg2+o8SSZzPpDc6XQpTmcQDeRZYWuFDwzMGf8Ty6Lx3Ne1TWLK4Np4n39NaC2QKbqHZhW0cE/SnE/rxP4+9NvVfBN0QKD0Bzf2Ud1pCHj1dtK/TBr5ySwYRaqLvmWlMpgVyRGjh/9NGxtqyWBMJhD4+OiJG3tyqM4JJ3CuhQw2z2tbHtnz9aJKbGN2jxGmwB1WhaWrT2avnv1rtY5D+8Msqg2cfuM0ZpaK/+y5p3EzzbRyC7y5aMGvDWgMVVhdx/OLPEuR69P5/3yZpC0d8v1dwAb6BdFoGCA/TS1Lf6W1P1xkbNHdPn0tL3hnUwyA0ESO/ocYnpJHKuo7LFWhpn1LKNjiR3p1eDGWYyq/xoIJ5MjrIh1P9tKbXJuyh0Bq6GJQaaH5DHqSvPZCR3QBzYY+NZvPK6aJx9A+sYkRhrBLsnFg6thLxUfOXPDRLTYwBoGy9gZMRhZ9gcDeZTcaOZPc6x0OJ4N8F4SmDMaYyUfGnvqGYN8sn+z8OG0PrE0gj/rBAPsAlUIJo6TpTSc5xCRXV8bQP2XMQhZLb8LsuvAAyjdIMi0HW+9fjQfOk+cgSD+XJfh62i8SD2tKEZ1ZP6ML8kbLN10Rbn6Kzl7vB8x9MlzsEvaUvv0wHhn8zl7PsA8KPhsW3ES6uQoeVBWMTvwnPjMiMsgbR2Y7aSxwwk9+5Iz8WS2RuxMGkv4/+t7ExFrQwJC+mP7Dy3jNbXRKqVBJKAJn1yNh4Vl/9ZgeMcGqeVUpPJdoA4fSjbnSa0oVEeGO3nVnWGlXQz/fbfhS6pRmCZxS3dxGmmtu1pWpwpI5XgK30DaYBTfFw0EzcAt9u6lEfRtTxO/20YFuc1HQFo9E7ey8/wFDlprAvFSV3fHObwN1ckbYHaiMpeCDuX3KpdAHp5kqEDP6nVjdUVQ6Y7KvL/xHfRc7HU1D3ZR5zi1rDQ8pnV/WOGp9l/1jOxxnzBVNmjZp1rGdzdOfqBLP0RkAHt7VVl3Xw5w3ZcmpoZr4gqr0UG2Dh2YWq+9IKaXRPMS0XK8+GS1zQwAvjpQAb8kmxNXnALTp5Yz0x/mz0Q3Jnbcnc+kn7ohkm//LAb+rcVfWGWqwUGTkp3g7JRYxcbtwyvZ+I99kT+GK+ONXMR2a5nDOwQtWtq6Cb90fjpOoMy3T+O9UIiFdHPbcdvjIUPWXPqYR+XJUSapNpjCX4Wu5BVYC4iY8pwS83mtrlqTvkXFxRIWLZ7Ccxx9cQnZMGnR5QWmJ6Ts7Pm0ECpRvnrHPaWGHfj1A6eRRIG9Y1fx7vcgzCUwZ+Lsy4P2oCNxn/2fhq8FY2aLvRXfn/q3nQzLXgQ6r5iNcbGkWp31qoZaoLxk36rEHiLHtbkWBW7Zk3fw5iwxqLoSuvGp8czHE5lrKQ7Ng8fvZ0UBOnuiiXGypQx9DOjzrmmXb0MJVRPjdszZYT5U93JbN4GDPYVDpAXKJlUF0lwP8j7kaR3hu6EERbjZBAC2VWPKyeCvcZrQ7qyfvMPdbJl6mzT7Cpa3Qt0ILqd9UM7HFcIDXKyjWdfsqWl7IqkC1b9MSR8Bfcm+zQsECD2ENbe8w0g/Lcb2zu9XodMtRTBX6Ww8gDUMl2Q6ZmXKPTr9mMXv9ojx9eA9qtEFq5Ra4iIgLPZRKfx8rVxzXvpWVLKWEbOPYbbWCMoqhaA/ySQfB0h84w/V6W86Y9l+RXOQgxV2MfJv5GM2eZgvbeKyxT1sv2IIqJCeKumzzlNfxXzvDoPCtHSYSrOSM2hOVX2UtkFeZXG6mu3Hk8X8y4znHS1tLWUOv0CttjyzeLX9tdc6xgWrWY5ZB9i2NOcOJUO0IUV/HIY98eZjjiRpNHZdEkpPBWwQrwOgZM5T4xVBqM2tzVXWcr9eMhvwNkmQxNTD4Qfvp1ufpVn7XCA3pZMdjpjP4y7XDvtwtM3vFHTLzLgjl5zGX8B3g8Ah1LR29RBAR0vMH9+8WqU8AC0VTAlLHfxjDYwqBh8ZSqeOzn8ZwBrXkLVYwVpd03n5tFqPNfLqN+nOzRlspCENWMiPgnK9ze96JoC1mcf3R3jv7bbyuMkVK5ah5Vn+DwC96Wlvff+iDqfmC6HSzgrBQrUv/o96l7Jl56DncEeAH5Q2CA3vrQdZGU6lS6XzO7xI7vvGJvrrrrUmjklqVR2xdMI2R5mtvm4tq+y7WoW4r4QZyLjSrmYW2Pcq9nqO5S4zwTf/pRJGWKa+VOjzXiYAr/nTcnDcUYZ+vLBv6lb2X5VhbdU6H9AnXuDRS6zb0ebbe0iZn9XNvqTLpw7VM8x1hR6vZbRAF62H/l9OJZyP6Nsm+GqQjNvOJ7OUo2qCKzxX5XAtsbe26S3LOcw7LQUfrIuvIbRvZ0Lad0mfpFqXQMcNPz9d5xv0S+9NOj52ZNqzVOjGaXD5GrLBnPHz62cEsxq8YwE2wPAP9FvPV7rm0smWsgP3T5vVdnvCN/3WmqwSgcvNHOvfzAqmbUi36r0e8l5DoxWnW6e8hAdqNOYq5oq1zZh6ooZEwK31Of+N8Xh5X5CcXT+gEyZ8tmH2EQ1oaQ9BcOjRO1x6uAdCIbS2CDCJAdA4MCVYWbBSS0+GtvNKFe0EuHH3i1k6XeAF6PM1+v5VZ/Ya4r8/E8mvyOo0r2Sja9ynB4o8OKTwGCpxSyh2Wew/x9r/M4ctQHlaKoazn5e/VL6pl+h7tk+Yf8Gr3UWaPBe+ebT5W8n4Cwl2AFQE3CL6k7dsjmmYZ/tvaIJ93pu/MeOeLvtjo2ZCxzMyzwY1/2Ymvf2YJbvUD+1NLbPJ8KVOnrHm5ekNsU6Vzv99CGbbRW7aFMtrdQgHJ2drg51Cyf2Mt28yfoQ8mp7OLHK1s1nCk6mCGh9B1SeM5lq+5H2kZbhqJtXG9lXlrg/3wNkSrHpR/QK8bZ9eVqxUq7o2+IoHCNfgC18pIO8zfCyH3uaNJIMgZ3/n+S7IQ+wIxMEJvpDjvhZM6kmkDKj4DVHgSa2hjXs+7/tL2xNy3sCQt0gIpzV0gxvB2E4rC4E+EWyVCr6iJ+y1nfm4EKXiSyeE85lYnxfvmTb+4DJJ71ff3kVaNYBKKl0JO7Oh2TBykwHwYH6GTMTYjNzoAMBkRczHd9vWwA78/FDOwpqt6fF8QF8/F1m9ANZL8

+fClJ1vJb3759ERMTg6NHjyrl/D9qTskeyg5BsNILZ+t+vpl29+st0iTb3zWnKzsEwbqQ9VbZIQjaqZd506P2e
tGOCjtW3Yh/FXasgkJYgxS+U2pqKmJiYjBz5kx07txZKQldfHw8Hj58iF27dgkuoSMiIipICuusVUUpUBMlFG3
37t0oWbIk3r9/j4ULF+bJOf67Vt+X25UrV9CuXTs0b94cgwYNwu+//54nMRAREREVmO5XofrOuDFZihUrBm1t7
XyMRvHY/frj2P36c9j9+uPY/frj2P36c/K6+/WKZSeFHatB5AGFHaugKNTdr/lB3lp9REREpDhi8NeyPEzqiIi
ISBCy2LcoV6EeUOdERETOq2BLHREREQlCFrtf5WJSR0RERILAMXXysfuViIiIqBBgSxOREREJQpayAyjgmNQRE
RGRILD7VT52vxIREREVAmypIiIkFg96t8TOqIiIhIEJjUycfuVyIiIqJCgC11RERERJAicKCEfkzoiIiIShCz
mdHIxqSMiIiJB4GPC5OOYOiIiIqJCgC11RERERJAhiZQdQwDGpIyIiIkHgkibysfuViIiIqBBgSxOREREJQpaIE
yXkYVJHREREgsAxdfKx+5WIiIioEGBLHREREQkCJ0rIx6SOiIiIBIFPlJCP3a9EREREhQBb6oiIiEgQ+Jgw+Zj
UERERkSBw9qt87H4lIiIiQcgSKW77EatWrYKNjQ20tLRQu3Zt3L5909e6GzZsQIMGDWBkZAQjIyM40TnJra8II
rFYzMSXcvXx5l5lhyBY47oeVHYIgmYtVld2CII1+t5sZYcgWEWsmyo7BEFLTgnL0+NvL/anwo7V8/U/31V/796
96NmzJ9auXYvatWtj6dKl2L9/P54+fQpzc/Mc9Xv06IF69euhbt26EKiQVmQspKC9PR0GBsbf/+Ff
yMmdURERPTL8fT0hKGhodTm6ekps25WlCwyMzNhYWEhVW5hYYHIyMhvOt/EiRNhZWUllRgqGme/EhERkSAocvF
hDw8PjBkzRqpMU1NTcSf4gpeXF/bs2YOLFy9CS0srT84BMKkjIiIigVDkY8I0NTW/OYkzzNTWFqqoqoqKipMqjo
qJgaWkpd99FixbBy8sL586dQ5UqVX443m/B7lciIiIiOTQ0NFCzZk2pSRLZkx7q1Kkj97gFCxZgzpw58P84ceHt7w8H
BIc/jZEsdERERCYIiW+q+15gxY9CrVy84ODigVq1aWLp0KZKTk9G7d28AQM+ePVGsWDCxZg+vTp2LVrF
2xsbCRj7/TO9KCnp5cnMTKpIyIiIkEQK/EpYV26dEFMTAymT5+OyMhIuIVKtWDd7e3pLJE+Hh4VBR+dwBumbNGqS
lpaFTp05Sx5kxYwZmzpyZJzEyqSMiIiL6BsOGDcOwYcNkvnfx4kWp16GFhYXkf0H8wqSMiIiJBUGb3qxAwqSMiI
iJBYFInH5M6IiIiEgQ+rF4+LmlCREREVAiwpY6IiIgEQZFPlCiMmNQRERGRIHBMnXzsfiUiIiIqBNhSROREREI
Aljr5mNQRERGRIHD2q3zsfiUiIiIqBNhSROREREILA2a/yMakjIiIiQeCY0vnY/UpEREEUCLCljoiIiASBEyXkY
1JHREREgpDFtE4uJnVEREQkCBxTJx/H1BEREREVAmypIyIiIkFg56t8T0qIiIhIENj9Kh+7X4mIiIgKAbbUERE
RkSDwiRLyFeiW0hsbGyxdulTyWiQS4fDhw0qLJy/89xqJiIhItiyFbYVTvid12dnZcHd3l/pe3Rt/Vuufu7t27c
vrkzp070NXVVfhxAaBx48b+4a4dOkSPD09MWnSJKn3WrWW1ZMa7/D/eEiRj/zsd09XJrtz69pkTKazo9Jxj/r+
g/ngVeggJBE+n/xaOyqHkFcsH4HIvcGXlbl/wmfCdBBtIDLQYuA5Jtl+9oo9I8pwMj3WXl1Y5aq4PXrBfP4Tql
VZFpC0eHqEDMTUHd2u/eiD/2zuCFf3Cu9au3rINWY11hUtwMMaGRO0y1EwEX/WTW7TqvH+r3+BOHZm/DxcOnpd6r1KQ6WozsCKv
yJZGRmoagW4HYMGCRIi9NKar3dEKtAa2ga2aI6MBwnJuxHZH+ITLrmtgVQ/2xHWFZuRQMS5jBZ9YO3Nt8WqpOt
T+bodqfzWBY3AwAEBvOCteXHULoxQd5fiOF1V2/h9iy6wACnjxHzNs4LPOchmYN6yo7rAJh6rTR6N27GwwNDXD
zxl2MHDkVr2ZRtmyZfDxwOfcvOWLaVO9EBTO+TNbqpQ15ntOgaOjAzQ1NXD27CexXXBFpCzf3VLn4uKCiIgIgIqa1UqVJ5ERvMzMygo6OTJ8cGGgBIlSmDr1q1SZa9fv4aPjw+KFi2aacoO4bt
533qIRbu9MbBdY+yZNQjlSlhi8KLteP8xGz/OQxq3wSHPIdjZp/2OH37EZYfOCeps3PGQPgsGy/Z1
k3oBQD4/bdK+XJNeaVGa0d0mNoTp5b9iwWtJuF1wAsM3T4ZeiYGMuuXrlUBd474Y2m3mVnYvjGmDHHQOnTU7H4MbeC/BqOQn+Z+5gwPrxKFq
2RI66VZx/g01107yPjMvxXjWXWuj59zDc3H8RXiOmYEnH6bh75KrCry+/lW9dGO2m9sC1ZYewrfVUxASGw3XHR
Ojkcj/VtTURHx6DSwv2Iin6vcw6iRFxuLxgL7a3nortbaOTZoRQ
oY8YMwuDBvTFixBQ0btQeyeSkfcOTodmhqaua6T/OGtbF+3Q40adwBbdq4QV1dDUePbYeOjjYAQEdHG0ePeP7YBYL
Earlt3h1KwTNDQ0sP/ARohEwu6/FCtwK4y+O6nT1NSEpaWl1LZs2TLY29tDV1cXJUqXJAhQ5CU9PmP8datW1G
kSBEcP34c5cqVg46ODjp16oSUlBRs27YNNjY2MDIywogRI5CZmSnZT17XZN0mTTFs2DCpspiWiYGGhoaMDHx+ebr
qV169aIjY3FtWvXJGXbtm1D8+bNYW5uLlU3D8+bNYW5uLlU3NTUV48aNQ7FixaCrq4vatWvj4sWLP32NAJCYmIhu3bpBV1cXxYo
Vw6pVq6Tef//+Pfr16wdzc3MYGBigadOm8Pf3l7w/IigpvSNcCwG7E+cpm8Pf3l7w/c+ZMVKtWDRs3bkSpUqWgpaUFADhw4ADs7e2hra0NExMTNO
Dk5ITk5+ZvuTX7b4X0dfzSqifYNa6BMMXNMdW8DLQ11HL7sK70+X1A4qtmVQEvHKihmZoS69rZwqWOPRyGvJXW
MDXRhWkRfsl32e4oS5sZwKG+TT1eVN5r2a4Xre3xwc/9FRD5/jT1TNiLtQxocXZvIrN+4TwsEXvKDz/pjiAp+j
RNL9uHl41A06uUsVc/QwgidZ/bG1pErkJmRIfWeiqoK0s5wx+H5/+DqzOIDo1A5PPXuH/iZp5dZ35x6NcCD/Z
cwKP9l/E26A10T96C9A+psHdtJLN+5IMQXJy/G0+03URmarrM0sE+9xFywR/vwqLwLjQSV/7aj7SUj7CqofgeD
aFo4PgbRgzoBadG9ZQdSoE0dFgfLJi/EidOnMWjR0/Qv98YFC1qgTZtWsqsv2bNVnhN3Dj67+VQUldDU4bh
bF0f16vYAAEdHB5QsWRwDB4zD48dP8P8fFxSwz/Y48dP8nxWGCy0grTY0m1C+Ox48fY9u2bTh//
jwmTJggqVmgVSc1JQXLly/Hnj/eyn4+y6jh/V7/pn2oHQTMQ1p11nG/jvnDcRuT9c5EyZhB5/Ts/7/XQzxt+9wNX8V41bU/g8+zHLOmrVHs6i8bu
zD4oVK4amTZt+0zEONDTQo0cPbNmyRVK2detW9f2gbb4Q6Ps/m1u2Xf/Yx8cXuNxoD2YBwe2i8cQe2FM7SoekAZcZ1Ye+Vvp1ATMOwVe72M59ZDzHi1wKn+8jh4kzjR/+p+WdqAXI1+wRRHRRjKP41ATpLsspVsbuaTgsejdNyQfyHRGzKg3nifpPUd/vsTQ==

XQcrvo/Q4Oqdrme48T1B2jfsLqgf6mqqquiROXSeHrtoaRMLBbj6bWHKFVD9rWXql4WT649kioLvOwPmxplJa9
FIhF6/jOMPuuPITIo5zOvUbkUjIqaQCwWY+IJL8y7vRaDt06S2donJCrqqrCOL4Wwq48/F4rFeHH1scISMJGKC
OXb1IG6tibe+AZ9fQf6ZdjYlIClpTkuXPjcsJCQkIg7d/xQu3aNbz6OgYE+AODdu/cAAA1NDYjFYqSmfu61+fg
xFVlZWXCs+5tigqcC6btnvx4/fhx6enqS1y1atMD+/fslr21sbDB37lwMGjQIq1evlpSnp6djzZo1KFMm+w93p
O6dsGPHDkRFRUFPTw8VK1ZEkyZNcOHCBXTpOuWrcfzxxx8YNmwYjhw5AldXVwDZCZm7u/t3/dHuO6cPGjRogGX
LluHevXuIj49H69atpcbShYeHY8uWLQgPD4eVlRUAYNy4cfD29saWLVswf/78n7rGevXqScb8n8m8XcPff
/+N33//HVevXsXt27cRHROtaY5ftGgRDh8+jAMHmDAgAEAsrtct2/fDjOz7DE8vr6+yMjIwB9//IGSJUsCAOz
t7eXei9TUVKkkGQDEaemSMWp55V1iCjKzsmBiKD1+OsRQF6ERMTL3aelYBe8SU+A+bxMAMTIys9C5yW/o1OZ26
8r5eO+QmPIRbetXV3T4+UrPyACqaqpIjI2XKk+IiYdFGSuZ+xiYFUHif8YWJsbEw8DUUPL698HtkJWRiYtbTsk
8hqm1BQCg5chOODh3O96+ikGz/q0xcs90zG4yCinxBbMF+Gt0jPShoqaKlP/cz+TYeBiX+bkhGKbliuPPQzOhp
qmOtOSPODxwKd4GvfmpY1LhYmGR/e91dLTOv3PROTEw//97XyMSibDwr+m4fvOOAgKeAQDu3L6P5OQUzJO7CTN
mLIRIJMLsOROhpqYGSOvzrxyxYCusExwU5btb6poOaQI/Pz/Jtnz5cpw7dw7NmjVDsWLFoK+vDzc3N7x9+xYpK
SmS/XROdCTJDgBYWFjAxsZGKkG0sLBAdHTON8WhpaUFNzc3bN68GUB2EvPPo0SOSO4u7t/1/VUrVoVdnZ2OHDgADZ
v3gw3NzeoqUnnug8fPkRmZibKli0OLPTO9yXbp0iUEB38enP6j1+jo6JjjdWBgIADA398fSUlJMDExkTp3aGio1
LlLliwpSeg+XVezZs1gb2+Pzp07Y8OGDXj37p3ce+Hp6Q6QlDQOOp7a/th79yB5XjTmAoNh2/jCk9W2PPrMFYMrw
rrvg/w7ojF2XWP3T5HupVsYW5kexxUr+yEpVLoXHvFvhn3Jpc63z6oXR61SH4ed/GyOeh+Gf8GojFQPVWjrnu9
yuLC4nA1hZTsKPdDPj944OWiwfCxE524k2/hi5d2iEq+rFkU1f/+R/Mfy+dg4oVy6FXr+GSstjYOLj9ORQtWjZ
DdEwAIiIfooihAe7ff4isLGF3PHJMnXzf3VKnq6srNdM1LCwMrVu3xuDBgzFv3jwYGxvj6tWr6Nu3L9LSOiQTH
f774RWJRDLLvucD169fP1SrVg2vXr3Cli1bOLRpUOmr1Pfo06cPVq1ahYCAANy+fTvH+OlJSVBVVcW9e/egqqo
q9d6XCVteXGNSUhKKFiOqNX7vkyJFikj++zhFVVXH27Flcv34dZ86cwUo4VKzBlyhTcunUr14ktHh4eGDNmj
FSZ2O/oN8f6o4z0daCqooK3/2nteRufDFNDfZn7rDrog9Z1q+KPxjUBAHYlLPAhNR1ztx5F/zYNoaLy+ffKm9j
3uPU4BEtGdM27i8gnSe8SkJmRCf0vWtkAwMDMEAkx72XukxDzHvqmA6s69/Hsupq
qaKP6a4oUmfFphRfzji/3/siC+6ZjPSMvD2ZRSMrUx+/sKUJOVdIrIyMqHzn/upa2qI5Jj4XPb6NlnpmXj/Igo
AEPUoDJZVS6Nmbxecmbz5p45LwnXixDncueMnea2pqQEAMDc3Q2Tk59Y6c3MzPHwQ8NXjLV4yCy1aNEXz313x5
nWk1Hs+PldgX7kRTEyMkJGRifj4BISE3sX7sCDslzXs/vfjwvXv3kJWVhcWLFOv+kO7bt++nA/sW9vb
2cHBwwIYNG7Br1y6sXLnyh47jdAGDDoajDRoO+Nmwc7h582aO01xUqVAAA1
KhRA5GRkVBTU4ONjc3HVckEqFevXqoV68eZk6fCuUqyU5k5ErdPNDU1c8y4+pjHXa8AoK6mhgo2RXErIAS/
Na2Zfd1ZWFm4FhKCrUy2Z+3xMTc/Rza6qkv36v7/AjlzxhbGBLhpULQuhy0OzPxMtHISpcgHX1x4PztwFkP3/c9m6l
XF5+2mZ+4Tef4ZydStLLU9Svr49wnz/31Vz8DKeXn0otc/Q7ZNx+9Bl3Nx/EQDw8mEIO0lPTYFHaCiF3nwIAVNR
UYVzMDHGvhbtEQlZ6JiIfhqJkvUp4fuZedqFIhJL1KsF321n05O38nkYoIqhpc7/1XlpSUjKQk6R+vkZHRaNy4
h78P4nT19fDb79Vw8YN/8g1uIls9C2rTNcnLvixQvZmghMzzuVal4Tvp/+FsbW1RXp
6OlasWIE2EbdrgrVrWLt2rSJ+yb9+vXDsOcP9OzZE4Ku9OzZE4sXL0bLb6QNDlbkNKoxDHx8fVKlSBa1atKq1a9ewcOFCtG/fdMmfPnsX+/ftx4sQJMcR3RvnlWbN29
w4sQJdOjQAQ4ODjKPeevWLj4+Ehm8t66dQ+sMTGSZLGgcXOpi2kbDqFSTuF0c/5y+gQ+paWjfIHuw8JR1/
8LcyAAjXX8HADSqXg47vG+gbEKLTuG0dMuTmGSrSyJ6JMHZm3ewhLc6a/TvNK0cVL9opcvKysKRK/fRpn41qH0Zo2m88AbfFQxD
+MBhhfsFoOrclNHUOJQmY2+KhiI+Kw9GFuwEAFzefwqi9M9C0X2s8vW5wE2s8vuCLmm3qwtq+DHZ7bAAAJL9PQvJ76aVjM
jMykBATj+iQCADAx6QPuLrzHFq4H5eCAiH4Qiwj8YDn1coK6jiYf7LwE
AWi4ZiKTId7i8MPvHqoq6Kkz/vzSJqoYa9C2NYV7RGmnJqZKWuYYTXBFyOR8Jb95CQ1cLFdvVhXWdCtjntlA5F
1kApKR8QPirz2MKX7+JwpNnwTAOOEdRgY/z+hmrVm7GhInD8Tw4DC/CXmLa9LGIiIjCsWNnJHVOnNiJo8dOY93
a7QCyu1xdXduhi2t/JCUlS8bmxccn4DTJc8TGvkXt2jWW8K8ZWLlik9RadkLEMXXy/XRSV7VqV
SxZsgQLFiyAh4cHGjZsCE9PT/Ts2fObx+zVqxdu3bqFW7duYWNjI/VeSEgIjY2NW7dulUmGrU6Y+LeP+LIrU5YtW7Zg7ty5GDt2LLl6/fg1TU1P
UqVMHrVv//CK2Y8eOxd27dzFr1iwYGBhgyZILh5MmTmDJ1CnR37o2YmBhYWlqiYcOGsLCwyPWYB
gYGuHz5MpYuXXYqEhASULFkSixcvRosWLWLX463rzgUtse7xJSsPrgecTGJ6GctSVWj3ODiWF293ZkXDxUVD63zPV
v2wgiiLDqXx9Ev0uAkb4uGlUvh2Edm0kd9+bjEES8jUf7ht8+k6yg8z1+A3rGBmg12hX6Zk
BczgVj8uZMi1PcZto5cgdZju6DN+K6ICYvE+gF/IeLZy+8676H5/yArIxM9lwyFupYGXvg9x/Luc/AhQZiTJD5
5cvwWtEOMUH9Mx+zFhwNeYH/PhUiJTQAAGFiZQpz1+Q+JnoUR3E/Nl7yuNbAVag1shfAbgdjTdR4AQMfUAK2
IKueRGkJqYg5slL7HNbiBdXpWch/OoePQlCn+ETJa8XrlgPAGjXwgnzpo5VVlhKt2TJWujoamlPlSk8YGhrvU
7aN+ul9SktVKlS8LExFjyesAANwDA6TN7pY41cMA4/PNP9uoKdnalMWv2BBgZGeLFi1dY9NdqrF69Rapu6dIlYWJiLHnd3/4tnAGx5lO6yn73zzGCQL4Cbkx
cJpyPVr6gWYS9tLe69PMMUoAZ9kBkgyPIxTy2WTqpDS1X1lmunVt8ajUzpqyb2KI3Bc8z2Z3oR8u7kBPQUYdumbvsRbv
ZNPJC6Ia1x8h7CwMJQpUwZ37txBjRqF5w93Qf0Hx5t6vVyKZxnU9q0wQBM1anPdd/4XV6HuzlR2CYBWx/rYlsUi

```
25JSwPD3+aBvFjY3+O2yPwo5VUAh2gEd6ejrevn2LqVOnok6dOkzoiIiICjlOlJBPsEndtWvXOKRJE5QtWzbHY
r5XrlyR28345dMuiIiISBjE7ICVS7BJXePGjXN9OoKDgwP8/PzyNyAiIiIiJRJsUiePtra21Fp6REREJHzsfpW
vUCZ1REREVPhwSRP5vvsxYURERERU8LCljoiIiASB7XTyMakjIiIiQWD3q3xM6oiIiEgQOFFCPo6pIyIiIioE2
FJHREREgsDFh+VjUkdERESCwO5X+dj9SkRERPQNVq1aBRsbG2hpaaF27dq4ffu23Pr79+9H+fLloaWlBXt7e5w
8eTJP42NSRORERIIgVuD/vtfevXsxZswYzJgxA76+vqhatSqcnZ0RHROts/7169fRrVs39O3bF/fv3Of79u3Rv
n17PHrO6GdvQ66Y1BEREZEgZClw+15LlixB//79Obt3b1SsWBFr166Fjo4ONm/eLLP+smXL4OLigvHjx6NChQq
YM2cOatSogZUrV/7A2b8NkzoiIiL65aSmpiIhIUFqS01N1Vk3LS0N9+7dg50Tk6RMRUUFTk5OuHHjhsx9bty4I
VUfAJydnXOtrwhM6oiIiEgQssRihW2enp4wjw9PWWeNzY2FpmZmbCwsJAqt7CwQGRkpMx9IiMjv6u+InD
2KxEREQmCIhcO8fDwwJgxY6TKNDU1FXiG/MekjoiiH45mpqa35zEmZqaQlVVFVFRUVL1UVFRsLSOlLmPpaXld
9VXBHa/EhERkSBkQayw7XtoaGigZs2a8PHxoaGigZs2a8PHx+RxLVhZ8fHzg6Ogocx9HROep+gBw9uzZXOsrAlvqiIiISBCU+US
JMWPGoFevXnBwcECtWrWwd0lSJCcno3fv3gCAnj17olixYpJxeSNHjkSjRo2wePFitGrVCnv27MHdu3exfv36P
IuRSROREREJgjKfKNGlSxfExMRg+vTpiIyMRLVq1eDt7S2ZDBEEeHg4Vlc8doHXr1sWuXbswdepUTJ48GXZ2djh
8+DAqV66cZzGKxGIxH6RGufp4c6+yQxCscVOPKjsEQbMWqys7BMEafW+2skMQrCLWTZUdgqAlp4Tl6fG7lGyvs
GPtfXFYYccqKNhSRORERILwvWPhfjVM6oiIiEgQlDmmTgg4+5WIioEGBLHREREQmCMidKCAGTOiIiIhIEzu2
Uj92vRERERIUAW+qIiIhIEDj7VT4mdURERCQIHFMnH5M6kkuv4RhlhyBYDqZ2yg5BOIJVtZUdgmDN4gK6P+x9+
HllhODOw5jUERERkSBwnTr5mNQRERGRIHBMnXxM6oiIiEgQuKSJfFzShIiKgQYEsdERERCQJnv8rHpI6IiIg
EgRMl5GP3KxEREVEhwJY6IiIiEgTOfpWPSROREREJAme/ysfuVyIiIqJCgC11RERERJAjsfpWPSROREREJAme/y
sfuVyIiIqJCgC11RERERJAhZnCghF5M6IiIiEgSmdPIxqSMiIiJB4EQJ+TimjoiIiKgQYEsdERERCQJb6uRjUkd
ERESCwCdKyMfuVyIiIqJCgC11RERERJAjsfpWPSROREREJAp8oIR+7X4mIiIgKAbbUERERkSBwooR8TOqIiIhIE
DimTj52vxIREREVAmypIyIiIkFg96t8TOqIiIhIENj9Kh+TOiIiIhIELmkiH8fUEREREREUCbKkjIiIiQcjimDq
52FJXQN24cQOqqqpo1aqVskPJVzNnjMPLF75IjH+BD1fzEW77ioH3OaF3k1nc9A1zHkg2lrY1Zyu7z5LeJ9QQ+jjxiC5VRx4Dh6+0Uo2HvrrXhvNHLDK3aJ6yBoEfYbTG48SMYZl0BAW1dc1+2/bvw+BPCbybgZ9lDdqbnqe3nHsz7+0oyKyFkXc7rdDWF+ftfXjoYCQDcjX8CnDsTfAH8LzOqGUZbbJEbt9XbgHbMKBk6XGbpTB/XLAb5VFRGzLlh3S7f/ZtgIJ4w9ADbp+iHDjfnGu5mWmbvp3vcVjOtOjftavMkKyHRAFqHkFqiFX7lGqw9vvCPfmmSpt8pkFm2rU3U9KcGl2bkLePy6njj31Jmyf0S6lR6K3fsN7j3TDHAMAiB5vDJh+W3HMnjS7/sxYg2AkdfpnrGhJm66Pf7N8JHeSY+PPWmvEY7gdhYtuUp1I3rfE7WzV/iKT0JHUhjbGyd+h5CbBXhAm3lUdbtFyGJiNuZSbx3npS7Nig3vc9aHlGnTlQp82wwrKcwEIfe7Uq1TIVw3HzdqIR1XDh2dNu1JfGMtsu9J0FudzUCx3OCGnFAX6FHl1bdRc3HPB5Pq4KZFOvAfV87nSAd1mQYWxv3+Aw+AL+9H9qyBu2mh0QkJnBkwVP0nsAP8QjaZ2nOhWoKXBUAoRNkM5H+5CaqBxGFvcyV2sGxnRq8A0AQkv9ApjI+L6zGOHAM8CqgSzGfxTWK5WP1JHr8UHNr5qkOe2PclcJcNvIWJIjK1yFD7XlPn9G9BI2fJqlZHqtbF2EG7GXTDF2pc2iHoCz2a3H3bI0n4ajUw3V6SWmeRzuGVsVqZWJSCr2A9vMgG4ZQYYNnrOp7JTnI6kAAbhIACSDgy5TJU1jMsZ5HKgCkCQwzZ2eNTl5CkGYHLeFUGWujXzcs9WFJBCFPbWf8OKnNvK1vJm9Nlz3c5Nch9NNfRlyiUmgi5Ph3j5kXuw4o1tZgYCYQh4PZjzpD4zgzpHz1wj8QO66WESvIyzcScnrQfZJvO6NVYw4rJNjMPLZmbaV0W5zupZYDWNvhFxaPVzhgfkZGL/AWQJmRIkmgYgYL1e2RyPqGfnfObjUzQnqDTVn2DXCfKrsWDv3RnHrnHNjIKKCSSmNlfzTaAK32aWOOlM0wFkq9d5Yp2emGYlxiPb3ajrC5yZbFEOjaWwwe8PRdNtEFUo4KAJYHmIRkcjMzVIoa0w+AJ0kY4OjrlBo5sdJNtZ8PeW0uDFhQ6x3OhRRTEjfqHgrmPYaW1n6Z/Izrxeb6mC1ZgO8OZB/wZ5r0Ezo2NVm+OvuBhmRsRbg4dxHtiyYE6YF7OWXJBOdbwFUPr1rtIaWPXbsvAcB4oQewWUGLxP4Y0qx+5jU+9Sc37LgtSPl9n7nR9b/JmrMnQGOPpc8FhgMTvRJL89MtPyRBsxk7e1lMUkYHi6Uc9SpTZZyJr1Z5Tb67t1oiP8gNlIHMq9KIWO0gBcMxYEtVGMG4+fvzyrCPX7ALuqhiJDEX9Z+BSFB1H+Pf1ri89FlM3PXb2nvTv3y2TnEmuZ26HU7E9u9aFSG+gCsKHHu8XL0mZkXlEmzXGrtYJ+MhUf9htVXC0s9tbDPqQoJbbnt2CVKyE+BPyDqWrUmD+Vs0bvznfi8rHPIhhdIp3YntGPHj8g2AqJnRvTzAWJ8Ut8A8P6Pe4E+0lyKOoO3cpe/tsZ2frNWLr9fB30KSBMWK5zbFv5sUfHSr/MPmHfqqq2DrHgQnAi0JxhT9XTWGsKJ5vbDn7VGJdsJ8CAKSOcdVqWq7FGZk/eRO0bYJb1TeH5KFDPWoLUTCOCa6UFSFsM9SdBkp6yh8WMB3GlnhKSFQiyRdJtFMrkm2zPfaf7+mC+nrGMzNg0+5uQdWAhMoDUiakqCZ2ygIqm0J6tcfJCYy3BzeHAd5/Sp8F5gSNHl4v3vkCZxo7Xtml+OgdbThN8lSOR1+VzpPklc3WwhfRjCCq9ZZqmO+iMfhp4h/Zq2VnO9ov1o4TH+ePFsHTyMfx6AF/fYNe0XaV+BcVF5mnN+lmnAEx2qH8nnEO+yuW9IShKQT6xnHWEuPaBDNmGlwnOc9l3ubkVRHNy3dGbEaAhupIjx27t/f3G5hnJVnSjSOsXcc6bdlmhQqPvJ+eR5JsAQbxCc3lXHKlY6Ol4HCJTzaqXr+sDXtsM6XNx6e8OlGO9cY7/dKB1g7U8zvhLP4kE9mVeS0OOC+3TP2YjISsEkELdeT6p8dLQOpUpZ+x2GjUD3MDIgJ2YdhrByrBJyXbmPtHYFzYOl5h2yIsrcpXAdM9jpE7NvtmbxSVNv4sxBIrxWVjL+bpQGdlVjmuSRwmrqDeqyHVNtZ6zYqGj5dpHCDg+LuXcA7pW+eFLSn6lsLmDKXddjlIeBXr3v/QJKxzG7t6BkVwPvD86GpyrHbV7xN+2yHU+QaH/TJpiH0SSVbyqW2xwFJqEK7mdHqvtiEfUjICEnxF3vXMnRSTYCdvbNyhjZ1VWpxQvIivfoYAhNQmlVBmKtQ/4WAYBBa2jXm4NXD7s/cgZL4GyvBaTQTwxLdcNgRzeJKIJwPexZqwcfZWS1ePxmYbEx1xQTIpbxmd+bN1BhHUqz8DpNfB8fjvXmJ3BH5hR6HjE9UAYFkZGmx7O55FYFOzF4+Cj3uzCGD75u6QOJU4C+rWNVIoI7c2sfAuGhdqzrW+/ACZtHuUbbbBPKKA8O0BAvzHNnEE7sXdDA8oH+TI0JW6hd1vW5mHf7mSlyWNfeu5J0u9ZdkFnTkCtw08g7TLqn8wVZw2tcj+8dJLPxbQy7JU1/5tKtRC7JkS16tRpq7EG8ofTnx8RZDXvO8QomO0mJKTT4wJULlNLUjG4F2DMeBsB/6VyYhZwKVNqUk7+nN9wHPb3K0bEq0PzQFZumD8FOOIlFhqNQHwZkfCM4Q2Q4m0DmKIg0NejNi2k6h4R+Z+8DJGKg0sKvmMWrM5fVSNeAg0MEH+f3X3+ncF1p3q4fUMCNFfLEAXtfHRnh6QrG5j4Fxl7WLKDGDhC0RojyNx7AIbbcMjb+q5zOPtFJQpjYPSLbkt5zsLx1ZbLsxuOPu+wPSlyGe2LIYU6sMgZX+CtQjBPiNuiH5aUzM8eXK9dUBcnq+/W3m8kIGvPIOFEOUCwDtsZa6Fxq0IBSmpIyCuCwWzv1vpYMJyTbX/Wq6MTUwVIUBuq/rlBSLzyRjAvqMqYZ4pSLxr5ohOYO6m
```

```
gHv455JNnX1H5+I/fRpMGr+1hx167XGuvXbsXnTUlSokLPrU19fD8eObEdg4DPMmi3sMXXOHZxwPuiUZFNTy9u
J7Peu3UfP3/uhf9thuHnxNuatmwkjkyJ5ek5FadK+CQ4+OSjZ1H7is/atNLQOOLhdY5zem3srXclyJTFjOwzsX
LoTvpfzrhtbkbpOaYeo6MeSTV099zGI3+rvpXNQsWI59Oo1XFIWGxsHtz+HokXLZoiOCUBE5EMUMTTA/fsPkZW
V9dPnJOHKEosVtuWVuLg49OjRAwYGBihSpAj69u2LpKTcW+Pj4uIwfPhwlCtXDtra2rC2tsaIESMQH//9P5y5p
EkBs2nTJmRkZMDK6vNMPLFYDE1NTaxcufKbjpGUlISBAwdixIgRd8d6ztrbOdT8PDw+MGTNGqszIRHZ31M86duw
Mbt/+3P2l+f/B/BYWZoiMjJaUW5ibws//sdxjpaenIzg4DADge/8hHGpWw/Bh/TBk6ERJHTO9XZw8vhOJicno2
LkfMjIyFHg1+e/KmWt4fP9zC5C6RvYfV2MzY7yN/txaZ2xmhKDHz3/6fB8/fMSrsNd4FfYaj3ODsP/qP2jTrSW
2r9z108fOazfP3sQTvyeS15/ulZGpEd5Ff255MzIzQvBj2V3KCXEJyMzIzNEqZ2RqlKP1DgDqt6wPTW1N+Bzwk
Xk8aztreO72xKldp7BnueyxnwXRiRPncOeOn+T1p++tubkZIiM/t9aZm5vh4YOArx5v8ZJZaNGiKZr/7oo3ryO
l3vPxuQL7yo1gYmKEjIxMxMcnICTODsJCv22cHhVOQug27dGjByIiInD27Fmkp6ejd+/eGDBgAHbtkv3v5Zs3b
/DmzRssWrQIFStWxIsXLzBoOCC8efMGBw4c+K5zM6krQDIyMrB9+3YsXrwYzZs313qvffv22L17t9L17N8qVK4eTJO9
KvXfnzh2p1zVq1EBAQABsbW2/6/yampo5Zqzl1UyzpKRkJCCUlS5VFREShaZP68P9/Eqevr4datapj7frt33VsF
RUVyR+bT8c5dWIXUlNTOf4Pd6Smpv78BShZSvIHpCS/liqLjXqL3+rXkCRxOno6qFS9Ig5u//ZJJt9KpCKSO6u
2IPmQ/AEfkj9IlcVFxaFa/WoICQgBkH2vylUrhxM7Tsg8RkZ6BoIeBqFavWq4cfoGgOzvRrX61XBOa87769zVG
bfO3pLZRW1d1hpee7xw7sA5bFu47WcvL1/J+t5GRkajceO6ePD/JE5fXw+//VYNGzf8I/dYi5fMQtu2znBx7oo
XL17lWu/t2+ykuVEjR5iZEXCwMBDe3t64c+cOHBwcAAArVqxAy5YtsWjRIqkGmm08qV66Mf//9V
/K6TJkymDdvHv78O9kZGR8V08Mk7oC5Pjx43j37h369u0Lo1OeOPp5SI6duyITZs2Yd++fViyzPvn37ws/
PD1u3bgXwOQGbOHEi6tSpg2HDhqFfv37Q1dVFQEAAzp49i82tfcqwfMVGTPYYgaDnIQgLeyZ
QvrjPdeHD5yCqvXbAUAzJs7Cd7eFxD+8jXO9fXQrWt7NGrkiPI/gPjfXI3tHWOONN9OAwM9CWDsmNi3ha
qrpy9Gw/AfaQbXoa+wpvwCAyYOBexUbFSs1lX7F2MS95XcWDLIQCAto42in+xlp1VCUvYVbJFwvsERL2Ohpa2F
txH/okrZ67jbdRbGBobolPv9jCzNIPPsYv5fIWKc3jTYXQd3hWvQ18j6mUU3Ma54W3UhWv1W3UW1w//Xn9L8/dnrjufR3
HtmW3DB3acAhjl4xFOIMgPPV7ivZ92ONTWxNn952VOnZRm6KoXLsypveanuO8JcuVhNceL9y7dA+HNhyStPxlZ
WYJdoziqpWbMWHicDwPDsOLsJeYNnOsIiKicOzYGUmdEyd24uix24t9L58DVtR26uPZHUlKyZGxefHw
CPn7M/tHl5tYZT548R2zsW9SuXQML/5qBlSs2ISgoJP8vUslSUj4g/NXnpa1ev4nCk2fBMDTQR0EL7ePxVbO5kzCHiEwOtBlsgh53fzp5X0/ClPsqfSoO6jX0MP5YcW6PlEA58fM26WQwQQrC9PVsLHXDxrTBPdTfHwbWmC4ZQCDmSB3R6A1ViCwdhDZyD9/lKDVWK2tt0WKXB4y/KEJ0tGRZh4Ccp0F6OP12iRT0zjwwxMTAeFHZRBdt79atNr2/Dd7ZPfpMTVNBHVeMhH2Chm8eTIcdLw+FAmCdCxODAD7Ok1O8J2I/rudR5NIWJgNr4ddqFNlJAHhFRyJtnljP58rKs2XLYf6NMIsBW0OJOw4YAL23zPhY0jjuDyS2aE8OPNtPVVF40Ke03hkAIMp7DHQqJ+dVFJ4tlrvqL97E8FPtVKvj/cMOHs9ebDDzRvkr7C/hKHRvkHGn/EudVKmPwlQfLgsKlUmfpSCEWQ5EXvHEWTfw2xTqgTXb2a9dMoUhbmufSQ2+VX4jkBTmc6fXTO3xxG1ms/pzfrUcFbtuPG/KxfOG2KamscfTFBYp/gZqY5Ae6Y7dWvWs7l6+UMD5aO5/vWOVmvJkaBuw8Fn64SxeY5MYhYIPa0xgDFUiC1P7NUUNi1wkbAQG/NsyEGcdy9Zv2m3jbuj6t+EAB+GEXFBvbeHSh3cN40RxFHCmXljVSpUi2j2lSg5F5VJxCx/bv/Iim1HyBoL+KcFShFP/p+ayW7bWkEkfSxtLD6zfbFOvnftB+vsuJuhgTjuAfYjjNiOcdXwL1J4R94XpqzQy1B7EWmHnEeh3WXi0CYuVIsMnIZ9qOiCVi2gwsxa/k0bl+Uj6DE5v59rljSimYCkHrAU+NvMC9lx1Pq3vsv3v1WuUxpSDpMZQMdAwiFHrq5u9Gm7CtZbgZYpYKMkpiGFs7XgP/t4j5VZsmzx+dxr3N/OhhZ6gA0oijEmd1p2Y7+GwMBoFH+yD7kTRW6Tn7OqMIhSGBlA9E5OzK0Z8ZJxX5g2h4hWm0kmbUoqb9e/PeDWk1cMRT5dS0q42Ubc0sq5Yl/D+4MvKSL2/F5gdoBGeZBDCOhy0d/d/sOSdZsPWWEBVF9hwZJcdUn5e9ZSSQMI2sbD+zE1uRHFEYrDiZ+8wbDJvFHPI5aQXOjCxOyLZYyIh5vPVMQ9X2LsY6V6s7D+1HfezhR/sHFhbhHEMqwFwdK9fdD+F5C9f9SUw76P9uL8zbzbXXlUfENUS0d+f+C2lB6IxDnR0z4Mgx5UA5wbzlTEdyXJZQsmyWH2Ge4wPWr6XQeaDPfBiaugmpuVEqwg+WG3YwLJnTMIv0oPIJ6cwVDqoYhwbhsdBmmd6Qd1NlpBYPo/uZ6qkn6eKvRtkXK9/Grw8YHiqhmVE4h8PZgwfzTwS3++LNxb8/Qd+3SZkPsgi1d3o2h7O72ov+e1F5/uL6YhX29HBpC/ymnDpX8PqnYBOlDl/S9zNtqeY2ZcWPLj7ybQ64+Rr9bxnL4wRs2vSgFu4lG2a9aVzZ2UMNCUQm0dEJ+XhREC8OrQLt8vUJa3+q+S3j7KrdsUtp7RzM2ZMeZ6WmVq1+Hb5wPbhvyt+EFSx9Q0vqiS6lYHe7ZiBvrY0pu+W
```

H5MmTkZaWpsTICr4PHz4gJSVF8vrFixdYunQpzpw5o8SoCr6UlBT07dsXOjo6qFSpEsLDwwEAw4cPh5eXl5Kjo
4KISR0VeIcOHcLBgwdlbuPHj4empiaTuq8YPXo01NTUEB4eDh0dHUl5ly5d403trcTIhGPgwIF49uwZACAkJAR
du3aFjo409u/fjwkTJig5uoKtXbt22L590wDg/fv3qF27NhYvXox27dphzZo1So6u4PLw8IC/vz8uXrwILSOtS
bmTkxP27t2rxMiowBITCdCTJ0/E7du3F6uqqop79uwpDgsLU3ZIBZqFhYXYz89PLBaLxXp6euLg4GCxWCwWBwc
Hi3V1dZUZmmAYGBiInz9/LhaLxWIvLy9x8+bNxWKxWHz16lVx8eLFlRlagWdiYiJ+9OiRWCwWizds2CCuUqWKO
DMzU7xv3z5x+fLllRxdwWVtbS2+ceOGWCyW/t4GBQWJ9fX1lRkaFVBsqSNBefPmDfr37w97e3tkZGTAz88P27Z
tQ8mSJZUdWoGWnJws1UL3SVxcHJ8j+Y3EYjGysrIAAOfOnUPLliOBACVKlEBsbKwyQyvwUlJSoK+vDwA4c+YM/
vjjD6ioqKBOnTp48eKFkqMruGJiYmBubp6jPDk5GSKRSAkRUUHHpI4EIT4+HhMnToStrSOeP34MHx8fHDt2DJU
rV1Z2aILQoEEDSfcXAIhEImRlZWHhwoVoOqSJEiMTDgcHB8ydOxc7duzApUuXOKpVKwBAaGgoLCwslBxdwWZra
4vDhw/j5cuXOH36NJo3bw4AiI6OhoGBgZKjK7gcHBxw4sQJyetPidzGjRvh6OiorLCoAONAJCrwFi5ciAULFsD
SOhK7d+9Gu3btlB2S4CxcuBDNmjXD3bt3kZaWhgkTJuDx48eIi4vDtWvXlB2eICxduhQ9evTA4cOHMWXKFNja2
gIADhw4gLp16yo5uoJt+vTp6N6900aPHo1mzZpJEpIzZ86gevXqSo6u4Jo/fz5atGiBgIAAZGRkYNmyZQgICMD
169dx6dIlZYdHBZBIL0aaBlSwqaioQFtbG050TlBVVc213sGDB/MxKuGJj4/HypUr4e/vj6SkJNSoUQNDhw5FO
aJFlR1agZeZmYlr167B3t4eRkZGUu99/PgRqqqqUFdXV1JOwhAZGYmIiAhUrVoVKirZnUS3b9+GgYEBypcvr+T
oCq7g4GB4eXlJfW8nTpwIe3t7ZYdGBRCTOirw3N3dv2n8yJYtW/IhGvhGvpVaWlpITAwEKVKlVJ2KIKSnp40bW1t+
Pn5cbgEUR5j9ysVeFu3blV2CIXCx48f8eDBAORHROsG/H/Stm1bJUUlHJUrVOZISAiTuu+krq4Oa2trZGZmKjs
UwUlISJBZLhKJoKmpCQ0NjXyOiAo6ttRRoRAdHS1zlhhl8/b2Rs+ePWXOOhSJRPyD+w28vb3h4eGBOXPmoGbNm
tDV1ZV6nwP+c7dpOyYcPHgQO3bsgLGxsbLDEQwVFRW5vRTFixeHu7s7ZsyYIenSpl8bkzoq8HROdPDixQuYmZk
BAFq1aoWNGzdKxoJFRUXBysqKiYkcdnZ2aN68OaZPn86Zmj/oyz+aX/6hFYvFTIy/onr16nj+/DnSO9NRsmTJH
Amxr6+vkiIr2LZv344pU6bA3d0dtWrVApA9DnHbtm2YON0nUqYmJisGjRIowfPx6TJO9WcrRUELD7lQq8jx8/Sj2
j9PLly5Jnl37C3ybyRUVFYcyYMUzofsKFCxeUHYJgtW/fXtkhCNK2bduwePFiuLq6SsratGkDe3t7rFu3Dj4+P
rC2tsa8efOY1BEAttSRAKioqCAyMlLSvaqvrw9/f3+3+UL0aAFvqvkWfPn1Qr149903bV9mhENE30tbWxoMHD2B
nZydVHhQUhKpVqyIlJQWhoaGoVKmS1LN16dfFljqiX8DKlSvRuXXNnXLlyBfb29jmW3xgxYoSSIhOelJQUhIeHI
y0tTaq8SpUqSopIGN6/f48DBw4gODgY48ePh7GxMXx9fWFhYYFixYop07wCqUSJEti0aRO8vLykyjdt2oQVJEti0aRO8vLykyjdt2oQSJUo
AAN6+fZtjmR36dTGpowJPJBJJjWH672vK3dq1a7G2tv6udXbv3q1Ufxt+PUqVMy32dLce4eP
HgAJycnGBoaIiwsDP3794exsTEOHjyI8PBwqaed0GeLFi1C586586dcerUKfz2228AgLt37yIwMBD//vsvAODOnTv
o0qWLMsOkAoTdr1TgqaiowNDQUJKIvH//HoYGBpKB62KxGAkJCUySfoK+vj78/f3LQzhFS1FfNOXPWBATVvHYtXUJIIlvWFOKLfkmvJ0DTB0dX6BmEJAjDXT57Oa+HbHByfpRgnUb8q1pcRpxS/f4Ie4/n+FsrCnCUYVh4iTdiM3sJCeg0D7f8pzdbL+kRUPMnD5mZbp9qDGLgPFvuMSLSpQfKZmGCUD6kSxVRjpuH+J/2HrW2Xkxmr0DJ6pEhjWZ+lhYbfaG7/O8MpE4uDi9T0yEsWQn8Uw+n7NXOsugmSXiUmYWKWDyQxnlyMOEAXdFn4j+ZtFvZ+mGWl2nakWIZVUS2Pga8kRHaSdUuHEeh9rUEfn0J/HAl8CA90eg3xnmK6kP7YjVqD9iEuYjgYA2U1WPGvB2h4h2c8Q8YHzPGJcLx4YKyaDkH9a6Nxz6J2A+aLpfjTfK6VQ18aaaBZNh9+MXPNAiKmiEXp4WKjp+qzhk+ZbG3FSiCLAVIdlsmrq+4XLKF44fWuuE78WM/4oDtzImW7F1hU8zvFchxdcGpExLLzVvchbk3ufbjjbl+3Z4b+7S6q8YZjLTFlOrIfHOg6+pwrdpp6Nv/YdDdrRnIZXA9G7UnemVJHpoBX5F9uGy2XaG2e7N23JQsFD7lcuiDcTHjWPVW8eJqdFJbrEAeSj8nHBFmTyNvQT4P6JlDOeGA+Ydf3W3SRVyuI1H4kVbF2/3xRXsgA9AXFFwNZwtcB2ZXtGaeIbMyQrWhoxD9v2fdh5dQ9AZLjdyk6tY0KsfFjPIFzMBH/qhX6N83CuQ5z8n4V/ApxwmMkDOEPbRzNaLxJ7SjU8JEtD/f3AgZkNLf/MgkxX5nFYVBNDeBzcBtjqjTqPKyf7iQHwvJhy4lCMVzLgR+UIyKAXiWMGoDKm53eGbC7ugkQYqZ3CS8qNGoUmnWy4JVyYWJGOo2IGHKVUIDFH+m5Tu8hl83h0fH+8jIhOXO8g7tTmwkXYWQL8RNyo7N0zTiEH8RCY6pWeTd8UEqN6MA0tRxY3eEeoF3PLQhpBNrWnESqwX70EgW8I+3UDojYO0EAzd9E2Q+SwD6zOQxjJeNiCH1bnVFnb+MXYtdI0b/84h9kbxRmXWxCVqkEv08oRLFxRb7DGERPphLg78B8e2Df/KrAIdFb7c2e7IBQ9bQuS8IE2c4V3i1BBzhkJMDqbzyM8R7wqA2Me4=

fLlkv+uWLEi5s2bh4sXLOo+fzdv3sS1a9cwduxYZYVIVOiw+5UE4/z58zh48CDCwsIgEolQqlQpdOrUCQ0bNlR
2aILw+vVrrF69Gk+ePAEAVKhQAUOGDIGVlZWSIxOO4OBgeHl5wd/fX7LW2sSJE2Fvb6/sOAqcUqVKfVM9kUiEk
JCQPI6G6NfApI4EYdCgQVi/fj2MjIxQtmxZiMViBAUF4f379xgyZAhWrFih7BCJiIiUimPqqMA7dOgQtmzZgs2
bNyM2NhY3btzAzZs3ERMTgwObNmD9+vWcBfYVW7Zswf79+3OU79+/H9u2bVNCRMLTqFEjbN++nZNKiKjAYksdF
Xht27ZFpUqV4OnpKfP9iRMn4smTJzhy5Eg+RyYcZcuWxbp163I8A/bSpUsYMGAAnj59qqTIhGPUqFHYtWsXUlN
T4erqir59+6JOnTrKDksQxGIxDhw4gAsXLiA6OjrHgtcHDx5UUmREhQtb6qjA8/X1RYcOHXJ9/48/fyM
SLhCQ8PlznGqWTJkpInI5B8S5cuxZs3b7BlyxEROejYcOGqFixIhYtWoSoqChlh1egjRo1Cm5ubggNDYWenh4
MDQ2lNiJSDLbUUYGnpaWFkJCQXAfOv379Gra2tuwWk8Pa2horV66UufjwOKFDJc+DpW8XHR2N9evXY968ecjMz
ETLliOxYsQING3aVNmhFTjGxsb4559/OLJlS2WHQlSosaWOCryOtDSoq6vn+r6amhrSOtLyMSLh6datGOaMGIE
LFy4gMzMTmZmZOH/+PEaOHImuXbsqOzzBuX37NmbMmIHFixfD3NwcHh4eMDU1RevWrTFFu3DhlhlfgGBoa8tF+R
PmALXVU4KmoqGDAgAHQOdGR+X5KSgo2bNjABYjlSEtLg5ubG/bv3w81tezlKbOystCzCzZO+sXbsWGhoaSo6w4Iu
OjsaOHTuwZcsWBAUFoU2bNujXrx+cnZOlCxFfvXoVLi4ufELCf2zbtg3e3t7YvHkztLW11ROOUaHFpI4KvMaNG
Ov+aMpz4cKFfIhG2J49ewZ/f39oa2vD3t4eJUuWVHZIgqGhoYEyZcqgT58+cHd3h5mZWY46CQkJaNeuHT+L//H
hwwdO6NAB165dg42NTY6Wd19fXyVFRlS4MKkjIvoGV65cQb5cQYMGDb5a79q1a3BwcICmpmY+RCUMrq6uuHDhAjp16
gQLC4scP9JmziHihpMiIChcmdSR4ISEhGDRoEM6cOaPsUAqszMxMbN26FT4+PjKXlDh//rySIiIlYGBgo
OwwiAo9JnVEv4A9e/Zg3759XFKHCl1q5dCxsbG2WHQ1RoMakj+gVoaGjA1tZW2WHQL+rPP/9ESko
KypQpAx0dnRwTJTeLi4pQUGVHhwqSO6BcwduxYLFu2DCtXrmTXmTXK+W7pUuXKjsEol8Ckzoq8Kpzoq8KpXry43EUlJScnHa
ITp6tWruHDhAk6dOoVKlSrlaCnhszcVh0lzTr169VJ2CES/bCZ1VOC1b99e2SEIXpEiReQ+P5cUhxMDsa
WLVsQHByMZcuWwdzcHKdOnYK1tTUqvaqk7PCICgUuaUJERHnq0qVLaNGiBerVq4fLly/z8fF369+7o3Tp0rh9+zb++OMPMPvHr1StkhEhUK7H6
lAu9b/7Dyj6qOGjVqwMfHBOZGRl+9h7x3sn3Z9f/x40esXr0aFStWhKOjIwDg5s2bePz4MYYYMGaKkCIXh4cOH2
LVrV45yc3NzxMbGcHKdOnYK1tTUqvaqk7PCQqVLaNGiBerVq4fLly/z8fF369+7o3Tp0rh9+zb++OMPMPvHr1StkhEhUK7H6
ERXM5NDO8uMCSlMCfVyGDqE4FDxqFFGgdqfhjTLMDlRT1IcOdk+4QeCfL2LM83cuJmJFEu+MvkAXnEFLmDnoczqZFHJhdW
DzFCOWHUrYJmwE2Cs+bAHhIZLWjxUu08+4Yzb+JhvELQrW+qpWqa1ff7ZrxRuEsDD8iqKDiZ5YG4+vO4/HGtFtE5RqJHwLpNGQUvXrpQWpDtfGpQTaAMZl

