ASSIGNMENT-2

REG NO:413019106301 NAME : E.ABHIRAMI


```
{
  "nbformat": 4,
  "nbformat_minor": 0,
  "metadata": {
    "colab": {
      "provenance": [],
      "collapsed_sections": [],
      "toc_visible": "true"
    },
    "kernelspec": {
      "name": "python3",
      "display_name": "Python 3"
    },
    "language_info": {
      "name": "python"
    }
  },
  "cells": [
    {
      "cell_type": "code",
      "execution_count": "null",
      "metadata": {
        "id": "aOqC2tJgN5MW"
      },
      "outputs": [],
      "source": [
        "import pandas as pd\n",
        "import numpy as np"
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "df=pd.read_csv('/content/Churn_Modelling.csv')"
      ],
      "metadata": {
        "id": "3oMGqpVzR57Z"
      },
      "execution_count": "null",
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "df.head()"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 270
        },
        "id": "HiIi2COjSCcO",
        "outputId": "46b897b4-2ce0-4bab-dbb4-0eb9e1b24a37"
      },
      "execution_count": "null",
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
```

```
            "text/plain": [
              "   RowNumber  CustomerId   Surname  CreditScore Geography  Gender
Age  \\\n",
              "0          1    15634602  Hargrave          619    France  Female
42   \n",
              "1          2    15647311      Hill          608     Spain  Female
41   \n",
              "2          3    15619304      Onio          502    France  Female
42   \n",
              "3          4    15701354      Boni          699    France  Female
39   \n",
              "4          5    15737888  Mitchell          850     Spain  Female
43   \n",
              "\n",
              "   Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \\\
n",
              "0       2       0.00              1          1               1   \
n",
              "1       1   83807.86              1          0               1   \
n",
              "2       8  159660.80              3          1               0   \
n",
              "3       1       0.00              2          0               0   \
n",
              "4       2  125510.82              1          1               1   \
n",
              "\n",
              "   EstimatedSalary  Exited  \n",
              "0        101348.88       1  \n",
              "1        112542.58       0  \n",
              "2        113931.57       1  \n",
              "3         93826.63       0  \n",
              "4         79084.10       0  "
            ],
            "text/html": [
              "\n",
              "  <div id=\"df-716ad4c0-ba57-44e2-b3f6-a2ec59ce364c\">\n",
              "    <div 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>RowNumber</th>\n",
              "      <th>CustomerId</th>\n",
              "      <th>Surname</th>\n",
```

```
"      <th>CreditScore</th>\n",
"      <th>Geography</th>\n",
"      <th>Gender</th>\n",
"      <th>Age</th>\n",
"      <th>Tenure</th>\n",
"      <th>Balance</th>\n",
"      <th>NumOfProducts</th>\n",
"      <th>HasCrCard</th>\n",
"      <th>IsActiveMember</th>\n",
"      <th>EstimatedSalary</th>\n",
"      <th>Exited</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>0</th>\n",
"      <td>1</td>\n",
"      <td>15634602</td>\n",
"      <td>Hargrave</td>\n",
"      <td>619</td>\n",
"      <td>France</td>\n",
"      <td>Female</td>\n",
"      <td>42</td>\n",
"      <td>2</td>\n",