[illegible]

```
e3loH1F+/fh1OTk6oq6sTmFS6WNQRvRwWdUREOnB1dUVQUBACAgKa3JhTU1ODpKQkKBQKPaczDnK5HGfOnGFRR
/QfsagjMgFr1qxp9vnKlSv1lISoaRypI3o5LOqITICHh4fWdW1tLQoLC2Fubo5evXpBrVYLSiYt5eXlSEhIwB9
//AEA6NevH4KCgmBtbSO4mWHLzMzEuHHj/vW97OxsDBs2DJaWlnpIRWR8WNQRmaiKigoEBARg6tSp8Pf3Fx3H4
OXm5mLChAmwsrLC8OHDAQC//fYbHj16hB9++AFDhgwRnNBwWVpa4vXXXOdgYCAUCgUcHBxERyIySizqiEzY+fP
n4ePjg6KiItFRDJ6npydcXFywbds2mJs/O4ynrq4OISEhKCgoYL+/Zty5cwcqlQo7d+7E77//Di8vLwQHB2PKl
CnNnglLRC+GRR2RCcvOzoaPjw/u3bsnOorBs7KywunTp+Hu7q51/+LFixg6dCgePnwoKJmOqNVqJCYmIikpCQA
wa9YsBAcHY9CgQYKTEUkfz34lMgGxsbFa1xqNBqWlpVCpVJg0aZKgVNIil8tRU1LSoKi7du0aXn31VUGppGfIk
CGws7NDp06dEB0dDaVSic2bN2PEiBGIj49Hv379Eckiy01BGZgOebM5uZmaFz587w8vLCp59+yqJEB2FhYdi
3bx/Wr1+PkSNHAgBycnIQHh60adOmYePGjWIDGrja2lqkpaVBqVTi6NGjGDp0OKIKDg/Hhhx/i9u3biIyMhFqtx
sWLFOVHJZIsFnVERDqoqalBeHg44uPj65sLW1hYIDQ0FNHR0dyx2YyPP/4YSUlJ0Gg08Pf3R0hICPr376/1zs2
bN2Fvb4+nT58KSkkkfSzqiIhewMOHD3H161lUAQK9evdC+fXvBiQzfO++8g5CQEPj5+TVZ/NbV1SEnJwdjxozRc
zoi48GijsgEVFVVITo6Gj/++CNu3brVYDSkoKBAUDIiImop3ChBZAJCQkLw0O8/wd/fH926dYNMJhMdSRL8/Px
Ofjc1NbUVk0jP/v37dX7X19e3FZMQmQ4WdUQm4NChQzhw4ABGjRolOoqk8KSI/27KlCk6vSeTyfDkyZPWDUNkI
ljUEZkAGxsb2Nraio4hOYmJiaIjSBY3PBDpH9fUEZmA3bt3Iy0tDTt37uTC/pd069Yt5OfnAwDc3NzQpUsXwYm
IiJ5hUUdkAjw8PHD161VoNBo4OTnBwsJC67larRaUTDoqKiqwYMECJCcn1O8XtmnTBjNnzkRcXBynap8TGxuBj
z76CO3atWvQ/Pp5YWFhekpFZNxY1BGZgNWrVzf7fFWqVXpKIl0zZ87E6E6dOn8fXXX2PEiBEAgBMnTmDRokUYPHg
wkpOTBSc0LD179kRubi46derUoPn1P8lkMu6+kRubi46derUoPn1P8lkMu6+JmohLOqIiHTQoUMHHDlyBKNHj9a6/
//PPP2PixImoqqoSlIyI6
BluICAyITU1NY32qXN0dBSUSDo6derU6BSrtbU1bGxsBCQiItLGkToiE3D58mUEBwfjl19+0bqvnUCfn5+mDdvnuCEhkujOWDv3r3IzMxs9EcFe/wRtQwWdUQmYNSoUTA3N8fy5cbb
T48aNAgQcmkw8PDA1euXMHjx4/rRzZLSkpgaWkWLsGXLFowbNw5du3Zt8PePrWOIWganX4l
MwJkzZ5CXlwd3d3fRUSRL12a61JBKpUJqaiq8vb1FRyEyaizqiExA3759fOHdExJI07hP87a2trODs7i45BZ
PQ4/UpkAjIyMhAZGYmoqCgMGDCgQZ86uVwuKJk0VVZWNlgXxm/YtJ07d+Lw4cNQKpWwsrISHYfIaLGoIzIBZmZ
mANBgLRM3SuiusLAQCxcuxPHjx1FdXV1/n9/w3z169Ah    Tp05FTk4Om18TtSJOvxKZgMzMzCafnT9/Xo9JpGvOn
DnQaDRQKpWNLvanpikUCuTl5WHOnDn8dkStiCN1RCbowYMHSEpKwTbWsZrbT25GXl8dRJh288soryMvLg5ubm+goktN
U42Yia1lmogMQkf5kZWVBoVCgW7duWL9Pby8vHDy5EnRsSRh2LBhuHbtmugYkuTg4MA1hOR6wOlXIiN38+ZN7
NixAwkJCaioqMCMGTPw+PFjfP/99+jbt6oeJKxfft2zJ8/Hzdu3ED//v0brAsbOHCgoGSG76uvvsKyZcsQHx8
PJycnOXGIjBanX4mmI+PD7KysvD+++9uzZmDhxItq0aQMLCwucPXuWRdOLOHnyJGbNNmoWioqL6ezKZjBsld
GBjY4OHDx+irq407du3b1AQl5WVCUpGwFw4Ukdkx A4dOoSwsDCEhoY20OPWAXkxQUBA8PDyQlJTExf4va0OPGjaI
jEJkEFnVERiw7OxsJCQl44403KdPH/j7++ODDz4QHUuSiouLsX//fri4uIiOIjkKhUJOBCKTwIOSREbsrbfew
rZt21BaWop58+YhOTkZ9vb2ePr0KY4ePYoHDx6IjigZXl5eOHv2rOgYkl ddXY2KigqtPOTUMrimjsjE5OfnIyE
hASqVCuXl5Xj33Xexf/9+ObEM3tatW/HFF18gKCio0VM5fH19BSUzfFVVVYiIiEBKSgru3r3b4DnXIxK1DBZ1R
CbqyZMnSE9Ph1KpZFGng/+fytEYbpRo3oIFC5CZmYnPP/8c/v7+iIuLw40bN7BlyxZER0dj9uzZoiMSGQUWdUR
E1KocHR2xa9cujBO7FnK5HGq1Gi4uL1CpVEhKSsLBgwdFRyQyClxTRO7OTUDG9vb9y/f7/+Ojo6GuXl5fXXd+/eZ
WuYf1FWVgZnZ2cAgFwur29hMnr0aGRlZYmMRmRUWNQRETXjyjEjePz4cf11VFSUVl+1uro65Ofni4gmGc7Ozig
sLAQAuLu7IyUlBQCQnp6Ojh07Ckx GZFxY1BERNePSpcsaK7go
ICPH36FIGBgfU7h5cvX464uDiOa9cOixcvR
nh4uOCURMaDfeqIiKhVuLq6orSOFIsXLwYAzJw5E7Gxsbh06RLy8vLg4uLC49WIWhBH6oiImiGTyRqcHsHTJHT
z/KjmwYMHUVVVhR49esDPz48FHVEL4OgdEVEzNBoNAgICYGlpCeBZ89z58+ejQ4cOAKC13o6iSCQWdUR
Ks5c+YOeGfu3Ln6iiMpHOUk0i/2qSMiolZhZmaGSZMm1Y9ypqenw8fHp77Qq6mpwaFDh/DkyRORMYnIUiBEKSgru3r3b4DnXIxK1DBZ1R
RtRyO1BEREREZAe5+JSIiIjICLOqIiIiIjACLOiIiIiIjwKKOiIiIyAiwqCMiIiIyAizqiIiIiIwAizoiIiIiI
/A/5mvTh1EczGMAAAAASUVORK5CYII=\n"
          },
          "metadata": {}
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "Correlation Matrix Interpretation\n",
        "\n",
        "The correlation matrix shows the correlation coefficients between sets of
variables. Each cell in the table shows the correlation between two variables. A
correlation of 1 implies a perfect positive correlation, while a correlation of -1
implies a perfect negative correlation. A correlation of zero means that there is no
relationship between the different variables.\n",
        "\n",
        "Here are some key observations from the matrix:\n",
        "\n",
        "- **CHILDREN and Family_Members**: A high positive correlation of 0.872259
suggests a strong relationship. As the number of children increases, the number of
family members also tends to increase.\n",
        "\n",
        "- **CHILDREN and Age**: A negative correlation of -0.320143 suggests that as
the age increases, the number of children tends to decrease.\n",
        "\n",
        "- **Employed_days and Age**: A positive correlation of 0.336351 suggests that
as the age increases, the number of employed days also tends to increase.\n",
        "\n",
        "- **Employed_days and Family_Members**: A negative correlation of -0.211551
suggests that as the number of family members increases, the number of employed days
tends to decrease.\n",
        "\n",
        "Other correlations are relatively weak and may not indicate a strong
relationship.\n"
      ],
      "metadata": {
        "id": "PphfREB-92b9"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "## Hypothesis Testing"
      ],
      "metadata": {
        "id": "_pcjzPXzkzV9"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "### Numerical Columns Vs. Target Variable"
      ],
      "metadata": {
        "id": "XO77fNeTty6N"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "#### T-Tests\n",
        "\n",
        "The t-test is a statistical hypothesis test where the test statistic follows
a Student's t-distribution if the null hypothesis is supported. It is used to
determine whether there is a significant difference between the means of two groups."
      ],
      "metadata": {
        "id": "OD69aKm68wWH"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "Before we perform the tests, we first declare our hypotheses:\n",
        "\n",
        "- **Null Hypothesis (H0)**: There is no significant difference between the
groups being compared.\n",
        "- **Alternative Hypothesis (H1)**: There is a significant difference between
the groups being compared.\n",
        "\n",
        "Now, we perform the tests and interpret the t-statistics and p-values:"
      ],
      "metadata": {
        "id": "NiF-2L2a9GM_"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "import scipy.stats as stats\n",
        "# t test for numerical columns\n",
        "numerical_columns =
[\"CHILDREN\",\"Annual_income\",\"Employed_years\",\"Family_Members\",\"Age\"]\n",
        "for column in numerical_columns:\n",
        "  approved = merged_data[merged_data[\"label\"]==0][column]\n",
        "  not_approved = merged_data[merged_data[\"label\"]==1][column]\n",
        "  t_statistic, p_value = stats.ttest_ind(approved, not_approved)\n",
        "  print(f\"t-statistic for {column}: {t_statistic:.4f}, p-value:
{p_value:.4f}\")\n"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "nNqQlXg7gzZn",
        "outputId": "5139cf3d-1563-4389-ba4e-91f5a09fc73d"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
            "t-statistic for CHILDREN: 0.8745, p-value: 0.3820\n",
            "t-statistic for Annual_income: -0.8394, p-value: 0.4014\n",
            "t-statistic for Employed_years: 3.8627, p-value: 0.0001\n",
            "t-statistic for Family_Members: 1.2452, p-value: 0.2132\n",
            "t-statistic for Age: -1.7722, p-value: 0.0766\n"
          ]
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "T-Test Results\n",
        "\n",
        "Here are the results of our t-tests:\n",
        "\n",
        "- **CHILDREN**: T-statistic: 0.8745, P-value: 0.3820. No significant
difference (p > 0.05).\n",
        "- **Annual_income**: T-statistic: -0.8394, P-value: 0.4014. No significant
difference (p > 0.05).\n",
        "- **Employed_days**: T-statistic: -3.2458, P-value: 0.0012. Significant
difference (p < 0.05).\n",
        "- **Family_Members**: T-statistic: 1.2452, P-value: 0.2132. No significant
difference (p > 0.05).\n",
        "- **Age**: T-statistic: -1.7722, P-value: 0.0766. No significant difference
(p > 0.05).\n"
      ],
      "metadata": {
        "id": "D9dEasKQ78Qs"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "### Categorical Columns Vs. Target Variable"
      ],
      "metadata": {
        "id": "ARsVe1hsteeP"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "#### Chi-square Tests"
      ],
      "metadata": {
        "id": "csRQxiUY8sQo"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "Chi-square (蝶虏) is a statistical test commonly used to determine whether
there is a significant association between two categorical variables. It's
particularly useful for analyzing data that's in the form of frequencies or counts
across different categories."
      ],
      "metadata": {
        "id": "vbqLLA378qoz"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "Before we perform the tests, we first declare our hypotheses:\n",
        "\n",
        "- **Null Hypothesis (H0)**: There is no significant difference between the
groups being compared.\n",
        "- **Alternative Hypothesis (H1)**: There is a significant difference between
the groups being compared.\n",
        "\n",
        "Now, we perform the tests and interpret the p-values:"
      ],
      "metadata": {
        "id": "EV4AKFHR8nhx"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "import pandas as pd\n",
        "from scipy.stats import chi2_contingency\n",
        "\n",
        "\n",
        "categorical_columns = ['GENDER', 'Car_Owner', 'Propert_Owner', 'Type_Income',
'EDUCATION',\n",
        "                       'Marital_status', 'Housing_type',
'Type_Occupation']\n",
        "\n",
        "chi_square_p_values = []\n",
        "\n",
        "for feature in categorical_columns:\n",
        "    contingency_table = pd.crosstab(merged_data[feature],
merged_data['label'])\n",
        "    chi2, p, dof, expected = chi2_contingency(contingency_table)\n",
        "    chi_square_p_values.append((feature, p))\n",
        "p_values_df = pd.DataFrame(chi_square_p_values, columns=['Feature', 'P-
value'])\n",
        "print(p_values_df.sort_values(by='P-value'))"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "1MFqAKWFtsB7",
        "outputId": "ad2556fc-ced5-4a70-bc5a-d7def8664d77"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
            "           Feature   P-value\n",
            "6     Housing_type  0.000036\n",
            "3      Type_Income  0.003506\n",
            "7  Type_Occupation  0.005120\n",
            "5   Marital_status  0.032467\n",
            "4        EDUCATION  0.056159\n",
            "0           GENDER  0.086645\n",
            "2    Propert_Owner  0.535110\n",
            "1        Car_Owner  0.618569\n"
          ]
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "P-Value Results\n",
        "\n",
        "- **Housing_type**: P-value: 0.000036. Significant difference (p < 0.05).\n",
        "- **Type_Income**: P-value: 0.003506. Significant difference (p < 0.05).\n",
        "- **Type_Occupation**: P-value: 0.003852. Significant difference (p <
0.05).\n",
        "- **Marital_status**: P-value: 0.032467. Significant difference (p <
0.05).\n",
        "- **EDUCATION**: P-value: 0.056159. No significant difference (p > 0.05).\n",
        "- **GENDER**: P-value: 0.086645. No significant difference (p > 0.05).\n",
        "- **Propert_Owner**: P-value: 0.535110. No significant difference (p >
0.05).\n",
        "- **Car_Owner**: P-value: 0.618569. No significant difference (p > 0.05).\n"
      ],
      "metadata": {
        "id": "2KiAev3R8R7k"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "#### Work_Phone Vs. Phone"
      ],
      "metadata": {
        "id": "Ghn8o9aYk2qn"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "import pandas as pd\n",
        "from scipy import stats\n",
        "crosstab = pd.crosstab(merged_data[\"Work_Phone\"],
merged_data[\"Phone\"])\n",
        "chi2, p, dof, expected = stats.chi2_contingency(crosstab)\n",
        "\n",
        "# Print the results\n",
        "print(\"Chi-square statistic:\", chi2)\n",
        "print(\"P-value:\", p)\n",
        "print(\"Degrees of freedom:\", dof)\n",
        "print(\"Expected values:\", expected)"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "lkD801nLjEXr",
        "outputId": "e4401615-cee5-4347-d603-a09f95bcf342"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
            "Chi-square statistic: 190.40864054667088\n",
            "P-value: 2.589012403796421e-43\n",
            "Degrees of freedom: 1\n",
            "Expected values: [[846.6369509 379.3630491]\n",
            " [222.3630491  99.6369509]]\n"
          ]
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "**P-value**: The p-value is approximately **2.59 脑 10 鋆?3**\n",
        ". This is far below the common significance level of **0.05**. Therefore, we
**reject the null hypothesis** that the variables are independent. There is a
statistically significant relationship between **work phone** and **phone**.\n"
      ],
      "metadata": {
        "id": "SjHDRDwWjtnm"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "Since Work_Phone and Phone columns are highly correlated one of them can be
dropped"
      ],
      "metadata": {
        "id": "pYL_WJuN4_ES"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "#dropping work phone\n",
        "merged_data = merged_data.drop(\"Work_Phone\",axis = 1)"
      ],
      "metadata": {
        "id": "eHJO7emkjqsQ"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "psko7ALOm3tG"
      },
      "outputs": [],
      "source": [
        "# selected_data = merged_data.drop(['EDUCATION', 'Car_Owner',
'Propert_Owner','Marital_status'], axis=1)\n",
        "selected_data=merged_data.copy()"
      ]
```

```
},
{
  "cell_type": "markdown",
  "metadata": {
    "id": "syMPJstIYzRj"
  },
  "source": [
    "## Encoding"
  ]
},
{
  "cell_type": "markdown",
  "metadata": {
    "id": "8nhADnikK3z_"
  },
  "source": [
    "### Label Encoding"
  ]
},
{
  "cell_type": "code",
  "execution_count": null,
  "metadata": {
    "id": "F3cfdg_pYxZO"
  },
  "outputs": [],
  "source": [
    "#change\n",
    "encoded_data = selected_data.copy()\n",
    "from sklearn.preprocessing import LabelEncoder\n",
    "encoder = LabelEncoder()\n",
    "binary_categorical_columns = [\"GENDER\",\"Car_Owner\",\"Propert_Owner\"]\n",
    "for column in binary_categorical_columns:\n",
    "  encoded_data[column] = encoder.fit_transform(encoded_data[column])"
  ]
},
{
  "cell_type": "markdown",
  "metadata": {
    "id": "w9FstWMIKkkl"
  },
  "source": [
    "### Ordinal Encoding"
  ]
},
{
  "cell_type": "code",
  "execution_count": null,
  "metadata": {
        "id": "dknqgUHbbliE"
      },
      "outputs": [],
      "source": [
        "# columns with more than 2 categories having inherent order (ordinal
encoding)\n",
        "from sklearn.preprocessing import OrdinalEncoder\n",
        "ordinal_encoder = OrdinalEncoder(categories=[['Lower secondary', 'Secondary /
secondary special', 'Incomplete higher', 'Higher education', 'Academic degree']])\n",
        "encoded_data[\"EDUCATION\"] =
ordinal_encoder.fit_transform(encoded_data[[\"EDUCATION\"]])"
      ]
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "-lb8hkDFKuUn"
      },
      "source": [
        "### One Hot Encoding"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "iKH7zaVgX3vg"
      },
      "outputs": [],
      "source": [
        "# change\n",
        "# columns with more than 2 categories (one hot encoding)\n",
        "columns =
[\"Type_Income\",\"Housing_type\",\"Type_Occupation\",\"Marital_status\"]\n",
        "for column in columns:\n",
        "  dummies = pd.get_dummies(encoded_data[column])\n",
        "  encoded_data = encoded_data.drop(column, axis = 1)\n",
        "  encoded_data = pd.concat([encoded_data,dummies],axis = 1)"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "demAsFUD6aco"
      },
      "outputs": [],
      "source": [
        "# from category_encoders import TargetEncoder\n",
        "\n",
        "# encoder = TargetEncoder()\n",
        "# encoded_data[\"Type_Occupation\"] =
encoder.fit_transform(encoded_data[\"Type_Occupation\"],encoded_data[\"label\"])\n"
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "encoded_data.head()"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 290
        },
        "id": "a5d_sFLw7YIa",
        "outputId": "fd6e1cd3-e61a-4c47-ba7c-2cf384ed0b6f"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "   GENDER  Car_Owner  Propert_Owner  CHILDREN  Annual_income  EDUCATION
\\\n",
              "0       1          1              1       0.0      12.100712        3.0
\n",
              "1       0          1              0       0.0      12.660328        3.0
\n",
              "2       0          1              0       0.0      12.660328        3.0
\n",
              "3       0          1              0       0.0      12.022751        3.0
\n",
              "4       0          1              0       0.0      12.660328        3.0
\n",
              "\n",
              "   Phone  EMAIL_ID  Family_Members  label  ...  Secretaries  Security
staff  \\\n",
              "0      0         0             2.0      1  ...            0
0   \n",
              "1      1         0             2.0      1  ...            0
0   \n",
              "2      1         0             2.0      1  ...            0
0   \n",
              "3      1         0             2.0      1  ...            0
0   \n",
              "4      1         0              2.0      1  ...            0
0   \n",
              "\n",
              "   Unemployed  Unknown  Waiters/barmen staff  Civil marriage  Married  
\\\n",
              "0           1        0                     0               0        1  
\n",
              "1           0        1                     0               0        1  
\n",
              "2           0        1                     0               0        1  
\n",
              "3           0        1                     0               0        1  
\n",
              "4           0        1                     0               0        1  
\n",
              "\n",
              "   Separated  Single / not married  Widow  \n",
              "0          0                     0      0  \n",
              "1          0                     0      0  \n",
              "2          0                     0      0  \n",
              "3          0                     0      0  \n",
              "4          0                     0      0  \n",
              "\n",
              "[5 rows x 47 columns]"
            ],
            "text/html": [
              "\n",
              "  <div id=\"df-69eee42c-2fa7-4c08-a9a3-54f3d2beac30\" class=\"colab-df-
container\">\n",
              "    <div>\n",
              "<style scoped>\n",
              "    .dataframe tbody tr th:only-of-type {\n",
              "        vertical-align: middle;\n",
              "    }\n",
              "\n",
              "    .dataframe tbody tr th {\n",
              "        vertical-align: top;\n",
              "    }\n",
              "\n",
              "    .dataframe thead th {\n",
              "        text-align: right;\n",
              "    }\n",
              "</style>\n",
              "<table border=\"1\" class=\"dataframe\">\n",
              "  <thead>\n",
              "    <tr style=\"text-align: right;\">\n",
              "      <th></th>\n",
              "      <th>GENDER</th>\n",
              "      <th>Car_Owner</th>\n",
″      <th>Propert_Owner</th>\n″,
″      <th>CHILDREN</th>\n″,
″      <th>Annual_income</th>\n″,
″      <th>EDUCATION</th>\n″,
″      <th>Phone</th>\n″,
″      <th>EMAIL_ID</th>\n″,
″      <th>Family_Members</th>\n″,
″      <th>label</th>\n″,
″      <th>...</th>\n″,
″      <th>Secretaries</th>\n″,
″      <th>Security staff</th>\n″,
″      <th>Unemployed</th>\n″,
″      <th>Unknown</th>\n″,
″      <th>Waiters/barmen staff</th>\n″,
″      <th>Civil marriage</th>\n″,
″      <th>Married</th>\n″,
″      <th>Separated</th>\n″,
″      <th>Single / not married</th>\n″,
″      <th>Widow</th>\n″,
″    </tr>\n″,
″  </thead>\n″,
″  <tbody>\n″,
″    <tr>\n″,
″      <th>0</th>\n″,
″      <td>1</td>\n″,
″      <td>1</td>\n″,
″      <td>1</td>\n″,
″      <td>0.0</td>\n″,
″      <td>12.100712</td>\n″,
″      <td>3.0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>2.0</td>\n″,
″      <td>1</td>\n″,
″      <td>...</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>1</th>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>12.660328</td>\n″,
″      <td>3.0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>2.0</td>\n″,
″      <td>1</td>\n″,
″      <td>...</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>2</th>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>12.660328</td>\n″,
″      <td>3.0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>2.0</td>\n″,
″      <td>1</td>\n″,
″      <td>...</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>3</th>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>12.022751</td>\n″,
″      <td>3.0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>2.0</td>\n″,
″      <td>1</td>\n″,
″      <td>...</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>4</th>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>12.660328</td>\n″,
″      <td>3.0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>2.0</td>\n″,
″      <td>1</td>\n″,
″      <td>...</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″    </tr>\n″,
″  </tbody>\n″,
″</table>\n″,
″<p>5 rows 脑 47 columns</p>\n″,
″</div>\n″,
″    <div class=\″colab-df-buttons\″>\n″,
              "\n",
              "  <div class=\"colab-df-container\">\n",
              "    <button class=\"colab-df-convert\"
onclick=\"convertToInteractive('df-69eee42c-2fa7-4c08-a9a3-54f3d2beac30')\"\n",
              "            title=\"Convert this dataframe to an interactive
table.\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\" viewBox=\"0
-960 960 960\">\n",
              "    <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220
220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-
160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",
              "  </svg>\n",
              "    </button>\n",
              "\n",
              "  <style>\n",
              "    .colab-df-container {\n",
              "      display:flex;\n",
              "      gap: 12px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert {\n",
              "      background-color: #E8F0FE;\n",
              "      border: none;\n",
              "      border-radius: 50%;\n",
              "      cursor: pointer;\n",
              "      display: none;\n",
              "      fill: #1967D2;\n",
              "      height: 32px;\n",
              "      padding: 0 0 0 0;\n",
              "      width: 32px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert:hover {\n",
              "      background-color: #E2EBFA;\n",
              "      box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n",
              "      fill: #174EA6;\n",
              "    }\n",
              "\n",
              "    .colab-df-buttons div {\n",
              "      margin-bottom: 4px;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert {\n",
              "      background-color: #3B4455;\n",
              "      fill: #D2E3FC;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert:hover {\n",
              "      background-color: #434B5C;\n",
              "      box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",
              "      filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",
              "      fill: #FFFFFF;\n",
              "    }\n",
              "  </style>\n",
              "\n",
              "    <script>\n",
              "      const buttonEl =\n",
              "        document.querySelector('#df-69eee42c-2fa7-4c08-a9a3-
54f3d2beac30 button.colab-df-convert');\n",
              "      buttonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "\n",
              "      async function convertToInteractive(key) {\n",
              "        const element = document.querySelector('#df-69eee42c-2fa7-4c08-
a9a3-54f3d2beac30');\n",
              "        const dataTable =\n",
              "          await
google.colab.kernel.invokeFunction('convertToInteractive',\n",
              "                                                    [key], {});\n",
              "        if (!dataTable) return;\n",
              "\n",
              "        const docLinkHtml = 'Like what you see? Visit the ' +\n",
              "          '<a target=\"_blank\"
href=https://colab.research.google.com/notebooks/data_table.ipynb>data table
notebook</a>'\n",
              "          + ' to learn more about interactive tables.';\n",
              "        element.innerHTML = '';\n",
              "        dataTable['output_type'] = 'display_data';\n",
              "        await google.colab.output.renderOutput(dataTable, element);\n",
              "        const docLink = document.createElement('div');\n",
              "        docLink.innerHTML = docLinkHtml;\n",
              "        element.appendChild(docLink);\n",
              "      }\n",
              "    </script>\n",
              "  </div>\n",
              "\n",
              "\n",
              "<div id=\"df-adeb9758-7329-4afa-a77e-6f55c079e487\">\n",
              "  <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-
adeb9758-7329-4afa-a77e-6f55c079e487')\"\n",
              "            title=\"Suggest charts\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "<svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0 0
24 24\"\n",
            "     width=\"24px\">\n",
            "    <g>\n",
            "        <path d=\"M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-
.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\"/>\n",
            "    </g>\n",
            "</svg>\n",
            "  </button>\n",
            "\n",
            "<style>\n",
            "  .colab-df-quickchart {\n",
            "      --bg-color: #E8F0FE;\n",
            "      --fill-color: #1967D2;\n",
            "      --hover-bg-color: #E2EBFA;\n",
            "      --hover-fill-color: #174EA6;\n",
            "      --disabled-fill-color: #AAA;\n",
            "      --disabled-bg-color: #DDD;\n",
            "  }\n",
            "\n",
            "  [theme=dark] .colab-df-quickchart {\n",
            "      --bg-color: #3B4455;\n",
            "      --fill-color: #D2E3FC;\n",
            "      --hover-bg-color: #434B5C;\n",
            "      --hover-fill-color: #FFFFFF;\n",
            "      --disabled-bg-color: #3B4455;\n",
            "      --disabled-fill-color: #666;\n",
            "  }\n",
            "\n",
            "  .colab-df-quickchart {\n",
            "    background-color: var(--bg-color);\n",
            "    border: none;\n",
            "    border-radius: 50%;\n",
            "    cursor: pointer;\n",
            "    display: none;\n",
            "    fill: var(--fill-color);\n",
            "    height: 32px;\n",
            "    padding: 0;\n",
            "    width: 32px;\n",
            "  }\n",
            "\n",
            "  .colab-df-quickchart:hover {\n",
            "    background-color: var(--hover-bg-color);\n",
            "    box-shadow: 0 1px 2px rgba(60, 64, 67, 0.3), 0 1px 3px 1px rgba(60,
64, 67, 0.15);\n",
            "    fill: var(--button-hover-fill-color);\n",
            "  }\n",
            "\n",
            "  .colab-df-quickchart-complete:disabled,\n",
            "  .colab-df-quickchart-complete:disabled:hover {\n",
            "    background-color: var(--disabled-bg-color);\n",
"    fill: var(--disabled-fill-color);\n",
"    box-shadow: none;\n",
"  }\n",
"\n",
"  .colab-df-spinner {\n",
"    border: 2px solid var(--fill-color);\n",
"    border-color: transparent;\n",
"    border-bottom-color: var(--fill-color);\n",
"    animation:\n",
"      spin 1s steps(1) infinite;\n",
"  }\n",
"\n",
"  @keyframes spin {\n",
"    0% {\n",
"      border-color: transparent;\n",
"      border-bottom-color: var(--fill-color);\n",
"      border-left-color: var(--fill-color);\n",
"    }\n",
"    20% {\n",
"      border-color: transparent;\n",
"      border-left-color: var(--fill-color);\n",
"      border-top-color: var(--fill-color);\n",
"    }\n",
"    30% {\n",
"      border-color: transparent;\n",
"      border-left-color: var(--fill-color);\n",
"      border-top-color: var(--fill-color);\n",
"      border-right-color: var(--fill-color);\n",
"    }\n",
"    40% {\n",
"      border-color: transparent;\n",
"      border-right-color: var(--fill-color);\n",
"      border-top-color: var(--fill-color);\n",
"    }\n",
"    60% {\n",
"      border-color: transparent;\n",
"      border-right-color: var(--fill-color);\n",
"    }\n",
"    80% {\n",
"      border-color: transparent;\n",
"      border-right-color: var(--fill-color);\n",
"      border-bottom-color: var(--fill-color);\n",
"    }\n",
"    90% {\n",
"      border-color: transparent;\n",
"      border-bottom-color: var(--fill-color);\n",
"    }\n",
"  }\n",
"</style>\n",
       "\n",
       "  <script>\n",
       "    async function quickchart(key) {\n",
       "      const quickchartButtonEl =\n",
       "        document.querySelector('#' + key + ' button');\n",
       "      quickchartButtonEl.disabled = true;  // To prevent multiple
clicks.\n",
       "      quickchartButtonEl.classList.add('colab-df-spinner');\n",
       "      try {\n",
       "        const charts = await google.colab.kernel.invokeFunction(\n",
       "            'suggestCharts', [key], {});\n",
       "      } catch (error) {\n",
       "        console.error('Error during call to suggestCharts:', error);\n",
       "      }\n",
       "      quickchartButtonEl.classList.remove('colab-df-spinner');\n",
       "      quickchartButtonEl.classList.add('colab-df-quickchart-
complete');\n",
       "    }\n",
       "    (() => {\n",
       "      let quickchartButtonEl =\n",
       "        document.querySelector('#df-adeb9758-7329-4afa-a77e-
6f55c079e487 button');\n",
       "      quickchartButtonEl.style.display =\n",
       "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
       "    })();\n",
       "  </script>\n",
       "</div>\n",
       "\n",
       "    </div>\n",
       "  </div>\n"
      ]
     },
     "metadata": {},
     "execution_count": 1333
    }
   ]
  },
  {
   "cell_type": "markdown",
   "metadata": {
    "id": "zgYuJvIkG9T3"
   },
   "source": [
    "## Scaling"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": null,
      "metadata": {
        "id": "WoJ2lzXYnibj"
      },
      "outputs": [],
      "source": [
        "# from sklearn.preprocessing import StandardScaler\n",
        "# columns =
[\"CHILDREN\",\"Annual_income\",\"EDUCATION\",\"Employed_days\",\"Family_Members\",\"A
ge\"]\n",
        "# scaler = StandardScaler()\n",
        "# encoded_data[columns] = scaler.fit_transform(encoded_data[columns])"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "rJcf58sifcX7",
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "outputId": "f66efa85-278c-4505-a706-e8b755dc6712"
      },
      "outputs": [
        {
          "output_type": "stream",
          "name": "stderr",
          "text": [
            "<ipython-input-1335-e2725277d122>:4: DeprecationWarning: In a future
version, `df.iloc[:, i] = newvals` will attempt to set the values inplace instead of
always setting a new array. To retain the old behavior, use either `df[df.columns[i]]
= newvals` or, if columns are non-unique, `df.isetitem(i, newvals)`\n",
            "  encoded_data.iloc[:,:] =
scaler.fit_transform(encoded_data.iloc[:,:])\n"
          ]
        }
      ],
      "source": [
        "#using mixmax scaler\n",
        "from sklearn.preprocessing import MinMaxScaler\n",
        "scaler = MinMaxScaler()\n",
        "encoded_data.iloc[:,:] = scaler.fit_transform(encoded_data.iloc[:,:])"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "YO7nNU5e7zse",
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 290
        },
        "outputId": "34c8833c-fa77-4833-d51f-015e2159bf86"
      },
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "   GENDER  Car_Owner  Propert_Owner  CHILDREN  Annual_income  EDUCATION  
\\\n",
              "0     1.0        1.0            1.0       0.0       0.534466       0.75   
\n",
              "1     0.0        1.0            0.0       0.0       0.761514       0.75   
\n",
              "2     0.0        1.0            0.0       0.0       0.761514       0.75   
\n",
              "3     0.0        1.0            0.0       0.0       0.502835       0.75   
\n",
              "4     0.0        1.0            0.0       0.0       0.761514       0.75   
\n",
              "\n",
              "   Phone  EMAIL_ID  Family_Members  label  ...  Secretaries  Security 
staff  \\\n",
              "0    0.0       0.0        0.285714    1.0  ...          0.0           
0.0   \n",
              "1    1.0       0.0        0.285714    1.0  ...          0.0           
0.0   \n",
              "2    1.0       0.0        0.285714    1.0  ...          0.0           
0.0   \n",
              "3    1.0       0.0        0.285714    1.0  ...          0.0           
0.0   \n",
              "4    1.0       0.0        0.285714    1.0  ...          0.0           
0.0   \n",
              "\n",
              "   Unemployed  Unknown  Waiters/barmen staff  Civil marriage  Married  
\\\n",
              "0         1.0      0.0                   0.0             0.0      1.0   
\n",
              "1         0.0      1.0                   0.0             0.0      1.0   
\n",
              "2         0.0      1.0                   0.0             0.0      1.0   
\n",
              "3         0.0      1.0                   0.0             0.0      1.0   
\n",
              "4         0.0      1.0                   0.0             0.0      1.0   
\n",
              ″\n″,
              ″   Separated  Single / not married  Widow  \n″,
              ″0        0.0                   0.0    0.0  \n″,
              ″1        0.0                   0.0    0.0  \n″,
              ″2        0.0                   0.0    0.0  \n″,
              ″3        0.0                   0.0    0.0  \n″,
              ″4        0.0                   0.0    0.0  \n″,
              ″\n″,
              ″[5 rows x 47 columns]″
            ],
            ″text/html″: [
              ″\n″,
              ″  <div id=\″df-11f7196f-340d-434a-8dfd-7d03f965e3a2\″ class=\″colab-df-
container\″>\n″,
              ″    <div>\n″,
              ″<style scoped>\n″,
              ″    .dataframe tbody tr th:only-of-type {\n″,
              ″        vertical-align: middle;\n″,
              ″    }\n″,
              ″\n″,
              ″    .dataframe tbody tr th {\n″,
              ″        vertical-align: top;\n″,
              ″    }\n″,
              ″\n″,
              ″    .dataframe thead th {\n″,
              ″        text-align: right;\n″,
              ″    }\n″,
              ″</style>\n″,
              ″<table border=\″1\″ class=\″dataframe\″>\n″,
              ″  <thead>\n″,
              ″    <tr style=\″text-align: right;\″>\n″,
              ″      <th></th>\n″,
              ″      <th>GENDER</th>\n″,
              ″      <th>Car_Owner</th>\n″,
              ″      <th>Propert_Owner</th>\n″,
              ″      <th>CHILDREN</th>\n″,
              ″      <th>Annual_income</th>\n″,
              ″      <th>EDUCATION</th>\n″,
              ″      <th>Phone</th>\n″,
              ″      <th>EMAIL_ID</th>\n″,
              ″      <th>Family_Members</th>\n″,
              ″      <th>label</th>\n″,
              ″      <th>...</th>\n″,
              ″      <th>Secretaries</th>\n″,
              ″      <th>Security staff</th>\n″,
              ″      <th>Unemployed</th>\n″,
              ″      <th>Unknown</th>\n″,
              ″      <th>Waiters/barmen staff</th>\n″,
              ″      <th>Civil marriage</th>\n″,
″      <th>Married</th>\n″,
″      <th>Separated</th>\n″,
″      <th>Single / not married</th>\n″,
″      <th>Widow</th>\n″,
″    </tr>\n″,
″  </thead>\n″,
″  <tbody>\n″,
″    <tr>\n″,
″      <th>0</th>\n″,
″      <td>1.0</td>\n″,
″      <td>1.0</td>\n″,
″      <td>1.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.534466</td>\n″,
″      <td>0.75</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.285714</td>\n″,
″      <td>1.0</td>\n″,
″      <td>...</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>1.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>1.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>1</th>\n″,
″      <td>0.0</td>\n″,
″      <td>1.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.761514</td>\n″,
″      <td>0.75</td>\n″,
″      <td>1.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.285714</td>\n″,
″      <td>1.0</td>\n″,
″      <td>...</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>1.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>1.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>2</th>\n″,
″      <td>0.0</td>\n″,
″      <td>1.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.761514</td>\n″,
″      <td>0.75</td>\n″,
″      <td>1.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.285714</td>\n″,
″      <td>1.0</td>\n″,
″      <td>...</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>1.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>1.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>3</th>\n″,
″      <td>0.0</td>\n″,
″      <td>1.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.502835</td>\n″,
″      <td>0.75</td>\n″,
″      <td>1.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.285714</td>\n″,
″      <td>1.0</td>\n″,
″      <td>...</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>1.0</td>\n″,
″      <td>0.0</td>\n″,
″      <td>0.0</td>\n″,
              "      <td>1.0</td>\n",
              "      <td>0.0</td>\n",
              "      <td>0.0</td>\n",
              "      <td>0.0</td>\n",
              "    </tr>\n",
              "    <tr>\n",
              "      <th>4</th>\n",
              "      <td>0.0</td>\n",
              "      <td>1.0</td>\n",
              "      <td>0.0</td>\n",
              "      <td>0.0</td>\n",
              "      <td>0.761514</td>\n",
              "      <td>0.75</td>\n",
              "      <td>1.0</td>\n",
              "      <td>0.0</td>\n",
              "      <td>0.285714</td>\n",
              "      <td>1.0</td>\n",
              "      <td>...</td>\n",
              "      <td>0.0</td>\n",
              "      <td>0.0</td>\n",
              "      <td>0.0</td>\n",
              "      <td>1.0</td>\n",
              "      <td>0.0</td>\n",
              "      <td>0.0</td>\n",
              "      <td>1.0</td>\n",
              "      <td>0.0</td>\n",
              "      <td>0.0</td>\n",
              "      <td>0.0</td>\n",
              "    </tr>\n",
              "  </tbody>\n",
              "</table>\n",
              "<p>5 rows 脳 47 columns</p>\n",
              "</div>\n",
              "    <div class=\"colab-df-buttons\">\n",
              "\n",
              "  <div class=\"colab-df-container\">\n",
              "    <button class=\"colab-df-convert\" 
onclick=\"convertToInteractive('df-11f7196f-340d-434a-8dfd-7d03f965e3a2')\"\n",
              "            title=\"Convert this dataframe to an interactive 
table.\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\" viewBox=\"0 
-960 960 960\">\n",
              "    <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220 
220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-
160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",
              "  </svg>\n",
              "    </button>\n",
              ″\n″,
              ″  <style>\n″,
              ″    .colab-df-container {\n″,
              ″      display:flex;\n″,
              ″      gap: 12px;\n″,
              ″    }\n″,
              ″\n″,
              ″    .colab-df-convert {\n″,
              ″      background-color: #E8F0FE;\n″,
              ″      border: none;\n″,
              ″      border-radius: 50%;\n″,
              ″      cursor: pointer;\n″,
              ″      display: none;\n″,
              ″      fill: #1967D2;\n″,
              ″      height: 32px;\n″,
              ″      padding: 0 0 0 0;\n″,
              ″      width: 32px;\n″,
              ″    }\n″,
              ″\n″,
              ″    .colab-df-convert:hover {\n″,
              ″      background-color: #E2EBFA;\n″,
              ″      box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n″,
              ″      fill: #174EA6;\n″,
              ″    }\n″,
              ″\n″,
              ″    .colab-df-buttons div {\n″,
              ″      margin-bottom: 4px;\n″,
              ″    }\n″,
              ″\n″,
              ″    [theme=dark] .colab-df-convert {\n″,
              ″      background-color: #3B4455;\n″,
              ″      fill: #D2E3FC;\n″,
              ″    }\n″,
              ″\n″,
              ″    [theme=dark] .colab-df-convert:hover {\n″,
              ″      background-color: #434B5C;\n″,
              ″      box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n″,
              ″      filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n″,
              ″      fill: #FFFFFF;\n″,
              ″    }\n″,
              ″  </style>\n″,
              ″\n″,
              ″    <script>\n″,
              ″      const buttonEl =\n″,
              ″        document.querySelector(’#df-11f7196f-340d-434a-8dfd-
7d03f965e3a2 button.colab-df-convert’);\n″,
              ″      buttonEl.style.display =\n″,
              ″        google.colab.kernel.accessAllowed ? ’block’ : ’none’;\n″,
              "\n",
              "      async function convertToInteractive(key) {\n",
              "        const element = document.querySelector('#df-11f7196f-340d-434a-
8dfd-7d03f965e3a2');\n",
              "        const dataTable =\n",
              "          await
google.colab.kernel.invokeFunction('convertToInteractive',\n",
              "                                                   [key], {});\n",
              "        if (!dataTable) return;\n",
              "\n",
              "        const docLinkHtml = 'Like what you see? Visit the ' +\n",
              "          '<a target=\"_blank\"
href=https://colab.research.google.com/notebooks/data_table.ipynb>data table
notebook</a>'\n",
              "          + ' to learn more about interactive tables.';\n",
              "        element.innerHTML = '';\n",
              "        dataTable['output_type'] = 'display_data';\n",
              "        await google.colab.output.renderOutput(dataTable, element);\n",
              "        const docLink = document.createElement('div');\n",
              "        docLink.innerHTML = docLinkHtml;\n",
              "        element.appendChild(docLink);\n",
              "      }\n",
              "    </script>\n",
              "  </div>\n",
              "\n",
              "\n",
              "<div id=\"df-3ac71ad0-f497-42d9-b338-bd2da5dfe3a4\">\n",
              "  <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-
3ac71ad0-f497-42d9-b338-bd2da5dfe3a4')\"\n",
              "            title=\"Suggest charts\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "<svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0 0
24 24\"\n",
              "     width=\"24px\">\n",
              "    <g>\n",
              "        <path d=\"M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-
.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\"/>\n",
              "    </g>\n",
              "</svg>\n",
              "  </button>\n",
              "\n",
              "<style>\n",
              "  .colab-df-quickchart {\n",
              "      --bg-color: #E8F0FE;\n",
              "      --fill-color: #1967D2;\n",
              "      --hover-bg-color: #E2EBFA;\n",
              "      --hover-fill-color: #174EA6;\n",
              "      --disabled-fill-color: #AAA;\n",
"      --disabled-bg-color: #DDD;\n",
"  }\n",
"\n",
"  [theme=dark] .colab-df-quickchart {\n",
"      --bg-color: #3B4455;\n",
"      --fill-color: #D2E3FC;\n",
"      --hover-bg-color: #434B5C;\n",
"      --hover-fill-color: #FFFFFF;\n",
"      --disabled-bg-color: #3B4455;\n",
"      --disabled-fill-color: #666;\n",
"  }\n",
"\n",
"  .colab-df-quickchart {\n",
"    background-color: var(--bg-color);\n",
"    border: none;\n",
"    border-radius: 50%;\n",
"    cursor: pointer;\n",
"    display: none;\n",
"    fill: var(--fill-color);\n",
"    height: 32px;\n",
"    padding: 0;\n",
"    width: 32px;\n",
"  }\n",
"\n",
"  .colab-df-quickchart:hover {\n",
"    background-color: var(--hover-bg-color);\n",
"    box-shadow: 0 1px 2px rgba(60, 64, 67, 0.3), 0 1px 3px 1px rgba(60,
64, 67, 0.15);\n",
"    fill: var(--button-hover-fill-color);\n",
"  }\n",
"\n",
"  .colab-df-quickchart-complete:disabled,\n",
"  .colab-df-quickchart-complete:disabled:hover {\n",
"    background-color: var(--disabled-bg-color);\n",
"    fill: var(--disabled-fill-color);\n",
"    box-shadow: none;\n",
"  }\n",
"\n",
"  .colab-df-spinner {\n",
"    border: 2px solid var(--fill-color);\n",
"    border-color: transparent;\n",
"    border-bottom-color: var(--fill-color);\n",
"    animation:\n",
"      spin 1s steps(1) infinite;\n",
"  }\n",
"\n",
"  @keyframes spin {\n",
"    0% {\n",
"      border-color: transparent;\n",
              ″      border-bottom-color: var(--fill-color);\n″,
              ″      border-left-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    20% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-left-color: var(--fill-color);\n″,
              ″      border-top-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    30% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-left-color: var(--fill-color);\n″,
              ″      border-top-color: var(--fill-color);\n″,
              ″      border-right-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    40% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-right-color: var(--fill-color);\n″,
              ″      border-top-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    60% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-right-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    80% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-right-color: var(--fill-color);\n″,
              ″      border-bottom-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    90% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-bottom-color: var(--fill-color);\n″,
              ″    }\n″,
              ″  }\n″,
              ″</style>\n″,
              ″\n″,
              ″  <script>\n″,
              ″    async function quickchart(key) {\n″,
              ″      const quickchartButtonEl =\n″,
              ″        document.querySelector(’#’ + key + ’ button’);\n″,
              ″      quickchartButtonEl.disabled = true;  // To prevent multiple
clicks.\n″,
              ″      quickchartButtonEl.classList.add(’colab-df-spinner’);\n″,
              ″      try {\n″,
              ″        const charts = await google.colab.kernel.invokeFunction(\n″,
              ″            ’suggestCharts’, [key], {});\n″,
              ″      } catch (error) {\n″,
              ″        console.error(’Error during call to suggestCharts:’, error);\n″,
              ″      }\n″,
              ″      quickchartButtonEl.classList.remove(’colab-df-spinner’);\n″,
              "      quickchartButtonEl.classList.add('colab-df-quickchart-
complete');\n",
              "    }\n",
              "    (() => {\n",
              "      let quickchartButtonEl =\n",
              "        document.querySelector('#df-3ac71ad0-f497-42d9-b338-
bd2da5dfe3a4 button');\n",
              "      quickchartButtonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "    })();\n",
              "  </script>\n",
              "</div>\n",
              "\n",
              "    </div>\n",
              "  </div>\n"
            ]
          },
          "metadata": {},
          "execution_count": 1336
        }
      ],
      "source": [
        "#scaled data\n",
        "encoded_data.head()"
      ]
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "K2AQzmpMlSJq"
      },
      "source": [
        "## Fixing class imbalance"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "_hY022eifneC",
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "outputId": "142b022e-aa0e-4a2f-e892-0fb5375e1032"
      },
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "0.0    1373\n",
              "1.0     175\n",
              "Name: label, dtype: int64"
            ]
          },
          "metadata": {},
          "execution_count": 1337
        }
      ],
      "source": [
        "encoded_data[\"label\"].value_counts()"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "1sua__3fkaUA"
      },
      "outputs": [],
      "source": [
        "# splitting the dataset into train and test\n",
        "from sklearn.model_selection import train_test_split\n",
        "X_train, X_test, y_train, y_test = 
train_test_split(encoded_data.drop(\"label\",axis = 
1),encoded_data[\"label\"],test_size = 0.20,stratify = 
encoded_data[\"label\"],random_state = 42)"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "BlF6u7C1eSzl"
      },
      "outputs": [],
      "source": [
        "# Adaptive Synthetic Sampling for balancing the dataset\n",
        "from imblearn.over_sampling import ADASYN\n",
        "\n",
        "sm = ADASYN()\n",
        "X_train_sm, y_train_sm = sm.fit_resample(X_train, y_train)"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "fackoFswoIfb",
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "outputId": "d1824d1e-a35d-461e-a071-26229f0e2b68"
      },
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "1.0    1123\n",
              "0.0    1098\n",
              "Name: label, dtype: int64"
            ]
          },
          "metadata": {},
          "execution_count": 1340
        }
      ],
      "source": [
        "#fixed class imbalance\n",
        "y_train_sm.value_counts()"
      ]
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "X6jjP3EoGaoe"
      },
      "source": [
        "# Modelling"
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "#making a dictionary to store the metric values\n",
        "eval_dict
={\"Model\":[],\"AUC\":[],\"Recall\":[],\"Precision\":[],\"Accuracy\":[]}"
      ],
      "metadata": {
        "id": "44e5PtKTSR2M"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
   "source": [
    "#defining a function to run and evaluate a model\n",
    "\n",
    "def model_train_eval(model, model_name: str):\n",
    "  # Append the model name to the evaluation dictionary\n",
    "  eval_dict[\"Model\"].append(model_name)\n",
    "\n",
    "  # Fit the model on the training data\n",
    "  model.fit(X_train_sm, y_train_sm)\n",
    "\n",
    "  # Predict the target variable for the test data\n",
    "  y_pred = model.predict(X_test)\n",
    "\n",
    "  # Predict the target variable for the training data\n",
    "  y_train_pred = model.predict(X_train)\n",
    "\n",
    "  # Calculate and append various evaluation metrics to the evaluation
dictionary\n",
    "  eval_dict[\"Precision\"].append(precision_score(y_test, y_pred))\n",
    "  eval_dict[\"Recall\"].append(recall_score(y_test, y_pred))\n",
    "  eval_dict[\"AUC\"].append(roc_auc_score(y_test, y_pred))\n",
    "  eval_dict[\"Accuracy\"].append(accuracy_score(y_test, y_pred))\n",
    "\n",
    "  # Calculate the training accuracy\n",
    "  training_accuracy = accuracy_score(y_train, y_train_pred)\n",
    "\n",
    "  # Print the training accuracy\n",
    "  print(\"Training Accuracy:\", training_accuracy, end=\"\\n\\n\")\n",
    "\n",
    "  # Print the classification report for the test data\n",
    "  print(\"Classification_Report -->\", end=\"\\n\\n\")\n",
    "  print(classification_report(y_test, y_pred))\n",
    "\n",
    "  # Perform cross-validation and predict the target variable for the training
data\n",
    "  y_pred_cv = cross_val_predict(model, X_train_sm, y_train_sm, cv=5)\n",
    "\n",
    "  # Print the classification report for the cross-validated predictions\n",
    "  print(\"Cross Validation Classification_Report -->\", end=\"\\n\\n\")\n",
    "  print(classification_report(y_train_sm, y_pred_cv))\n",
    "\n",
    "  # Calculate the confusion matrix for the test data\n",
    "  cm = confusion_matrix(y_test, y_pred)\n",
    "\n",
    "  # Plot the confusion matrix\n",
    "  plt.figure(figsize=(8, 6))\n",
    "  sns.heatmap(cm, annot=True, fmt=\"d\", cmap=\"Blues\")\n",
    "  plt.xlabel('Predicted labels')\n",
    "  plt.ylabel('True labels')\n",
        "  plt.title('Confusion Matrix')\n",
        "  plt.show()\n"
      ],
      "metadata": {
        "id": "HO28j4-aOib2"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "jAANMKlzGXW9"
      },
      "source": [
        "## LogisticRegression"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "xVxKrPDOGXEW",
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 980
        },
        "outputId": "593dae35-30e0-4d7c-a939-4f0684cc2fbb"
      },
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
            "Training Accuracy: 0.6001615508885298\n",
            "\n",
            "Classification_Report -->\n",
            "\n",
            "              precision    recall  f1-score   support\n",
            "\n",
            "         0.0       0.94      0.62      0.75       275\n",
            "         1.0       0.19      0.69      0.29        35\n",
            "\n",
            "    accuracy                           0.63       310\n",
            "   macro avg       0.56      0.65      0.52       310\n",
            "weighted avg       0.85      0.63      0.69       310\n",
            "\n",
            "Cross Validation Classification_Report -->\n",
            "\n",
            "              precision    recall  f1-score   support\n",
            "\n",
            "         0.0       0.58      0.57      0.57      1098\n",
            "         1.0       0.59      0.60      0.59      1123\n",
            "\n",
            "    accuracy                           0.58      2221\n",
            "   macro avg       0.58      0.58      0.58      2221\n",
            "weighted avg       0.58      0.58      0.58      2221\n",
            "\n"
          ]
        },
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 800x600 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAAAokAAAIjCAYAAABvUIGpAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAABF4El
EQVR4nO3deXRUVbr+8adCkkoIJCEMCVEIIAhBkFk6MjeRSREG1HUgAitAgoBxHTL6BBFBAQVnBpQoR1aQUBFE
WRQIjIYREVkiKBAgoIhJpAiJOf3hz/qWmzAFFSlEur76VVrde1z6py36i7ufe+z99mxWZZlCQAAAPiTAF8XAAA
AgNKHJhEAAAAGmkQAAAAAYaBIBAABgoEkEAACAgSYRAAAABppEAAAAGGgSAQAAYKBJBAAAgIEmEcB57dq1S126d
FFERIRsNpuWLFni0ev/+OOPstlsmj17tsPPstv/+OOPstlsmj9/vkevW5Z17+Nhx2bPQBcJd/NhRHTt29HUZAPwcTSJQBuzZs0f//Oc/VadOHYWEhFx
t2rTRM888oxMnTnj13klJSdq+fbsee+wxvfbaa2rZsqVX71xX71eSBg4cKJvNpvDw8LP+jrt27ZLNZpPNZt0adPcv
v7BgwcladIkpaene6BaAChZgb4uAMD5vf/++rHP/4hu92u0++U40aNdLJkyf12WefaerXysrQ1222/14osveuX
eJO6cUFpamv79739r+PDhXrlHXFycTpvpwoaCgIK9c/68EBgbq+PHjWrZsmfre+dybjd+dybOHChQoJCVF+fv4FXfvgw
YOaHmyatWqZNNmxb7cx9//PoyMnEShi4BNxTPG+1D3G3EPOD62cS0Xnfft7Q6TydsoHq1nDh0U2XWuSmQbv/Bv7bmYpgZXVaR6SR
Q/ezz3OB5BKJ0yQYrGYGrgxgYA1pMmpJEZuu8AtPjIoTBLfX9oDYW44ZNrRRPLq7/4AiMOYwGhnPYrpF9tTgDI
WQMm4/Bc9nX2PvsmVuUMO7Hik8rOcpTfvtnvo7G8H0KFJbwGTELpQ2mCcFi0nBw7lJF7uWXM9AyAXSj1bxXtMPd+n
ozvNc2iPbbB6TKQgsF0HvmMm+W3azZLRzvW0ZdhUHYE3QB0XEuNQqRnkyNg7NGnUTDh9PjkNnBmcOCWSO4BwwHJJ8W4
FWrVpKkQYMG0aetT3/Pjh07qlGjRtyZYZvat2+v8uXL03+XM/t67ArKVlXTw4MFif1cAKC6aHR6TkoYlWrKVnGP29jtqAb
aRKAUW7ZsmerUqaNrr722Wv0ffdZxmbZM2aoQ4c0OSk1NVf/+/Y1lzd+/erZtvvlneddnn76aVWqV
EkDBw7Ut99+KFuucq+VKtwrYzf8rSioiLNmDFD9erVk91uV/fXXXs9uV5UqVVS1alV9/fXX0nbswHLHvedll7n1kMq0adMUFRWl9
```

PROzZo1S9WqVSv2ZwHAXTSJQCkVHh6u2NhYffPNN2597swHR86lXLlyZx23LOuC73F6vdxpoaGhWrdunT755BP
dcccd+vrrr3XLLbfouuuuM869GBfzXU6z2+3q06ePFixYoMWLF58zRZSkxx9/XMnJyWrfvr1ef/11ffTRR1q5c
qWuuuqqYiem0h+/jzu++uorHT58WJK0fft2tz4LAO6iSQRKsRtuuEF79uxRWlraX54bFxenoqIi7dq1y2U8Kyt
L2dnZzieVPaFSpUouTwKfdmZaKUkBAQHq3Lmzpk+fru+++06PPfaYVq9erU8//d586d041lj33//vapUq
aKwsLCL+wLncNttt+mrr77S77//ftaHfU773//+p06dOumVV15R//791aVLFyUmJhq/SXEb9uLIy8voEGD1LB
hQw0dOlRTp07Vpk2bPHZ9ADgTTSJQij344IMCvwT3XffraysLOP4nj179Mwzz0OzpTaXP3JbntOrVq6tpO6ZasGCBS9P1zTff6OOPP3Z+T2/
o1KmTHnnkET377LOKiYk553nlypUzq3335bBw4ccBk73cyeraF217hx47R//34tWLBA06dPV61atZSU1HTO3
xEALhabaQO12BVXXKFFixbplltuUXx8vMtfXNmwYYPefvttDRw4UJLUpEkTJSUl6cUXX1R2drY6dOigL7/8Ugs
WLFDv3r3Pub3Khejfv7/GjRunm266Sffff7+OHz+uOXPm6Morr3R5cGPKlClat26d+Fxenw4cN6/vnnd
fnll6tt27bnvP5TTz217t27KyEhQYMYMHD9aJEyc0e/ZsRUREaNKkSR77HmcKCAjQww8//Zs3Pub3Khejfv7/GjRunm266Sffff7+OHz+uOXPm6Morr3R5cGPKlClat26drr/+esXFxenw4cN6/vnnd
fnll6tt27bnvP5TTz2l7t27KyEhQYMHD9aJEyc0e/ZsRUREaNKkSR77HmcKCAjQww8//Jfn3XDDDZoyZYoGDRq
ka6+9Vtu3b9fChQtVp04dl/OuuOIKRUZGau7cuapYsaLCwsLUunVr1a5d2626Vq9ereeff14TJ050bskzb948d
ezYUePHj9fUqVPduh4AF1uPn64GUAw//PCDNWTIEKtWrVpWcWCwVbFiRatNmzbW7Nmzrfz8fz8fOd5BQUF1uTJk63
atWtbQUFBVoOaNayUlBSXcyzrjy1wrr/+euM+Z269cq4tcCzLsj7++GOrUaNGVnBwsFW/fn3r9ddfN7bAWvqL
dWrVy8rNjbWCg4OtmJjY61bb73V+uGHH4x7nLPGCgONtcLDw62ePXta3333npp9nhY78+935hY78+b
NsyRZGRkZ5/xNLct1C5xzO0dcWOKNHj7aqV69uhYaGWm3atLHS0tWe++KuSN0tLOunXNe++9ZzVs2NAKDAx0
+Z4dOnSwrrqq4zmzZtbBQUFLueNGjXKCgIsNLS0s77HQDgQtgsy42V3QAAAPALrEkEAACAgSYRAAAAAApp
EAAAAGGgSAQAAYKBJBAAAgIEmEQAAAAaaRAAAABguyb+4EtpsuK9LAOAl192b5OsSAHjJOqGtfHZvb/YOJ7561
mvX9iaSRAAAAABguySQRAADALTZyszPRJAIAANhsvq6g1KFtBgAAgIEkEQAAgOlmA78IAAAADCSJAAAArEkOkCQ
CAADAQJIIAADAmkQDvwgAAAAMJIkAAACsSTTQJAIAADDdbOAXAQAAgIEkEQAAgOlmAOkiAAAADCSJAAAArEk08
IsAAADAQJIIAADAmkQDSSIAAAAMJIkAAACsSTTQJAIAADDdbKBtBgAAgIEkEQAAgOlmA78IAAAADDSJAAAAtgD
vvdy0bt069ezZU7GxsbLZbFqyZIlxzo4d03TjjTcqIiJCYWFhatmypTcqIiJCYWFhatWql5fbv3+88np2HDhqly5cqqUKGC+vbtq
6ysLLfqoEkEAAAoRfLy8ty8SkSRM999xkZz2+Z88etW3bVg0KzO0fa/z48QoORjnEXOxwVdq6ioSO+//76uvPWqVVP
r1q1dpqS3bNmigoICJSYm0oscaNGjigmjVrki0trdj3okkEAADw4prE/Pz8s/5wU5+TGUZ99CTESe3pvvPJKde3aV+8
rlq1dpqS3bNmigoICJSYm0oscaNGjigmjVrki0trdj3okkEAADw4prE/Pz8s/5wU5+TGUZ99CTESe3pvvPJKde3aV+8
dLf//53SdK8efMUHx+vL774Qn/729+KVQdJIgAAgBc5HA7l5O4S4vBwOxNwvdq6ioSO+//76uvPWqVVP
r1q1dpqS3bNmigoICJSYm0oscaNGjigmjVrki0trdj3okkEAADw4prE1NRURUREuEuEuLxSU1MvqMzxNzdXTzzxh
Lp166aPP/5YN/5YN910aPP/5YN910k/r06a01a9dKkjIzMxUcHKzIyZHRyszM7PY92K6GQAAwIubaadekaubaaekpCg50dllzG63X0yZS49DHsZKkXq3d
LioqKJEm9evXSyJiYZvxF5YN2AAA6Y3uqyGvUTIm81xPw1TmuqOqkneOeOLA6K9A
WefSZJiYmJO8uRJZWdnu6SJWVlZomZtnFvgnE9wcLBatWqlnTt3uoz/8MMPiv8MMPiouLkySlaNFCQUF
BWrVqlfP4zp07tX//fiUkJBT7XqiJB1T7XiSJAAAAApUhubq52797tfJ+RkaH09HRFRkaqZs2aateurcaNG6eQkBDFxcVp7dq1eu211zR9+nR
ooVK7Rs2TKtWbNGkhQREaHBgwdr9OjRioyMVHh4uO677z4lJCQU+8llkwgAAEBtiKF3H0lHIjrkhGwgSbvBHP9xU/7UiPCCdmEkQ
BLSDD4gQgSRSPQBIUAAAGh2gTTGwYNXcCUdLWOqo5BxjM0MlFxvNR3zpA/Un3H7qA7NP+jD1FP0I5X99+9O
mSGVtZXFYSRVvzknS4GbtclJIWRjY2hlTc5T9F93yT/R3ppjvCqoWl1nLVcqQjF+Cog3ao6xmSFbSt2j+v+KrVUqhvX73PrBLNdXD+xIGO6+c3NrR
AwCY5F4Yq8DuM6XxzH/12aj3WhrN5DMLOTOU38GfdkFhsU9Ck71FcyJsOJB6qDFTAF7k7+GH6q1W9VdenDF6ajmJ01PEAk57GqRL2N+ciFrnXLVPYFKuCFFWqBHK7fkLxXo6Cz
8Kq8trp6zxGcFVh4vRkz2NLIdF0RAFa6YCJ+RD9cIkEeoGmnvF/J9uIYJ+CKNXXFX8gvJ8BJZGq0YqSMmo0sgrSMMSfKXTECSYNDnqSK5aKh1h6PB/vUGKMFBKq0jnNHCDP2Y9s6K2w
ZQGmD/WBfI8H1mH3Y8s8ER8A4H7Vc3R2+N1qTa8t+8h5L/+SK94o2RzfLC6Y3hbcnp6KCYF0IZ11ZXsFaFmTl8XEJyFzL9Dg+G61Ev5XDqV6fmh2g1z0Rvt5SC7z/6WUaJnLOmQnGGFp3u9U8QWIcAJl
V1O8QpYVT2Ndi/L/35WYuJdEUWn+9V1u8z6PHRPJvZOKk9UcfwLGp1yyaqFDSJtyPZX9WJIQAkZRr8bTpJ8WvCUYHr4Vrm8y4Se3Ycug0f1fUeRlvXUY3vNfLrUQQJc5YBdnALO73qCCeH8E3PHRfA4hZOVqvqQuH
5qHq8QAzcg4RbO6S2LmXyzvflUNfWVt+fR8lZk9TWsuyHeKDQ9nUDDoC+4JmR9MbNDDCUzEOzkB3ZqPOy7kBH2Jza7BiZ2dDvu31bHg9hXkLmXqRFvnGNSFNYfbfORGeLzkTXUSuYOxTNyDXKYe9RdXc4FuOvhhqFDnS
MJ/HCXTrfxOUrVOFpL9mjLpBcG/YcajcBKxwLHzwpzBDnlCKsWU0R1CG7CXHuHOCSdHtT61pXqTuHjQddALb7K3vbBhjMBv5exhW+mfK1AWEkpfZnPNL0OWj9o9Vm9GnqhH2qCfEwNzD76Hf8NJK/4BdJXlZ4lq7rE7T4LN
2pHVVv6cWFWjcvn5cL2bFXG8N1k0gSr3ALs2WqPKxeDqiQmAwHDZ4LXq+6L3T8x3K4sPg5QDqRe8ifNKnNMmLYQtJsfYJ8kh2mGW6qKQ16Hv7GuVbWhKEhN/RnnebgJe0fGsL4/DNPHW4t7jTdcpHsrHu6U8u3yhfq
0d1wmRSDP+mzJG+N1TfNk0fu3JtJMsGz5uDzL5bXJAk6XGGeZRaJtCKLbTyuuw2oAHeeK8YxVuqQqlIJq/iHR6Vg2nEOo5ChvbGdRf+PLWbXJF7vTRvSWJwOwCF4TSY2y6jyKiMTL4qxL5qDMFeTPYF4GF9UkLFmeKXlZ
uPW/IcVLavz5XNcaRUAR2k4bhsf9YB9Y9GrxIaMKsc+RwbKrn7Dv/0+s7N8NlqoK1f5p0Fbbs6HtScoGZ2Nhjns0ZSwQY0Jk1U+BcI6MX7aVMjHtFm1QbPvuRjFaoFbSd1btEDt+vOhm4KdM7e66GtXQ9lIq5v7sR+M8x
kPP2Om0XBm11xx2RHSHJMYfqbIYFV5dImqz4B8pBTEOlHKTIMXfuX57O0M/dz33P9WlHGqJG4r20P8M8QOq5Mh5p5UEWzz9X8XdkA4HxOrbpUNyFp9LwKUWmCWSDUm6ww3T5tYnWWlvBJVHHzxb7JHJsk0gSgTVyhTf3
2hCEgqLTDNPr8hy9Wv9U0+T+k1wjXlqwaQJ3fOIh9g/XvQL3bVTe8FlrNm2oWiFJOa3mSPBO/Q3cv6ghY7XvJ9hMUXBuBE+2LQ/eKNR8iqA=

NJIkAAAAwkCQCAAC/R5JoIkkEAACAgSQRAAD4PZJEE0OiAAAAPaKB6WYAAAAYSBIBAIDfY7rZRJIIAAAAA0kiA
ADweySJJpJEAAAAGEgSUSq1aX6FRt2ZqOYNa6p61Qj1G/Wilq352nn8xFfPnvVz/5qxWDNeXSVJqhReXtPH/UM
92jdSkWVpyap0jZn6P+WdOFki3wHA2V0VUOE3NamuK6qUV+WwYD32OS5t3Jftcs5tLWLVJb6qwoIDtSPzd835b
J8O5Ticx1+69WpFV7S7fGbBxp/0zrbMkvgKuASRJJpoElEqhYXatf2HA3r1vTS9OX2ocbxWYorL+y5trtLcibd
p8ap059i8x5MUUyVCN9z7rIICy+mFybfrufG3aeC/5nu5egDnYw8qp4wjx/XJzl/0ry71jON9msTohkbRemZNh
rJ+d2hAy8sOuceVGvb2NyootJznLdz0sz76/hfn+xMFRSVSP+AvaBJRKn38+Xf6+PPvznk868jvLu97dmystZt
26ccDRyRJ9WtHq2ubq9RmwFRt/W6/JCn5ybe1ZPa9SpmxWId+Oea94gGc19afjmnrT+f+N3hj42i99dUhZ7o44
9MMvXpHU/2tViWt33PUed6JgiJlnzjl7XLhJ0gSTT5tEn/99Vf95z//UVpamjIz/5giiImJ0bXXXquBAweqatW
qviwPZUS1qIrq1raRhkx4zTnW+ura+i3nuLNBlKTVG3eqqMhSq0ZxWvrp12e7FAAfi65oV1T5YGO78H9N5PGCQ
v1wOFf1q1VwaRL7Nq2ufs1j9WuuQ2t3H9V72zNVZJ3tqkAx0CMafNYkbtq0SV27dlX58uWVmJioK6+8UpKUlZW
lWbNm6YknntBHH32kli1bnvc6DodDDofDZcwqKpQtoJzXakfpcnvP1vr9eL6WrE53jkVXDtcvR13TxsLCIh3NO
a7oKuElXCGA4qpUPkiSlH3cNSHMPnHKeUySln+TpT2/Hleu45QaRFfQnddcrkrlg/SfL34q0XqBS5nPmsQRIOb
oH//4h+bOnWtEvJZl6Z577tGIESOUlpZ23uukpqZq8uTJLmPlolspqPo1Hq8ZpdOdvf6mNz/cLMdJpp0Af/He9
iznf//x6AmdKrJOX7s4vfrlzzpFnIgLwHSzyWdb4Gzbtk2jRo066/9QbDabRo0apfTO9L+8TkpKio4dO+byCox
u4YWKURq1aXaF6teOObzFG1zGs47kqGpURZexcuUCFBVeXlm/5pRkiQDc8NvxAklSZHnXDCMyNNB57Gx2Hs5VY
ECA8cQzgAvnsyYxJiZGX3755TmPf/nll4q0jv7L69jtdoWHh7u8mGr2HOm9E7Tlu/3a/sMBl/GNX2eoUnh5NYu
v4Rzr2OpKBQTYtOmbfSVdJoBiyvrdoaPHT6pJ7P8tCwkNCtCV1Spo5+Hcc36uTuXyKiyylH3i3IOkcD42m81rr
7LKZ9PNY8aMOdChQ7VlyxZ17tzZ2RBmZWVp1apVeumllzRt2jRflQcfCwsN1hU1/u/BpVqXVdbVV16m33KO66f
M3yRJFcNC1Oe6Znpo+mLj8zszsvTR59/qufG36f7H3lBQYDnNeKif3v5oK082Az4WEhig6hH/l/hFh9tVu3Kof
s8v1K95J7V0e5b6NY/VwRyHsnIcGtDqMh09flJf/PjHv/361cJUv1oFfX0wRycKitQgOkyDE2pq7e4jyjtZ6Ku
vBVxyfNYkDhs2TFWqVNGMGTPO/PPPq7Dwj3/Y5cqVU4sWLTR//nz169fPV+XBx5o3jNPHLz/gfD91TF9Jo0mtLv
9DQia9Lkv7RtYVssumtFZvPeo1B/1qgGQ/10wcvjFBR0R+baY+e+rb3iwdwXnWrhunxng2c7+9OqClJWrXzVz2
zNkPvbstUSGCAhrWrpbDgcvou83dN+vAH5x6JBYWW2l0Rpf4tYhVULkBZvzu0dHuWlnzNRtq4cGU48PMam2VZP
l/hW1BQoF9//VWSVKVKFQUFBf3FJ84vtNlwT5QFoBS67t4kX5cAwEuWDm3ls3vXHf0h1669e1p3r13bm0rFZtp
BQUGqXr26r8sAAAB+qiyvHfSWUtEkAgAA+BI9oslnTzcDAACg9CJJBAAAfo/pZhNJIgAAAAwkiQAAw08RJJpIE
gEAAGAgSQQAAH4vIIAo8UwkiQAAADCQJAIAAL/HmkQTTSIAAPB7bIFjYroZAAAABpJEAADg9wgSTSSJAAAAMJA
kAgAAv8eaRBNJIgAAAAwkiQAAw0+RJJpIEgEAAGAgSQQAAH6PINFEkwgAAPwe08OmppsBAABKkXXr1qlnz56Kj
Y2VzWbTkiVLnMcKCgo0btw4NW7cwd96pgwcPulzj6NGjGjBggMLDwxUZGanBgwcrNzfXrTpoEgE
AgN+z2bz3cXp6aNGmi5557lhw1bTiWi5557zjh2/Phxbd26Ve26vvvaufOnbrxxhtdzhswsYIC+/fZbrVy5L
9e6des0d0hQt+pguhkAAKAU6d69u7p37/OcYxERETpx46eJSEJa2yIjc0jZTZtUsuWLFCmzZtUsu
WLSVJs2fPVo8ePTRt2jjTFxsYWqw6SRAAA4PdsXvw4HA7l5OS4fBwOh0dqP3bsmGw2mypWrChJ2rhxoypWrOhsECUpMTFRAQEB2rRpU7GvS5MIAADgRampqYqIiHD5pKamXvR18/PzNW7cON12220KDw+XJGVmZqpatWou5wUGBqpy5crKzMws9rWZbgYAAAj03uK1lJQUJScnu4zZ7faLumZBQYH69esny7I0Z86ci7rWudAkAgAAeJHdbr/opvBMpxvEvXv3avXq1c4UUZJiYmJ08OBBl/NPnTqlI0eOKCYmptj3YLoZAAD4PZsX/+NJpxvEH374QR9//LEiIyNdjickJOjo0aPasmWLc2z16tUqKipS69ati30fkkQAAIBSIjc3V7t373a+T09P17Zt21S5cmVVr15dN998s7Zu3arly5ersLDQuc6wcuXKCg4OVsOGDdWtWzcNGTJEc+fOVUFBgZKSktS/f/9i72wp0SQCAACUmp/l++qrr9SpUyfn+9NrGQcOHKjHH39cS5culSQ1a9bM5XNr1qxRx44dJUlvvvmmkpKS1LlzZwUEBKhv376aNWuWW3XQJAIAAJQSHTt2lGVZ5z3+Z8dOq1y5shYsWHBRdbAmEQAAAAaSRAAAgFIy3VyakCQCAADAQJIIAAD8nqe3qrkUkCQCAADAQJIIAAD8HkGiiSQRAAAABpJEAAAAokQDTSIAAPB7NnbSNjDdDAAAAANJIgAA8HtsgWMiSQQAAICBJBEAAPg9gkQTSSIAAAAMJIkAAABEiQaSRAAAABhIEgEAgN9jn0QTTSIAAPB7bIFjYroZAAAABpJEAADg9wgSTSSJAAAAMJAkAgAAECUaSBIBAABgIEkEAAB+jy1wTCSJAAAAMJAkAgAAv8c+iSaSRAAAABhIEgEAgN8jSDTRJAIAANAlGphuBgAAgIEkEQAA+D22wDGRJAIAAMBAkggAAPweW+CYSBIBAABgIEkEAAB+jyDRRJIIAAAAA0kiAAAAUaKBJhEAAPg9tsAxMd0MAAAAA0kiAADwe2yBYyJJBAAAgIEkEQAA+D2CRBNJIgAAAAwkiQAAAESJBpJEAAAAGEgSAQCA32OfRBNJIgAAAAwkiQAAwO+xT6KJJhEAAPg9ekQT080AAAAwkCQCAAC/x3SziSQRAAAABpJEAAAAViUaSBIBAABgIEkEAAB+jzWJJpJEAAAAGEgSAQCA3yNINNEkAgAAv8d0s4npZgAAABhIEgEAgN+zMeFsIEkEAACAgSQRAACAINFAkggAAAADSSIAAPB7BIkmkkQAAAAYSBIBAIDfY59EE00iAADwe2yBY2K6GQAAAAaSRAAAAIJEA0kiAAAADCSJAADA7xEkmkgSAQAAYCBJBAAAfo8tcEwkiQAAADCQJAIAAL/HPokmmkQAAOD3mG42Md0MAAAAA00iAAAADDSJAAAAMLAmEQAA+D3WJJpIEgEAAGAgSQQAAH6PLXBMJIkAAAAwkCQCAAC/x5pEE00iAADwe/SIJqabAQAAYCBJBAAAIEo0kCQCAADAQJIIAAD8HlvgmEgSAQAAYCBJBAAAfo8tcEwkiQAAADCQJAIAAL9HkGiiSQQAAKBLNDDdDAAAAANNIgAA8Hs2L/7HXevWrVPPnj0VGxsrm82mJUuWuBy3LEsTJkxQ9erVVb58eSUmJuqHH35wOefIkSMaMGCAwsPDVbFiRQ0ePFi5ublu1UGTCAAAUIrk5eWpadOmeu655855/Mknn9SsWbM0d+5cbdq0SaGhoeratavy8/Od5wwYMEDffvutVq5cqeXLl2vdunUaOnSoW3WwJhEAAPi90rQFTvfu3dW9e/dzHrMsSzNnztSjjz6qXr16SZJee+01RUdHa8mSJerfv7++++47rVixQps3b1bLli0lSbNnz1aPHj00bdo0xcbGFqsOkkQAAAAvcjgcysnJcXk5HI4LulZ6eroyMzOVmJjoHIuIiFDr1q21ceNGSdLGjRtVsWJFZ4MoSYmJiQoICNCmTZuKfa9LMkk8+Y9nfV0CSojD4VBqaqpSUlJkt9t9XQ4AD+HvNkpSiBe7oceTUzVp0iSXsYkTJ+rxxx93+1qZmZmSpOjoaJfx6Oho57HMzExVq1bN5XhgYKAqV67sPKc4SBJRpjkcDk2aNOmC/xcZgNKJv9u4VKSkpOjYsWMun5SUFF+X9ZcuySQRAACgtLDb7R5Lw2NiYiRJWVlZql69unM8KytLzZo1c55z8OBBl8+dOnVKR44ccX6+OEgSAQAAyog6deooJiZGq1atco7l5ORo06ZNSkhIkCQlJCTo6NGj2rJli/Oc1atXq6ioSK1bty72vUgSAQAASpHc3Fzt3r3b+T49PV3btm1T5cqVFRcXpwcffFBPPPGE6tevrzp16uixxx5TbGysevfuLUlq2LChunXrpiFDhmju3LkqKChQUlKS+vfvX+wnmyWaRJRxdrtdEydOZGE7cInh7zb82VdffaVOnTo53ycnJ0uSBg4cqPnz5+vhhx9WXl6ehg4dqqNHj6pdu3ZasWKFQkJCnJ958803lZSUpM6dOysgIEB9+/bVrFmz3KrDZlmW5ZmvBAAAgEsFaxIBAABgoEkEAACAgSYRAAAABppEAAAAGGgSUaY999xzql27tkJCQtS6dWt9+eWXvi4JwEVYt26devbsqdjYWNlsNi1ZssTXJQF+iyYRZdbChQuVnJysiRMnauvWrWratKm6du1q7DIPoOzIy8tT06ZN9dxzz/m6FMDvsQUOyqzWrVurVatWevbZZyVJRUVFqlmzpkaMGKHx48f7uDoAF8tms2nx4sXODYIBlCySRJRJJ0+e1JYtW5SYmOgcCwgIUGJiojZu3OjDygAAuDTQJKJM+vXXX1VYWKjo6GiX8ejoaGVmZvqoKgAALh00iQAAADDQJKJMqlKlisqVK6esrCyX8aysLMXExPioKgAALh00iSiTgoOD1aJFC61atco5VlRUpFWrVikhIcGHlQEAcGkI9HUBwIVKTk7WwIED1bJlS1177bWaOXOm8vLydPfdd/u6NAAXKDc3V7t373a+T09P17Zt21S5cmXFxcX5sDLA/7AFDsq0Z599Vk899ZQyMzPVrFkzzZo1S61bt/Z1WQAu0Nq1a9WpUydjfODAgZo/f37JFwT4MZpEAAAAGFiTCAAAAANNIgAAAAw0iQAAADDQJAIAAMBAkwgAAAADTSIAAAAMNIkAAAAw0CQCAADAQJMIwGMGDRqk3r17O9937NhRDz74YInXsXbtWtlsNh09evS859hsNi1ZsqTY13z88cfVrFmzi6prz549stls2rZt20VdBwBKAk0icIkbNGiQbDabbDabgoODVa9ePU2ePFmnTp3y+r3ff/99TZkypVjnFqexAwCUnEBfFwDA+7p166Z58+bJ4XDoP//5j4YPH66goCClpKQY5548eVLBwcEeuW/lypU9ch0AQMkjSQT8gN1uV0xMjGrVqqVhw4YpMTFRS5culfR/U8RTp05VbGys4uPjJUn79u1Tv379VLFiRVWuXFm9evXSnj17nNcsLCxUcnKyKlasqMjISD388MM6+6fgz55udjgcGjdunGrWrCm73a569erplVde0Z49e9SpUydJUqVKlWSz2TRo0CBJUlFRkVJTU1WnTh2VL19eTZs21bvvvutyn//85z9q0KCBypcvr06dOrnUWVzjxo1TgwYNVKFCBdWtW1ePPfaYCgoKjPNeeOEF1axZUxUqVFC/fv107Ngxl+Mvv/yyGjZsqJCQEF155ZV6/vnn3a4FAEoDmkTAD5UvX14nT550vl+1apV27dqllStXavny5SooKFDXrl0VFham9evX6/PPP9dll12mbt26OT/39NNPa/78+frXv/6lzz77TEeOHNHixYv/9L533XWX3nrrLc2aNUvfffedXnjhBV122WWqWbOm3nvvPUnSrl27dODAAf3zn/+UJKWmpuq1117T3Llz9e2332rUqFG644479Omnn0r6o5nt06ePevbsqW3btunee+/V+PHj3f6ZhIWFaf78+fp//+//6Z///KdeeuklzZgxw+Wc3bt3a9GiRVq2bJlWrFihr7/+Wvfff7/z+JtvvqkJEyZo6tSp+u677/SPf/xDjz32mF599VW36wEAn7MAXNIGDhxo9erVy7IsyyoqKrJWrlxp2e12a8yYMc7j0dHRlsPhcH7m9ddft+Lj462ioiLnmMPhsMqXL2999NFHlmVZVvXq1a0nn3zSebygoMCqUaOG816WZVkdOnSwRo4caVmWZe3atcuSZK1cufKcda5Zs8aSZP3222/Osfz8fKtChQrWhg0bXM4dPHiwddttt1mWZVkpKSlWo0aNXI6PGzfOuNbZJFmLFy8+7/GnnnrKatGihfP9xIkTrXLlylk///yzc+zDDz+0AgICrAMHDliWZVmXX365tWDBApfrTJkyxUpISLAsy7LS09MtSdbXX3993vsCQGnBmkTADyxfvlyXXXaZCgoKVFRUpNtvv12PP/6483jjxo1d1iFu375du3fvVlhYmMt18vPz9eOPP+rYsWM6cOCAWrdu7TwWGBioli1bGlPOp23btk3lypVThw4dil337t27dfz4cV1//fUu4ydPnlTz5s0lSd99951LHZKUkJBQ7HuctnDhQs2aNUs//vijcnNzderUKYWHh7ucExcXp8suu8zlPkVFRdq1a5fCwsL0448/avDgwRoyZIjznFOnTikiIsLtegDA12gSAT/QqVMnzZkzR8HBwYqNjVVgoOtf/dDQUJf3ubm5atGihd58803jWlWrVr2gGsqXL+/2Z3JzcyVJH3zwgUtzJv2xztJTNm7cqAEDBmjSpEnq2rWrIiIi9Pbbb+vpp592u9aXXnrJaFrLlSvnsVoBoKTQJAJ+IDQ0VPXq1Sv2+ddcc40WLlyoatWqGWnaadWrV9emTZv0t7/9TdIfidmWLVt0zTXXnPP8xo0bq6ioSJ9++qkSExON46eTzMLCQudYo0aNZLfblZGRcd4EsmHDhs6HcE774osv/vpLnmHDhg2qVauWHnnkEefY3r17jfMyMjK0f/9+xcbGOu8TEBCg+Ph4RUdHKzY2Vj/99JMGDBjg1v0BoDTiwRUAhgEDBqhKlSrq1auX1q9fr/T0dK1du1YPPPCAfv75Z0nSyJEj9T//8z9asmSJvv/+e91///1/usdh7dq1NXDgQN1zzz1asmSJ85qLFi2SJNWqVUs2m03Lly/XoUOHlJubq7CwMI0ZM0ajRo3Sq6++qh9//FFbt27V7NmznQ+D3Hffffrhhx80duxY7dq1SwsWLND8+fPd+r7169dXRkaG3n77bf3444+aNWvWOR/CCQkJ0cCBA7V9+3atX79eDzzwgPr166eYmBhJ0qRJk5SamqpZs2bpv//9r3bs2KF58+Zp+vTpbtUDAKUBTSIAQ4UKFbRu3TrFxcWpT58+atiwoQYPHqz8/HxnsvjQQw/pzjvv1MCBA5WQkKCwsDDddNNNf3rdOXPm6Oabb9b999+vK6+8UkOGDFFeXp4k6bLLLtOkSZM0fvx4RUdHKykpSZI0ZcoUPfbYY0pNTVXDhg3VrVs3ffDBB6pTp46kP9YJvvfee1qyZImaNm2quXPn6h//+Idb3/fGG2/UqFGjlJSUpGbNmmnDhg167LHHjPPq1aunPn36qEePHurSpYuaNGnissXNvffeq5dfflnz5s1T48aN1aFDB82fP99ZKwCUJTbrfKvMAQAA4LdIEgEAAGCgSQQAAICBJhEAAAAGmkQAAAAYaBIBAABgoEkEAACAgSYRAAAABppEAAAAGGgSAQAAYKBJBAAAgIEmEQAAAIb/D3EhvfbsGV25AAAAAElFTkSuQmCC

L169ZIkvfrqq4qOjtaSJUvUv39/7dixQytWrNCmTZvUsmVLSdLs2bPVoOcPTZs2TbGxscWqw+OmMSgoSGPGjDH
GR40a5e6lAAAASgVvLklOOBxyOBwuY3a7XXa73e1rZWRkKDMz07ncT5IiIiLUunVrpaWlqX///kpLS1NkZKSzQ
ZSkxMREBQQEaOPGjbrpppuKda9iNYlLly4tdvE33nhjsc8FAAC41KWmpmry5MkuYxMnTtSkSZPcvlZmZqYkKTo
62mU8OjraeSwzM1PVqlVzOR4YGKioqCjnOcVRrCaxd+/exbqYj4RUAAFDmeHM/w5SUFCUnJ7uMXUiKWNKK1
SQWFRV5uw4AAACfCfDifPOFTi2fTUxMjCQpKytL1atXd45nZWWpadOmznMOHz7s8rlTp07p6NGjzs8Xx0X9Wb4
/POUDAAAA76pdu7ZiYmKOatUq51hOTo42btyohIQESVJCQoKys7O1ZcsW5zmrV69WUVGRW3tau9OkFhYW6pFHH
tFll12mChUqaO/evZKk8ePH65VXXnH3cgAAAD5Xmv4sX25urtLTO5Weni7pj4dV0tPTtX//ftlsNoOcOVKPPvq
oli5dqu3bt+vOO+9UbGysc3lgfHy8unXrpiFDhujLL7/U559ruHDh6t///7FFfrJZuoAm8bHHHtP8+fM1depUB
QcHO8cbNWqkl19+2d3LAQAA4E82b96sZs2aqVmzZZpKk5ORkNWvWTBMmTJAkPfjjggxoxYoSGDh2qVq1aKTc3Vyt
WrFBISIjzGgsXLlSDBg3UuXNn9ejRQ23btt25dvfDCC+rcubMqVqyobdu2qU6d0vr++
++VkJCg3377za0CvCGO2XBflwDAS667N8nXJQDwkqVDW/ns3ncs30a1a782oInXru1NbieJBw4c0OtfVikqKlJ
BQYFHigIAAIBvud0kNmzYUOvXrzfG//e//GCioiK9++672rlzp
1599VUtX77cGzUCAACghLmdJPbq1UvLli3i3TJ598orCwME2YMEE7duzQsmXLdN1113mjRgAAAK8KsHnvVVa5nSR
KUrt27bRy5UpP1wIAAOATZXla2FsuqEmU/ng8eOHZL+WKfYokULjxUFAAAA33K7Sfz5559166236vPPP1dkZ
KQkKTs7W9dee63eOMNXX755Z6uEQAAwKvIEU1urOm8++67VVBQoBO7dujoOaM6evSoduzYoaKiIt19993eqBE
AAAAlzOOkce3atdqwYYYPq16/vHKtfv75mz56tdu3aebQ4AACkhDAmkSD2OlijRo1zrppDMmFhoVt/DxAXX
9tN4lNPPaURIOZo8+bNzrHHHNmzfrgQce0LRp0zxaHAAAAQEmw2bz3PqyPKdNd1cqVII+ll0fD8/Ly6R2u+7tlUIBAABQcorVJM6cOdPLZQAAAPgOO0Z0QMMtAAAAPg0+ySaitUkJiUlebsOAAAAlCIXvvJm2JOXn5+vkyZZZ3Lhxeuutt3TkyBHjeGFhYMBxnW4sFFjhoUcKAwAAKClsgWNy++ntTm5eVp2LKnkfWaDwi+HHt3GFhoUcKAwAAAKClsgWNy++nmBx98UKtXr9acOXPkQT9zHW2YEjheGFhoUcKAwAAKClsgWNy++nmBx98UKtXr9acOXPkQT9zHW2YEjheGFhoUcKAwAAKClsgWNy++nmBx98UKtXr9acOXPkQT9zHW2YEjheGFhoUcKAwAAKClsgWNy++nmBx98UKtXr9acOXPkQT9zHW2YEjheGFhoUcKAwAAKClsgWNy++nmBx98UKtXr9acOXPkQT9zHW2YEjheGFhoUcKAwAAKClsgWNy++nmBx98UKtXr9acOXPkQT9zHW2YEjheGFhoUcKAwAAKClsgWNy++nmBx98UKtXr9acOXPkQT9zHW2YEjheGFhoUcKAwAAKClsgWNy++nmBx98UKtXr9acOXPkQT9zHW2YEjheGFhoUcKAwAAKClsgWNy++nmBx98UKtXr9acOXPkQT9zHW2YEjheGFhoUcKAwAAKClsgWNy++nmBx98UKtXr9acOXPkQT9zHW2YEjheGFhoUcKAwAAKClsgWNy++nmBx98UKtXr9acOXPkQT9zHW2YEjheGFhoUcKAwAAKClsgWNy++nmBx98UKtXr9acOXPkQT9zHW2YEjheGFhoUcKAwAAKClsgWNy++nmBx98UKtXr9acOXPkQT9zHW2YEjheGFhoUcKAwAAKClsgWNy++nmBx98UKtXr9acOXPkQT9v

AAPweU6smfhMAAAAYSBIBAIDfszHfbCBJBAAAgIEkEQAA+D1yRBNJIgAAAAwkiQAAwO+xmbaJJBEAAAAGkkQAA
OD3yBFNNIkAAMDvMdtsYroZAAAABpJEAADg99hM20SSCAAAAANJIgAA8HukZiZ+EwAAABhIEgEAgN9jTaKJJBE
AAAAGkkQAAOD3yBFNJIkAAAAwkCQCAAC/x5pEEOOiAADweOytmvhNAAAAYCBJBAAAfo/pZhNJIgAAAAwkiQAAw
O+RI5pIEgEAAGAgSQQAAH6PJYkmkkQAAIBS5MCBA7r99ttVuXJlhYaGqnHjxtq8ebPzuGVZmjBhgqpXr67QOFA
lJiZq165dHq+DJhEAAPi9ANm89nLHb7/9pjZt2igoKEgffvihvvvu0z399NOqVKmS85ypU6dq1qxZmjt3rjZu3
KiwsDB17dpV+fn5Hv1NbJZlWR69Yim/8rXFQDwluzjBb4uAYCXxIQH+ezey7/J8tq1b2gUXexzH3roIX3++ed
av379WY9blqXY2FiNHj1aMkSQdO3ZMOdHRmj9/vvr37++++RmiWSRAAAAK9yOBzKyclxxeTkcjrOeu3TpUrVs2
VL/+Mc/VK1aNTVr1kwvvfSS83hGRoYyMzOVmJbm0bppEgEAgN+zefE/qampioIcHmlpqa
etY69e/dqzpw5qlevnj766CPde++9uv/++7VgwQJJUmZmpiQpQp0to1nYyOjnYe8xSebgYAAPCilJQUJScnu4zZ7
faznltUVKSWLVvq8ccflyQ1a9ZM33zzjeb0naukpCSv1/pnJIkAAMDv2Wzee9ntdoWHh7u8ztUkVq9eXQ0bNnQ
Zi4+P1/79+yVJMTExkqSsLNcllFlZWc5jnkKTCAAAUEq0adNGO3fudBn74YcfFBcXJOmqXbu2YmJitGrVKufxn
Jwcbdy4UQkJCR6the1mAADg99zdqsZbRo0apWuuvvVaPP/64+vXrpy+//FIvvrrj27q9EqIbKrWnPzY0SfB+bDBM
evXqqXbu2xo8fr9jYWPXu3dujtdAkAgAAlBKtWrXS4sWLlZKSoilpqh27dqa0XOmBgwY4DznwQcfcfVF5eno6YOH
ars7Gy1bdtWK1asUEhIiEdrYZ9EAGUK+yQCly5f7pP40Xe/e03aXRtW9dq1vYkkEQAA+D3+drOJB1cAAABgIEk
EAAB+z1ZKHlwpTUgSAQAAYCBJBAAAfi+AINFAkggAAAAADSSIAAPB7rEk0kSQCAADAQJIIAAD8HvskmmgSAQCA3
2O62cROMwAAAAwkiQAAwO+xBY6JJBEAAAAGkkQAAOD3WJNoIkkEAACAgSYRZcKWzZs04r57lNixrpV+rV33
icvyTlR/rnOPuUvtrW6vJVfX1/Y4dPqoUgLten/eSht55i7p1uEa9urTXv8fcr/O/Zpz1XMuyNPb+e9ShVSOtX
7OqhCvFpcxm896rrKJJRJlw4sRx1a9fX+fXykPTzzn8WbNmmtk8pgSrgzAxdq2dbNu+setmv0fRXr62Rd16lSBxow
YqhMnjhvnVvv3f12Qry/9Y4dPqoUgLten/eSht55i7p1uEa9urTXv8fcr/0/Zpz1XMuyNPb+e9ShVSOtX
7OqhCvFpcxm896rrKJJRJlw4sRx1a9fXykPTzzn8WbNmmtk8pgSrgzAxdq2dbNu+setmv0fRXr62Rd16lSBxow
YqhMnjhvnvv3f12Qry/9XFyhDWJOIMqFtuw5q267DOY/3vLG3JOnAgZ9LqCIAnvLU7Bdc3qdMfEy9urTXDzu+U
5PmLZ3ju3Z+r7cWLtALC95Un+4dS7hKXOr4fz1MNIkAgFIlNzdXklQxPMI5l5p9/Qo+Mf1AjH/y3Klep4qvScAk
LIKE2lOrp5p9++kl33XXXec9xOBzKyclxxeTkcjhKqEADgSUVFRXp2+hNq3KSZ6tSt5xx/dvpUNbq6/LsPq
wP8S6luEo8ePaoFCxac95zU1FRFRES4vJ6c8ngJVoiaAXj0QGI8/PH2EDKKsvrt8O8ZGC02D0v0OWf1ORfN5GTb3bh
Lr7LKp9PNS5cuPe/xvXv3/uU1UlJSlJyc7DJmleFNfVF0AgJI3c+pjSlu/VrNfXKBq0THO8a2bN+rgzz/phr8nu
Jw/YdwoXd20uZ55YX4JVwr4B582ib1795oZNpNlWec856+dvZeWbHlXZvC/FMeKQ8AUAIsy9Iz
Tz2u9WtW6bbvbLXZvC/FMeKQ8AUAIsy9IzTz2u9WtW6Zm
581T9sstdjt+WdLeu79XXZWzQrTdp2KgH1aZdxxKsFJe0shz5eYlPm8Tq1avr+eefV69evc56PD09XS1atjq
lAaHc/L0/79+53vD/z8s77fsUMRERGqHhurY9nZ0nTokH755bAk/v8dalSpVVKVqVZ/UDKB4Zjz5qFZ99IE
emzZLoeXDdOTXkRJWrVDnrwyrRMdWNhhKA5/iOSWzRooW2bNlyzibxr1JGG+I9vv/1Gdw+60/l+2
tQ/1p3e2OsmPfL4E1rz6WpNeDjFeXzcmFGSpHvw6WIBuOW9d96UJD1wzyCYX8YcmPKruPXv7oCL4I/4
sn8lm+bALW79+vfLy8tStW7ezHs/c0598c7G6abgUtX9vECX5cAwEtiwoN8du+Ne4557dqtr4j46
5NKIZ82id5CkwhcumgSgUuXL5vEL/d6r0m8pk7ZbBLZTBsAAPg9JptNpXqfAAAAPgGSSIAAABRooEkEQAAAAa
SRAAA4PfYAsdEkggAAAADSSIAAPB7f/FXgP0SSSIAAAAMJIkAAMDvESSaaBIBIBAADoEg1MNwMAAMBAkggAAPweW
+CYSBIBAABgIEkEAAB+jy1wTCSJAAAAMJAkAgAAv0eQaCJJBAAAgIEkEQAAgCjRQJMIAAD8Hlvg mJhuBgAAgIE
kEQAA+D22wDGRJAIAAJRSTzzxhGw2m0aOHkcy8/P17Bhw1S5cmVVqFBBffv2VVZWlsfvTZMIAAD8ns2Lrwu1a
dMmvfDCC7r66dxkeNGqVly5bp7bff1tq1a3Xw4EH16dPnIu50djSJAAAAMXuRw0JSTk+Pycjgc5/1Mbm6g
YoJdeekmVKlVyjh87dkyvvvPKKpk+frr///e9q0aKF5s2bpy+++EIbNmzw6G9CkwgAAPweWwFIAC6a8LE55vpbw0/
dMmvfDCC7r66qtdxkeNGqVly5bp7bff1tq1a3Xw4EH16dPnIu50djSJAAAAMXuRwOJSTk+PycjgcS5/1Mbm6g
YoJdeekmVKlVyjh87dkyvvvPKKpk+frr///e9q0aKF5s2bpy+ufWOOUUEbffbQwAAAAF6MElNTUxURGHyS
W85w4YNO/XXX6/ExESX8S1btbqigoMBlvEGDBpqZs6bSbdS0tIu8kdwxYMrAAAAApSSkqq8eXbz7/3/nHmBRJH
dulWbNmOjyjmVmZCg4OVmRkpMt4dHS0MjMzPVoHTSIAAPB7pWVX2eXLl+uBBx7QypUrFRIS4tNamG4GAAB+z5s7
4FzMFjPbtm3T4cOH1bx5c5UrV07lypXT2rVrNWvWLJUrV07R0dE6efKksrOzXT6XlZWlmJgYj/4mJIkAAAClRKd
OnbR9+3aXscGDB6tBgwYaO3asatasqaCgIK1atUp9+/aVJO3cuVP79+9XQkKCR2uhSQQAAH6vlMw2q2LFimr
Uu3F80XPC4oyUwq4KkKb3uzBZAMCx62DDdWpBZFUyZLMeaIaWgTADAOdSA1DIYN5E+ZS8HOA7MNJG+ZOv8XWTj
JDeekmVklVyjh87dkyvvvPKKpk+frr///e9q0aKF5s2bpy+ufWOoUSEbmhCgPXtiVcp

7neax6qVLjw1X2//PKLqlWrprVr16p9+/Y6duyYqlatqkWLFunmm2+WJH3//feKj49XWlqa/va3v3mqbNYkAgA
AeHOfRIfDIYfD4TJmt9tlt9v/8rPHjh2TJEVFRUmStmzZooKCAiUmJjrPadCggWrWrOnxJpHpZgAA4PdsXvxPa
mqqIiIiXF6pqal/WVNRUZFGjhypNm3aqFGjRpKkzMxMBQcHKzIy0uXc6OhoZWZmevQ3IUkEAADwopSUFCUnJ7u
MFSdFHDZsmL755ht99tln3irtvGgSAQCA3/PmdHNxp5b/bPjw4Vq+fLnWrVunyy+/3DkeEx0jkydPKjs72yVNz
MrKUkxMjKdKlsROMwAAQKlhWZaGDx+uxYsXa/Xq1apdu7bL8RYtWigKEirVq1yju3cuVP79+9XQkKCR2shSQQ
AACglhg0bpkWLFum9995TxYoVnesMIyIiFBoaqoiICA0ePFjJycYSEBI8+tCKxBQ6AMoYtc
IBLly+3wPnteKHXrl2pfLlin2s7x7z3vHnzNHDgQEl/bKY9evRo/fe//5XD4VDXrl31/PPPPe3y6mSYRQJlCkwh
cunzZJGaf8F6TGBla/CaxNGFNIgAAAAysSQQAAH7P5vYf0Lv00SQCAAC/58OtcMoqppsBAAAgIEkEAAB+jyDRR
JIIAAAAAOkiAAAAUaKBJBEAAAAGkkQAAOD32ALHRJIIAAAAAOkiAADwe+yTaCJJBAAAgIEkEQAA+D2CRBNNIgA
AAF2igelmAAAAGEgSAQCA32MLHBNJIgAAAAwkiQAAwO+xBY6JJBEAAAAGm2VZlq+LAC6Uw+FQamqqUlJSZLfbf
VOOAA/i3zfgWzSJKNNycnIUERGhY8eOKTw83NflAPAg/n0DvsVvMwAAAAwOiQAAADDQJAIAAMBAk4gyzW63a
EiSxqBy5B/PsGfIsHVwAAAGAgSQQAAICBJhEAAAAGmkQAAAAYaBIBAABgoElEmfbcc8+pVq1aCgkJUevWrfXll
1/6uiQAF2nduntn9Xq2bOnYmNjZbPtWZtFPKTMzU02bNtWsWbPPUunrVr+XBPPgUd9TfQq1GAXBzUhKQoRmwY
zpg8rA/wLW+CgTHv22Wf11FNPKTMzU02bNtWsWbPUunVrX5cF4CKsWbNGnTp1MsaTkpI0YhLhkBKaRRTpM
Imf9rlKSkpSy5Ytdc0112jmzJnKy8vToEGDfF0agIuw4AuFDJyclKSkpSy5Ytdc0112jmzJnKy8vToEGDfF0agIuQm5ur3bt3099nZGQoPT1dUVFRqlm
zpg8rA/wLW+CgTHv22Wf11FNPKTMzU02bNtWsWbPUunVrX5cF4CKsWbNGnTp1MsaTkpI0f/78ki8I8FM0iQAAA
DCwJhEAAAAGmkQAAAAYaBIBAABgoEkEAACAgSYRAAAABppEAAAAGGgSAQAAYKBJBAAAgIEmEcBFGzhwoHr37u1
837FjR40cOLE61izZo1sNpuys7PPpeuys7PPeY7NZtOSJUkfc1JkyapadOmF1XXjz/+LHeeqaEjz3apGbnQkn
OBA2Ww22Ww2BQcHq27du2dupoyZZYpn9rMbUvs7Fv6qPi6ul4EU2HB2yNYPCUVE18PjMoHSOYCgdNknUR
bNkxBQUFKSUkxzj158qCg4M9ct+oqCiPmrPE3TP4YAcA4DskicAlzG63iHRMs/YR5sfnWcTbyUaVWYv/l3qWVbb/3OosSUaqhBMAJLq3DNU926dtqpJ
O1ToBo2MhdpUTfT/8l7F/cSn1GAlMqcsMsPo8sNtyxAaCwQj+tGF4aJjFZpPp2QlRcYIdIbSwXuOdrHkQl5o0e6fEPetswnH2mcs6ubA/3HCFPkTr0r4
pN+Lla7Vk/RRvfeT/b3T2bRbgVSdSA6dWymp9XdsBmhnVB5WrnvU6G3pPoQSg3V+yjQYP/FGJG5OxE26Zh7JuD1o2LsyEsqzXHh3l5VuT9cLNlaUmZ/b
MvWtVjjD9i0zZvKlqkYFPdYnEbENmCd22bYaNdYy+KpTpdlHjXi1IbnRCYMgM5aO0KmIAqcNZXaRYHMeC0e7zU
gRUM9KaiatKU2ka3eKZsTajBoUXuiXpWkZejIdKFEk03wewv5WMhqLOKZX1qq5qTeF1FhR3UKl6pRwjv4KzMvHoVuVlI4f1cYUhO6jNhJEtxvEgfQ93
yc8xiCm/v9VfevbuJ/8rtk8Km3dZXfCFtZ9aU7nQqzdrMu28JDJ6MWz55HLwIFs14knxGALRXdhsVuF4TepvVpyP
OR0ALHD9xTFlSM33Np3xwHUUP74nwWcGzVqhP0aSRu/HTxKBDANEOVXtYwWyIU/0u7u0hcqOx3gebqXlfXBwu
pvkrC7sHrBvt1ZYrKG+FuPI6FoJv8xDoqvwd1Pv6VBSYbb1iwy9yaTtFSQ8NLL8DLxtOXsBs6Wy9yjrbUxx1dl
GvVQMFB1OHXRI2qTRqdqaTGIdT4ea+1c4QCsH5fmzjWvJ0fP4BmV2Pde9TXF49BbthBVytOsKe9TOr+rcMrkhJ
zeEFX6iqgbAPfdAlYHbxPW9GpBB+QqmXrtZyoH8Y/2OajRoUq9Hdtp/r5R5e67ODjMxj0XPmHuLgODS/W8/Nt
Mb/mBnZjY6ZVMuSebqOqtpMXi1gaPUZmNcUedtwGD6BGHSIPvKZdtPtLlv4l9rGSBtWxhpLDyNlumZsg25CtkM
IcFBRaEwWUF+nHjAIkN6BLyNoEGHqqStS0G0KT6ObPl9Vb0sL5bx2HlJPwoDtQNuxSbaPkvJo4/yUeMdHKTSQu
aZ2F0JI8qsMQnXeLmNNYZOHIjRuGFoWtWJuLsrZaiI28lxi6PmGS7G7V7A5c1ja5Zb88/g7hl6k9lNJjg5jKKTW
WWf/xgU4ka3nqaebK/q+zfYlXnlGmd1bcC+B3pZzbj8u4wdm1hJ9LHLxBtk6oPpO/CqSLxj0qa+yIWt8bsaXqN
iTN0UnaLN0GF5iKpltbKTMVdGKmy+O7rLcRFSPCk0nHh0ydmejJdtMAjXpT5MlFDb8u6EhbaLyyjHTnWmP0/9JXR
JwVt6ps/dCgt++p7Fq6ljrzpp8CrKMKqXrRyLnoozvQJ1gU7TqRVHL7++z5GHoYi9CHc2xqpBpyqs8g4WwB2WHW
I1eqRMmlWFyxBfmLXwq4fOqZ+zZ8OFRJtwdOZDp1kZAj0TMDB4t1VfUSz1F1J1Kr1ZzDL2X+FKOsLT2uQmCQh1d
M3rT8l2DkUhHUOOKCVXLgeDsbUHJA8B5+PTGt10gmXmtKZxE/8lPWFFcVLgTzyRmlJ1UdaXbyY+o53fj0NsZGVN
s8LWnS0VNbhV3m3/Yc0Z2UzdrDJ2cv7uF17cgZPsdpuZqQ97Sxp6Z0DSWkfHvnKlbv92PMfOX5bmtf5V0FOBDd4
t3T/j5m0HW/u4/MLIb//+a7rs7Hr3WV9l9xd9zr9HI/qAS9c/Dr9YS0MfRbkPo48fA7rq17ZduP4L6ZYm2sNm+3
zJfKfZk7e+SCvtE0NZ4Gb4yVAGt+rI/7KWh1bf2qSdmldOB2nJKGG6Kh1SPmvAZLhNVBVOC4tdlQldAVfpUKLw7
Pa/tfEO9zcCfG6R/3Zkl2Nr2/sZx/1QfNZ8uiXG1GGr/TPIHffUv/cb8TaFb3Z4uYwzd/2AfXdAqUsgKtvVM6A2
+98fX98/7v+7zLSbtnrdrXvgN+7dmQqB9Pfj3Nkb/LNcrnzVkdZy35EOZ+r3vv3ZDsN3hwS63pQzrKZ4yB07O+

```
vv6/atWtL+mOd4DvvvKMlS5aoSZMmmjt3rh5//HG3vu+NN96oUaNGafjw4WratKk2bNig8ePHG+fVrVtXffrOU
Y8ePdSlSxddffXVLlvc3H333Xr55Zc1b948NW7cWBO6dND8+fOdtZ4pODhYKSkpuvrqq9W+fXuVK1dOb7zxhlu
1A4C32axzrTYHAACA3yJJBAAAgIEmEQAAAAaaRAAAABhoEgEAAGCgSQQAAICBJhEAAAAGmkQAAAAYaBIBAABgo
EkEAACAgSYRAAAABppEAAAAGP4fgKZliyOEvmcAAAAASUVORK5CYII=\n″
          },
          ″metadata″: {}
        }
      ],
      ″source″: [
        ″from sklearn.linear_model import LogisticRegression\n″,
        ″\n″,
        ″lr = LogisticRegression()\n″,
        ″model_train_eval(lr,\″LogisticRegression\″)″
      ]
    },
    {
      ″cell_type″: ″markdown″,
      ″metadata″: {
        ″id″: ″lnZZKmzrGk8e″
      },
      ″source″: [
        ″## Gradient Boosting″
      ]
    },
    {
      ″cell_type″: ″code″,
      ″execution_count″: null,
      ″metadata″: {
        ″id″: ″5ZFnlXCBGj-B″,
        ″colab″: {
          ″base_uri″: ″https://localhost:8080/″,
          ″height″: 980
        },
        ″outputId″: ″52d44f88-aa14-4f8c-bfcf-02ec2d531977″
      },
      ″outputs″: [
        {
          ″output_type″: ″stream″,
          ″name″: ″stdout″,
          ″text″: [
            ″Training Accuracy: 0.9192245557350566\n″,
            ″\n″,
            ″Classification_Report -->\n″,
            ″\n″,
            ″              precision    recall  f1-score   support\n″,
            ″\n″,
            ″         0.0       0.90      0.96      0.93       275\n″,
            ″         1.0       0.37      0.20      0.26        35\n″,
            ″\n″,
            "    accuracy                           0.87       310\n",
            "   macro avg       0.64      0.58      0.59       310\n",
            "weighted avg       0.84      0.87      0.85       310\n",
            "\n",
            "Cross Validation Classification_Report -->\n",
            "\n",
            "              precision    recall  f1-score   support\n",
            "\n",
            "         0.0       0.88      0.93      0.90      1098\n",
            "         1.0       0.93      0.87      0.90      1123\n",
            "\n",
            "    accuracy                           0.90      2221\n",
            "   macro avg       0.90      0.90      0.90      2221\n",
            "weighted avg       0.90      0.90      0.90      2221\n",
            "\n"
          ]
        },
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 800x600 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAAAokAAAIjCAYAAABvUIGpAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAABAQ0l
EQVR4nO3deZzNdf//8eeZYc4MZjGYrRhbtmwluSbZsm/ZSqIMWVqGyiBNV3Y114UoFCOuI1HqKorWQbbLUJaRJ
BlEYqwZDRlj5vP7o5/z7XiTOcxxZpzH/bp9bjfn8/mcz+d1Tper1/V8vz/vY7MsyxIAAADwFz6eLgAAAAAFD00
iAAAADDSJAAAAAMNAkAgAAwECTCAAAAAANNIgAAAAw0iQAAADDQJAIAAMBAkwgAAAADTSKAv7Vr1y61atVKwcHBs
tlsWrx4cb5e/+eff5bNZlNUNSU1K+Xrcwa9q0qZo2berpMgB4OZpEoBDYvXu3Hn3OUVWsWFH+/v4KCgpSw4YN9co
rr+iPP/5w671jY201lbds2vfDCC5o3b57uuOMt97veurTp49b57sNptNkydPdvn6Bw8e1
JgxY5SampoP1QLA9VXE0wUA+Huffvqp7r//7//ftntdvXu3Vs1a9bUuXPntHbtWg0fWg0fPlzbt2/XG2+84ZZ7//HHHHOp
JSdE///lPDRo0yC33iI601h9//KGiRYu65fpXUqR9+7dnY7Nnz9f/v7+Onv27bFVd++DBgxo7d
qzKly+vunXr5r/l9X3311VXdDwDyE00iUIDt3btXvXr1UkxMjL/aMKjHsbi4OKWlpenTTz912/2PHHnh9OKWlpenTTz912/2PHj0qSQo
JCXHbPWw2m/z9/d12/TW0m/z9/d12/Sux2+1q2LCh3n68zz4q22LCh3n33XaNJXLBggdq3b68PP/fflfgDwdxhuBgqwi
RMnKjMzU7Nnz3ZqEC+oXLmynnrqKcfr8+fPa/z48apUqZLsdrvKly+v5/7v+fPOOfPx+8vuz+f6gh8CyGgl37vN1NLoJ7A4HBqvV2KqWjk+5gWvH4M


















RMnKjMzU7Nnz3ZqEC+oXLmynnrqKcfr8+fPa/z48apUqZLsdrvKly+v5557Tl1ZWU7vK1++vDp06KC1a9fqzjv
vlL+/vypWrKi3337bcc6YMEUHR0tSRo+fLhsNpvKly8v6c9h2gt//NmulZHhsY8bIZrM57UtOTtbdd9+tkJAQ1lShRQ
lWrVtVzzz3nOH65OYkrVqxQoOaNVLx4cYWEhKhTpMHBwerbt6/NnDlqZ9+S/2Ij1
79tTnn3+ukydPOvZ9++232rVrl27Gmcf+LECQ0bNky1atVSiRIlFBQUpLZt22rr1q2Oc1auXXn69etLkvr27esLkvr27
esYtr7w0Zs2baqaNWtq06Z6ZNaty4sYoVK+b4b/f3j87+b4Xi6ekxgbGyt/f3/j87du3VolS5bUwYMH8/xxCvaBKBAmzJkiW
qWLGi7rrrrjyd379d379/f40aNUq33367pk6dqiZNmigz9ihLiLX0fg8gvX/ZJp8J4zWX2n8e99e73iY3KQ3rVZfGNkQo0pTp/dgtf5rfm1aHF2+Vp9hbfzW
ZLUtWtXTZ06VZLO4IMPat68eXr55fddqn/79u3b9/d2jKeHs33jt25TvZeZIeOeJ0jefvpLBPjTb1+3F8tdVKcKrgt4Wq7bN8ODQMG2t8HpC7
cOzgxYxQfH69169apYcOG+vnnn843jzu3fvrt9//12JiYnL93sn2kuhtG11AAhmdU8Nbd/2OFPe8Ut5pdN2+Tg6C4WQpWjVK9f3nVFs1F1a
tVO++23G+fv2bNHHixcvVocOHTX5c27ZtU5NTmTRwNW/Xq1+TV9dvuAtAMjAg9Hmzd
jatu2rerWrauXX35zZZo1u2R9r7zyisqUKaPYmY7WFjl5ORIkl5//XV99dXdVXmj59uqKiovL8wQEgzywABVAoYly
erUqVOezk9NTbUkWf3793faP2zYMEuS9dVXXznn2xevVzL9xdDyHrg3I2PfxRcGBc7h3vcGjjpnqGPf3r17DL9zEK8unWneWkf9zLpv
ZGxtrRUdHGzwWMHj3a+uv+uv+zI2tJGVfV3deDgo2Z4IwDtyd+4ghKnC8i4jq30hOrMUWKnpyM8dz
4884nTNLl26WKVK1brsPf/6OYXL25ZlmXdd999VrPiqvcaKNmmDQjd9cN3Vv7PmzS3LsqcnBwrIiCGjt27CW/g7Nnz1ooo5OTG57Db7da




4ceMc+7799lvjs13QpEkTS5L1a9asSx9r0qSJ1WiHSx5RHv5y74xy9Y02Ea6Q51vtU7DIm1/33+b2OT5O0ldA5BLiH42irYRpWLnhRd8AthSTCKc8pBEx1bxZdPsK6

```

VOSpMDAwDyd/9lnnOmS4uPjnfYPHTpUkoy5izVq1FCjRoOcr8uUKaOqVatqz549V13zxS7MZfz444+Vm5ubp/c
cOnRIqamp6tOnjOJDQx37a9eurZYtWzo+51899thjTq8bNWqk48eP077DvOjZs6dWrlyp9PROrVixQunp6Zcca
pb+nMfo4/Pn/3zm5OTo+PHjjqHOzZs35/medrtdffv2zdO5rVq1OqOPPqpx48apa9eu8vf31+uvv57newGAq2g
SgQIqKChIkvT777/n6fx9+/bJx8dHlStXdtofERGhkJAQ7du3z2l/uXLljGuULFlSv/3221VWbHrggQfUsGFD9
e/fX+Hh4erRo4fef//9v2OYL9RZtWpV41j16tV17NgxnT592mn/xZ+lZMmSkuTSZ2nXrp0CAwz0lcOFCzZ8/X/X
r1ze+ywtyc3M1depU3XLLLbLb7SpdurTKlCmj7777ThkZGXm+5OO33eTSQyqTJO9WaGioUlNTNW3aNIWFheX5v
QDgKppEoIAKCgpSVFSUvv/+e5fed/GDi1sFAq1ev1rJly/Twww/ru+++0wMPPKC
WLVsa516La/ksF9jtdnXt2lVz587Vok
suWLTpy5Igkadu2bS69FwBcRZMIFGAdOnTQ7t27lZKScsVzo6OjlZubq127djntP3z4sE6ePO14Ujk/lCxZOul
J4AsuTislycfHR82bN9eUKVPOww8/6IUXXtCKFSv09ddfX/LaF+rcuXOncezHH39U6dLPz48Wv7AJfRs2dPb
dmyRb///vslH/a54L///a+aNWum2bNnqOePHmrVqpVatGhhfCd5bdjz4vTpO+rbt69q1KihgQMHauLEifr222/
z7foAcDGaRKAAe+aZZ1S8eHH1799fw8fNo7v3r1br7zyiqQ/hOslGU8gT5kyRZLUvn37fKurUqVKysjjIOHfff
efYd+jQIS1atMjpvBMnThjvvbbCo9MXL8lwQGRmpunXrau7cuU5N1/fff6+vvvrK8TndoVmzZiZo/frxmzZJihiIi
Iy57n6+trpJQffPCBfv31V6d9F5rZSzXUrhoxYoT279+vuXPnasqUKSpfvrxiY2Mv+zOCwLViMW2gAKtUqZLWL
FigBx54QNWrV3f6xZW1tvevrxiY2Mv+zOCwLViMW2gAKtUqZIWL
FigBx54QNWrV3f6xZV169bpgw8+UJ8+fSRJderUUWxsrN544w2dPHlSTZoOTfffKO5c+eqc+fOl11e5Wr06NF
DIOaMUJcuXfTkkO/qzJkzmjlzpqpUqeLO4Ma4ceOOevVqtW/fXtHROTpy5Ihee+O13Xzzzbr77rsve/1Jkyapb
du2iomJUb9+/fTHH39o+vTpCg4O1pgxY/Ltc1zMx8dHzz//yZM66e+ffvqrrvu0rZt2zR//nxVrFj
R6bxKlSopJCREs2bNUmBgoIoXL64GDRqoQoUKLtW1YsUKK+fbaaxo9erRjSZ45c+aoadOmGjlypCZOnOjS9QAg
zz8dDWAPPjpp5+sAQMGWOXLl7f8/PyswMBAq2HDhtb06dtb2fPOs7Lzs62xo4da1WoUMEqrSoVbZsWShcH
pHMv6cwmc9u3bG/e5e0mVyy2BY1mW9dVXX1k1a9a0/Pz8rKpbU6d0llRUVGWn5+fFRUVZ
T344IPWTz/9ZNzj4mVili1bZjVs2NAKCAiwgoKCrI4d0lo//PCD0zkX7nfxEjl79+697Hdq
O5ZbAGTp0qBUZGWkFBARYDRs2tFJSUi65dM3HH39slahpSpIjT52zSpIl16623XvKef73OqVOnr0joaOv22
2+3srOznc4bMmSI5ePjY6WkpPztZwCAq2GzLBdmdgMAMAArMCcRAAAABppEAAAAGGgSAQAAYKBJBAAAgIEmEQA
AAaaRAAAABhoEgEAAGC4IX9xJeC2QZ4uAYCbHF0/3dMlAHCTEvb8+71zV7mzd/hjywy3XdudSBIBAABguCGTR
AAAAJfYyMOuRpMIAABg89xQd0FF2wwAAAADSSIAAADDzQa+EQAAABhIEgEAAJiTaCBJBAAAgIEkEQAAgDmJBr4
RAAAAGEgSAQAAmJNooEkEAABguNnANwIAAAADSSIAAADDzQaSRAAAABhIEgEAAJiTaOAbAQAAgIEkEQAAgDmJB
pJEAACAAiIxMVH169dXYGCgwsLC1LlzZ+3cudPpnKZNm8mszltjz32mNM5+/fvV/v27VWsWDGFhYVp+PDh0n/
+vEu1kCQCAAAUkDmJq1atrXr6/z58/rueeeU6tWrfTDDz+oePHi3jvnR6sKDrvGH0zb8eecnBy1b
99eERERWrdunQ4dOqTevXurSfvYfTVDz+oePHijvMGDBigcePGOV4XK1bM8eecnBy1b
99eERERWrdunQ4dOqTevXurSfvYfTVDz+oePHijvMGDBigcePGOV4XK1bM8eecnBy1b
99eERERWrdunQ4dOqTevXuraNGievHFF/NcCOOiAABAARlu/uKLL5xexeJyUlKSwsTJs2bVLjxo3lKVsTJs2bVLjxo0d+4sVK6aIh
LXuOrr77SDz/8oGXLlik8PwZ/5eDU1/4PGjL+KinoVRaNyPm9gaCbo/LCRmNQbtaWxuowSJAdgOpdOOmGMh3vowgYtxSbi4adF9Ec6HmBxbZXVr
LXuOrr77SDz/8oGXLlik8PFx169bV+PHjNWLECIOZMOZ+fn55qqVgtMOAAAA3qKysLJO6dcppy8rKyrN7I
EmhoaFO++fPn6/SpUurZs2aSnh4uHwbcm2lJZCYQIQG7dQ9kAEdTmX1tOdUxSNIa3fuhJNtk7Sn1kOstSpWU
Cg4OdtsTExCuWlJubq6efflo1NGzZUzZo1Hft79uypYPHj5T1o/7+OtRvoQlGu1X3XmMpO7fvRgNpFUS+pTdl/N9SXYSRWwdM2gzkW4hY8hHU
H8lDDcDAAC4UUJCguLj45ayzcz45S5OcuaUDRdQGbeclZ8c2d6uJN83ELj9dfUCvBUrhhqjQJSkRqVuobRbfrRA6Eh53D7rnD1xJ6H
EAABw64Mrdrs9T03hXw0aNOeq/kPs+nZ/Kt7ENI7r9tGzhWtUWnmEmmDmCiBKfpF0F8uiWwOFKG3VIvgMF5AnRUvi5R0qLEk5xdXmf4f9yuQ2
WYAAIACwrIsDRo0SIs5OSIsWLdKKFStUoKFK74nNTVVkhQZGSljAhTIaMUnIgijh0f0LEqCVf1nDCKT0ToHBLOrOIEmBGQGHlonkLoElxNyoIGCoRWg7wnAMWiQ3sA
AgE/BeLo5Li5OCxYsOMcff6zAwEDHHMLg4GAFBAy+1GSxpuMJYBZgyxKOoGI6bYDmuWLFMPAtCLYl2ZuP7GGBsPO6FKARK43u5bUmydpgFwRlKzFkwRRnJGx3HlkBYzkDpMJgtAIcMkSzc5A2Qx8oIvr9BPJSpoEEBFAM8mCTBtRjPDIeRdgFLH0AUtDO2AEHw
SvVqFFDDz/8sCZOnKjO9HQ9//zzioucLynRJEkEAAAoIGbOnKmMJjAw1bdpUkZGGRjm3hwoWSJD8/Py1btkytWrV
StWrVNHToUHXr1k1LlixxMPX11dLly6Vr6+vYmBJi9NBDD6l37379bdzsBUkiAABAAV1M27qetykWrIpaJTEm6Wlv
orXiY6O1meffXZNtdkAkAgAAFJDFtAuSgtsKHd8IAAAADSJAAAADASJIAAADDzQaSRAAAABhIEgEAAJiTaOAbAQAAgIEkEQAAgDm
IAADAnEQD3wgAAAAAAMJIkAAADMSTTQJAIAADDcbOAbAQAAgIEkEQAAgOFmAOkiAAAADCSJAAAAzEk08IOAAADAQ
JIIAADAnEQDSSIAAAAMJIkAAADMSTTQJAIAANAkGvhGAAAAYCBJBAAA4MEVAOkiAAAADCSJAAAAzEk08IOAAAD
AQJIIAADAnEQDSSIAAAAMJIkAAADMSTTQJAIAADDcbKBtBgAAgIEkEQAAeDObSaKBJBEAAAAGkkQAAOD1SBJNJ
IkAAAAwkCQCAAAQJBpIEgEAAGAgSQQAAF6POYkmmkQAAOD1aBJNDDcDAADAQJIIAAC8HkmiiSQRAAAABpJEAAD
g9UgSTSSJAAAAMJAkAgAAECQaSBIBAABgIEkEAABejzmJJpJEAAAAGEgSAQCA1yNJNNEkAgAArOeTaGK4GQAAA
AaSRAAA4PVIEk0kiQAAADCQJAIAABAkGkgSAQAAYCBJBAAAXo85iSaSRAAAABhIEgEAgNcjSTSQJAIAAK9Hk2h
iuBkAAAAGkkQAAACCRANJIgAAAAwkiQAAwOsxJ9FEkggAAAADSSIAAPB6JIkmkkQAAAAYSBIBAIDXIOkOOSQCA
ACvR5NoYrgZAAAABpJEAAAAgkQDSSIAAAAMJIkAAMDrMSfRRJIIAAAAAOkiAADweiSJJpJEAAAAGEgSAQCA1yN
JNNEkAgAAOCMaGG4GAACAgSQRAAB4PYabTSSJAAAAMJAkAgAAr0eSaCJJBAAAgIEmEQXOsEdaae07w3Vk7WTtW
56o96cMOC3RYcZ5DWpX0OevD9axdS/p8OevD9axdS/p8JpJSp79tPztRR3HP3j5Uf302Tj9tn9q9nz1gmaP763IMsHX86MAyIP

```
NG7/V04MeU+vmjVSvdjV9vWKZ41h2dramTZ2s7l07quGdt6l180Ya9dwIHT1y2IMV40Zks9nctrkiMTFR9evXV
2BgoMLCwtS5c2ft3LnT6ZyzZ88qLi50pUqVUokSJdStWzcdPuz8d2L//v1q3769ihUrprCwMA0fPlznz593qRa
aRBQ4jW6vrFkLV6tJ78nq8PgMFSniq6UzB6mYv5/jnAa1K+jjGU9o+fof1eihSbr7oUma9d4q5eZajnNWf/uTH
hrxH9XpMk49h7+limVLa8Gkfp74SAD+xh9//KEqVatpxHOjjGNnz57Vjzt+UP9Hn9D8hR9q8pTp+vnnvRry5BM
eqBRwv1WrVikuLk7r169XcnKysr0z1apVK50+fdpxzpAhQ7RkyRJ98MEHWrVqlQ4ePKiuXbs6jufk5Kh9+/Y6d
+6c1q1bp7lz5yopKUmjRpl/x6OzbIs68qnFS4Btw3ydAnIR6VLltAvK/6lFv2m6n+bd0uSVs0dquUbftS41z7
N83XaN6ml96cMUHCDp3X+fK67yoWbHV0/3dMlwI3q1a6myS/PULN7Wlz2nO3fb1Pvnvdr6ZcrFBkZdR2rg7uVs
HtuXmCFp/P+7xNX/fjvFsrKynLaZ7fbZbfbr/jeo0ePKiwsTKtWrVLjxo2VkZGhMmXKaMGCBbrvvvv+vP6P6p
69epKSUnRP/7xD33++efq0KGDDh48qPDwcEnSrFmzNGLECB09Gx44d08SJE9WlSxfFxMQoJ
iZGXbp00aRJk3T06FFPloYCJKiEvyTpt4wzkqQyJUvoztoVdPREpr5OitfPy17UV289pbvqrzsNUoGFVOPtnd
o/da9NIhAIZeZ+btsNpsCA4M8XQpuJDb3bYmJiQoODnaEhMT81RStkFi//7P1RWRkaGaJCk0NFSStGnTJmVnZ6tFi//7P1LVq
lVTuXLllJKSIklKSUlRrVq1HA2iJLVu3Vqn/p3S9u3b8/yVeKxJ/Pbbb1Wl1WlShVNmzZNwcHBaty4sRo3bqzg4GB
NmzZN1apV08aNG694naysLJ06dcpps3Jzrsm3MnwPVGs9k0adh9Wrdlt37YfUiSVOHm0pKkfz7aTv/5aJ06xb2m1
B2/6LPXB6tSuTJ075/wZCcdW/eSDq6aqLKRobp/yBvX/TMAyD9ZWVmaNnWyWrdtrxIlSni6HCBPEhISl
lJCRc8X25ubl6+umn1bBhQ9WsWVVOSlJ6eLj8/P4WeEhDidGx4ervT0dMc5f20QLxy/cyvPLYEzuDBg3X//fdr1
qxZxqqRO0y7L02GOPafDgwY6u+HISExM1duxYp32/+4fP/77M/nCt5n2
yXpK0decBNb2zqmI7xWjUtFxSFqeoXGSo/vloW701l/mF1fXLW9f0QAPJFdna2nh32tCxLGjGSnlf88kIGf
KFLw
Q3GnUvg5HVo+WJxcXH6/vvvvtXbtWjdUdWUeSxK3bt2qIUOGXPIfis1mO05Ahq5SamnrF61yqOy8F
ifrVrVFOtB0zTr0dO0vYf0npKkrRjj/P/G9q5N11lI0o67Tt+8rTS9h/Rig0/qvezc9S2UU01qF3B7bUDyF/Z2
dl6dvgQHTp0UK+9MZsUEtTe8QYMGaenSpfr/ZHRETo3LlzOnnypP5hw8fVkREhOOci592vvMVfu
54bEmMSIiQt98881lj3/zzTdGVHopvNxzc/S4UHTB1xv+69p47aPDpN+w4edzq27+BxHTxyUlXKO
y+LUzk6TPsPnbjsNS8kkH5FWUMeKEwuNIi/7NunmW/MUUhIySu/CXBRQVkCx7IsDRo0
vXj0VLVpUy5cvd+zbZubXOn9u/fr5iYGElStMtm3bpiNHjjzjo5Zp4MCBB2rRpk
5o3b+5oCA8fPqzly5fzrzTff10TJkz1VHjzo5YTueqDtHbp/yBvKPH1W4aUCJUkZmWd1NitbkjR17jI9I9/1h7bfv
pV23deUAPdWygquXD1XP4bElS/ZrRqndt4Tv59RhZvLaPQT7bV7/1Ft+G6vjf79jtcHf
z2gnT/uUFBwsEqXLLqMRQ5/Sjzt+0MszikNfHjv35YGNwcLCKFs3bU5pAYREXF6cFCxbo448
wsAICAhQcHKx+/fopPj5eoaGhCgoK0uDBgxUTE6N//OMfkqRWrVqpevXq6tmrlyZNmiTbtm16/vnnFRcXl6XE6Rma9Z58kPtwS+zH2ofZ1ixcDGO10+b+yMRuT9fkkMtGDdCPmapiPSfkCQHdDKs3DrmUvsmu2M5J2+W/7jIcapn2YH4FsHTtEd6/fv2zVI2xcu9ZkbGUH5xrGv0/QGJE
BJCjMLrNxVDk+ZLF6FONWcnfVkNy6u3znWGr1tR0NsqOLp5Ir0C1o4EDl4ZrHvvN+/4A5LTS2h3Bnq9rJtyaRFEZdOy7ZMGxJqmQyPvIhLSIzZKmjFPPj4jmUn/ZXezzN7z9S3uXWq9Xe+T
VKMT52ucBc/n7b+qTaGScW+N7fbjxAfdVUaV5Hjxwn3Nt7z1wQH6uRyBtgdWnk2fHEzF+dzMY1XHcTa+QqWB7EM1iiuOP3PfhGPQ9ypXoSxxfhu08+XYxK5rvWGfrH8Ovc7sm9/TtRT+q+uXs8f9HtnP2XBxZmUoxaDNOV6jI8sJQOX8kfsq5RfOXtgE8c56fPVuvL4D6etlw
```

hI7d+6cp4vZbDYeXgEAAIWOO9czTExM1NixY532jR49WmPGjLmq67VpOOZdu3ZVhQoVtHv3bj333HNq27atUlJ
S5Ovrq/TOdIWFhTm9p0iRIgoNDVV6enqe75OnJjE3N9e16gEAAAoRHzeONyckJCg+Pt5p37WkiD169HD8uVatW
qpdu7YqVaqklStXqnnz5ld93Ytd08/ynT17Nr/qAAAAuCHZ7XYFBQU5bdfSJF6sYsWKKl26tNLSOiRJERERORnL
kiNM558+f14kTJy47j/FSXG4Sc3JyNH78eN10000qUaKE9uzZIOkaOXKkZs+e7erlAAAAPK4w/yzfgQMHdPz4c
UVGRkqSYmJidPLkSW3atMlxzooVK5Sbm+vSD5+43CS+8MILSkpKOsSJE+Xn5+fYX7NmTb311luuXg4AAAB/kZm
ZqdTUVKWmpkqS9u7dq9TUVO3fv1+ZmZkaPny41q9fr59//lnLly9Xp06dVLyZbVu3VqSVL16dbVpOOYDBgzQN
998o//9738aNGiQevTokecnm6WraBLffvttvfHGG+rVq5d8fXOd++vUqaMff/zR1csBAAB4nM3mvs1VGzdu1G2
33abbbbrtNkhQfH6/bbrtNo0aNkq+vr7777jvde++eppzZo1TkPY8+fPV7Vq1dS8eXO1a9d0d
999t9544w2X6nB5ncRff/31kr+skpubq+zsbFcvBwAAgL9o2rSpLMu67PEvv/zyiitcIDQ3VggULrqkOl5PEGjV
qaM2aNcb+//73v46OFwAAoDApzHMS3cXlJHHUqFGKjY3Vr7/+qtzcXH30UufUfauXOn3n77bS1dutQdNQIAAOA6c
zlJ7NSpk5YsWaJly5apePHiGjVqlHbs2KElS5S5aoZcuW7qgRAADArXxs7tsKK5eTRElq1KiRkpOT87swsWAAAAjyj
Mw8LuclVNovTnkzc7duyQ9Oc8xXr16uVbUQAAAPAsl5vEAwc06MEHH9T//vvc/hYSESJJOnjypu+66SwsXLtTvaF
vnm/K4RAADArcgRTS7PSezfv7+ys7O1Y8cOnThxQidnNCOHTuUm5ur/v37u6NGAAAAAXGcuJ4mrVq3Sr
qVce+qlWravr06WrUqFG+FgcAAHA9+DAn0eByk1i2bNlLbhNLpqdk5Pj0k+9AAAAoOByuUmcNGmSBg8erIObNzr2b
dy4UU899ZQmT56cr8UBAAAcDwXpZ/kKijwllaGLpP0fDTp0+rQYMGKlLkz7efP39eRYoUOSOPPKLOnTu7pVA
AAABcP3lqEl9++WU3lwEAAA5rJNoylOTGBsb6+46AAAAAUIBc9WWLakZF8jWD3aAza7XW+99ZbGjh2rqKgovf3
G4ST58+rREjRuj999/X8ePHjeM5OTn5UhgAAMD1whCLI4Jpefbn7mmmWe0YsUKzZw5U3a7XW+99ZbGjh2rqKgovf3
22+6oEQAAANeZy0nikiVL9PbbkyoVL9Pbbb6tp06bq27evGjVqpMqVK6sO1lrz589Xr169hhWtbXLyqpdu3693FEnAACA2xAkmlxOkE+cOKGKGF
StK+nP+4YkTJyRJd999999t1avXp2/1QEAAMAjXXG4SK1asqL1790J790qXp2/1QEAAMAjXG4SK1asqL1790qSqlWrpvffff1/SnwnjhQXOAQAAChM
thZXLTWLfvn21ZfLdetWSdKzzz6rwKvB/+/hoyZIiGDx+e7wSe7wUCAADg+rNZ127rdsqjdl9bvbsUjVXtzaMXWPUWqWEjWHvzsupNZzqXLlyqpdu3Z+1
XVN0jPM35YGcGMIKV7U0yUAcBP/a1qY79oMXrTDbdee3qW6267tTtf8jyM60lrROdH5UQsAAAAKiDw1id0mTcv
zBZ988smrLgYAAMATCvPcQXfjU5M4derUPF3cZrtZNbpd4JjpcX
gIHAAAANz6SRAAA4PV4cMVEkggAAAADSSIAAPB6zEk0XVWSuGbNGj300NWKiYnRr7/+KkmaN2+q5dm6/FAQA
AwDNcbhI//PBDtW7dWgEBAdqyZYYuysrIkSRkZGXrxxRzvUAAAAB3s9nctxVWLjeZ0MOKxZs/Tmm2+qaNH/+
w3Vhg0bavPmzflaHAAAwPXgY705bSusXG4Sd+7qcqaNGxv7g4ODdfdfLkyfyoCQAAAB7mcpMYERGhtLQ0Y//atWt
VsWLFfCkKAADgevJxJ41ZYuVz7gAED9NRTT2nDhg2y2Ww6dHhg2y2Ww6ePCg5s+fr2HDhunxxx93R40AAAC4zlxe0aufZZ59Vb
m6umjdvrjNnzqhx48ay2+0aNmyYBnnQz8/DqQ3wMPBFQ0ur+vXO+ebE5rOuFMQs1Ri6n8Y9e/v1j2IRkyjlljfwaPK3kBm/X7Lz0Kps0B5pb7vRcRS0Jis
2Z/+/uGK1asuKaCAAAA4HkuN4l169Z1FR9//33RtmMNNFaTHHhW9JgkBFvRhnyiRvL3DY8Sc37a6bZHA8ZPFT9STGJGQgkd0/3gf3jBX+hDbRENPTZWAnh2a7kCG1CTbFe1DQzv/6jI4GJXPHDSzbYLT4Z0BYe+OD1J7nL4TWbpRJSENFqPJqf6n8lAxpUzW/ZK/1KTZanbZwNaHJLqXJgJLXTyc+Jm5l+ZD0jZd9bm6EGg4M0LqMmJ+AsVGIzZ8JW6Fg3sxpk5Ls9QqvEkK1xgwMRqJMA6vjhtNEDsm3Y4ra8OPM4vaW1gWYuIa+I8YUcSLCdtXuA3DWt3BdV8Sm5E5MQmGJ84cu4qGeMgsAIB0HW2pAZA4xLXKiWu8ezo9y1+hLZs+7pP+ff1evPkGs8G5Z7n3gW
G7bXLV69Wp17NhRUVFRstlsWrx4sdNxy7J03ILa7kiZRWYNHcsoJo1B6mm9i8ZDwzHT2+R0dLeGm4gHaPJkvd1JMTObt+F8kDkNz0AJL+4DkWwezKPAA4JDWgBAlKEWq2uYnWI1nzkDDYPC01OcQOF5u8h7Kup+ju9o82/vJyOZSwpHZMHDhuCcNKCjH7hVdVrqbeTZ0QKNB8+Lk9DKeY0WZl8f/kFsr4jVnB3c28o8HOwZT8EGxtU55qYbVfNcVwIL0O2WUaBWBJWSVH4w3n6oxOGl3MCH3XBBE2hezHPuv09+NS+t2wOfVxuD
...

8d8uvbrv2g7fd5LZruxNJIgAA8Ho8pGHiOwEAAICBJBEAAHi9gjQnsaAgSQQAAICBJBEAAHg9ckQTSSIAAAAMJ
IkAAMDrMSfRRJMIAAC8HkOrJr4TAAAAGEgSAQCA12042USSCAAAAANJIgAA8HrkiCaSRAAAABhIEgEAgNdjSqK
JJBEAAAAGkkQAAOD1fJiVaKBJBAAAXo/hZhPDzQAAADCQJAIAAK9nY7jZQJIIAAAAAOkiAADwesxJNJEkAgAAw
ECSCAAAvB5L4JhIEgEAAGAgSQQAAF6POYkmmkQAAOD1aBJNDDcDAADAQJIIAAC8HotpmOgSAQAAYCBJBAAAXs+
HINFAkggAAAADSSIAAPB6zEkOkSQCAADAQJIIAAC8HuskmmgSAQCA12042cRwMwAAAAwkiQAAwOuxBI6JJBEAA
AAGkkQAAOD1mJNoIkkEAACAgSQRBd47SW9q9dfLtH/fXtnt/qpZq64eHTxE5aIrOM45fuyYZk6frEObUnTmzBm
VjS6vh/sOVJN7WnqwcgBXo23Le3Tw4K/G/gd69NRzIOd7oCJ4A5bAMdEkosDbunmjutz/oKpVr6mcnPN6c+YrG
jZ4oOYu/FgBAcUkSS+OTVDm77/rxZdmKDgkRMu++Exjnhuq1+cuVJWq1T38CQC4Yv7C/yo3J8fxOi1tlx7t31c
tW7fxYFWA92G4GQXepGmvq22HzqpQqbIqV6mmhFEv6HD6If204wfHOdu/S1XX7j1V/dZairqprHr3e1QlSgTqp
x3bPVg5gKsRGhqqOmXKOLbVK79W2bLldEf9Oz1dGm5gNjduhRVNIgqdzMxMSVJgcLBj36216+rr5C9OKiNDubm
5Wv7VZzp37pzq1uNfKkBhln3unD5d+ok6d+OmG+OBcCMfm81tW2FVoJvEX375RY888sjfnpOVlaVTpO45bVlZW
depQlxvubm5mjHlX6pV5zZVrHSLY/+YF1/S+fPn1bFlQ7VoeLteShynCRNf1s1ly3mwWgDXasWKZfr99991b+c
uni4F8DoFukk8ceKE5s6d+7fnJCYmKjg42GmbPuXf16lCXG9TJO7Q3j1pGjVhktP+2bNmKDPzd02Z8bemPueu
vfsrTHPDdPutJ88VCmA/LDoww/V8O7GCgsL93QpuMEx3Gzy6IMrn3zyyd8e37NnzxWvkZCQoBMjHy/bl2rCT
JdtFvKvn4+Cj3b/67BaBg+3jRRwoNLaVGjZt6uhR4g8Ic+bmJRyO3yMhIffTRR8rNzb3ktnnnzZk+WhwJi6sQJS
v58qUaO/7cCihXX8WPHdPzYMWX9/+YwunwF3VS2nF5KHKcd27fp1wPT+7xcPUArkZubq4+XvS
ROnbqrCJFWKON8ASP/s2rV6+eNm63dOl3y+JVSRniHjz9cKEl66rG+TvufHTVBbTtOVpEiRTx6KX6Wz2SzPNiFrVmzRqdPn6deurPn4T/4fhZgA
A4PUKOmBzkSJFFBERYezPyWDBAt1zzz2SpHSSJAAAAMJIkAAABEiQaSRAAAABhIEgEAgNdjCRwTSSIAAAAMJIkAAMDrXeFXgLO
SSSIAAAAMJIkAAMDrESSaaBIBAADoEgOMNwMAAMBAkggAALweS+CYSBIBAABgIEkEAABejyVwTCSJAAAAMJAkA
gAArOeQaCJJBAAAgIEkEQAAgCjRQJMIAAC8HkvgmBhuBgAAgIEkEQAAeD2WwDGRJAIAAMBAkggAALweQaKJJBE
AAAAGkkQAAACiRANJIgAAAAwkiQAAwOuxTqKJJBEAAKCAGDNmjGw2m9NWrVo1Vo1x/GzZ88qLi5OpUkSJdStW
zcdPnzYLbXQJAIAAK9ns7lvc9Wtt96qQ4cOOObba1a9c6jgOZMkRLlizRBx98oFWrVungwYP2rVrVrPn4T/4fhZgA
A4PUKOmBzkSJFFBERYezPyMjQ7NmztWDBAt1zzz2SpHSSJAAAAbpSVlaVTpO45b
VlZWZc9f9euXYqKilLFihXVq1cv7d+/X5KOadmZWdnqOWLFo5zq1WrpnLlyikl
JSXf66ZJBAAAsLlvSOxMVHB
wsNOWmJh4h4yTIaNGigpKQkpKQkffHFF9q7d6/atGmjU6dO6fDhwwoICFDp0qWdfl+4cGEdPnzY7a9NTKpXuJ7zlH4
8fbs5mqeHHNGigpKQkpKQknvrqK3qPuXw7vz/GzHGahnx4AKCGdKc0jp0/VrRRO3
5mWDwNzd1TdbjKR/xQgUPHAA6B6vT6gOJx4s+NuW2yw7x86kzZ+kFEo/xoOW1EnAAgmOIJNfWWvx6sSZUrd+Sx20h+7/ZWj25TiHTZ2pOHVx/4M4GAPSL0GzaalnAoQCNPRZwg0NwgBeOoo1aOZFpnXd7rM+RgQgcHMgsgAA2TYfgKSLgiTkppXE
2REmaZDZ+c+p3aPdAG9oY8PEkYhR/L5RCjwUEW+HLnrR1ppXmkAHHiQBSSOQQrk2A/hdo8N4SlfNrBgNYmsmcgP+8+aCJGGfESR1lJJlXhNcymlbM7fw3rGUAbNb47LfR4JNWoGVDJnbMg5Lm7U2WTZmKmgvJa
bB5HqLx57TpXMkt5ooOS/xdNzl1RZBVjd+bnL8WvFQWYTI0Vsr2CJg8NMa3QEfPq7++vSVVVwPv5Ve0Vb9ndmXRhqJLiVoUqE97t7akQaBMJwxS5UVrxWm8SQsXlHEj9IESKvx4CYpn6OCO0Fz4o0fbSgXlNzI/DOz7EyoadQCLgvxybQiW+tU4QQ9VyoZGLMa+Rz0dvRcWdjLLNtDbs+7wYzWXRYMTTsJM5OfjnrGs3Fr7Y+Eg3WlHThQJcF6M3jr0kAkCn6FGqBxYpoGgGQKQKZaWzUSSp1gYhAPI2ZnWK

```
sXHxys2NlZ33HGH7rzzTr388ss6ffq0+vbt6+nSAFyDzMxMpaWlOV7v3btXqampCgONVbly5TxYGeBdWAIHhdq
MGTMOadIkpaenq27dupo2bZoaNGjg6bIAXIOVK1eqWbNmxv7Y2FglJSVd/4IALOWTCAAAAANzEgEAAGCgSQQAA
ICBJhEAAAAGmkQAAAAYaBIBAABgoEkEAACAgSYRAAAABppEAAAAGGgSAVyzPn36qHPnzo7XTZs21dNPP33d61i
5cqVsNptOnjx52XNsNpsWL16c52uzVWHWvav6c52uOGTNGdevWvaa6fv75Z9lsNqWmpl7TdQDgeqJJBG5Qffr0Okc1mk81mk5+fn
ypXrqxx48bp/Pnzbr/3Rx99dpzbr/3Rx99pPHjx+fp3Lw0OdgCA66+IpwsA4D5t2rTRnDlzlJWVpc8++OxxcXEqWrSohEhSjHP
PnTsnPz+/fL1vaGhov2vlwHAOA5JInADcxutysiIkLR0dF6/PHH1aJFC33yySeS/m+I+IUXX1apUJSqVq0qSfrll
l/UvXt3hYSEKDQOVJO6ddLPP//suGZO7i4+MVEhKiUqVK6ZlnntHFPwF/8XBzVlaWROwYobJly8put6ty5cq
aPXu2fv75Zz2vr1kySVLJkSdlsNvXp0qFWoUEEBAQGqU6eO/vvf/zrd57PPPtxHHpAJOJSpHjsq0bQKvs7IsS/m+I+IUXXlBUVJSqVq0qSfrll
1/UvXt3hYSEKDQOVJO6ddLPP//suGZOTo7i4+MVEhKiUqVK6Z1nntHFPwF/8XBzVlaWRowYobJly8put6ty5cq
aPXu2fv75ZzVr1kySVLJkSdlsNvXp0OeSlJubq8TERFWoUEEBAQGqU6eO/vvf/zrd57PPPlOVKlUUEBCgZs2aO
dWZVyNGjFCVKlVUrFgxVaxYUSNHjlR2drrZx3uuvv66yZcuqWLFi6t69uzIyMpyOv/XWW6br8/f1VrVo1vfb
aa5e952+//aZevXqpTJkyCggIOC233KI5c+a4XDsAuBNJIuBFAgICdPz4ccdfr5cuXXr16rdvXt6O/jnX7z3dPxaWEsM+lD0L+TE3k4ckk+7F+9AeJx3Wt76rv/7+/ff++f+a5a76TH+TJPyX95VwhO9jzWBsaKy0i6YfZ2ng6e
Nm6Y6depo79690NbsmMqWLasPP/xQ3bp1086dOxUUFKSAgABJUmJiot555x3Nmj8/4qZzvG5H/RVX6sxXA
ZMmTfTLL7+oa9euiouLO8CBA7Vx40YNHTrU5e8kMDQSUlJioqKOrZt2zRgwAAFBgbqmWeecZyT1pam999/XOu
WLNGpU6fUr18/PfHEE5o/f74kaf78+Ro1apmRmzJih2267TVu2bNGAAQNUvHhhxxcbGGvcc0XKkfvjhB33++ecqX
bqOOtLS9Mcff7hcOwC4lQXgh8bG2t16t16tT36TKky273W4NGzbMcTw8PNZKyspyvGfvHlW1apVrdz
cXMe+rKwsKyAgwPryrrQpsmIVKsr
3/9tSXJ+u233xz7zp49axUrVsxat26d07n9+vWzevToYa17qYJGvRokWXPT5p0iS
rXr16jtejR4+2fH19rQMHDjj2ff7555aPj49xqnB3ESS8pijWKuWL2rt3ryXJSkS5OvmSd
2VZltWxY0erb9++160BAAoCkkTgBrZO6VKVKFFC2dnZys3N1QiFPrbW/m5ljT9MLB
zZ89q9+7dysjIUVLPeOPVvmSbBjH8Vq1ajnQNQ9y6davS2T1
ZJ27NjhVIckxcTE5PkeFyxcuFFp7k3T7t27lZmZqfPnzYso+yKMjpnHLlyummm25yuk9ubq9ubq527typwMBA7d69W/3/3/3
69d0AAQMc55w/f1/17BwcGXvf1fjjz+ubt26afPmzpS0c+fn0fG1Wp8KSoqS
kWKOP+VL168uNPrzMzXjc3+aqbFjx6p169Y
KDg7We++9p5deesnlWt98802/bps88+U3Jysp6O3b664uDhnNnjz56j8MAOQzmkTgBla8e
HFVrlw5z+fWrVtXSycOFrTfhM/ccW20e9q2bat9+/Y+9bwHFvRKV1pK9bq6wXO5uXn
dBhV7Sq9tyv/9Ye7TF1LBx7cGFHl1qRXhYt07H9+c/39ubWpwZQqrLCVLTKt4sGbewurSRiRQvQ6V23
Ozs6lzUVdTlYj+6m2b3r7...
```

```
      "cell_type": "markdown",
      "metadata": {
        "id": "oQzx7Y5fFI-3"
      },
      "source": [
        "## RandomForestClassifier"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "dc-8XtjcmWPS",
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 980
        },
        "outputId": "31d9aeb0-3d21-447d-c5a9-d93e743d7ae3"
      },
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
            "Training Accuracy: 0.9935379644588045\n",
            "\n",
            "Classification_Report -->\n",
            "\n",
            "              precision    recall  f1-score   support\n",
            "\n",
            "         0.0       0.94      0.98      0.96       275\n",
            "         1.0       0.78      0.51      0.62        35\n",
            "\n",
            "    accuracy                           0.93       310\n",
            "   macro avg       0.86      0.75      0.79       310\n",
            "weighted avg       0.92      0.93      0.92       310\n",
            "\n",
            "Cross Validation Classification_Report -->\n",
            "\n",
            "              precision    recall  f1-score   support\n",
            "\n",
            "         0.0       0.92      0.98      0.95      1098\n",
            "         1.0       0.98      0.91      0.94      1123\n",
            "\n",
            "    accuracy                           0.95      2221\n",
            "   macro avg       0.95      0.95      0.95      2221\n",
            "weighted avg       0.95      0.95      0.95      2221\n",
            "\n"
          ]
        },
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 800x600 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAAAokAAAIjCAYAAABvUIGpAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAAA/T0l
EQVR4nO3deViU9f7/8deAMqCyiAtLKe6KuZWZkbkdCfckLbOsOEExPBWaiZtSxXCpOtlhaaufUkTI9Zae0tNUON
ZPcMVMzUdNcwNKAQEWF+/dHP+fb+NFklHHAeT667uty7vue+37PdFnv6/X53J+xWZZlCQAAAPgTH08XAAAAgLK
HJhEAAAAGmkQAAAAYaBIBAABgoEkEAACAgSYRAAAABppEAAAAGGgSAQAAYKBJBAAAgIEmEcBf2rFjh+Li4hQcH
CybzaYFCxaU6vV/+ukn2Ww2paW1lep1y7NOnTqpqU6d0ni4DgJejSQTKgZ07d+rvf/+76tWrJ39/fwUFBaldu3Z
6+eWXdezYMbfeOyBQZs3b9bTTz+t2bNn69rr3Xr/S61QYMGyWazKSgo6Kzf444fP3fPODB
w5o/PjxysjIKIVqAeDSquDpAgD8tY8//3DDPn2//li33Xab7HHa77rnnHjVr1kwnTpzQypYzX2fSz2fT888//X8+PELuvaBAwc0Y
cIE1alTR61atSrx+7744twYMGKCoqCgtXbpUERERjmOKzMzUxx9/7Lb7//LL5K
kkJAQt93DZrPJ39/fbdc/H7vdrnbt2um///2v0STTtXPXv21Pvvv39Jajl69KgqVaokPz+/S3I/APgrDDcDZ
djkyZ0Vn5+vN954w61BPK1w5w6bvzsgGWbz+u1Vu9Fz2Ri9X2yYVBHv9Xr10sqVK3X
dddfJ399f9erV0tvveU4Z/z9gzc
UoVNW7cWI899pjj+LnmJC5dulTt27dX5cqVFRISor6kkJAQBQcHa/DgwTp69d
z3Hnnnfr000+Vk5PjJ2Ld27Vrt2LFDd955p2Ld955p3H+kSNHNHnj0qjOWbZsmdq0aeeHJNXj0HCQkgyZ
jx4sGPY+vTn7NSpk5o1a6b169erQ4c0qlSpkuN7HTNmjLm+vLt2...
```

7z+c889p+7duysmJkZDhgzRsWPHNG3aNAUHB2v8+PGl9jnO5OPjo3/84x/nPa9Xr16aOHGiBg8erBtuuEGbN2/
WnDlzVK9ePafz6tevr5CQEM2cOVOBgYGqXLmy2rZtq7p167pU19KlSzV9+nQ9+eSTjiV5Zs2apU6dOmncuHGaP
HmyS9cDgBLx8NPVAErgxx9/tIYOHWrVqVPH8vPzswIDA6127dpZO6ZNs44fP+447+TJk9aECROsunXrWhUrVrR
q1aplpaSkOJ1jWX8sgdOzZO/jPmcuvXKuJXAsy7K++OILq1mzZZpafn5/VuHFj6+233zaWwFmyZInVp08fKzIy0
vLz87MiIyOtO+64w/rxxx+Ne5y5TMyXX35ptWvXzgoICLCCgoKs3r17W1u3bnU65/T9zlxiZ9asWZYka/fu3ef
8Ti3LeQmccznXEjijRo2yIiIirICAAKtdu3ZWenr6WZeu+fDDD60mTZtaFSpUcPqcHTt2tK666qz3vPP18nLy
7OioqKsa665xjp58qTTeSNHjrR8fHys9PT0v/wMAHAhbJblwsxuAAAAeAXmJAIAAMBAkwgAAAADTSIAAAAMNIk
AAAAwOCQCAADAQJMIAAAAAOOiAAAADJflL64EXJ3k6RIAuMlva1/xdAkA3MTfg12JO3uHYxvL53+3SBIBAABgu
CyTRAAAAJfYyM3ORJMIAABgs3m6gjKHthkAAAAGkkQAAACGmw18IwAAADCQJAIAADAnOUCSCAAAAANJIgAAAHM
SDXwjAAAAMJAkAgAAMCfRQJMIAADAcLOBbwQAAAAGkkQAAACGmw0kiQAAADCQJAIAADAn0cA3AgAAAANJIgAAA
HMSDSSJAAAAMJAkAgAAMCfRQJMIAADAcLOBthkAAAAGkkQAAACGmw18IwAAADCQJAIAAJAkGvhGAAAAYCBJBAA
A8OHp5jORJAIAAMBAkggAAMCcRANNIgAAAItpG2ibAQAAyojU1FS1adNGgYGBqlmzpuLj47V9+3anczp16i6iSbz
ea03X///U7n7N27Vz179lSlSpVUs2ZNjRkzRqdOnXKpFpJEAACAMjLcvHz5ciUmJqpNmzY6deqUHnvnvsMcXFxWn
r1q2qXLmy47yhQ4dq4sSJjteVKlVy/LLmoqEg9e/ZUeHi4VqlapYMHD+qee+5RxYoV9cwzz5S4FpJEAACAMuKzz
z5zep2WlqaNWtq/fr16Chg2N/pUqVFB4eftZrfPHFF9q6q6dau+/PJLy8/PrOS
1lI22GQAAwJNsNrdthYWFysvLc9oKCwtLVFtubq4kKSQ+GbYAOQAAwdRbQoMAAAyZQv4KTQ01Gn/nDlzVL16dTVr1kwpKSk6evSo41h6erqaN2+us
LAwx76uXbsqLy9PW7ZsKfFXQpMIAADgRqmpqQoODnbaUlNT//Z9RUVFGjFihNq1a6dmzZo59t9555369ttva5ly5UqdVXkWzRXZerBm+dGAx/3bj8p3
ffaWUlBTNnj1bd9111+N4VlaWU4pZjUU4MoyfE6Kyurx7ZSlKt5bWWqQrQ8Wy+KDi4abnd7bsmDx3mt7yl+Kat5ip3lKBPSz+c1yoIXTcJJGKOULl+mDDXTdbaWmbKgCN7ZbpnfgtPDbdqHuk6uUFM03CSHxXNdTfJvDY0fvrNHPxH5YYS3OaRfAcdzI0d6+dT1KZsQwjIeHeMDIwAD/2iFd0DL5o

1U3Z2ttM5e/fuVc+ePVWpUiXVrFlTY8aM0alTrv3/z2NzEhMTE1W9enVNmTJF06dPV1HRHw+b+Pr6qnXr1kpLS
1P//v09VR4860/900iSFr/+sNP+oU/M1tsLVzteJ/SJ0f7sHH2Z/oNxjeJiS/1GzNDLjw3QsrRRKjheqDkL12j
ijI/dWjuAi5ednaVHxyQrJydHVUNDdfU1rTV77jyFhoZ6ujRcxspK4Ld8+XIlJiaqTZs2OnXqlB577DHFxcVp6
9atqly5siRp5MiR+vjjj/Xee+8pODhYSUlJ6tu3r7755htJUlFRkXr27Knw8HCtWrVKBw8e1D333K0KFSvqmWe
eKXEtNsuyLLd8ShecPHlSv/76qySpevXqqlix4nne8dcCrk4qjbIAlEG/rX3l/CcBKJf8PbgwX4PRn7rt2pnPd
7/g9/7yyy+qWb0mli9frg4d0ig3N1c1atTQ3Llzdeutt0qSfvjhBOVHRys9PV3XX3+9Pv30U/Xq1UsHDhxwTN2
b0X0mxo4dq19++UV+fn4luneZ+O3mihUrKiIiQhERERfdIAIAALjKnXMSz7ZcX2FhYYnqys3NlSRVWXFkr5+/XqdP
HlSsbGxjn0aNGmi2rVr01aESU9PV/PmzZ2euWEn8nZaJJBAAA8CSbzX1bamqqgo0DnbbU1NT
z1lRcXKyHH35Y7dq1U7NmzSRJWVlZ8vPzU0hIiN05YWFhjh8mycrKMh7+Pf369Dklwc/yAQAAuFFKSoqSk50d9
tnt9v0+LzExUd9//71WrlzprtL+Ek0iAADweu5cqsZut5eoKfyzpKQkLVVq0SCtWrNCVV17p2B8eHq4TJ04oJyf
HKU3Mzs5WeHi445wzlxk8/fTz6XNKguFmAACAMsKyLCUlJWn+/PlaunSp6tat63S8devWqlixYsWeLYt337d
u3du1cxMTGSpJiYGG3evFmHDh1ynLN48WIFBQWpadOmJa6FJBEAAHi9srIETmJioubONasPP/xQgYGBjjmEwcH
BCggIUHBwsIYMGaLk5GShoYqKChIw4cPV0xMjK6//npJUlxcnJo2baq6//npJUlxcnJo2baq7775bkydPV1ZW1v7xj38oMTHRpUSTJ
hEAAKCMmDFjhiSpU6d0Tvtnz+ZqlQYMGSZKmTJkiHx8f9evXT4WFherataumT5/uONfX11eLFi3SAw88oJiYGFW
uXFkJCQmaOHGiS7WUiXUSSxvrJAKXL9ZJBC5fnlwr5fnlwnseljX7jt2lufiXPbtd2JOYkAAAAwMNwMAAC8XlmZk1iW0
CQCAACv584lcMorhpsBAABgIEkEAABejyDRRJIIAAAAA0kiAADwesxJNJEkAgAwECSCAAAvB5JookkEQAAAAa
SRAAA4PUIEk00iQAAw0sx3GxiuBkAAAAGkkQAAOD1CBJNJIkAAAAwkCQCAACvx5xEE0kiAAAADCSJAADA6xEkm
kgSAQAAYCBJBAAAXo85iSaSRAAAABhIEgEAgNcjSDTRJAIAAK/HcLOJ4WYAAAAYSBIBAIDXIOg0kSQCAADAQJI
IAAC8HnMSTSSJAAAAMJAkAgAAr0eQaCJJBAAAgIEkEQAAeD3mJJpoEgEAgNejRzQx3AwAAAADSSJAADwegSJJpJEAAAAGEgSAQCA12NOookmEQAAeD16RBPDzQAAADCQJAI
AAK/HcLOJJBEAAAAGkkQAAOD1CBJNJIkAAAAwkCQCAACv50OUaCBJBAAAgIEkEQAAeD2CRBNNIgAA8HosgWNiu
BkAAAAGkkQAAOD1fAgSDSSJAAAAMJAkAgAAr8ecRBNJIgAAAAwkiQAAwOsRJJpIEgEAAGAgSQQAAF7PJqLEM9E
kAgAAr8cSOCaGmwEAAGAgSQQAAF6PJXBMJIkAAAAwkCQCAACvR5BoIkkEAAAAgSQRAAB4PR+iRANJIgAAAAyl0
iTm5OSUxmUUAAAA8wmZz31ZeudwkPvvss3r33Xcdr/v3v3769q1arpiiuuu0KZNm0Oq10AAAgEvBZr05bSuvXG4SZ86
cqVq1akmkSFi9erVq1akmSFi9erMWLF+vTTz9V9+7dNWbMmFIvEVEAAAAJeeyw+uZGVl0ZrERYsWqX///oqLi0OdOnXUhSkeOHDnnP
ADA3cpx4Oc2LieJVatW1c8//yxJSmTdcvdU2+QkJVatW1c8//yxJ+uyzzPFVpsdAAAAPMLlJLFv376688471b9+fR09elTdu3eXJG3cuFH169cv9QIBAADcjTmJJpebxClTpqhOnTr6+eefNXnyZFWpUkWSdPDgQT344IOlXiAAAAAuPZebxIoVK2r06NHG/pEjR5ZKQQAAAJcaQaKpRE3iRx99VOIL3nzzzRdcDAAAgCdcyB3flvtSt9EZFNzPKfK/E3+g7+Eg+ae6gTjmNVe0vkaAe1/vutO8T9GWvGOgaS5fxq2vOvHA2oM+7/6P5gEY3RWs6Rb/ZmbYkVW/8gk8rgc+f9Pt2ZdSfH1wMJKj8oRzi1zUUyTrCx8Yp3MnfgqDiBwAb2ByGWjf/CqMcgzWRirIG4XmsE5B08DVjKpJbAp2NMk3b75uWqNqoS5+Iid2nS2yhyQGnotE3hAhGPJnBRVZLKrgPLmSIbiKM=:k3bS5jsBWR4WjF8AmVh2fVXxMZPmdTIDR4wc9uW34l7jYhKKr1eD9FL6mEoWuEBShfwFqLF2ChOuNDQ2q6bmjVyUkRUi3/MGAcxCK=0OGH/ftXaTGxzbbnYDYjtxz9aN/1O5WRDJzfMn0xd8sqvJgjThLNM4r5SCdtyMWVe3d0Ccjpu8JZHGs5gUjjrCQjk81hRi4hZLBmUzRNPBVizG8h6x+kJlQ

JTmzJmjgQMHuqNOAAAAtyFINLmcJB45ckT16tWT9Mf8wyNHjkiSbrzxRq1YsaJ0qwMAAIBHuNwk1qtXT7t375Y
kNWnSRPPmzZPOR8IYEhJSqsUBAABcCjabzW1beeVykzh48GBt2rRJkvToo4/q1Vdflb+/v0aOHKkxY8aUeoEAA
AC49GyWZVkXc4E9e/Zo/fr1atCggVq0aFFadV2U347yhDUAAOVN1Uq+Hrv38Pnb3HbtabdEu+3a7nRR6yRKUlR
UlKKiokqjFgAAAJQRJWoSp06dWuILPvTQQxdcDAAAgCeU57mD7lKiJnHKlCklupjNZqNJBAAA5Y4PPaKhRE3i6
aeZAQAA4BOuek4iAABAeUeSaHJ5CRwAAABc/kgSAQCA1+PBFRNJIgAAAAwkiQAAwOsxJ9FOQUni119/rbvuuks
xMTHav3+/JGn27NlauXJ1qRYHAAAAz3C5SXz//ffVtWtXBQQEaOPGjSosLJQk5ebm6plnnin1AgEAANzNZnPfV
l653CQ+9dRTmjlzpv7973+rYX2Tr2XThg0ObSrU4AACAS8HHZnPbVl653CRu375dHTpOMPYHBwcrJyenNGo
CAACAh7ncJIaHhyszM9PYv3L1StWrV69UigcN1u1+jRozV8+HB31AgAAOBW5XjqoNvYLMuyLuSNJ06
cUGZmpvLz89WOaVVqVKltGu7YL8dLfJ0CQAAwEVVK/l67N6Pf/qj2679dPdGbru2O13wYtp+fn5q2rRpadYCA
ADgEeX5KWR3cblJ7Ny581Y/+vuHSpUsvqiAAAAB4nstNYqtWrZxen4xx5UhkZGfr++++VkJBQWnUBAABcMgSJJpe
bxClTppx1//jx45Wfn3/RBQEAAFxq/HazqdSW77rrWyrrrv0n//8p7QuBwAAAA+64AdXzpSeni5f/f//SuhwAAMAlw
4MrJpebxL59+zq9tixLBw8e1Lp16zRu3LhSKwwAAACe43KTGDwc7dHjRs31sSJExUXF1dqhQEAAFwqBIk
ml5rEoqIiDR48WM2bN1fVqlXdVRMAAADp+djct5VRMAAAFwqBIk
mMCHPLMsNTTaeJFfOqppzR69GgtWrRIBw8eVF5entMGAAAAVtF9E7uJ9VeI5iRMnTtSoUaPUo0cPSdLNN9/s9qPXDYjsjvBD3T12AqTWVQJajvxeJn8OgNs4x5DX4AGLbU+AAwB7WAYeZ+e3xG1Ca5SpN7rRR3EAvQ6daYgRvN8r2WFp6j5mLQiPaAGTz0qRPvtdN0AAC3yfNN7uAuFwdl2kbo9bpYSeH1pQ3zeqnwsnnx6Lp2Jq4aGZuWtXDzi+JtLOa0Ka4ncCKMdLr8o2rdv3AhmWG99HYAAe6gqVrmqY0AAEWKqHfG1tJbFGXQrrRTPZkJQWwJNU1NYOaT0CqP8YFDwEAAAAAAAAhvgAAAAbr4sxNS/AAAAAAMAAEQ0y/dVrbnSsgl1TbwwgvXQ6PmXoc7kLbP9yD4S6s4VV9cz6E3AK+NvUq3yAAAAAAAAAAMfVA3uI8NJgbzv8fCq9TWXkN0xzq5A8kOhMvgvWhbAdEkEdoP+WfkMhm9TXyeSbqAqm3ZGqNF+4pzRWIxI5/O9ZmHHk3c2Ngu5RaAAAAAOtGDcUPWAAxXWARXu5P1G52ATnxd4x+Q4CrKyAFAFN0VkCUhX

JTmzJmjgQMHuqNOAAAAtyFINLmcJB45ckT16tWT9Mf8wyNHjkiSbrzxRq1YsaJ0qwMAAIBHuNwk1qtXT7t375Y
kNWnSRPPmzZPOR8IYEhJSqsUBAABcCjabzW1beeVykzh48GBt2rRJkvToo4/q1Vdflb+/v0aOHKkxY8aUeoEAA
AC49GyWZVkXc4E9e/Zo/fr1atCggVq0aFFadV2U347yhDUAAOVN1Uq+Hrv38Pnb3HbtabdEu+3a7nRR6yRKUlR
UlKKiokqjFgAAAJQRJWoSp06dWuILPvTQQxdcDAAAgCeU57mD7lKiJnHKlCklupjNZqNJBAAA5Y4PPaKhRE3i6
aeZAQAA4BOuek4iAABAeUeSaHJ5CRwAAABc/kgSAQCA1+PBFRNJIgAAAAwkiQAAwOsxJ9FOQUni119/rbvuuks
xMTHav3+/JGn27NlauXJlqRYHAAAAz3C5SXz//ffVtWtXBQQEaOPGjSosLJQk5ebm6plnnin1AgEAANzNZnPfV
l653CQ+9dRTmjlzpv7973+rYsWKjv3t2rXThg0ObSrU4AACAS8HHZnPbVl653CRu375dHTpOMPYHBwcrJyenNGo
CAACAh7ncJIaHhyszM9PYv3L1StWrV69UigcN1u1+jRozV8+HB31AgAAOBW5XjqoNvYLMuyLuSNJ06
cUGZmpvLz89WOaVNVqVKltGu7YL8dLfJ0CQAAwEVVK/l67N6Pf/qj2679dPdGbru2O13wYtp+fn5q2rRpadYCA
ADgEeX5KWR3cblJ7Ny581/+vuHSpUsvqiAAAAB4nstNYqtWrZxen4xx5UhkZGfr++++VkJBQWnUBAABcMgSJJpe
bxClTppx1//jx45Wfn3/RBQEAAFxq/HazqdSW77nrrrv0n//8p7QuBwAAAA+64AdXzpSeni5f/f//SuhwAAAMAlw
4MrJpebxL59+zq9tixLBw8e1Lp16zRu3LhSKwwAAACe43KTGBwc7PTax8dHjRs31sSJExUXF1dqhQEAAFwqBIk
ml5rEoqIiDR48WM2bN1fVqlXdVRMAAAA8zKUHV3x9fRUXF6ecnBD3lQMAAHDp+djct5VXLj/d3KxzM+3atcsdt
QAAAKCMcLlJf0qppzR69GgtWrRIBw8eVF5entMGAAAAVtF9E7uJ9VeI5iRMnTtSoUaPUo0cPSdLNN9/s9PN8lmX
JZrOpqKio9KsEAABwo/I8LOwuJW4S9KsEAABgEAABAGVDiJtGyLElSx44d3VYMAAAACAJ5Akmlyak
2hjESEAAACv4NI6iY0aNpvo3jkyJGLKggAAA0BSIwgzudQkTpgwwfFwtFQQAAAFx+XGoSBSwBhMSiWEBAAC8h8tPNwMAAFxuyMJW4Si4uL3VkHAACAx/jQJRpc/lk+AAAAAXP5cenAFAADgcsSDKyaSR
AAAABhIEgEAgNdjSqKJJBEAAAAGkkQAAOD1fESUeCaSRAAAABhIEgEAgNdjTqKJJhEAAHg9lsAxMdwMAAAAAA0k
iAADwevwsn4kkEQAAAAaSRAAA4PUIEk0kiQAAADCQJAIAAK/HnEQTSSIAAAAMJIkAAMDrESSaaBIBAIDXY2jVx
HcCAAAAA0kiAADwejbGmw0kiQAAADDQJAIAAK9nc+PmqhUrVqh3796KjIyUzWbTggULnI5blqXExERFREQoICgx
IUGxsrHbs2OF0zpEjRzRw4EAFBQUpJCREQ4YMUX5+vkt10CQCAAAXWrZsmUaNGqV27dupy9vbXq1bt1ZBQYG++eYbV6N7CxB1q3Rm6j8vPx2XeQg7u/x/31+rs4wQ5VRaRnWMFxvjM2Pye1Bh+8AaiQXwdc1ZvekNWX/1ZqMwsaEHDOZp6YXSQWdFYmsr2exd1Jf6v2Qz7vy9bvUCQcFCU8jDdGl0KBl2qNjQDqSHQ6qnkmvgyuFQ8WApqKB0fPlfLgkGOqdpMzxQAAhXDvb0ZV1JIs3G0YyB7+1Msl8gbvh1AipK9Gvqc1Vs9JYhVKVbmGYo7MVbUVm4UuJRtjHx9qdM+vcNfwcvYCZkdyJqfdmAtDBhtDI5AjQlmTg2a8S4Eb7A3FAP/p9aIBKAW/I2I0YVEAjF5aTdnp49M2G9WYJFGIF2jYI4VNDbfwbvYTYXQnebOKUnzz9sgcfZJbAoGc9NE8qBw+2P6iBoLbbfY/ag7/EPO3ljkLxdWP2QpAcCFOIa/p6V1jFvUEA/SrzXMgMqlJRUZ4E7XxE4dfxIUYwrMrLmDIKwAAAAAAA