"      <td>0.00</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>101348.88</td>\n",
"      <td>1</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>1</th>\n",
"      <td>2</td>\n",
"      <td>15647311</td>\n",
"      <td>Hill</td>\n",
"      <td>608</td>\n",
"      <td>Spain</td>\n",
"      <td>Female</td>\n",
"      <td>41</td>\n",
"      <td>1</td>\n",
"      <td>83807.86</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>1</td>\n",
"      <td>112542.58</td>\n",
"      <td>0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>2</th>\n",
"      <td>3</td>\n",
"      <td>15619304</td>\n",
"      <td>Onio</td>\n",
"      <td>502</td>\n",
"      <td>France</td>\n",
"      <td>Female</td>\n",
"      <td>42</td>\n",
"      <td>8</td>\n",
"      <td>159660.80</td>\n",
```

```
              "      <td>3</td>\n",
              "      <td>1</td>\n",
              "      <td>0</td>\n",
              "      <td>113931.57</td>\n",
              "      <td>1</td>\n",
              "    </tr>\n",
              "    <tr>\n",
              "      <th>3</th>\n",
              "      <td>4</td>\n",
              "      <td>15701354</td>\n",
              "      <td>Boni</td>\n",
              "      <td>699</td>\n",
              "      <td>France</td>\n",
              "      <td>Female</td>\n",
              "      <td>39</td>\n",
              "      <td>1</td>\n",
              "      <td>0.00</td>\n",
              "      <td>2</td>\n",
              "      <td>0</td>\n",
              "      <td>0</td>\n",
              "      <td>93826.63</td>\n",
              "      <td>0</td>\n",
              "    </tr>\n",
              "    <tr>\n",
              "      <th>4</th>\n",
              "      <td>5</td>\n",
              "      <td>15737888</td>\n",
              "      <td>Mitchell</td>\n",
              "      <td>850</td>\n",
              "      <td>Spain</td>\n",
              "      <td>Female</td>\n",
              "      <td>43</td>\n",
              "      <td>2</td>\n",
              "      <td>125510.82</td>\n",
              "      <td>1</td>\n",
              "      <td>1</td>\n",
              "      <td>1</td>\n",
              "      <td>79084.10</td>\n",
              "      <td>0</td>\n",
              "    </tr>\n",
              "  </tbody>\n",
              "</table>\n",
              "</div>\n",
              "      <button class=\"colab-df-convert\"
onclick=\"convertToInteractive('df-716ad4c0-ba57-44e2-b3f6-a2ec59ce364c')\"\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 0 24 24\"\n",
              "       width=\"24px\">\n",
              "    <path d=\"M0 0h24v24H0V0z\" fill=\"none\"/>\n",
              "    <path d=\"M18.56 5.44l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-
2.06-.94 2.06-2.06.94zm-11 1L8.5 8.5l.94-2.06 2.06-.94-2.06-.94L8.5 2.5l-.94 2.06-
2.06.94zm10 10l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94
2.06-2.06.94z\"/><path d=\"M17.41 7.96l-1.37-1.37c-.4-.4-.92-.59-1.43-.59-.52 0-
1.04.2-1.43.59L10.3 9.45l-7.72 7.72c-.78.78-.78 2.05 0 2.83L4 21.41c.39.39.9.59
1.41.59.51 0 1.02-.2 1.41-.59l7.78-7.78 2.81-2.81c.8-.78.8-2.07 0-2.86zM5.41 20L4
```

```
18.59l7.72-7.72 1.47 1.35L5.41 20z\"/>\n",
               "  </svg>\n",
               "      </button>\n",
               "      \n",
               "  <style>\n",
               "    .colab-df-container {\n",
               "      display:flex;\n",
               "      flex-wrap:wrap;\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",
               "    [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-716ad4c0-ba57-44e2-b3f6-
a2ec59ce364c 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-716ad4c0-ba57-
44e2-b3f6-a2ec59ce364c');\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",
              "  </div>\n",
              "  "
            ]
          },
          "metadata": {},
          "execution_count": 4
        }
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "df.info()"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "_pb9gjJBSyUn",
        "outputId": "3741a4a4-e9ae-436d-d0df-902fe080ba98"
      },
      "execution_count": "null",
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
            "<class 'pandas.core.frame.DataFrame'>\n",
            "RangeIndex: 10000 entries, 0 to 9999\n",
            "Data columns (total 14 columns):\n",
            " #   Column           Non-Null Count  Dtype  \n",
            "---  ------           --------------  -----  \n",
            " 0   RowNumber        10000 non-null  int64  \n",
            " 1   CustomerId       10000 non-null  int64  \n",
            " 2   Surname          10000 non-null  object \n",
            " 3   CreditScore      10000 non-null  int64  \n",
            " 4   Geography        10000 non-null  object \n",
            " 5   Gender           10000 non-null  object \n",
            " 6   Age              10000 non-null  int64  \n",
            " 7   Tenure           10000 non-null  int64  \n",
            " 8   Balance          10000 non-null  float64\n",
            " 9   NumOfProducts    10000 non-null  int64  \n",
            " 10  HasCrCard        10000 non-null  int64  \n",
            " 11  IsActiveMember   10000 non-null  int64  \n",
```

```
        " 12  EstimatedSalary  10000 non-null  float64\n",
        " 13  Exited           10000 non-null  int64  \n",
        "dtypes: float64(2), int64(9), object(3)\n",
        "memory usage: 1.1+ MB\n"
      ]
    }
  ]
},
{
  "cell_type": "code",
  "source": [
    "df.isnull().sum()"
  ],
  "metadata": {
    "colab": {
      "base_uri": "https://localhost:8080/"
    },
    "id": "0N2u84f7S4kt",
    "outputId": "96cf3f59-382a-4f2c-fd52-01602d044197"
  },
  "execution_count": "null",
  "outputs": [
    {
      "output_type": "execute_result",
      "data": {
        "text/plain": [
          "RowNumber          0\n",
          "CustomerId         0\n",
          "Surname            0\n",
          "CreditScore        0\n",
          "Geography          0\n",
          "Gender             0\n",
          "Age                0\n",
          "Tenure             0\n",
          "Balance            0\n",
          "NumOfProducts      0\n",
          "HasCrCard          0\n",
          "IsActiveMember     0\n",
          "EstimatedSalary    0\n",
          "Exited             0\n",
          "dtype: int64"
        ]
      },
      "metadata": {},
      "execution_count": 6
    }
  ]
},
{
  "cell_type": "code",
  "source": [
    "from sklearn.preprocessing import LabelEncoder"
  ],
  "metadata": {
    "id": "lr5jBVUdTZpe"
  },
  "execution_count": "null",
  "outputs": []
},
```

```
    {
      "cell_type": "code",
      "source": [
        "le=LabelEncoder()"
      ],
      "metadata": {
        "id": "D_W6c4u4Tm_2"
      },
      "execution_count": "null",
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "df['Surname']=le.fit_transform(df['Surname'])\n",