d2Ndff62CggJ169btrMcLCgq0bt06dezY0aXr/sZwMwAA5U7VSr4eu/fqnbluu3bb+sFuu7Y7ebRJdBeaRAAAy
h9PNo1rdrmvSbyuXvlsEnkMGAAAeD0Gm01lep1EAAAAeAZJIgAAAFGigSQRAAAABpJEAADg9VgCx0SSCAAAAAN
JIgAA8Hrn+RVgr0SSCAAAAANJIgAA8HoEiSaaRAAAALpEA8PNAAAAMJAkAgAAr8cSOCaSRAAAABhIEgEAgNdjC
RwTSSIAAAAMJIkAAMDrESSaSBIBAABgIEkEAAAgSjTQJAIAAK/HEjgmhpsBAABgIEkEAABejyVwTCSJAAAAMJA
kAgAAr0eQaCJJBAAAgIEkEQAAgCjRQJIIAAAAA00iAADwejY3/uOK8ePHy2az0W1NmjRxHD9+/LgSExNVrVo1V
alSRf369VN2dnZpfx2SaBIBAADKlKuuukoHDx50bCtXrnQcGzlypBYuXKj33ntPtPy5cv14EDB9S3b1+31MGcRAA
A4PXK0jqJFSpUUHh4uLE/NzdXb7zxxhubOnau//e1vkqRZs2Yp0jpa3377ra6//vpSrYMKEQAAeD2bG7fCwkLl5
eU5bYWFheesOBB7d27V5K0fv16nTx5UrGxsYRpotq1ays9Pb5vCbsBbGh/Zs2c08Zbd7joz/jyYRAADAjVJ
TUxUcHOy0paamnvXctm3bKi0tTZ999plmzZ8Jih3bt3q3379vr999+VlZUlPz8/hYSE0LOnLCxMWVlZpV43w80AA
ABuHG50SUlRcnKy0z673X7Wc7t37+74c4sWLdS2bVtFRUVp3rx5CggIcF+RZ0GSCAAA4EZ2u11BQUF027maxDD
FhISoUaNGyszMVHh4uE6cOKGcnByns7Kzs886h/Fi0SQCAACvV1aWwDlTfn6+du7cqYiICLVu3voVoVK1bUkiVLH
Me3b9+uvXv3KiYm5mK/AgPDzQAAAGXE6NGj1bt3b0VFRenAgQN68skn5evrqvzvuuEPBwcEaMmSIkpOTFRoaqqC
gIA0fPlwxMTG1/mSzRJMIAABQZpbA2bdvn+644w4dPnxYNWrU0I033qhv/1WNWrUkCRNmTJFPj4+6tevnwoLC
9W1a1dNnz7dLbXLMuy3HJ1D/rtaJGnSwAAAC6qWsnXY/fennXUbdduHF7Jbdd2J5JEAADg9cpIkFim0CQCAAD
QJRp4uhkAAAAGkkQAAOD1LnapmssRSSIAAAAMJIkAAMDrlZUlcMoSkkQAAAAYSBIBAIDXI0g0kSQCAADAQJIIA
ABAlGigSQQAAF6PJXBMDDcDAADAQJIIAAC8HkvgmEgSAQAAYCBJBAAAXo8gOUSSCAAAAANJIgAAAFGigSQRAAA
ABpJEAADg9VgnOUSTCAAAvB5L4JgYbgYAAICBJBEAAHg9gkQTSSIAAAAMJIkAAMDrMSfRRJIIAAAAA0kiAAAAs
xINJIkAAAAwkCQCAACvx5xEE00iAADwevSIJoabAQAAYCBJBAAAXo/hZhNJIgAAAAwkiQAAwOvZmJVoIEkEAAC
AgSQRAACAINFAkggAAAADSSIAAPB6BIkmmkQAAOD1WALHxHAzAAADCSJAADA67EEjokkEQAAAAaSRAAAAIJEA
0kiAAAADCSJAADA6xEkmkgSAQAAYCBJBAAAXo91Ek00iQAAw0uxBI6J4WYAAAAYSBIBAIDXY7jZRJIIAAAAA00
iAAAADDSJAAAAMDAnEQAAeD3mJJpIEgEAAGAgSQQAAF6PdRJNNIkAAMDrMdxsYrgZAAAABpJEAADg9QgSTSSJA
AAAMJAkAgAAECUaSBIBAABgIEkEAABejyVwTCSJAAAAMJAkAgAAr8c6iSaSRAAAABhIEgEAgNcjSDTRJAIAANA
lGhhuBgAAgIEkEQAAeD2WwDGRJAIAAMBAkAgAALweS+CYSBIBAABgsFmWZXm0+CYSBIBAABgsFmWZXm6COBCFRYWKjU1VSkpKbLb7Z4uB
0Ap4u834Fk0iSjX8vLyFBwcrNzcXAUFBXm6HAC1iL/fgGcx3AwAAADTSIAAAAMNIkAAAAw0CSiXLPb7Xrysey
Z1A5chvj7DXgWD64AAADAQJIIAAAAA00iAAAADDSJAAAAMNAkAgAAwECTiHLt1VdfVZ06deTv76+2bdtqzZo1n
i4JwEVasWKFevfurjSNlsNi1YsMDTJQFeiSYR5da7776r50RkPfnkk9qwYYYNatmypr1276tChQ54uDcBFKCg
oUMuWLfXqq696uhTAq7EEDsqttm3b3bqk2bNnrll1VckScXFxapVq5aGDx+uRx991991MPVASgNNptN8+fPV3x8vKdLA
bwOSSLKpRMnTMnTmj9+vWKj9vWKjY117PPPx8VFsbKzS09M9X9WBkAAJcHmkSUS7//vX=n+vWKjY117PPx8VFsbKzSO9M9WBkAAJcHmkSUS7/++quKiooUFhbmtD8sLExZWVkeqgoAgMs
HTSIAAAAMNIko16pXry5fX19lZ2c77c/Oz1Z4eLiHqgIA4PJBk4hyyc/PT61bt9aSJUsc+4qLi7VkyRLFxMR4s
DIAAC4PFTxdAHChkpOTNlZCQoGuvvVbXXXedXnrpRUUFGjw4MGeLg3Arc/ju9qRcHQ0FDVrl3
bg5UB3oUlcFCuvfLKK3uueeUlZWlVq1aaerUqWrbtq2nywJwEZYtW6Ytb+W6b0nTsb+xMSEpSWlnbpCwK8FE0iAAAAD
MxJBAAAgIEmEQAAAAaaRAAAABhoEgEAAGCgSQQAAICBJhEAAAAGmkQAAAAYaBIBAABgoEkEcNEGDRqk+Ph4x
0nTrp4YcfvuR1LFu2TDabTk50ec8x2azacGCBSW+5+5azcGCBSW+5vjx49WqVauLuquhnnn36SzWZTRkbkbGRV0HAC4lmkxjMjVo0
CDZbDbDbDbDb5+fmpQYMGjmGQaL6dGHlSic4tSWMHALj0Oi6KnkhaANlsic4tSWMHALj0Oi6AADu061bN5N62aNUuFhYhX6VMhkVMD+4HbNnwN7
YqIoVKyolJcUoBJkAAAAwkSRCxWU8RQ0APIoCAwUvJGR2eEQ2sHKVWDDMd06Ti0a9zAbRa4DjqJFSpUUHh4uL
MhINW7cWJL0888/888/goJCVFoaKgaN24gbw0wVosw/Fl8BxPyO2Q9Yz1U0E65HmGCBzmmVw9deRI2cXKUCg6a
1atWS3W5X5XgwYN9UOyaqVq1aaerUqWrbtq2nywJwEZYtW6Ytb+W6b0nTsb+xMSEpSWlnbpCwK8FE0iAAAAD
WrUSAEBAercubNTnSU1duxYNWrUSJUqVVK9evUOdtw4nTx50jjvtddeU2uU0Vtw4nTx5Ojjvt4ddeU61atVSpUiX1799puUbd1dWOX/ALldzWOqx59oQwUvzD
Hy9/fXO2aNNHO6dPPec/ffvtNAwc0VIOaNRQQEKCGDRtqlqxZLtc0OAO5Ekgh4kYCAAB0+fNjxesmSJQoKCtLix
YslSSdPnlTXrl0VEx0Jr7/+WhUqVEOr7/WHUqVNBTTz2lbt266vbvVvvpOfn59eeOEF/6P+/frb3/5
2zvvec889Sk9P9P19SpU9WyZUvt3rlbvt76q2rVqX3339f/fr10/bt2xUVdUt2o3LRs21IoVK3TXXXepoOa0tio37++Wf17dtXiYmOa0o5/J0E
BgYqLS1NkZGR2rkzR4GR2rx5s4YOHarwAwEA98sg
jjnMyMzM1b948LVy4UH15eRoyZIgefPBBBzZkzR5IOZ84cPfHEE3r11Vd09dVXa+PGjRo6dKgqgV66shIQE457jx
o3T1q1b9emnnn6p69erKzMzUsWPHX4dAnZNzKAnBZSkhIsPr06PHEIGNcZlmUVFxdbixcvtux2uzU
xntmzZ1uNGze2iouLHfsKCwutgIAA66sLmLv7KvdxJsKJYKx33siL6tixwzViXxAjLsixr+
/btliRr8eLFZ63zq6++siRZv/32m2Pf8ePHr1UqVKlmrVq1yOnfIkCHWHXfxcYVmWZaWkpFhNmzZ1O0jWme
SZM2fP/+cx5977jmrdevWjdevWjtdPPPvmk5evra+3bt8+x79OnT62FpNPP7V8fHysysbgcwCPWpZlWfXf17fmzp3drdJ1JkyZZZTTMTEx1
mVZ1u7duy1JsaNGy3Ly3LsqsqevtbgwcPPmcAFCl7FFixapWpNnqUqOnnyPzmDaunVrpmTpn3qa/m5sa1Ql6zHa0ZnNKVjPAw0Ok6x48f86d3moe
4adMmZWZmKjAw00k6x48f186d05Wbm6uDBw+qbdu2jmMVK1TQtddeaww5n5aRkSFfX1917XNixxHVnZmbq6NGju

```
ummm5z2nzhxQldffbUkadu2bU51SFJMTEyJ73Hau+++q6lTp2rnzp3Kz8/XqVOnFBQU5HRO7dq1dcUVVzjdp7i
4WNu3b1dgYKB27typIUOGaOjQoY5zTp06peDg4LPe84EHHlC/fv20YcMGxcXFKT4+XjfccIPLtQOAO9EkApexz
p07a8aMGfLz81NkZKQqVHD+K1+5cmWn1/n/n5+WrdurVjGPXPatSocUE1nB4+dkV+fr4k6eOPP3ZqzqQ/5lmWlvT
OdAOcOFATJkxQ165dFRwcrHfeeUcvvPCCy7X++9//NppWX1/fs76ne/fs/fu2rNnjz755BMtXrxYXbpOUWJiop5//
vkL/zAAUMpoEoHLWOXKldWgQYMYMSn3/NNdfo3XffVc2aNY007bSIiAitXr1aHTp0kPRHYrZ+/Xpdc801Zz2/ef
mKi4u1vLlyxUbG2scP51kFhUV0fY1bdpUdrtde/fuPWcCGRwd7XgI57Rvv/32/B/yTPaclmuGXr1Iigx8Ir1Xbo
uwxztu7d68OHDigyMhIx318fHzUuHFjhYWFKTIyUrt27dLAgQNLf08aNWooISFBCQkBM4el
mAA4DBw5U9erV1adPH3399dfavXu3li1bpoceekj79u2TJIOYMUL//Oc/tWDBAv3www968MEH/3KNwzp16ighI
UH33nuvFixY4LjmvHnzJElRUVGy2WxatGiRfvnlF+Xn5ysMFCjR4/WyjEj9eabb2rnzp3asGGDpk2bpjfffF0
SdP/992vHjh0aM2aMtm/frrlz5yotLc2lz9uwYUPt3btXeCP//E4KmpU6dq/vz5xnn+/v5KSEjQpk2b9PXXX
+uhhx5S//79FR4eLkmaMGGGCU1NTNXXqVP3444/avHmzZs2apRdffPGs933qiSf04YcfKjMzU1u2bNGiRYYsUHR3
tUu0A4G40iQAcKlWqpBUrVqh27drq27evoq0jNWTIEB0/ftyRISEhISFBMTo8DAQN1yyy1/ed0ZM
2bo1ltv1YMPPqgmTZpo6NChKigokCRdccUVmjBhgh599599FGFhYUpKSlJkjJRp0iSNGzdOqampio6OVrdu3fTxxx
rbt26kv6YJ/j+++9rwYIFatmypWb0nKlnnnnGpc978+803a+TIkUpKSlKrVq20atUqjsnWtChkRKSVD6EVZS0tQcQ
KG4uDi1aNHCaYmb++67T6+//8/rpmzZq15s2bq2PHjkpLS3PUeiY/Pz+svsurVnRxXEO0vnfO++4TsVvruJv
NOtdscwAAAHgtkkQAAAAYaBIBAABgoEkEAACAgSYRAAAABppEAAAAAGGgSAQAAYKBJBAAAgIEmEQAAAAaaRAAAA
BhoEgEAAGCgSQQAAIDh/wGqolfozqOAPAAAAABJRU5ErkJggg==\n"
          },
          "metadata": {}
        }
      ],
      "source": [
        "from sklearn.ensemble import RandomForestClassifier\n",
        "\n",
        "rf = RandomForestClassifier(random_state=42)\n",
        "model_train_eval(rf,\"RandomForestClassifier\")"
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "from sklearn import metrics\n",
        "from sklearn.metrics import roc_curve\n",
        "y_pred_prob = rf.predict_proba(X_test)[:, 1]\n",
        "fpr, tpr, thresholds = roc_curve(y_test, y_pred_prob)\n",
        "plt.figure(figsize=(6,6))\n",
        "plt.plot(fpr, tpr, label='ROC curve (area = %0.2f)' % metrics.auc(fpr,
tpr))\n",
        "plt.plot([0, 1], [0, 1], 'k--')\n",
        "plt.xlim([0.0, 1.0])\n",
        "plt.ylim([0.0, 1.05])\n",
        "plt.xlabel('False Positive Rate')\n",
        "plt.ylabel('True Positive Rate')\n",
        "plt.title('Receiver Operating Characteristic')\n",
        "plt.legend(loc=\"lower right\")\n",
        "plt.show()"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 564
        },
        "id": "SOI9nlFNyWm1",
        "outputId": "656583c7-f1bb-4b84-e3b4-6afeb0735581"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 600x600 with 1 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAAAiMAAAIjCAYAAADRKhuxAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAACFJEl
EQVR4nO3dd1gUV8MF8LPOXpQiIgp2jYqKSiyABcVeYsFoEEsOKhpL1NjRFEuMLYnGFkUNxhZ7QYVYkBg19l4Qx
IIoovS6e78//Ng3CCiLCwPL+T2PT7LD7OzZUdjDnTszMiGEABEREZFEtKQOQEREZRGUbywgRERFJimWEiIiIJMU
yQkRERJJiGSEiIiJJsYwQERGRpFhGiIiISFIsIORERCQplhEiIiKSFMsI0f9zdHTE4MGDpY5R5R5R5rRu3RqtW7eWO
sZ7zZkzBzKZDLGxsVJHKXFkMhnmzJmjmlm1FRkZCJpMhICBALduj0oFlhIpFQEAAZDKZ8o+0jg7s7e0xePBgPHn
yROp4JVpycjK+/fZbNGjQAEZGRjA3N4ebmxs2bdqE0nI3h2LOnDmIjIyU0koucrkcGzZsQOvvWrVGuXDno6
+vDOdERQ4YMwb///itlPLXYsmUULli1bJnWMHEpiJpKOjtQBqGz55ptv40TkhLS0NPzzzzzz8ICAjA6d0ncf36dRg
YGEia7c6dO09DSKln9PCYmbBu3atcOtW7fQv39/jBkzBgkJCVi5vu6+hEqzKHlDYqJDI/z7jczDdkq8DzRY9+dYd0Y
+vD0dERQ4YMwb///it1PLXYsmUL1i1bJnWMHEpiJpKOjtQBqGz55ptv4OTkhLS0NPzzzzzz8ICAjA6d0ncf36dRg
YGEia7c6dO9DSKln9PCYmBu3atc0tW7fQv39/jBkzBmlpafjzzz/h6+uLQ4cOITAwENra2lJHfaebN29i7ty5a
N26NRwdHN87ejRo9KEApCamopPPvkEQUFBcHd3x/Tp01GuXRkZi+/bt2LhLhxI6Kiolcpu1wC/4h2kl8f2bNPP3
6dYwfP75Itp+amgodKXFaSmpkJXV1eNCamkYXmhYtWpUycX9adIEAPD555555/DysuL59CvX59l+9Cv
z9Js+nr6xf7a6alUFTy/f7a6alpUFPTy/fEuTr64tbt25h+7d6d9u3L5l19+icmTJ+9cm/+mt7L5l19+
VsiO9PT21bKcwJk+ejKCgIxduTxH6K/vz+WL11arHmEEEEhLS4OOhoWGwA5HZumMjTJ+zLdC6wm99HbqRJ=
Renp6wt7eXvj4+IgXL14o10lLSx0zZ88W1apVE3p6eqJSpUpi8uTJIi0tLce2qlSpInx9fYUQQpw/f14AEAEBA
bleMygoSAAQ+/fvVy57/PixGDJkiLCxsRF6enqibt264rfffvxv0PHjwsA4o8//hAzZswQFStWFDKZTLx69Sr
PfXbmzBkBQAwdOjTPr2dmZooaNWoISOtLkZKSIoQQIiIiQgAQixYtEkuWLBGVK1cWBgYGwt3dXVy7di3XNgqyn
7P/7k6cOCFGjRolrK2thYWFhRBCiMjISDFq1ChRs2ZNYWBgIMqVKyf69Okj9W8j4DiIIcj3/7T/Hjx8XQgjh4eEhPDw
8cu2nbdu2ie+07Y29sLfX19bZtW3Hv3r1c7+GXX34rRTk5OwsDAQDRt2lSc0nUq1zbz8ujRI6GjoyPat2//z
vWy+fv7CwDi3r17wtfXV5ibmwszMxePBgkZycnGPd9evXizZt2ghra2uhp6cn6tSpl2auXJlrml1WqVBFdunQ
RQUFBwsXFRejr64tlS5aMyYh7AYlWhXxP9Wsd9JHAwbe/Hh+YuHJUNAfBHtZdFfffMGTeM51+cnf
v/9d1G3bl2ho6Mjdu/erfyav7+/ct2EhAQxbtg/ct2EhAQxbtvw45fel5tbW18PTOFBcuXHhvpux/wxs2bMjx+rdu3RJ9+/YYVV1Z
WwsDAQNsWVNMnz79XX91VIpwZIQklT2HwNLSUrnsxxoObaNmyJezt7TF16lQYGxtj+/btenJzmzJ/7880/o6UtL
JCUlAQ3NzfcunULQ4c0ORePGjREbG4t9+/bh8ePHHsLKygkKhQPfu3XX69GmMVBTr17EvdXhEb8wEcRrXR+Gi
fY7WSPkwddlgWvfKbHdpghrsQIoVsDDC0+7T07+/zdl6PkSpcTqfFXRTRbAg7Bw8WZ1H6UNhwGE3wYprDeyv
syNTXkyZNLuLVqVWzfvh2+vr45e2gkhzPwYtBzOksw63tWl0zLrtJ2e7EjzPkb/DBJrdCrLRF8ZL5FDs70KdNTBFGj
fZb60npYdKkSUhPT893ZGD//g9GoD//g7AkWXOWh3ZQqcjjeASvZnJBnTdQmX5wYPDoSjzW/ChlEMeh
ojm1GC5AT0kHX9uGbIY4N80tgx8gXvvivdh9i/Xr1n9m9Hz4y90Wi0EK7gBzOZp1BGe9tfZO4CO7e1bvw
AgGHNcUCedtrVg6ux7yH9l0Ttq4dxhEnWe4MC0qPpJ7M2nGsV5oRzFgjoGUgw7dNjW8DlxjMcbH5XZW0yN7PLCkCpDLFgr
u7u+PLLL/HTTz9h+vTpqF0nDgAo/5ufBQsWQEtLC5MmTUJ8fDdrvO2D0Bw4EGfPn1Wu8+uvv2LMDPhAk
TEBkZiZ49e8LS0vK9hlY0Hz6MrKws+Pj4vHO09t/XXlxw0Tk6YP38+L168iHXr1sHGzgYLFy7Mkeujjz5z5C9+7do
a0jg/3792P06NFQKBTw8/xRdffIHhW4ejVaka0mjI0ICAAQ4c0xUcffYRp0BwsIC1y5dQ1B
QEAYMGIAZM2bPj4eix8/Vo70mJiYAIDK3x9//fUzIzMzOzFmzBjUrVsXLo++x
0nTp3Hr1i00btz4nZnycvXqXbi5uuUFXVxcjRoyAiedV/wTGx/PsX//pz//fcz//fcF+4ujkk3qNkRlQ/Zvx8HBweLFixf
i0aNHYufK0ncLa2lro6+uLR48eKddt166uR69eKrcmaqnHVGK0r4Md6cr6dQKIQQQmzev
FloaWmJ0NDQHF9ftVACDCwsKUy/47MiKEENOmTRO6urowiLi50uSw9PV1YWFjkGK0YNmyYsLOzE7GxsTleo3/
```

```
//sLc3Fw5apH9G3/VqlWVy96lZ8+eAkC+IydCCLFr1y4BQPz0009CiP/9VmloaCgeP36sXO/s2bMCgJgwYYJyW
UH3c/bfXatWrURWVla018/rfWSP6GzatEm5bMeOHTlGQ/4rv5GROnXqiPTOdOXy5cuXCwDKEZ709HRRvnx5ObR
pU5GZmalcLyAgQAB478jIhAkTBABx6dKld66XLXtk502Rql69eony5cvnWJbXfvHy8hJVq1bNsaxKlSoCgAgKC
sq1fkG28fr1a2FqaipcXV1FampqjnWzvweEEKJLly45RkOyqfL9AUBoaWmJGzdu5NoO3hoZMTc3F35+frnW+6/
8MuU1MuLu7i5MTU3Fw4cP832PVLqVrNl6pPE8PT1hbWONBwcH9OnTB8bGxti3b5/yt9i4uDj89ddf6NevHxITE
xEbG4vY2Fi8fPkSXl5euHfvnvLsmz///////BPOzs65foMH3hx3BoAdO3agTp06qF27tnJbsbGxaNu2LQDg+PHj+Wb
19vZGZmYmdu3apVx29OhRvH79Gt7e3gDeHOP/888/0a1bNwghcryGl5cX4uPjcfHixRzb9fX1LdCcgMTERACAq
alpvutkfyOhISHH8p49e8Le3l75uFmzZnB1dcWhQ4cAqLafsw0fPjzXRNn/vo/MzEy8fPkS1atXh4WFRa73rao
hQ4bkGDVyc3MDADx48AAA8O+//+Lly5cYPnx4jsmTAwcOzDHSlp/sffau/ZuXkSNH5njs5uaGly9f5vg7+O9+i
Y+PR2xsLDw8PPDgwQPEx8fneL6Tk5Nyl02/CrKNY8eOITExEVOnTs01zyL7e+BdVP3+8PDwQN26dd+7XQsLC5w
9exZPnz5977rv8+LFC5w+LFC5w6dQpDhw5F5cqVc3ytIO+RSgcepqFitWLFCtSsWRPx8fFYv349Tp06lWPi6P379yGEw
KxZszBr1qw8t/H8+XPY29sjPDwcvcXv3fufr3bt3D7du3YK1tXW+28qPs7MzateujW3btmHYsGEA3hyisbKyUv6
wfvHiBV6/fo01a9ZgzZo1BXoOJyenAr2H7B/4iYmJ7xxCMYt4W4wTPK0agDVhsKpD2byHWHHqOC1lQH8oe6/HZkUxp2Z+dQddHfcABCzXaexX3kxsUnj+sjEeXJ3uJdKbcaD9MQ6x7Ggg7wWhg/FpE0/kWlWpF03/aPt3bqUD6RbnPa73kHF+6gEdSlEWw2iZzSJA8ZOWVW7xLIo2Pd1g0yahqT1qJT+A8bNKG6BqKmOdOT9v7+E0TAIQ8gHeHVATu9gzkvyWDVlXf7C/dOt8RKeOSNoQzrn14Ne1VHHgqOPdoMs9TS1L9rLPuz8A0qQqXOv9Cy3OVKq7mDSL4x/ynQH3xP4i/CwnyvYqTeMUAA5RnA==
```

TJCREREkmIZISIiIkmxjBAREZGkWEaIiIhIUiwjREREJCmWESIiIpIUywgRERFJimWEiIiIJMUyQkRERJJiGSE
iIiJJsYwQERGRpFhGiIiISFIsIORERCQpScvIqVOnOK1bN1SsWBEymQx79ux573NOnDiBxoObQ19fH9WrVOdAQ
ECR5yQiIqKiI2kZSU5OhrOzM1asWFGg9SMiItClSxeOadMGly9fxvjx4/H555/jyJEjRZyUiIiIioqOlC/eqVM
ndOrUqcDrr1q1Ck5OTli8eDEAoE6dOjh9+jSWLlOKLy+voopJRERUKkW9TMHN6Hi1bS85KVFt2/ovScuIqs6cO
QNPT88cy7y8vDB+/Ph8n5Oeno709HTl44SEhKKKROREVGIcvBqN8dsuIVMu1LZNRXqK2rb1X6WqjDx79gy2trY
5ltna2iIhIQGpqakwNDTM9Zz58+dj7ty5xRWRiIhIcn+ci8LO3dcgBFDDxgTmhrpq2W5mqh4eqWVLOZWqMlIYO
6ZNw8SJE5WPExIS4ODgIGEiIiKiovPriXAsDLoNABjgWhnf9qgHbS1Zobd34cIFrFixAqtXrOZqairMv1JXOv8
pVWWkQoUKiImJybEsJiYGZmZmeY6KAIC+vj709fWLIx4REZFkhBBYEHQbq08+AACMblONk71qQSYrfBEJDQ1Fl
y5dkJiYCEdHx3dOi/gQpaqMNG/eHIcOHcqx7NixY2jevLlEiYiIiKQnVwjM3HMNf5x7cxBlWqfa+MKj2gdt88i
RI+jVqxdSU1Ph4eGBCRMmQAj1zT/5LOlP7U1KSsLly5dx+fJlAG9O3b18+TKioqIAvDnEMmjQIOX6IOeOxIMHD
zBlyhTcvnObK1euxPbt2zFhwgQp4hMREUkuIOuBL/+4hD/OPYKWDFjwSfOPLiK7du1Ct27dkJqaik6dOuHw4cM
wNTVVU+LcJCOj//77LxolaoRGjRoBACZOnIhGjRph9uzZAIDo6GhlMQEAJycnHDx4EMeOHYOzszMWL16MdevW8
bReIiIqk1IysjBs43kcvBYNXWOZfhnQGP2bVf6gbW7evBn9+vVDZmYm+vbtiz179uQ7FUJdZKKoxlxKqISEBJi
bmyM+Ph5mZmZSxyEiIiqU+JRMDAk4h4tRr2Gkp43VPi5wq2H9Qdt88eIFqlatiqSkJAwZMgRr166Ftra28utF9
RlaquaMEBEREfA8IQ2D1p/D7WeJMDfUxYYhTdG4suUHb9fa2hq7d+9GUFAQfvjhB2hpFc8BFI6MEBERlSJRL1P
w2W9nERWXAhtTfWwe5opaFQo/n0MIgadPn8Le3v696xbVZyjv2ktERFRK3HmWiD6r/kZUXAoqlzPCzpEtPqiIK
BQKjBs3DgObNsTt27fVmFQ1LCNERESlwMWoV+i3+gyeJ6ajlqOpdo5sjsrljQq9vaysLAwbNgw///wzYmNj8c8
//6gxrWo4Z4SISMMJIfDPgzj8cS4KzxPTpI5DhXT1cTxSMuRoVNkCGwY3hYWRXqG3lZGRgYEDB2Lnzp3Q1tbGh
gOb4OPjo8aOqmEZISLSUGmZcuy/8hTrwyJxK5o3CdUEbjWssNrHBUZ6hf/4TklJQZ8+fXD48GHo6upi27Zt6NW
rlxpTqo5lhIhIwzxPTMPv/ORhy9mHiE3KAAAY6mqjt4s9XJ3K4wOuDk4SMtHXQcvqVtDVLvwMi4SEBHTr1g2nT
p2CoaEhdu/eXSKu1cUyQkSkIa4/icf6sAjsv/JUedt4O3MD+LZwRP+mDh8OrE+aQVtbG1lZWTAzM8PBgwfRqlU
rqSMBYBkhIirV5AqBYzdjsD4sAuci4pTLG1e2wNBWTvD6qMIH/SZNmsXY2BgHDx7Ew4cP4ezsLHUcJZYRIqJSK
CEtE9vPP8LGM5F4FJcKANDRkqFzZfTsMaemIRmq4ABZphocPH+LAgQPw8/MDAFhYWMDCwkLaUG9hGSEiKkUiY5M
R8Hckdvz7CMkZcgCAhZEuBjSrDJ/mVWBnXrT3EKHS5e7du/DO9MSjR4+gp6eH4cOHSxOpTywjREQlnBACZx68x
PrTkQi5HYPs62bXsDHBOFZO6NnQHoZ62u/eCJU5V69eRfv27fH8+XPUrlObnTt3ljpSvlhGiIhKqLRMOfZdeYr
1pyNw+1micnnrWtYY2tIJbjWsIOOpMZSHs2fPomPHjnj9+jUaNmyIIOeOwMbGRupY+WIZISIqYbJPzQ385yFeJ
v/v1Nw+LpXg28IR1W1MJE5IJdnx48fRrVs3JCcno3nz5jhO6FCJmyPyNpYRIqISIq9TcysqT82tDHMjXYkTUkn
3+PFjdO7cGWlpaWjXrh327NkDE5OSX15ZRoiIJJTfqbkuVSwxtKUTvD6yhQ5PzaUCqlSpEr7//nucPHkS27Ztg
4GBgdSRCkQmRPZUqLKhqG5/TEQlTOaWAhGxyRAoeT/mhADC7sci4O9IPH71v1NzuzSww5CWTmjoYCFtQCpVMjI
yoKf3v4vaKRQKaGmpv8QW1WcoROaISGMN23geofdipY7xXhZGuhjoWhk+Hzuignnp+E2WSo7ly5dj8+bNCAkJg
bm5OQAUSREpSiwjRKSR/o2MQ+i9WGjJgHLG+lLHyVMFc3OMdK3CU3OpUIQQ+P777zFr1iwAQGBgIEaPHi1xqsJ
hGSEijfTriXAAgHdTB8z/pIHEaYjUSwiBr7/+GosWLQIAzJO7F6NGjZI4VeGxjBCRxrn9LAEht59DSwaMcK8md
RwitVIoFPDz88OqVasAAEuWLMGECRMkTvVhWEaISOOs+v9RkU717eBkZSxxGiL1ycrKwuDBgxEYGAiZTIY1a9b
g888/lzrWB2MZISKN8iguBfuvRgMARnlwVIQOSOxMDI4fPw4dHR38/vvv8Pb2ljqSWrCMEJFGWXPqAeQKAfea1
qhnby51HCK1sre3R3BwMCIiIkrOvWZUVbrO/SEieocXienY/u8jABwVIcORHx+PEydOKB/XqVNHo4oIwDJCRBp
kQ1gEOrMUaFTZAh9XLSd1HKIP9uLFC7Rp0wYdO3ZESEiI1HGKDMsIEWmEhLRMbD7zEMCbURHezZZKuydPnsDDw
wOXLl2CmZkZypcvL3WkIsM5IOSkEQL/iUJiehZq2JjAs46t1HGIPkhERATatWuHiIgIVKpUCcHBwahVq5bUsYo
MROaIqNRLy5Tjt9MRAICRHtWgpcVRESq9bt26hVatWiEiIgLVqlVDaGioRhcRgCMjRKQBdl54jNikdNhbGKJ7w
4pSxyEqtIiICLi7uyM2NhYfffQRjhO7Bjs7O6ljFTmWESIq1bLkCqw+9eYiZ8PdnKCrzQFfKrOqV66MDhO6407
duwgKCtLoeSL/xTJCRKXawWvReBSXinLGevBuWlnqOEQfRFtbGwEBAUhLS4OpqanUcYoNf4UgolJLCKG8Id6QF
o688y2VSnv37sXQoUMhl8sBALq6umWqiAAcGSGiUuz4nee4/SwRxnraGNTcUeo4RCoLDAyEr68v5HI5mjdvjuH
DhOsdSRIcGSGiUit7VGTgx1VgbqQrcRoi1axevRo+Pj6Qy+Xw9fXFkCFDpI4kGZYRIiqVzkfG4XzkK+hpa2FYK
yep4xCp5Mcff8TIkSMhhICfnx/Wr18PHZ2ye7CCZYSISqXsUZHeLvawNTOQOA1RwQgh4O/vj8mTJwMAcp2bhp9
//hlaWmX747js1jAiKrVuRSfgr9vPoSUDvnDnDfGo9Lh37x4WLFgAAJg3bx6mTZsmcaKSgWWEiEqdVSffjIp0q
m8HRytjidMQFVzNmjWxY8cOPHrOCH5+flLHKTFYRoioVIl6mYL9V54CeHNDPKKSLjMzEO+ePIGjoyMAoHv37tI
GKoHK9kEqIip11oSGQyEA95rWqGdvLnUcondKTU1Fr169OLJlSOREREgdp8TiyAgRqUwIgfQsRbG/7svkDGz/9
zEAYHRrjopQyZaYmIgePXrg+PHjMDAwQHh4OJyceOZXXlhGiEgl6VlydP85DHdiEiXLOKiyBVydykn2+kTvExc
Xh86dO+Ps2bMwNTXFgQMH4O7uLnWsEotlhIhU8m/kKOmLiJ62FiZ3qAWZTCZZBqJ3iYmJQYcOHXD16lWUK1cOQ
UFBaNqOqdSxSjSWESJSSei9WABAz4YV8V2v+sX++rraMujr8B4OVDI9fvwYbdu2xb1791ChQgUcO3YM9erVkzp

WiccyQkQqOX3/BQDAo5Y1TPT5I4TovOxMTGBiYoLKlSsjJCQE1atXlzpSqcCfJERUYHHJGbjxNAEAOLK6lcRpi
EoeCwsLHDlyBGlpaXBwcJA6TqnBU3uJqMDC7sdCCKB2BVPYmPIS7EQAcP78eaxcuVL52NramkVERRwZIaICO/3
/8OVacVSECABw8uRJdO3aFUlJSahQoQI++eQTqSOVShwZIaICEULg9PO3ZaRlDZYRosOHD6Njx45ISkpCmzZtO
L59e6kjlVosIORUIBGxyXjyOhV62lq8xgeVeTt27ECPHj2QlpaGrl274tChQzA1NZU6VqnFMkJEBZI9KtK4igW
M9HiEl8qugIAA9O/fH5mZmfD29sauXbtgYMA5VB+CZYSICiR7vohbDWuJkxBJ58qVKxgyZAgUCgWGDRuGwMBA6
OrqSh2r1OOvNOTOXllyBc6EvwTAyatUtjk7O2Pu3Ll149eoVlixZwisBqwnLCBG915XH8UhMz4K5oS7vlEtljhA
CqampMDIyAgDMmjULAFhE1IiHaYjovbIPObSsXh7aWvwBTGWHQqHAmDFjOL59eyQnJwN4UOJYRNSLZYSI3iv7E
vCtqnO+CJUdWVlZGDJkCFauXIkzZ87gr7/+kjqSxuJhGiJ6p6TOLFyKeg0AcOP1RaiMSE9Px4ABA7Br1y5oa2t
jO6ZN6Natm9SxNBbLCBG9Oz/hL5GlEKhS3gg05YykjkNU5FJSUvDJJ5/gyJEjONPTw/bt29GjRw+pY2kOlhEie
iflVVd5Fg2VAfHx8ejWrRtCQONhZGSEvXv3wtPTU+pYGo9lhIjeKfTem/kibiwjVAY8f/4ct2/fhrm5Q4dOoQ
WLVpIHalMYBkhonxFx6ci/EUytGRAi2osI6T5atSogaNHjOIIgUaNGkkdp8xgGSGifGWf0lu/kgXMjXiVSdJMk
ZGRePjwITw8PAAADRs2lDZQGcRTe4koX9nzRXiIhjTVnTt34ObmBp+OmhS6dO+Off/6ROk6ZxTJCRHlSKATC/r+MtOI
pvaSBrly5Ajc3Nzx+/BhvVqlSBg4OD1JHKLJYRIsrT7WeJiNQGeNhpXtpQ6DpFanTlzBq1bt8aLFy/QuHFj
Dp1Cvb29lLHKrNYRogoT91lXXXV1Kgc9Hf6oIM3x119/oX379nj9njIjvCmJv/76C1ZWHP2Tt2g7w7cxIG
rOVLHULv41EwAQKsavAQ8aY5///OXnTt3Rnp6OjpO6jp06IBdu3bB2NhY6lhlHssIOQf6/exDpGUqpI5RJPSOtdC+j
q3UMYjUxtnZGRO7doSWlhb++OMP6OvrvSx2JwDJC9MGEePPf9YObwMbUQNowamZrZgBrU/6wptJPCAGZTAZdXV1
s27YN2traONHhR2BJwb8JIjWpaWuKSpa8dwtRSbNkyRLLcuXMHq1atgkwm42hICcQyQkREGvkGW+++QW+
HTv3h1dunSRiWWEiIg0sIoPyhACkydPxvr16/Hll1/ixo0bSEpKkjpSmVfaJR4tIDEjVpwZRHbSwlWf5ybnQa2qlLGu3cGGIZ2t59Rnh9Oj
ccHmUlmbdXHjyvvdSw3rBdhsSwjYxAB8Ep0IUTKWoTV6fivOLaZqAxfRq+8xO8LkIs+48YWvvH2HLZRkewCHZDCgygcLz45wd45IP4GP6y
reQf/xNRKQ+XmcGjD6KVvjn5pezR1k4HC8aAxbAuahzxRzrXqk5bwArgJwyTy7nsvtR5WQOo6d9m/XTs4DV4NHPmjDXsHfgqHSNmj2QCT
vMskI8cIZ4iAaMEfzgBdIMgVmICwXWTagmD3s4V8/PxS6fxvq1OfwVSOwSEV6r1LgShtf13bLhFx0SUrgMTsczOTyvjGOhBhlCrNZ1PH
M87h7GlygaWl7N3q4AwKoZqHZGqJ9QGDFeGqwfUrORCKH95qBlsy/mjXxKDcoHvQzUAwCEz3MdG/s6aJUgLg8ZcTDMZd8+UgLNGcy5nZ
ZWSRC8o3qc+xUc4YlF8Mjsvat9NVuFBrsNWeaHDBLhDv4sMyeUxMlvCPTmSgPAjrWUJ1VQ2uaNWpcMSK3rzWMONSsP80rbdwJmWsg2g3m
GumGRPdFx+IOHqeRkEQMnNoQ1yI1htYpkDhK0+bdz1ZdGCUcxfT2ZMBHQiOy0DFAMBMWVFOTRkI+YbSrn1Kc5gj2cN

HFysLc0xIJP6qOOnZnUUYgoH+3bt0f79u3RoEEDLFq0iKOXVCAsI5RD1Ms3h2dq2ZoiaLy7xGmIqDQQQgAAZDI
Z9PX1ceDAAejq6rKIUIFpSR2ASpaouDdlpDIvo05EBaBQKDBq1Ch89dVXylKip6fHIkIq4cgI5ZBdRqqUZxkho
nfLysrC4MGDERgYCJlMhoEDB/KGd1QoLCOUw8OXHBkhovdLT09H//79sWfPHujo6GDz5s0sIlRoLCOUw6PswzT
ljSVOQkQlVXJyMnr16oVjx45BX18fO3bsQLdu3aSORaUYywgpCSE4Z4SI3ik+Ph5dunRBWFgYjI2NsW/fPrRt2
1bqWFTKsYyQ0oukdKRmyqElA+wtDKWOQ0QlUFhYGP7++29YWFjg8OHD+Pjjj6WORBqAZYSUsg/R2JkbQk+HJ1o
RUW6dO3fGxoObOaBBAzg7OOsdhzQEywgpcfIqEeUlIiICurq6qFSpEgDAx8dH4kSkafjrLynxtF4ietutW7fQq
lUrtG/fHs+fP5c6DmkolhFSyr76qgNHRogIwMWLF+Hu7o6nT59CW1sbcrlc6kikoVhGSOlOTCIAwMmKp/USlXV
hYWFoO6YNYmNj0aRJE5w8eRJ2dnZSxyINxTJCAIC45AzcjE4AADR1LCdxGiKSOrFjx9ChQwckK
F++vNSxSIOxjBAAIOx+LIQAalcwhbVpybpbLxEVnyNHjqBr165ISUmBl5cXgoKCYGbGO2VT0eLZNAQAOHOvFgD
QqrqVxEmISEr169dHpUqVOKhRIwQGBkJfn7+cUNFjGSEIIXD6/v+XkRosIOR1WcWKFXH69GlYW1tDR4cfEVQ8e
JiGEBGbjCevU6GnrQVXJx4XJiprFi9ejMDQzs7FhEqVvzXRspREZcqljDU05Y4DREVFyEE/P398e2330J
bWxuNGjVC3bp1pY5FZRDLCP1vvggPORCVGUIITJw4EcuWLQMAfPvttywiJBmWkTIuS67AmfCXADh5laiskMvl+
OKLL/Dbb78BAH755Rf4+flJrKMpaRMu7K43gkpmfB3FAX9ezNpY5DREUsIyMDPj4+2L5907S0tLB+/Xr4+vp
KHYvKOJaRMi77EE3L6uWhrSWT0A0RFbXAwEBs374durq62LJlC/r06SN1JCKWkdIqPUsOIT58O6H3XgAAW1W3/
vCNEVGJN3jwYFy9ehUd0nRAp06dpI5FUijfXq1SsYGhrCwMAAMpkMS5culTo
SUQ68zkgpdPr/L92uLm41rHinXiINFRMTg9atW6Nfv37IzMyUOg5RnjgyUoot+KQ+ujpX/ODtGPPaIkQaKSoqC
u3bt8fdu3fx/PlzPH78GE5OTlLHIsqFZaQUM9DVhok+/wqJKLd79+7B09MTjx49QuXWOgtDgBTb+/e4vwawQjbKyw/PSQHeHN6pDPyTbTqOPckUodb8Tt4ZDyMgclE4QU8VSHdAgVYvoGQvIrYSfSz2OpO/OcFt0MvpiX7xwCP8qAoSKRJmtKWn5oNJsFxkDGEwn0/gVIpbNb2ckS1a1axsDbD+Sx9t20tyGo6H8AUkwdCfRfdVglxhUhKGNt8zB8rBFUJVy6kDT0E5hWWNMvafd4ydR9k6fnE/Lxa7gF5i48nTp+Eh+QaoWudr0D18zDpmtDIYV/h+19cZTHRXH4XERavo8dl+/SeTHRw0jH/vJv8JuI3d2qsVBSeqT8mSM96QYDzZ0pDtGSsaISdpotUU+lrv1Egj2xQw9FjwBQdw0T5DcTVBU1eBkDjaWLrIMM0NXEvjI4wLbZfybNV0NJuCnPm5h0DWLq3rxeuh0/AGxJMh1UyK/JhEMXmG8y4WqwIMbP4DOU6r3R5g19wq3DO63RFLFD05Ql8J8H2N1Ee3j3k+Rf9Nq1g+0wfUBNONxymGgNY9ooBkXiNYfZu2mE3oS7aOFvCmZEmh5QtRdyqLRxIfmDpl7QRMfc6d7/ra+d4rFhF/bZHeGJKy3H/ngpsHvaE4RsPl/+JPUmptFV0Mt7RJAWhJdkS1X/3W+Fzx0E4pfJkTtFa9ZnKH5O5wLWPlH/13r+GT76QnqvOUK8VGZ0sSZBRfI/y8pOA/HIoQg4SbxzdBXf8ANddDgzXAGRzEWg9T7kmfL6fNOSzi6pqPm3+p3bW0eXWIJFTxnVJbFvzXNvVx3fdP3bXN9+gVNn+m8hZVFNhZj1tbv5wUH+tN3ZRAmQO47XABwfOnhb8Wx1r3ko89CFTdfJpQ19HPPe0Whu9U

vX3z88cfYvn07KlWqVKgXd3V1RdOmTfHLL78AeHMzPgcHB4wdOxZTp07Ntf6qVauwaNEi3L59G7q6uoV6zeK4a
68QAucjXyEuOf096wGTdlxBcoYcB8a2Qj178yLJQ0TFR6FQYOLEiRg6dCgaNGggdRwitSqqz9AClxFjY2PcuHE
Djo6OEEJAX18fx48fR8uWLQv1whkZGTAyMsLOnTvRs2dP5XJfX1+8fv0ae/fuzfWczp07o1y5cjAyMsLevXthb
W2NAQMG4Ouvv4a2dt6HONLT05Ge/r9SkJCQAAcHhyItI3/djsHQgH8LvL6lkS4uzGzPq60SlVKPHj1ChQoVCv1
LElFpUVRlpMCHaVJTU2FkZAQAkMlk0NfXV57iWxixsbGQy+WwtbXNsdzW1ha3b9/08zkPHjzAX3/9hYEDB+LQo
UO4f/8+Ro8ejczMTPj7++f5nPnz52Pu3LmFzlkYz+LflB9zQ13UsHn3vWW0ZDL0b+bAIkJUSt24cQPt27eHu7s
7AgMD8/3FiIjyp9KpvevWrYOJyZsP16ysLAQEBMDKufpqEV5ozyFQgEbGxusWbMG2tracHFxwZMnT7Bo0aJ8y
8i0adMwceJE5ePskZHi40pUDmsGNXn/ikRUKl24cAFeXl54+fIlbt68ifjj4eJQrx7PiiFRV4DJSuXJlrF27Vvm
4QoUK2Lx5c451ZDJZgcuIlZUVtLW1ERMTk2N5TExMvufi29nZQVdXN8dvHnXq1MGzZ8+QkZEBPT29XM/R19fnb
bmJSO1CQ0PRtWtXJCQkoFmzZjh8+DCLCFEhFbiMREZGqvWF9fT04OLigpCQEOWcEYVCgZCQEIwZMybP57Rs2RJ
btmyBQqFQ3h/n7t27sLOzy7OIEBEVhSNHjqBXr15ITU2Fh4cH9u/fD1NTU6ljEZVakt4ucuLEiVi7diO2btyIW
7duYdSoUUh0TsaQIUMAAIMGDcK0adOU648aNQpxcXEYN24c7t69i4MHD2LevHnw8/OT6iOQURmze/dudOvWDam
pqejUqRMOHz7MIkL0gVS+HLw6eXt748WLF5g9ezaePXuGhg0b0bIigoSDmpNSoqKsfttR0cHHDkyBFMmDABDRo0g
L29PcaNG4evv/5aqrdARGWMhYUFtLS00LdvX/z+++NjCC1iNNNClS5f
QoEEDnjlDZU5RfYZKepiGiKg0WL58O0W7evKl83KhRIxYRIjViGSEiyocQAjNmzMD48ePh6emJuLg4qSMRaaRCl
ZHw8HDMnDkTn376KZ4/fw4AOHz4MG7cuKHWcEREUlEoFBg3bhy/fjxP3SUqIiqXkZMnT6J+/fo4e/Ysd48ejR49/Y
sdu3ahaSkJADAlStX8r3wGBFRaSKXyzFs2DAD8/PPPAICVK1diypQpEqci0lwql5GpU6U6fiiu+++w7FLjx3LMIm/bt
i3++ecftYYXrjX48cgfrQh9IHY0ICikjIwOffvopAgICoK2tjU2bNmHUqFFSxyLSaCqf2nvt2jVs2bIl13IbVrIl13IbGxv
ExsaqJVRp9So5A78cv698bGtmIGEaIiqMuXPnYseOHdV1cW2bdvQq1cvqSMRaTyVROYsLCwQHR2da/nly5dh
2+vllClVZbif2dJr/ZxwbTOtSVMQ0SFMWXKFLi7u7u2P//vjzFSqxmBf8/wRDTSFROWRkf79++VPoUiMvqNYYWMLJKo
TJmHQoEFFkbHUkckAr4/yvr80EZU8qampMDQ0BAAYGRnhwIEDEpchKm3ydZ5h
i3++ecftYYrjX48cgfrQh9IHY0ICikjIw0ffvopAgICoK2tjU2bNmHUqFFSxyLSaCqf2nvt2jVs2bIl13IbGxv
ExsaqJVRp9So5A78cv698bGtmIGEaIiqMuXPnYseOHdDV1cW2bdvQq1cvqSMRaTyVR0YsLCwQHR2da/mlS5dgb
2+vllClVZbif2dJr/ZxwbT0tSVMQ0SFMWXKFLi7u2P//v0sIkTFR0WRkf79++Prr7/Gjh07IPPJoFAoEBYWhkm
TJmHQoEFFkbHUkckAr4/yvr80EZU8qampMDQ0BACYm5vjxIkTkMl4J22i4qLyyMi8efNQu3ZtOGBoHq8hIdeU3ZsAa2tnR3y+X4nlcfa3QmbTD
oW7uztatGBiBmTNnFkVGIqIi8+zZMzRr1gyLFi1SLmMRISpehb4CaR1UFK5fv46kpCQ0atQIsNbN8UtVc3DQ0atQINWrUUHe2IlGUV2B
9kZiOpt8HQyYDIuZ3Ueu2iUj9Hj58CE9PT9y/fx92dna4efMd3uqzv3Rj5tGqcTwxfJZVdsBzFSZwd9wWhtN2xpI5Rs4HW5eQXkH5
Lmy2oIQERWnu3fvwtPTE48ePYKjoyOg4NZRIgkovJhmrZt28LJyQnTp0/PcXlkIqLS4urVq3B3d0ejR49Qu3ZtHDhw48YoGJ9UKm
thIaGolq1alLHIiqzVC4jT58+iVdffYWT4qfu8+Ytx6qk1DmRQVDwTnRT4UmdgSC5hbOyrkHyPu2ZDrXU6Rb5ABRc
55EpUqVpI5FVKapXEasrKwwZswYhIWF4e7du/D29sbmzZvh6OiItm3bFkVGIqIi8NXXZLfmNlbbaaJBMzFrDoXMw
NjdSRiMo8leeM/JeTkxOmTp0KZ2dnzJo1CydPNzX0RDUmMgQ6/Ewe8TUvoK2YsUNwOCnLzlb8JpRk6zpbBGnRHYPAs5Ly
NnZYcCAAahXrx4OHjyozmxERGpx6NAhvHrSvl44MCBLCJEJYjKZWTatGlwcnLYRVBy0YcqNlnecGQULVheXL1+PZs2fYvHk
zOnbsWBQZiYgKbf369eRqeMQUR5UPkxz6tQpTJ48Gf369GfPjqvYNSyr0NkOYbf369YOVlVVRZCIiUovly5z/PjxAABnZ
2cYGPB+UUQlkcplJDhkJCwsrvCsrChyEBGphZWVVdjRACF7//PjXVWohCpQGdHzhi5hD1cRw+2CA+hBx8aZzbeWVb3ryvVoz6QE+cGRmUfnOkZiBqWsOo2gUk07aJSShRcKwEXmFGosQ2QGkbmJ6eb2yU7uXCs1tf7uF1f5NN4GHDsm+o0oXTs30gCeQ2x5Ejeof
2bKrKpwB9N0kJ4MaNqlnRvSB3d/mlX5Vc2Tzu73biJdVDaTO9jEBSUbW+6UZrzf6HXR9gL6MuO3Y83a8bRRxO5zdtPkDKiA7U2dFwpmqlbQqc9WvuHzQYJlmwXUAUOsd2uDPbA6cxCeMCEiI3DQI2u3Oc8CYPBGmoAz8AFqiu7XmFvfgKjaZAtl6LrbxyTq9zx6L5mnVYCdEhQbbCYoUMPsGYf4zU9uEoOa/Bi3QkbZNLAGtwLINbnxDFyijHmlrkqdhE3jxXd0iIBTVx3yOQZYEJLljtE6vqYs6xVPkKhXIqRMPZgsSCWCOdHUd9xyHFsDdn8xQxtZZe4v7Pbt

BFpnri4OHTu3Blnz56FqakpDhw4AHd3d6ljEZGEVC4jCoUCurq5bwSnq6urvP4IEVF+Jk6ciLNnz6JcuXIICgp
CO6ZNpY5ERBJT+TBN27ZtMW7cODx9+lS57MmTJ5gwYQLatWun1nBEpHkWL16M9u3b4+TJkywiRASgEGXkl19+Q
UJCAhwdHVGtWjVUq1YNTk5OSEhIwM8//1wUGYmolIuPj1f+f/ny5XHO6FHUq1dPwkREVJKofJjGwcEBFy9eREh
IiPLU3jp16sDT01Pt4Yio9Lt27Ro6dOgAf39/jBw5Uuo4RFQCqVRGtm3bhn379iEjIwA8
+fPo2PHjoiLi8Pq1asxbNiwPOecEVHZVuAy8uuvv8LPzw81atSAoaEhdu3ahfDwcCxatKgo8xFRKXXy5El07do
VSUlJcHV1xeHDh1lEiChPBZ4z8ssvv8Df3x937tzB5cuXsXHjRqxcubIosxFRKRUUFISOHTjbdq0wbFjx
2BpaSl1LCIqoQpcRh48eABfX1/l4wEDiArKwvROdFFEoyISqedO3eie/fuSEtLQ9euXXHoOCGYmppKHYuISrA
Cl5HO9HQYGxv/74laWtDTOONqamqRBCOiOunu3bvIzMyEt7c3du3aBQMDA6kjEVEJp9IE1lmzZsHIyEj5OCMjA
99//z3Mzc2Vy5YsWaK+dERU6kybNg21atVCz549eSdvIiqQApcRd3d33LlzJ8eyli1b4sGDB8rHMplMfcmIqNQ
ICAhA7969YWpqCplMxvvMEJFKClxGTpw4UYQxiEgIO3hBCYPnO6FixYgIObN+LYsWPQ0VH58kVEV
MbxpwaAhUF38OfFxwAAmQzQ1VH5wrREZY5CocDYsWOVZ9V16dKFRYSICoU/OQDEp2YCAFyqWKKPSyWYGfBaCET
vkpWVhaFDh2Lz5s2QyWRYtWoVRowYIXUsIiqlWEb+o69LJfRvVlnqGEQlWnp6OgYMGIBdu3ZBW1sbmzZtwoABA
6SORUSlGMsIEalk+PDh2LVrF/TO9LB9+3b06NFD6khEVMpxcgQRqWTSpEmoVKSDh48yCJCRGpRqDISGhqKzz7
7DM2bN8eTJO8AAJs3b8bp06fVGo6ISgYhhPL/GzRogv37/NO3USkNiqXkT///BNeXl4wNDTEpUuXkJ6eDgCIj
4/HvHnz1B6QiKQVHRONV1dXhIaGKpfp6+tLmIiINI3KZeS7777DqlWrsCoiUuXryo1nBEJK3IyEi
4ubnh/Pnz+OKLLyCXy6WOREQaSOUycufOHbi7u+dabm5ujri4OHVkIqISICMnz8/MHnT4Ji5Kr2cOHbi7u+dabm5ujtevX6sjExGVALdv3OarVqOQHh6OqlWr4uDBg7y8O
xEVCZXLSIUKFXD//v1cy0+fPo2qVauqJRQRSevy5ctwd3fHkydPULduXYSGhsLJyUnqWESkoVQuI8OHD8e4ceN
w9uxZyGQPyPH36FIGBgZg0aRJGjRpVFBmJqBidOXMGrVu3xosXL9C4cWPHkSFStWlDoWEWkwla8zMnXqVCgUC
rRr1w4pKSlwd3eHvr4+Jk2ahLFjxxZFRiIqRr/++ivi4+Ph5uaGAwcOwAmd7t+PRsmVLHDx4MMdduYmIioLKZUQmk2HGjBmYPHky7t+
/j6SkJNStWxcmJiZFkY+IitnatWvh5OSEKVOmwNjYWOo4RFQGFPoKrHrHp4KH6eqhbt646sxCRk1VYMlpa0
NfXx9y5c6WORERliMplpE2bNpDJZDNpDJZPl+/a+//vqgQERUvNatW4cRIOZg7NixWLZs2Tu/v4mlIoLKZaRhw4Y5Hmd
mZuLy5cu4fv06fH191ZWLiIIrBOqVLMXHiRABARkYGhBAsIORU7FQuI0uXLs1z+ezZs5GUlPTRgYio+Wfz3cg2Akh8MO33
2DOnDkAgClTpmDBggUsIkqQkCbXdKO+zzz7D+vXr1bU5IioiQghMmTIFM2bMAICvvvsOvz77LIsIEUmi0BOC/y0sLEx5p19iUkzFIkRUidoIgGLEXAvrIRAdHY1/
//OXs2bNbAeL5/5iaoio5n8xLeNu5sgUpOlJ1iMxsXjMJwKTmGIK1wqV0npUwAAAVCxjMjlcgwZMgT169e
HpaVlUWUiJikrC2LFj8c0338DBwQEAWESIoISqrERaQKrtrY2Y2nTooPa7865sQKO9o4wMDCAq6srp07V6Dnb
d26FTKZDD179lRrHiJN8fr1a3To0AAEBAQHo1asXhBBSRyIiykXls2nq1auHBw8eFEUWoi0IoukPsMdMiTcatWMDvz5c7Rp0wZnzpyBpaUlli1bxiJCRJJRuYx89913mDRpEvbv34/
o0hQ4ZIHYeISPUy8vQrqKnUsIq8FbiMfPPPNN6i7t+u
wdnaGl5cXnj9//s7nRUZGZGYtKkSXBzc1NbFi1NbFiJN8vz5c7Rp0wZnzpyBpaUlq9Ynzx2IZJNPUYh9Er9wKxLLyXYzxfdsbIPNPfbd5g0aRIOH
DiA6OhoJCQk5PijjIVLmD48OEYMmQI6tatil1WrVshIyOidpwnL5XIMHDgQc+fORdWqVVV+TSJN+/jxY7i6uiLR+P
y5cuwtbXFiRMn4uKjOrqKnUsIq8FbiMfPPPNN0hOTkbnzp2xevIssLS1hYWEBuLvw5coVd0/eHZUqVYYKlpSUsLS1hYWGWh8W8jySX0vDSiBJRWMXLhwA
Z6env8LpKUFTO9PnDlz5p1ZbGxsMGzYPez+Rnp6+gcXJqLSJDw+cHBxw6tQpNGjQQpYRET5KvA
E1rlz52LkyJE4fvw4Lly4QF4CorxKRCVegctIu9ZbbW2zVZbbWzVEgehTdeMPNo3uD5NNQQ4GjHeNAEhISFCeUUCkiTZu3J
EhUr14dISEhnKxKRCVegctI9sQ3Dw+PIgvzPomJZjhtfBDx8cHatWthZWVMmzZNmzZN
Ro0dj5cqVqFChgtRxIjeS6VTe9U9+c3Kygra2tqIiYnJMTwmJgYVKlTA3Ed8OHCJ+YZDvNZjmJbtTkJvDnbZm4eGCKfJjStbqjvhSVSS1XmfNLtm8XcsaPuWJ4IAhZ
3fu3
EG1atVyPEdfXx/6vpqzU1U0jx//hw2NjYAgAoVKmDXrl0SJyrFkT5cqVe+cfVejp6cHFxQUhISH3R+aQUhISH
KZQqFAiEhIWjevHmu9WvjWvXro1r167h/ywjR6UAVWNSJrkpFn1MhzHfH6yLJKySoVg5bTZDpDXCyxMlJD2TtU8EJp4E1ct
txcV2D9UBMnToSvry+aNGmCpl274+L5MxzeEQF5tEL7gBxm35Q3
9+/dHZmYmduzYgc8++4yn7xJRqaNSGenlfvL298eLFC8yePp5q9nj1Dw4YNERQUpJzUGhUVBS0ttdlcm
EhjbN68GUOGDIFcLkffffvn3x+++/s4gQUalU4DJSlD/kxowSgT58+OHz4MPTuzGK5NnXvnpG1lK9ZZ3MBY8NpKxp4c+5m4BE7Y2j2Sg4/sVpsQQrVWbNMS9PLDMg2FwpPM79+BKuFsM2+h4XkMcvhkrjqpaI9qRR5cUMJ6L0T1T1m5IInhlv
AIUOGYO3atdDW1pY4GRFR4RRoAuvPP/+Mb775BrVr18bdu3fRsmVLfPHFF8pHnTp1kJTr8VOlzhJwTnL1qImP4Q5J5t5K5J+wA8Pq
2++4aEZIir1VJozQkTSmzdvHtq2bYv27dvnmgxMRKUxmZK6
bLuLCwzipoxDlKSMjA59++ikT56M/v37c+4WEWmMAp3am5dTp04hKSlJnVlUVlpO7SUi1bm6uqJLly5o2LCh1FGIiPJUbKf2EhEVldu3buHFiydS8wQY8ZFOnh0rHZ6ePSGGr+jMfPDUd0Q9Rbb2jpsGqwh9i6cj7J3s6udWtZEe9L03+/Yr/rbbHKOE2vuhm1v3ECW6rNsZ+UmxY5yGZkV0pHpjNd6D0sSIxKqR7pJv3t0qhpVUVjbdPsyCWGRVBMd4ph+wPHpw4OXz1llHkjgh2mPG+0iHfWYPhphe0m3LUGRJjISp8I5fMDXn6tx77RIUqlThzP3T+Vd2wCWFhZ6bzkLjo5O4AaGpfr68vD5m2kc15hqGp6CMP3+I5PI8hpjkSlGXPpNPk6TSC2xNdWg6pNg9JlGxhyGHAgHuWmz7ZCcWdNkQGOkvK7KkKjRqO5fwx5DDDypD94ZspHuZJwTbO3J7Ppr8vbpkK9xEoOKzrp5AtADNmX2xYoKTj62j5ixFA4nEjTiXoTG6NqrP/N5aDhxAy2W+cwvB93XLQ5MxbFjqMVGpHyJ0TGc5Ft66k7ZlDk5QCEdRGr2xXCLMj9nsOC+C6FmUcVlmOuJ2M5uSSKexZQnTNqn7z4M9m8nKgm8LXvLNmpjW7Vh+cR1pI5kdPFH5ukS4Y4dDpe2u5jlrkfqX79J06bmqL7kWHH6yj6tmdEnJxrcJXtKoiMWn67E5pnJFOy3nZ8NsUUQOrxqWYWt3ULaYaxoyJLNo0G9Zqjcc1NVmTV4vSdAbhklhQKl2kcQrl9rWlnKVmj8o5ZJaMWOsT2UsJ8dkRxS6lKTwAuKj5B8FVHjtTJbBh7hILj4MdpMF5MTgQtXm4nwtcR4Gkc3Uhsp7cZtkYgzUNTTEiElrJlBnhmbfbIsazIy4JxEB8Nw1ZDHLzYkDm1F8Dqbrf7L6JHfk0jrXkzUE5qZkaOYpdQqIWwK9nFqO8TLJxTO6FLmxjPoBoHUtoBEpNuLsRQwzoVfMnAqdhiVN+Fd9LfZGWI2G/TxwtiKNrkjQK9C4HTAV+6KCFpIPdmzJkDT6KcTUZ1G7Kp8HaP3vvj8QaX4Ne6cnbRSFx4fojpo+PmaQeJntIeekWpjdXC71XZhCv/74M42dSTbtm1OXWjTBntOUuoUg2DAS9vTRC0DVR3jk3HxOzKLR2KAqjDiwbPD5XQ8W+YSZSv1WdEJpgGJ9TmGtKzd+AuClKWM8vxHxl2Ztq4AIhAcZBoStO+qZERvbv+lkBEH2EKPaoyVE5Ahn8yhQVyA19BbNMqqCJkR6ZmjNxqI9kDzG2Z6Wqz+MBQ+VKGZhUMCyZjlkcR2d+bTF44hrQ==

```
qVasAAFZWVhInIiIqmVhGiIpAVlYWBg8ejMDAQMhkMqxZswaff/651LGIiEoklhEiNUtPTOf//v2xZ88e6Ojo4
Pfff4e3t7fUsYiISiyWESI1EkKgZ8+eCAoKgr6+Pnbu3ImuXbtKHYuIqETjBFYiNZLJZPD29oapqSkOHTrEIkJ
EVAAcGSFSs8GDB6NLly6wtraWOgoRUanAkRGiD/TO6VPO7NkTMTExymUsIkREBceREaIPEBERAU9PTzx48ACZm
Zk4ePCg1JGIiEodjowQFdKtW7fQqlUrPHjwANWqVcOKFSukjkREVCqxjBAVwqVLl+Du7o6nT5/io48+QmhoKBw
dHaWORURUKrGMEKkoLCwMbdqOQWxsLJoOaYKTJO/Czs5O6lhERKUWywiRChQKBcaMGYP4+Hi4ubkhJCQE5cuXl
zoWEVGpxjJCpAItLS3s2bMHQ4cORVBQEMzMzKSORERU6rGMEBXAw4cPlf9fpUoV/PbbbzAyMpIwERGR5mAZIXq
P1atXoOaNGvjzzz+ljkJEpJFYRojeYdGiRRg5ciQyMzNx+vRqeMQEWkklhGiPAghMHv2bEyZMgUAMHXqVCxZs
kTiVEREmolXYCV6ixACEydOxLJlywAA8+bNw7Rp06QNRUSkwVhGiP5DoVBgxIgR+O233wAAP//8M8aMGSNxKiI
izcYyQvQfMpkMhoaGONLSwvr16+Hr6yt1JCIijccyQvQfMpkMy5cvh4+PD5o1ayZ1HCKiMoETWKnMSOxMxKxZs
5CRkQHgzYXNWESIiIoPROaoTIuLiOOnTp1w7tw5PHnyBOvXr5c6EhFRmcMyQmVWTEwMOnTogKtXr6JcuXIYPXq
O1JGIiMoklhEqk6KiouDp6Yl79+6hQoUKOHbsGOrVqyd1LCKiMollhMqce/fuwdPTE1FRUahSpQqCg4NRvXp1q
WMREZVZnMBKZUpWVha6du2KqKgo1KxZE6GhoSwiREQSYxmhMkVHRwdr1qzBxx9/jFOnTsHBwUHqSEREZR4P01C
ZkJaWBgMDAwCAh4cH/v77b8hkMolTEREwJERKgOCgoJQoOYNXL9+XbmMRYSIqORgGSGNtnPnTnTv3h2PHz/mX
XeJiEoolhHSWAEBAfD29kZmZia8vb2VqqSMREVEeWEZII/3yyyyy8YMmQIFAoFhgObhsDAQOjq6kodi4iI8sA
yQhpn/vz5GDt2LABg/PjxWLt2LbS1tVSVORURE+WEZIY2SmZmJIOeOAABmz56NJUuWcLIqEVEJx1N7SaPo6upi
7592LNnDwYBgvPzyzKAMs/C/rQnSQkhMGnSJDRs2BBBQUFSxyEiIMGnSJDRs2BBBQUFSxyEiIglwDJwkI5fLMXLkSKxbt
BKZmZnw9fXFH3/8AS0tLaxduxZDhw6VYREUmAZYSKXVpaGvr164f9+/dDR+Nevn9SxiIhIIiwjRU+hCzZeWF58+f57n+iRMn8Omnn+L
48eM4c+YMHBwzx58qSYk1NhZM8POX/+PFq3bi11HCIiKgEkLyNLlizB8OHDh1lEiIhI+jKybds2+/f78+g
q9fn+f6gYGBGD16NBo2bIjatWtj3bp1UCgUCAkJAkJKebkVFCPHz/GpEmTkTkJWVBQDQ0dFB7dq1JU5FREQlhaRzRjI
yMnDhwgVMmzZNZZNUxLSwuenp44c+ZMgbaRkpKCzMxMGJqa5rk+PT1d5a7DSUlJAICUlJRc16SqSbQzaKNfZ3Pb/a3fPl
lpa+OGHH6SOREREJYykIyOGBwcDg/jQXBIgEZzZXbXNsdzW1hbPnj0Or0Da+/vprVKxYEZ6enn+ff78+TA3N1f+cXBw+OD
cVDA3btyAm5sbHj58iOrVq2PMmDFSRyIiohJI8sMOH2LBggXXYunUrdu/eDQMDgzzXmTZtGuLj45V/Hj16VMwpy
6YLFy7Aw8MDOdHRqF+/PkJDQ1G5cmWpYxERUQkk6WEaKysraGtrIyYmJsfymJgVymJgYVKhQ4Z3P/fHHH7FgwQIEBwe
jQYMG+a6nr6/Pu74Ws9DQUgWJiYlo1qwZDh8+n09hNCIiIklHRvTO9ODi4pJj8mn2ZNTmzZvn+7wfCgB3
377LYKCgtCkSZPiiEoFlJSUhF69eiExMREtMREeHh4IDg5mESEioneS/DS/DDNxIkTsXbtWmzuBG3bt3CqFGjkJycjCF
DhgAABg0alGlGOC68KFCzFr1iysX78eNsZ49e4akpCQkJSVJ9REojvT/Xf07NkThw8fhqmpdSRiIioh
JP8Cqze3t548eIFZs2ahWfPnsKOFSZs+ejWfPninv3po9qTUqKirHreR//fVMFZGRkoE+fPjm24+vri9mzZxdnVPqPjzPO8k2pK
jOnbsKHEiIIqLSQvIwAwZsyYfM+wOHnyZI6vIyMjiz4QFRMXR4OIIY2Hk9UgqCF1HCIiKmUkPOxDpdvCh
Qvh5+eH6Oho/PHHH1LHISKiUohlhApFCIEZM2bA6tSpAIApU6agQIECEqciIqLSqEQcpqHSRaFQYMKECVi+f
DkAYOHChZg8ebLEqYiIqLRiGSG1yOVyfPHFF/j9998BAKtXr4afn5/EqYiIqDRjGSG1JCQk4OzZs9DS0sLvv/+OQ
YMGSR2JiIhKOZYRUoulpSWCgoIQFhaG7t27Sx2HiIg0ACew0nvFxcVh3759yq8dHR1ZRIiIqNCwjNA7xcbGwtPT
E7169cKOHTukjkNERBqIZYTy9fTpU3h4eOD8+fMwNzdHrVq1pI5EREQaiHNGKE9RUVHw9PTEzZs3YWNjg+DgY
NStW1fqWEREpIFYRiiXe/fuwdPTE/fv30flypUREhKCjz76SOpYRESkoVhGKIeXL1/Czc0NT58+hbOzM0JDQ1G1
alWpYxERkQbjnBHKwcLCAp9//jnq1KmD06dPs4gQEVGRYxmhXGbMmIG///4bFStWlDoKERGVASwjhODgYLi7
LdVL69/y+ZT2ETSw1ojG5xCUzspdpFsSOdStLaIyJajFGyDEdy4gNtyHllN1BnZdlnh2ywjAbRHLRawhDGlRMZjfiHs0S/dlX20hDhEmZoeEXLIKXWFIByrWj
7DGIFIHwk1rXd+ztmCtRqUjLWHRoeHaZ4sHJ6O7fXz9G3C+u7X/ReoJdKmJ/Eao/lOYbRNMCdq4/U4WO45bYtc+B
7Vv1tm47O2EOT+WVcVoV/Y+QJ/Mv/93Jtm2RBuhzdIJ1P8dhpIHgePubFUUW88XGw7CsV6m0J1DLl9ftxLd3r5O
7dxVLWiEOGz9WsOXIEbUXHXr5/N+ejoqa/2Q==
```

iopyu5hmrTs+9LoSZykaLx+/RpdunTB33//DWNjY+zbtw9t27aVOhYREVEuZbeMpGruYZoXL17Ay8sLly5dgoW
FBQ4fPoyPP/5Y6lhERER5KrtlREPnjMTGxsLDww03bt2CtbU1jh49ioYNG0odi4iIKF9lt4ykZgLQ0rgyYmFhg
Y8++giJiYkIDg5GrVq1pI5ERET0TmW2jMSnZgAyA1hoWBnR0dFBYGAgXrx4AXt7e6njEBERvVeZPZsmJUMBQDM
O01y8eBETJkyAQvHmPenp6bGIEBFRqVFmR0aAN3fsNTUo3bsgLCwMnTt3RkJCAuzt7TFp0iSpIxEREamkdH8Sf
4DO9StgWNuPoKNdegeHjh07hp49eyIlJQVubm4YMWKE1JGIiIhUVno/iT9Q+7q2aOpYTuoYhbZnzx507doVKSk
p8PLyQlBQEMzMzKSORUREpLIyWOZKs8DAQPTp0wcZGRno3bs39u7dCyMjI6ljERERFQrLSCnz9OlTfP7555DL5
Rg0aBC2bt0KfX19qWMREREVWpmdM1JaVaxYEYGBgThx4gSWLVsGLS32SSIiKt1YRkoBIQRevnwJKysrAMAnn3y
CTz75ROJURERE6sFfq0s4IQQmTJiAxoOb4+HDh1LHISIUjuWkRJMLpdj+PDhWL58OR49eoSTJO9KHYmIiEjte
JimhMrIyICPjw9bICPjzIcPjw+2b98OLS0trF+/HoMGDZI6FhERkdqxjJRAqamp6Nu3L4XEB37ccf6N27t9SxiIiIigT
LSAmTmJiIHj164Pjx4zAwMMDu3bvRsWNHqWMRREVGZaREiYrKwuxsbEwNTXFgQMH4O7uLnUkIiIsUyUsJYW
lri2LFjePz4MVxcXKSOQOREVOR4Nk0J80jRI2zevFn52NbWlkWEiIjKDI6MSOz+/fto164doqKioK2tjXHjx0GdowHeeePXsGZ2dnhIaGwsHBQepYREREx
Y6HaSRw6tQpdO3aFYmJifj4449x6NAhWFpaSh2LqEQTQiArKwtyuVzqKEQaTVdXF9ra2sX6miwjxSwiIgJeX15
ISOtD27ZtsXfvXpiYmEgdi6hEy8jIQHRONFJSUqSOQqTxZDIZKlWqVyfTSwjxczJyQlfffUVrly5gh07dsDAw
EDqSEQlmkKhQEREBLS1tVGxYkXo6elBJpNJHYtIIwkh8OLFCzx+/BgatQothESlpFiIpfLLlX+p3377LeRyOXR
OuPuJ3icjIwMKhQIODg4wMjKSOg6Rxr02tkZkZCQyMzOLrYxwAmsx+Pnnn9GXTvlELNMJmMRIVKRlhZ/XBEVB
ylGHvndXcTmzZuHL7/8EidPnsSWLVukjKEwLRp07Bw4UIAgL+/P4YNGyZxKiIiopKHIyNFQKF
QYMyYMcoi8uOPP2LOnDmcdEdEVEB37txBhQoKjiYKHUUjZKRkQFHR0f8+++/UkgsXXLkSM
pkMq1evxldffSV1LCKSwODBgyGTCTyaCrqwsnJydMmTIFaWlpudY9Y9cOAAPDw8YGpqCiMjIzRt2hQBAQF5bvf
PP/9E69atYW5uDhMTEzRo0ADffPMN4uLiivrJ6Zh7NixMDU1lTpKkVmxYgUcHR1hYGAAV1dXnDt37r3PW
bZsGWWrVqgVDQ0M4ODhgwoQJ0f49Zf/i+98/tWvXVn5dT08PkyZNwtdff8Ok76nQRBkTHx8vAIdZ+4UyfbDw8N
FuXLlhLa2tggMDCyS1yAqaTIzQDFamqq1FFU5uvrKzp27Ciio6NFVFSU2L17tzAzMpk2pkzJsd5pPP/Okt
LS0xLRp08SNGzfEvXv3xI8//ij09PTE9vaSsZhaggFmtzsVKv5fNVp3Gqlwl0f5PP/Sm2ok8cQ1l1jRwCk7nU/0a+MR7
/9Zq1/ZfN2UBrN0rYvmqiRhBFbQmDHuNLOORKnmp6AOT80XtDgeQoPxRcQPGpgrIWRAKCJB/KV5NWkrbHiarpBRJ
YXrULQ3jYUIlDZGyhsYSYuQBrgJYbq3mIrNuVqegL2PFvCYekhLC3IqoQIWEJMCNjpahi1Ab8JKcRVvjEXMjJQQk
4J6XSc3lP4lsKcHabJlAtIEbh4AGd6oRFOVOIiByWQllcAlLjUiZlyokWHnEvFmvPYL2sBmUHBUMJA+r18iRC+SY
5DaE0oyWrTAeUS+n8xBsQtyhaOLiPPQCQ0Aj5l8Wsd6qUkajUsOG9Y4pOl+h5ZyTZnrszkWdzJLkvaU2hQvT+DQ
sREKfbRPUpM1mmTkIowTYxrK8wPOUkoVVwEMVqiQzKnRTVk+bUkd3Wb9Jd3vDvlnNUJ0rnWwZJ1A6ZnsQbzvSz87
g0Y6/f4+J9/Ye2Z48UixU2nK+oSjSWK6YLVakhOMAu5Rq+AUBwcE+6YmbaMEnnbKxmnhFT8+kgOzeX+MUtyrjvNi
Re/T3kIL4+WdqDz/2JlIuVCGq9mv6pTrU2v9ZtMibKS/YAqRZ1dcbuv8p6rY2gQJmoZUmJ2mJZS8pTyWEYbEUQEc
yNPE7IGQ0EI0gNwSqRY2UeYL5Hv8bQP/BrRXKJAWcqCz3EZJAxd0wxBu4vTu6MXjwZxuV++aNNcf9DUb7gvb+UOJ
wdmwzIiJSqiTbY5oWDgSSsuRCnTzHLHqJPdMcQPTzEHoBUzVnk5UH58JRlXCcMJA0jrB+evOeR6IR3GlsK8e2MGzj
Ezd8O4R0TJGj6ldPVVTiTb/gxXHi0NdNN0bFH/HAVpyTb2MF9B+mA11QGfe4VHJi6yEZ+kZj9FSCsmQIaXuqDWVa
NNA5OL1/+8LR+bHxyP7fBeH83ZNuQpM7hXQ1nNhAVs5Nic+xDOlOo+UjM8wnJPEM+/bxPbLNdbsbeR2B8jjFx+aA
8aVdNQ95yl1yPHlU+G9uUkRaQw2EBbPpBSnYKxKa+5Y0ns/SBg6nLpQ17HP/bJSuKyTrzGWcYfMvHhsozAO0AsLWl
yMZ5zDoIIHMMS9gOOB6o3nxkqG5ZGBjM6GqjL5kmv/iXHoVMStzFJLSkkP2Hm+YrOvRAQK4eeQzw6GsLy2uKo+TM2
5OpPIk8uxqqMf0FZAcz0cRdK1kl5yzZUbgSaW0Bo8U7gAPC/aX/g68BcEr9r/QMLq2kUjdUBafHdKsyF+w0W8Xe6
jkwQB+aKiOJxWCa/QGuFk7ogQTJPUgl93hQl9UYQnm0Ng7VK1h/9B6CxgbgrGCsVhDbGkk0/IKAG9bAvNECi8Ui4t

hW7duykst/Ff2aFS3bt1QpUoVrF27FhUrVoRCoUC9evVybfu/PuQwjZWVFbS1tRETE5NjeUxMDCpUqJDv9mbNm
gUfHx98/vnnAID69esjOTkZIOaMwIwZM/I8pGRhYYGaNWvi/v37OZbHxcXB2tr6nfmLEO/tLYTbt2+jVatWCA8
Ph5OTEOJDQ1lEiOi9tLSOMH36dMycORNpqakAgN69eONXVxeLFy/Otf6qVauQnJyMTz/9FMCbYfqkpCSsXLkyz
+2/fv06z+UNGjTA5cuX8z3119raGtHROTmWXb58uUDvqUWLFnBwcMC2bdsQGBiIvn37KotS3bp1oa+vj6ioKFS
vXj3HHwcHh3y32ahRI9y8eTPHsrCwMPTo0QOffffYZnJ2dcxy+epfGjRvjxoObcHROzJXB2NgYL1++xJO7dzBz5
ky0a9cOderUyVWE8jJy5Ehcvnz5nX/yOyP/yO0yjp6cHFxcXhISEKJcpFAqEhIQoD2flJSUlJVfhyL53TH6jZklJSQg
PD891GPD69evvHJOqdmqdDlsKfOjZNPfu3RPW1tYCgKhTp84Hn3pGRO9W2k/tffsslczMTGFvby8WLVqkXLZO6
VKhpaUlpk+fLm7duiXu378vFi9en0epvVOmTBHa2tpi8uTJ4u+//xaRkZEiODhY9OnTJ9+zbmjWFm5u
bOH36tAgPDxc7d+4Uf//9txBCiKCgIClCGICGTycTGjRvF3bt3xezZs4WZmVmus2nGjRuX5/ZnzJgh6tatK3ROdERoa
Giur5UvX14EBASI+/fviwsXLoiffvpJBAQE5Lvf9u3bJ2xsbERWVpZy2YQJE4SDg4MICwsTN2/eFJ9//rkwMzP
LsX/zyvjkyRNhbW0t+vTpI86dOyfu378vgoKCxODBgOVWVpaQy+WifPny4rPPPhP37tOTISEhomnTpgKA2L17d
74ZP9TWrVuFvr6+CAgIEDdv3hQjRowQFhYW4tmzZ8p1fHx8xNSpU5WP/f39hampqfjjjz/EgwcPxNGjROW1atV
ynNHO1VdfiRMnToiIiAgRFhYmPDO9hZWVlXj+/HmO169SpYrYtGlTntl4am8x+NAykp6eLjp37iwaN26c69xtI
lI/TSsjQggxf/58YW1tLZKSkpTL9u7dK9zc3ISxsbEwMDAQLi4uYv369Xlud9u2bcLd3V2YmpoKY2NjOaBBA/H
NN9+889TeyMhI0Obt3b2FmZiaMjIxEkyZNxNmzZ5Vfnz17trC1tRXm5uZiwoQJYsyYMQUuIzdv3hQARJUqVXKd+
qxQKMSyZctErVq1hK6urrC2thZeXl7i5MmT+WbFStWFEFBQcplL1++FD169BAmJibCxsZGzJw5UwwaNOi
9ZUQIIe7evSt69eolLCwshKGhoahdu7YYP368MuuxY8dEnTp1hL6+vmjQoIE4ceJEkZcRIYT4+eefXXlYWen
p5o1qyZ8lTr/74fX19f5ePMzEwxZ84cUa1aNWFgYCAcHBzE6NGjc/y9e3t7Czs706Gnpyfs7e2Ft7d3rtOU//7
7b2FhYSFSUlLyzCVFGZEJUcgZUaVUQkICzM3NsfPMHfT+uGahtpGamoqLKN
amRNNeKFSuwb98+HDkizSncmszb2xv0zs45zsL5r3d9z2V/hsbHx8PMzExtmThnpAD279+PiRMnKo/JGRoasog
QERWhL774Au7u7rw3jZplZGSgfv36mDBhgtRRcigd09El9Mcff8DHxwdyuRyNGzfGZ599JnUkIiKNp6OjgxkzZ
kgdQ+Po6elh5syZUsfIhSMj77B27Vo6MHDgQcrkcn332Gfr37y91JCIIo3DMpKPJUuWYMSIERBCYOTIkdi4cSM
vakZERFQEEbeIoTAnDlz8NVXwEAJk+ejJUrV+Z7fwIIKh5luKNJKZbW0ElVkUYyc6Pn+F7fjJ+xbrl+/ju+w7Am3saLFy4E
DKZTOJURGVX9gWOStKlq4k0WfaVZ7MvqFYceNzhLfXr18eGDRsQFxeHePGSR2HqMzT1taGhYUFnj9/DgAwMjL
iLwhERUShUODFixcwMjKCjo5qP0z44O2Q2yb3GGHvHhKJTJnEmUcVdeKBXqmpIzN16eQpxnAXe2/q5szUzgXXAcDAzpXnm3hjcsxvuzXCgTxqMNTjrrE
lsOVmLtKljWLuyZoyYXISdcfmWcFiqtVEUIRRI//ZyWZWaX6mYkHNzrzXtDQxIUozvLO9lEROVdAAeG
uKoVMTkREVHa9fPlSrXev1/izaaZNm26Dn6FCEVvUNEl
+xZU0RSxpicutDBiN4ub5IDbGgmlK6rtGC8QjjBIEm5KdbpvVKF5ZFSgGDn7lpi3sEOcU/vOjNGzK6urhl8hnoE6TRjyhtyzf9+YJKiMQEgy4oe6abqFg6m/5pD