        "df['Geography']=le.fit_transform(df['Geography'])\n",
        "df['Gender']=le.fit_transform(df['Gender'])\n"
      ],
      "metadata": {
        "id": "TA3ny5METyVH"
      },
      "execution_count": "null",
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "df.head()"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 270
        },
        "id": "eOJd_mGDUJm8",
        "outputId": "adac78ef-94e8-4ba2-e3cb-a769c2348da0"
      },
      "execution_count": "null",
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "   RowNumber  CustomerId  Surname  CreditScore  Geography  Gender
Age  \\\n",
              "0          1    15634602     1115          619          0       0
42   \n",
              "1          2    15647311     1177          608          2       0
41   \n",
              "2          3    15619304     2040          502          0       0
42   \n",
              "3          4    15701354      289          699          0       0
39   \n",
              "4          5    15737888     1822          850          2       0
43   \n",
              "\n",
              "   Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \\\
n",
              "0       2       0.00              1          1               1   \
```

```
n",
              "1       1   83807.86              1          0               1   \
n",
              "2       8  159660.80              3          1               0   \
n",
              "3       1       0.00              2          0               0   \
n",
              "4       2  125510.82              1          1               1   \
n",
              "\n",
              "   EstimatedSalary  Exited  \n",
              "0        101348.88       1  \n",
              "1        112542.58       0  \n",
              "2        113931.57       1  \n",
              "3         93826.63       0  \n",
              "4         79084.10       0  "
            ],
            "text/html": [
              "\n",
              "  <div id=\"df-caf96f38-7e5f-4f89-9308-940d23387ab6\">\n",
              "    <div 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>RowNumber</th>\n",
              "      <th>CustomerId</th>\n",
              "      <th>Surname</th>\n",
              "      <th>CreditScore</th>\n",
              "      <th>Geography</th>\n",
              "      <th>Gender</th>\n",
              "      <th>Age</th>\n",
              "      <th>Tenure</th>\n",
              "      <th>Balance</th>\n",
              "      <th>NumOfProducts</th>\n",
              "      <th>HasCrCard</th>\n",
              "      <th>IsActiveMember</th>\n",
              "      <th>EstimatedSalary</th>\n",
              "      <th>Exited</th>\n",
              "    </tr>\n",
              "  </thead>\n",
              "  <tbody>\n",
              "    <tr>\n",
              "      <th>0</th>\n",
              "      <td>1</td>\n",
```

```
"      <td>15634602</td>\n",
"      <td>1115</td>\n",
"      <td>619</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>42</td>\n",
"      <td>2</td>\n",
"      <td>0.00</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>101348.88</td>\n",
"      <td>1</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>1</th>\n",
"      <td>2</td>\n",
"      <td>15647311</td>\n",
"      <td>1177</td>\n",
"      <td>608</td>\n",
"      <td>2</td>\n",
"      <td>0</td>\n",
"      <td>41</td>\n",
"      <td>1</td>\n",
"      <td>83807.86</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>1</td>\n",
"      <td>112542.58</td>\n",
"      <td>0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>2</th>\n",
"      <td>3</td>\n",
"      <td>15619304</td>\n",
"      <td>2040</td>\n",
"      <td>502</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>42</td>\n",
"      <td>8</td>\n",
"      <td>159660.80</td>\n",
"      <td>3</td>\n",
"      <td>1</td>\n",
"      <td>0</td>\n",
"      <td>113931.57</td>\n",
"      <td>1</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>3</th>\n",
"      <td>4</td>\n",
"      <td>15701354</td>\n",
"      <td>289</td>\n",
"      <td>699</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>39</td>\n",
"      <td>1</td>\n",
"      <td>0.00</td>\n",
```

```
"      <td>2</td>\n",
"      <td>0</td>\n",
"      <td>0</td>\n",
"      <td>93826.63</td>\n",
"      <td>0</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>4</th>\n",
"      <td>5</td>\n",
"      <td>15737888</td>\n",
"      <td>1822</td>\n",
"      <td>850</td>\n",
"      <td>2</td>\n",
"      <td>0</td>\n",
"      <td>43</td>\n",
"      <td>2</td>\n",
"      <td>125510.82</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>79084.10</td>\n",
"      <td>0</td>\n",
"    </tr>\n",
"  </tbody>\n",
"</table>\n",
"</div>\n",
"      <button class=\"colab-df-convert\"
onclick=\"convertToInteractive('df-caf96f38-7e5f-4f89-9308-940d23387ab6')\"\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 0 24 24\"\n",
"       width=\"24px\">\n",
"    <path d=\"M0 0h24v24H0V0z\" fill=\"none\"/>\n",
"    <path d=\"M18.56 5.44l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-
2.06-.94 2.06-2.06.94zm-11 1L8.5 8.5l.94-2.06 2.06-.94-2.06-.94L8.5 2.5l-.94 2.06-
2.06.94zm10 10l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94
2.06-2.06.94z\"/><path d=\"M17.41 7.96l-1.37-1.37c-.4-.4-.92-.59-1.43-.59-.52 0-
1.04.2-1.43.59L10.3 9.45l-7.72 7.72c-.78.78-.78 2.05 0 2.83L4 21.41c.39.39.9.59
1.41.59.51 0 1.02-.2 1.41-.59l7.78-7.78 2.81-2.81c.8-.78.8-2.07 0-2.86zM5.41 20L4
18.59l7.72-7.72 1.47 1.35L5.41 20z\"/>\n",
"  </svg>\n",
"      </button>\n",
"      \n",
"  <style>\n",
"    .colab-df-container {\n",
"      display:flex;\n",
"      flex-wrap:wrap;\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",
              "    [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-caf96f38-7e5f-4f89-9308-
940d23387ab6 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-caf96f38-7e5f-
4f89-9308-940d23387ab6');\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",
              "  </div>\n",
              "  "
```

```
        ]
      },
      "metadata": {},
      "execution_count": 10
    }
  ]
},
{
  "cell_type": "code",
  "source": [
    "df.info()"
  ],
  "metadata": {
    "colab": {
      "base_uri": "https://localhost:8080/"
    },
    "id": "RA9ApVLTUL24",
    "outputId": "bbf620f4-2393-4cfe-f29f-53493c52681f"
  },
  "execution_count": "null",
  "outputs": [
    {
      "output_type": "stream",
      "name": "stdout",
      "text": [
        "<class 'pandas.core.frame.DataFrame'>\n",
        "RangeIndex: 10000 entries, 0 to 9999\n",
        "Data columns (total 14 columns):\n",
        " #   Column           Non-Null Count  Dtype  \n",
        "---  ------           --------------  -----  \n",
        " 0   RowNumber        10000 non-null  int64  \n",
        " 1   CustomerId       10000 non-null  int64  \n",