LlzsWQui9T1GSr5FVjVKTY2FnK5XHn11my2tra4fft2ns959uxZnus/e/asyHJqksLs87d9/fXXqFixYq5/OJS
3wuzzO6dP47fffsPly5eLIaHmKcw+f/DgAf766y8MHDgQhw4dwv379zF69GhkZmbC39+/OGKXaoXZ5wMGDEBsb
CxatWoFIQSysrIwcuRITJ8+vTgilOn5fYYmJCQgNTUVhoaGBdqORo2MUOmzYMECbN26Fbt374aBgYHUcTRSYmI
ifHx8sHbtWlhZWUkdp8xQKBSwsbHBmjVr4OLiAm9vb8yYMQOrVq2SOprGOnHiBObNm4eVK1fi4sWL2LVrFw4eP
Ihvv/1W6mjOHho1MmJlZQVtbW3ExMTkWB4TE4MKFSrk+ZwKFSqotD7lVJh9nu3HH3/EggUsgULEBwcjAYNGhRlTI2
i6j4PDw9HZGQkunXrplymUCgAADo6Orhz5w6qVatWtKFLucL87q07ezs4Ouri6OtbWVy+rUqYNnz54hIyMDenp6R
Zq5tCvMPp81axZ8fHzw+eefAwDq16+P5ORkjBgxAjNmzFD7be8p/89QMzOzAo+KABo2MqKnpwcXFxeEhIQolyk
UCoSEhKB58+Z5Pqd58+Y51geAY8e05bs+5VSYfQ4AP/zwA7799lsEBQWhSZMmxRFVY6i6z2vXro1r167h8uXLy
j/du3dHmzZtcPnyZTg4OBRn/FKpMP/OW7Zsifv37yuLHwDcvXsXdnZ2LCIFUJh9npKSkqtwZJdBwduwFQm1fYa
qNre25Nu6davQ19cXAQEB4ubNm2LEiBHCwsJCPHv2TAghhI+Pj5g6dapy/bCwMKGjoyN+/PFHcevWLeHv789Te
1Wk6j5fsGCBONPTEzt37hTROdHKP4mJiVK9hVJH1X3+Np5NozpV93lUVJQwNTUVY8aMEXfu3BEHDhwQNjY24rv
vvpPqLZQ6qu5zf39/YWpqKv744w/x4MEDcfToUVGtWjXRr18/qd5CqZOYmCguXbokL126JACIJUuWiEuXLomHD
x8KIYSYOnWq8PHxUa6ffWrv5MmTxa1bt8SKFSt4am+n3/+WVSuXFno6emJZs2aiX/++Uf5NQ8PD+Hr65tj/e3
bt4uaNWsKPTO98dFHH4mDBw8Wc+LST5V9XqVKFQEg1x9/f//iD16Kqfrv/L9YRgpH1X3+999/C1dXV6Gvry+qV
q0qvv/+e5GVlVXMqUs3VfZ5ZmammDNnjqhWrZowMDAQDg4Ovo0eLVq1fFH7yUOn78eJ4/n7P3s6+vr/Dw8Mj
1nIYNGwo9PT1RtWpVsWHDBpVfVyYEx66IiIhOho1Z4SIiIhKH5YRIiIikhTLCBEREUmKZYSIiIgkxTJCREREk
mIZISIiIkmxjBAREZGkWEaIiIhIUiwjRBomICAAFhYWUscoNJlMhj179rzznCGDB6Nnz57FkoeIih7LCFEJNHj
wYMhksl/7t+/L3UOBAQEKPNoaWnB3sUOBAQEKPNoaWmhUqVKGDJkCJ4/f66W7UdHR6NTp04AgMjISMhkSMly+fDnHOsuXLOdAQIBaX
i8/c+bMUb5PbW1tODg4YMSIEYiLi1NpOyRO+nI3UAIspbx44dsWHDhhzLrK2tJUqTk5mZGe7cuQOFQoErV65
gyJAhePrOKY4cOfLB287v9vvD/ZW5u/sGvUxAfffQRgoODIZfLcWvWLQwdOhTh4e+DHBAZo+ZYbwb+nvB1X57cfdp6bsWqrLpwZd7mj2o+ldJbTr3IC5u5
ffsKNGzewceNG/PXXX5gyZUq+6w8cOBCVKlXC+fPnceHCBUydOjBmLt/oXw1Bj2Z/WS8ZRXbpfMTN+p8fuwwIrq2K/Gcdkr1/DzaxeJ/9DCz/dhdqeIZSH9Cuha3TKr+7+GnYPmdGQhSSVu3dk7ZwT+5OeFpzuXg
RpjxoxRKZOhoSEUCgWysrKwfPlyLF68GD/++COuXrOKLy8vdO/eHffu3QMA/PTTT9i9i3bx+2b9+O03fuIDAwEI6
OjnlU99y5cwCA4OBgREdHY9euXbnW6du3L16+8XFxSEoKAgDBw4EAISGhmLQoEEYN24cbt68idWrVyMgIADz5
yMgIADff/99gd9jZGQkjhw5Aj09PeUyhUKBSpUqYceOHbh58yZmz56N6dOnY/v27QCASZMmoV+/fujYsSOio
6MRHR2NFi1a5Np2eno6vDw8YGJigtOnTyMsLAwmJibo2LEjMjIykJWVhZ49e8LDwwNXr17F2bNnMWLECMhks
u7Ro4cYOnSo8vHq1atFxYoVhVhVwuF0II0a5dOzFv3rwc29i8eb0ws7PLM4MQQvj7+wstLS1hbGwsDAwMBAABQ
CQI0GjdxjMk0hPdSVzMAg3+3q6ekp/w4WLFiALl26YO7
kSb7PEUIIPz8/Obt373zZr92rVq1cuyD9PR0YWhoKI4cOfL07RNpIs4ZISqh2rRpg
ZBCCIIA9+dl8vfud8AoFatWti3bx/S0tLw+++/4/L
lyxg7dmyOdVasWIGIH169cjK1GDPjIQMOGDd+Z98qVK/SFr8aNGyMbPCOqYhH
3r16jmWRUZGomvXrhg1ahS+//v57lCtXDmfOnMGwYcOQkZEBIyOjAu/z9evXw87ODocPH8aUKVNQvXp1uOT
6JDhw4ftK8GDhyI4cOHY+XKldDX8XltgYCD69+Z98PLS0t5fsMCwvLtWtwvLMRIil8vfud8AoFatWti3bx/S0tLw+++/4/L
lyxg7dmyOdVasWIH169cjKioKqampyMjIQMGDd+Z98qVK7h//z5MTU1zLE9LS0N4eHg9gBR6cYyQlRCGRsbo3
3r16jmWRUZGomvXrhg1ahS+//571CtXDqdPn8awYcOQkZGR54fqnDlz4MDAABw/fhxLly7NtZ2kZGSkpKShYAgISEhPe+r8aNGyMiIgKHDx9Gx9GcHAw9vPT

```
srNCjRw+EhoYiIiICJO6cwJdffonHjx+rlIlIIOg9aYWIcstrOmO2JUuWCDs7O2FoaCi8vLzEpk2bBADx6tUrI
UTOCabp6emif//+wsHBQejp6YmKFSuKMWPG5Jiceu7cOdG+fXthYmIijI2NRYMGDXJNQP2vtyewvk0ul4s5c+Y
Ie3t7oaurK5ydncXhw4eVX1+zZo1o2LChMDY2FmZmZqJdu3bi4sWLyq/jPxNYhRBi7dq1wsHBQWhpaQkPD4989
49cLhd2dnYCgAgPD8+VKygoSLRoOUIYGhoKMzMzOaxZM7FmzZp834e/v79wdnbOtfyPP/4Q+vr6IioqSqSlpYn
BgwcLc3NzYWFhIUaNGiWmTp2a43nPnz9X7l8A4vjx4OIIIaKjo8WgQYOElZWVONfXF1WrVhXDhw8X8fHx+WYiO
lQyIYSQtg4RERFRWcbDNERERCQplhEiIiKSFMsIERERSYplhIiIiCTFMkJERESSYhkhIiIiSbGMEBERkaRYRoi
IiEhSLCNEREQkKZYRIiIikhTLCBEREUnq/wB1xx8wNNtuGwAAAABJRU5ErkJggg==\n"
          },
          "metadata": {}
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "Using optimal threshold and evaluating"
      ],
      "metadata": {
        "id": "ETelrzbu1hM6"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "from sklearn import metrics\n",
        "import numpy as np\n",
        "\n",
        "\n",
        "y_pred_proba = rf.predict_proba(X_test)[:, 1]\n",
        "\n",
        "fpr, tpr, thresholds = metrics.roc_curve(y_test, y_pred_prob)\n",
        "J = tpr - fpr\n",
        "optimal_idx = np.argmax(J)\n",
        "optimal_threshold = thresholds[optimal_idx]\n",
        "\n",
        "print(\"Optimal Threshold: \", optimal_threshold)"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "DR8YBSDM0J1Z",
        "outputId": "53fada55-7c04-4db8-c914-ddc0b5832e00"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
            "Optimal Threshold:  0.23\n"
          ]
        }
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "y_pred_proba[y_pred_proba>=optimal_threshold] = 1\n",
        "y_pred_proba[y_pred_proba<optimal_threshold] = 0"
      ],
      "metadata": {
        "id": "qsOnMqUdOTur"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "y_pred_proba"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "ypADub1_Opbj",
        "outputId": "e430bbd3-63be-4b54-864c-5e2a8565a7a7"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "array([0., 0., 0., 0., 0., 0., 1., 1., 0., 1., 1., 0., 0., 0., 1., 0.,
0.,\n",
              "       0., 0., 0., 1., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0.,
0.,\n",
              "       0., 0., 0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0.,
1.,\n",
              "       0., 0., 0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 1., 1.,
0.,\n",
              "       0., 0., 0., 0., 0., 1., 0., 1., 1., 0., 1., 0., 0., 0., 0., 0.,
1.,\n",
              "       0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
0.,\n",
              "       0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0.,
0.,\n",
              "       0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 0., 1., 1., 0., 0.,
1.,\n",
              "       1., 0., 0., 0., 0., 0., 0., 1., 1., 0., 1., 0., 0., 0., 0., 0.,
0.,\n",
              "       0., 1., 0., 1., 1., 0., 0., 0., 0., 0., 0., 0., 0., 1., 0., 0.,
1.,\n",
              "       0., 0., 0., 0., 0., 0., 0., 1., 0., 1., 0., 0., 0., 0., 0., 0.,
0.,\n",
              "       0., 0., 1., 1., 0., 1., 0., 1., 1., 0., 0., 0., 0., 0., 1., 0.,
0.,\n",
              "       0., 0., 1., 1., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 1.,
0.,\n",
              "       0., 0., 0., 1., 1., 0., 0., 0., 1., 0., 1., 0., 0., 0., 0., 0.,
0.,\n",
              "       0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0.,
0.,\n",
              "       0., 1., 0., 1., 0., 0., 0., 0., 0., 0., 1., 0., 0., 1., 0., 0.,
1.,\n",
              "       1., 0., 0., 0., 0., 1., 0., 1., 0., 0., 0., 1., 1., 1., 1., 1.,
1.,\n",
              "       0., 0., 0., 0., 1., 0., 0., 1., 0., 0., 1., 0., 1., 0., 0., 0.,
0.,\n",
              "       0., 0., 1., 1.])"
            ]
          },
          "metadata": {},
          "execution_count": 1349
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "Classification report with optimal threshold"
      ],
      "metadata": {
        "id": "dg9dES9mp6mH"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "from sklearn.metrics import classification_report, confusion_matrix\n",
        "\n",
        "# Print classification report\n",
        "print(\"Classification Report:\")\n",
        "print(classification_report(y_test, y_pred_proba))\n",
      "\n",
      "# Print confusion matrix\n",
      "print(\"Confusion Matrix:\")\n",
      "print(confusion_matrix(y_test, y_pred_proba))"
    ],
    "metadata": {
      "colab": {
        "base_uri": "https://localhost:8080/"
      },
      "id": "zLmxxhm71gHz",
      "outputId": "08a07919-3c0e-4240-ca9c-2a013b5f34c2"
    },
    "execution_count": null,
    "outputs": [
      {
        "output_type": "stream",
        "name": "stdout",
        "text": [
          "Classification Report:\n",
          "              precision    recall  f1-score   support\n",
          "\n",
          "         0.0       0.96      0.84      0.90       275\n",
          "         1.0       0.38      0.74      0.50        35\n",
          "\n",
          "    accuracy                           0.83       310\n",
          "   macro avg       0.67      0.79      0.70       310\n",
          "weighted avg       0.90      0.83      0.85       310\n",
          "\n",
          "Confusion Matrix:\n",
          "[[232  43]\n",
          " [  9  26]]\n"
        ]
      }
    ]
  },
  {
    "cell_type": "markdown",
    "metadata": {
      "id": "TP-5V9V79dyy"
    },
    "source": [
      "## XG Boost"
    ]
  },
  {
    "cell_type": "code",
    "execution_count": null,
    "metadata": {
      "id": "Lm72tF1b-b6n",
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 980
        },
        "outputId": "8519536c-1f53-4b2d-b7f1-26d363a62f67"
      },
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
            "Training Accuracy: 0.9919224555735057\n",
            "\n",
            "Classification_Report -->\n",
            "\n",
            "              precision    recall  f1-score   support\n",
            "\n",
            "         0.0       0.93      0.96      0.95       275\n",
            "         1.0       0.59      0.46      0.52        35\n",
            "\n",
            "    accuracy                           0.90       310\n",
            "   macro avg       0.76      0.71      0.73       310\n",
            "weighted avg       0.89      0.90      0.90       310\n",
            "\n",
            "Cross Validation Classification_Report -->\n",
            "\n",
            "              precision    recall  f1-score   support\n",
            "\n",
            "         0.0       0.92      0.96      0.94      1098\n",
            "         1.0       0.96      0.92      0.94      1123\n",
            "\n",
            "    accuracy                           0.94      2221\n",
            "   macro avg       0.94      0.94      0.94      2221\n",
            "weighted avg       0.94      0.94      0.94      2221\n",
            "\n"
          ]
        },
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 800x600 with 2 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAAAokAAAIjCAYAAABvUIGpAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAAA/6kl
EQVR4nO3deXgUVfr28bsDpBMgCxGyKYQdgqyixggEEGRVQBiRRQ3IIhhQCSBmzcMAQRRnEBBZVhk
22Iyi4iomFHElCQRAKEkNT7By/9szmgaUinE/r7mauui66qrnq6Nc7DfU6d2CzLsgQAAAD8gY+nCwAAAEDxQ5M
IAAAAAA0OiAAAADDSJAAAAMNAkAgAAwECTCAAAAANNIgAAAAw0iQAAADDQJAIAAMBAkwjgT/30009q166dgoKCZ
```

LPZtHDhwkK9/r59+2Sz2ZSSklKo1y3JWrVqpVatWnm6DABejiYRKAF2796thx9+WNWrV5efn58CAwPVrFkzvfT
SSzp9+rRb7x0fH6/t27fr2Wef1Zw5c3TzzTe79X5FqV+/frLZbAoMDLzk9/jTTz/JZrPJZrNp6tSpLl//8OHDG
j9+vLZu3VoI1QJA0Srt6QIA/LnPPvtM9957r+x2ux588EHVr19fZ8+e1bp16zR69Gjt2LFDb775p1vuffrOaaW
mpuof//iHhg0b5pZ7REVF6fTp0ypTpoxbrv9XSpcurVOnTmnRokXq2b58c+fKz89PZ86cuaJrHz58WBMmT
FDVqlXVuHHjAr9v6dKl3V3Q/AChMNIlAMbZ371716tVLUVFRWrlypSIiIhzHEhISlJaWps8++8xt9//lll8kScH
BwW67h81mk5+fn9uu/1fsdruaNWm//znP0aTOG/ePHXu3FkfffRRkdRy6tQplS1bV6+vkVyPwD4Mww3A8XY5
MmTdfLkSc2cOdOpQbygZs2aeuyxxxyvyxxyvz507pOmTJqlGjRqy2+2qWrWq/v73vysnJ8fpfVWrVtVdd92ldevW6dZ
bb5Wfn5+qV6+ud955x3HO+PHjFRUVJUkaPXq0FP//R+PHjZbPZnPYtW7ZMzZs3V3BwsMqXL69aTeq1aatKun8MO2FP//R+PHjZbPZnPYtW7ZMzZs3V3BwsMqXL
686dero73//u+P45eYkrly5Ui1atFC5cuUUHBysr127aufOnZe8X1pamvr166fg4GAFBQWpf//+OnXq1OW/2Iv
O6dNHX3zxhU6cOOHYt2HDBv3000/q06ePcf7x48c1atQoNWjQQQ0XLl1dgYKA6duyobdu20c5ZtWqVbrnlFklS/
/79HcPWFz5nq1atVL9+fW3atElxcXEqW7as43u5eE5iflHy8/Pz8jM/fvn17VahQQYcPHy7wZwWAgqJJBIqxRYs
WqXr16rr99tsLdP7AgQM1btw43XTTZo2bZpatmyp50Rk9erVyzg3LS1MfNf/vb33TnnXfa4ZSvXz2
LFDktS9e3dNmzZNktS7d2/NmTNHL774okv179ixQ3fddZdycpcnI0ceJEPf/88+rSpYv+97///en7li9frvbt2+v
oOaMaP368EhMTtX79ejVr1kz79u0zzu/Zs6d+//13JScnq2fPnkpJSdGECRMKXGf37t1ls9n08ccfO08ccf0/bNmzdPd
evW1U033WScv2fPHi1cuFB33XWXXnjhBY0ePVrbt29Xy5YtHQ1bdHS0Jk6cKEkaPHiw5syZzlz5iguLs5xnWP
HjqljX45q3LixXnzxRbVu3fqS9b3OOkuqVKmS4uPjlZeXJO164403tHTp388uKKjIws8GcFgAKzABRLmZmZl
iSra9euBTp/69atli/Rr4MCBTvtHjRpl8SbJWrlzp2BcVFWVJVJstasWePYd/ToUctut1tut1sjR4507Nu7d68lyZoyZYyr
TNePj462oqCijhqefftr6439Wpk2bZkmyfvnll8vWfEes2bNcuxr3LixFRoaah07du4IMPG
vd76KGHnK55zz33WNddd917/nHz1GuXDnLsizrb3/7m9WmTRvLsiwrLy/PCg8PtyZMmHDJ7+DMmTNWXl6e8Tn
sdrs1ceJEx74NGzYYn+2C0lbWpKsGTNmXPJYy5Ytnfb997//tSRZzzzjLVnzx6rfPnyVrdu3f7yMwLAlSJJB
IqprKwsSVJAQECBzv/8888lSYmJiU77n3jJiU77R44cKUnG3MV69eqpRYsWjteVKlSrpGtp1tGfPniuu+WIX5jJ+8sknys/
PL9B70tPTtXr108hISGO0/r7tXXrVvXr1NGTIEKfXLVqO0LFjxxzfYUH06dNHq1atUkZGhlZmr2pNqt/q1cDqpU4wx2/e+t5/1D1tr4evt7gqatBQNqbd26Vd0nT1doa
GiB3wsArqJJBIqpwMBARUZG6rvvvnPpfRc/OHI5pUqVuuR+y7X5/stdx4O/vrzVr1mj58uV64IEH9O233+q
+++7TnXfeaZx7Na7ms1xgt9X9vVvXt3zZ49WwsWLLhsiihJzz33nBITExUF6d3331X/TGT1m4mBk5zgfH2u9fT4HSvX
fPfjyu2bNmio0ePSpK2b9/u0nsBwFUOiUAxdtdd2n37t1KTU39f3jyWn/+vn/376yWn+5r8X7//mZiQ4wmjM9bQC9pk5o9KmjAkT8mmP5tLm4+HpbNTfBMTdfvKT3nBMP/EQeHsQbUPV4Zhp68B40GkLVMxcEfjyL/lkmX8ytf2bAyWPcQnWbWkGWNNSW11Ht46
+pDg2rHqGlpcoDcq+b6jMrp2X5vDpUbJbYq27U1pH7FyqGrrHsz+u4YEHZsRAfEaUQH5hkGX4tFnmiIRaAdlwxlyBmUhDwj8lrt+eahGHCqG4j/jVuFAFLZUgqyZSdtxYkwdKoLNN0EP7xzCzsHtoauGbW3Ks7PlPvn5Ec5hbkwTQTJ2Mz7GWyxKZl72gG78zdNbGfMkNEmrnzVNzvlRuBIpKpi/ft4XfavaRxfpIyXG6hnlNYyEWIyegaS6SNHGhGeuSNZ6o3lYXh0hWGKMaeDl3QaSQ+hy61Hn9NyO0ip9Nh3p8PJQmMKJ7QzePNyCWiVOhYSU02sP8hEBa2hPqpQpTZ8k5mLOM/1zNzRQAHKpSmGlv5IYq8d6Hrx6Gwd
lSvTq1UtjxozRPffco0cffVSnTp3S66+/rtq1a6szZs9uDFx4kStWbNGnTt3VlRUlI4ePar3XXnttb3o3b37Z6
O+ZMkUdO3ZUbGysBgwYoNOnT+vl19WUFCQxo8fX2if42I+Pj566qmnFFaeqkldM//uQwFBbZK3Ea7O3MnAK
7d66qV6/udF6NGjUUHBysGTNmKCAgQOXKlVNMTIyqvaVavmUl0rV67Ua6+9pqefftqxJM+sWbPPUqlUrjRO7VpMnT
3bpegBQIB5+uhpAAfz444/WoEGDDrKpVq1q+vr5WQECA1axMv+vllg+2zpw54zgvNzfXmjBhglWtWjWrTJkyVuX
Kla2kpCSncyzyr/BI4nTt3Nu5z8dIrlls1+sCx7Ia+nSpVbg9+vUtX19q06d0ta7775rLIGzYsUKq2vXrlZkZKT16+trRUZGWr179
+trRUZGWr1797Z+/PFH4x4x4XLx0zfPlyq1mzZpa/v78VGBho3X333db33dbt333/vdM6F+128xM6sWbMsSdbevXsv+51
alvMSOJdzuSVwro4caUERFj+/v5Ws2bNrNTU1EsuxfTkIj+Qp5AAAfDQAF/2ZSMtTSxhLAAAF6BOYkAAAAw0CQCAADAQJMIAAAAAOOiAAA
ADDSJAAAAMNAkAgAAwECTCAAAAAMM1+RtX/JsM83QJANzktw2veLoEAG7i58GuxJ29w+ktJf0/WySJAAAAMFyTS
SIAAIBLbORmF6NJBAAAsNk8XUGxQ9sMAAAAA0kiAAAAw80GvhEAAAAYSBIBAACYk2ggSQQAAICBJBEAAAIA5iQa
+EQAAABhIEgEAAJiTaKBJBAAAYLjZwDcCAAAAA0kiAAAAw80GkkQAAAAYSBIBAACYk2jgGwEAAICBJBEAAAIA5i
QaSRAAAABhIEgEAAJiTa0AbAQAAsNnct7kg0TlZt9xyiwICAhQaGqpu3bpp165dTTue0atVKNpvNaRsyZIjTOQc
OHFDnzp1VtmxZhYaGavo0Tp37pxLtZAkAgAAFB0rV69WQkKCbrnlFp07d06TJk1KCbrnlFp07d05///
wZp4sSJjtdly5Z1/DkvL0+dO3dWeHi4vv76a6Wnp+vBBx9UmTJl9Nxzz7ntMwMo+nFvU+d/NhcneNjGHZAh1G+xNvRR2lvA+KCNt

h2rRpk+Li4hz7y5Ytq/Dw8EteY+nSpfr++++1fPlyhYWFqXHjxpo0aZLGjBmj8ePHy9fXt0C1FI9vBAAA4BqVk
5OjrKwspy0nJ6dA783MzJQkhYSEOO2f03euKlasqPr16yspKUmnTp1yHEtNTVWDBg0UFhbm2Ne+fXtlZWVpx44
dBa6bJhEAAMDm47YtOTlZQUFBTltycvJflpSfn6/HH39czZo1U/369R37+/Tpo3fffVdffvmlkpKSNGfOHN1//
/2O4xkZGU4NoiTH64yMjAJ/JQw3AwAAuFFSUpISExOd9tnt9r98X0JCgr777jutW7f0af/gwYMdf27QoIEiIiL
UpkOb7d69WzVq1CicokWTCAAAIPm4bzFtu91eoKbwj4YNG6bFixdrzZo1uuGGG/703JiYGElSWlqaatSoofDwc
H3zzTdO5xw5xw5ckSSLjuP8VIYbgYAACgmLMvSsGHDtGDBAq1cuVLVqlX7y/ds3bpVkhQRESFJio2N1fbt23X06FH
HOcuWLVNgYKDq1atX4FpIEgEAAIrJO8OJCQmaN2+ePvnkEwUEBDjmEAYFBcnf31979+7d+/WvHnz1KlTJ1133XX69
ttvNWLECMXFxalhw4aSpHbt2qlevXp64IEHHNnyZGVkZ0ipp55SQkKCS4kmTSIAAEAx+d3Nr7/+uqTzC2b/0ax
Zs9SvXz/5+vpq+fLlevHFF5Wdna3KlSurR48euqpxzuXKKKT4+3mldxYKgSQQAA
CgmLMv60+OVK1fW6tWr//I6UVFR+vzzz6+qFppEAACAYjLcXJzwjQAAAMBAkggAAFBM5iQWJySJAAAAMJAkAgA
AMCfRwDcCAAAAA0kiAAAAcxINNIkAAAAMNxv4RgAAAGAgSQQAAGC42UCSCAAAAANJIgAAAHMSDXwjAAAAMJAkA
gAAMCfRQJIIAAAAA0kiAAAAcxINNIkAAAA0iQa+EQAAABhIEgEAAHhwxUCSCAAAAANJIgAAAHMSDXwjAAAAMJA
kAgAAMCfRQJIIAAAAA0kiAAAAcxINNIkAAAAMNxtomwEAAGAgSQQAAF7PRpJoIEkEAACAgSQRAAB4PZJEE0kiA
AAADCSJAAAABIkGkkQAAAAYSBIBAIDXY06iiSYRAAB4PZpEE8PNAAAAMJAkAgAAr0eSaCJJBAAAgIEkEQAAeD2
SRBNJIgAAAAwkiQAAAAASJBpJEAAAAGEgSAQCA12NOokkEQAAAAaSRAAA4PVIEk0OiQAAw0vRJJoYbgYAAICBJ
BEAAHg9kkQTSSIAAAAMJIkAAAAEiQaSRAAAABhIEgEAgNdjTqKJJBEAAAAGkkQAAOD1SBJNNIkAAMDr0SSaGG4
GAACAgSQRAACAINFAkggAAAADSSIAAPB6zEk0kSQCAADAQJIIAAC8HkmiiSQRAAAABpJEAADg9UgSTTSJAADA6
9EkmhhuBgAAgIEkEQAAgCDRQJIIAAAAA0kiAADwesxJNJEkAgAAwECSCAAAvB5JookkEQAAAAaSRAAAA4PVIEk0
0iQAAAPSIBoabAQAAYCBJBAAAXo/hZhNJIgAAAAwkiQAAw0uRJJpIEgEAAGCgSUSxM+qhdlr37mgdXTdV+1cka
/4Lg1QrKtQ4L6ZhNX3xxnD9uv55HVk7RctmPi4/exnjGPCPkJgTqD4YNPGDRr+yBC1bdV
cjW6so5UrljsdX75sqR4e9JDibo9Roxvr6IedOz1UKa5lNpvNbVtJRZOIYqfFTTU14/01avngVN019BWVLl1Ki
18fprJ+vo5zYhpW0yevPKIVX/2gFvdPUfP7p2jGe6uVn28Z13vu8a5K/yWzKD8CABecPn1KderUUdJTT1/2eJM
mN+nxxFFFXBng3ZiTiGKn67DXnF4PfvpdHVz5TzWpV1n/27xkjR5ZHe99t4zqTZ21zHHHeT/uPGtdq16ye2twWr
d6j31aH5je6t3AAV6R5i5Zq3qLlZY/f3aWbJOnnnw8VUUXwRiU58XMXjyaJv/76qyZPnqx77rlHsbGxio7N1T3
33KMpU6bZlmh4bol19+8WRpKEYCy/tJkn7LPCVJqlShvG5tWE2/HD+pL1MStW/5c1c1l9d4Wp7u/sbdRnUuq+ZTpCtD7Ta7L

t1xPC8vT507d9bZs2e1fv16zZ49WykRo3bpxLtXhsuHn48OG69957NWPGDCPitSxLQ4YMW0fDhw5Wamvqn10o
0TtaECROc9pUKu0VlIm4t9JpR9F5M6qkba0aoTf9pHmqBEvf6pHe
rdUQFk/Tfn3Uo/UDQAoWYrLcPOSJUucXqekpCgONFSbNm1SXFycmMjMzNXPmTM2bN
9pdtuu01Lly7V999/r+XLlyssLEyNGzfP16+vr6XurXBYOnitm3bNGLEiEv+Q7HZboxYoS2b
t36l9dJSkpSZmam01Y6rKkbKkZRmzbmXnVqUV/tBO3Xz0dPOPan/5IlSdq5J8Jo
yv35Rv294STs+PT8h/n9zn9BbEx8omg8AAIAuPeqZk5NToPdmZmZKkipWrChJ2rRpk86ePav4+HjHOXXq1FGVKlW0
ao4grXU1FQ1aNBAYWFhjn3t27fXb7/9pu+++67AdTNxBQAAwVYNXXFhjnPat2+fnx8vMLCwlq1atWgT6rx+PJLO3zfDLg7fGB9AHC+Ok+RBG
HTBtzr7rcOUgdHp6u/YePOR3bf/iZ/iYDh89vqpbc/Vtf11JhSjlwbPJ0KQR4ceFMGG1SxbZP0oeN+pLwMiJs0fUyO8PbBt1WkGh4XgCyM5vMq
fGvLCqaDwEAKDHcuQRqcnKygoKCnKfk5OS/fH9+fr4ee+wxNW/eXDfeeKMkKSMjQ76+vgoODnY6NywsTBkZGY5zLpzXtPY/e4g3Hvnqo0Obi8z7odVCGXGSFjVQaN00hKsi9VUpP2sf0RQLgfwZC6u5l2X/4Qu5p4fxmpxCvYzG6aYKsL8z6+ZBAuYWc5WBN4dX7NZPBhu+xkc/G9Jag
n4h7qwusL5xSEx4abR40apcGDB2vTpk1q06aN5uuvfMCMf4jGUxmhxFlE43gawhcDI/UTQ1EHBZXbX95PX8SgfcpzoDuCHhyiD6PVL25oS/Tq0+8YlvGcZMaT2Kgo01ByZv5Rj
dh1AZKkzJNndCYnV5I0bfZyPTks7GPLw7XPHnq/N/Y9hz8xSmVBOB5p7K
zdeDA/80d/vnQIf2wc6eCgoIUREmpzBMnlJ6elStNLoIHwKHf4erl6xFkYxFRb0J5NGSyI7Ei6+I2dt9YAQZMxvPLrCvE
+9LSEjQd999p3Xr1rmrtD/lsSYxVkkhhOYiLBl7JTFmVm2UE5EsdSWAtxEFB26bTsbWeNmpAvxo+Tp+S9iUTqSrbrUYY6h0hDGErAFB2
L3n7caf+gcXP07qKvJUmvzFslP3sZTR7RRRWCyun8Ro9GVWmL77jPmUMCeJgwBR6B6YzN3thCoy2Q5Y3WhqPMMkL/gpIDOMu0EM7c9OqKcXh4dM
50e+6dWfblS455KchwfM2qEJGnII8MONGF40RaLa5Y7pyTa7fYCNYVJfRtk9+d+xA
P6sSJE05p4pEjRxQeHu4455tvvnG63oXZzxfOKQiPNbv6TU83yq5JkrbSTufSEMkFp

```
H379nrttf/7bWWlSpXS4sWLNXToUMXGxqpcuXKKj4/XxIkTXaqlWAw3FzaGm4FrF8PNwLXLk8PNtZ9Y8tcnXaE
fJ3dw27XdiSQRAAB4veIy3FycePR3NwMAAKB4IkkEAABejyDRRJIIAAAAA0kiAADwej4+RIkXI0kEAACAgSQRA
AB4PeYkmmgSAQCA12MJHBPDzQAAADCQJAIAAK9HkGgiSQQAAICBJBEAAHg95iSaSBIBAABgIEkEAABejyTRRJI
IAAAAA0kiAADwegSJJppEAADg9RhuNjHcDAAAAANJIgAA8HoEiSaSRAAAABhIEgEAgNdjTqKJJBEAAAAGkkQAA
OD1CBJNJIkAAAAwkCQCAACvx5xEE0kiAAAADCSJAADA6xEkmmgSAQCA12O42cRwMwAAAAwkiQAAwOsRJJpIEgE
AAGAgSQQAAF6POYkmkkQAAAAYSBIBAIDXI0g0kSQCAADAQJIIAAC8HnMSTTSJAADA69EjmhhuBgAAgIEkEQAAe
D2GmO0kiQAAADCQJAIAAK9HkmgiSQQAAICBJBEAAHg9gkQTSSIAAAAMJIkAAMDrMSfRRJMIAAC8Hj2iieFmAAA
AGEgSAQCA12O42USSCAAAAANJIgAA8HoEiSaSRAAAABhIEgEAgNfzIUo0kCQCAADAQJIIAAC8HkGiiSYRAAB4P
ZbAMTHcDAAAAANJIgAA8Ho+BIkGkkQAAAAYSBIBAIDXY06iiSQRAAAABpJEAADg9QgSTSSJAAAAMJAkAgAAr2c
TUeLFaBIBAIDXYwkcE8PNAAAAMJAkAgAAr8cSOCaSRAAAABhIEgEAgNcjSDSRJAIAAMBAkggAALyeD1GigSQRA
AAAhkJpEk+cOFEYlwEAAPAIm819W0nlcpP4r3/9S++//77jdc+ePXXdddfp+uuv17Zt2wq1OAAAgKJgs9nctpV
ULjeJM2bMUOXKlSVJy5Yt07Jly/TFF1+oY8eOGj16dKEXCCkRkZjiZx8eLF6tmzp9q1a6eqVasqJ
iamOAsEAABwtxIc+LmNy0lihQoVdPDgQUnSkiVL1LZtWOmSZVnKy8sr3OoAAADgES4nid27d1efPn1Uq1YtHTt
2TB07dpQkbdmyRTVr1izO0AgEAANyNJXBMLjeJO6ZNU9WqVXXw4EFNnjxZ5cuXlySlp6frkUceKfQCQCAQAAUPRcb
hLLlCmjUaNGGftHjBhRKAUBAAAUNXJEU4GaxE8//IIXLKfM//cSpcsVFwMAAIDioUBNYrdu3WOVUEi5ZkASMv3QpOMZvNxsMrAACgxCl
O6xmuWbNGU6ZMOaZNm5Senq4FCxY49WL9+vXT7T7Nmznd7Tvn17LVmyxPH6+PHjGj58uBeBYtWiQfHx/16NFDL73O
mOaYEEU6Onm/Pz8AmOOiAAAoCTysblvc1V2drYbaNWqkV1999bLndLnd0bLnd0jQQenp6Y7tP//5j9P7sMNZp6
WL16sNWvWaPDgwS7V4fKcxD86c+aM+aM/Pz8ruYSAAAA+IOOHTs6ZQ5OGbo5HL5vdrvDw8EseZ7lzp5YsWaINGzbo5ptvl
iS9/PLL6tSpk6Onar1yMgCle1HyOol5eXmaNGmrr/+epUvX1579Wr9+verUqePYV6ZdzaqLtvbUvmSHAQbvxSQ6Dq
dq1TkPYc+fOVd26ddWmTRt16tRJzZs315tvvvulSHTbLsizXyy/e/JsM83QJANzktw2veLoEAG7id1WrN1+d/u9
td9ulZ/Vq4LZru9MV/+PYuHGjdu7cKen8PMWmTZsUlEAADwLJebxEOHDql379763//+pqAQJFoRYHAABQFHyKYKmaksblJL3X78WLVqoZs2ai3mayioK0ty5c9W1W3b1931AkAAO2B
Ikml5PE48ePq3r16pLOzz88fvy4+Jas2ZN4VYHAAAAj3C5Saxevbbr27t0rSapbt67mz58v6XCGBwcXKj
FAQAAFAWbzea2raRyuUns37+/tm3bJkl66ir9Zj3+/Pz+Tm3bJkl68skn9eqrr8rPz08jRozoQ6NGjC71AAAAAFD2bZVnW1Vxg//792rRpk
2rWrKmGDRsWVl1XJetMvqdLA0AmPIEIXLvK2z338z18wU63Xfvle6Lddm13uqp1EiUpKipKUVFRhVELAAAAAiok
CNYnTp08v8AUfffTRyTKy4GAADAE0ry3EF3KVCTOHLkyAJdzGaz0SQCAAAAAAQA0gVOgSQCAIASx4ce0VCgJvHC04wAAAAKm0y+KDl6OvWIO/ZGL1GizB9UjpgNVohC
2TB07dpQkbdmyRTVr1izO0AgEAANyNJXBMLjeJO6ZNU9WqVXXw4EFNnjxZ5cuXlySlp6frkUceKfQCQCAQAAUPRcb
hLLlCmjUaNGGftHjBhRKAUBAAAUNXJEU4GaxE8//bTAF+zSpcsVFwMAAIDioUBNYrdu3WOVUEi5ZkASMv3QpOMZvNxsMrAACgxCl
O6xmuWbNGU6ZMOaZNm5Senq4FCxY49WL9+vXT7T7Nmznd7Tvn17LVmyxPH6+PHjGj58uBYtWiQfHx/16NFDL73O
m0aYEEU6Onm/Pz8Am00iAAAoCTysblvc1V2drYaNWqkV1999bLndLnd0jQQenp6Y7tP//5j9P7sMNZp6
bJk
```

aNGigChUquKsmAAAAeJhLD66UKlVK7dq104kTJ9xUDgAAQNHzsblvK6lcfrq5fv362rNnjztqAQAAQDHhcpP4z
DPPaNSoUVq8eLHS09OVlZXltAEAAJQ0Njf+r6Qq8JzEiRMnauTIkerUqZMkqUuXLk6/ns+yLNlsNuXl5RV+lQA
AAG5UkoeF3aXATeKECRM0ZMgQffnll+6sBwAAAMVAgZtEy7IkSS1btnRbMQAAAJ5AkmhyaU6ijUWEAAAAvIJL6
yTWrl37LxvF48ePX1VBAAAARY0gz0RSkzhhwgTjN64AAADg2uNSk9irVy+Fhoa6qxYAAACPYE6iqcBzEolhAQA
AvIfLTzcDAABca8jCTAVuEvPz891ZBwAAgMf40CUaXP61fAAAALj2ufTgCgAAwLWIB1dMJIkAAAAwkCQCAACvx
5REE0kiAAAADCSJAADA6/mIKPFiJIkAAAAwkCQCAACvx5xEE00iAADweiyBY2K4GQAAAAaSRAAA4PX4tXwmkkQ
AAAAYSBIBAIDXI0g0kSQCAADAQJIIAAC8HnMSTSSJAAAAMJAkAgAAr0eQaKJJBAAAXo+hVRPfCQAAAAwkiQAAw
0vZGG82kCQCAADAQJIIAAC8HjmiiSQRAAAABpJEAADg9VhM20SSCAAAUIysWbNGd999tyIjI2Wz2bRw4UKn45Z
lady4cYqIiJC/v7/atm2rn376yemc48ePq2/fvgoMDFRwcLAGDBigkydPulQHTSIAAPB6NjdursrOzlajRo306
quvXvL45MmTNX36dM2YMUNff/21ypUrp/bt2+vMmTO0c/r27asdO3Zo2bJlWrx4sdasWaPBgwe7VIfNsizrCuo
v1rLO5Hu6BABuwpAQc00qb/fcz/e8zYfcdu0+N91wxe+12WxasGCBunXrJul8ihgZGamRI0dq1KhRkqTMzEyFh
YUpJSVFvXr10s6d01WvXj1t2LBBN998syRpyZIl6tSpkw4d0qTIyMgC3ZskEQAAwI1ycnKUlZXlt0Xk5FzRtfb
u3auMjAy1bdvWsS8oKEgxMTFKTU2VJKWmpio40NjRIEpS27Zt5ePjo6+//rrA96JJBAAAXs9ms7ltS050VlBQk
N0WnJx8RXVmZGRIksLCwpz2h4WF0Y5lZGQoNDTU6XjpOqUVEhLiOKcgeLoZAADAjZKSkpSYm0i0z263e6iagN
JBAAAXs+dQ6t2u73QmsLw8HBJ0pEjRxQREeHYf+TIETVu3NhxztGjR53ed+7cOR0/ftzx/oJguBkAAKCEqFatm
sLDw7VixQrHvqysLH399deKjY2VJMXGxurEiRPatGmT45yVK1cqcqPz9fMTExBb4XSSIAAPB6tmKOcsLJkyeVlpb
meL13715t3bpVISEhqlKlih5//HE988wzqlWrlqpVq6axY8cqMjLS8QR0dHS0OnTooEGDBmnGjBnKzc3VsGHD1
KtXrwI/2SzRJAIAABQrGzduVOvWrR2vL8xnjI+PV0pKip544gllZ2dr80DB0nHihJo3b64lS5bIz8/P8Z65c+d
q2LBhat0mjXx8fNSjRw9Nnz7dpTpYJxFAicI6icC1y5PrJH6w9bDbrn1v44Knd8UJcxIBAABgYLgZAAB4veI0J
7G4oEkEAABej6FVE98JAAAADCSJAADA6zHcbCJJBAAAgIEkEQAAeD1yRBNJIgAAAAwkiQAAw0sxJdFEkggAAAA
DSSIAAPB6PsxKNNAkAgAAr8dws4nhZgAAABhIEgEAgNezMdxsIEkEAACAgSQRAAB4PeYkmkgSAQAAYCBJBAAAX
o8lcEwkiQAAADCQJAIAAK/HnEQTTSIAAPB6NIkmhpsBAABgIEkEAABej8W0TSSJAAAAMJAkAgAAr+dDkGggSQQ
AAICBJBEAAHg95iSaSBIBAABgIEkEAABej3USTTSJAADA6zHcbGK4GQAAAAaSRAAA4PVYAsdEkggAAAADSSIAA
PB6zEk0kSQCAADAQJ0IEmHzpg0aMXyo0raN0y2NorVq5XKn48e0/arxY5PUsW2cmsc00fChg3Rg/z7PFAvAJZs
3btDjw4aofZsWatqwrr6860dbkvbu2a0Rw4cq7vab1ezWJnqg99+Unn7YA9XiWmWzuW8rqWgSUSKcPn1atevU0
RNJY41jlmVp90PDdPjQQU198VW9+/7HioiIVMLDD+n0qVMeqBaAK87/fNfVmL+Pu+TxgwcPaEB8H1WtVl1vznx
H7330iQY0fkR2X3sRVwp4F+YkokRo1jx0zZrHXfLYgf37tP3bbXrvo09Vo2YtSdKTTz2tDne00H+XfKZu3e8ty
lIBuKhZizg1a3Hpn29Jeu3lF9WsRUs9ljjasa9y5SpFURq8SAk0/NyGJBElXm5uriTJbv+/VMHHx0dlfH21dct
mT5UFoBDk5+dr3ZpVqhJVVQlDBqhty9v1YJ+elxySBq6Gj83mtq2kKtZN4sGDB/XQQw/96Tk50TnKyspy2nJyc
oqoQhQHVatWU3hEhF6dPk1ZWZnKzT2r2f9+S0ePZ0jYL794ujwAV+H48WM6deqUUma+pdubtdCrb8xU6zZtNXr
EcG3a+I2nyw0uacW6STx+/Lhmz579p+ckJycrKCjIaXthyj+LqEIUB6XLlNHkF17W/v371KbFbWoRc5M2bvhGt
zdvIZtPsf5XHMBfsPLzJUktW9+hvg/0U5260eo/YLBaxLXSR/Pf83B1uJbY3LiVVB6dk/jpp5/+6fE9e/b85TW
SkpKUmJjotC/HKnNVdaHkia53o+bNX6CTv/+u3NxcVQgJUb++9yn6xhs9XRqAqxBcoYJKlS6t6jVqOu2vVr2Gt
m7Z5KGqA0/g0SaxW7dustlssizrsufY/mIs3263081Fk6SsM/mFUh9KnvIBAZL0P8yy8/vvNCThUQ9XB0BqlCn
jqxtvrK/9+/Y67d+/f5/CIyI9VBWuSSU58nMTj47FRURE600PP1Z+fv4lt82beegA5506la1dP+zUrh92SpI0/
3xIu37YqYz/v07a8qVLtGnDNzp06KBWf7lCw4YMUMvWbXTb7c08WTaAArjcz/eFdRAf6DdAS5d8oY8/nK+DB/b
r/f+8q7Wrv9S99/XxZNnANc9m/VmM52ZdunRR48aNNXHixEse37Ztm5o0aaL8fNeSQZLEa8+mDd9oyMB4Y3/nL
t00flKy3ps7R3Nm/1vHjx1TxUoV1emurhr48FCVKePrgWrhTiX5SUFc2sYNX+vhAebP911dumnCM+fnmH+y4CP
Nmvmmjh7JUFTVanr4keFq1bpNUZcKNytv99zP99e7M9127ZgaQW67tjt5tElcu3atsr0z1aFDh0sez8701saNG
9WyZUuXrkuTCFy7aBKBaxdNYvHi0SbRXWgSgWsXTSJw7fJkk/jNHvc1ibdWL5lNIr9xBQAAeD3++mliETkAAAA
YSBIBAACIEg0kiQAAADCQJAIAAK9nI0o0kCQCAADAQJIIAAC8HqtrmUgSAQAAYCBJBAAAXo8g0USTCAAAQJdoY
LgZAAAABpJEAADg9VgCx0SSCAAAAANJIgAA8HosgWMiSQQAAICBJBEAAHg9gkQTSSIAAAAMJIkAAABEiQaaRAA
A4PVYAsfEcDMAAAAMJIkAAMDrsQS0iSQRAAAABpJEAADg9QgSTSSJAAAAMJAkAgAAECUaSBIBAABgIEkEAABej
3USTSSJAAAAMJAkAgAAr8c6iSaaRAAA4PXoEU0MNwMAAMBAkggAAECUaCBJBAAAgIEkEQAAeD2WwDGRJAIAABQ
T48ePl81mc9rq1q3r0H7mzBklJCTouuuuU/ny5dWjRw8d0XLELbXQJAIAAK9ns7lvc9WNN96o9PR0x7Zu3TrHs
REjRmjRokX64IMPtHr1ah0+fFjdu3cvxG/i/zDcDAAAUIyULl1a4eHhxv7MzEzNnDlT8+bN0x133CFJmjVrlqK
jo/XVV1/ptttuK9Q6SBIBAIDXs7lxy8nJUVZWltOWk5Nz2Vp++uknRUZGqnr16urbt680HDggSdq0aZNyc3PVt
m1bx7l169ZVlSpVlJqaWnhfxv9HkwgAAODGLjE5OVlBQUFOW3Jy8iXLiImJUUpKipYsWaLX39de/fuVYsWLfT
7778rIyNDvr6+Cg40dnpPWFiYMjIyCvXrkBhuBgAAcKukpCQljJiY67bPb7Zc8t2PHjo4/N2zYUDExMYqKitL8+

```
fPl7+/v1jovRpMIAAC8njuXwLHb7ZdtCv9KcHCwateurbSONN155506e/asTpw44ZQmHjly5JJzGK8Ww80AAAD
F1MmTJ7V7925FRESoadOmKlOmjFasWOE4vmvXLh04cECxsbGFfm+SRAAA4PWuZKkadxg1apTuvvtuRUVF6fDhw
3r66adVqlQp9e7dWOFBQRowYIASExMVEhKiwMBADR8+XLGxsYX+ZLNEkwgAAFBsHDpOSL1799axY8dUqVIlNW/
eXF999ZUqVaokSZo2bZp8fHzUoOcP5eTkqH379nrttdfcUovNsizLLVf2oKwz+Z4uAYCb+BSXv+4DKHTl7Z77+
d599LTbrlOjtGgfOCkszEkEAACAgeFmAAAABikMNIkAAMDruXMJnJKK4WYAAAAYSBIBAIDX45k4EOkiAAAADCS
JAADA6xEkmkgSAQAAYCBJBAAAIEoOkCQCAADAQJIIAAC8HuskmmgSAQCA12MJHBPDzQAAADCQJAIAAK9HkGgiS
QQAAICBJBEAAHg95iSaSBIBAABgIEkEAABgVqKBJBEAAAAGkkQAAOD1mJNookkEAABejx7RxHAzAAAADCSJAAD
A6zHcbCJJBAAAgIEkEQAAeDObsxINJIkAAAAwkCQCAAAQJBpIEgEAABgIEkEAAGAgSQQQAAF6PINFEkwgAALweS+CYGG4GA
ACAgSQRAAB4PZbAMZEkAwECSCAAAQJBoIEkEAACAgSQRAAB4PYJEEOkiAAAADCSJAADA67FOookmEQAAeD2
WwDEx3AwAAAADSSIAAPB6DDebSBIBAABgoEkEAACAgSYRAAAAAuYkAgAAr8ecRBNJIgAAAAwkiQAAwOuxTqKJJ
hEAAHg9hptNDDcDAADAQJIIAAC8HkGiiSQRAAAABpJEAAAAokQDSSIAAAAMJIkAAMDrsQSOiSQRAAAAABpJEAAD
g9VgnOUSSCAAAAANJIgAA8HoEiSaaRAAAALpEA8PNAAAAMJAkAgAAr8cSOCaSRAAAABhIEgEAgNdjCRwTSSIAA
AAMNsuyLE8XAVypnJwcJScnKykpSXa73dPlAChE/P4GrhfEG75jAfVY5t98JbXCIVmCqGVKgeSuB2MAlT7BUBEA
AABhoEgEAAGCgSQQQAAICBJhElmt1u19NPP82kbqAE4vc3cL1g4woAAAAMJMkAAAAw0CQCAADAQJMIAAAAA0ALZ
AAAA0OiSrRXX31VVatWZ+fn2JiYvTNN994uiQA12j16tXq3LmzoqKiZLPZtGjRIk+XBAD/iyYRJdZ7772nxMRE
Pj3lZWVpU2NjZWfXW2fnP3KCzFyKmXSbJzQNCxGMqSvnz2ze8ny63Bv6vcQubPgg9K2nvgNTJmrLhm4s3PlY3Jq
op59+Wps3b1ajRo3Uvn17ZtDZa8m2t0Uu6joAUJRoEoFrVL9+ffJxSmsSJJKqLNnzxYa+oODbY1P+Ivo/j7j6DDeGN
+RUoaJM0aGh6W6Z80ta4IgAM7KuV0H6dRKO+BwZvz73ox9JNo+/x+bWFi2LjZLBsz5O0VQ9P4Nd6LovBVL5SoS9p
ZKiQRnzPqwLkrzaz6B5y2P/PgqKsP/8/BZnlqqpXbFwF+NZHEtv1/a+8rrzJhGRWqbArk3x0dlzC1XYxX9e2pJ
9WxcfEC9LHWSkiEb/eDfHx0+1hEXZfgxZ3bZvMz9t4EbyEEEDexTvfpjrc4/fGpwYPKlQjAFBLc5OMYb8BTd
HoWQ6j/D5oBjJWFnSCoHZ+BwfuHqvrLWN5HE12FO5kLLNdB49W3kkgnyaTXq0HuvlF+lpEXDIx4j7YuNzdtsp
mrDnpVwuX78BqL4cktVdY2XSrlHd5gUABmKNQNmcPv+B7CF46TqjBLr66eABkrq62j/G0/pwPBJmVZa6BqagBqdj
ALDZvqzn9ayk2dLJ/nV0TWYAcOQA58pxP4gBRU3BPnuIX4EMUJP2cBYpOwiGjyYqEI8qXafvvp1ZkEZuEamVy+
Sf8/pplTAXrJj9kNnPt2BgAEOm1NTDmxqF7rmNbu1ZoIqLN1I+1mrOgCAqCZBmRO9Otm0DttH+K+Mt2AZ/NWN
y4ttKYsvIJsLLmAhNbdlddkZ6XEsoOtn1BQCxsCNeE9vHVkTkKOVhDwQHZ/NWWiN9tz/RKH25nnkTyu2jMTc3Ty
XcUg9Ec+Dv66uFr2A+wYa/9SEu9Hw2VPiyRjOJXvw2nLw7pVxRGb3fmDJ2XDaI+OAt5xyaZSZ7m7cs0GNK0TH
Gk7VOlwcMmTsb5aLafgP6swHQ/2ywddRTc7YF4ktOs2rX28eJWm67HOv5UwyB3OFHWBMF1s1V9CqEP2Ic/L8KR/fW
0+qHp8jCS8OUEnnhJ1K1WEFtarpmqYMfS1MrrxTgNZ2ZdupuFqxhsQgGOZ97AKDhr9d5Vu0wXSnXHqL3Z+rr5
m3CLxqEf1qFqfLOIjnA/oqLx1YDtTrNMZMBDdFD/oyeqn36Th5Xgle0Zm91vVkx/Ne/uL1AsU0svdkHnZ/rrSl
Tc4gCqhHSt9x7pX7zPT7XlGQj9Z1ibIsbTlD4wUrZ8Xl/eE4cLs96RPcpmvJBr7DMmO4HfzvczlImzuqNDF4L
I3nDqqjX/ZH5iMm1jFhT8FzTp+DnIDyHmUT06O1Dyh4VNs8z6gW7tPoRkiWBh+Y2PnNVLMqGdWXfgU9QaHVsT
Ojf6/QXbA/dUW7WA+DRIfvt7L5rQ/vj/X5h5mnZt5bqo2yvNk3V1VpqJFpd3PoN8/Vq0n+WyBQ6cWN+ftq/AN
3qBeZ+bNP/g18kfeaEf/hX5Bxq2WzfXM3f1U5eTa9SeqvFH/9rqx6mvv0qlb2pmbL5q+G2HZe8jUAAL8AUCnN
V7gAAM+79gP4nS1+2f0KnsNUz7xPp0cW7LvT9w1P9fI9frp16lX0YwaT0RKw3LPfFgW2IuG3ddu0oTWFyqXr
Bz6l9V43YZbAjb2A+b0DlWnLu9x/3fTvmq5T7R/AUf/d9X2cL1k9eyE2Nq4fAp1CFFhC8ZYHl8/E/xKUGIp
55sd1Zqn1vS8UMmkJ1XKZVb9G/2XtrbrfZ5Zrrwnsbff9bbbz/bezCDz5bdtn3NKwx/uzfTr2s1LvvJU
0CQm6c7c2ZNNmn7ut4Ybrv/gYvNl5Lv/z9/Sk7lqKo9CgFsZ6+wrL6+jNKFSXQSUiaWZsTG4wG7+G1eZ2YnjO
vNLUzLzW5P7nnb5Rx4pPh6HMD1Ry8X5ujwnTrmlyO4f6ZbPwNDFFc2rMmMfMi1tnzZ2L0pn0LRwSzKoT
wMOzf2OkDbRVWR0L6qIO0MgwdOXJwHf0+Vl1Lqm62fhbXRYfuEq+Fpa0rWfBIPrgDa6gP0VcBd3NSwa4AVTaCnA
dY63WXaZMFjoQRdZwaS+NI5XLW+oL0RlQJN3cuB0Fx0M58v4Bx8zGQW1r71yP5aK3jGQ5dZx+8K5dX1ZxbR9+
TyuVIzyBTg2KXJrq3EdvnDz6ZIg0E8o8kXD9dwpaeeLe9LEIzGJj2AeEyT81XTUV2lo7bmlkZVDVVdRt3Jj+t
+0qUSz5VJX7VmSkbk8rbbvWJhs9R4bGMH+N6Xos4r5zGyBr/+DcWKqbx1/WM1ru+fOTVdb+WVpBR3DzT6
```

```
44ZL3HTdunD755BOlpaVpx44dWrx4saKjo12qHQDcjSYRgEPZsmW1Zs0aValSRd27d1d0dLQGDBigM2fOOJLFk
SNH6oEHHlB8fLxiY2MVEBCge+6550+v+/rrr+tvf/ubHnnkEdWtW1eDBg1Sdna2JOn666/XhAkT9OSTTyosLEz
Dhg2TJE2aNEljx45VcnKyoq0j1aFDB3322WeqVq2apPPzBD/66CMtXLhQjRo1OowZM/Tcc8+59Hm7dOmiESNGa
NiwYWrcuLHWr1+vsWPHGufVrFlT3bt3V6d0ndSuXTs1bNjQaYmbgQMH6u2339asWbPUoEEDtWzZUikpKY5aL+b
r66ukpCQ1bNhQcXFxKlWqlN577z2XagcAd7NZ15ttDgAAAK9FkggAAAADTSIAAAAMNIkAAAAwOCQCAADAQJMIA
AAAAOOiAAAADDSJAAAAMNAkAgAAwECTCAAAAANNIgAAAAw0iQAAADD8P6cfeI31cWCfAAAAAElFTkSuQmCC\n"
          },
          "metadata": {}
        }
      ],
      "source": [
        "from xgboost import XGBClassifier\n",
        "\n",
        "XGB = XGBClassifier()\n",
        "model_train_eval(XGB,\"XG Boost\")"
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "# Evaluation"
      ],
      "metadata": {
        "id": "xvoPV75eymT-"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "Creating a dataframe to summarize the performance across all models"
      ],
      "metadata": {
        "id": "Feo4ucrBO5y6"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "evaluation_df = pd.DataFrame(eval_dict)\n",
        "evaluation_df.iloc[:,1:] = evaluation_df.iloc[:,1:].apply(lambda x:
round(x,2))"
      ],
      "metadata": {
        "id": "Oo44Cu7IS76y"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "# summary of all models\n",
        "evaluation_df"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 175
        },
        "id": "ncn-p2NyUg8X",
        "outputId": "6041da86-ae7d-4056-8f22-921be6bff354"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "                    Model   AUC  Recall  Precision  Accuracy\n",
              "0      LogisticRegression  0.65    0.69       0.19      0.63\n",
              "1       Gradient Boosting  0.58    0.20       0.37      0.87\n",
              "2  RandomForestClassifier  0.75    0.51       0.78      0.93\n",
              "3                XG Boost  0.71    0.46       0.59      0.90"
            ],
            "text/html": [
              "\n",
              "  <div id=\"df-9399ef6c-029b-44b6-8daa-c51f3510830c\" class=\"colab-df-
container\">\n",
              "    <div>\n",
              "<style scoped>\n",
              "    .dataframe tbody tr th:only-of-type {\n",
              "        vertical-align: middle;\n",
              "    }\n",
              "\n",
              "    .dataframe tbody tr th {\n",
              "        vertical-align: top;\n",
              "    }\n",
              "\n",
              "    .dataframe thead th {\n",
              "        text-align: right;\n",
              "    }\n",
              "</style>\n",
              "<table border=\"1\" class=\"dataframe\">\n",
              "  <thead>\n",
              "    <tr style=\"text-align: right;\">\n",
              "      <th></th>\n",
              "      <th>Model</th>\n",
              "      <th>AUC</th>\n",
              "      <th>Recall</th>\n",
              "      <th>Precision</th>\n",
              "      <th>Accuracy</th>\n",
              "    </tr>\n",
              "  </thead>\n",
              "  <tbody>\n",
              "    <tr>\n",
              "      <th>0</th>\n",
              "      <td>LogisticRegression</td>\n",
              "      <td>0.65</td>\n",
              "      <td>0.69</td>\n",
              "      <td>0.19</td>\n",
              "      <td>0.63</td>\n",
              "    </tr>\n",
              "    <tr>\n",
              "      <th>1</th>\n",
              "      <td>Gradient Boosting</td>\n",
              "      <td>0.58</td>\n",
              "      <td>0.20</td>\n",
              "      <td>0.37</td>\n",
              "      <td>0.87</td>\n",
              "    </tr>\n",
              "    <tr>\n",
              "      <th>2</th>\n",
              "      <td>RandomForestClassifier</td>\n",
              "      <td>0.75</td>\n",
              "      <td>0.51</td>\n",
              "      <td>0.78</td>\n",
              "      <td>0.93</td>\n",
              "    </tr>\n",
              "    <tr>\n",
              "      <th>3</th>\n",
              "      <td>XG Boost</td>\n",
              "      <td>0.71</td>\n",
              "      <td>0.46</td>\n",
              "      <td>0.59</td>\n",
              "      <td>0.90</td>\n",
              "    </tr>\n",
              "  </tbody>\n",
              "</table>\n",
              "</div>\n",
              "    <div class=\"colab-df-buttons\">\n",
              "\n",
              "  <div class=\"colab-df-container\">\n",
              "    <button class=\"colab-df-convert\"
onclick=\"convertToInteractive('df-9399ef6c-029b-44b6-8daa-c51f3510830c')\"\n",
              "            title=\"Convert this dataframe to an interactive
table.\"\n",
              "            style=\"display:none;\">\n",
            "\n",
            "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\" viewBox=\"0
-960 960 960\">\n",
            "    <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220
220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-
160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",
            "  </svg>\n",
            "    </button>\n",
            "\n",
            "  <style>\n",
            "    .colab-df-container {\n",
            "      display:flex;\n",
            "      gap: 12px;\n",
            "    }\n",
            "\n",
            "    .colab-df-convert {\n",
            "      background-color: #E8F0FE;\n",
            "      border: none;\n",
            "      border-radius: 50%;\n",
            "      cursor: pointer;\n",
            "      display: none;\n",
            "      fill: #1967D2;\n",
            "      height: 32px;\n",
            "      padding: 0 0 0 0;\n",
            "      width: 32px;\n",
            "    }\n",
            "\n",
            "    .colab-df-convert:hover {\n",
            "      background-color: #E2EBFA;\n",
            "      box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n",
            "      fill: #174EA6;\n",
            "    }\n",
            "\n",
            "    .colab-df-buttons div {\n",
            "      margin-bottom: 4px;\n",
            "    }\n",
            "\n",
            "    [theme=dark] .colab-df-convert {\n",
            "      background-color: #3B4455;\n",
            "      fill: #D2E3FC;\n",
            "    }\n",
            "\n",
            "    [theme=dark] .colab-df-convert:hover {\n",
            "      background-color: #434B5C;\n",
            "      box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",
            "      filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",
            "      fill: #FFFFFF;\n",
            "    }\n",
              ″  </style>\n″,
              ″\n″,
              ″    <script>\n″,
              ″      const buttonEl =\n″,
              ″        document.querySelector('#df-9399ef6c-029b-44b6-8daa-
c51f3510830c button.colab-df-convert');\n″,
              ″      buttonEl.style.display =\n″,
              ″        google.colab.kernel.accessAllowed ? 'block' : 'none';\n″,
              ″\n″,
              ″      async function convertToInteractive(key) {\n″,
              ″        const element = document.querySelector('#df-9399ef6c-029b-44b6-
8daa-c51f3510830c');\n″,
              ″        const dataTable =\n″,
              ″          await
google.colab.kernel.invokeFunction('convertToInteractive',\n″,
              ″                                                    [key], {});\n″,
              ″        if (!dataTable) return;\n″,
              ″\n″,
              ″        const docLinkHtml = 'Like what you see? Visit the ' +\n″,
              ″          '<a target=\″_blank\″
href=https://colab.research.google.com/notebooks/data_table.ipynb>data table
notebook</a>'\n″,
              ″          + ' to learn more about interactive tables.';\n″,
              ″        element.innerHTML = '';\n″,
              ″        dataTable['output_type'] = 'display_data';\n″,
              ″        await google.colab.output.renderOutput(dataTable, element);\n″,
              ″        const docLink = document.createElement('div');\n″,
              ″        docLink.innerHTML = docLinkHtml;\n″,
              ″        element.appendChild(docLink);\n″,
              ″      }\n″,
              ″    </script>\n″,
              ″  </div>\n″,
              ″\n″,
              ″\n″,
              ″<div id=\″df-d3251cf5-f067-41aa-9881-cbd54c059d2e\″>\n″,
              ″  <button class=\″colab-df-quickchart\″ onclick=\″quickchart('df-
d3251cf5-f067-41aa-9881-cbd54c059d2e')\″\n″,
              ″            title=\″Suggest charts\″\n″,
              ″            style=\″display:none;\″>\n″,
              ″\n″,
              ″<svg xmlns=\″http://www.w3.org/2000/svg\″ height=\″24px\″viewBox=\″0 0
24 24\″\n″,
              ″     width=\″24px\″>\n″,
              ″    <g>\n″,
              ″        <path d=\″M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-
.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\″/>\n″,
              ″    </g>\n″,
              ″</svg>\n″,
              ″  </button>\n″,
              "\n",
              "<style>\n",
              "  .colab-df-quickchart {\n",
              "      --bg-color: #E8F0FE;\n",
              "      --fill-color: #1967D2;\n",
              "      --hover-bg-color: #E2EBFA;\n",
              "      --hover-fill-color: #174EA6;\n",
              "      --disabled-fill-color: #AAA;\n",
              "      --disabled-bg-color: #DDD;\n",
              "  }\n",
              "\n",
              "  [theme=dark] .colab-df-quickchart {\n",
              "      --bg-color: #3B4455;\n",
              "      --fill-color: #D2E3FC;\n",
              "      --hover-bg-color: #434B5C;\n",
              "      --hover-fill-color: #FFFFFF;\n",
              "      --disabled-bg-color: #3B4455;\n",
              "      --disabled-fill-color: #666;\n",
              "  }\n",
              "\n",
              "  .colab-df-quickchart {\n",
              "    background-color: var(--bg-color);\n",
              "    border: none;\n",
              "    border-radius: 50%;\n",
              "    cursor: pointer;\n",
              "    display: none;\n",
              "    fill: var(--fill-color);\n",
              "    height: 32px;\n",
              "    padding: 0;\n",
              "    width: 32px;\n",
              "  }\n",
              "\n",
              "  .colab-df-quickchart:hover {\n",
              "    background-color: var(--hover-bg-color);\n",
              "    box-shadow: 0 1px 2px rgba(60, 64, 67, 0.3), 0 1px 3px 1px rgba(60,
64, 67, 0.15);\n",
              "    fill: var(--button-hover-fill-color);\n",
              "  }\n",
              "\n",
              "  .colab-df-quickchart-complete:disabled,\n",
              "  .colab-df-quickchart-complete:disabled:hover {\n",
              "    background-color: var(--disabled-bg-color);\n",
              "    fill: var(--disabled-fill-color);\n",
              "    box-shadow: none;\n",
              "  }\n",
              "\n",
              "  .colab-df-spinner {\n",
              "    border: 2px solid var(--fill-color);\n",
              "    border-color: transparent;\n",
              ″    border-bottom-color: var(--fill-color);\n″,
              ″    animation:\n″,
              ″      spin 1s steps(1) infinite;\n″,
              ″  }\n″,
              ″\n″,
              ″  @keyframes spin {\n″,
              ″    0% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-bottom-color: var(--fill-color);\n″,
              ″      border-left-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    20% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-left-color: var(--fill-color);\n″,
              ″      border-top-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    30% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-left-color: var(--fill-color);\n″,
              ″      border-top-color: var(--fill-color);\n″,
              ″      border-right-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    40% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-right-color: var(--fill-color);\n″,
              ″      border-top-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    60% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-right-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    80% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-right-color: var(--fill-color);\n″,
              ″      border-bottom-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    90% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-bottom-color: var(--fill-color);\n″,
              ″    }\n″,
              ″  }\n″,
              ″</style>\n″,
              ″\n″,
              ″  <script>\n″,
              ″    async function quickchart(key) {\n″,
              ″      const quickchartButtonEl =\n″,
              ″        document.querySelector(’#’ + key + ’ button’);\n″,
              ″      quickchartButtonEl.disabled = true;  // To prevent multiple
clicks.\n″,
             ″      quickchartButtonEl.classList.add(’colab-df-spinner’);\n″,
             ″      try {\n″,
             ″        const charts = await google.colab.kernel.invokeFunction(\n″,
             ″            ’suggestCharts’, [key], {});\n″,
             ″      } catch (error) {\n″,
             ″        console.error(’Error during call to suggestCharts:’, error);\n″,
             ″      }\n″,
             ″      quickchartButtonEl.classList.remove(’colab-df-spinner’);\n″,
             ″      quickchartButtonEl.classList.add(’colab-df-quickchart-
complete’);\n″,
             ″    }\n″,
             ″    (() => {\n″,
             ″      let quickchartButtonEl =\n″,
             ″        document.querySelector(’#df-d3251cf5-f067-41aa-9881-
cbd54c059d2e button’);\n″,
             ″      quickchartButtonEl.style.display =\n″,
             ″        google.colab.kernel.accessAllowed ? 'block’ : 'none’;\n″,
             ″    })();\n″,
             ″  </script>\n″,
             ″</div>\n″,
             ″\n″,
             ″  <div id=\″id_8c17cedb-798b-49f2-8df7-99134484426e\″>\n″,
             ″    <style>\n″,
             ″      .colab-df-generate {\n″,
             ″        background-color: #E8F0FE;\n″,
             ″        border: none;\n″,
             ″        border-radius: 50%;\n″,
             ″        cursor: pointer;\n″,
             ″        display: none;\n″,
             ″        fill: #1967D2;\n″,
             ″        height: 32px;\n″,
             ″        padding: 0 0 0 0;\n″,
             ″        width: 32px;\n″,
             ″      }\n″,
             ″\n″,
             ″      .colab-df-generate:hover {\n″,
             ″        background-color: #E2EBFA;\n″,
             ″        box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n″,
             ″        fill: #174EA6;\n″,
             ″      }\n″,
             ″\n″,
             ″      [theme=dark] .colab-df-generate {\n″,
             ″        background-color: #3B4455;\n″,
             ″        fill: #D2E3FC;\n″,
             ″      }\n″,
             ″\n″,
             ″      [theme=dark] .colab-df-generate:hover {\n″,
             ″        background-color: #434B5C;\n″,
              ″        box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n″,
              ″        filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n″,
              ″        fill: #FFFFFF;\n″,
              ″      }\n″,
              ″    </style>\n″,
              ″    <button class=\″colab-df-generate\″
onclick=\″generateWithVariable(’evaluation_df’)\″\n″,
              ″            title=\″Generate code using this dataframe.\″\n″,
              ″            style=\″display:none;\″>\n″,
              ″\n″,
              ″  <svg xmlns=\″http://www.w3.org/2000/svg\″ height=\″24px\″viewBox=\″0
0 24 24\″\n″,
              ″       width=\″24px\″>\n″,
              ″    <path
d=\″M7,19H8.4L18.45,9,17,7.55,7,17.6ZM5,21V16.75L18.45,3.32a2,2,0,0,1,2.83,0l1.4,1.43a
1.91,1.91,0,0,1,.58,1.4,1.91,1.91,0,0,1-.58,1.4L9.25,21ZM18.45,9,17,7.55Zm-
12,3A5.31,5.31,0,0,0,4.9,8.1,5.31,5.31,0,0,0,1,6.5,5.31,5.31,0,0,0,4.9,4.9,5.31,5.31,0,
0,0,6.5,1,5.31,5.31,0,0,0,8.1,4.9,5.31,5.31,0,0,0,12,6.5,5.46,5.46,0,0,0,6.5,12Z\″/>\n
″,
              ″  </svg>\n″,
              ″    </button>\n″,
              ″    <script>\n″,
              ″      (() => {\n″,
              ″      const buttonEl =\n″,
              ″        document.querySelector(’#id_8c17cedb-798b-49f2-8df7-
99134484426e button.colab-df-generate’);\n″,
              ″      buttonEl.style.display =\n″,
              ″        google.colab.kernel.accessAllowed ? ’block’ : ’none’;\n″,
              ″\n″,
              ″      buttonEl.onclick = () => {\n″,
              ″        google.colab.notebook.generateWithVariable(’evaluation_df’);\n″,
              ″      }\n″,
              ″      })();\n″,
              ″    </script>\n″,
              ″  </div>\n″,
              ″\n″,
              ″    </div>\n″,
              ″  </div>\n″
            ]
          },
          ″metadata″: {},
          ″execution_count″: 1353
        }
      ]
    },
    {
      ″cell_type″: ″code″,
      ″source″: [
        ″fig, ax = plt.subplots(1, 4, figsize=(15, 5))\n″,
        "colors = [\"blue\", \"green\", \"red\", \"orange\"]\n",
        "for i in range(4):\n",
        "  ax[i].bar(evaluation_df[\"Model\"], evaluation_df[evaluation_df.columns[i +
1]], color=colors)\n",
        "  ax[i].set_xlabel(\"Model\")\n",
        "  ax[i].tick_params(axis=\"x\", rotation=90)\n",
        "  ax[i].set_ylabel(evaluation_df.columns[i + 1])\n",
        "  ax[i].set_title(evaluation_df.columns[i + 1])\n",
        "plt.tight_layout()\n",
        "\n",
        "\n",
        "# Show the plot\n",
        "plt.show()"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 482
        },
        "id": "lDM-rVSPsE4z",
        "outputId": "026cafe4-ecfe-4863-d470-4bbdc59bba20"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 1500x500 with 4 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAABdIAAAHqCAYAAAAAkLx0AAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlc
nNpb24zLjcuMSwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy/bCgiHAAAACXBIWXMAAA9hAAAPYQGoP6dpAACnc0l
EQVR4nOzde1xUdf7H8feAXEQFMQQUSfKSxnrBIIlM7YKy5lp2MVILxaSyKFeyTbIku4iWWbqZlZsuZ
ahblkWreWsV76IlCJmgWJAwvz/60dvEMAwKc7i8no/HeTyc73zPmc+ZB72ZPpz5HpbDYLAAAAAAAAAADY5GJ
0AQAAAAAAAAAA1Gc00gEAAAAAAAAAsINGOgAAAAAAAAAAdtBIBwAAAAAAAAAADtBIBwAAAAAAAAAADDhrpAAAAAAAAAADYQSMdAAAAA
AAAAAA7aKQDAAAAAAAAAGAHjXQAAAAAAAAAAOygkQ4AAAAAAAAAgB000gEAAOqIyWTSk08+aXm8ePFimUwHHTp
0yLCaAKApGTt2rEJCQs2z4z4YNG2QymbRhw4Y6qQkAAAAAAAAAA5l++WWZTCZEuMymUyaPXu2zXnAw
55dZbPk/fff15l5Ahq+Tn5yd3d3e1b99et99+u2SqKcLZqjo/fffAAKRJf8x
mT09PXXrppUpMTFR+fr7R5QNBIH1jrmR2eXwRoapoZXQBny1dulQhISHKzs7sWvn371KVLlws6rhhx47R48WL16
dNHSU1lJCgwM1T/T+++uuvKGrrrqqls4ogAAAAAAAAGpGp556ewdy/++/pdccnf03e4Q3iQs7xQpnUVu/
+uLVwAf+b//u2aXxAD2bxxlx6+nkMB0UlaonbSFUNpcJ6Do0U48eZkO0+LlaANKqJ/uZ
lccOsAAKT+/ftLkvbv328zxerdnWl5WKLqRbYz30zpSmVVBVxBqXDeR/oxeOXHSrcxUEvw9WiqLSqZbO0mqZz4
Q5U4eDBg/ryyy81Z84tW3btOWFT3h9hqy3OfUYCXGyH99YxE05t/n/+7vsx
/Pl3dtesdPy3/o/9yDDXdlb6ahvxTH8XiM0Ao6MjK6M5rAbbnqsAX3/N7yGFNI1HgXfncnwZUwjMEVcyN02
5+7MfTfcyOXVzJm9yhxs8vd+/CzE+kUzM1rcLFJaAHjAcV6uXOsDH9W96Kw+/dhs/Q5wXvkQFk=
Ky8sv5OODAKdzpC9S3WdnSfrll1/05JNP6tJLL5Wnp6fatWunW265xfJZvarcPbek7uLFiy1j9n5/fP755xoxY
oQuvvhieXh4KDg4WJMmTdLPP/9cqe7du3fr9ttvV9u2bdW8eXN169ZNU6dOlSStX79eJpNJ77//fqX93n77bZl
MJm3atKnG7ycaPhrpQBWWLl2qW265Re7u7ho5cqT27t2rb7755ryPt3HjRp04cUKjRo2Sq6trLVYKAPijc/en8
PX1lSR99913uvLKK7Vr1y5NmTJFL7zwglq0aKHhw4dbfUA+ffq0+vfvr5deekmDBw/W3Llzdd9992r37n06evS
oJKm4uFivvfaarrnmGs2aNUtPPPGvmkCgoKFBMTo61btzr7VAGgwTjXMLnoooskSWfPnlVMTIz8/f01e/Zs3Xrrr
ZKke++9V4888oj69eunuXPnKj4+XkuXLlVMTIx+/fVXy/EWL16soUOH6sSJEOpOTtbMmTMVFhamzMzMKmtYu3a
tRo4cKV9fX82aNUszZ87UNddcoy+++MJu7YUNddcoy+++MJu7YsXL9btt98uV1dXpaamKiEhQStXrtTVV1+tkydPWs0tLy9XTEyML
rroIs2ePVsDBw7UCy+8oFdfffV83jYAMEx1fRFHPjuXl5frL3/5i6ZPn67w8HC98MILmjhxooqKirRz587zqu
q3x/vvvuuzpw5owkTJuill15STEyMXnrpJcXFxVntv337dkVGRmrdunVKSEjQ3LlzNXz4cP3zn/+UJF1zzTUKD
g62+YeDpUuXqnPnzoqKijqv2tHAmQFU8p///////Mcsybx27Vqz2Ww2V1RUmDtO6GCeOHGiZc7BgwfNkszPP/+8zWM
8//zzZknmgwcPms1ms3nu3LlmSeb333+/jqsHgKbj9ddfNOsyf/rpp+aCggLzkSNHzCtWrDC3bdvW7OHhYT5y5
IjZbDabr7/+enPPnj3Nv/zyi2XfiooK81VXXWu2rWrZWzatGlmSeaVK1dWeq2Kigqz2Ww2nz171lxaWmr13E8
//WQOCAgwjxs3zmpckjlJaVSved+NwBAY2Qrm5ctW2a+6KKLzM2bNzcfPXrUPGbMGLMk85QpU6z2/fzzz82Sz
EuXLrUaz8zMtBo/efKkuVWrVubIyEjzzzz//bDX3XF6bzWbzmDFjzB07drQ8njxotnb29t89uzZZKutfv369WZJ
5/fr1ZrPZbC4rKzP7+/ube/ToYfVaH330kOkVmSedq0aVavJ8n81FNPWR2zT58+5vDwcDvvGgDUL47ORRz57JyRk
WGWZJ4zZO6Vc/6Yu+ec67u8/vrrlrGqfn+YzWbzmTNnKo2lpqaaTSaT+fDhw5axAQMGmFu1amc19vt6zOMUn
52ezh4WE+efKkZez48ePmZs2aWX2+R9PCFemADUuXLlVAQICuvfZaSZLJZFJsbKAVaSUl1atWq1
uoEAPwmOJpabdu2VXBwsG677Ta1aNFCq1atUocOHXTixXAmtW7dOt99L99+u99LvX/pbu3AHeA9NNt9xxxxxxxq2bK13n//f
QUFBVnmTJgwwWqfd999Vz4+Pho0aoaJAlrwsLCxUeHq6WLVtaltalFau3atTp06pSlTplS6lt6Pad26tSa4sS2TItW7as0vNLly7V4M
DLVej2LpphSSdOXNGkizzunfvLknasWOHhg8fXgeVAOgDT1bdvXOVEREiShg8frquvvlqjROI6OKioqJEmTJ
O+2auD8Xk0+DL/11lsa03ashg8frkceeUT+/v6WNXN/f3NTAGjqfp/Ntnh4eMjFxfraroqKCvn7+9tcl1ZSpQZ
1Tfn7+2vr1q36+OOP9a9//Uv/+te/9/2QADR0jvZFaktVV6ZXtSKArd8f5eXlGjRok
E6cOKFH31U3bt3V4sWLFfT9999r7Nixqlv8nqIm4uDhNnDhRR48eVW1pqb766ivNmzevsdB40EjHfiDpUuXyt/
fX/Pnz6/O3MqVK/X+X3+rPT1dbdu2l2ZeXl3JycmweJycnR5Ls85Mkx85MkXX311fL19V19dU777yP19dU777yjxx57jA/YAFBHz
jW1r732Ws2bNN0/jxo2TJLm5uSk6Otruvp07d672pkcrVqxQp06dtHL1SqsP/eeubgcAnL/ONvr008/Vb9+/ay
+om9rniTt3LmzxlcGuru7a9iwYRo2bJgqKip0//33a8Nvvn++uuu87quZYcHMvzANBYO
NoXceSzc+f0nfX111/r119/lZubm805vr6+klTp5s1//MaPPt27NCePXu0ZMkSq5uHrUzumL/nEpr06dOkmSQzc6veO
OO5SU1KR33nlHP//8s9zc3BQbG+twX8hfddttllbbExESdOnVKq1atkqqurqwYPH
qx//vOfys3NtTp0bm6u/vnPf2rw4MGWhRz7LxPTcdLFMzMzNmfYEuZGnl74fXrbntKBrcrwKvOpd+
+fZWW1iZvb29dc801euWVVV3Ts2LFKcwsKCKiz/vvXWW7Vt2za9//77lady+1zu+77HP/666j2j4NAGhyb
r/9dpWXl+vpp5+u5Zs2ctTZbBgwerVatWWk1N1S+//7hWG5Lpz9bn7HN+/PFHq8cueLi7q1auXJKmkpM
391d6errVnH/961atWuXC5AUBDUJO+iCOfnW+99VYTFhbavJL73JyOHvK1dVVn332VkFGpl/PH3x9+fn4TSzRYY
/qMbjabNXYztPD1e7AlmOkRq75Kq9Ob00OBbzLSQnWqnpBvZzmu3pqlLfVGSTfVFF/fuaMDwy5d6VgfUhmRbX/8
pxhtvtPn8lVdeqbZt22rp0qWKjY3VjBkzdOWVV+ryyy/XoccTW/zkLvTWXzwtXSrHHb/zcdVZrXqTXsb/aVsYtjXOPVRVoUXCEXyvhntb/r7uE/
04vvPCClq9fr9tuu02BgYHKy8vTBx355Zf00RqI0RjRgXQosXL9b8+fN19dXV2fPnkpISFCM0BgJFUZWc6pfPhMhsMAUAaPQeeeQRjRgxQosXL9b8+fN19dXV2fPnkpISFC
nTp2Un5+vTZs26ejRo9q2bZtlnxUrVm2jlypW6+eabNXToU3bu3/vKXv2jlypW6+eabNXToU
B08eFDp6ekKDQ3V6dOnDT5rAGjYBg4cqHvvvvVepqanaunWrBg8eLDc3N+3du1fvvvvuu5s6dq9tuu03e3t568cU
XNX78eF1xxRUaNWqUfH19tW3bNp05c0bLliyRUaNWqUfH19tW3bNp05c0bLliyRUaNWqUfH19tW3bNp05c0bLliyRUaNWqUfH19tW3bNp05c0bLliyRUaNWqUfH19tW3bNp05c0bLliyRUaNWqUfH19tW3bNp05c0bLliyRUaNWqUfH19tW3bNp05c0bLliyRUaNWqUfH19tW3bNp05c0bLliyRUaNWqUfH19tW3bNp05c0bLliyRUaNWqUfH19tW3bNp05c0bLliy
```

```
/H3v/9dV199taWXc8kll+jQoUNavXq1tm7dajU3Li5Ot912myTZ/CMvmhgzAIthw4aZPT09zSUlJVXOGTt2rNn
Nzc1cWFhoNpvN5l27dpljY2PN/v7+5mbNmpn9/f3Nd9xxh3nXrl1VHmPFihXmwYMHm9u0aWNu1qyZuV27dubY2
Fjzhg0bav2cAKAxe/31182SzN98802l58rLy82d03c2d+7c2Xz27Fnz/v37zXFxcebAwECzm5ub0SgoyPyXv/z
FvGLFCqv9fvzxR3NiYqI5KCjI7O7ubu7QoYN5zJgxltyvqKgwz5gxw9yxY0ezh4eHuU+fPuaPPvrIPGbMGHPHj
h2tjiXJnJKSUqnegwcP1vZbAQD1hr1sPmfMmDHmFi1aVPn8q6++ag4PDzc3b97c3KpVK3PPnj3Nf/vb38w//PC
D1bxVq1aZr7rqKnPz5s3N3t7e5r59+5rfeecdq9f5fTaf+xzu7+9vdnd3N1988cXme++3zs2DHLnPXr15s1m
devX2/1WsuXLzf36dPH7OHhYW7Tpo1590jR5qNHjzp0XikpKWb+9xtAQ1DTvkh1n53NZrP5zJkz5qlTp5ovueQ
Ss5ubmzkwMNB82223mffv32+ZU1BQYL711lvNX15eZ19fX/O9995r3rlzp1mS+fXXX7fMs/f747/f7///a85Ojra3
LJlS7Ofn585ISHBvG3btkrHMJvN5p07d5pvvvlmc+vWrc2enp7mbt26mZ944lKxywtLTX7+vqafXx8zD///LO
D7yIaK5PZbOd7bwAAAAAAAADQBJ09e1bt27fXsGHDtGjRIqPLgcFYIx0AAAAAAAAA/uCDDz5QQUGB1Q1M0XRxR
ToAAAAAAAAA/L+vv/5a27dv19NPPy0/Pz9t2bLF6JJQD3BF0gAAAAAAAAD8vwULFmjChAny9/fXG2+8YXQ5qCe
4Ih0AAAAAAAAAADu4Ih0AAAAAAAAADtopAMAAAAAAAAAAYEczowtwtoqKCv3www9q1aqVTCaT0eUAQI2YzWad0
nVK7du314tL0/5bKHkOoKEiy/+HLAfQkJHnvyHLATRkNcnyJtdI/+GHHxoXahyHM
ADR1ZTpYDaByaep6T5QAaA0eyvMk1Olu1aiXptzfH29vb4GoAoGaKi4sVHBxsybKmjDwHC0FCR5f9DlgNoyMjz3
5DlABqymmR5k2ukn/uakbe3Nw/PoMHiK5PkOYCGrz5m+fz58/X8888888rLy9PvXv3lksvvaS+fftWOT8tLUOLFix
Qbm6u/Pz8dNttyk1NVWenp4OvR5ZDqAxqI957kxkOYDGwJEsb3KNdAAAAAAAgIYqIyNDycnJWlhYYW
omJ0fHjx23Of/vttzVlyhSlpKRo165dWrRokZYvX67dHHnvMyZUDAAAAdY9GOgCgVsyfP18hISHy9PRUZGSksr0
zq5x7zTXXyGQQyVdqGDh3qxIoBAL83Z84cJSQkKD4+XqGhupPT5eX15cyMjJszv/yyy/jRo1Si8e
LBGjhxpN/8BAACAhopGOgDgtXOKsaVK1fq2LFjlm3nzzp1ydXXViBEjnFw5AECSysrKtHncnzZkVHR1vGXFxcFBO
drU2bNtnc56qrrtLmzZstjfMDBw5ozo1uuGGG6p8ndLSUhUXF1ttAAAAaNxiczMFqehVjmzZtFBgYaNnWr
l0rLy8vGukAUAjIKV5crICDAajwgIEB5eXk29xuKlapSeeuopXX311XJ2Jzc1Pnzp11zTXX2F3ajTUPQBIS5e
t0Di4Vs8DAAAAqCs0OgEAF+R8rmL8oOWLFumOO+5QixqbtGpqpMAEAt27Bhg2bMmKGXX35ZW7ZsVs0cqVK7V69Wo9/
fTTVe6nnJysoqIiy3b48GEnVAsAAAAA0OegTAw0EfuAvfPzs7jOuSevbwqZKSEt1zzz2aOnVqaX8d+uPkmq4O3Pk0e
UHv8/Pzk6uqq/Px8q/H8/HwFBgba3OeJJ57QPffco/Hu4fm8bhl/T4OegTAwFCLFi1Sz5491bdvX7vwWA4AAOqOu7u7wsPDlZmZaR2rZNnqNgoMjJGI/Uo1LaKRWDfPSYpLAHFMiTF1bxYwGVrErJ/nt9yIiE2GDNUhlQo9LmZsFrnGo7tNoSd1i4rRnr
zq5x7zTXXyGQQyVdqGDh3qxIoBAL83Z84cJSQkKD4+XqGhoUpPT5eXl5cyMjJszv/yyy/Vr18/jRo1SiEhIRo8e
LBGjhxpN/8BAACAhopGOgDggtXOKsaVK1fq2LFjlm3nzp1ydXXViBEjnFw5AECSysrKtHnzZkVHR1vGXFxcFBO
drU2bNtnc56qrrtLmzZstjfMDBw5ozZo1uuGGG6p8ndLSUhUXF1ttAAAAQENAIx0AcMFqehVjmzZtFBgYaNnWr
l0rLy8vGukAYJDCwkKVl5crICDAajwgIEB5eXk29xuk1apSeeuopXX311XJ2Jzc1Pnzp11zTXX2F3aJTU1VT4+PpY
tODi4Vs8DAAAAqCs0OgEAF+R8rmL8oOWLFumOO+5QixYt6qpMAEAt27Bhg2bMmKGXX35ZW7ZsVs0cqVK7V69Wo9/
fTTVe6TnJysoqIiy3bkyBEnVgwAAACcv2ZGFwAAaNjsXcW4e/fuavRIvzSktLVVpaannMcgA
AUHv8/Pzk6uqq/Px8q/H8/HwFBgba30eJJ57QXXfdpfHjx0uSevbvsqZKSEt1zzz2aOnWqXFwqX7Pj4eEhDw+P2
j8BAAAAoI5xRToAwFCLFi1Sz5491bdvX7vzWA4AAOqOu7u7wsPDlZmZaR2rqKhQVlaWoqKibibO5z5syZSs1yV1d
XSZLZbK67YgEAAAAD0EgHAFyQ87mK8ZySkhItW7dkKwAwW7ZMZd999d7Wvvw3IAAFC3kpKStHDhQilZskS7du3ShAkTVFJSo
vj4eE1SXFyckpOTLfOHDRumBQsWaWaNmyZTp48KDWrl2rJ554Qs0GDbMO1AEAAIDGgqVdAAAX5PdXMQ4fPlzS/67iS
iTExMtLvvu+++q9LSUt15553Vvg7LAQBA3YqNjVVBQYGmTZumvLw89enTR2vWrLHcgDQ3N1cuLv+7/uCf7/1n4/+6tk9M6/rEf6eOc//rVE28/b9KlJSAbMZ9V+PUS5lSmfp5fZ8ufxDb3B7rl9RvR/
iTExMtLvvu+++q9LSUt15553Vvg7LAQBA3YqNjVVBQYGmTZumvLw89enTR2vWrLHcgDQ3N1cuLv+7/uCf7/1n4/+6tk9M6/rEf6eOc//rVE28/b9KlJSAbMZ9V+PUS5lSmfp5fZ8ufxDb3B7rl9RvR/
```

```
27eTh4aFLL71Ua9ascVK1AIDfc3d3V3h4uLKysixjFRUVysrKUlRUlN193333XZWWlurOO++s6zIBAAAAw3BFO
gDggi1fvlxJSUlKTO9XZGSk0tLSFBMTo5ycHPn7+1eaX1ZWpkGDBsnf318rVqxQUFCQDh8+rNatWzu/eACAJCk
pKUljxoxRRESE+vbtq7S0NJWUlCg+Pl6SFBcXp6CgIKWmplrtt2jRIg0fPlwXXXSREWUDAAAATsEV6QCACzZnz
hwlJCQoPj5eoaGhSk9Pl5eXlzIyMmzOz8jI0IkTJ/TBBx+oX79+CgkJ0cCBA9W7d28nVw4AOCc2NlazZ8/WtGn
TFBYWpq1btyozM9NyA9Lc3FwdO3bMap+cnBxt3LhRd999txElAwBsqOk3RdPS0tStWzc1b95cwcHBmjRpkn755
RcnVQsADQdXpAMALkhZWZk2b96s5ORky5iLi4uio601adMmm/usWrVKUVFReuCBB/Thhx+qbdu2GjVqlB599FG
5urra3Ke0tFSlpaWWx8XFxbV7IgAAJSYmKjEx0eZzGzZsqDTWrVs3mc3mOq4KAOComn5T902339aUKVOUkZGhq
666Snv27NHYsWNlMpk0Z84cA84AAOovrkgHAFyQwsJClZeXW65YPCcgIEB5eXk29zlw4IBWrFih8vJyrVmzRk8
88YReeOEFPfPMM1W+Tmpqnx8fCxbcHBwrZ4HAAAAONDV9JuiX375pfr166dRoOYpJCREgwcP1siRI6u9ih0Am
iIa6TVgMjWdDQDqUkVFhfz9/fXqq68qPDxcsbGxmjp1qtLT06vcJzk5WUVFRZbtyJEjTqwYAAAAqN/OfVMOOjr
aMlbdNOWvuuoqbd682dI4P3DggNasWaMbbrihytcpLS1VcXGx1QYATQFLuwAALoifn59cXV2Vn59vZ6fn6/Aw
ECb+7Rr1O5ubm5Wy7hcdtllysvLU1lZmdzd3Svt4+HhIQ8Pj9otHgAAAGgk7H1TdPfu3Tb3GTVqlAoLC3X11Vf
LbDbr7Nmzuu+++/TYY49V+TqpqamaPn16rdYOAA0BV6QDAC6Iu7u7u7wsPDlZWVZRmrqKhQVlaWoqKib07Tr18/7
du3TxUVFZaxPXv2qF27djab6AAAAABq34YNGzRjxgy9/PLL2rJli1auXKnVq1fr6aefrnIfifvikKoKniinQAwAV
LSkrSmDFjFBERob59+yotLU0lJSWKj4+XJMXFxSkoKEipqamSpAkTJmjevHmevHmaOHGiHnzwQe3du1czZszQQw89Z
ORpAAAAAA3W+XxT9IknntBddd9218ePS5J69uypkpISx2TXPPPZo6dapcXCpff8k3RQEOVTTSAQAXLDY2VgUFBZo
2bZry8vIUFhamzMxMy9dKc3NzrT6EB6wcH6+OPP9akSZPUq1cvBQUF6eGLLEiXr0OUeN0gUAAACgQfv9NOWHDx8u6
X/fFE1MTLS5z5kzZyo1y+u0XgboaGikAwBqRWJiYqRXWJiYpUf0Dds2FBpLCoqS1999VUdVwUAAAAAOHTX9pui
wYcMOZ84c9enTR5GRkdq3b5+ee0IJDRs2zOp+RgAAGukAAAAAAAACNQk2/Kfr444/LZDLp8ccf1/fff6+2bdtq2
LBhevbZZZ4O6BQCot2ikAwAAAAAAANBI1+aZos2bNlJKSopSUFCdUBgANW+W7RgAAAAAAAAAAAAsa6QAAAAAAAAA
A2EEjHQAAAAAAAAAOwxvpM+fP18hISHy9PRUZGSksrOz7c4/efKkHnjgAbVr104eHh669NJLtWbNGidVCwAAA
AAAAABoagy92ejy5cuVlJSk9PRORUZGKiOtTTExMcrJyZG/v3+l+WVlZRoOaJD8/f21YsUKBQUF6fDhw2rdurX
ziwcAAAAAAAAANAmGNtLnzJmjhIQExcfHS5LS09O1levVqZWRkaMqUKZXmZ2Rk6MSJE/ryyy/l5uYmSQoJCXFmy
QAAAAAAAACAJsawpV3KysqsqOefNmRUdH/68YFxdFROdr06ZNNvdZtWqVoqKi9MADDyggIEDDyggIEA9evTQjBkzVF5e7qy
yAQAAAAAAAABNjGFXpBcWFqq8vFwqBAQFW4wEBAdq9e7fNfNfQ4cOKB169Zp9OjRWrNmjfbt26f769FS3bt26f7779fv/76q1JSU
mzuU1paqtLSUsvj4uLi2jsJAAAAAAAAAECjZ/NRmuioqJC/v7+v7evXqHq7Y2FhnNnTpV6enpVe6TmpoqHx8
fyxYcHOzEigEAAAAAAAAADZ1hjXQ/Pz+5uroqPz/fajw/P1+BgYE292nXrp0uvfRSubq6WsYuu+wy5eXlqayszOb
OY+ycnJKioqsmxHjhypvZMAAAAAAAAAADR6hjXS3d3dFR4erqysLMtYRUWFsrKyFBUVZXOfAq3t5Z9+sq0ICE8I8sP+Ws
M7dmzR+3atZ07u7vNfTw8P0Tt7W21AQAAAAAAAAADgKE0Xdk1KStLChQu1ZMkS7b2DIIM/t4C39aqR5uqIBIGgh2WUBNd2Hn3Uw1q0ZpTbg
UpOTrbMnzBhgk6cOKGJEydqz549549WrF0Fk0KHHnjAqFMAAAAAAGo358+crJCREnp6eioyMVHZ2dpV9FD0wFi8irtg0rT5nZjCMUlKx9b5KCdV
PPKB27drJw8NDl156qdasWeVEoiOaWgEAAADnMexmo5IUGxurgs/L+frz/frz8/O6ZdDd48AMUAU1Gx6BuU+bkYjLaHPk9AM8Uqbp
Fgff/yxJk2apF69eikoKEgTJO7Uo48+atQtQpAAAAAI3C8uXlzZSUpqT5tGjR09uNcIpNvTXbo14TzZ1zO3JbEfGg
aNGiQ/P39ctWLFCgUFBenw4cNq3bq184sNHbq184sHAAAA6pihjXRJskKipKX331VzOgcVxVU4Z2zWnuZm0yjJZl+7lRr+Yx9ZCd2F+kxJkcdsrfjtxetALJZUq9ZFUEX9qQ9V1ijV99G6ppmKptKvhC+Lh6S0Ws+
+bMUUJCguXboenp6Vq9erUyMjI0ZcqUSvtnZGToxINfHgeNlq2rs+JqA2ca+SPQf5uUNcKgTzlvTD0ew0dq0
YU8xGlwAAAFBrysrKtHnzZqtlF9ZYtlFV1cXBQdHa1NmzbZ3GfVqb8qnXP0Ssij8rV1dXmP
qWlpSotLbU8Li4urt0TAQAAAOqIoWukAwAAADBeYWGhysvLFhysvLLUssnhMQEKC8vr4e/xw4cEArVqxQeXm51qxZ0oye
eeEIvvPCCnnnnmmSpfJzU1VT4+PpYt0Di4VsdDAAAAqCs0OgEAAADUWEVFhfz9/fXqq68qPDxcsbGxmjp1qtLT06vcJzk5WUVFRbLtLTO
6vcJzk5WUVFRZbtyJEjTqwYAAAA0H8s7QIAAAA0cX5+fnJ1dVV+fr7VeH5+vgIDA23u065dO1116665ed5hL8dd3Zd
pLy9PZWVlcnd3r7SPh4eHPDw8ard4AAAAAm4Ih0AAABo4tzd3RUeHq6srCzLWEVFhbKyshQVFWVzn6urqzr379+mnfv
n2qqKiwjO3Zs0ft2rWz2UQHAAAAGjKuSAcAAACgpKQkLhy5UpGREeAcE+dA0rbcJ0eBhnhtjDc49apdzLeGUmpoVb8eaBpQ+SVJcXJyCgoKUmpo
qSZowYYLmzZuniRMRn6sEHH9TevXs1avdaBdaq9rbJ6Aqc5TZ6AoaAPUcjXQAAAAio2NVUFBgaZNmyZNm6ZrZ6W0Q6crurHM
6a8vDyFhYUpMzPPTcgPS3NxcubEbj87wutwcHB+vjjjzVp0iT1iT16CsDzVp0iTRd6FqT6VLQUFBmjhxo599cdmvFTgjTgEAAAsO0zMzSAAAAEi
SEhMT1ZiYaPO5DRs2VBqLiorSV199VcdVAQAAAMZjjXQAAK2YP3++QkJC5OnpqcjISGVnZ1fZR/0gbLdNX/Oq2cd5d/0/HixTKZTFabp
6enE6sFAAAAABwHFekAwAu2ly5ZWWy3JScnKzMy0ac/HZBwEpPh+TyPaFH7sdjdLyqrxA+0QkQX4T9gUA2Ldi6jUjWdHLc+3KUQiU6I
ln1cza8YCAAAAAFekAwAu2Jw5c5CCCXlN5FafMzQYY0LjjvMhf4ID0gA9B0bEJ0sHhtUmgW2VeSMIGHKbbp6syAyajAiMXRVAC9JYVbmR8v
OCC1JWVafPmzYqOjraMubi4KO6WWp2bapyv9OnT6tjx44KDg7WTTfdp0+++aXvq4uNhqAwAAAAAAcAY
a6QCAC1JYWKjy8vJKg9z5q7du0n3dOD5iVXy7dQ0pWsMe8zxn5YkOoaxf8Vp0Lh8Cxs2mxWfBOfswgLxHCGTCAD3pQnpi
XoeAAAAAAAAAAdGoG3Zq9jU7nLJ8sCDoUK0X3fhbC5W1jPNjEpRXUAmPyk3+xp8yTkJFQVP5RdkArUjiEJG0VEZSqptDTAeaAmPLc0mN
uNgoAuCB+fn5ydXX/fn6+1Xh+fr4CeZ1Av2iHrO4lVfUbe1RAgvDpU5iLMV3BFOyDwtfh2Nxh5Ok1yPB9m0DEAAJ3FSn6KlFANbw06TA8C3EHD
```