        " 2   Surname          10000 non-null  int64  \n",
        " 3   CreditScore      10000 non-null  int64  \n",
        " 4   Geography        10000 non-null  int64  \n",
        " 5   Gender           10000 non-null  int64  \n",
        " 6   Age              10000 non-null  int64  \n",
        " 7   Tenure           10000 non-null  int64  \n",
        " 8   Balance          10000 non-null  float64\n",
        " 9   NumOfProducts    10000 non-null  int64  \n",
        " 10  HasCrCard        10000 non-null  int64  \n",
        " 11  IsActiveMember   10000 non-null  int64  \n",
        " 12  EstimatedSalary  10000 non-null  float64\n",
        " 13  Exited           10000 non-null  int64  \n",
        "dtypes: float64(2), int64(12)\n",
        "memory usage: 1.1 MB\n"
      ]
    }
  ]
},
{
  "cell_type": "code",
  "source": [
    "import matplotlib.pyplot as plt\n",
    "%matplotlib inline"
  ],
  "metadata": {
    "id": "qRG4VXvfUPTK"
  },
```

```
      "execution_count": "null",
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "import seaborn as sns"
      ],
      "metadata": {
        "id": "saMKDs2_UdM9"
      },
      "execution_count": "null",
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "plt.scatter(df.index,df['Balance'])\n",
        "plt.show()"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 265
        },
        "id": "JoBAzwypUgAP",
        "outputId": "712b31c8-b35a-44d3-bb95-a0fedfd91053"
      },
      "execution_count": "null",
      "outputs": [
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 432x288 with 1 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAAAYoAAAD4CAYAAADy46FuAAAABHNCSVQICAgIfAhkiAAAAAlwSFlzAAALEg
AACxIB0t1+/
AAAADh0RVh0U29mdHdhcmUAbWF0cGxvdGxpYiB2ZXJzaW9uMy4yLjIsIGh0dHA6Ly9tYXRwbG90bGliLm9y
Zy+WH4yJAAAgAElEQVR4nO2df5RcxXXnv7d7WqJHtjUjW+GgRkKYcMRBq0hjZkGOcvYYnCB+BBgDRrAQkyx
rctb2rpE5WotEGwlMFiVaDPau4wEwIJ+colszIJIyELS0GYH1IjG9o0o+uar
1586pevffqdb/
uvp9zRuqufj+qXtWrW3XvrVskhADDMAzD6Mg1OwMMwzBMtmFBwTAMwxhhQcEwDMMYYUHBMAzDGGFBwTAMwx
jpanYGXPOhD31IzJ8/v9nZYBiGaSl27tz5KyHE7KDf2k5QzJ8/H0NDQ83OBsMwTEtBRK/
pfmPVE8MwDGOEBQXDMAxjhAUFwzAMY4QFBcMwDGOEBQXDMAxjpO28npjoDA6XsWHfrxUsGcniJWLV+Agb
5Ss7PFMExGCJ1RENFcItpKRC8S0V4i+rxMX0dEZSLaJf8u8ZxzGxEdIKL9RLTck36acT0Q6Zv
pGIpsn06fL7Afn7fJeFZ2pC4rYn9qA8UoEAUB6p4LYn9mBwuNzsrDEMkxFsVE9jAG4VQpwNYCmAzxLR2fK3
e4QQS+TfMwAgf7sWwEIAFwH4GyLKE1EewNcAXAzgbADADXea7zV/
Javw3gGICbZPpNAI7J9HvkcYxDNmzej0p1fFJapTqODZv3NylHDMNkjVBBIYQ4LIT4ifz8GwA/
A2DSS1wB4GEhxHtCiFcAHABwrvw7IIT4uRDICHAVxBRATgAgCPyfMfADDgudYD8vNjAD4uj2cc8cZIJVI
6wzCdRyRjtlT99AHYIZM+R0QvENH9RNQr00oADnpfkBDAihBjzpU+6lvz9LXm8P183E9EQEQ0dOX
IkSpE6njk9xUjpDMN0HtaCgojeB+BxALcIId4G8HUAZwwBYAuAwgLtTyaEFQoj7hBD9Qoj+2bMDQ5UwGlYtX
4BiIT8prVjIY9XyBU3KEcMwWcPK64mICqgJiQeFEE8AgBDil57
RTZRo06b8G0ENEXXLW4D1eXesQEXUBmCmPZxyhvJvY64lhGB2hgkLaBL4J4GdCiC970k8k8B4Kfy8
yYA3yWiLwOYA+BMAD8GQADOJKLTURMA1wL4j0IIQURbAVyVyNmt3iRgDf81zrRgA/
kr9vEbzJt3MG+kosGBiG0WIzo1gG4I8A7CGiXTLtz1DzWloCQB4FcCfAoAQYi8RPQLgRdQ8pj4rhBgHACL
6HIDNAPIA7hdC7JXX+yKAh4noTgDDqAkmyP//
gYgOADiKmnBhGIZhGgi12wC9v79fDA0NNbsZDAOAkmnBhGIZhGgi12wC9v79fcJhxhmGYaBDRTiFEf9BvvDKbyRy8UpxhsgULCiZTqJXXiahGgWikOgIU
```

FwzQJDgrIZApeKc4w2YMFBZMpeKU4w2QPFhRMpuCV4gyTPVhQMJmCV4ozTPZgYzaTKXilOMNkDxYUTObgle
IMky1Y9cQwDMMYYUHBMAzDGGFBwTAMwxhhQcEwDMMYYUHBMAzDGGGvJ6Yl4cCBTKPp5DbHgoJpOThwINNoO
r3NseqJaTk4cCDTaDq9zbGgYFoODhzINJpOb3MsKJiWgwMHMo2m09scCwqm5eDAgc1ncLiMZeu34PTVT2PZ