1d4eLiysrIsYxUVFcrKylJUVJRDxygvL9eOHTvUrl27uioTAAAAAADgvHFFOgDggiUlJWnMmDGKiIhQ3759lZa
WppKSEsXHxOuS4uLiFBQUpNTUVEnSUO89pSuvvFJdunTRyZMn9fzzz+vw4cMaP368kacBAAAAAABgE41OAMAFi
42NVUFBgaZNm6a8vDyFhYUpMzPTcgPS3Nxcubj870tQP/3OkxISEpSXlydfX1+Fh4fryy+/VGhoqFGnAAAAAAA
AUCUa6QCAWpGYmKjExESbz23YsMHq8YsvvqgXX3zRCVUBAAAAAABcONZIBwAAAAAAAADAdhrpAAAAAAAAAADYQ
SMdAAAAAAAAAAA76kUjff78+QoJCZGnp6ciIyOVnZ1d5dzFixfLZDrJZbZ6enk6sFgAAAAAAAADQlBjeSF++fLm
SkpKUkpKiLVu2qHfv3oqJidHx48er3Mfb21vHjh2zbIcPH3ZixQAAAAAAAACApsTwRvqcOXOUkJCg+Ph4hYaGK
jO9XV5eXsrIyKhyH5PJpMDAQMsWEBDgxIoBAAAAAABwoUymprEBaBwMbaSXlZVp8+bNio6ozy5uLgoOjpamzZ
tqnK/O6dPq2PHjgoODtZNN92k7777rsq5paWlKi4uttoAAAAAAAAAHBUMyNfvLCwUOXl5ZWuKA8ICNDu3btt7
tOtWzdlZGSoV69eKioq0uzZs3XVVVfpu+++U4cOHSr16WuKA8ICNDu3btt7
tOtWzdlZGSoV69eKioq0uzZs3XVVVfpu+++U4cOHSrNTO1N1fTpO+ukfgAAAFTh7SZO+dUos9EVAAAAAKhjhi/
tUlNRUVGKi4tTWFiYBg4cqJUrV6pt27Z65ZVXbM5PTk5WUVGRZTty5IiTKwYAAAAAAHCO+fPnKyQkRJ6enoqMj
FR2drbd+SdPntQDDzygdu3aycPDQ5deeeqnWrFnjpGoBoOEwtJHu5+cnV1dX5efnW43n5+crMDDQoWO4ubmpT58
+2rdvn83nPTw85O3tbbUBAAAAqKwmzZfFixfLZDJZbZ6enk6sFgDwR8uXL1dSUpJSUlKOZcsW9e7dWzExMTp+/
LjN+WVlZRo0aJAOHTqkFStWKCcnRwsXL1lRQUJCTKweA+s/QRrq7u7vCw8OVlZVlGauoqFBWVpaioqIcOkZ5ebl
27Nihdu3a1VWZAAAAAQKNXO+aLJHl7e+vYsWOW7fDhw06sGADwR3PmzFFCQoLi4+MVGhqq9PRePOeX15KSMjw+b8j
IwMnThxQh988IH69eunkJAQDRw4UL1793Zy5QBQ/xm+tEtSUpIWLLLyoyoJUuWaNeuXZowYwYYYJKSkoUHx8vSYqLi1N
ycrJl/lNPPaVPPvlEBw4cOJYtW+vFGnAAAAADR4NW2+SJLJZFJgYKAB9NW2+SJLJZFJgYKAB9NW2+SJLJZFJgYKAB9NW2+SJLJZFJgYKAB9NW2+SJLJZFJgYKAB9
zcrOjraMubi4qLo6Ght2rTJ5j6rVq1SVFSUHnjgAQUEBKhHjx6aMWOGysvLbVYNAQUEBKhHjx6aMWOGysvLnVU2ADQYht5sVJJiY2NVUFCgad0
mKS8vT2FhYcrMzLR8CM/NzZWLy//6/T/99JMSEhKUl5cnX19fhYeH68svv1RoaKhRpwAAAAA0a0eaL7+/gKW65
osknT59Wh07dlRFRYUuv/xyzZgxQ3/605+cUTIANBohISEaN26cxo4dq4svvvvi8j1NYWKjy8vJKf9QMCAjQ7t2
7be5z4MABrVu3TqNHj9aaNWu0b98+3X///fr111+VkpJic5/SOlKVlpZaHhcXF593zQDQkBh+RbokJSYm6vDhw
yotLdXXX3+tyMhIy3MbNmzQ4sWWLLY9ffPFFy9y9y8vDytXr1affr0MaBqAAAoHGw13zJy8uzuU+3bt2UkZGhDz/
8UG+99ZYqKip01VVX6ejRo1W+TmlpGXL1lk1qutSRUWF/P399eqrr
yo8PFyxsbGaOnWq0tPTbGpSBgmpGXLllk1qutSRUWF/P399eqrr
yo8PFyxsbGaOnWq0tPTq9wnNTVVPj4+li04OAgQCgBYCCAjSRAtAAAAABB6xNNWzq25Wh
XXnmlyn1ovgBAZX/961+ldetWZWdn67LLLtODDz6odu3aKTExUVu2bHH4OH5+fnz+fnJ1dVV+fr7VeH5+vgIDA23u0
65dO1166aVydXW1W1jF122WXKy8tTWVmZzX2Sk5NVVFRk2Y4c0eJwjQDQkNFIBwAAAqw8vJyLVq0So7dLS
uu+46q80R59N8+SM3Nzf16dNH+/btq3ILzRcA3rhhx+UkpKi157TtVnZ5wFvz9vP2e7aRMl9f7aHVaFeLvLL
393d3eFh4crKyvRK9IsRUaGsrCxFRU3EUaGsrCxFRU3Kdfv37at2+fKioqLGN79uxRu3t5O7ubnMfDw8PeXt7
AAAAasIkTJ2rixIkqXy9491Lt3b6vNEefTfPmj8vNESfM9X0ZcknUwuRkLuidUlycAAAAUYnaWDaqEz9KC
ysiIkKvvfaabr31V31V11l31VVXaWuwsbFq6kl5FKXNJgDA6wbrMSWdJc+2JqLSM2U+2q3sWNAczi4L7rP6k9FbnJ5V95ZzSMWNMmxZfMHqvtcmkmv1K
6wbUAQFNyxRVXaNCgqQVqwYIGGDx8uNze3bIdq+NmMuueQS3XHHHQ4dLHTGzGz+FrxGa7NdF7kxbzHQWxqKgofFXVzWqGQCaI
hrpAAAAQAP28MMPa+7cuZo3b55MJmteTHasmzZcXX3xRL7744gW9HgBnDggDp27Gh3TosWLfT66687qSIAAAA
OOgEAAIAGbOPGjVq/fr3+9a9/qd+/fUP+l0lOKxlXrlxpUGUAAAAAAACAO+2nLb53c21+Gw2Aqk8LL8A7lgFpH2jv
jcbBQAAABqw1q1b6+abbza6DAAAAAAAAKgCtXNf+pK1fs7XmO4yrA2fu7avZ+tu4zh4p+2LL
773//q8svv7zSeJ8++fTf//7XgIoAALbQSAcAAAAGagYYKCAuXk5EiSunXrprxZt2xpcEQAAAOh4cOXz1enTp2sx
o8dO6ZmzmzWjbAEB9wdIuANDEbN++3eGtJubnPn6+QkBB5UlQZqMIjS2dnZDu23bUk2NX1TOX1EBkicn
GqV27dhowYIAGDBggb9u3bvTtmbHi6Xc7l3fZn4/ULzWL9gxAUA4ReVVLrgTLJ0/qscce0OBgwyBBwEeP
LCwMJlMJpnnZPNNNzvP3vOZDKpvLzcoQBcDAUD0SL5nTkSOnPuwhQTIE60cnZBz5+/tMud+hQ4ceefJk9e/
/7z
OBQAgJSU16d///rf++c9/ql+/fpJ+uwHxz4aNJH/XP0eStHXrVgUB
OjNN980uDoAwDk00gGgGiiJGGwTQByRXyfHywTB9H3nOnD1KSEhQfHy8y8JCk9PV2rVq2yeVz99C9k+ac+ubw/jbE
ukydP1npdANAUvPfee1uqxYuWG9yNN9yG5s2b6/bF9yV6FbZ9rD+Vmg8RtC1yzMzVSVlDZ62Wtat68ueLj4gmK9l5Z
Ei5ubkZXR4A4P/RSAeAaqAmG1WwBr6MUQ9l3SRFSUhu7xNNFPyd/fxX3fffcC+//zzal+
ntLRUpaW1lsfFxcUXVjgANVVNwQAWTbDbGYSKxcmd4Z0p+E4Q23cRJe+v7vNWbCs6VYQ/OYZ
jmFhYUqLy+v1MQJCAjQ7t27be5z4MABrV27VqNHj9aaNWu0b98+3X///fr111+VkpJic5/SOlKVlpZaHhcXF593zQDQkBh+RbokJSYm6vDhw
zpk+frqioKIOrAwA46r///a9yc3NVlZmNe7IZ3IAQN1zuJH+ww8/qH379rUWRIeakjIXZ1atE/WjNxx0tMpAXiWHgpMi
7waBgBQfzh6U8+arJFeE6dOndJJddKQsdd9lhQsXys/S4fXzmne7uiHPVRJpGUhgwiyhOdWFD

```
v3r0lSdu2bZ0np6c+/vhjg6sDAFTnwIEDuvnmm7Vjxw6rexmZTCZJqpPP5ACAmnO4kT5nzhwVFxdXaqJLko+Pj
06dOqU5c+ZolqxZtVogAKB2VVRU1Orx/Pz85Orqqvz8fKvx/Px8BQYGVpq/f/9+HTp0SMOGDatUU7NmzZSTk6P
OnTtX2s/Dw0MeHh61WjsANAY9evTQ3r17tXTpUss3gUaOHKnRo0erefPmBlcHAKj0xIkTdckllygrK0uXXHKJs
rOz9eOPP+rhhx/W7NmzjS4PAPD/HG6kZ2ZmKj09vcrn4+4+Lil+JCQQCMdAJoYd3d3hYeHKysry3K1e0VFhbKyspS
YmFhpfvfu3bVnxcnqOA/Hx8j4VFq09Ph9bTBY9nXjLq6uj2GKXZlPu+8XrHrlsnPz08uLi5ycXHR1Vdfr
dTUVD300EP69ttvjS4RAKAaNNIPHjyoiy++uMrnO3TooEOHDtVGTQAAJyopKdG///1vm+sxPvTQQw4dIykpSWP
GjFFERIT69u2rtLQ0lZSUKD4+Xtjvf2wNCgpSamqqPD091aNHD6v9W9W7duLUmVxgEAtq1atUpDhgyRm5tbtfe+Y
G1dAKjfysvLlapVK0m/fdvzhx9+ULdu3dSxY0fl5QY0A4ByHG+nNmzfXoUOHqmymHzp0iK+OAkAD8+233+q
GG27QmTNnVFJSonbt2qmgqUWsFBeXl7y9/d3uJEeQoLC1NmZqblvhm5ublycXGpy1MBgCZl+
PDhysvLk7+/v917X9TV/S4AALWnR48e2rZtmy655BJRkbqueeek7u7u1599VV16tTJ6PIAAP/P4UZ6Gk3nz
zTQ0YMMDm82+88Yb69u1ba4UBAOrepEmTNGTNGzGzYMKWnp8vHx0dd2riQPWnU/8vHx0dffWV3NzcdOedd2rixIk0lZiYqLNpVwkacOGD
Xb3Xbx4cY1eCwCaut/f76K2730OBAHCuxx9XCUlJZKkp556SrV/5y1/+Sn/5y1/Uv39/XXTRRVq+fLnH48sDJkyf
r9ddf1+TJk61uKJefn6+HH35Yixcv1uTJk8+riPXQdna2Q/stTJk8+riPnz5yskJESenp6KjIxUdna2Q/stW7ZMJpPJ7lU4AICqbd26V
Q8//LBcXFzk6uqq0tJSBQcHa0z9jJBQcH67nnntNjjz1mdHkAgPN08uRJo0sADgoJiZGt9xyiySpSp5cu2r17twoLC3X8+HF
dd911lBlcHADjH4Ub6tddeq/nz9f49vd79ChQ5oa9+f49cEAPzGzc3NsuSKv7+/cj89vf79+/cnNzJUk+Pj46cuSIkaUBABw0a9YsqysWR4w
YoTZt2igoKEjbtm0ozsDIAQHV+/fVXNWvWTDt37rQab9LuCzsvaN++jQab9OmjUwmk0FVAQBsqdGCtGbjj
jv0wgsvaN++fZOowYcJ5FTBnzhwlJCQoPj5eoaGhSk9Pl5eXlzIyqrcp7y8XKNHj9b06dNZLwwALkCfPn30ozTf
fSJIGDhyoadOmaenSpfrrX//KjT8BoIF1ubm66+OKLu
Z8FADQADq+Rfk5QUJAmTZpUKy9eVlamzs3TLm4uKi6OhobdqOqcr9nnrqKfn7++vuu+/W559/Xiu1AEB
TNGPGDJO6dUqS9OyzyouLk4TJkxQ165dtWnTJiRIoOrAwA4Ii8vz9JI/+ijj3T77bdr8ODBCgkJUWRkpMHVAQCq
3XqVD322GN688031aZNG6PLAQBUweFG+t+t///ll///zrMw+7C8SLPVYFXEe6PLAQBUweFG+t///ne/
s3LhRixYt0tatWx16jdLSUpYUvxWWeFxdJWUpWWloeFxcX17h0AGisIiLP/29
/dXZmamgdUAAM6Hr6+vTu/z5ZfXXXWuIqtmzxpyKwpacXatL1+eqhrfnVjhPDn2A/mcQXx/ws6bdurZDeekl9+/atdr9ly5Zp5MiuVum
mm/TBBx+cz2kAQJM1fPhwosAADjA4Ub6wYMHq3zuwIEDuvPO0/X444/r5ZdfdvjF/fz85ZjBx8d8eetgjrpBjvkUcl/4dTxDi
QYGVpq/f/9+HTp0SMOGDb0MVVRUSJKaNWumnJwcde7c2Wqf50RkJSU1WR4XFxdbvvoKAE3d2LFcjNW7cuEqN9K+
//lqvvfaaNmzYYYExhAACHvfjiiwoJCdGRI0fPO3HPPqWXLlpKkY8e06f7773f4duhZ0LSO9MVGRmptLQ0x
cTEKCcnR/7+/lXud+jQIZU/+kb1mPHu5Ptl1M9Sr6bKhSJu9PLqcZLj1vbe1Sdv/bdvq6sb8DhqoZtCjzZJWPyJYGmwoDZPN9Nm1nBYr0wqtC1bRa2iFEC/tLs0g+3
V8+zcKwjyIB2LCDH+y+r/bJYLDnPsSYbH+PKwyUTs8PctL6hEbx+UEudhTUBhv9vk9jQ4hJvnlH9MP
GM1RbT7AjS9GADoEGnUzCAcDI+GlFtlpu3jkFpD0k6p6w9thsv4OUB3Z14Kovu1LnbhLBkKY1IQ7dCbLRVr8t+FE6dgRGXrjrMJCuBkg2gv
Vk0+7QVzD40X9TCEhk1ptRK1JfVNqCtB9fsvAvR9XW3g1Kj+xp5c6EVpSWi0ye3CfNYBAqBdL+jF9XDstNzHQdr6OUbtVCbeDGMIN2Yhfhn6eJpBKbBjXjWAe
Srv8sLTk5ay8UpZS0M0ZPXwM51q/m/WVvIRVYN3KtMEYEnYFF+1RuD8DVFdt9UuK3lmfUcEEZsOkRx4tdZbGMVuz63RPlLLpw61dQ6FRnaCpPZTb27FT
4ocJMBUZz9P7zeKz1v+HRuOFcWZPl/a1n6lz5Hnd+x60HzuX4DdUu97PJjt0FZCKbPk7F57QbxcH2fjxQ1P9dn8O/XhCQDBwnuBx4Ypz8+0FMcwbI5VB
XSNZ1m7jBwYhzgKsEsuhrwPQpbdR1D+8yfkT1ZjU8R3nbCXp61c/P8zl4ocXwBV7cUYOVewWs3fBxdbkrGRRc+FTzH6zj9QMrQkh4CUdrPWPm5pn52uT/
z2uAQ8yEmSttbPEOjQRKbfa8EzURwW0R6TWkvNMrGCYB/NO4mZj2YiePUi79+XHrU1RtIp0hn9GFiMafr7f3jptLUKVAJdDwtgPx1KeDmVxt2QA4L
TNo4oF5nffJz85KZ5Q8ibOUtqRz/jm+Y2aNkQZy64dwuIX/KoSO+bBb6WAtDLp8sRQnaTqgO5I4JKyCJTUaY5w6VBEH7cy0dkNj9s+U0YuE9cuN6C
kmZuSbkBd84+kQiqH5rI+0Dq9c6oYV0XZNRhw4OLTB8uHfRTFXxXbM56lbwunDAD9yZpMGRsrqP+V79Ei3bD0QbM3IjAD2yvO2w/UrhPZt+7WwGAjLw
KEhO6QfdvPXU6s9LwR4yE0R4tA1cVpUN2TWuOdOZxVxT12UAEQmUV8d9O+P23IR3dlbZ+MHpnmZ6ue91ZcuuTxbv+GzywAeKxW2aY0QbfX8Fv
pWjQeDJ/nUaOmm3q5mgEvtNhd34fpNwhNVFOm8fwzhdYDyAZuvm6YJpJ1rMJ3QnOPsdgRF+HTh3/aJkLNpDBBLoMwbNvVDVBfUdyCDlYW
bbj1pXfqfpYXq6dfPuahmTU+g3SdPawR1SFXVyG6VhPrPTPKb1Qs/lr4Lmrtrtdq7kLA40Xaq+TdBq5WuQUOZiqKTjnDu0WUcJWNtoXrrOSg
X4vX3NCDgnS8SEnKHqrS7X/+vhQ2MSOqIqD/zFlHzVdgB3/s8g==
```

xcv1qJFi1RcXKzbb79dpaWl+uCDDxQaGlpXNQIA6sgzzzyj9PR0xcXFadmyZZbxfv366ZlnnjGwMgCAo7p06aL
MzMxKS7D861//Uqd0nWp0rMTERJtLuUjShg0b7067ePHiGr0WA0A3Y8e0NboEAIADHG6kDxs2TJ999pmGDh2qt
LQ0/fnPf5arq6vS09Prsj4AQB3KycnRgAEDKo37+Pjo5MmTzi8IAFBjSUlJSkxMVEFBga677jpJUlZWll544QW
Hl3UBABjn9ddfV8uWLTVixAir8XfffVdnzpzRmDFjDKoMAPB7DjfS//Wvf+mhhx7ShAkT1LVr17qsCQDgJIGBg
dq3b59CQkKsxjdu3FjjqxgBAMYYN26cSktL9eyzz+rpp5+WJWEhGjBggWKi4szuDoAQHVSU1P1yiuvVBr39/f
XPffcQyMdA0oJhxc53Lhxo06d0qXw8HBFRkZq3rx5NVpzEQBQ/yQkJGjixIn6+uuvZTKZ9MMPP2jp0qV6+OGHN
WHCBKPLAwA4aMKECTp69Kjy8/NVXFysAwc00EQHgAYiNzdXl1xySaXxjh07Kjc314CKAAC20HxF+pVXXqkrr7x
SaWlpWr58uTIyMpSUlKSKigqtXbtWwcHBatWqVV3WCgCoZV0mTFFFRYWuv/56nTlzRgMGDJCHh4ceeeQR8/3
ujyAAA00nv2rDZs2KD9+/dr1KhRkqQffvhB3t7eatmypcHVwSim6abqJzUC5hSz0SUAF8Tf31/bt2+v9C3Rbdu
26aKLLjKmKABAJQ5fkX50ixYtNG7c0G3cuFE7duzQww8/rJkzZ8rf31833nhjXdQIAKgjJpNJU6d01YkTJ7Rz5
0599dVXKigokI+Pj82rYgAA9c/hw4fVs2dP3XTTTXrggQdUUFAgSZo1a5YmT55scHUAg0qMHDlSDz30kNavX6/
y8nKVl5dr3bp1mjhxou644w6jywMA/L8aN9J/r1u3bnruued090hRvfPO07VVEwCgjpWWlio50VkRERHq16+f1
qxZo9DQUH333Xfq1q2b5s6dq0mTJhldJgDAARMnTlRERIR++uknNW/e3DJ+8803Kysry8DKAAC0ePrppxUZGan
rr79ezZs3V/PmzTV48GBdd911mjFjhtHlAQD+n8NLu9jj6uqq4c0Ha/jw4bVx0ABAHZs2bZpeeeUVRUdH68svv
9SIESMUHx+vr776Si+88IJGjBghV1dXo8sEADjg888/15dffil3d3er8ZCQEH3//fcGVQUAcJS7u7uWL1+uZ55
5Rlu3blXz5s3Vs2dPdezY0ejSAAC/UyuNdABAw/Luu+/qjTfe0I033qid03eqV69e0nv2rLZt2yaTqWmspwoAj
UVFRYXKy8srjR89epR7GAFAA9K1a1d17drV6DIAAFW4oKVdAAAN09GjRxUeHi5J6tGjhzw8PDRp0qQLaqLPnz9
fISEh8vT0VGRkpLKzs6ucu3LlSkVERKh169Zq0aKFwsLC90abb573awNAUzZ48GClpaVZHptMJp0+fVopKSm64
YYbjCsMAOCQW2+9VbNmzao0/txzz2nEiBEGVAQAsIVG0gA0QeXl5VZLADRr1kwtW7Y87+MtX75cSUlJSklJ0ZY
tW9S7d2/FxMTo+PHjNue3adNGU6d01aZNm7R9+3bFx8crPj5eH3/88XnXAABN1ezZs/XFF18oNDRUv/zyi0aNG
mVZ1sVWYwYAUL989tlnNv/w0WTIEH322Wc1Pl5NLnD5vWXLlslkMrFsLwBUgaVdAKAJMpvNGjt2rDw8PCRJv/z
yi+677z61aNHCat7KlSsd0t6c0X0UkJCg+Ph4SVJ6erpWr16tjIwMTZkypdL8a665xurxxIkTtWTJEm3cuFExM
THncUYA0HQFBwdr27ZtWr58ubZt26bTp0/r7rvv1ujRo61uPgoAqJ90nz5d6T4XkuTm5qbi4uIaHevcBS7p6em
KjIxUWlqaYmJilJ0TI39//yr303TokCZPnqz+/fvXuH4AaCq4Ih0AmqAxY8bI399fPj4+8vHx0Z133qn27dtbH
p/bHFFWVqbNmzcr0jraMubi4qLo6Ght2rSp2v3NZr0ysrKUk50jAQMGnPc5AUBT90uvv6pz587au3evRo8eree
ee04vv/yyxo8fTxMdABqInj17avny5ZXGly1bptDQ0Bod6/cXuISGhio9PV1eXl7KyMiocp/y8nKNHj1a06dPV
6d0nWpcPwA0FVyRDgBN00uvv15rxyosLFR5ebkCAgKsxgMCArR79+4q9ysqKlJQUJBKS0vl6uqql19+WYMGDap
yfmlpqUpLSy2Pa3p1DgA0Rm5ubvrll1+MLgMAcAGee0IJ3XLLLdq/f7+uu+46SVJWVpbefvttrVixwuHjnLvAJ
Tk52TLmyAUuTz31lPz9/XX33Xfr888/P/8TAYBGjkY6AMAQrVq10tatW3X69G1lZWUpKSlJnTp1qrTsyzmpqam
aPn26c4sEgAbggQce0KxZs/Taa6+pWTM+3gNAQzNs2DB98MEHmjFjhlasWKHmzZurd+/eWrdundq0aePwcc7nA
peNGzdq0aJF2rp1q80vwwUuAJoqPmkDAC6In5+fXF1dlZ+fbzWen5+vwMDAKvdzcXFRly5dJElhYWHatWuXUlN
Tq2ykJycnKykpyfK4uLhYwcHBF34CANDAffPNN8rKytInn3yinj17nvf9LgAAxhk6dKiGDh0q6bfPue+8844mT
56szZs3q7y8vE5e89SpU7rrrru0c0FC+fn50bwfF7gAaKpopAMALoi7u7vCw80VlZWl4c0HS5IqKiqUlZWlxMR
Eh49TUVFhdWXLH3l4eFhujgoA+J/WrVvr1ltvNboMAMAF+uyzz7Ro0SK99957at++vW655RbNnz/f4f1reoHL/
v37dejQIQ0bNswyVlFRIUlq1qyZcnJy1Llz50r7cYELgKaqXjTS58+fr+eff155eXnq3bu3XnrpJfXt29fm3JU
rV2rGjBnat2+ffv31V3Xt2lUPP/yw7rrrLidXDQA4JykpSWPGjFFERIT69u2rtLQ0lZSUKD4+XpIUFxenoKAgp
aamSvrtKpaIiAh17txZpaWlWrNmjd58800tWLDAyNMAgAaloqJCzz//vPbs2a0ysjJdd911evLJJ7nJKAA0IHl
5eVq8eLEWLVqk4uJi3X777SotLdUHH3xQ4xuN1vQCl+7du2vHjh1WY48//rh0nTqluXPnVtkc5wIXAE2V4Y305
cuXKykpSenp6YqMjFRaWppiYmKUk5Mjf3//SvPbtGmjqV0nqnv37nJ3d9dHH32k+Ph4+fv7KyYmxoAzAADExsa
qoKBA06ZNU15ensLCwpSZmWlZnzE3N1cuLi6W+SUlJbr//vt19OhRNW/eXN27d9dbb72l2NhYo04BABqcZ599V
k8++aSio6PVvHlz/f3vf1dBQYEyMjKMLg0A4IBhw4bps88+09ChQ5WWlqY///nPcnV1VXp6+nkfsyYXuHh6eqp
Hjx5W+7du3VqSKo0DAOpBI33OnDlKSEiwhHp6erpWr16tjIwMTZkypdL8P66d03HiRC1ZskQbN26kkQ4ABkpMT
KxyKZcNGzZYPX7mmWf0zDPPOKEqAGi83njjDb388su69957JUmffvqphg4dqtdee83qj5cAgPrpX//6lx566CF
NmDBBXbt2rZVj1vQCFwCA4wxNz7KyMm3evFnR0dGWMRcXF0VHR2vTpk3V7m82m5WVlaWcnBwNGDCgLksFAAAA6
pXc3FzdcMMNlsfR0dEymUz64YcfDKwKAOCojRs36tSpUwoPD1dkZKTmzZunwsLCCz5uYmKiDh8+rNLSUn399de
KjIy0PLdhwwYtXry4yn0XL16sDz744IJrAIDGyNBGemFhocrLyy1/GT0nICBAeXl5Ve5XVFSkli1byt3dXU0HD
tVLL72kQYMG2ZxbWlqq4uJiqw0AAABo6M6ePStPT0+rMTc3N/3666/nfcz58+crJCREnp6eioyMVHZ2dpVzV65
cqYiICLVu3VotWrRQWFiY3nzzzfN+bQBoaq688kotXLhQx44d07333qtly5apffv2qqio0Nq1a3Xq1CmjSwQA/
I7hS7ucj1atWmnr1q06ffq0srKylJSUpE6dOlVa9kX67YZ206dPd36RAAAAQB0ym80a03as1Q3ffvnlF913331

```
q0aKFZWzlypUOHY97FwGAMVq0aKFx48Zp3LhxysnJ0aJFizRz5kxNmTJFgwYN0qpVq4wuEQAgg69I9/Pzk6urq
/Lz863G8/PzFRgYWOV+Li4u6tKli8LCwvTwww/rttuU2pqqs25ycnJKioqsmxHjhyp1XMAAAAAjDBmzBj5+/v
Lx8fHst15551q37691Zijfn/votDQUKWnp8vLy6vKm5dec801uvnmm3XZZZepc+fOmjhxonr16qWNGzfW1ikCQ
JPTrVs3Pffcczp69Kjeeecdo8sBAPyOoVeku7u7Kzw8XFlZWRo+fLgkqaKiQllZWVXesM6W6wooKlZaW2nzOw8P
D6iodAAAAoDF4/fXXa+1Y5+5dlJycbBmr6b2L1q1bp5ycHM2aNavW6gKApsrV1VXDhw+39EoAAMYzfGmXpKQj
RkzRhERErbt6/SOtJUUlKi+Ph4SVJcXJyCgoIsV5ynpqYqIiJCnTt3VmlpqdasWaM333xTCxYsMPIOAAAAgAb
L3r2Ldu/eXeV+RUVFCgoKUmlpqVxdXfXyyy9Xee8i6bf7F/3+AhjuXwQAAICGwvBGemxsrsrAoKCjRt2jTl5eUpL
CxMmZmZlg/xubm5cnH53wo0JSUluv/++3XO6FE1b95c3bt311tvvaXY2FijTgEAAABokmpy7KJ+xcBAACg4TK
8kS5JiYmJVS7lsmHDBqvHzzzjjjJ555hkhVAUAAAA0DRd67yJCgsL065du5l5Sam1plIz05OVlJSUmWx8XFxQoOD
r7wEwAAAADqmKE3GwUAAABgvN/fu+icc/cuioyoqKcvg49u5dJP12/yJvb2+rDQAAAGgI6sUV6QAAAACMxb2LAAA
AgKrRSAcAAADAvYsAAAAAO2ikAwAAAJDEvYsAAACAqrBGOgAAAAAAAAAAAdtBIBwAAAAAAAAADADhrpAAAAAAAAAAA
ADYQSMdAAAAAAAAAAAA7aKQDAAAAAAAAAGAHjXQAAAAAAAAAAAOygkQ4AAAAAAAAAAgBOOOgEAAAAAAAAAAAsINGOgA
AAAAAAAAAdtBIBwAAAAAAAAADADhrpAAAAAAAAAAADYQSMdAAAAAAAAAAAAAA7aKQDAGrF/PnzFRISIk9PTOVGRio7O
7vKuQsXLlT//v3l6+srX19fRUdH25OPAAAAAABgJBrpAIALtnz5ciUlJSklJUVbtmxR7969FRMTo+PHj9ucv2H
DBoOcOVLr16/Xpk2bFBwcrMGDB+v77793cuUAAAAAAADVo5EOALhgc+bMUUJCguLj4xUaGqrO9HR5eXkpIyPD5
vylS5fq/vvvV1hYmLp3767XXntNFRUysrKcnLlAAAAAAAA1aORDgC4IGVlZdq8ebOio6MtYy4uLoqOjtamTZs
cOsaZM2f066+/qk2bNNnVVJgAAAAAAwHlrZnQBAICGrbCwUOXl5QoICLAaDwgIOO7dux06xdwxqOPPqr27dtbNeP/q
LSOVKWlpZbHxcXF51cwAAAAAABADdFIBwAYaubMmVq2bJk2bNNggTO/PKuelpqZq+vTpTqys6TJNNxldgtOYU8x
GlwAAAAAAaADqxdIu8+fPVOhIiDw9PRUZGans7Owq5y5cuFD9+/eXr6+vfH19FR0dbXc+AKBu+fn5ydXXVfn5+
Vbj+fn5CgwMtLvv7NmzNXPmTH3yySfq1auauX3bnJyckqKiqybEeOHLng2gEAAAAABxheCN9+fLlSlpKrZ
s2aLevXsrjJiZGx48ftzt5zlzvw/w4YNGjlypNavX6lypNavX69NmzYpODhYgwcP1vi0sZGk25y9
dulT333+/wsLC1L17d7322muwhg0AwBhJSUlauHClixZol27dmnChAkqkqSlRfHy8JCkuLk7JycmW
zyhjIwMhYSEKC8vT3l5eTp9+rRpwAAEN8UBQAAAKpiaC09rKxMmzdvtrq5nIuLi6Kjo7VpOyaHjnHmzBn9+uu
vatOmjc3nS0tLderUKaspKSlEXbt25VZmam5QakubmOnbsmGX+ggULVFZWVpKWpttuu03t2
rWzbLNnzzbqbFACgyeObogAAAEDVDL3ZaGFqrLyy6x6NPqn37
9lbN+N/j5nQA4ByJiYlKTEy0+dyGDRusHh88eLDhCwIAAAAAAAAAAAAcAAAAAAAAAAAAABAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

```
x13yMPDw8nVAgAAAAAA1IzhjXSWAgCAhq2srEybN29WdHS0ZczFxUXR0dHatGmTgZUBAAAAAADUDkMb6SwFAAA
NX2FhocrLyxUQEGA1HhAQoLy8vFp7ndLSUhUXF1ttAIDaxUUuAAAAgG2GNtJZCgAA4KjU1FT5+PhYtuDgYKNLA
oBGhYtcAAAAgKoZ1khnKQAAaBz8/Pzk6uqq/Px8q/H8/PxabawkJyerqKjIsh05cqTWjg0A4CIXAAAAwB7DGuk
sBQAAjY07u7vCw80VlZVlGauoqFBWVpaioqJq7XU8PDzk7e1ttQEAagcXuQAAAAD2NT06gLqWmpqq6d0nG10GA
DRqSUlJGjNmjCIiItS3b1+lpaWppKRE8fHxkqS4uDgFBQUpNTVV0m8Nm//+97+Wf3///ffaunWrWrZsqS5duhh
2HgDQVNm7yGX37t219jqlpaUqLS21P0YiFwAAADQUhl2RzlIAANB4xMbGavbs2Zo2bZrCwsK0detWZWZmWhoyu
bm50nbsmGX+Dz/8oD59+qhPnz46duyYZs+erT59+mj8+PFGnQIAwAm43wUAAAAaKs0uSP/9UgDDhw+X9L+lABI
TE2vtdTw8PFizEQCcIDExscr83rBhg9XjkJAQmc1mJ1QFAHCEMy9ySUpKsjwuLi6mmQ4AAIAGwbAr0qXflgJYu
HChlixZol27dmnChAmVlgJITk62zC8rK9PWrVu1detWq6UA9u3bZ9QpAAAAAA0e97sAAAAA7DN0jfTY2FgVFBR
o2rRpysvLU1hYWKWlAFxc/tfrP7cUwDmzZ8/W7NmzNXDgwEpX0wIAAABwHPe7AAAAAKpm+M1GWQoAAAAAMB4Xu
QAAAABVM7yRDgAAAKB+4CIXAAAAwDZD10gHAAAAAAAAAAKC+o5EOAAAAAAAAAAIAdNNIBAAAAAAAAAALCDRjoAAAA
AAAAAAHbQSAcAAAAAAAAAAAwA4a6QAAAAAAAAAAA2EEjHQAAAAAAAAAAAO2ikAwAAAAAAAAABgB41OAAAAAAAAAAADso
JEOAAAAAAAAAAAIAdNNIBAAAAAAAAAALCDRjoAAAAAAAAAAAHbQSAcAAAAAAAAAAwA4a6QAAAAAAAAAAA2EEjHQAAAAA
AAAAAO2ikAwAAAAAAAAABgB41OAAAAAAAAAAADsoJEOAAAAAAAAAAIAdNNIBAAAAAAAAAALCDRjoAAAAAAAAAAAHbUi
0b6/PnzFRISIk9PT0VGRio709vu/HfffVfdu3eXp6enevbsqTVr1jipUgBAVchyAGj4yHIAaPjIcgCoG4Y30pc
vX66kpCSlpKRoy5Yt6t27t2Jt7t2JiYnT8+HGb87/88kuNHDlSd999t7799lsNHz5cw4cP186d051cOQDgHLIcABo+s
hwAGj6yHADqjslsNpuNLCAyMlJXHGF5s2bJ0mqqKhQcHCwHnzwQU2ZMqXS/NjYWBXn/NjYWJWU10ijjz6yjF155ZUKCwt
Tenp6ta9XXFwsHx8fFRUVydvbu0a1mkw1mt6gne9PhW1603iTzCkX8J9NU/lBupBoebuJvEejav4eXUiG1SVnZ
710/u9FU/lPUCLLHXHeec4PUvWaSpZLNc5zsvx/+GxevQv5SNVU8pzP5g7gs3n1Gslnc7K8fiLLq0eWO4Asr14
dZ7mhV6SXlZVp8+bNio60toy5uLgo0jpamzZtsrnPpk2br0ZLUkxMTJXzAQB1iywHgIaPLAeAho8sB4C61czIF
y8sLFR5ebkCAgKsxgMCArR7926b++TlB++Tl5dmcn5eXZ3N+aWmpSktLLY+Liook/fbXBlTtvN+eX2q1jHqLnx8HXMh
7dKb2yqjXzuM90vezZ/CXiaw4I8sl8vx8k0XV4+fHAef7HjWVLJdq/B6R5WR5TVzQW9NE8pyfHwfw2bx6jeCz0
Vlef5Hl1ePnxwFkefXq0MsNbaQ7Q2pqqqZPn15pPDg42Bqg4fH6MrqN98ZvIGVYsfouo1nP97d0rUKfk0sfe
YPK+5JvYjcl7IcwfwglS988xzsvx/yPKqNbEfkfNCljuAH6Tq8dncYWR5zTWhH4/zRpY7gB+k6tVxlhvaSPfz8
50rq6vy8/0txvPz8xUYGGhzn8DAwBrNT050VlJSkuVxRUWFTpw4oYsuukimer6Dg3XkyJF6s95afcN
7VD3eo+o1pPfIbDbr5MnqfcoPCMDGmpRiaO+x8i1DaCXQzIwNXzRyfv9kjaCUyhPTeSrT9cSWW6JHbkbMY9LRBn
8sovEfV4z2qXkN6j2qS5YY20t3d3RUeHq6srCwNHz5c0m8BnTWpRvrTERJv7REVFKSsrS3/9618tY2vXrlVUVJTN+R4eHvLw8LAa8/X1rRbnx9XVwxmfGnc0Y85e3eztGl+ajTgEtKw7qTs8ejsw+N
N+R4eHvLw8LAaa926dW2U7zTe3t71/of0aLxH1eM9ql5DeY/q29UuzshyiTxvKniP7OP9qV5DeY/I8v8hyxsf3
qPq8R5Vr6G8R/Upz8lyxzWUny8j8R5Vj/eoeg3lPXI0yw1f2iUpKjxoxRRESE+vbtq7S0NJWU1Cg+Pl6SFBc
Xp6CgIKWmpkqSJk6cqIEDB+qFF17Q0KFDtWzZMv3nP//RqRpaAECTRpYDQMNHlgNAw0eWA0DdMbyRHhsbq
4KCAk2bNk15eXkKCwtTZmam5WYXubm5cnFxscy/6qqr9Pbbb+vxxx/XY489pksvvVQXvIXqd5du+qDDz5Qjx49jDoFAGjyyHI
AaPjIcgBo+MhyAKg7hjfSJSkxMbHKrlt2LCh0tiIESM0q7KeB4eHkpJSan0lSn8D+9R9XiPqsd7VDvI8
qrxM1Y93iP7eH+qx3tUO8jyqvEzVj3eo+rxHlWP9+jCkeVV4+ererxH1eM9ql5jfY9MZrPZbHQHQRAAAAAAAAAAD
UVy7VTwEAAAAAAAAAoOmikQ4AAAAAAAAAAgBO0OgEAAAAAAAAAsINGOgAAAAAAAAAAdjQzugCgJv7+97/bHDeZTLr/bHDeZT
PL09FSXLl00YMAAubq6nUrkyNCRPPfWUk+eLC8vL6vxvxn3/+Wc8//7ymTZtmUGVA00Ge40KR5YDxyHLUBvcIcMBZ
ZjtrQVLLcZDabzUYXATjqkksuUUFBgc6cOSM/Pz9JUkFBgZZS8//WQ/X+/XoFBwcbXK1xfv31/5mrXXuzmStG/fPt177777/39t6/H/fNqnptbbtHrybvR2/urWTh
/35z39Wenq6unbtanQ59Y6rmuOHTsmf39/q/Eff/xR/xR/v7+Ki8vN6gyoOkz6tHlttHlgPGI8urR5ZXjzwHjEW
WV48sr15TyXKuSK+n9u7dq/Xr1+v48eaqKiweq6x/BXnfMyYMUOvvvqqXnvtNXXu3FmStG/fPt177777/fPt1777265571
K9fP91xxx2aNGSVqxYYXC1xnFzc9PGjUGZeLxkleNfF1yHXEzDsbHAEvBGgIGGf26vzvvpdd11/fHyvyzPF+XFP5DRLEfyNVZF4vj89l7jfe
1srx6ZHn1yHMA4A1leNbK8emR5/y6C5h7NxRG59ZpKlnNFej3KyrjnYjyRCPZnKSLl2RjK0DVlWnZGgFPEiZGwAQyWVm6Hjs38dkRDK52leqKsAXW3uaPCZLeKAc4Q1n5Tr6D9twOfZgqiGB8fVs2LnOlgoqhq+D/yVtZgzcm1nYqHpZ2OMb+9d8ECE+pbefr5sf52rJWtE6GQNxHDLEXMGeC2BLwoOeP0h2LP+ZiN86l5x2hDEmw0NFfj6Xe0OhF6q0tIFdBFiWtG5uI7+pbJs24lMIUAnyXJYNy9Qb+0zWYWhqtRsf8fJ7QJ+MHwgx4e/oiGgI9ljLAXwEmgvNl5GTwLV7IN/
K9fP91xxx2aNGmSVqxYYXC1xnFzc9P27duNLqPeMpvNMplca3bdumNm3aGFARGiuyvGrkefXIcvvIcjgLWV4
1srx6ZHn1yHM4A1leNbK8emR59ZpKlnNFej20cOFCTZgwQX5+fgoMDJTr/7Ooalp/Mlf+vbL+nZG1l1Lg+A12S2YsiIrxrMnfVTrg+1bbfwnLYpX0xc0Xe4xWK6z7ot6P8vp6wm8/fc8bDmgfqWpZvyHy0bHqMUEWsGBqmYs7eO/WmeLzKPw3Qy4yQQQLF+6xI0nx3S1e4MyAdcOhaNnLs1h07f6J/MuEtyPmcQlPu08Lo4pp3ZZP6tcvkW56edO+VZyrJN5Rm+9j/a
mY1/u233+rWW2/VgQMH9OWXX+rWW2/VsWPHjCmynpg0aZI8PDw0c+ZMo0upN3x9fWUymVCRvb2+r/7bKy8t
1+vRp3XfffZo/f76BVaKxIMvt88dQ5ZXRpbDDmchy+8hyx5DltpHncBay3D7y3DFkuW1keZ3ojuaZZZ/TSs
8/q0UcfNbqUeufYsWM6e/ZspfGRoY+/fRThYeHq0WLFbPz5kzx6D
KjJ0WliaZ2axx48Zp+vTp8vHxsTzn7u6ukJAQFGVghGh0y3D7y3DFkeWVk0ZyJLLePLHcMWW4beQ5nIcvtI
8sdQ5bb1tSynEZ6PfTTTTTz9pxIgRRdRL1177bW699dprr6lPz6Sfvsr6YQJE3TdddmS2KFLLrnEyDL
rhZ07d+ryyyy+XJ03Zs8fq0Vtft2kKxowZI+m3DwYvmn3NeD69eunZ6Zfz6MjClY3jyx3DFluG3k0OZ
yHL7SPLHUOW29bUspyWwlXeqhu+++W1deccxppmLUuzXBlhpgVXiIB6j...
```

```
1b98+7d+/XwMGDFDz5s2rvAFEU3LFFVdoypQplvXeQkNDdcstt+ibb77ROKFDlZaWZnSJaATIcseQ544hyysjy
+EMZLljyHLHkOW2keeoa2S5Y8hyx5DltjWVLKeRXg/Z+7qMyWTSgQMHnFgNGqoff/xRt99+u9avXy+TyaS9e/e
qU6dOGjdunHx9ffXCCy8YXaJhfHx8tGXLFnXu3FmzZs3SunXr9PHHH+uLL77QHXfcoSNHjhhdIhoBshy1gSyvG
lkOZyDLURvIcvvIc9Q1shy1gSy3r6lkeeNeuKaBOnjwoNEl1Fvl5eVavHixsrKydPz4cVVUFg9v27d0oMqq38
mTZokNzc35ebm6rLLLrOMx8bGKikpqUmHvNlstvzsfPrpp/rLX/4iSQQoODlZhYaGRpaERIcvtI88dQ5ZXjSyHM
5Dl9pHljiHL7SPPUdfIcvvIcseQ5fY1lSynkV7PnfvCAF8T+c3EiRO1ePFiDR06VD1690B9seOTTz7Rxx9/rA4
dOliNd+3aVYcPHzaoqvohIiJCzzzzjKKjo/Xvf/9bCxYskPTbB6yAgACDq0NjRJZXRp47hiyvGlkOZyPLKyPLH
UOW2Oeew5nI8srIcseQ5fYllSynkV5PvfHGG3r++ee1d+9eSdKll6qRx55RHfddZfBlRlr2bJl+sc//qEbbrj
B6FLqvZKSEnl5eVUaP3HihDw/ayoqP5ISOvT6NGj9cEHH2jq1Knq0qDqrYXXXpjXy3DFke
dXIcjgLWV41stwxZLl95DmcgSyvGlnuGLLcvqaS5TTS66E5c+boiSeeUGJiomJiYmRmWrfTbU/bJ2rhxo+677z4VFhZq0qR
JBldoHHd3d8t/jLCvf//+Wn19L/+uOv7hUVFXruuedO7bXXGlydsXr16qUdO3ZUGn/++eVEoj
Mhy+8hzx5DlVSPL4QxkuX1kuWPIcvvIc9Q1stw+stwxZLl9TSXLudloPXTJJZdo+vTpiouLsxpfsmSJnnnzyySa
9vtcLL7ygAwc0aN68eXzdqBo7d+7U9ddfr8svv1zr1q3TjTfeq0+++04nTpzQF198oc6dxtdouE2b96sXbt2S
ZJCQ0N1+eWXG1wRGh0y3D7y3DFkefXIctQlstw+stwxZLljyHPUFbLcPrLcMWS5Yxp7ltNIr4c8c8PT21c+f0Sn8
R3Lt3r3r27KlffvnFoMqMd/PNN2v9+vvVq06aN/vSnP/Sz2bTp9+rQuv/xyP
fDAA2rXrp3RpRnq+PHjio2N/Wwgfwbe2+21WrZsmdq2bWtsgWgUyHL7yHPHkeW2keVwBrLcPrLcMWS5YxpHx
OdTp041uox658EHH7TuAJvb/LVhBVbGbG0LBBSUyHL7yHPHkeW2keVwBrLcPrLcMWS5YXpHx
ccWR51chz1DWy3D6y3HFkedWaSpaztEs91KVLF/3jH//QY489zjW+fPrLqVU+ArTLQjrDQm5gGdfBaBfsP+aAd
8mk8vJyAyqsHyoqzIkip9XU2S3NzcVFFRSkefPmadu2bTp9+rQuv/xyde3a1ejSDHOp7LNnSVsmM4ZdVFVj/sI5Ysi
yqpHnqGtkuX1kuePI8vqo1llSznivR66DLZn/0AvVJpsHrrd2uVvCbd/rnc+Ksb8DzZOa97Dk0XvfTg+xSRXPSgU
pHlNUOeo7aR5Y4jy6tGlsOZyPKqkeWOI8vtI89R18hy+8hyx5DljmnroAV/+sU3m+ScQY1X6KvTmo1J0WwdLb6KrbCSG
L65aPllKmb6J+l2N8bmyeX9fvZ89mwtFd8o4U1lYXKMyjIaLDKEMTtrE2GB83+CzyeblB9+zMv+6FBgTevfzbrdFG
r9ytMNmt3Q3sUkT4+m2gZVrB2OaeXjxjXHR8FgIl+PjFedNOsbOmQG+EqeBMF7FLkGuu6l9GN3GrZa6RIS
oZt3bR5BZWGDC+4+DBLuXpeAVS8H2ZFuowfjPwt7y3ZeAP+3AHgMHY1e+mq/lVqtW6EjDjC1KedO3+G/tyR62NzSTt
IMzPpeUT4drfW7t8NuNXmdxfvs6dunZmGZLszXDTzkNdxyfrEqNPbLdOMRmzyH55PqZuAtuJLR/LzRiZq/IHcSbcH
Kq1ed7AvMyoV0Hux8qNOeSFoEEWz4QgVK5VSZmgbG+1Ldv8ksu4K8MFukRuFZR7b6R2ib/iKmLsyi6sQvz7KffMOtyXc
P8yMVvdUuz+TBtHV3uaDpsCGgSOg6Cf53Cr4+Onb1fHmSxTc9iaZE0jmLpSsWwvAJwvXZ8dBTr8hXxaMLBiQjzS/vPPS
dT9Sd5R9mdPA4+vqhJBeE9gA+JCCSUWnIUTcxZY3zaLw5L9fsnKeshUusMA0B0HLj+2lNpKPONB4zN3qkfz6p76JDe
SVe/FkD8eJp5d3F93mpJXS3nhNKDTyKKyVxzLdXrjaqBtq6d8w07Y+f1AGQEH7fqGQR9obKG+0BtvrHzt27BbHvZ+zP7
b6tszB6vzZYmS6vMPY4zOZQ7dFVsVqBcNQp6VQ8XkSzxurTEyt5LWkcbfSWRz7HunHvJrF6ngtbIVBmyOVrcY4Ghl3
Htyi/39iJoEYtqz8wovm5neHuJ9R0JRjvbDeFTOXr07tmYsZxpnGd+K3Pq/ykXenKUKdjYBXPLE+tJvHXyaEuxtz
IHVbbUsY0S1CJWhk0PDZcj79TuxbN2QDJtd3RcGR2BJ7i4LdL6ZQFWbrG5i0E4fUj/QtsUFWeB9RhZMnAadwTj9vWP
```

69KSUnRsGHD9Oyzz6q8vFwjR45U165d1atXL/XoOUOHDhOyukw0EmS5feS5Y8jyyshyOBNZbh9Z7hiy3DbyHM5
ClttHljuGLLetqWU5jfR6aM+ePQoLC5MkvfvuuxowYIDefvttLV68WO+9956xxRnsxRdfVGJiokJDQ7V27Vq1b
NlSknTs2DHdf//9BldXv/BX5cqmTJmiBQsWKDAwUBkZGbrlllv07bff6u2339ayZcvUrFkzTZ061egy0UiQ5fa
R544hyysjy+FMZL19ZLljyHLbyHM4C1luH1nuGLLctqaW5c2MLgCVmc1my1cfPv30U/3lL3+RJAUHB6uwsNDIO
gzn5uamyZMnVxqfNGmSAdXUb+f+qtyly1L17d61evdry3vevj+2v06NEGV4n
Ggiy3jzx3DFleGVkOZyLL7SPLHUOW20aew1nIcvvIcseQ5bY1tSynkV4PRURE6JlnnlF0dLT+/e9/a8GCBZKkg
wcPKiAgwODqjLd//36lpaVp165dkqTQ0FD99a9/VadOnQyurH558cUX9jSvjhB8sNQC6
99FJ5eHhY/cK79NJLlZeXZ1R5aGTI8uqR59Ujyys+FMZHn1yPLqkeW2kedwFrK8emR59chy25palpvMZrPZ6
CJgbfv27Ro9erTRyc3Oc30VljJQUSdKDDz6oH3/8UW+//bbRn448H4CQFUTi//bbBFRrn448/1o033qiwsDD169dPkvTFF19o27Zt+uc
//6lBgwYZXCHqMxcXF+X15cnf31+S1KpVK23vbHLHUOW20aew1nIcvvIcseQ5bY1tSynkV4PRURE6JlnnlF0dLT+/e9/a8GCBZKkg
1nIcvvIclyIppblNNIbkF9++yUWurq5TD90nTRzxMZo5c6YTZaWUUTMTRzxMZo5c6bV+JQJQpU/TJJ59oy5YtSvfnfn4
aOnSoJOlvf/ubXn31VYWHDuqdd95VX0Rx44dDa7Q+VxcxXLRkyRL5+PhIkaaOHKmQtDTLVQgnT55UfHx8ow41E9k+
W/Ic8eQ5ZWR5agPyPyPLfkOWOIcttI89hNLL8N2S5Y8hy25paltNIr4e+//57zZw50+4vWFP+w2Qn69lfmu+5cz
BAQoKCjK6PMN4enqfqnq1q9X4Xnj171KtXL/3yyy/GVVb/dOvVWTQsWLNB1112nTZs2KTo6Wi+++KI++ugjuzmNk6fi+++KI++ugjN
WvWTCtXrjS6RKdzcan+/somk6nRBDyMR5ZXjTx3DFleGVkOZyPLq0aWO408Ysd408hzOR5VUjyx1D1tvtvW1LKcNdL
roe3bt+v6669X69atdejQIZZauXKXK1cnNz9cYbbxhdomHatm0rk1/atm3bnmvu3Vgr4rVu3OXLEsi7VB
x98oFtvvvVX33HOP+vXrp2uucbY4gYzgxy7gYzgDOQ5faR544hyysjy+FMZL19ZLljyHLbyHM4C1luH1nuGLLctqa
W5dX/2QBOl5SUpPP6enp6W8RtuEGfffffffaZgZUZLyEhQhQffcc49mzJ8P8mhHQffcc49mzZqlzz//XsVe//Ly//XJ9//rlmzpype++
UaXV6+0bNlSP/74oyTpk08+UwZ5CaZ9//tnIOoAmgSy3jzx3DFkOGIsst48sd0+sdwx5DhiLLLePLHcMWQ4Yiyy3jxo6lJ
rpW+++UavvPKKBgW8e0IhJSRRpLd8L+Uo4SQkBCUgfhS/3sSyoPMWx8XZhOEUVmpRWLFJZ4tBfjJtMBBmQYw1N3BlWrXvwQf/DtMXuy5NztNZrbxjr
GfPHt1www2Sp2pO+++04hSc0AWS5feS5Y8hywFhkuXlkuWPIcsBYZLl+TZWXjZWl9ZLljyHJIXfFeL3l4eKi4uL+Jyp4
9e9S2bVsDKqo/TCaJk2apKNHhJaehSsEyK6bcKJPZHR59cr8+fMVFRWlgoICvffee6e7VsWu
9bIkSMNrg5o/Mhy+8hzx5DllZHlqA/I/I8t+Q5Y4hy20jz2E0svw3ZLljyHLbmlqW00ivh77//nvNnDnT7i9YU/7DZCfr2V+a77lz
11fDhwzVgwAClpaUZXCGqMkNt0TOnq3r27/Pz8i9KBz89A9x5WlgN6M4IAyHUYJgyRwPnPjfTsWJy0W
Ei33Xab/vOf/+jTTz9VVGNXVWx++fqHP+pTBy99mfiVGNmpQhYlmzpxpuKUXpnZqVGunC2hvoe
cYf1z5swZebmbmqaszGq8eB1UENA1kuX3kuWLH4PKaIcsBY5D19pHnjiHLbSPPoTSyvDdkuWPIctuaWpbTSK+HvqP3
dPp06d1+eXWAIq7Pgjf1NP/7b6ljpDpQhYSOlP6qu1KHpvqObcOcFBIuxtoizZyOCB2/3IVb5p9d+QFjZtpvsbmf+3
fX17u5uWNl5E9O5i17qHFW1kNOb3WCwbn5yCmgcsxGZzJkhnX+J0ciJGcNfmYTjwlk2qb4GK5WT1Xv7GHt+75
baR53AGsrx6ZLlj2HLb02cDhn2cDQW6bhVIiPHp0uv7D9ysmeWqI7bsbEEYm1+Z
IubiqGPhjhoaJC8vb2VmpqoUOHGl2ik5iBc9ZSH2XjbiJ+W0rvcfcl+XXzPNBQUFywW2KyMjQ3//9/15ptvGluhGuyv
HrkuWPI8srIcjgLWL49stwxZLlt5DmcRy6vGlnuGLLctqaW5TTS66GbiO9A3MR4hsYrwFyn6d0VQ3+yLFb23aU6Sf+Sz
lll8sYz///LOmT5+uqKgfdb2tOmMHJDZPCNRbhuryLoyA+ew4FUmWKPjQwKu+ufFePPHUPAaGTS6pWFIEuwQQZsDvrw
Ndcc40eeOAB8bZzZwVh8aGLLePPHcMWV44ZWQ5nInsddQ5nlsvtl8sdQ5bx5bbRp7ZeEZLuMH+9PDSatahTfvdHm+B
vn9q3b6+OHTtWuvvSdSl/V7kK2bdhvk6empjz/+2Dq6pdu3BopJydHISE6t27kt1555
RWFhIQoPT1d7a3a1M7oHsK1ajCxnyOBgt27efnG1EaGiGy3D7y3D+efnXfX1JiL1rmsBh
beQ5nIcvtI8sdQ5bb1tSynkV4PRURE6JlnnlF0dLT+/e9/a8GCBZKkg3i1b+nO1HxHdbGOLJC1klRHgJbSk7HB7
NgxSVJKSor+/Oc/a+nSpXJ3d9fixfz+4gnp6e+OnTp4sDOBJmjRpktXqXXukuX3uX43BX44sDhDVuX3kuW
TWW4fWe4Ystw28hzsQQpbbR5Y7hiyHLbmlqW00ivh77//nvNnDnT7i9YU/7DZCfr2V+a77lz
fzDDz+srVu3Kisry8ry8mY8mVAYBjyHKyHEDDR5b/hjwHOJCR5b9palnOFeloUH788UFFkn77q9bChQv1a9i
wYRowYIDB1dVvXl15euvzy8ycqTkEvl5zH3rdd4vxa95PeWDMvpkHy08xx3zJLL+ntt982uSvd9Gz73g
w9ZTpYD9QlZfn7I8t+Q5Y4hy20jz2E0svw3ZLljyHLbmlqW00ivh77//nvNnDnT7i9YU/7DZCfr2V+a77lz
Dhg3TkSNH1LVrVy1btkVJJSIhcJXF5WU1GjFihSbtb/7QR+4zFgHxivK/VR/Vvb1+Xq3Xz7Hs
EmSdODAAZO+fVpJSU1V/hUVqCmy3DbyHM5C1luH1nuGLLctqaW5c2MLgCVmc1my1cfPv30U/3lL3+RJAUHB6uwsNDIO
Rzz77rIYMGaK+fftKrKkyzs5WZmalJkJybp4MGDPd998t8ZmqqJ22Qyp5mfGiSxdGO+1/+gFSpQcFu3Tm6crPeLN9Zuaat
mKOcOFCSdKDDz6ozZs366uvvjK6vnpdqtUbezEJJUHvclXRhzq8pmwr4OlAjdaaJRBbUTuy5FQDD/YW+J+o7O3Ti3yXFSjlm3wFUBnXp2P5
GV4bGhCy3jTyvHlnuGLLcNvIcZyLLq0aWO4Yst408h1PJ9mLKrI43hcwwbfKKUvvvazGo9ObZdAc4qCSXaO4MkOnMfnhXdB8SuoTvtyXbbrFvZ/
Mknn+i9997TSy+97Bqd3ra4Q+vPdvVt2WPx+5f9T8vdPn9v69Xd2PUtZ8SFdR9t2f4f2LtLWnVzLtb

```
B4P+R5baR5wAaErLcNrIcQENClttGlgM152J0Aajs448/VnR0dKXx66+/Xh9//LEk6YYbbtCBAwecXZphTpw4o
cDAQElSy5Yt1aJFC/n6+lqe9/X11alTp4wqr14pLy/X9u3b9fPPP1d67ueff9b27dtVUVFhQGXGe+KJJ3T27Nk
qn8/NzdWgQYOcWBEaM7LcNvLcMWR51chyOBNZbhtZ7hiy3D7yHM5ClttGljuGLLevqWU5jfR66P/au/voqOoD/
+OfIc+EhxAUJRBIEBCwhAcparBYrAultFDjwdOyyqOCLvgAUghLRWxdBLryJD2VagXlgI+NPK2IASwYgoILAYK
ESISFdhUo2WhDCCEz9/eHP1PGJM7PMz33pn36xV8/fr1Sk5OliSdPXtWT
Zs2DfU0o757T7O67nEGaeXKlRo7dqxiY2NrnYuJidHYsWPD6kEPwXjllVf0/e9/X4WFhbXOLVu2TN/73vcUHc0
zmHFl0PL60fOG0fL60XKEEi2vHy1vGC0PjJ4jVGh5/Wh5w2h5YJHW8vD5TsLIk08+qYcfffglgffPBBzf27du/er
XfffVcvvPCCJCk3N1d33HGHyZkhN3r0aMXFxUmSKisr9dBDDykxMVGSdP78eZPTHOVPf/qTpk6dqioFrnoq0
jNW3aNC1dulT33XefgXVmFRYWatKkSerTp4+eeuopTZ8+XX/96+Cdz//Fc96JMEHL60fPG
0bL60fLEUq0vH60vGG0PDB6jlCh5fWj5Q2j5YFFWsu5R7pD7dixQ0uXLtXhw4clSTfeeKMeeeQRZWZmGl5mht2
nZy9fvvwqL3G+Vq1aadeuXUpLS6vz/NGj0bNmQkZ8/n9+/rtttu09mzZ00
8zDl0njyp++67T1u2bFFiYqI2bNgQkZ8+AOBctLxhtByA09Fye+g5ACej5fZESsu5R7pDlZSU6uNe//xxe5jZ
6I82ZM0ddunRRamqysvL1a1bN/Xv31+ZmaOXuK//bx8+fP6
29/+5uBRUBkoeWB0XN7aDlgFi0PjJbbQ8sBs2h5YLTcHlHloSYo2PQC1xcbG6sUXX9SsUXX9SsWbN04MAB1Ze
euD9SZGVlSZI8Ho9Gjx6tuLi4mnNer1f79u1LTzQmXlMWXbCmjlpRlYBkQWW
l43em4PLQecgZbXjZbbQ8sBZ6DldaPl9tByXIwL6Q6Wmpq1NTUmtc5OTmaPXu29u/fb3CVGd+GyrIsNW3aVAk
JCTXnYmNjdeutt+rBBx6eb+tbHDWMQ+D2Z2LiYlRWqannvu0QPLgMhEy/3Rc3toOeAstNwfLbeHl
gPOQsv90XJ7aDkuxoV0h1m2bJlyc3MVGxurx57QsvXF4KJ9e8+nUpPT9fu3bvXAuhaBnLbHrZ5LTpL554QsXFxRF7/67ly5dLktqJ554QsV64nTQ
7qRF7/7DgHaNWNo0HwdNWSZmEA5dTeYXgAEulMUMKjlRBO+zbzGhrx2bwVrUiqLyydHaMbcj5BlkFmmKj2N
An88nSUpPT9fu3bt1zTXXXGF4ERB5aXj96bg8tB8yj5fWj5fbQcsA8Wl4/Wm4PLcfFPJZlWaZH4Btz587VrFmzJ7ZZ7aBpgzJFB5cfbQcVxstB64+Wo6rjZZ
JGRoaKiIlmWpZkzZ+r555/XY489Xg5cfbQcVxstB64+Wo6rjZZ
HHh426iDLly/Xiy++qE8++UQBN27UuXPnlJ+fryNHjig705vASzp58qTuv/9+paSkKDo6WlFRUWFGRUjq+aXNEHFgBBWjrh9z3N+D8/pnTlh4YUYQBTG2ZPtxvD2FTACHHhjzpsMyK5hFmU2SFBCHgiY3K1V4DsP8uWAgPjpsVw1+UcTdfPH0afFxgJ1ztIU+QKwiLGAVVN+sTvQxWgPE+1dqDnvXR3KGjNfmRe0U+Gu6C8vvjjdXvIrmUCIM5+88GkuPRh9mRq0Zj4rY5W8VeMZhsvEdUhNeVPFs+zGUXt6kBFUwuGfKqttKBdtpFvifBFu7wtQpkGb
xSKZfLqqTj93k7pTYvdXZDjAGGv1XjGQu8JeH9
hFk7I5Bm3VcQIhL5GU0omKhRTdODOFqtrVGs0dqD9pi4b01Q2iD5GuwOh+OTg8X52VCTG8VnJiwGi9m3X1fQgR6E9HdSxf0wIMsjmrTmDhDmQkq8+K4A9dZkuHfjP7GkvJHbQnfzakWF6pf6qx3U2cFa7CIYHjt8q35VGj5gbdHuj4gEhHo1Q1VfrWm9+vkCmLgYD
Lc8cxmxh8QfK/OYXtOsbE0MBJ3qsvYMP8ld1c3nLjcUU3udb8gzwTvFncc9DMhbZb1+jSbJ0Fmfbk7mhZxGAGq5pdbfjZ+9AbW3rDAF8yEFrF8MQAkvFnbfRvwC2IRMD67aQoHQmAkKz/Xm3j0Lsl4jFx5lHNN4L4V9nZ4uMkyy+BGPf+zqGjVHlWeR1sjvOJt7rfVmEfrvWDBQpgAKgrB8FHu8tfx6LMhPVwD/sGdj0O+GJFSsY74y3MRyyHgR1yNgYaYBgzUvpF8QEYJiRCK1uJi+jMUOd2RP1rfBlyuWMLOe8vG6RxNAV2kXv/1B2DT7tsPkm6iIZs7WwjrtP+WAX7VBCtk+xKb1VOBaQGsZgAbtyy+4zkXGSwJaLeIuPuhaZsbDn3hQ7BYszm9EhqTXNwjRMu5LTOO2BDWKWkDkZGuthdqA6KFXPVAnZ0ZqzD+yCXFt+WJZ6xbI43CO0nR92BR2EQsEpP+rMBANxWw2+PDz3K+xxQZt1PubHPZN+kNN9mOCoV37sChFODQbbvRxFK9FW5W0Bmd7zwwnfYwRDQtl2ua5vBsfWVuHtGM2+n1FdmZrHI+WZnUPbIBdbDmxFSp0HCcBHYNNF4wyxVhsdB2+/gYvfknGQ5JjmtThdmMa2o7P24sFJ3SZFOXc6RXhqZq9NuYEl59X3H+aqB6n4FrUm3l1yN3wsJJNHTOUEe30DA1YkpTl5cPxHKiV
```