+i0YHC43O0up0ultjo3ZGaGTPSqiwoEDm0snGnY7vc2REKLZeXBKf3+/GBoaSnSNRnfa/
hcPqI1W7rpyUeh9syZgOD/
tz7L1W1AO0M2XeorYtvqCJuSIcQER7RRC9Af9xjMKH3FGS0k7I5NHhek6WRvZcX46g0407Hb6gINtFD6ius
Gpzqg8UoHAic4ois427ouXNZc9zk9n4Nqwm3V7h4t3vNVhQeEjaqdt6oxsX4C4L17WRnacn87ApWG3FTph3
Tt+y8ZdmRRsacCCwkfUTlvX6agGb/MCxH3xsuayx/lpPo0YnQ/
0lXDXlYtQ6imCULNN2NjTgmiFWZ9pYJFFwZYGbKPwsWr5gkDDsq7TntNTDDTs5Yms7Q5xPSqi5jVt4uYnLf
1v1p5P2jTSJqPbhTBqXaY163PZpnTvuMIr2NrVjsGCwkfUTlvXGfmFhEL3AsTZ/
jNrLntx8pNm5+bi+Xg7nJnFAoiAkdFq0591EHGdIlwRpy51nXCUWZ9fKJx/
1mw8vrPsrE0FveN+1D3a1XGC3WMdEDR62bB5f2wXwk7ysMiyq2WQ27IXWxdmm/u4qO/TVz+NoLeZALyy/
tJEebQhTl0mcQ3XnU9A4HPoKRawa+2FodfU3Uf3TgM1DcJ4QF+aZjt23U+we2zK6GYDcdUw7Twy8ZNlg3PQ
CN2Li9G6y/
qOMzp32dnEqUubWZ8pj0F1pBv6jlSqGBwuxyqfesd1gi2qBiEpje4nWFCkRFy1R5hxr91mGi5UD2lh85In7
Qhcqoui2mRcdzZx69Kkdg3LY9Tnn1Sw695r3WwjrXbcaDVjqKAgorkAvg3gZNSE9X1CiK8Q0SwAGwHMB/
AqgGuEEMeIiAB8BcAlAEYB/
LEQ4ifyWjcCWCMvfacQ4gGZfg6AbwEoAngGwOeFEEJ3j8SlbhBx7A5hXlRZm2kkHZFm2eAcZsRUxyTB5Ywq
6uAkrLOJWreu6tJ731yASsebR5s68uJihO9SgxCVMBVYWjMYG/
fYMQC3CiHOBrAUwGeJ6GwAqwH8QAhxJoAfyO8AcDGAM+XfzQC+DgCy018L4DwA5wJYS0S98pyvA/
i057yLZLruHm2LruMxeVE1Cxsf+DB3TZeulnHLoMtfkNuyFxcdgWsX3oG+ElYtX4A5PUW8MVKpr+cJwiSk4
qxvcFGX/vsG6f29ede5ls+YFlxvaY3wG9GOvc9GR1rlC51RCCEOAzgsP/
+GiH4GoATgCgAfk4c9AOCHAL4o078talby7UTUQ0SnyGOfE0IcBQAieg7ARUT0QwAfEEJsl+nfBjAA4PuGe
7QtrryoGoHNiNRmFhRn5uWCsPz5R+hpeD1FGYXbjPBtn/ngcDlwtA7UOhtd3a7btDcwD/
683bNiibN1FUGoDlE3iwIaM8L3knY7Dns2aZYvko2CiOYD6AOwA8DJUogAwC9QU00BNSFy0HPaIZlmSj8Uk
A7DPfz5uhm12QvmzZsXpUhNJcxI10wdqA1hapNmu2uGYZO/
tF5+v9vtSYWcUQDZCgCbMqlrBQkJ1dms3LgrMN8jlSpGKtV6HlZu3IVHh17HT15/
y4la1Gbg4+8QTXXUTjY907MppVw+a0FBRO8D8DiAW4QQb9dMETWkPSFVP1vTPYQQ9wG4D6i5x6aZD1fYjma
9ZE2XH2a8zLJHE9C8/
PnrfqRSRbGQN47CbYWuTZl0I9M8UV1dYtKDexEAtr18dEp63AGBaQHrhBCROnxXQj4r7upR7TEusQrhQUQF
1ITEg0KIJ2TyL6VKCfL/N2V6GcBcz+mnyjRT+qkB6aZ7ZBqbMApRQxc0W5cfRFjokayH0GhW/
uKErbAVajZl0l1rQoh6ewqzz9hgI3D978r5Z80ObFN3X7MYr6y/
FNtWX9DQNp+lWFSmOkk7X6GCQnoxfRPAz4QQX/b8tAnAjfLzjQC+50n/
FNVYCuAtqT7aDOBCIuqVRuwLAWyWv71NREvlvT7lu1bQPTKLbcOK47Uw0FfCttUXNOWF0eVHCS/
ghMFdGVBdBY9LK35Rs3YtizOTsRVqNmWyuVZQ3UYlTOAGvSuP7yzjqnNKmRkQZSkWlb9O/
KSZLxvV0zIAfwRgDxEpxeWfAVgP4BEiugnAawCukb89g5pr7AHU3GP/
BACEEEeJ6EsAnpfH3aEM2wA+gxPusd+XfzDcI7PY6oh1q0ezMtq2pT4CfXQ3qhO1EpVHKlj16G5s+ORi3HX
losQhNLIc4iMov0pto1br+vXHtusN/HaMQp5QHT/
RagjA+WfNjlwmWwO6OicsfIXKi789jx4fMy5w070rW/
cdafqqfIXJXf301U83XBWl7nOLxoaUlmqKQ3gYiKObNIVRuGfFEqPulwBcv3Qetu470hR9aFxd7JLbn60bO
L0kCZmg6LvjWRwbnXrttEIjRH0G/
s78neNjkzpzhTcshU3YiqBjcgAmfNdVbebOgUWplFMXlsNftqvOKeGp3YentIOgcByDw2Ws27Q3sM2oMrkO
ORK3bduWP0rIEZcDJz9Jnh2H8IhB3JGsbrQ4s1gIHZkJwBjMzNTIXDfAKCN33QuvS4+SpyAhAaQTXRSA8Rm
EBZ8zldc7q7QZ9QeNtv1CAqi1mQe3v47+02ZFqm9bQ6/pORMwKe9b9x2Z8gyCZtPe2WcQUWfVYW0/
Sdu2CQhoa7h3MTsOc5EV8j6uB5csKDRE9SNXBDWsQo7w9rtVGN4NAOGL6nSNzPRbkgaoNmfZsHm/
1mUzTV2t6dozi4XY19W9sCcVcpGe/
4PbX9fGFQrC2+mGddRRBKFA8tAUOnQDn6AZnY3t5fYn9xqFBBCusvJi0/
kmcdP2C3VdzstykaLpei7cxW3aRRptgQWFBl2F+P3I/
Y0yaJHWO8fHMBE0HPQQtqguzKiWZgMMKmfYFBgAervjd+ZheXonoDOxnVXpnmXU5x9VaRtlpOwyNEWS2eb5
Z83Gd7a/Hphum2dVbtMM0cux0ar1QMem803iBu1/dqPHx7RlWLlxF4ZeO1pXA/
rPdRF2w6ZdpOHezYIiANOqVT9BHbJ3tLhs/
ZZQFUxJqjF0I1QVjiEIU6Ow7TzUimNTcSvVcdz6yG6s3Lir/
sKEGTiPjVaxbP2WSNFAvcwsFrTPrjouJj33NYN7Jj0/06wq6otkev62eI3Fawb34KEdBzEuBPJEuO68ueg/
bZZRrQXUZqa60XiOKNC4mlTdsXXfEev0MCN5lNlnkMoqqM2YjM3L1m+p/
a7xHOkJGcgEPbtCjqY4FSi8akBg6gxUR1g+vNiowtJwiOGtUH2YVq3qiNtZAzWD77bVF2DrviNaI7iK3RPE
nJ5i5PUAfrfEkUq4Wgyoxd1Rbow2I0PIY2/
ZuAvXf+NHgfc2+X+HeWSqZzs4XA4Usjp3wagv0ujxMe3LbOM06nXxXDO4B9/Z/