```
l3reKtWrfT3v//dwCLnSk9PV0FBgdq3b+93/L333lPXrl0NrQIiBy0PjJ7bQ8sBs2h5YLTcHloOmEXLA6Pl9tB
ySFxId6Sf/exnWrNmjebMma03335bCQkJysjIOObNm3XHHXeYnmdUUlKSvvjiC6Wnp/sd37t3r9q0aWNolTNNm
TJFEydOVGVlpSzL0q5du/Taa6/p2Wef1UsvvWR6HhD2aHlg9NweWg6YRcsDo+X20HLALFoeGC23h5ZDkjyWZVm
mRwB2TZ06VR9//LHeeustde7cWXv27NHJkyc1cuRIjRw5kvt3fceqVas0e/ZslZSUSJJSUlLO9NNPa9y4cYaXA
Yh09Nw+Wg7AqWi5fbQcgFPRcvto0biQDlepqrSxIkTtWLFCnm9XKkVHR8vr9WrEiBFasWKF3wwMfIlll1dbVWr16
tQYMG6brrrlNFRYXKy8vVql9Ur09MAQBI9t40WA3A6Wt4wWg7A6Wh5w2g5vsWFdIdITk5WcXGxrrnmGrVo0UIej
6fery0tLQ3hMmc6fvy4CgsLVV5erl69eqlTp06mJzl048aNdejQoVr37wJw9dDy4NHzwGg5EHq0PHi0PDBaDoQ
eLQ8eLQ+MlkPiHum0sXDhQjVt2rTm14EiD6ldu3Zq166d6Rm01rdvX+3du5fIAyFEy4NHzw0j5UDo0fLg0fLAa
DkQerQ8eLQ8MFo0iU+kwwWmTJmi3/72t0pMTNSUKVMCfu2CBQtCtMr53nzzTc2YMU0TJ0/WzTffrMTERL/zGRk
ZhpYBiFT0PHi0HIDT0PLg0XIATkPLg0fLIXEh3ZGioqL0xRdf1LrX0pkzZ9SqVSt5vV5Dy8wYMGCA3nnnHSUlJ
WnAgAH1fp3H49WrVtDuMzZGjVqV0uYx+0RZVnyeDwR998REGq0vDZ6HjxaDphFy2uj5cGj5YBZtLw2Wh48Wg6
JW7s4Un3vbZw/f16xsbEhXmPeBx98U0evEdjRo0dNTwAiGi2vjZ4Hj5YDZtHy2mh58Gg5YBYtr42WB4+WQ+JCu
qMsWbJE0jfvaL30ktq0qRJzTmv16vt27erS5cupubBZbhvF2AGLceVRMsBM2g5riRaDphBy3ElOXJI3NrFUdL
T0yVJ//M//60bdsqKiqq5lxlxbKzSOtL0m9/8Rrfcc0F9u5ZkZWXZ/tqcnJyruMRh06dEiS1K1bN
z322G064YYbDC8Dwhctrx89vzSOHAg9Wl4/Wn5paDkQerS8frT80tBy8I10B/n2x0Q0QGDBignJwctWjRwvAiZ2j
evHnNry3L0jvvvKPmzZurT58/kqT//u+kqT//u//VllZWVB/EESCTZs2aejQoerZs6f69esnSdqxY4duuukmrV+/Xv/yL
/9ieCEQnmh5/eh58Gg5YAYtrx8tDx4tB8yg5fWj5cGj5ZD4RLoreL1eHThwQ03bt4/48E+fPl2laV64YUXat5
N9nq9+rd/+zc1a9ZMv/vd7wwvvdI5evXppm0KBBmjt3rt/x//qqvvfDxW5Oy3DzZCAdsqfd/zgcdF+f1ltAqrx6Qk
rgoKCWscLCgrUqlWr0A8CIgwtD4ye2PLAbNoeWC03B5aDphFyw0j5fbQckjcI92Rzpw5o5jHVGZcsxFGn0V1g2f5r48
0Hq3Lmzxo4dq8WLFxteZ9aYMWMr3OWBx98U0PHj9fnn3+uzMxMSd/
cv2vevHmaMmWK4XVPA+KPlgdFze2g5YBYtD4yW20PLAbNoeWC03B5aDkmSBcdp166dtWnTJqu6utpKTU21Nm3a
FmWZRUWFlpJSUmG15nl9XqtkZ9Xqtrf1PmWSkpKZbH47E8Ho+VkpjzZs3z6qurjY9C3jaGvFEpAfr7XwRTJkVaNmdxrrwvBx33ZtQOqzEVLKHCjd9zIB+7L1jd4yPP39THENc2QeH1Q=
=4aMlAmNUhfnSoGPpaBQGyYT1xgn2yiVHKaqwGNZYuE8Bo3g9CVUVB5LakbbVzqWC8bWtMB/rHzIPN/46QaXBC
+Gk89i7ugLJ/ZWVzrtddyybWTqJ3bpG25zNJi2fQTL+yzgLmvb7iotoa8oPqWVMd9Vd8w3zX+QSsObXv4yPtsG
HSQQJ4L5AqCFOTQfc5zSeeS2jlQ5jK6t2VfZ9LddkfQYWVTGCL2MGYRnAQtB2avvH/gUW1NPM+VIMCdIXnF2xkF
+N+VQDKWL0tYpdWA4aw8T9p+D1tGXatnT+0tFuWfrFNKcL72Irx/2yFjGLmkT6FixiPe+77WFlQK1HTnKY9HwSoX
J8n3bNzrDJgtnsoEOKaEb3phJzJMSb3L90DqFmUj+uz98qoxYdk5ggI+8VnOoG5URGXHphbYSrUD+qv5OtjD2pnE
9Y4Jl+Vl1jU1G+dkkavcqw4ptq49lLFa2QF2HOMoxojQnG9Jc5rJpH2/VR/B5u+9HjDgRIpwXvTzkVROb4Jz6OdLX9
7IE9eFQwVw4UTYDVe1VfyidgftIF9lnSZLbXLWCNHPY/z0+GN3XrNqrFjfrEF9nD2djN1nQKmTaCySHAX2BkWy
UyUyXNhmTHumKfWEfJ5a7jfw0D8fZ7eBWpZKeNmfCmDFoTJ+JjLRoyA7ePSd2RcQdkrO7rf43S2eGc4ERpGRqfh
```

+n5555hk999xzKi8vlyQ1bdpUTzzxhGb0nBmR92M6d+6ctmzZop/+9KeSpBkzZuj8+fM156OiovTb3/5W8fHxp
iY6TosWLbRw4UKNHj3a9BQgItHyutHz4NBywCxaXjdaHhxaDphFy+tGy4NDyyHxiXRHmjlzpv70pz9p7ty56te
vnyQpLy9Ps2fPVmVlpf7jP/7D8MLQe+WVV/Rf//VfNYFfunSpbrrpJiUkJEiSioqKlJKSosmTJ5uc6ShxcXE1/
/0ACD1aXjd6HhxaDphFy+tGy4NDywGzaHndaHlwaDkkSRYcp3Xr1tbatWtrHV+zZo2VkpJiYJF5t99+u7Vu3bq
a102aNLFKSkpqXq9cudK69dZbTUxzrDlz5liPPPKI6RlAxKLldaPnwaHlgFm0vG60PDiOHDCLlteNlgeHls0yL
ItPpDtQaWmpunTpUut4ly5dVFpaamCReUeOHFH37t1rXsfHx/v9+FXfvn01ceJEE9Mca9euXdq2bNigm26
6qdaDMHJycgwtAyIDLa8bPQ8OLQfMouV1o+XBoeWAWbS8brQ8OLQcErd2caQePXpo6dKlWrJkid/xpUuXKiMjw
9Aqs8rKyvzu1XX69Gm/8z6fz+88pKSkJGVlZZmeAUQsWl43eh4cWg6YRcvrRsuDQ8sBs2h53Wh5cGg5JC6kO9L
8+fM1ZMgQbd68Wbfddpskaef0nTpx4oTeffddw+vMaNu2rQoLC2uejvxd+vV19l9uegIQ0Wh53
eh5cGg5YBYtrxstDw4tB8yi5XWj5cGh5ZCkyHw0scPdcccdKi4u1t13362ysjKVlZUpKytLBw8e1MqVK03PM+I
nP/mJZs2apcrKylrnzp07p7p8p+rby8PX15dV6hxnA1UPL60bPLw0tB8yg5XWj5ZeGlgNm0
PK60fJLQ8sjm8eyLMvOCNizb98+e7dW16v1/SUkDt58qR69uyp2NhYTZo0SXO7d5YkHT58WEuXLL1V1dbX27t2
r6667zvBS5Qzh79qweeeeQRvfrqq/L6fJKkgojRw5Us5Us8//SDt4g8zvKAWCBMjRuRPq7pd8qLS3VkSNHEkd03ZUcnKy4UX0dM011
+jtt9/WD3/4Q7/jH3zwge69915+BAkwhb/Ez1vGC0HnImW/xMtbxgtB5yJlv8TLW8YLYfEw0bhQsnJyerbt6/
pGY5XUFR57vHrVq1UkVFhYFFAOCPnje0Ml4nmeMF0oOw0lo6cNoaSQ+ke4oWVlZAc+XlZVp27ZtvFsKW37o
x+pZcuWevXVVxUfHy/Wg4upAMAwDkAuB8tBwD3o+XgQjoQpp5++mn95je/5jBjAJyjRkyBBJUmxsrM6ePx/bt6
2pucBAGyg5QDgfrQcANyPlIkSjRkyBBJUmxsrM6ePSuPx6PJkyfrj3/8o+F1AAA7aDkAuB8tBwD3o
+WQuJA0hK0WLVroH//4hySpTZs2Kiws1lCSV1ZWpoqLC5DQAgE20vnwZeWV7/HgHrXXX79/Q8AIAgO0OWL
Vro4cPSd/mJyLMv0CABXmlpqSorK5WSkiKfz6f58+f58/h4gzH6kA4D7OXJnAf
C5J63z8WCuB4LThDoiHiNOHLJVKfibzyBP/NAhaKDC+tFJbsMpGkPFBQbDAw8X3dstbc3mDovGm1AmfNrFcGfUfzRJYLbG5ySR11IO84xDFQNnUaFt+/ysr
xM9iL6zZvn56lhSgIJgEBAwAw8Xe4hsjh3SAmhNkfE+MDMBAOnfFzrkF4xPaIT1ZoxIMUJyZjUzLumdBQKaMv7
3ZZ81zZ46JLJJEjUz4oh4B4cBLfYK8z1MmXgcYOCzBCz
cx4Ok7dUN4X3A8ZZAI3o98/Y+NLmb/F/f8+x5rp2+CJv8ShemTUYhG1ej2X0BJ3Q1bLn8g/XKNCDkHMNnfnZiNz
nvy8w2sqxGrCr6ZvG9CxK7F3W1K/ACyQbh9y/a6eLAyuZBDKhWeHEpl36SuLtWwhd+ZXBfGIhg1KXP5Q+rsz
Rs3TU7nV9YrOa72o+RkNW+l/M/XArlXv6B/nmBzPXGJ+eFO6Zh2yWZ6BIkVc7Ub1nKJlDu4f0MwAqr4ptWvc6+L
AZOH2Fv/DfrAwAaw/JDI+uCkA/TuBXw6g4zxEWZx4C/Nqj3h0YiRl8ZTJzUJdJnGZ4hMx7ghKxs+a3SlOJo4o
PhOLn1U6Xt8yf+9hHw7Ty5ZIMR1EyN8I5EdLUcENtQ1Cm9k7E1ANpEV+i7cJSYcQ4ll6xfgN1fVDMQbYPjXmNlg
vJyMY0G+zpTNi2bJh/XRB+4xFuMl7wT5WeazABi4WsC2v/HHyJdPcXy+0dEWw3Yab9+VoAxZFPCTc/pzDY/Wm+rEh
ZHPP5mVFTdyZpqSyTk5jRgPKv8yTeE+FWuVbymC/J5j+5wJngn+oeRHE4E2Ip9ZWlNyu+M4v0qqz3fJ0+x8OZZaX
9D5BPZx/sqfh1zdO8sBTY7bYmjNeTcVpWAyeWJNwl1l4o4kYJtWEFnKOe8ZGfqvPfQPa5W5eDe7kxnHQ6ZcL4lcXn

```
aWkpFjz58+3LMuyZsyYYXXr1s3v958+fbolyfq///s/y7Isa9y4cdb48eP9vubDDz+OGjVqZJO7d86yLMtq376
9tXDhwqv3TQNAGKLnAOB+tBwA3I+WIxxFm72MDzjLtddeqyFDhmjFihWyLEtDhgzRNddcU3O+pKREFy5cUL9+/
WqOxcTEqG/fvjpO6JAk6dChQ7rlllv8ft/bbrvN7/W+ffu0f/9+rVq1quaYZVny+Xw6evSounbtejW+PQCIGPQ
cANyPlgOA+9FyhBMupAPfMXbs2Jof/fn97394Z5eXlmjBhgh599NFa53jgBgBcGfQcANyPlgOA+9FyhAsup
APf8eMf/1hVVVXyeDwaNGiQ37kbbrhBsbGx2rFjh9q3by9JunDhgnbv3q3HH39cktS1a1etW7fO7+/76KOP/F7
37t1bn376qTp27Hj1vhEAiHDOHADcj5YDgPvRcoQLHjYKfEdUVJQOHTqkTz/9VFFRUX7nEhMT9fDDD+tXv/qV3
nvvPX366ad68MEHVVFRoXHjxkmSHnroIX322Wf61a9+pcOHD2v16tVasWKF3+8zffp05efna9Kk+SSooKNBnn32
mtWvX8hAMALiC6DkAuB8tBwD3o+UIF1xIB+rQrFkzNWvWrM5Ztm99fvXv3lpEjR7RpOya1aNFCO
jc/MvTnP/9Za9asUY8ePfTCCy9ozpw5fr9HRkaGtm3bpuLiYv3gBz9Qr169NGvWLKWkpFz17w0AIgk9BwD3o+U
A4H6OHOHAY1mWZXoEAAAAAAAAAABOxSfSAQAAAAAAAAAIgAvpAAAAAAAAAAAEwIVOAAAAAAAAAAAC4EI6AAAAA
AAAAAABcCEdAAAAAAAAAIAAuJAOAAAAAAAAAEAAXEgHAAAAAAAACAALqQDAAAAAAAAAABAAF9IBAAAAAAAAAAi
AC+kAAAAAAAAATAhXQAAAAAAAAAAALgQjoAAAAAAAAAAAH8P9xk3Cdysp6DAAAAAElFTkSuQmCC\n"
```

```
          },
          "metadata": {}
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "## Model Evaluation Results\n",
        "\n",
        "The following table summarizes the performance metrics of different
models:\n",
        "\n",
        "| Model                  | AUC  | Recall | Precision | Accuracy |\n",
        "|------------------------|------|--------|-----------|----------|\n",
        "| Logistic Regression    | 0.63 | 0.66   | 0.18      | 0.62     |\n",
        "| Gradient Boosting      | 0.57 | 0.17   | 0.43      | 0.88     |\n",
        "| RandomForestClassifier | 0.75 | 0.51   | 0.75      | 0.93     |\n",
        "| XG Boost               | 0.74 | 0.51   | 0.69      | 0.92     |\n",
        "\n",
        "### Explanation:\n",
        "\n",
        "- **Logistic Regression:**\n",
        "  - The model achieves moderate performance with an AUC of 0.63. It has
relatively high recall but low precision, indicating that it captures a good portion
of positive instances but also misclassifies many negative instances.\n",
        "  \n",
        "- **Gradient Boosting:**\n",
        "  - This model shows lower performance across all metrics compared to other
models. It has the lowest AUC and recall, indicating poor discrimination power and
difficulty in correctly identifying positive instances.\n",
        "  \n",
        "- **Random Forest Classifier:**\n",
        "  - The RandomForestClassifier performs well overall, achieving the highest
AUC and accuracy among the models. It also has a relatively high precision, indicating
that the positive predictions it makes are mostly correct.\n",
        "  \n",
        "- **XG Boost:**\n",
        "  - The XG Boost model performs similarly to the RandomForestClassifier, with
slightly lower AUC and precision but comparable recall and accuracy. It remains a
strong performer in the evaluation.\n",
        "\n",
        "In summary, while RandomForestClassifier and XG Boost demonstrate robust
performance across multiple metrics, Logistic Regression and Gradient Boosting show
weaker performance in different aspects of classification.\n",
        "\n"
      ],
      "metadata": {
        "id": "_TnzSk0pqd07"
      }
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "tMB3CeT6FPah"
      },
      "source": [
        "## Hyper Parameter tuning for the Two best models"
      ]
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "U0OvDHlPFfDU"
      },
      "source": [
        "### RandomForestClassifier"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "metadata": {
        "id": "IQybiLNQeWzH",
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "outputId": "9a2ec09e-8baa-4282-a521-84843dee32e8"
      },
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
            "              precision    recall  f1-score   support\n",
            "\n",
            "         0.0       0.94      0.97      0.96       275\n",
            "         1.0       0.71      0.49      0.58        35\n",
            "\n",
            "    accuracy                           0.92       310\n",
            "   macro avg       0.82      0.73      0.77       310\n",
            "weighted avg       0.91      0.92      0.91       310\n",
            "\n"
          ]
        }
      ],
      "source": [
        "from sklearn.model_selection import GridSearchCV\n",
        "param_grid = {\n",
        "    'n_estimators': [100, 200, 300, 400],\n",
        "    'max_depth': [None, 10, 20, 30],\n",
        "    'min_samples_split': [2, 5, 10],\n",
        "    'class_weight': [None, 'balanced', 'balanced_subsample']\n",
        "}\n",
        "\n",
        "rf = RandomForestClassifier(random_state=42)\n",
        "\n",
        "grid_search = GridSearchCV(estimator=rf, param_grid=param_grid, cv=3,
scoring='f1')\n",
        "grid_search.fit(X_train_sm, y_train_sm)\n",
        "best_params = grid_search.best_params_\n",
        "\n",
        "# Train the model using the best parameters\n",
        "best_rf = RandomForestClassifier(**best_params)\n",
        "best_rf.fit(X_train_sm, y_train_sm)\n",
        "\n",
        "# predict class and evaluate model\n",
        "y_pred = best_rf.predict(X_test)\n",
        "print(classification_report(y_test,y_pred))"
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "**Summary:**\n",
        "\n",
        "Before hyperparameter tuning:\n",
        "\n",
        "|           | Precision | Recall | F1-score | Support |\n",
        "|-----------|-----------|--------|----------|---------|\n",
        "| Class 0.0 |    0.94   |  0.98  |   0.96   |   275   |\n",
        "| Class 1.0 |    0.78   |  0.51  |   0.62   |   35    |\n",
        "|-----------|-----------|--------|----------|---------|\n",
        "| Accuracy  |           |        |   0.93   |   310   |\n",
        "| Macro avg |    0.86   |  0.75  |   0.79   |   310   |\n",
        "| Weighted avg |  0.92   |  0.93  |   0.92   |   310   |\n",
        "\n",
        "After hyperparameter tuning:\n",
        "\n",
        "|           | Precision | Recall | F1-score | Support |\n",
        "|-----------|-----------|--------|----------|---------|\n",
        "| Class 0.0 |    0.94   |  0.97  |   0.96   |   275   |\n",
        "| Class 1.0 |    0.71   |  0.49  |   0.58   |   35    |\n",
        "|-----------|-----------|--------|----------|---------|\n",
        "| Accuracy  |           |        |   0.92   |   310   |\n",
        "| Macro avg |    0.82   |  0.73  |   0.77   |   310   |\n",
        "| Weighted avg |  0.91   |  0.92  |   0.91   |   310   |\n",
        "\n",
        "**Interpretation:**\n",
        "\n",
        "Before tuning, the Random Forest classifier showed high precision and recall
for Class 0.0 but lower metrics for Class 1.0. After tuning, precision and recall for
Class 1.0 decreased slightly, impacting overall accuracy. Further adjustments may be
necessary to balance performance across classes.\n"
      ],
      "metadata": {
        "id": "Nub4-4xI1ZeG"
      }
    },
    {
      "cell_type": "markdown",
      "metadata": {
        "id": "QszOHwDGFymm"
      },
      "source": [
        "### XG Boost"
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "param_grid = {\n",
        "    'max_depth': [3, 4, 5],\n",
        "    'learning_rate': [0.05, 0.1, 0.2],\n",
        "    'n_estimators': [100, 150, 200],\n",
        "    'min_child_weight': [1, 3, 5],\n",
        "}\n",
        "\n",
        "\n",
        "XGB = XGBClassifier(objective='binary:logistic', random_state=42, n_jobs=-
1)\n",
        "grid_search = GridSearchCV(estimator=XGB, param_grid=param_grid,
scoring='accuracy', cv=5, verbose=2, n_jobs=-1)\n",
        "grid_search.fit(X_train_sm, y_train_sm)\n",
        "\n",
        "best_xgb_clf = grid_search.best_estimator_\n",
        "y_pred = best_xgb_clf.predict(X_test)\n",
        "print(classification_report(y_test, y_pred))"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "vT1mxHQAs39t",
        "outputId": "426337b4-7993-4ecc-acb2-a236be15985f"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
            "Fitting 5 folds for each of 81 candidates, totalling 405 fits\n",
            "              precision    recall  f1-score   support\n",
            "\n",
            "         0.0       0.93      0.97      0.95       275\n",
            "         1.0       0.64      0.46      0.53        35\n",
            "\n",
            "    accuracy                           0.91       310\n",
            "   macro avg       0.79      0.71      0.74       310\n",
            "weighted avg       0.90      0.91      0.90       310\n",
            "\n"
          ]
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "**Summary:**\n",
        "\n",
        "Before hyperparameter tuning:\n",
        "\n",
        "|           | Precision | Recall | F1-score | Support |\n",
        "|-----------|-----------|--------|----------|---------|\n",
        "| Class 0.0 |    0.93   |  0.96  |   0.95   |   275   |\n",
        "| Class 1.0 |    0.59   |  0.46  |   0.52   |   35    |\n",
        "|-----------|-----------|--------|----------|---------|\n",
        "| Accuracy  |           |        |   0.90   |   310   |\n",
        "| Macro avg |    0.76   |  0.71  |   0.73   |   310   |\n",
        "| Weighted avg |  0.89   |  0.90  |   0.90   |   310   |\n",
        "\n",
        "After hyperparameter tuning:\n",
        "\n",
        "|           | Precision | Recall | F1-score | Support |\n",
        "|-----------|-----------|--------|----------|---------|\n",
        "| Class 0.0 |    0.93   |  0.97  |   0.95   |   275   |\n",
        "| Class 1.0 |    0.64   |  0.46  |   0.53   |   35    |\n",
        "|-----------|-----------|--------|----------|---------|\n",
        "| Accuracy  |           |        |   0.91   |   310   |\n",
        "| Macro avg |    0.79   |  0.71  |   0.74   |   310   |\n",
        "| Weighted avg |  0.90   |  0.91  |   0.90   |   310   |\n",
        "\n",
        "**Interpretation:**\n",
        "\n",
        "Before tuning, the XGBoost classifier had decent precision and recall for
Class 0.0 but lower metrics for Class 1.0. After tuning, precision and recall for
Class 1.0 slightly improved, while those for Class 0.0 remained relatively stable.
Overall, accuracy improved slightly after hyperparameter tuning.\n"
      ],
      "metadata": {
        "id": "donzHgmd1otn"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "# SQL"
      ],
      "metadata": {
        "id": "BmW_BvhCRFzT"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "In this notebook, I am utilizing DuckDB for executing SQL queries as part of
my project. DuckDB is a modern analytical database management system (DBMS) that is
designed to be embedded within applications and is particularly suited for analytical
workloads and data science tasks."
      ],
      "metadata": {
        "id": "kz7fZsBQR7z9"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "## What is DuckDB?\n",
        "\n",
        "DuckDB is an open-source, in-memory analytical database management system
built primarily for read-heavy workloads. It is designed to deliver high performance
```

```
while consuming minimal resources, making it ideal for analytical tasks, data
exploration, and interactive querying."
   ],
   "metadata": {
    "id": "AhycgKwTRbU3"
   }
  },
  {
   "cell_type": "markdown",
   "source": [
    "Let's Connect to duckdb"
   ],
   "metadata": {
    "id": "qYNjB56QSHtV"
   }
  },
  {
   "cell_type": "code",
   "source": [
    "# inititating a connection\n",
    "import duckdb\n",
    "connection = duckdb.connect()"
   ],
   "metadata": {
    "id": "HY2vEvwBRgOU"
   },
   "execution_count": null,
   "outputs": []
  },
  {
   "cell_type": "code",
   "source": [
    "# registering my dataframe with duckdb\n",
    "connection.register('credit_data', df_sql)"
   ],
   "metadata": {
    "colab": {
     "base_uri": "https://localhost:8080/"
    },
    "id": "8M-a5DbhSK1o",
    "outputId": "63b45e45-2efe-46b8-c47d-c74383613f3c"
   },
   "execution_count": null,
   "outputs": [
    {
     "output_type": "execute_result",
     "data": {
      "text/plain": [
       "<duckdb.duckdb.DuckDBPyConnection at 0x7f4e8f260ef0>"
```

```
            ]
          },
          "metadata": {},
          "execution_count": 1361
        }
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "df_sql.head()"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 313
        },
        "id": "76L5e8zzVJDf",
        "outputId": "c3072cf0-4d52-4cdc-e81d-a0331d0192f3"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "    Ind_ID GENDER Car_Owner Propert_Owner  CHILDREN  Annual_income
\\\n",
              "0  5008827      M         Y             Y         0       180000.0
\n",
              "1  5009744      F         Y             N         0       315000.0
\n",
              "2  5009746      F         Y             N         0       315000.0
\n",
              "3  5009749      F         Y             N         0       166500.0
\n",
              "4  5009752      F         Y             N         0       315000.0
\n",
              "\n",
              "             Type_Income         EDUCATION Marital_status
Housing_type  \\\n",
              "0            Pensioner  Higher education        Married  House /
apartment   \n",
              "1  Commercial associate  Higher education        Married  House /
apartment   \n",
              "2  Commercial associate  Higher education        Married  House /
apartment   \n",
              "3  Commercial associate  Higher education        Married  House /
apartment   \n",
              ″4  Commercial associate  Higher education        Married  House /
apartment   \n″,
              ″\n″,
              ″   Mobile_phone  Work_Phone  Phone  EMAIL_ID Type_Occupation
Family_Members  \\\n″,
              ″0             1           0      0         0      Unemployed
2   \n″,
              ″1             1           1      1         0         Unknown
2   \n″,
              ″2             1           1      1         0         Unknown
2   \n″,
              ″3             1           1      1         0         Unknown
2   \n″,
              ″4             1           1      1         0         Unknown
2   \n″,
              ″\n″,
              ″   label  Employed_years   Age  \n″,
              ″0      1            0.00  52.0  \n″,
              ″1      1            1.63  38.0  \n″,
              ″2      1            1.63  44.0  \n″,
              ″3      1            1.63  38.0  \n″,
              ″4      1            1.63  38.0  ″
            ],
            ″text/html″: [
              ″\n″,
              ″  <div id=\″df-756f4a22-a538-4af8-a34a-ec8457bab32c\″ class=\″colab-df-
container\″>\n″,
              ″    <div>\n″,
              ″<style scoped>\n″,
              ″    .dataframe tbody tr th:only-of-type {\n″,
              ″        vertical-align: middle;\n″,
              ″    }\n″,
              ″\n″,
              ″    .dataframe tbody tr th {\n″,
              ″        vertical-align: top;\n″,
              ″    }\n″,
              ″\n″,
              ″    .dataframe thead th {\n″,
              ″        text-align: right;\n″,
              ″    }\n″,
              ″</style>\n″,
              ″<table border=\″1\″ class=\″dataframe\″>\n″,
              ″  <thead>\n″,
              ″    <tr style=\″text-align: right;\″>\n″,
              ″      <th></th>\n″,
              ″      <th>Ind_ID</th>\n″,
              ″      <th>GENDER</th>\n″,
              ″      <th>Car_Owner</th>\n″,
              ″      <th>Propert_Owner</th>\n″,
″      <th>CHILDREN</th>\n″,
″      <th>Annual_income</th>\n″,
″      <th>Type_Income</th>\n″,
″      <th>EDUCATION</th>\n″,
″      <th>Marital_status</th>\n″,
″      <th>Housing_type</th>\n″,
″      <th>Mobile_phone</th>\n″,
″      <th>Work_Phone</th>\n″,
″      <th>Phone</th>\n″,
″      <th>EMAIL_ID</th>\n″,
″      <th>Type_Occupation</th>\n″,
″      <th>Family_Members</th>\n″,
″      <th>label</th>\n″,
″      <th>Employed_years</th>\n″,
″      <th>Age</th>\n″,
″    </tr>\n″,
″  </thead>\n″,
″  <tbody>\n″,
″    <tr>\n″,
″      <th>0</th>\n″,
″      <td>5008827</td>\n″,
″      <td>M</td>\n″,
″      <td>Y</td>\n″,
″      <td>Y</td>\n″,
″      <td>0</td>\n″,
″      <td>180000.0</td>\n″,
″      <td>Pensioner</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
″      <td>House / apartment</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Unemployed</td>\n″,
″      <td>2</td>\n″,
″      <td>1</td>\n″,
″      <td>0.00</td>\n″,
″      <td>52.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>1</th>\n″,
″      <td>5009744</td>\n″,
″      <td>F</td>\n″,
″      <td>Y</td>\n″,
″      <td>N</td>\n″,
″      <td>0</td>\n″,
″      <td>315000.0</td>\n″,
″      <td>Commercial associate</td>\n″,
"       <td>Higher education</td>\n",
"       <td>Married</td>\n",
"       <td>House / apartment</td>\n",
"       <td>1</td>\n",
"       <td>1</td>\n",
"       <td>1</td>\n",
"       <td>0</td>\n",
"       <td>Unknown</td>\n",
"       <td>2</td>\n",
"       <td>1</td>\n",
"       <td>1.63</td>\n",
"       <td>38.0</td>\n",
"     </tr>\n",
"     <tr>\n",
"       <th>2</th>\n",
"       <td>5009746</td>\n",
"       <td>F</td>\n",
"       <td>Y</td>\n",
"       <td>N</td>\n",
"       <td>0</td>\n",
"       <td>315000.0</td>\n",
"       <td>Commercial associate</td>\n",
"       <td>Higher education</td>\n",
"       <td>Married</td>\n",
"       <td>House / apartment</td>\n",
"       <td>1</td>\n",
"       <td>1</td>\n",
"       <td>1</td>\n",
"       <td>0</td>\n",
"       <td>Unknown</td>\n",
"       <td>2</td>\n",
"       <td>1</td>\n",
"       <td>1.63</td>\n",
"       <td>44.0</td>\n",
"     </tr>\n",
"     <tr>\n",
"       <th>3</th>\n",
"       <td>5009749</td>\n",
"       <td>F</td>\n",
"       <td>Y</td>\n",
"       <td>N</td>\n",
"       <td>0</td>\n",
"       <td>166500.0</td>\n",
"       <td>Commercial associate</td>\n",
"       <td>Higher education</td>\n",
"       <td>Married</td>\n",
"       <td>House / apartment</td>\n",
"       <td>1</td>\n",
"       <td>1</td>\n",
              "      <td>1</td>\n",
              "      <td>0</td>\n",
              "      <td>Unknown</td>\n",
              "      <td>2</td>\n",
              "      <td>1</td>\n",
              "      <td>1.63</td>\n",
              "      <td>38.0</td>\n",
              "    </tr>\n",
              "    <tr>\n",
              "      <th>4</th>\n",
              "      <td>5009752</td>\n",
              "      <td>F</td>\n",
              "      <td>Y</td>\n",
              "      <td>N</td>\n",
              "      <td>0</td>\n",
              "      <td>315000.0</td>\n",
              "      <td>Commercial associate</td>\n",
              "      <td>Higher education</td>\n",
              "      <td>Married</td>\n",
              "      <td>House / apartment</td>\n",
              "      <td>1</td>\n",
              "      <td>1</td>\n",
              "      <td>1</td>\n",
              "      <td>0</td>\n",
              "      <td>Unknown</td>\n",
              "      <td>2</td>\n",
              "      <td>1</td>\n",
              "      <td>1.63</td>\n",
              "      <td>38.0</td>\n",
              "    </tr>\n",
              "  </tbody>\n",
              "</table>\n",
              "</div>\n",
              "    <div class=\"colab-df-buttons\">\n",
              "\n",
              "  <div class=\"colab-df-container\">\n",
              "    <button class=\"colab-df-convert\"
onclick=\"convertToInteractive('df-756f4a22-a538-4af8-a34a-ec8457bab32c')\"\n",
              "            title=\"Convert this dataframe to an interactive
table.\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\" viewBox=\"0
-960 960 960\">\n",
              "    <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220
220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-
160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",
              "  </svg>\n",
              "    </button>\n",
              "\n",
              "  <style>\n",
              "    .colab-df-container {\n",
              "      display:flex;\n",
              "      gap: 12px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert {\n",
              "      background-color: #E8F0FE;\n",
              "      border: none;\n",
              "      border-radius: 50%;\n",
              "      cursor: pointer;\n",
              "      display: none;\n",
              "      fill: #1967D2;\n",
              "      height: 32px;\n",
              "      padding: 0 0 0 0;\n",
              "      width: 32px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert:hover {\n",
              "      background-color: #E2EBFA;\n",
              "      box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n",
              "      fill: #174EA6;\n",
              "    }\n",
              "\n",
              "    .colab-df-buttons div {\n",
              "      margin-bottom: 4px;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert {\n",
              "      background-color: #3B4455;\n",
              "      fill: #D2E3FC;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert:hover {\n",
              "      background-color: #434B5C;\n",
              "      box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",
              "      filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",
              "      fill: #FFFFFF;\n",
              "    }\n",
              "  </style>\n",
              "\n",
              "    <script>\n",
              "      const buttonEl =\n",
              "        document.querySelector('#df-756f4a22-a538-4af8-a34a-
ec8457bab32c button.colab-df-convert');\n",
              "      buttonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "\n",
              "      async function convertToInteractive(key) {\n",
              "        const element = document.querySelector('#df-756f4a22-a538-4af8-
a34a-ec8457bab32c');\n",
              "        const dataTable =\n",
              "          await
google.colab.kernel.invokeFunction('convertToInteractive',\n",
              "                                                    [key], {});\n",
              "        if (!dataTable) return;\n",
              "\n",
              "        const docLinkHtml = 'Like what you see? Visit the ' +\n",
              "          '<a target=\"_blank\"
href=https://colab.research.google.com/notebooks/data_table.ipynb>data table
notebook</a>'\n",
              "          + ' to learn more about interactive tables.';\n",
              "        element.innerHTML = '';\n",
              "        dataTable['output_type'] = 'display_data';\n",
              "        await google.colab.output.renderOutput(dataTable, element);\n",
              "        const docLink = document.createElement('div');\n",
              "        docLink.innerHTML = docLinkHtml;\n",
              "        element.appendChild(docLink);\n",
              "      }\n",
              "    </script>\n",
              "  </div>\n",
              "\n",
              "\n",
              "<div id=\"df-a3f02811-cda3-4b88-9c1e-5242312cfbc8\">\n",
              "  <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-
a3f02811-cda3-4b88-9c1e-5242312cfbc8')\"\n",
              "            title=\"Suggest charts\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "<svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0 0
24 24\"\n",
              "     width=\"24px\">\n",
              "    <g>\n",
              "        <path d=\"M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-
.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\"/>\n",
              "    </g>\n",
              "</svg>\n",
              "  </button>\n",
              "\n",
              "<style>\n",
              "  .colab-df-quickchart {\n",
              "      --bg-color: #E8F0FE;\n",
              "      --fill-color: #1967D2;\n",
              "      --hover-bg-color: #E2EBFA;\n",
              "      --hover-fill-color: #174EA6;\n",
              "      --disabled-fill-color: #AAA;\n",
              ″      --disabled-bg-color: #DDD;\n″,
              ″  }\n″,
              ″\n″,
              ″  [theme=dark] .colab-df-quickchart {\n″,
              ″      --bg-color: #3B4455;\n″,
              ″      --fill-color: #D2E3FC;\n″,
              ″      --hover-bg-color: #434B5C;\n″,
              ″      --hover-fill-color: #FFFFFF;\n″,
              ″      --disabled-bg-color: #3B4455;\n″,
              ″      --disabled-fill-color: #666;\n″,
              ″  }\n″,
              ″\n″,
              ″  .colab-df-quickchart {\n″,
              ″    background-color: var(--bg-color);\n″,
              ″    border: none;\n″,
              ″    border-radius: 50%;\n″,
              ″    cursor: pointer;\n″,
              ″    display: none;\n″,
              ″    fill: var(--fill-color);\n″,
              ″    height: 32px;\n″,
              ″    padding: 0;\n″,
              ″    width: 32px;\n″,
              ″  }\n″,
              ″\n″,
              ″  .colab-df-quickchart:hover {\n″,
              ″    background-color: var(--hover-bg-color);\n″,
              ″    box-shadow: 0 1px 2px rgba(60, 64, 67, 0.3), 0 1px 3px 1px rgba(60,
64, 67, 0.15);\n″,
              ″    fill: var(--button-hover-fill-color);\n″,
              ″  }\n″,
              ″\n″,
              ″  .colab-df-quickchart-complete:disabled,\n″,
              ″  .colab-df-quickchart-complete:disabled:hover {\n″,
              ″    background-color: var(--disabled-bg-color);\n″,
              ″    fill: var(--disabled-fill-color);\n″,
              ″    box-shadow: none;\n″,
              ″  }\n″,
              ″\n″,
              ″  .colab-df-spinner {\n″,
              ″    border: 2px solid var(--fill-color);\n″,
              ″    border-color: transparent;\n″,
              ″    border-bottom-color: var(--fill-color);\n″,
              ″    animation:\n″,
              ″      spin 1s steps(1) infinite;\n″,
              ″  }\n″,
              ″\n″,
              ″  @keyframes spin {\n″,
              ″    0% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-bottom-color: var(--fill-color);\n″,
              ″      border-left-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    20% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-left-color: var(--fill-color);\n″,
              ″      border-top-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    30% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-left-color: var(--fill-color);\n″,
              ″      border-top-color: var(--fill-color);\n″,
              ″      border-right-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    40% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-right-color: var(--fill-color);\n″,
              ″      border-top-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    60% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-right-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    80% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-right-color: var(--fill-color);\n″,
              ″      border-bottom-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    90% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-bottom-color: var(--fill-color);\n″,
              ″    }\n″,
              ″  }\n″,
              ″</style>\n″,
              ″\n″,
              ″  <script>\n″,
              ″    async function quickchart(key) {\n″,
              ″      const quickchartButtonEl =\n″,
              ″        document.querySelector('#' + key + ' button');\n″,
              ″      quickchartButtonEl.disabled = true;  // To prevent multiple
clicks.\n″,
              ″      quickchartButtonEl.classList.add('colab-df-spinner');\n″,
              ″      try {\n″,
              ″        const charts = await google.colab.kernel.invokeFunction(\n″,
              ″            'suggestCharts', [key], {});\n″,
              ″      } catch (error) {\n″,
              ″        console.error('Error during call to suggestCharts:', error);\n″,
              ″      }\n″,
              ″      quickchartButtonEl.classList.remove('colab-df-spinner');\n″,
              "      quickchartButtonEl.classList.add('colab-df-quickchart-
complete');\n",
              "    }\n",
              "    (() => {\n",
              "      let quickchartButtonEl =\n",
              "        document.querySelector('#df-a3f02811-cda3-4b88-9c1e-
5242312cfbc8 button');\n",
              "      quickchartButtonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "    })();\n",
              "  </script>\n",
              "</div>\n",
              "\n",
              "    </div>\n",
              "  </div>\n"
            ]
          },
          "metadata": {},
          "execution_count": 1362
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "## Q1.\n",
        "Group the customers based on their income type and find the average of their
annual income.\n"
      ],
      "metadata": {
        "id": "gAu_kM6GUx08"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "query = \"SELECT Type_Income,AVG(Annual_income)  AS avg_annual_income FROM
credit_data GROUP BY Type_Income\"\n",
        "result = connection.execute(query).fetchdf()\n",
        "result"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 175
        },
        "id": "-f-2IXRSRGU2",
        "outputId": "5054be0f-81a1-4e6c-c27b-26aecbd7e679"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "            Type_Income  avg_annual_income\n",
              "0             Pensioner      155343.496283\n",
              "1               Working      180848.210526\n",
              "2         State servant      211422.413793\n",
              "3  Commercial associate      233107.397260"
            ],
            "text/html": [
              "\n",
              "  <div id=\"df-25d56e8e-e4a9-4406-aa68-e4bcd8d27568\" class=\"colab-df-
container\">\n",
              "    <div>\n",
              "<style scoped>\n",
              "    .dataframe tbody tr th:only-of-type {\n",
              "        vertical-align: middle;\n",
              "    }\n",
              "\n",
              "    .dataframe tbody tr th {\n",
              "        vertical-align: top;\n",
              "    }\n",
              "\n",
              "    .dataframe thead th {\n",
              "        text-align: right;\n",
              "    }\n",
              "</style>\n",
              "<table border=\"1\" class=\"dataframe\">\n",
              "  <thead>\n",
              "    <tr style=\"text-align: right;\">\n",
              "      <th></th>\n",
              "      <th>Type_Income</th>\n",
              "      <th>avg_annual_income</th>\n",
              "    </tr>\n",
              "  </thead>\n",
              "  <tbody>\n",
              "    <tr>\n",
              "      <th>0</th>\n",
              "      <td>Pensioner</td>\n",
              "      <td>155343.496283</td>\n",
              "    </tr>\n",
              "    <tr>\n",
              "      <th>1</th>\n",
              "      <td>Working</td>\n",
              "      <td>180848.210526</td>\n",
              "    </tr>\n",
              "    <tr>\n",
              "      <th>2</th>\n",
              "      <td>State servant</td>\n",
              "      <td>211422.413793</td>\n",
              "    </tr>\n",
              "    <tr>\n",
              "      <th>3</th>\n",
              "      <td>Commercial associate</td>\n",
              "      <td>233107.397260</td>\n",
              "    </tr>\n",
              "  </tbody>\n",
              "</table>\n",
              "</div>\n",
              "    <div class=\"colab-df-buttons\">\n",
              "\n",
              "  <div class=\"colab-df-container\">\n",
              "    <button class=\"colab-df-convert\"
onclick=\"convertToInteractive('df-25d56e8e-e4a9-4406-aa68-e4bcd8d27568')\"\n",
              "            title=\"Convert this dataframe to an interactive
table.\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\" viewBox=\"0
-960 960 960\">\n",
              "    <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220
220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-
160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",
              "  </svg>\n",
              "    </button>\n",
              "\n",
              "  <style>\n",
              "    .colab-df-container {\n",
              "      display:flex;\n",
              "      gap: 12px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert {\n",
              "      background-color: #E8F0FE;\n",
              "      border: none;\n",
              "      border-radius: 50%;\n",
              "      cursor: pointer;\n",
              "      display: none;\n",
              "      fill: #1967D2;\n",
              "      height: 32px;\n",
              "      padding: 0 0 0 0;\n",
              "      width: 32px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert:hover {\n",
              ″      background-color: #E2EBFA;\n″,
              ″      box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n″,
              ″      fill: #174EA6;\n″,
              ″    }\n″,
              ″\n″,
              ″    .colab-df-buttons div {\n″,
              ″      margin-bottom: 4px;\n″,
              ″    }\n″,
              ″\n″,
              ″    [theme=dark] .colab-df-convert {\n″,
              ″      background-color: #3B4455;\n″,
              ″      fill: #D2E3FC;\n″,
              ″    }\n″,
              ″\n″,
              ″    [theme=dark] .colab-df-convert:hover {\n″,
              ″      background-color: #434B5C;\n″,
              ″      box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n″,
              ″      filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n″,
              ″      fill: #FFFFFF;\n″,
              ″    }\n″,
              ″  </style>\n″,
              ″\n″,
              ″    <script>\n″,
              ″      const buttonEl =\n″,
              ″        document.querySelector('#df-25d56e8e-e4a9-4406-aa68-
e4bcd8d27568 button.colab-df-convert');\n″,
              ″      buttonEl.style.display =\n″,
              ″        google.colab.kernel.accessAllowed ? 'block' : 'none';\n″,
              ″\n″,
              ″      async function convertToInteractive(key) {\n″,
              ″        const element = document.querySelector('#df-25d56e8e-e4a9-4406-
aa68-e4bcd8d27568');\n″,
              ″        const dataTable =\n″,
              ″          await
google.colab.kernel.invokeFunction('convertToInteractive',\n″,
              ″                                                    [key], {});\n″,
              ″        if (!dataTable) return;\n″,
              ″\n″,
              ″        const docLinkHtml = 'Like what you see? Visit the ' +\n″,
              ″          '<a target=\″_blank\″
href=https://colab.research.google.com/notebooks/data_table.ipynb>data table
notebook</a>'\n″,
              ″          + ' to learn more about interactive tables.';\n″,
              ″        element.innerHTML = '';\n″,
              ″        dataTable['output_type'] = 'display_data';\n″,
              ″        await google.colab.output.renderOutput(dataTable, element);\n″,
              ″        const docLink = document.createElement('div');\n″,
              ″        docLink.innerHTML = docLinkHtml;\n″,
              "        element.appendChild(docLink);\n",
              "      }\n",
              "    </script>\n",
              "  </div>\n",
              "\n",
              "\n",
              "<div id=\"df-402bda93-2a7a-4091-a92b-d066e6b59c4a\">\n",
              "  <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-
402bda93-2a7a-4091-a92b-d066e6b59c4a')\"\n",
              "            title=\"Suggest charts\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "<svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0 0
24 24\"\n",
              "     width=\"24px\">\n",
              "    <g>\n",
              "        <path d=\"M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-
.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\"/>\n",
              "    </g>\n",
              "</svg>\n",
              "  </button>\n",
              "\n",
              "<style>\n",
              "  .colab-df-quickchart {\n",
              "      --bg-color: #E8F0FE;\n",
              "      --fill-color: #1967D2;\n",
              "      --hover-bg-color: #E2EBFA;\n",
              "      --hover-fill-color: #174EA6;\n",
              "      --disabled-fill-color: #AAA;\n",
              "      --disabled-bg-color: #DDD;\n",
              "  }\n",
              "\n",
              "  [theme=dark] .colab-df-quickchart {\n",
              "      --bg-color: #3B4455;\n",
              "      --fill-color: #D2E3FC;\n",
              "      --hover-bg-color: #434B5C;\n",
              "      --hover-fill-color: #FFFFFF;\n",
              "      --disabled-bg-color: #3B4455;\n",
              "      --disabled-fill-color: #666;\n",
              "  }\n",
              "\n",
              "  .colab-df-quickchart {\n",
              "    background-color: var(--bg-color);\n",
              "    border: none;\n",
              "    border-radius: 50%;\n",
              "    cursor: pointer;\n",
              "    display: none;\n",
              "    fill: var(--fill-color);\n",
              "    height: 32px;\n",
            "    padding: 0;\n",
            "    width: 32px;\n",
            "  }\n",
            "\n",
            "  .colab-df-quickchart:hover {\n",
            "    background-color: var(--hover-bg-color);\n",
            "    box-shadow: 0 1px 2px rgba(60, 64, 67, 0.3), 0 1px 3px 1px rgba(60,
64, 67, 0.15);\n",
            "    fill: var(--button-hover-fill-color);\n",
            "  }\n",
            "\n",
            "  .colab-df-quickchart-complete:disabled,\n",
            "  .colab-df-quickchart-complete:disabled:hover {\n",
            "    background-color: var(--disabled-bg-color);\n",
            "    fill: var(--disabled-fill-color);\n",
            "    box-shadow: none;\n",
            "  }\n",
            "\n",
            "  .colab-df-spinner {\n",
            "    border: 2px solid var(--fill-color);\n",
            "    border-color: transparent;\n",
            "    border-bottom-color: var(--fill-color);\n",
            "    animation:\n",
            "      spin 1s steps(1) infinite;\n",
            "  }\n",
            "\n",
            "  @keyframes spin {\n",
            "    0% {\n",
            "      border-color: transparent;\n",
            "      border-bottom-color: var(--fill-color);\n",
            "      border-left-color: var(--fill-color);\n",
            "    }\n",
            "    20% {\n",
            "      border-color: transparent;\n",
            "      border-left-color: var(--fill-color);\n",
            "      border-top-color: var(--fill-color);\n",
            "    }\n",
            "    30% {\n",
            "      border-color: transparent;\n",
            "      border-left-color: var(--fill-color);\n",
            "      border-top-color: var(--fill-color);\n",
            "      border-right-color: var(--fill-color);\n",
            "    }\n",
            "    40% {\n",
            "      border-color: transparent;\n",
            "      border-right-color: var(--fill-color);\n",
            "      border-top-color: var(--fill-color);\n",
            "    }\n",
            "    60% {\n",
″      border-color: transparent;\n″,
″      border-right-color: var(--fill-color);\n″,
″    }\n″,
″    80% {\n″,
″      border-color: transparent;\n″,
″      border-right-color: var(--fill-color);\n″,
″      border-bottom-color: var(--fill-color);\n″,
″    }\n″,
″    90% {\n″,
″      border-color: transparent;\n″,
″      border-bottom-color: var(--fill-color);\n″,
″    }\n″,
″  }\n″,
″</style>\n″,
″\n″,
″  <script>\n″,
″    async function quickchart(key) {\n″,
″      const quickchartButtonEl =\n″,
″        document.querySelector(’#’ + key + ’ button’);\n″,
″      quickchartButtonEl.disabled = true;  // To prevent multiple
clicks.\n″,
″      quickchartButtonEl.classList.add(’colab-df-spinner’);\n″,
″      try {\n″,
″        const charts = await google.colab.kernel.invokeFunction(\n″,
″            ’suggestCharts’, [key], {});\n″,
″      } catch (error) {\n″,
″        console.error(’Error during call to suggestCharts:’, error);\n″,
″      }\n″,
″      quickchartButtonEl.classList.remove(’colab-df-spinner’);\n″,
″      quickchartButtonEl.classList.add(’colab-df-quickchart-
complete’);\n″,
″    }\n″,
″    (() => {\n″,
″      let quickchartButtonEl =\n″,
″        document.querySelector(’#df-402bda93-2a7a-4091-a92b-
d066e6b59c4a button’);\n″,
″      quickchartButtonEl.style.display =\n″,
″        google.colab.kernel.accessAllowed ? ’block’ : ’none’;\n″,
″    })();\n″,
″  </script>\n″,
″</div>\n″,
″\n″,
″  <div id=\″id_6f218864-6b58-43ec-8764-0bb119749bd7\″>\n″,
″    <style>\n″,
″      .colab-df-generate {\n″,
″        background-color: #E8F0FE;\n″,
″        border: none;\n″,
″        border-radius: 50%;\n″,
″        cursor: pointer;\n″,
              ″        display: none;\n″,
              ″        fill: #1967D2;\n″,
              ″        height: 32px;\n″,
              ″        padding: 0 0 0 0;\n″,
              ″        width: 32px;\n″,
              ″      }\n″,
              ″\n″,
              ″      .colab-df-generate:hover {\n″,
              ″        background-color: #E2EBFA;\n″,
              ″        box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n″,
              ″        fill: #174EA6;\n″,
              ″      }\n″,
              ″\n″,
              ″      [theme=dark] .colab-df-generate {\n″,
              ″        background-color: #3B4455;\n″,
              ″        fill: #D2E3FC;\n″,
              ″      }\n″,
              ″\n″,
              ″      [theme=dark] .colab-df-generate:hover {\n″,
              ″        background-color: #434B5C;\n″,
              ″        box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n″,
              ″        filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n″,
              ″        fill: #FFFFFF;\n″,
              ″      }\n″,
              ″    </style>\n″,
              ″    <button class=\″colab-df-generate\″
onclick=\″generateWithVariable('result')\″\n″,
              ″            title=\″Generate code using this dataframe.\″\n″,
              ″            style=\″display:none;\″>\n″,
              ″\n″,
              ″  <svg xmlns=\″http://www.w3.org/2000/svg\″ height=\″24px\″viewBox=\″0
0 24 24\″\n″,
              ″       width=\″24px\″>\n″,
              ″    <path
d=\″M7,19H8.4L18.45,9,17,7.55,7,17.6ZM5,21V16.75L18.45,3.32a2,2,0,0,1,2.83,0l1.4,1.43a
1.91,1.91,0,0,1,.58,1.4,1.91,1.91,0,0,1-.58,1.4L9.25,21ZM18.45,9,17,7.55Zm-
12,3A5.31,5.31,0,0,0,4.9,8.1,5.31,5.31,0,0,0,1,6.5,5.31,5.31,0,0,0,4.9,4.9,5.31,5.31,0,
0,0,6.5,1,5.31,5.31,0,0,0,8.1,4.9,5.31,5.31,0,0,0,12,6.5,5.46,5.46,0,0,0,6.5,12Z\″/>\n
″,
              ″  </svg>\n″,
              ″    </button>\n″,
              ″    <script>\n″,
              ″      (() => {\n″,
              ″      const buttonEl =\n″,
              ″        document.querySelector('#id_6f218864-6b58-43ec-8764-
0bb119749bd7 button.colab-df-generate');\n″,
              ″      buttonEl.style.display =\n″,
              ″        google.colab.kernel.accessAllowed ? 'block' : 'none';\n″,
              "\n",
              "      buttonEl.onclick = () => {\n",
              "        google.colab.notebook.generateWithVariable('result');\n",
              "      }\n",
              "      })();\n",
              "    </script>\n",
              "  </div>\n",
              "\n",
              "    </div>\n",
              "  </div>\n"
            ]
          },
          "metadata": {},
          "execution_count": 1363
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "## Q2.\n",
        " Find the female owners of cars and property.\n"
      ],
      "metadata": {
        "id": "sZyqToNdVBRi"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "query = \"SELECT * FROM credit_data WHERE (GENDER = 'F') AND (Car_Owner = 'Y')
AND (Propert_Owner = 'Y')\"\n",
        "result = connection.execute(query).fetchdf()\n",
        "result"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 790
        },
        "id": "iGDMn0vlVj3o",
        "outputId": "843c7eaa-962d-472d-efa9-c758932ba36d"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              ″      Ind_ID GENDER Car_Owner Propert_Owner  CHILDREN  Annual_income
\\\n″,
              ″0    5018498      F         Y             Y         0        90000.0
\n″,
              ″1    5018501      F         Y             Y         0       166500.0
\n″,
              ″2    5018503      F         Y             Y         0        90000.0
\n″,
              ″3    5024213      F         Y             Y         0       540000.0
\n″,
              ″4    5036660      F         Y             Y         0        76500.0
\n″,
              ″..       ...    ...       ...           ...       ...            ...
\n″,
              ″174  5048458      F         Y             Y         1       126000.0
\n″,
              ″175  5023719      F         Y             Y         0       175500.0
\n″,
              ″176  5033520      F         Y             Y         3       180000.0
\n″,
              ″177  5024049      F         Y             Y         1       144000.0
\n″,
              ″178  5053790      F         Y             Y         0       225000.0
\n″,
              ″\n″,
              ″            Type_Income                      EDUCATION Marital_status
\\\n″,
              ″0                Working  Secondary / secondary special        Married
\n″,
              ″1                Working  Secondary / secondary special        Married
\n″,
              ″2                Working  Secondary / secondary special        Married
\n″,
              ″3    Commercial associate               Higher education        Married
\n″,
              ″4              Pensioner  Secondary / secondary special        Married
\n″,
              ″..                   ...                            ...            ...
\n″,
              ″174              Working               Higher education        Married
\n″,
              ″175            Pensioner               Higher education        Married
\n″,
              ″176              Working  Secondary / secondary special        Married
\n″,
              ″177              Working               Higher education        Married
\n″,
              ″178              Working               Higher education        Married
\n″,
              ″\n″,
              ″            Housing_type  Mobile_phone  Work_Phone  Phone  EMAIL_ID
\\\n″,
              ″0      House / apartment             1           1      1         0
\n″,
              ″1      House / apartment             1           1      1         0
\n″,
              ″2      House / apartment             1           1      1         0
\n″,
              ″3      House / apartment             1           0      1         0
\n″,
              ″4      House / apartment             1           0      0         0
\n″,
              ″..                   ...           ...         ...    ...       ...
\n″,
              ″174    House / apartment             1           0      0         0
\n″,
              ″175    House / apartment             1           0      1         0
\n″,
              ″176  Municipal apartment             1           0      1         0
\n″,
              ″177    House / apartment             1           0      0         0
\n″,
              ″178    House / apartment             1           0      0         0
\n″,
              ″\n″,
              ″    Type_Occupation  Family_Members  label  Employed_years   Age  \n″,
              ″0     Cooking staff               2      1            2.78  53.0  \n″,
              ″1     Cooking staff               2      1            2.78  53.0  \n″,
              ″2     Cooking staff               2      1            2.78  53.0  \n″,
              ″3           Unknown               2      1            0.51  44.0  \n″,
              ″4        Unemployed               2      1            0.00  66.0  \n″,
              ″..              ...             ...    ...             ...   ...  \n″,
              ″174         Unknown               3      0            0.44  26.0  \n″,
              ″175      Unemployed               2      0            0.00  61.0  \n″,
              ″176  Medicine staff               5      0           16.28  35.0  \n″,
              ″177     Accountants               3      0            8.12  36.0  \n″,
              ″178         Unknown               2      0            7.94  46.0  \n″,
              ″\n″,
              ″[179 rows x 19 columns]″
            ],
            ″text/html″: [
              ″\n″,
              ″  <div id=\″df-fc610353-baec-41d1-9243-4cb3090462c2\″ class=\″colab-df-
container\″>\n″,
              ″    <div>\n″,
              ″<style scoped>\n″,
              ″    .dataframe tbody tr th:only-of-type {\n″,
              ″        vertical-align: middle;\n″,
"    }\n",
"\n",
"    .dataframe tbody tr th {\n",
"        vertical-align: top;\n",
"    }\n",
"\n",
"    .dataframe thead th {\n",
"        text-align: right;\n",
"    }\n",
"</style>\n",
"<table border=\"1\" class=\"dataframe\">\n",
"  <thead>\n",
"    <tr style=\"text-align: right;\">\n",
"      <th></th>\n",
"      <th>Ind_ID</th>\n",
"      <th>GENDER</th>\n",
"      <th>Car_Owner</th>\n",
"      <th>Propert_Owner</th>\n",
"      <th>CHILDREN</th>\n",
"      <th>Annual_income</th>\n",
"      <th>Type_Income</th>\n",
"      <th>EDUCATION</th>\n",
"      <th>Marital_status</th>\n",
"      <th>Housing_type</th>\n",
"      <th>Mobile_phone</th>\n",
"      <th>Work_Phone</th>\n",
"      <th>Phone</th>\n",
"      <th>EMAIL_ID</th>\n",
"      <th>Type_Occupation</th>\n",
"      <th>Family_Members</th>\n",
"      <th>label</th>\n",
"      <th>Employed_years</th>\n",
"      <th>Age</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>0</th>\n",
"      <td>5018498</td>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>0</td>\n",
"      <td>90000.0</td>\n",
"      <td>Working</td>\n",
"      <td>Secondary / secondary special</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>Cooking staff</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>2.78</td>\n",
"      <td>53.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>1</th>\n",
"      <td>5018501</td>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>0</td>\n",
"      <td>166500.0</td>\n",
"      <td>Working</td>\n",
"      <td>Secondary / secondary special</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>Cooking staff</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>2.78</td>\n",
"      <td>53.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>2</th>\n",
"      <td>5018503</td>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>0</td>\n",
"      <td>90000.0</td>\n",
"      <td>Working</td>\n",
"      <td>Secondary / secondary special</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>Cooking staff</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>2.78</td>\n",
"      <td>53.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>3</th>\n",
"      <td>5024213</td>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>0</td>\n",
"      <td>540000.0</td>\n",
"      <td>Commercial associate</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>Unknown</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>0.51</td>\n",
"      <td>44.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>4</th>\n",
"      <td>5036660</td>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>0</td>\n",
"      <td>76500.0</td>\n",
"      <td>Pensioner</td>\n",
"      <td>Secondary / secondary special</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>Unemployed</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>0.00</td>\n",
"      <td>66.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>...</th>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"      <td>...</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>174</th>\n",
"      <td>5048458</td>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>1</td>\n",
"      <td>126000.0</td>\n",
"      <td>Working</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>Unknown</td>\n",
"      <td>3</td>\n",
"      <td>0</td>\n",
"      <td>0.44</td>\n",
"      <td>26.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>175</th>\n",
"      <td>5023719</td>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
″      <td>0</td>\n″,
″      <td>175500.0</td>\n″,
″      <td>Pensioner</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
″      <td>House / apartment</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>Unemployed</td>\n″,
″      <td>2</td>\n″,
″      <td>0</td>\n″,
″      <td>0.00</td>\n″,
″      <td>61.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>176</th>\n″,
″      <td>5033520</td>\n″,
″      <td>F</td>\n″,
″      <td>Y</td>\n″,
″      <td>Y</td>\n″,
″      <td>3</td>\n″,
″      <td>180000.0</td>\n″,
″      <td>Working</td>\n″,
″      <td>Secondary / secondary special</td>\n″,
″      <td>Married</td>\n″,
″      <td>Municipal apartment</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>Medicine staff</td>\n″,
″      <td>5</td>\n″,
″      <td>0</td>\n″,
″      <td>16.28</td>\n″,
″      <td>35.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>177</th>\n″,
″      <td>5024049</td>\n″,
″      <td>F</td>\n″,
″      <td>Y</td>\n″,
″      <td>Y</td>\n″,
″      <td>1</td>\n″,
″      <td>144000.0</td>\n″,
″      <td>Working</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
              "      <td>House / apartment</td>\n",
              "      <td>1</td>\n",
              "      <td>0</td>\n",
              "      <td>0</td>\n",
              "      <td>0</td>\n",
              "      <td>Accountants</td>\n",
              "      <td>3</td>\n",
              "      <td>0</td>\n",
              "      <td>8.12</td>\n",
              "      <td>36.0</td>\n",
              "    </tr>\n",
              "    <tr>\n",
              "      <th>178</th>\n",
              "      <td>5053790</td>\n",
              "      <td>F</td>\n",
              "      <td>Y</td>\n",
              "      <td>Y</td>\n",
              "      <td>0</td>\n",
              "      <td>225000.0</td>\n",
              "      <td>Working</td>\n",
              "      <td>Higher education</td>\n",
              "      <td>Married</td>\n",
              "      <td>House / apartment</td>\n",
              "      <td>1</td>\n",
              "      <td>0</td>\n",
              "      <td>0</td>\n",
              "      <td>0</td>\n",
              "      <td>Unknown</td>\n",
              "      <td>2</td>\n",
              "      <td>0</td>\n",
              "      <td>7.94</td>\n",
              "      <td>46.0</td>\n",
              "    </tr>\n",
              "  </tbody>\n",
              "</table>\n",
              "<p>179 rows 脳 19 columns</p>\n",
              "</div>\n",
              "    <div class=\"colab-df-buttons\">\n",
              "\n",
              "  <div class=\"colab-df-container\">\n",
              "    <button class=\"colab-df-convert\"
onclick=\"convertToInteractive('df-fc610353-baec-41d1-9243-4cb3090462c2')\"\n",
              "            title=\"Convert this dataframe to an interactive
table.\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\" viewBox=\"0
-960 960 960\">\n",
              "    <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220
220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-
160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",
              "  </svg>\n",
              "    </button>\n",
              "\n",
              "  <style>\n",
              "    .colab-df-container {\n",
              "      display:flex;\n",
              "      gap: 12px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert {\n",
              "      background-color: #E8F0FE;\n",
              "      border: none;\n",
              "      border-radius: 50%;\n",
              "      cursor: pointer;\n",
              "      display: none;\n",
              "      fill: #1967D2;\n",
              "      height: 32px;\n",
              "      padding: 0 0 0 0;\n",
              "      width: 32px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert:hover {\n",
              "      background-color: #E2EBFA;\n",
              "      box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n",
              "      fill: #174EA6;\n",
              "    }\n",
              "\n",
              "    .colab-df-buttons div {\n",
              "      margin-bottom: 4px;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert {\n",
              "      background-color: #3B4455;\n",
              "      fill: #D2E3FC;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert:hover {\n",
              "      background-color: #434B5C;\n",
              "      box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",
              "      filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",
              "      fill: #FFFFFF;\n",
              "    }\n",
              "  </style>\n",
              "\n",
              "    <script>\n",
             "      const buttonEl =\n",
             "        document.querySelector('#df-fc610353-baec-41d1-9243-
4cb3090462c2 button.colab-df-convert');\n",
             "      buttonEl.style.display =\n",
             "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
             "\n",
             "      async function convertToInteractive(key) {\n",
             "        const element = document.querySelector('#df-fc610353-baec-41d1-
9243-4cb3090462c2');\n",
             "        const dataTable =\n",
             "          await
google.colab.kernel.invokeFunction('convertToInteractive',\n",
             "                                                    [key], {});\n",
             "        if (!dataTable) return;\n",
             "\n",
             "        const docLinkHtml = 'Like what you see? Visit the ' +\n",
             "          '<a target=\"_blank\"
href=https://colab.research.google.com/notebooks/data_table.ipynb>data table
notebook</a>'\n",
             "          + ' to learn more about interactive tables.';\n",
             "        element.innerHTML = '';\n",
             "        dataTable['output_type'] = 'display_data';\n",
             "        await google.colab.output.renderOutput(dataTable, element);\n",
             "        const docLink = document.createElement('div');\n",
             "        docLink.innerHTML = docLinkHtml;\n",
             "        element.appendChild(docLink);\n",
             "      }\n",
             "    </script>\n",
             "  </div>\n",
             "\n",
             "\n",
             "<div id=\"df-72f4a262-b342-43c6-89b4-7ef453c61852\">\n",
             "  <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-
72f4a262-b342-43c6-89b4-7ef453c61852')\"\n",
             "            title=\"Suggest charts\"\n",
             "            style=\"display:none;\">\n",
             "\n",
             "<svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0 0
24 24\"\n",
             "     width=\"24px\">\n",
             "    <g>\n",
             "        <path d=\"M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-
.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\"/>\n",
             "    </g>\n",
             "</svg>\n",
             "  </button>\n",
             "\n",
             "<style>\n",
             "  .colab-df-quickchart {\n",
             "      --bg-color: #E8F0FE;\n",
             "      --fill-color: #1967D2;\n",
             "      --hover-bg-color: #E2EBFA;\n",
             "      --hover-fill-color: #174EA6;\n",
             "      --disabled-fill-color: #AAA;\n",
             "      --disabled-bg-color: #DDD;\n",
             "  }\n",
             "\n",
             "  [theme=dark] .colab-df-quickchart {\n",
             "      --bg-color: #3B4455;\n",
             "      --fill-color: #D2E3FC;\n",
             "      --hover-bg-color: #434B5C;\n",
             "      --hover-fill-color: #FFFFFF;\n",
             "      --disabled-bg-color: #3B4455;\n",
             "      --disabled-fill-color: #666;\n",
             "  }\n",
             "\n",
             "  .colab-df-quickchart {\n",
             "    background-color: var(--bg-color);\n",
             "    border: none;\n",
             "    border-radius: 50%;\n",
             "    cursor: pointer;\n",
             "    display: none;\n",
             "    fill: var(--fill-color);\n",
             "    height: 32px;\n",
             "    padding: 0;\n",
             "    width: 32px;\n",
             "  }\n",
             "\n",
             "  .colab-df-quickchart:hover {\n",
             "    background-color: var(--hover-bg-color);\n",
             "    box-shadow: 0 1px 2px rgba(60, 64, 67, 0.3), 0 1px 3px 1px rgba(60,
64, 67, 0.15);\n",
             "    fill: var(--button-hover-fill-color);\n",
             "  }\n",
             "\n",
             "  .colab-df-quickchart-complete:disabled,\n",
             "  .colab-df-quickchart-complete:disabled:hover {\n",
             "    background-color: var(--disabled-bg-color);\n",
             "    fill: var(--disabled-fill-color);\n",
             "    box-shadow: none;\n",
             "  }\n",
             "\n",
             "  .colab-df-spinner {\n",
             "    border: 2px solid var(--fill-color);\n",
             "    border-color: transparent;\n",
             "    border-bottom-color: var(--fill-color);\n",
             "    animation:\n",
             "      spin 1s steps(1) infinite;\n",
              "  }\n",
              "\n",
              "  @keyframes spin {\n",
              "    0% {\n",
              "      border-color: transparent;\n",
              "      border-bottom-color: var(--fill-color);\n",
              "      border-left-color: var(--fill-color);\n",
              "    }\n",
              "    20% {\n",
              "      border-color: transparent;\n",
              "      border-left-color: var(--fill-color);\n",
              "      border-top-color: var(--fill-color);\n",
              "    }\n",
              "    30% {\n",
              "      border-color: transparent;\n",
              "      border-left-color: var(--fill-color);\n",
              "      border-top-color: var(--fill-color);\n",
              "      border-right-color: var(--fill-color);\n",
              "    }\n",
              "    40% {\n",
              "      border-color: transparent;\n",
              "      border-right-color: var(--fill-color);\n",
              "      border-top-color: var(--fill-color);\n",
              "    }\n",
              "    60% {\n",
              "      border-color: transparent;\n",
              "      border-right-color: var(--fill-color);\n",
              "    }\n",
              "    80% {\n",
              "      border-color: transparent;\n",
              "      border-right-color: var(--fill-color);\n",
              "      border-bottom-color: var(--fill-color);\n",
              "    }\n",
              "    90% {\n",
              "      border-color: transparent;\n",
              "      border-bottom-color: var(--fill-color);\n",
              "    }\n",
              "  }\n",
              "</style>\n",
              "\n",
              "  <script>\n",
              "    async function quickchart(key) {\n",
              "      const quickchartButtonEl =\n",
              "        document.querySelector('#' + key + ' button');\n",
              "      quickchartButtonEl.disabled = true;  // To prevent multiple
clicks.\n",
              "      quickchartButtonEl.classList.add('colab-df-spinner');\n",
              "      try {\n",
              "        const charts = await google.colab.kernel.invokeFunction(\n",
              "            'suggestCharts', [key], {});\n",
              "      } catch (error) {\n",
              "        console.error('Error during call to suggestCharts:', error);\n",
              "      }\n",
              "      quickchartButtonEl.classList.remove('colab-df-spinner');\n",
              "      quickchartButtonEl.classList.add('colab-df-quickchart-
complete');\n",
              "    }\n",
              "    (() => {\n",
              "      let quickchartButtonEl =\n",
              "        document.querySelector('#df-72f4a262-b342-43c6-89b4-
7ef453c61852 button');\n",
              "      quickchartButtonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "    })();\n",
              "  </script>\n",
              "</div>\n",
              "\n",
              "  <div id=\"id_4406f2bf-1a90-455e-a363-bd29c9e1d2e3\">\n",
              "    <style>\n",
              "      .colab-df-generate {\n",
              "        background-color: #E8F0FE;\n",
              "        border: none;\n",
              "        border-radius: 50%;\n",
              "        cursor: pointer;\n",
              "        display: none;\n",
              "        fill: #1967D2;\n",
              "        height: 32px;\n",
              "        padding: 0 0 0 0;\n",
              "        width: 32px;\n",
              "      }\n",
              "\n",
              "      .colab-df-generate:hover {\n",
              "        background-color: #E2EBFA;\n",
              "        box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n",
              "        fill: #174EA6;\n",
              "      }\n",
              "\n",
              "      [theme=dark] .colab-df-generate {\n",
              "        background-color: #3B4455;\n",
              "        fill: #D2E3FC;\n",
              "      }\n",
              "\n",
              "      [theme=dark] .colab-df-generate:hover {\n",
              "        background-color: #434B5C;\n",
              "        box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",
              "        filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",
              "        fill: #FFFFFF;\n",
       "      }\n",
       "    </style>\n",
       "    <button class=\"colab-df-generate\"
onclick=\"generateWithVariable('result')\"\n",
       "            title=\"Generate code using this dataframe.\"\n",
       "            style=\"display:none;\">\n",
       "\n",
       "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0
0 24 24\"\n",
       "       width=\"24px\">\n",
       "    <path
d=\"M7,19H8.4L18.45,9,17,7.55,7,17.6ZM5,21V16.75L18.45,3.32a2,2,0,0,1,2.83,0l1.4,1.43a
1.91,1.91,0,0,1,.58,1.4,1.91,1.91,0,0,1-.58,1.4L9.25,21ZM18.45,9,17,7.55Zm-
12,3A5.31,5.31,0,0,0,4.9,8.1,5.31,5.31,0,0,0,1,6.5,5.31,5.31,0,0,0,4.9,4.9,5.31,5.31,0,
0,0,6.5,1,5.31,5.31,0,0,0,8.1,4.9,5.31,5.31,0,0,0,12,6.5,5.46,5.46,0,0,0,6.5,12Z\"/>\n
",
       "  </svg>\n",
       "    </button>\n",
       "    <script>\n",
       "      (() => {\n",
       "      const buttonEl =\n",
       "        document.querySelector('#id_4406f2bf-1a90-455e-a363-
bd29c9e1d2e3 button.colab-df-generate');\n",
       "      buttonEl.style.display =\n",
       "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
       "\n",
       "      buttonEl.onclick = () => {\n",
       "        google.colab.notebook.generateWithVariable('result');\n",
       "      }\n",
       "      })();\n",
       "    </script>\n",
       "  </div>\n",
       "\n",
       "    </div>\n",
       "  </div>\n"
      ]
     },
     "metadata": {},
     "execution_count": 1364
    }
   ]
  },
  {
   "cell_type": "markdown",
   "source": [
    "## Q3.\n",
    " Find the male customers who are staying with their families.\n"
   ],
   "metadata": {
        "id": "RNZKcAR7Vg8P"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "query = \"SELECT * FROM credit_data WHERE Housing_type = 'With parents' AND
GENDER = 'M'\"\n",
        "result = connection.execute(query).fetchdf()\n",
        "result"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 1000
        },
        "id": "rUf8x_MsWwtH",
        "outputId": "e711a026-453a-43a5-a49a-84e3a41876f9"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "      Ind_ID GENDER Car_Owner Propert_Owner  CHILDREN  Annual_income
\\\n",
              "0   5021303      M         N             N         1       472500.0
\n",
              "1   5079166      M         Y             N         0       202500.0
\n",
              "2   5079167      M         Y             N         0       202500.0
\n",
              "3   5079168      M         Y             N         0       202500.0
\n",
              "4   5050729      M         Y             N         0       180000.0
\n",
              "5   5028383      M         Y             N         0       337500.0
\n",
              "6   5143019      M         N             N         0       270000.0
\n",
              "7   5067982      M         Y             Y         0       153000.0
\n",
              "8   5143573      M         Y             N         0       157500.0
\n",
              "9   5038751      M         Y             Y         2       157500.0
\n",
              "10  5010203      M         Y             Y         0       135000.0
\n",
            ″11  5058267    M      N         N      1      225000.0
\n″,
            ″12  5024352    M      Y         Y      0      585000.0
\n″,
            ″13  5150038    M      N         N      0      180000.0
\n″,
            ″14  5113302    M      N         Y      1      180000.0
\n″,
            ″15  5126311    M      N         N      0      180000.0
\n″,
            ″16  5126751    M      Y         N      2      135000.0
\n″,
            ″17  5037125    M      Y         N      0      180000.0
\n″,
            ″18  5054414    M      N         N      0      225000.0
\n″,
            ″19  5145694    M      Y         Y      0      225000.0
\n″,
            ″20  5146429    M      Y         N      0      261000.0
\n″,
            ″21  5054407    M      N         N      0      225000.0
\n″,
            ″22  5068648    M      N         Y      0      135000.0
\n″,
            ″23  5143560    M      N         N      2      202500.0
\n″,
            ″24  5125542    M      N         N      0      180000.0
\n″,
            ″25  5142781    M      Y         N      2      157500.0
\n″,
            ″26  5024592    M      Y         Y      0      157500.0
\n″,
            ″27  5094884    M      N         Y      0      450000.0
\n″,
            ″28  5033496    M      Y         N      1      202500.0
\n″,
            ″29  5067203    M      N         N      1      405000.0
\n″,
            ″30  5045537    M      Y         N      2      225000.0
\n″,
            ″31  5009269    M      Y         N      0      337500.0
\n″,
            ″32  5069007    M      Y         Y      0      337500.0
\n″,
            ″33  5066915    M      Y         Y      0      180000.0
\n″,
            ″34  5143025    M      N         N      0      270000.0
\n″,
              "35  5010202      M         Y             Y         0       135000.0
\n",
              "36  5028612      M         Y             Y         0       225000.0
\n",
              "\n",
              "            Type_Income                      EDUCATION
Marital_status  \\\n",
              "0             Pensioner               Higher education
Married   \n",
              "1               Working  Secondary / secondary special
Married   \n",
              "2               Working  Secondary / secondary special
Married   \n",
              "3               Working  Secondary / secondary special
Married   \n",
              "4               Working  Secondary / secondary special  Single / not
married   \n",
              "5               Working  Secondary / secondary special  Single / not
married   \n",
              "6               Working  Secondary / secondary special
Married   \n",
              "7               Working               Higher education
Separated   \n",
              "8               Working              Incomplete higher
Married   \n",
              "9   Commercial associate               Higher education
Married   \n",
              "10              Working                Lower secondary
Married   \n",
              "11  Commercial associate  Secondary / secondary special
Married   \n",
              "12  Commercial associate  Secondary / secondary special
Married   \n",
              "13              Working  Secondary / secondary special  Single / not
married   \n",
              "14              Working               Higher education
Married   \n",
              "15              Working              Incomplete higher        Civil
marriage   \n",
              "16              Working  Secondary / secondary special
Married   \n",
              "17              Working               Higher education
Married   \n",
              "18  Commercial associate               Higher education
Married   \n",
              "19              Working               Higher education  Single / not
married   \n",
              "20  Commercial associate              Incomplete higher        Civil
marriage   \n",
              ″21  Commercial associate               Higher education
Married   \n″,
              ″22               Working  Secondary / secondary special  Single / not
married   \n″,
              ″23               Working  Secondary / secondary special
Married   \n″,
              ″24               Working               Higher education
Married   \n″,
              ″25               Working  Secondary / secondary special
Separated   \n″,
              ″26               Working  Secondary / secondary special
Married   \n″,
              ″27  Commercial associate  Secondary / secondary special  Single / not
married   \n″,
              ″28  Commercial associate               Higher education
Married   \n″,
              ″29               Working  Secondary / secondary special
Married   \n″,
              ″30         State servant               Higher education
Married   \n″,
              ″31         State servant               Higher education
Married   \n″,
              ″32               Working               Higher education
Married   \n″,
              ″33               Working               Higher education
Married   \n″,
              ″34               Working  Secondary / secondary special
Married   \n″,
              ″35               Working                Lower secondary
Married   \n″,
              ″36  Commercial associate               Higher education  Single / not
married   \n″,
              ″\n″,
              ″    Housing_type  Mobile_phone  Work_Phone  Phone  EMAIL_ID  \\\n″,
              ″0   With parents             1           0      0         1   \n″,
              ″1   With parents             1           0      0         0   \n″,
              ″2   With parents             1           0      0         0   \n″,
              ″3   With parents             1           0      0         0   \n″,
              ″4   With parents             1           0      0         0   \n″,
              ″5   With parents             1           0      0         0   \n″,
              ″6   With parents             1           1      1         0   \n″,
              ″7   With parents             1           0      0         1   \n″,
              ″8   With parents             1           1      0         0   \n″,
              ″9   With parents             1           0      0         0   \n″,
              ″10  With parents             1           0      0         1   \n″,
              ″11  With parents             1           0      0         0   \n″,
              ″12  With parents             1           0      0         0   \n″,
              ″13  With parents             1           0      0         0   \n″,
              ″14  With parents             1           0      0         0   \n″,
        "15  With parents            1           1      0         1   \n",
        "16  With parents            1           0      0         0   \n",
        "17  With parents            1           0      1         0   \n",
        "18  With parents            1           1      0         0   \n",
        "19  With parents            1           1      1         0   \n",
        "20  With parents            1           0      0         0   \n",
        "21  With parents            1           1      0         0   \n",
        "22  With parents            1           0      0         0   \n",
        "23  With parents            1           0      0         0   \n",
        "24  With parents            1           1      1         0   \n",
        "25  With parents            1           0      0         0   \n",
        "26  With parents            1           1      0         0   \n",
        "27  With parents            1           0      0         0   \n",
        "28  With parents            1           0      0         0   \n",
        "29  With parents            1           0      0         0   \n",
        "30  With parents            1           0      0         0   \n",
        "31  With parents            1           1      1         0   \n",
        "32  With parents            1           0      0         0   \n",
        "33  With parents            1           0      0         0   \n",
        "34  With parents            1           1      1         0   \n",
        "35  With parents            1           0      0         1   \n",
        "36  With parents            1           0      1         0   \n",
        "\n",
        "          Type_Occupation  Family_Members  label  Employed_years   Age  
\n",
        "0                 Unknown               3      1            2.54  25.0  
\n",
        "1               Laborers                2      1            3.87  38.0  
\n",
        "2               Laborers                2      1            3.87  38.0  
\n",
        "3               Laborers                2      1            3.87  38.0  
\n",
        "4                Unknown                1      0            1.48  24.0  
\n",
        "5             Core staff                1      0            9.19  32.0  
\n",
        "6               Laborers                2      0            1.73  33.0  
\n",
        "7               Managers                1      0           21.22  56.0  
\n",
        "8                Drivers                2      0            2.67  25.0  
\n",
        "9                Drivers                4      0            1.74  39.0  
\n",
        "10              Laborers                2      0            5.18  28.0  
\n",
        "11               Drivers                3      0            5.31  50.0  
\n",
                ″12               Drivers              2      0             4.57  46.0
\n″,
                ″13               Drivers              1      0             4.27  36.0
\n″,
                ″14         Cooking staff              3      0             4.59  44.0
\n″,
                ″15            Core staff              2      0             0.80  21.0
\n″,
                ″16               Unknown              4      0             1.56  46.0
\n″,
                ″17            Core staff              2      0             6.19  32.0
\n″,
                ″18              Managers              2      0            14.12  36.0
\n″,
                ″19               Unknown              1      0             1.84  26.0
\n″,
                ″20           Sales staff              2      0             1.87  25.0
\n″,
                ″21              Managers              2      0            14.12  36.0
\n″,
                ″22               Drivers              1      0             2.93  32.0
\n″,
                ″23               Drivers              4      0             2.07  30.0
\n″,
                ″24  High skill tech staff             2      0             6.71  37.0
\n″,
                ″25              Laborers              3      0             4.52  39.0
\n″,
                ″26               Drivers              2      0             4.49  25.0
\n″,
                ″27              Managers              1      0             3.56  29.0
\n″,
                ″28              Managers              3      0             1.23  28.0
\n″,
                ″29               Drivers              3      0             3.69  35.0
\n″,
                ″30  High skill tech staff             4      0             4.28  30.0
\n″,
                ″31              Managers              2      0             4.59  29.0
\n″,
                ″32              Laborers              2      0             3.26  30.0
\n″,
                ″33              Laborers              2      0             4.08  31.0
\n″,
                ″34              Laborers              2      0             1.73  33.0
\n″,
                ″35              Laborers              2      0             5.18  28.0
\n″,
              "36            Sales staff               1      0            5.98  30.0
"
            ],
            "text/html": [
              "\n",
              "  <div id=\"df-86a17c24-b24c-45dc-ab6d-8e4b06b2ca98\" class=\"colab-df-
container\">\n",
              "    <div>\n",
              "<style scoped>\n",
              "    .dataframe tbody tr th:only-of-type {\n",
              "        vertical-align: middle;\n",
              "    }\n",
              "\n",
              "    .dataframe tbody tr th {\n",
              "        vertical-align: top;\n",
              "    }\n",
              "\n",
              "    .dataframe thead th {\n",
              "        text-align: right;\n",
              "    }\n",
              "</style>\n",
              "<table border=\"1\" class=\"dataframe\">\n",
              "  <thead>\n",
              "    <tr style=\"text-align: right;\">\n",
              "      <th></th>\n",
              "      <th>Ind_ID</th>\n",
              "      <th>GENDER</th>\n",
              "      <th>Car_Owner</th>\n",
              "      <th>Propert_Owner</th>\n",
              "      <th>CHILDREN</th>\n",
              "      <th>Annual_income</th>\n",
              "      <th>Type_Income</th>\n",
              "      <th>EDUCATION</th>\n",
              "      <th>Marital_status</th>\n",
              "      <th>Housing_type</th>\n",
              "      <th>Mobile_phone</th>\n",
              "      <th>Work_Phone</th>\n",
              "      <th>Phone</th>\n",
              "      <th>EMAIL_ID</th>\n",
              "      <th>Type_Occupation</th>\n",
              "      <th>Family_Members</th>\n",
              "      <th>label</th>\n",
              "      <th>Employed_years</th>\n",
              "      <th>Age</th>\n",
              "    </tr>\n",
              "  </thead>\n",
              "  <tbody>\n",
              "    <tr>\n",
              "      <th>0</th>\n",
″      <td>5021303</td>\n″,
″      <td>M</td>\n″,
″      <td>N</td>\n″,
″      <td>N</td>\n″,
″      <td>1</td>\n″,
″      <td>472500.0</td>\n″,
″      <td>Pensioner</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>Unknown</td>\n″,
″      <td>3</td>\n″,
″      <td>1</td>\n″,
″      <td>2.54</td>\n″,
″      <td>25.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>1</th>\n″,
″      <td>5079166</td>\n″,
″      <td>M</td>\n″,
″      <td>Y</td>\n″,
″      <td>N</td>\n″,
″      <td>0</td>\n″,
″      <td>202500.0</td>\n″,
″      <td>Working</td>\n″,
″      <td>Secondary / secondary special</td>\n″,
″      <td>Married</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Laborers</td>\n″,
″      <td>2</td>\n″,
″      <td>1</td>\n″,
″      <td>3.87</td>\n″,
″      <td>38.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>2</th>\n″,
″      <td>5079167</td>\n″,
″      <td>M</td>\n″,
″      <td>Y</td>\n″,
″      <td>N</td>\n″,
″      <td>0</td>\n″,
"      <td>202500.0</td>\n",
"      <td>Working</td>\n",
"      <td>Secondary / secondary special</td>\n",
"      <td>Married</td>\n",
"      <td>With parents</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>Laborers</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>3.87</td>\n",
"      <td>38.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>3</th>\n",
"      <td>5079168</td>\n",
"      <td>M</td>\n",
"      <td>Y</td>\n",
"      <td>N</td>\n",
"      <td>0</td>\n",
"      <td>202500.0</td>\n",
"      <td>Working</td>\n",
"      <td>Secondary / secondary special</td>\n",
"      <td>Married</td>\n",
"      <td>With parents</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>Laborers</td>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>3.87</td>\n",
"      <td>38.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>4</th>\n",
"      <td>5050729</td>\n",
"      <td>M</td>\n",
"      <td>Y</td>\n",
"      <td>N</td>\n",
"      <td>0</td>\n",
"      <td>180000.0</td>\n",
"      <td>Working</td>\n",
"      <td>Secondary / secondary special</td>\n",
"      <td>Single / not married</td>\n",
"      <td>With parents</td>\n",
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Unknown</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>1.48</td>\n″,
″      <td>24.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>5</th>\n″,
″      <td>5028383</td>\n″,
″      <td>M</td>\n″,
″      <td>Y</td>\n″,
″      <td>N</td>\n″,
″      <td>0</td>\n″,
″      <td>337500.0</td>\n″,
″      <td>Working</td>\n″,
″      <td>Secondary / secondary special</td>\n″,
″      <td>Single / not married</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Core staff</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>9.19</td>\n″,
″      <td>32.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>6</th>\n″,
″      <td>5143019</td>\n″,
″      <td>M</td>\n″,
″      <td>N</td>\n″,
″      <td>N</td>\n″,
″      <td>0</td>\n″,
″      <td>270000.0</td>\n″,
″      <td>Working</td>\n″,
″      <td>Secondary / secondary special</td>\n″,
″      <td>Married</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
″      <td>1</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>Laborers</td>\n″,
"      <td>2</td>\n",
"      <td>0</td>\n",
"      <td>1.73</td>\n",
"      <td>33.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>7</th>\n",
"      <td>5067982</td>\n",
"      <td>M</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>0</td>\n",
"      <td>153000.0</td>\n",
"      <td>Working</td>\n",
"      <td>Higher education</td>\n",
"      <td>Separated</td>\n",
"      <td>With parents</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>1</td>\n",
"      <td>Managers</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>21.22</td>\n",
"      <td>56.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>8</th>\n",
"      <td>5143573</td>\n",
"      <td>M</td>\n",
"      <td>Y</td>\n",
"      <td>N</td>\n",
"      <td>0</td>\n",
"      <td>157500.0</td>\n",
"      <td>Working</td>\n",
"      <td>Incomplete higher</td>\n",
"      <td>Married</td>\n",
"      <td>With parents</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>Drivers</td>\n",
"      <td>2</td>\n",
"      <td>0</td>\n",
"      <td>2.67</td>\n",
"      <td>25.0</td>\n",
"    </tr>\n",
″    <tr>\n″,
″      <th>9</th>\n″,
″      <td>5038751</td>\n″,
″      <td>M</td>\n″,
″      <td>Y</td>\n″,
″      <td>Y</td>\n″,
″      <td>2</td>\n″,
″      <td>157500.0</td>\n″,
″      <td>Commercial associate</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Drivers</td>\n″,
″      <td>4</td>\n″,
″      <td>0</td>\n″,
″      <td>1.74</td>\n″,
″      <td>39.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>10</th>\n″,
″      <td>5010203</td>\n″,
″      <td>M</td>\n″,
″      <td>Y</td>\n″,
″      <td>Y</td>\n″,
″      <td>0</td>\n″,
″      <td>135000.0</td>\n″,
″      <td>Working</td>\n″,
″      <td>Lower secondary</td>\n″,
″      <td>Married</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>Laborers</td>\n″,
″      <td>2</td>\n″,
″      <td>0</td>\n″,
″      <td>5.18</td>\n″,
″      <td>28.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>11</th>\n″,
″      <td>5058267</td>\n″,
″      <td>M</td>\n″,
″      <td>N</td>\n″,
"      <td>N</td>\n",
"      <td>1</td>\n",
"      <td>225000.0</td>\n",
"      <td>Commercial associate</td>\n",
"      <td>Secondary / secondary special</td>\n",
"      <td>Married</td>\n",
"      <td>With parents</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>Drivers</td>\n",
"      <td>3</td>\n",
"      <td>0</td>\n",
"      <td>5.31</td>\n",
"      <td>50.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>12</th>\n",
"      <td>5024352</td>\n",
"      <td>M</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>0</td>\n",
"      <td>585000.0</td>\n",
"      <td>Commercial associate</td>\n",
"      <td>Secondary / secondary special</td>\n",
"      <td>Married</td>\n",
"      <td>With parents</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>Drivers</td>\n",
"      <td>2</td>\n",
"      <td>0</td>\n",
"      <td>4.57</td>\n",
"      <td>46.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>13</th>\n",
"      <td>5150038</td>\n",
"      <td>M</td>\n",
"      <td>N</td>\n",
"      <td>N</td>\n",
"      <td>0</td>\n",
"      <td>180000.0</td>\n",
"      <td>Working</td>\n",
"      <td>Secondary / secondary special</td>\n",
″      <td>Single / not married</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Drivers</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>4.27</td>\n″,
″      <td>36.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>14</th>\n″,
″      <td>5113302</td>\n″,
″      <td>M</td>\n″,
″      <td>N</td>\n″,
″      <td>Y</td>\n″,
″      <td>1</td>\n″,
″      <td>180000.0</td>\n″,
″      <td>Working</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Cooking staff</td>\n″,
″      <td>3</td>\n″,
″      <td>0</td>\n″,
″      <td>4.59</td>\n″,
″      <td>44.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>15</th>\n″,
″      <td>5126311</td>\n″,
″      <td>M</td>\n″,
″      <td>N</td>\n″,
″      <td>N</td>\n″,
″      <td>0</td>\n″,
″      <td>180000.0</td>\n″,
″      <td>Working</td>\n″,
″      <td>Incomplete higher</td>\n″,
″      <td>Civil marriage</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>Core staff</td>\n″,
″      <td>2</td>\n″,
″      <td>0</td>\n″,
″      <td>0.80</td>\n″,
″      <td>21.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>16</th>\n″,
″      <td>5126751</td>\n″,
″      <td>M</td>\n″,
″      <td>Y</td>\n″,
″      <td>N</td>\n″,
″      <td>2</td>\n″,
″      <td>135000.0</td>\n″,
″      <td>Working</td>\n″,
″      <td>Secondary / secondary special</td>\n″,
″      <td>Married</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Unknown</td>\n″,
″      <td>4</td>\n″,
″      <td>0</td>\n″,
″      <td>1.56</td>\n″,
″      <td>46.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>17</th>\n″,
″      <td>5037125</td>\n″,
″      <td>M</td>\n″,
″      <td>Y</td>\n″,
″      <td>N</td>\n″,
″      <td>0</td>\n″,
″      <td>180000.0</td>\n″,
″      <td>Working</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>Core staff</td>\n″,
″      <td>2</td>\n″,
″      <td>0</td>\n″,
″      <td>6.19</td>\n″,
"      <td>32.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>18</th>\n",
"      <td>5054414</td>\n",
"      <td>M</td>\n",
"      <td>N</td>\n",
"      <td>N</td>\n",
"      <td>0</td>\n",
"      <td>225000.0</td>\n",
"      <td>Commercial associate</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>With parents</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>Managers</td>\n",
"      <td>2</td>\n",
"      <td>0</td>\n",
"      <td>14.12</td>\n",
"      <td>36.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>19</th>\n",
"      <td>5145694</td>\n",
"      <td>M</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>0</td>\n",
"      <td>225000.0</td>\n",
"      <td>Working</td>\n",
"      <td>Higher education</td>\n",
"      <td>Single / not married</td>\n",
"      <td>With parents</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>Unknown</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>1.84</td>\n",
"      <td>26.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>20</th>\n",
"      <td>5146429</td>\n",
″      <td>M</td>\n″,
″      <td>Y</td>\n″,
″      <td>N</td>\n″,
″      <td>0</td>\n″,
″      <td>261000.0</td>\n″,
″      <td>Commercial associate</td>\n″,
″      <td>Incomplete higher</td>\n″,
″      <td>Civil marriage</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Sales staff</td>\n″,
″      <td>2</td>\n″,
″      <td>0</td>\n″,
″      <td>1.87</td>\n″,
″      <td>25.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>21</th>\n″,
″      <td>5054407</td>\n″,
″      <td>M</td>\n″,
″      <td>N</td>\n″,
″      <td>N</td>\n″,
″      <td>0</td>\n″,
″      <td>225000.0</td>\n″,
″      <td>Commercial associate</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Managers</td>\n″,
″      <td>2</td>\n″,
″      <td>0</td>\n″,
″      <td>14.12</td>\n″,
″      <td>36.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>22</th>\n″,
″      <td>5068648</td>\n″,
″      <td>M</td>\n″,
″      <td>N</td>\n″,
″      <td>Y</td>\n″,
″      <td>0</td>\n″,
″      <td>135000.0</td>\n″,
″      <td>Working</td>\n″,
″      <td>Secondary / secondary special</td>\n″,
″      <td>Single / not married</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Drivers</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>2.93</td>\n″,
″      <td>32.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>23</th>\n″,
″      <td>5143560</td>\n″,
″      <td>M</td>\n″,
″      <td>N</td>\n″,
″      <td>N</td>\n″,
″      <td>2</td>\n″,
″      <td>202500.0</td>\n″,
″      <td>Working</td>\n″,
″      <td>Secondary / secondary special</td>\n″,
″      <td>Married</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Drivers</td>\n″,
″      <td>4</td>\n″,
″      <td>0</td>\n″,
″      <td>2.07</td>\n″,
″      <td>30.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>24</th>\n″,
″      <td>5125542</td>\n″,
″      <td>M</td>\n″,
″      <td>N</td>\n″,
″      <td>N</td>\n″,
″      <td>0</td>\n″,
″      <td>180000.0</td>\n″,
″      <td>Working</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>High skill tech staff</td>\n",
"      <td>2</td>\n",
"      <td>0</td>\n",
"      <td>6.71</td>\n",
"      <td>37.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>25</th>\n",
"      <td>5142781</td>\n",
"      <td>M</td>\n",
"      <td>Y</td>\n",
"      <td>N</td>\n",
"      <td>2</td>\n",
"      <td>157500.0</td>\n",
"      <td>Working</td>\n",
"      <td>Secondary / secondary special</td>\n",
"      <td>Separated</td>\n",
"      <td>With parents</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>Laborers</td>\n",
"      <td>3</td>\n",
"      <td>0</td>\n",
"      <td>4.52</td>\n",
"      <td>39.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>26</th>\n",
"      <td>5024592</td>\n",
"      <td>M</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>0</td>\n",
"      <td>157500.0</td>\n",
"      <td>Working</td>\n",
"      <td>Secondary / secondary special</td>\n",
"      <td>Married</td>\n",
"      <td>With parents</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>Drivers</td>\n",
"      <td>2</td>\n",
″      <td>0</td>\n″,
″      <td>4.49</td>\n″,
″      <td>25.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>27</th>\n″,
″      <td>5094884</td>\n″,
″      <td>M</td>\n″,
″      <td>N</td>\n″,
″      <td>Y</td>\n″,
″      <td>0</td>\n″,
″      <td>450000.0</td>\n″,
″      <td>Commercial associate</td>\n″,
″      <td>Secondary / secondary special</td>\n″,
″      <td>Single / not married</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Managers</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>3.56</td>\n″,
″      <td>29.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>28</th>\n″,
″      <td>5033496</td>\n″,
″      <td>M</td>\n″,
″      <td>Y</td>\n″,
″      <td>N</td>\n″,
″      <td>1</td>\n″,
″      <td>202500.0</td>\n″,
″      <td>Commercial associate</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
″      <td>With parents</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Managers</td>\n″,
″      <td>3</td>\n″,
″      <td>0</td>\n″,
″      <td>1.23</td>\n″,
″      <td>28.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
"      <th>29</th>\n",
"      <td>5067203</td>\n",
"      <td>M</td>\n",
"      <td>N</td>\n",
"      <td>N</td>\n",
"      <td>1</td>\n",
"      <td>405000.0</td>\n",
"      <td>Working</td>\n",
"      <td>Secondary / secondary special</td>\n",
"      <td>Married</td>\n",
"      <td>With parents</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>Drivers</td>\n",
"      <td>3</td>\n",
"      <td>0</td>\n",
"      <td>3.69</td>\n",
"      <td>35.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>30</th>\n",
"      <td>5045537</td>\n",
"      <td>M</td>\n",
"      <td>Y</td>\n",
"      <td>N</td>\n",
"      <td>2</td>\n",
"      <td>225000.0</td>\n",
"      <td>State servant</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>With parents</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>High skill tech staff</td>\n",
"      <td>4</td>\n",
"      <td>0</td>\n",
"      <td>4.28</td>\n",
"      <td>30.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>31</th>\n",
"      <td>5009269</td>\n",
"      <td>M</td>\n",
"      <td>Y</td>\n",
"      <td>N</td>\n",
"      <td>0</td>\n",
"      <td>337500.0</td>\n",
"      <td>State servant</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>With parents</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>Managers</td>\n",
"      <td>2</td>\n",
"      <td>0</td>\n",
"      <td>4.59</td>\n",
"      <td>29.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>32</th>\n",
"      <td>5069007</td>\n",
"      <td>M</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>0</td>\n",
"      <td>337500.0</td>\n",
"      <td>Working</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>With parents</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>Laborers</td>\n",
"      <td>2</td>\n",
"      <td>0</td>\n",
"      <td>3.26</td>\n",
"      <td>30.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>33</th>\n",
"      <td>5066915</td>\n",
"      <td>M</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>0</td>\n",
"      <td>180000.0</td>\n",
"      <td>Working</td>\n",
"      <td>Higher education</td>\n",
"      <td>Married</td>\n",
"      <td>With parents</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>Laborers</td>\n",
"      <td>2</td>\n",
"      <td>0</td>\n",
"      <td>4.08</td>\n",
"      <td>31.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>34</th>\n",
"      <td>5143025</td>\n",
"      <td>M</td>\n",
"      <td>N</td>\n",
"      <td>N</td>\n",
"      <td>0</td>\n",
"      <td>270000.0</td>\n",
"      <td>Working</td>\n",
"      <td>Secondary / secondary special</td>\n",
"      <td>Married</td>\n",
"      <td>With parents</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>Laborers</td>\n",
"      <td>2</td>\n",
"      <td>0</td>\n",
"      <td>1.73</td>\n",
"      <td>33.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>35</th>\n",
"      <td>5010202</td>\n",
"      <td>M</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>0</td>\n",
"      <td>135000.0</td>\n",
"      <td>Working</td>\n",
"      <td>Lower secondary</td>\n",
"      <td>Married</td>\n",
"      <td>With parents</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>1</td>\n",
              "      <td>Laborers</td>\n",
              "      <td>2</td>\n",
              "      <td>0</td>\n",
              "      <td>5.18</td>\n",
              "      <td>28.0</td>\n",
              "    </tr>\n",
              "    <tr>\n",
              "      <th>36</th>\n",
              "      <td>5028612</td>\n",
              "      <td>M</td>\n",
              "      <td>Y</td>\n",
              "      <td>Y</td>\n",
              "      <td>0</td>\n",
              "      <td>225000.0</td>\n",
              "      <td>Commercial associate</td>\n",
              "      <td>Higher education</td>\n",
              "      <td>Single / not married</td>\n",
              "      <td>With parents</td>\n",
              "      <td>1</td>\n",
              "      <td>0</td>\n",
              "      <td>1</td>\n",
              "      <td>0</td>\n",
              "      <td>Sales staff</td>\n",
              "      <td>1</td>\n",
              "      <td>0</td>\n",
              "      <td>5.98</td>\n",
              "      <td>30.0</td>\n",
              "    </tr>\n",
              "  </tbody>\n",
              "</table>\n",
              "</div>\n",
              "    <div class=\"colab-df-buttons\">\n",
              "\n",
              "  <div class=\"colab-df-container\">\n",
              "    <button class=\"colab-df-convert\"
onclick=\"convertToInteractive('df-86a17c24-b24c-45dc-ab6d-8e4b06b2ca98')\"\n",
              "            title=\"Convert this dataframe to an interactive
table.\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\" viewBox=\"0
-960 960 960\">\n",
              "    <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220
220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-
160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",
              "  </svg>\n",
              "    </button>\n",
              "\n",
              "  <style>\n",
              ″    .colab-df-container {\n″,
              ″      display:flex;\n″,
              ″      gap: 12px;\n″,
              ″    }\n″,
              ″\n″,
              ″    .colab-df-convert {\n″,
              ″      background-color: #E8F0FE;\n″,
              ″      border: none;\n″,
              ″      border-radius: 50%;\n″,
              ″      cursor: pointer;\n″,
              ″      display: none;\n″,
              ″      fill: #1967D2;\n″,
              ″      height: 32px;\n″,
              ″      padding: 0 0 0 0;\n″,
              ″      width: 32px;\n″,
              ″    }\n″,
              ″\n″,
              ″    .colab-df-convert:hover {\n″,
              ″      background-color: #E2EBFA;\n″,
              ″      box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n″,
              ″      fill: #174EA6;\n″,
              ″    }\n″,
              ″\n″,
              ″    .colab-df-buttons div {\n″,
              ″      margin-bottom: 4px;\n″,
              ″    }\n″,
              ″\n″,
              ″    [theme=dark] .colab-df-convert {\n″,
              ″      background-color: #3B4455;\n″,
              ″      fill: #D2E3FC;\n″,
              ″    }\n″,
              ″\n″,
              ″    [theme=dark] .colab-df-convert:hover {\n″,
              ″      background-color: #434B5C;\n″,
              ″      box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n″,
              ″      filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n″,
              ″      fill: #FFFFFF;\n″,
              ″    }\n″,
              ″  </style>\n″,
              ″\n″,
              ″    <script>\n″,
              ″      const buttonEl =\n″,
              ″        document.querySelector('#df-86a17c24-b24c-45dc-ab6d-
8e4b06b2ca98 button.colab-df-convert');\n″,
              ″      buttonEl.style.display =\n″,
              ″        google.colab.kernel.accessAllowed ? 'block' : 'none';\n″,
              ″\n″,
              ″      async function convertToInteractive(key) {\n″,
              ″        const element = document.querySelector('#df-86a17c24-b24c-45dc-
ab6d-8e4b06b2ca98');\n″,
              ″        const dataTable =\n″,
              ″          await
google.colab.kernel.invokeFunction('convertToInteractive',\n″,
              ″                                                    [key], {});\n″,
              ″        if (!dataTable) return;\n″,
              ″\n″,
              ″        const docLinkHtml = 'Like what you see? Visit the ' +\n″,
              ″          '<a target=\″_blank\″
href=https://colab.research.google.com/notebooks/data_table.ipynb>data table
notebook</a>'\n″,
              ″          + ' to learn more about interactive tables.';\n″,
              ″        element.innerHTML = '';\n″,
              ″        dataTable['output_type'] = 'display_data';\n″,
              ″        await google.colab.output.renderOutput(dataTable, element);\n″,
              ″        const docLink = document.createElement('div');\n″,
              ″        docLink.innerHTML = docLinkHtml;\n″,
              ″        element.appendChild(docLink);\n″,
              ″      }\n″,
              ″    </script>\n″,
              ″  </div>\n″,
              ″\n″,
              ″\n″,
              ″<div id=\″df-3a8bca42-457e-45a9-a0dd-d242938b001a\″>\n″,
              ″  <button class=\″colab-df-quickchart\″ onclick=\″quickchart('df-
3a8bca42-457e-45a9-a0dd-d242938b001a')\″\n″,
              ″            title=\″Suggest charts\″\n″,
              ″            style=\″display:none;\″>\n″,
              ″\n″,
              ″<svg xmlns=\″http://www.w3.org/2000/svg\″ height=\″24px\″viewBox=\″0 0
24 24\″\n″,
              ″     width=\″24px\″>\n″,
              ″    <g>\n″,
              ″        <path d=\″M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-
.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\″/>\n″,
              ″    </g>\n″,
              ″</svg>\n″,
              ″  </button>\n″,
              ″\n″,
              ″<style>\n″,
              ″  .colab-df-quickchart {\n″,
              ″      --bg-color: #E8F0FE;\n″,
              ″      --fill-color: #1967D2;\n″,
              ″      --hover-bg-color: #E2EBFA;\n″,
              ″      --hover-fill-color: #174EA6;\n″,
              ″      --disabled-fill-color: #AAA;\n″,
              ″      --disabled-bg-color: #DDD;\n″,
              ″  }\n″,
              "\n",
              "  [theme=dark] .colab-df-quickchart {\n",
              "      --bg-color: #3B4455;\n",
              "      --fill-color: #D2E3FC;\n",
              "      --hover-bg-color: #434B5C;\n",
              "      --hover-fill-color: #FFFFFF;\n",
              "      --disabled-bg-color: #3B4455;\n",
              "      --disabled-fill-color: #666;\n",
              "  }\n",
              "\n",
              "  .colab-df-quickchart {\n",
              "    background-color: var(--bg-color);\n",
              "    border: none;\n",
              "    border-radius: 50%;\n",
              "    cursor: pointer;\n",
              "    display: none;\n",
              "    fill: var(--fill-color);\n",
              "    height: 32px;\n",
              "    padding: 0;\n",
              "    width: 32px;\n",
              "  }\n",
              "\n",
              "  .colab-df-quickchart:hover {\n",
              "    background-color: var(--hover-bg-color);\n",
              "    box-shadow: 0 1px 2px rgba(60, 64, 67, 0.3), 0 1px 3px 1px rgba(60,
64, 67, 0.15);\n",
              "    fill: var(--button-hover-fill-color);\n",
              "  }\n",
              "\n",
              "  .colab-df-quickchart-complete:disabled,\n",
              "  .colab-df-quickchart-complete:disabled:hover {\n",
              "    background-color: var(--disabled-bg-color);\n",
              "    fill: var(--disabled-fill-color);\n",
              "    box-shadow: none;\n",
              "  }\n",
              "\n",
              "  .colab-df-spinner {\n",
              "    border: 2px solid var(--fill-color);\n",
              "    border-color: transparent;\n",
              "    border-bottom-color: var(--fill-color);\n",
              "    animation:\n",
              "      spin 1s steps(1) infinite;\n",
              "  }\n",
              "\n",
              "  @keyframes spin {\n",
              "    0% {\n",
              "      border-color: transparent;\n",
              "      border-bottom-color: var(--fill-color);\n",
              "      border-left-color: var(--fill-color);\n",
              "    }\n",
              "    20% {\n",
              "      border-color: transparent;\n",
              "      border-left-color: var(--fill-color);\n",
              "      border-top-color: var(--fill-color);\n",
              "    }\n",
              "    30% {\n",
              "      border-color: transparent;\n",
              "      border-left-color: var(--fill-color);\n",
              "      border-top-color: var(--fill-color);\n",
              "      border-right-color: var(--fill-color);\n",
              "    }\n",
              "    40% {\n",
              "      border-color: transparent;\n",
              "      border-right-color: var(--fill-color);\n",
              "      border-top-color: var(--fill-color);\n",
              "    }\n",
              "    60% {\n",
              "      border-color: transparent;\n",
              "      border-right-color: var(--fill-color);\n",
              "    }\n",
              "    80% {\n",
              "      border-color: transparent;\n",
              "      border-right-color: var(--fill-color);\n",
              "      border-bottom-color: var(--fill-color);\n",
              "    }\n",
              "    90% {\n",
              "      border-color: transparent;\n",
              "      border-bottom-color: var(--fill-color);\n",
              "    }\n",
              "  }\n",
              "</style>\n",
              "\n",
              "  <script>\n",
              "    async function quickchart(key) {\n",
              "      const quickchartButtonEl =\n",
              "        document.querySelector('#' + key + ' button');\n",
              "      quickchartButtonEl.disabled = true;  // To prevent multiple 
clicks.\n",
              "      quickchartButtonEl.classList.add('colab-df-spinner');\n",
              "      try {\n",
              "        const charts = await google.colab.kernel.invokeFunction(\n",
              "            'suggestCharts', [key], {});\n",
              "      } catch (error) {\n",
              "        console.error('Error during call to suggestCharts:', error);\n",
              "      }\n",
              "      quickchartButtonEl.classList.remove('colab-df-spinner');\n",
              "      quickchartButtonEl.classList.add('colab-df-quickchart-
complete');\n",
              ″    }\n″,
              ″    (() => {\n″,
              ″      let quickchartButtonEl =\n″,
              ″        document.querySelector(’#df-3a8bca42-457e-45a9-a0dd-
d242938b001a button’);\n″,
              ″      quickchartButtonEl.style.display =\n″,
              ″        google.colab.kernel.accessAllowed ? ’block’ : ’none’;\n″,
              ″    })();\n″,
              ″  </script>\n″,
              ″</div>\n″,
              ″\n″,
              ″  <div id=\″id_c5099022-1f79-4f29-baed-b1b3da7e5d33\″>\n″,
              ″    <style>\n″,
              ″      .colab-df-generate {\n″,
              ″        background-color: #E8F0FE;\n″,
              ″        border: none;\n″,
              ″        border-radius: 50%;\n″,
              ″        cursor: pointer;\n″,
              ″        display: none;\n″,
              ″        fill: #1967D2;\n″,
              ″        height: 32px;\n″,
              ″        padding: 0 0 0 0;\n″,
              ″        width: 32px;\n″,
              ″      }\n″,
              ″\n″,
              ″      .colab-df-generate:hover {\n″,
              ″        background-color: #E2EBFA;\n″,
              ″        box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n″,
              ″        fill: #174EA6;\n″,
              ″      }\n″,
              ″\n″,
              ″      [theme=dark] .colab-df-generate {\n″,
              ″        background-color: #3B4455;\n″,
              ″        fill: #D2E3FC;\n″,
              ″      }\n″,
              ″\n″,
              ″      [theme=dark] .colab-df-generate:hover {\n″,
              ″        background-color: #434B5C;\n″,
              ″        box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n″,
              ″        filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n″,
              ″        fill: #FFFFFF;\n″,
              ″      }\n″,
              ″    </style>\n″,
              ″    <button class=\″colab-df-generate\″
onclick=\″generateWithVariable(’result’)\″\n″,
              ″            title=\″Generate code using this dataframe.\″\n″,
              ″            style=\″display:none;\″>\n″,
              ″\n″,
              "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0
0 24 24\"\n",
              "       width=\"24px\">\n",
              "    <path
d=\"M7,19H8.4L18.45,9,17,7.55,7,17.6ZM5,21V16.75L18.45,3.32a2,2,0,0,1,2.83,0l1.4,1.43a
1.91,1.91,0,0,1,.58,1.4,1.91,1.91,0,0,1-.58,1.4L9.25,21ZM18.45,9,17,7.55Zm-
12,3A5.31,5.31,0,0,0,4.9,8.1,5.31,5.31,0,0,0,1,6.5,5.31,5.31,0,0,0,4.9,4.9,5.31,5.31,0,
0,0,6.5,1,5.31,5.31,0,0,0,8.1,4.9,5.31,5.31,0,0,0,12,6.5,5.46,5.46,0,0,0,6.5,12Z\"/>\n
",
              "  </svg>\n",
              "    </button>\n",
              "    <script>\n",
              "      (() => {\n",
              "      const buttonEl =\n",
              "        document.querySelector('#id_c5099022-1f79-4f29-baed-
b1b3da7e5d33 button.colab-df-generate');\n",
              "      buttonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "\n",
              "      buttonEl.onclick = () => {\n",
              "        google.colab.notebook.generateWithVariable('result');\n",
              "      }\n",
              "      })();\n",
              "    </script>\n",
              "  </div>\n",
              "\n",
              "    </div>\n",
              "  </div>\n"
            ]
          },
          "metadata": {},
          "execution_count": 1365
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "## Q4.\n",
        " Please list the top five people having the highest income."
      ],
      "metadata": {
        "id": "vlR6n74KWu6I"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "query = \"SELECT * FROM credit_data ORDER BY Annual_income DESC LIMIT 5\"\n",
        "result = connection.execute(query).fetchdf()\n",
        "result"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 347
        },
        "id": "TZ3GvFwDXafM",
        "outputId": "875ab098-337d-4ede-8153-32bb35a11444"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "    Ind_ID GENDER Car_Owner Propert_Owner  CHILDREN  Annual_income
\\\n",
              "0  5143231      F         Y             Y         1      1575000.0
\n",
              "1  5143235      F         Y             Y         1      1575000.0
\n",
              "2  5090470      M         N             Y         1       900000.0
\n",
              "3  5079016      M         Y             Y         2       900000.0
\n",
              "4  5079017      M         Y             Y         2       900000.0
\n",
              "\n",
              "            Type_Income                      EDUCATION
Marital_status  \\\n",
              "0  Commercial associate               Higher education  Single / not
married   \n",
              "1  Commercial associate               Higher education  Single / not
married   \n",
              "2               Working  Secondary / secondary special
Married   \n",
              "3  Commercial associate               Higher education
Married   \n",
              "4  Commercial associate               Higher education
Married   \n",
              "\n",
              "        Housing_type  Mobile_phone  Work_Phone  Phone  EMAIL_ID  \\\n",
              "0  House / apartment             1           0      0         0   \n",
              "1  House / apartment             1           0      0         0   \n",
              "2  House / apartment             1           0      0         0   \n",
              "3  House / apartment             1           0      0         0   \n",
              "4  House / apartment             1           0      0         0   \n",
              ″\n″,
              ″  Type_Occupation  Family_Members  label  Employed_years   Age  \n″,
              ″0        Managers               2      0            6.89  28.0  \n″,
              ″1        Managers               2      0            6.89  28.0  \n″,
              ″2        Laborers               3      0           12.52  43.0  \n″,
              ″3        Managers               4      0            2.78  27.0  \n″,
              ″4        Managers               4      0            2.78  27.0  ″
            ],
            ″text/html″: [
              ″\n″,
              ″  <div id=\″df-4c11fb55-61a4-4abe-b021-fe03acbe3c6b\″ class=\″colab-df-
container\″>\n″,
              ″    <div>\n″,
              ″<style scoped>\n″,
              ″    .dataframe tbody tr th:only-of-type {\n″,
              ″        vertical-align: middle;\n″,
              ″    }\n″,
              ″\n″,
              ″    .dataframe tbody tr th {\n″,
              ″        vertical-align: top;\n″,
              ″    }\n″,
              ″\n″,
              ″    .dataframe thead th {\n″,
              ″        text-align: right;\n″,
              ″    }\n″,
              ″</style>\n″,
              ″<table border=\″1\″ class=\″dataframe\″>\n″,
              ″  <thead>\n″,
              ″    <tr style=\″text-align: right;\″>\n″,
              ″      <th></th>\n″,
              ″      <th>Ind_ID</th>\n″,
              ″      <th>GENDER</th>\n″,
              ″      <th>Car_Owner</th>\n″,
              ″      <th>Propert_Owner</th>\n″,
              ″      <th>CHILDREN</th>\n″,
              ″      <th>Annual_income</th>\n″,
              ″      <th>Type_Income</th>\n″,
              ″      <th>EDUCATION</th>\n″,
              ″      <th>Marital_status</th>\n″,
              ″      <th>Housing_type</th>\n″,
              ″      <th>Mobile_phone</th>\n″,
              ″      <th>Work_Phone</th>\n″,
              ″      <th>Phone</th>\n″,
              ″      <th>EMAIL_ID</th>\n″,
              ″      <th>Type_Occupation</th>\n″,
              ″      <th>Family_Members</th>\n″,
              ″      <th>label</th>\n″,
              ″      <th>Employed_years</th>\n″,
              ″      <th>Age</th>\n″,
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>0</th>\n",
"      <td>5143231</td>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>1</td>\n",
"      <td>1575000.0</td>\n",
"      <td>Commercial associate</td>\n",
"      <td>Higher education</td>\n",
"      <td>Single / not married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>Managers</td>\n",
"      <td>2</td>\n",
"      <td>0</td>\n",
"      <td>6.89</td>\n",
"      <td>28.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>1</th>\n",
"      <td>5143235</td>\n",
"      <td>F</td>\n",
"      <td>Y</td>\n",
"      <td>Y</td>\n",
"      <td>1</td>\n",
"      <td>1575000.0</td>\n",
"      <td>Commercial associate</td>\n",
"      <td>Higher education</td>\n",
"      <td>Single / not married</td>\n",
"      <td>House / apartment</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>Managers</td>\n",
"      <td>2</td>\n",
"      <td>0</td>\n",
"      <td>6.89</td>\n",
"      <td>28.0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>2</th>\n",
″      <td>5090470</td>\n″,
″      <td>M</td>\n″,
″      <td>N</td>\n″,
″      <td>Y</td>\n″,
″      <td>1</td>\n″,
″      <td>900000.0</td>\n″,
″      <td>Working</td>\n″,
″      <td>Secondary / secondary special</td>\n″,
″      <td>Married</td>\n″,
″      <td>House / apartment</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Laborers</td>\n″,
″      <td>3</td>\n″,
″      <td>0</td>\n″,
″      <td>12.52</td>\n″,
″      <td>43.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>3</th>\n″,
″      <td>5079016</td>\n″,
″      <td>M</td>\n″,
″      <td>Y</td>\n″,
″      <td>Y</td>\n″,
″      <td>2</td>\n″,
″      <td>900000.0</td>\n″,
″      <td>Commercial associate</td>\n″,
″      <td>Higher education</td>\n″,
″      <td>Married</td>\n″,
″      <td>House / apartment</td>\n″,
″      <td>1</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>0</td>\n″,
″      <td>Managers</td>\n″,
″      <td>4</td>\n″,
″      <td>0</td>\n″,
″      <td>2.78</td>\n″,
″      <td>27.0</td>\n″,
″    </tr>\n″,
″    <tr>\n″,
″      <th>4</th>\n″,
″      <td>5079017</td>\n″,
″      <td>M</td>\n″,
″      <td>Y</td>\n″,
″      <td>Y</td>\n″,
″      <td>2</td>\n″,
              "      <td>900000.0</td>\n",
              "      <td>Commercial associate</td>\n",
              "      <td>Higher education</td>\n",
              "      <td>Married</td>\n",
              "      <td>House / apartment</td>\n",
              "      <td>1</td>\n",
              "      <td>0</td>\n",
              "      <td>0</td>\n",
              "      <td>0</td>\n",
              "      <td>Managers</td>\n",
              "      <td>4</td>\n",
              "      <td>0</td>\n",
              "      <td>2.78</td>\n",
              "      <td>27.0</td>\n",
              "    </tr>\n",
              "  </tbody>\n",
              "</table>\n",
              "</div>\n",
              "    <div class=\"colab-df-buttons\">\n",
              "\n",
              "  <div class=\"colab-df-container\">\n",
              "    <button class=\"colab-df-convert\"
onclick=\"convertToInteractive('df-4c11fb55-61a4-4abe-b021-fe03acbe3c6b')\"\n",
              "            title=\"Convert this dataframe to an interactive
table.\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\" viewBox=\"0
-960 960 960\">\n",
              "    <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220
220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-
160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",
              "  </svg>\n",
              "    </button>\n",
              "\n",
              "  <style>\n",
              "    .colab-df-container {\n",
              "      display:flex;\n",
              "      gap: 12px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert {\n",
              "      background-color: #E8F0FE;\n",
              "      border: none;\n",
              "      border-radius: 50%;\n",
              "      cursor: pointer;\n",
              "      display: none;\n",
              "      fill: #1967D2;\n",
              "      height: 32px;\n",
              ″      padding: 0 0 0 0;\n″,
              ″      width: 32px;\n″,
              ″    }\n″,
              ″\n″,
              ″    .colab-df-convert:hover {\n″,
              ″      background-color: #E2EBFA;\n″,
              ″      box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n″,
              ″      fill: #174EA6;\n″,
              ″    }\n″,
              ″\n″,
              ″    .colab-df-buttons div {\n″,
              ″      margin-bottom: 4px;\n″,
              ″    }\n″,
              ″\n″,
              ″    [theme=dark] .colab-df-convert {\n″,
              ″      background-color: #3B4455;\n″,
              ″      fill: #D2E3FC;\n″,
              ″    }\n″,
              ″\n″,
              ″    [theme=dark] .colab-df-convert:hover {\n″,
              ″      background-color: #434B5C;\n″,
              ″      box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n″,
              ″      filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n″,
              ″      fill: #FFFFFF;\n″,
              ″    }\n″,
              ″  </style>\n″,
              ″\n″,
              ″    <script>\n″,
              ″      const buttonEl =\n″,
              ″        document.querySelector(’#df-4c11fb55-61a4-4abe-b021-
fe03acbe3c6b button.colab-df-convert’);\n″,
              ″      buttonEl.style.display =\n″,
              ″        google.colab.kernel.accessAllowed ? ’block’ : ’none’;\n″,
              ″\n″,
              ″      async function convertToInteractive(key) {\n″,
              ″        const element = document.querySelector(’#df-4c11fb55-61a4-4abe-
b021-fe03acbe3c6b’);\n″,
              ″        const dataTable =\n″,
              ″          await
google.colab.kernel.invokeFunction(’convertToInteractive’,\n″,
              ″                                                    [key], {});\n″,
              ″        if (!dataTable) return;\n″,
              ″\n″,
              ″        const docLinkHtml = ’Like what you see? Visit the ’ +\n″,
              ″          ’<a target=\″_blank\″
href=https://colab.research.google.com/notebooks/data_table.ipynb>data table
notebook</a>’\n″,
              ″          + ’ to learn more about interactive tables.’;\n″,
              "        element.innerHTML = '';\n",
              "        dataTable['output_type'] = 'display_data';\n",
              "        await google.colab.output.renderOutput(dataTable, element);\n",
              "        const docLink = document.createElement('div');\n",
              "        docLink.innerHTML = docLinkHtml;\n",
              "        element.appendChild(docLink);\n",
              "      }\n",
              "    </script>\n",
              "  </div>\n",
              "\n",
              "\n",
              "<div id=\"df-d0ef9d78-a6d9-4801-9bc1-eb0060602b68\">\n",
              "  <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-d0ef9d78-a6d9-4801-9bc1-eb0060602b68')\"\n",
              "            title=\"Suggest charts\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "<svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0 0 24 24\"\n",
              "     width=\"24px\">\n",
              "    <g>\n",
              "        <path d=\"M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\"/>\n",
              "    </g>\n",
              "</svg>\n",
              "  </button>\n",
              "\n",
              "<style>\n",
              "  .colab-df-quickchart {\n",
              "      --bg-color: #E8F0FE;\n",
              "      --fill-color: #1967D2;\n",
              "      --hover-bg-color: #E2EBFA;\n",
              "      --hover-fill-color: #174EA6;\n",
              "      --disabled-fill-color: #AAA;\n",
              "      --disabled-bg-color: #DDD;\n",
              "  }\n",
              "\n",
              "  [theme=dark] .colab-df-quickchart {\n",
              "      --bg-color: #3B4455;\n",
              "      --fill-color: #D2E3FC;\n",
              "      --hover-bg-color: #434B5C;\n",
              "      --hover-fill-color: #FFFFFF;\n",
              "      --disabled-bg-color: #3B4455;\n",
              "      --disabled-fill-color: #666;\n",
              "  }\n",
              "\n",
              "  .colab-df-quickchart {\n",
              "    background-color: var(--bg-color);\n",
              "    border: none;\n",
              "    border-radius: 50%;\n",
              "    cursor: pointer;\n",
              "    display: none;\n",
              "    fill: var(--fill-color);\n",
              "    height: 32px;\n",
              "    padding: 0;\n",
              "    width: 32px;\n",
              "  }\n",
              "\n",
              "  .colab-df-quickchart:hover {\n",
              "    background-color: var(--hover-bg-color);\n",
              "    box-shadow: 0 1px 2px rgba(60, 64, 67, 0.3), 0 1px 3px 1px rgba(60,
64, 67, 0.15);\n",
              "    fill: var(--button-hover-fill-color);\n",
              "  }\n",
              "\n",
              "  .colab-df-quickchart-complete:disabled,\n",
              "  .colab-df-quickchart-complete:disabled:hover {\n",
              "    background-color: var(--disabled-bg-color);\n",
              "    fill: var(--disabled-fill-color);\n",
              "    box-shadow: none;\n",
              "  }\n",
              "\n",
              "  .colab-df-spinner {\n",
              "    border: 2px solid var(--fill-color);\n",
              "    border-color: transparent;\n",
              "    border-bottom-color: var(--fill-color);\n",
              "    animation:\n",
              "      spin 1s steps(1) infinite;\n",
              "  }\n",
              "\n",
              "  @keyframes spin {\n",
              "    0% {\n",
              "      border-color: transparent;\n",
              "      border-bottom-color: var(--fill-color);\n",
              "      border-left-color: var(--fill-color);\n",
              "    }\n",
              "    20% {\n",
              "      border-color: transparent;\n",
              "      border-left-color: var(--fill-color);\n",
              "      border-top-color: var(--fill-color);\n",
              "    }\n",
              "    30% {\n",
              "      border-color: transparent;\n",
              "      border-left-color: var(--fill-color);\n",
              "      border-top-color: var(--fill-color);\n",
              "      border-right-color: var(--fill-color);\n",
              "    }\n",
              "    40% {\n",
              "      border-color: transparent;\n",
              "      border-right-color: var(--fill-color);\n",
              "      border-top-color: var(--fill-color);\n",
              "    }\n",
              "    60% {\n",
              "      border-color: transparent;\n",
              "      border-right-color: var(--fill-color);\n",
              "    }\n",
              "    80% {\n",
              "      border-color: transparent;\n",
              "      border-right-color: var(--fill-color);\n",
              "      border-bottom-color: var(--fill-color);\n",
              "    }\n",
              "    90% {\n",
              "      border-color: transparent;\n",
              "      border-bottom-color: var(--fill-color);\n",
              "    }\n",
              "  }\n",
              "</style>\n",
              "\n",
              "  <script>\n",
              "    async function quickchart(key) {\n",
              "      const quickchartButtonEl =\n",
              "        document.querySelector('#' + key + ' button');\n",
              "      quickchartButtonEl.disabled = true;  // To prevent multiple
clicks.\n",
              "      quickchartButtonEl.classList.add('colab-df-spinner');\n",
              "      try {\n",
              "        const charts = await google.colab.kernel.invokeFunction(\n",
              "            'suggestCharts', [key], {});\n",
              "      } catch (error) {\n",
              "        console.error('Error during call to suggestCharts:', error);\n",
              "      }\n",
              "      quickchartButtonEl.classList.remove('colab-df-spinner');\n",
              "      quickchartButtonEl.classList.add('colab-df-quickchart-
complete');\n",
              "    }\n",
              "    (() => {\n",
              "      let quickchartButtonEl =\n",
              "        document.querySelector('#df-d0ef9d78-a6d9-4801-9bc1-
eb0060602b68 button');\n",
              "      quickchartButtonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "    })();\n",
              "  </script>\n",
              "</div>\n",
              "\n",
              "  <div id=\"id_67948a31-18ef-4ed7-989c-45ee4cf35cdf\">\n",
              "    <style>\n",
              ″      .colab-df-generate {\n″,
              ″        background-color: #E8F0FE;\n″,
              ″        border: none;\n″,
              ″        border-radius: 50%;\n″,
              ″        cursor: pointer;\n″,
              ″        display: none;\n″,
              ″        fill: #1967D2;\n″,
              ″        height: 32px;\n″,
              ″        padding: 0 0 0 0;\n″,
              ″        width: 32px;\n″,
              ″      }\n″,
              ″\n″,
              ″      .colab-df-generate:hover {\n″,
              ″        background-color: #E2EBFA;\n″,
              ″        box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n″,
              ″        fill: #174EA6;\n″,
              ″      }\n″,
              ″\n″,
              ″      [theme=dark] .colab-df-generate {\n″,
              ″        background-color: #3B4455;\n″,
              ″        fill: #D2E3FC;\n″,
              ″      }\n″,
              ″\n″,
              ″      [theme=dark] .colab-df-generate:hover {\n″,
              ″        background-color: #434B5C;\n″,
              ″        box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n″,
              ″        filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n″,
              ″        fill: #FFFFFF;\n″,
              ″      }\n″,
              ″    </style>\n″,
              ″    <button class=\″colab-df-generate\″
onclick=\″generateWithVariable(’result’)\″\n″,
              ″            title=\″Generate code using this dataframe.\″\n″,
              ″            style=\″display:none;\″>\n″,
              ″\n″,
              ″  <svg xmlns=\″http://www.w3.org/2000/svg\″ height=\″24px\″viewBox=\″0
0 24 24\″\n″,
              ″       width=\″24px\″>\n″,
              ″    <path
d=\″M7,19H8.4L18.45,9,17,7.55,7,17.6ZM5,21V16.75L18.45,3.32a2,2,0,0,1,2.83,0l1.4,1.43a
1.91,1.91,0,0,1,.58,1.4,1.91,1.91,0,0,1-.58,1.4L9.25,21ZM18.45,9,17,7.55Zm-
12,3A5.31,5.31,0,0,0,4.9,8.1,5.31,5.31,0,0,0,1,6.5,5.31,5.31,0,0,0,4.9,4.9,5.31,5.31,0,
0,0,6.5,1,5.31,5.31,0,0,0,8.1,4.9,5.31,5.31,0,0,0,12,6.5,5.46,5.46,0,0,0,6.5,12Z\″/>\n
″,
              ″  </svg>\n″,
              ″    </button>\n″,
              ″    <script>\n″,
              ″      (() => {\n″,
              "      const buttonEl =\n",
              "        document.querySelector('#id_67948a31-18ef-4ed7-989c-45ee4cf35cdf button.colab-df-generate');\n",
              "      buttonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "\n",
              "      buttonEl.onclick = () => {\n",
              "        google.colab.notebook.generateWithVariable('result');\n",
              "      }\n",
              "      })();\n",
              "    </script>\n",
              "  </div>\n",
              "\n",
              "    </div>\n",
              "  </div>\n"
            ]
          },
          "metadata": {},
          "execution_count": 1366
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "## Q5.\n",
        "How many married people are having bad credit?\n"
      ],
      "metadata": {
        "id": "DLm4v7WCXaD8"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "# categories of marital status\n",
        "df_sql[\"Marital_status\"].unique()"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "TVdAg3UsYsE3",
        "outputId": "20325568-299d-4063-badc-719cdf3f1bbe"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "array(['Married', 'Single / not married', 'Civil marriage',
'Separated',\n",
              "       'Widow'], dtype=object)"
            ]
          },
          "metadata": {},
          "execution_count": 1367
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "Marital status has two categories of married people \"Married\" and \"'Civil
marriage'\""
      ],
      "metadata": {
        "id": "3ocgOxyPZArI"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "query = \"SELECT COUNT(*) married_bad_credit FROM credit_data WHERE
(Marital_status = 'Married' OR Marital_status = 'Civil marriage') AND label = 1\"\n",
        "result = connection.execute(query).fetchdf()\n",
        "result"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 89
        },
        "id": "A004Bv-pYQRY",
        "outputId": "0b6b0d02-e126-48cd-8b6b-046da006a30a"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "   married_bad_credit\n",
              "0                118"
            ],
            "text/html": [
              "\n",
            "  <div id=\"df-cd17db22-c2fa-47ef-911a-f8a1f75824bf\" class=\"colab-df-
container\">\n",
            "    <div>\n",
            "<style scoped>\n",
            "    .dataframe tbody tr th:only-of-type {\n",
            "        vertical-align: middle;\n",
            "    }\n",
            "\n",
            "    .dataframe tbody tr th {\n",
            "        vertical-align: top;\n",
            "    }\n",
            "\n",
            "    .dataframe thead th {\n",
            "        text-align: right;\n",
            "    }\n",
            "</style>\n",
            "<table border=\"1\" class=\"dataframe\">\n",
            "  <thead>\n",
            "    <tr style=\"text-align: right;\">\n",
            "      <th></th>\n",
            "      <th>married_bad_credit</th>\n",
            "    </tr>\n",
            "  </thead>\n",
            "  <tbody>\n",
            "    <tr>\n",
            "      <th>0</th>\n",
            "      <td>118</td>\n",
            "    </tr>\n",
            "  </tbody>\n",
            "</table>\n",
            "</div>\n",
            "    <div class=\"colab-df-buttons\">\n",
            "\n",
            "  <div class=\"colab-df-container\">\n",
            "    <button class=\"colab-df-convert\"
onclick=\"convertToInteractive('df-cd17db22-c2fa-47ef-911a-f8a1f75824bf')\"\n",
            "            title=\"Convert this dataframe to an interactive
table.\"\n",
            "            style=\"display:none;\">\n",
            "\n",
            "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\" viewBox=\"0
-960 960 960\">\n",
            "    <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220
220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-
160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",
            "  </svg>\n",
            "    </button>\n",
            "\n",
            "  <style>\n",
       ″    .colab-df-container {\n″,
       ″      display:flex;\n″,
       ″      gap: 12px;\n″,
       ″    }\n″,
       ″\n″,
       ″    .colab-df-convert {\n″,
       ″      background-color: #E8F0FE;\n″,
       ″      border: none;\n″,
       ″      border-radius: 50%;\n″,
       ″      cursor: pointer;\n″,
       ″      display: none;\n″,
       ″      fill: #1967D2;\n″,
       ″      height: 32px;\n″,
       ″      padding: 0 0 0 0;\n″,
       ″      width: 32px;\n″,
       ″    }\n″,
       ″\n″,
       ″    .colab-df-convert:hover {\n″,
       ″      background-color: #E2EBFA;\n″,
       ″      box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n″,
       ″      fill: #174EA6;\n″,
       ″    }\n″,
       ″\n″,
       ″    .colab-df-buttons div {\n″,
       ″      margin-bottom: 4px;\n″,
       ″    }\n″,
       ″\n″,
       ″    [theme=dark] .colab-df-convert {\n″,
       ″      background-color: #3B4455;\n″,
       ″      fill: #D2E3FC;\n″,
       ″    }\n″,
       ″\n″,
       ″    [theme=dark] .colab-df-convert:hover {\n″,
       ″      background-color: #434B5C;\n″,
       ″      box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n″,
       ″      filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n″,
       ″      fill: #FFFFFF;\n″,
       ″    }\n″,
       ″  </style>\n″,
       ″\n″,
       ″    <script>\n″,
       ″      const buttonEl =\n″,
       ″        document.querySelector('#df-cd17db22-c2fa-47ef-911a-
f8a1f75824bf button.colab-df-convert');\n″,
       ″      buttonEl.style.display =\n″,
       ″        google.colab.kernel.accessAllowed ? 'block' : 'none';\n″,
       ″\n″,
       ″      async function convertToInteractive(key) {\n″,
              ″        const element = document.querySelector(’#df-cd17db22-c2fa-47ef-
911a-f8a1f75824bf’);\n″,
              ″        const dataTable =\n″,
              ″          await
google.colab.kernel.invokeFunction(’convertToInteractive’,\n″,
              ″                                                    [key], {});\n″,
              ″        if (!dataTable) return;\n″,
              ″\n″,
              ″        const docLinkHtml = 'Like what you see? Visit the ' +\n″,
              ″          '<a target=\″_blank\″
href=https://colab.research.google.com/notebooks/data_table.ipynb>data table
notebook</a>'\n″,
              ″          + ' to learn more about interactive tables.';\n″,
              ″        element.innerHTML = '';\n″,
              ″        dataTable['output_type'] = 'display_data';\n″,
              ″        await google.colab.output.renderOutput(dataTable, element);\n″,
              ″        const docLink = document.createElement('div');\n″,
              ″        docLink.innerHTML = docLinkHtml;\n″,
              ″        element.appendChild(docLink);\n″,
              ″      }\n″,
              ″    </script>\n″,
              ″  </div>\n″,
              ″\n″,
              ″\n″,
              ″  <div id=\″id_a1f39e2b-4448-42e5-9989-72cc8c61f9a3\″>\n″,
              ″    <style>\n″,
              ″      .colab-df-generate {\n″,
              ″        background-color: #E8F0FE;\n″,
              ″        border: none;\n″,
              ″        border-radius: 50%;\n″,
              ″        cursor: pointer;\n″,
              ″        display: none;\n″,
              ″        fill: #1967D2;\n″,
              ″        height: 32px;\n″,
              ″        padding: 0 0 0 0;\n″,
              ″        width: 32px;\n″,
              ″      }\n″,
              ″\n″,
              ″      .colab-df-generate:hover {\n″,
              ″        background-color: #E2EBFA;\n″,
              ″        box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n″,
              ″        fill: #174EA6;\n″,
              ″      }\n″,
              ″\n″,
              ″      [theme=dark] .colab-df-generate {\n″,
              ″        background-color: #3B4455;\n″,
              ″        fill: #D2E3FC;\n″,
              ″      }\n″,
              "\n",
              "      [theme=dark] .colab-df-generate:hover {\n",
              "        background-color: #434B5C;\n",
              "        box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",
              "        filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",
              "        fill: #FFFFFF;\n",
              "      }\n",
              "    </style>\n",
              "    <button class=\"colab-df-generate\"
onclick=\"generateWithVariable('result')\"\n",
              "            title=\"Generate code using this dataframe.\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0
0 24 24\"\n",
              "       width=\"24px\">\n",
              "    <path
d=\"M7,19H8.4L18.45,9,17,7.55,7,17.6ZM5,21V16.75L18.45,3.32a2,2,0,0,1,2.83,0l1.4,1.43a
1.91,1.91,0,0,1,.58,1.4,1.91,1.91,0,0,1-.58,1.4L9.25,21ZM18.45,9,17,7.55Zm-
12,3A5.31,5.31,0,0,0,4.9,8.1,5.31,5.31,0,0,0,1,6.5,5.31,5.31,0,0,0,4.9,4.9,5.31,5.31,0,
0,0,6.5,1,5.31,5.31,0,0,0,8.1,4.9,5.31,5.31,0,0,0,12,6.5,5.46,5.46,0,0,0,6.5,12Z\"/>\n
",
              "  </svg>\n",
              "    </button>\n",
              "    <script>\n",
              "      (() => {\n",
              "      const buttonEl =\n",
              "        document.querySelector('#id_a1f39e2b-4448-42e5-9989-
72cc8c61f9a3 button.colab-df-generate');\n",
              "      buttonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "\n",
              "      buttonEl.onclick = () => {\n",
              "        google.colab.notebook.generateWithVariable('result');\n",
              "      }\n",
              "      })();\n",
              "    </script>\n",
              "  </div>\n",
              "\n",
              "    </div>\n",
              "  </div>\n"
            ]
          },
          "metadata": {},
          "execution_count": 1368
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "118 married people have bad credit"
      ],
      "metadata": {
        "id": "PwZMEsuJZctn"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "## Q6.\n",
        " What is the highest education level and what is the total count?\n"
      ],
      "metadata": {
        "id": "1rpRh-epYPos"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "# categories of Education\n",
        "df_sql[\"EDUCATION\"].unique()"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "ruoHmz5qZvM8",
        "outputId": "e460f3d7-99e9-4509-8b66-15bf39b565e3"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "array(['Higher education', 'Secondary / secondary special',\n",
              "       'Lower secondary', 'Incomplete higher', 'Academic degree'],\n",
              "      dtype=object)"
            ]
          },
          "metadata": {},
          "execution_count": 1369
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "The highest education level is Academic degree"
      ],
      "metadata": {
        "id": "fjhvg2BPZpBI"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "query = \"SELECT COUNT(*) count_highest_education FROM credit_data WHERE EDUCATION = 'Academic degree'\"\n",
        "result = connection.execute(query).fetchdf()\n",
        "result"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 89
        },
        "id": "01qXpVNGZk0q",
        "outputId": "0470e37c-3360-4343-cce3-c34a16b8147c"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "   count_highest_education\n",
              "0                        2"
            ],
            "text/html": [
              "\n",
              "  <div id=\"df-eac3cec8-ec47-447c-9606-2e1405de765b\" class=\"colab-df-container\">\n",
              "    <div>\n",
              "<style scoped>\n",
              "    .dataframe tbody tr th:only-of-type {\n",
              "        vertical-align: middle;\n",
              "    }\n",
              "\n",
              "    .dataframe tbody tr th {\n",
              "        vertical-align: top;\n",
              "    }\n",
              "\n",
              "    .dataframe thead th {\n",
              "        text-align: right;\n",
              "    }\n",
              "</style>\n",
              "<table border=\"1\" class=\"dataframe\">\n",
              "  <thead>\n",
              "    <tr style=\"text-align: right;\">\n",
              "      <th></th>\n",
              "      <th>count_highest_education</th>\n",
              "    </tr>\n",
              "  </thead>\n",
              "  <tbody>\n",
              "    <tr>\n",
              "      <th>0</th>\n",
              "      <td>2</td>\n",
              "    </tr>\n",
              "  </tbody>\n",
              "</table>\n",
              "</div>\n",
              "    <div class=\"colab-df-buttons\">\n",
              "\n",
              "  <div class=\"colab-df-container\">\n",
              "    <button class=\"colab-df-convert\"
onclick=\"convertToInteractive('df-eac3cec8-ec47-447c-9606-2e1405de765b')\"\n",
              "            title=\"Convert this dataframe to an interactive
table.\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\" viewBox=\"0
-960 960 960\">\n",
              "    <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220
220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-
160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",
              "  </svg>\n",
              "    </button>\n",
              "\n",
              "  <style>\n",
              "    .colab-df-container {\n",
              "      display:flex;\n",
              "      gap: 12px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert {\n",
              "      background-color: #E8F0FE;\n",
              "      border: none;\n",
              "      border-radius: 50%;\n",
              "      cursor: pointer;\n",
              "      display: none;\n",
              "      fill: #1967D2;\n",
              "      height: 32px;\n",
              "      padding: 0 0 0 0;\n",
              "      width: 32px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert:hover {\n",
              "      background-color: #E2EBFA;\n",
              "      box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n",
              "      fill: #174EA6;\n",
              "    }\n",
              "\n",
              "    .colab-df-buttons div {\n",
              "      margin-bottom: 4px;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert {\n",
              "      background-color: #3B4455;\n",
              "      fill: #D2E3FC;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert:hover {\n",
              "      background-color: #434B5C;\n",
              "      box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",
              "      filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",
              "      fill: #FFFFFF;\n",
              "    }\n",
              "  </style>\n",
              "\n",
              "    <script>\n",
              "      const buttonEl =\n",
              "        document.querySelector('#df-eac3cec8-ec47-447c-9606-
2e1405de765b button.colab-df-convert');\n",
              "      buttonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "\n",
              "      async function convertToInteractive(key) {\n",
              "        const element = document.querySelector('#df-eac3cec8-ec47-447c-
9606-2e1405de765b');\n",
              "        const dataTable =\n",
              "          await
google.colab.kernel.invokeFunction('convertToInteractive',\n",
              "                                                    [key], {});\n",
              "        if (!dataTable) return;\n",
              "\n",
              "        const docLinkHtml = 'Like what you see? Visit the ' +\n",
              "          '<a target=\"_blank\"
href=https://colab.research.google.com/notebooks/data_table.ipynb>data table
notebook</a>'\n",
              "          + ' to learn more about interactive tables.';\n",
              "        element.innerHTML = '';\n",
              "        dataTable['output_type'] = 'display_data';\n",
              ″        await google.colab.output.renderOutput(dataTable, element);\n″,
              ″        const docLink = document.createElement(’div’);\n″,
              ″        docLink.innerHTML = docLinkHtml;\n″,
              ″        element.appendChild(docLink);\n″,
              ″      }\n″,
              ″    </script>\n″,
              ″  </div>\n″,
              ″\n″,
              ″\n″,
              ″  <div id=\″id_5d6509b0-05a5-4189-a8ca-1abdf33fc1fe\″>\n″,
              ″    <style>\n″,
              ″      .colab-df-generate {\n″,
              ″        background-color: #E8F0FE;\n″,
              ″        border: none;\n″,
              ″        border-radius: 50%;\n″,
              ″        cursor: pointer;\n″,
              ″        display: none;\n″,
              ″        fill: #1967D2;\n″,
              ″        height: 32px;\n″,
              ″        padding: 0 0 0 0;\n″,
              ″        width: 32px;\n″,
              ″      }\n″,
              ″\n″,
              ″      .colab-df-generate:hover {\n″,
              ″        background-color: #E2EBFA;\n″,
              ″        box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n″,
              ″        fill: #174EA6;\n″,
              ″      }\n″,
              ″\n″,
              ″      [theme=dark] .colab-df-generate {\n″,
              ″        background-color: #3B4455;\n″,
              ″        fill: #D2E3FC;\n″,
              ″      }\n″,
              ″\n″,
              ″      [theme=dark] .colab-df-generate:hover {\n″,
              ″        background-color: #434B5C;\n″,
              ″        box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n″,
              ″        filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n″,
              ″        fill: #FFFFFF;\n″,
              ″      }\n″,
              ″    </style>\n″,
              ″    <button class=\″colab-df-generate\″
onclick=\″generateWithVariable(’result’)\″\n″,
              ″            title=\″Generate code using this dataframe.\″\n″,
              ″            style=\″display:none;\″>\n″,
              ″\n″,
              ″  <svg xmlns=\″http://www.w3.org/2000/svg\″ height=\″24px\″viewBox=\″0
0 24 24\″\n″,
              "       width=\"24px\">\n",
              "    <path
d=\"M7,19H8.4L18.45,9,17,7.55,7,17.6ZM5,21V16.75L18.45,3.32a2,2,0,0,1,2.83,0l1.4,1.43a
1.91,1.91,0,0,1,.58,1.4,1.91,1.91,0,0,1-.58,1.4L9.25,21ZM18.45,9,17,7.55Zm-
12,3A5.31,5.31,0,0,0,4.9,8.1,5.31,5.31,0,0,0,1,6.5,5.31,5.31,0,0,0,4.9,4.9,5.31,5.31,0,
0,0,6.5,1,5.31,5.31,0,0,0,8.1,4.9,5.31,5.31,0,0,0,12,6.5,5.46,5.46,0,0,0,6.5,12Z\"/>\n
",
              "  </svg>\n",
              "    </button>\n",
              "    <script>\n",
              "      (() => {\n",
              "      const buttonEl =\n",
              "        document.querySelector('#id_5d6509b0-05a5-4189-a8ca-
1abdf33fc1fe button.colab-df-generate');\n",
              "      buttonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "\n",
              "      buttonEl.onclick = () => {\n",
              "        google.colab.notebook.generateWithVariable('result');\n",
              "      }\n",
              "      })();\n",
              "    </script>\n",
              "  </div>\n",
              "\n",
              "    </div>\n",
              "  </div>\n"
            ]
          },
          "metadata": {},
          "execution_count": 1370
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "Only two people have the highest education degree"
      ],
      "metadata": {
        "id": "BPBYDDUVaC9j"
      }
    },
    {
      "cell_type": "markdown",
      "source": [
        "## Q7.\n",
        "Between married males and females, who is having more bad credit?"
      ],
      "metadata": {
        "id": "sE_dQunTZkSv"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "query = \"SELECT GENDER, SUM(label) AS num_bad_credit FROM credit_data WHERE Marital_status IN('Married','Civil marriage') GROUP BY GENDER\"\n",
        "result = connection.execute(query).fetchdf()\n",
        "result"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 125
        },
        "id": "jFoyFKsaXZb_",
        "outputId": "d31d1b39-e2b8-4fc4-a6fd-5908106233bb"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "  GENDER  num_bad_credit\n",
              "0      M            54.0\n",
              "1      F            64.0"
            ],
            "text/html": [
              "\n",
              "  <div id=\"df-054a54f4-fd07-4c07-afb6-fe07fe8f2f83\" class=\"colab-df-container\">\n",
              "    <div>\n",
              "<style scoped>\n",
              "    .dataframe tbody tr th:only-of-type {\n",
              "        vertical-align: middle;\n",
              "    }\n",
              "\n",
              "    .dataframe tbody tr th {\n",
              "        vertical-align: top;\n",
              "    }\n",
              "\n",
              "    .dataframe thead th {\n",
              "        text-align: right;\n",
              "    }\n",
              "</style>\n",
              "<table border=\"1\" class=\"dataframe\">\n",
              "  <thead>\n",
              ″    <tr style=\″text-align: right;\″>\n″,
              ″      <th></th>\n″,
              ″      <th>GENDER</th>\n″,
              ″      <th>num_bad_credit</th>\n″,
              ″    </tr>\n″,
              ″  </thead>\n″,
              ″  <tbody>\n″,
              ″    <tr>\n″,
              ″      <th>0</th>\n″,
              ″      <td>M</td>\n″,
              ″      <td>54.0</td>\n″,
              ″    </tr>\n″,
              ″    <tr>\n″,
              ″      <th>1</th>\n″,
              ″      <td>F</td>\n″,
              ″      <td>64.0</td>\n″,
              ″    </tr>\n″,
              ″  </tbody>\n″,
              ″</table>\n″,
              ″</div>\n″,
              ″    <div class=\″colab-df-buttons\″>\n″,
              ″\n″,
              ″  <div class=\″colab-df-container\″>\n″,
              ″    <button class=\″colab-df-convert\″
onclick=\″convertToInteractive(’df-054a54f4-fd07-4c07-afb6-fe07fe8f2f83’)\″\n″,
              ″            title=\″Convert this dataframe to an interactive
table.\″\n″,
              ″            style=\″display:none;\″>\n″,
              ″\n″,
              ″  <svg xmlns=\″http://www.w3.org/2000/svg\″ height=\″24px\″ viewBox=\″0
-960 960 960\″>\n″,
              ″    <path d=\″M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220
220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-
160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\″/>\n″,
              ″  </svg>\n″,
              ″    </button>\n″,
              ″\n″,
              ″  <style>\n″,
              ″    .colab-df-container {\n″,
              ″      display:flex;\n″,
              ″      gap: 12px;\n″,
              ″    }\n″,
              ″\n″,
              ″    .colab-df-convert {\n″,
              ″      background-color: #E8F0FE;\n″,
              ″      border: none;\n″,
              ″      border-radius: 50%;\n″,
              ″      cursor: pointer;\n″,
              ″      display: none;\n″,
              "      fill: #1967D2;\n",
              "      height: 32px;\n",
              "      padding: 0 0 0 0;\n",
              "      width: 32px;\n",
              "    }\n",
              "\n",
              "    .colab-df-convert:hover {\n",
              "      background-color: #E2EBFA;\n",
              "      box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n",
              "      fill: #174EA6;\n",
              "    }\n",
              "\n",
              "    .colab-df-buttons div {\n",
              "      margin-bottom: 4px;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert {\n",
              "      background-color: #3B4455;\n",
              "      fill: #D2E3FC;\n",
              "    }\n",
              "\n",
              "    [theme=dark] .colab-df-convert:hover {\n",
              "      background-color: #434B5C;\n",
              "      box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",
              "      filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",
              "      fill: #FFFFFF;\n",
              "    }\n",
              "  </style>\n",
              "\n",
              "    <script>\n",
              "      const buttonEl =\n",
              "        document.querySelector('#df-054a54f4-fd07-4c07-afb6-
fe07fe8f2f83 button.colab-df-convert');\n",
              "      buttonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "\n",
              "      async function convertToInteractive(key) {\n",
              "        const element = document.querySelector('#df-054a54f4-fd07-4c07-
afb6-fe07fe8f2f83');\n",
              "        const dataTable =\n",
              "          await
google.colab.kernel.invokeFunction('convertToInteractive',\n",
              "                                                    [key], {});\n",
              "        if (!dataTable) return;\n",
              "\n",
              "        const docLinkHtml = 'Like what you see? Visit the ' +\n",
              "          '<a target=\"_blank\" href=https://colab.research.google.com/notebooks/data_table.ipynb>data table notebook</a>'\n",
              "          + ' to learn more about interactive tables.';\n",
              "        element.innerHTML = '';\n",
              "        dataTable['output_type'] = 'display_data';\n",
              "        await google.colab.output.renderOutput(dataTable, element);\n",
              "        const docLink = document.createElement('div');\n",
              "        docLink.innerHTML = docLinkHtml;\n",
              "        element.appendChild(docLink);\n",
              "      }\n",
              "    </script>\n",
              "  </div>\n",
              "\n",
              "\n",
              "<div id=\"df-3966255a-2f16-4c32-8c58-52a7f42f244b\">\n",
              "  <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-3966255a-2f16-4c32-8c58-52a7f42f244b')\"\n",
              "            title=\"Suggest charts\"\n",
              "            style=\"display:none;\">\n",
              "\n",
              "<svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0 0 24 24\"\n",
              "     width=\"24px\">\n",
              "    <g>\n",
              "        <path d=\"M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\"/>\n",
              "    </g>\n",
              "</svg>\n",
              "  </button>\n",
              "\n",
              "<style>\n",
              "  .colab-df-quickchart {\n",
              "      --bg-color: #E8F0FE;\n",
              "      --fill-color: #1967D2;\n",
              "      --hover-bg-color: #E2EBFA;\n",
              "      --hover-fill-color: #174EA6;\n",
              "      --disabled-fill-color: #AAA;\n",
              "      --disabled-bg-color: #DDD;\n",
              "  }\n",
              "\n",
              "  [theme=dark] .colab-df-quickchart {\n",
              "      --bg-color: #3B4455;\n",
              "      --fill-color: #D2E3FC;\n",
              "      --hover-bg-color: #434B5C;\n",
              "      --hover-fill-color: #FFFFFF;\n",
              "      --disabled-bg-color: #3B4455;\n",
              "      --disabled-fill-color: #666;\n",
              "  }\n",
              ″\n″,
              ″  .colab-df-quickchart {\n″,
              ″    background-color: var(--bg-color);\n″,
              ″    border: none;\n″,
              ″    border-radius: 50%;\n″,
              ″    cursor: pointer;\n″,
              ″    display: none;\n″,
              ″    fill: var(--fill-color);\n″,
              ″    height: 32px;\n″,
              ″    padding: 0;\n″,
              ″    width: 32px;\n″,
              ″  }\n″,
              ″\n″,
              ″  .colab-df-quickchart:hover {\n″,
              ″    background-color: var(--hover-bg-color);\n″,
              ″    box-shadow: 0 1px 2px rgba(60, 64, 67, 0.3), 0 1px 3px 1px rgba(60,
64, 67, 0.15);\n″,
              ″    fill: var(--button-hover-fill-color);\n″,
              ″  }\n″,
              ″\n″,
              ″  .colab-df-quickchart-complete:disabled,\n″,
              ″  .colab-df-quickchart-complete:disabled:hover {\n″,
              ″    background-color: var(--disabled-bg-color);\n″,
              ″    fill: var(--disabled-fill-color);\n″,
              ″    box-shadow: none;\n″,
              ″  }\n″,
              ″\n″,
              ″  .colab-df-spinner {\n″,
              ″    border: 2px solid var(--fill-color);\n″,
              ″    border-color: transparent;\n″,
              ″    border-bottom-color: var(--fill-color);\n″,
              ″    animation:\n″,
              ″      spin 1s steps(1) infinite;\n″,
              ″  }\n″,
              ″\n″,
              ″  @keyframes spin {\n″,
              ″    0% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-bottom-color: var(--fill-color);\n″,
              ″      border-left-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    20% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-left-color: var(--fill-color);\n″,
              ″      border-top-color: var(--fill-color);\n″,
              ″    }\n″,
              ″    30% {\n″,
              ″      border-color: transparent;\n″,
              ″      border-left-color: var(--fill-color);\n″,
              "      border-top-color: var(--fill-color);\n",
              "      border-right-color: var(--fill-color);\n",
              "    }\n",
              "    40% {\n",
              "      border-color: transparent;\n",
              "      border-right-color: var(--fill-color);\n",
              "      border-top-color: var(--fill-color);\n",
              "    }\n",
              "    60% {\n",
              "      border-color: transparent;\n",
              "      border-right-color: var(--fill-color);\n",
              "    }\n",
              "    80% {\n",
              "      border-color: transparent;\n",
              "      border-right-color: var(--fill-color);\n",
              "      border-bottom-color: var(--fill-color);\n",
              "    }\n",
              "    90% {\n",
              "      border-color: transparent;\n",
              "      border-bottom-color: var(--fill-color);\n",
              "    }\n",
              "  }\n",
              "</style>\n",
              "\n",
              "  <script>\n",
              "    async function quickchart(key) {\n",
              "      const quickchartButtonEl =\n",
              "        document.querySelector('#' + key + ' button');\n",
              "      quickchartButtonEl.disabled = true;  // To prevent multiple
clicks.\n",
              "      quickchartButtonEl.classList.add('colab-df-spinner');\n",
              "      try {\n",
              "        const charts = await google.colab.kernel.invokeFunction(\n",
              "            'suggestCharts', [key], {});\n",
              "      } catch (error) {\n",
              "        console.error('Error during call to suggestCharts:', error);\n",
              "      }\n",
              "      quickchartButtonEl.classList.remove('colab-df-spinner');\n",
              "      quickchartButtonEl.classList.add('colab-df-quickchart-
complete');\n",
              "    }\n",
              "    (() => {\n",
              "      let quickchartButtonEl =\n",
              "        document.querySelector('#df-3966255a-2f16-4c32-8c58-
52a7f42f244b button');\n",
              "      quickchartButtonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "    })();\n",
              "  </script>\n",
                ″</div>\n″,
                ″\n″,
                ″  <div id=\″id_9b1c36d1-6095-42ba-8fea-16c634460ffd\″>\n″,
                ″    <style>\n″,
                ″      .colab-df-generate {\n″,
                ″        background-color: #E8F0FE;\n″,
                ″        border: none;\n″,
                ″        border-radius: 50%;\n″,
                ″        cursor: pointer;\n″,
                ″        display: none;\n″,
                ″        fill: #1967D2;\n″,
                ″        height: 32px;\n″,
                ″        padding: 0 0 0 0;\n″,
                ″        width: 32px;\n″,
                ″      }\n″,
                ″\n″,
                ″      .colab-df-generate:hover {\n″,
                ″        background-color: #E2EBFA;\n″,
                ″        box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n″,
                ″        fill: #174EA6;\n″,
                ″      }\n″,
                ″\n″,
                ″      [theme=dark] .colab-df-generate {\n″,
                ″        background-color: #3B4455;\n″,
                ″        fill: #D2E3FC;\n″,
                ″      }\n″,
                ″\n″,
                ″      [theme=dark] .colab-df-generate:hover {\n″,
                ″        background-color: #434B5C;\n″,
                ″        box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n″,
                ″        filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n″,
                ″        fill: #FFFFFF;\n″,
                ″      }\n″,
                ″    </style>\n″,
                ″    <button class=\″colab-df-generate\″
onclick=\″generateWithVariable('result')\″\n″,
                ″            title=\″Generate code using this dataframe.\″\n″,
                ″            style=\″display:none;\″>\n″,
                ″\n″,
                ″  <svg xmlns=\″http://www.w3.org/2000/svg\″ height=\″24px\″viewBox=\″0
0 24 24\″\n″,
                ″       width=\″24px\″>\n″,
                ″    <path
d=\″M7,19H8.4L18.45,9,17,7.55,7,17.6ZM5,21V16.75L18.45,3.32a2,2,0,0,1,2.83,0l1.4,1.43a
1.91,1.91,0,0,1,.58,1.4,1.91,1.91,0,0,1-.58,1.4L9.25,21ZM18.45,9,17,7.55Zm-
12,3A5.31,5.31,0,0,0,4.9,8.1,5.31,5.31,0,0,0,1,6.5,5.31,5.31,0,0,0,4.9,4.9,5.31,5.31,0,
0,0,6.5,1,5.31,5.31,0,0,0,8.1,4.9,5.31,5.31,0,0,0,12,6.5,5.46,5.46,0,0,0,6.5,12Z\″/>\n
″,
              "  </svg>\n",
              "    </button>\n",
              "    <script>\n",
              "      (() => {\n",
              "      const buttonEl =\n",
              "        document.querySelector('#id_9b1c36d1-6095-42ba-8fea-
16c634460ffd button.colab-df-generate');\n",
              "      buttonEl.style.display =\n",
              "        google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
              "\n",
              "      buttonEl.onclick = () => {\n",
              "        google.colab.notebook.generateWithVariable('result');\n",
              "      }\n",
              "      })();\n",
              "    </script>\n",
              "  </div>\n",
              "\n",
              "    </div>\n",
              "  </div>\n"
            ]
          },
          "metadata": {},
          "execution_count": 1374
        }
      ]
    },
    {
      "cell_type": "markdown",
      "source": [
        "Among married males and females, married females have more bad credit"
      ],
      "metadata": {
        "id": "d80eNWGIgYmS"
      }
    }
  ],
  "metadata": {
    "colab": {
      "provenance": [],
      "authorship_tag": "ABX9TyPE5nnmDMZr6xex5fRiaeQE",
      "include_colab_link": true
    },
    "kernelspec": {
      "display_name": "Python 3",
      "name": "python3"
    },
    "language_info": {
      "name": "python"
    }
  },
  "nbformat": 4,
  "nbformat_minor": 0
}
```