nq9fY0Lge9sfx1feGRXqJvohk8uxg1L5wXe01s33ueZ1L0zygAlbK1PVGHrrV9dm9HVJcnjBPQDINMrPjhc
xq2P7J7y7KoTAjOm6cfXAsCtj+zGLRt3WYUiUfmwdQH3u8j620KQJ5wLeEbhwzbWjBeTvjxsqviWHDHrXiK

```
lfy6PVKYMjLyjtSirtuOUMSnbXj6KNYN7sHXfEesw6mHCSI2ic0RaNZC3szG5mpo4NloNHEkqb5+t+45o69
ivy39ox8HA4/
yCWucmOtBXmjT7CIuumnT1eU93IbAedB20dzatnrmahequpUPdwyTsVi1fgFWP7Z5SlzY165+tel2bde7rQ
O2dLRne6yiDTJWPKLM+7zP2z6SVQ0xU54YwWFD4iKNiGKlUsWZwzyQXRdtGp14Gk0BR6QInZtFBsV1s9dDN
CqHx0I6DmNC8REFh1E3PDcCkUbmOOT3FwJAZUafS1QmBnmIBM6Z3ab3ObIR13JmqzktL50+vzk2y38fgcBn
/
9u7YlPRCnkKDFvpdhU0u4d52rfA+u1BhF9PDn3DCQ8hff6ZLziwW8M57U59LEuLaGIM0EWnEU2NB4SNuPJX
vbH8dT79wGGsvWwgAVo3O+zKsWr4g1G1QXUsJCe9obdXyBYHrCoI6mKgjO539opADqiFGei9q8Zkulk+QsT
hMWJhQajtbV9MwRipVrLt8Yf15qllQkMursvus9HmN6bbMDMJrBPYL0VWP7jbqvdS5SfaI2LB5f2B7nDGta
8qswS/
YbVyje7sLWHvZwlDXb5Ow0+XRBq+3mO0su5Aj7VoZm3Oj5DXJxlmuB4MsKDwMDpcTjRSUt0aQmyWAwBEpgM
lGNwtsNzHSGTL1ipoa/sVhX9i4K/
CMKEICqAmDoI6LoB9pJxES1y+dh4G+kjYSqimfuvx41RxBgRyDRqfe4647b26gF1FQ/
pWuOagTM3U4XkEQZ/
V52OY4Sl0aZRQeRLdH4JjchU3CLmrd+lFltOlY80R430ld2kFW2KBmxbl2da+wmfXpHD6SuI8HwcZsiWr0/
ofe213ADUvnWce6qVTHtQ1ppFLF6PEx3LNiSX307zXS2WolbDcx0ul2R0N6+Ep1HLc/
ubfeYUSRB3kinPlbMwJ/
u+68uYHGOFOxSz1FbWwbHb3dBdyzYkldFRjFqyRsxO8fSaq1Jl4DpEmnfufAIiw7Y1ZoPpSueXC4HHl06F8
lrGKE3bNiCYCauuqM257B/ADDaZTNcZLaumzLZTKSJ/
XwUeqnsOuowIQjEWbifp7afThSW1aeW6bYZrpuKUZoLiMsKCS6Rj9SqeLB7a/j/Sd1oZBP/
vSPjVax6rHd9U44zosWtuuX7nsUVD6jquEmhMBzX/
jYJOGaJ8INnjATquMq9RRDR6GH36pEzoMQk9d7BOnZdUQ1RCq8njhhtqYHP/
1R3Cs7bRNKCOUivPV5Iqz0CS5gqgBQ5bTxkvJiYzuwxdZOsmz9lvrMQQ2y1MwtTAPQUywYB3lK/
RQUTFGdZSuYwlrOSKUauS2bvAJN61KSCLQgWPUk0XodydofqVSRo9poNYp+P4jquMCtj+yO3Snp8DfipPHr
q+Mikk7dm4c7BxZNMe4rFZvS39s8xzjq55FKtW6kTKLDjoqaNehmSd7uSuXNpn6Cnn8hRwBNneH4BYC615/
/o9733sZLCpjqQJGkfekM4l4Gh8tTVH2rHttd/
z3I28nPWxa2ElUGmwCWNusYXOOtH79NSIfrtRQ8o5DYPNgJUevcSj1FK/
WBCVPnm5MtIMr8JchAef5ZsyNdI4hxIWqdkiVBPtxrBvdg5cYT6wRGKtXEwjYMpYZrtIeXKcyDisOjiLrnQ
55o0pqKDVcvrqtjgkbNqoNZM7gH7xw3d2xeL6kglJuvP1yNTf6D2s+4hTH49if3ThEE1XGB25/cG/
hbEKZ1Rl6UUA0L46/
UYI3mjZHKlFmhjZOMK1hQSKI82PJIBf8SsKuXK6Z35azUMgC0MfsHh8t4fGc5tjFYkSfCinPnosfSOPbQjo
NTVB5RYyK5IKzjS4uw+92ycRf67ni2PuO56pyStf1rQoh6JwZMNlCb1JG6tRtB+Q7q/As5wujxMcxf/
fQk2waAKbaDe1cswb0rlkxKC1LZTgBY+cgu4wIz3WDi2KjdQEN1mKuWLwgdMHn3UQlb+DbQV4psN0uKbh/
zINLYv4PDjHvQhbTOKqZQ2zbhkW0p5Akbrl6Mgb4S5q9+Ovz4XM07ZGS0ah0KJQ1UaBR/
KAxbTPG3dMffdeUi3P7k3tB2pBbrRcmbqu+gNRs6dZdpYZg/30HhMmxDp5uwaTNB17I5T0dvdwGX/
s4p9ZD9NU/EcLeMoPUcQc/
GZp2PK1QeVmq8D70kCb9vCjPOgsKDf5VjlvF2xl5X27Aol3HpLuQwrSufOHR4o7FZPR2E0sd/
4ZFdWjvJDZq9Q/y6dR1R7T8qX7rAdLpOLsweNmNaHn/5iakdvgpjYTo3rGMaHC5rFwX6yRPh7msW1/
Oh2+ekWMhhekhb1Nlw4mAS0GnR213AyGi1bs+zGXQVcoQNn1wcezbB+1FY4EpV0whI/
qM6CxVPKU1GqxOhbrVZpFIdN64ID8LbMZgaxNZ9RwINnup7WJ3EmWmZhJ1ajOkXXEOvHTX6779zfHySkTjK
iDks8KQ3FH4Y40JMisC67vKF+MLGXVPcs8cmBK5afAq+u/11ret2VAcGU1nVM7/9yb0NM2IP/
8WFUwRTaHtx7BI76dI8o6jhUlXDuEUtVPRuL5oWXlWDrSrSu8LYi25ErHBdFtPo/vpv/
AjbQuxqPcUC3hubiNQZmu658C/
+KdSIHoR3p0fdO9ljUImFoTwXvVvVhr37jVIzAbV28fJdl8Tqk9JSPbExW9Ks+EeMGQKw7vKFdSNrmkJCOQ
UAtU7e1l51bLSKWzbuqht71wzWRtHrLl9ofMHGhXA2CFTRUnUG2L1v/
Cb0GiOVamSbjM4JxMbTSodALSSOqZMcqVRjq5W6p3Xh1fWX4u5rFtdnYGEOBY0cTo8LEXvgmlY/
xqonSdI1B0w6CNRUOcvWb0l12u9KD63Chj/y/EEct+jIXHVA6jr+nQ/TslkBwd41YeE/
GoUprlJ5pIIltz87aUbSLIcLHXGfn+vQHQoWFBLboHxMY1FuiGnP+NT1XYVgtxESOnIEFPI5vDcWzyaktuy
NqkaKitom9/yzZtfVRDYqmhnT8rFnGwrTfZQjgsn7LGtOGa7UkK5DdyhY9eQlRWMQEx0V/
RVIfz2E2tui2SNhoLawM66QUERVI8WlPFKZpCay6epGj48nDocjgCnXKBbyuGHpPAC1qL1CTD0mq7ia0bgO
3aHgGYVkw+b9TlzpGHcI1EatJhdVV2RN9dDOCPlP0nA41XGB7kKu7o1HENj4/
MH6ezxSqW045T2m3UlrQMUzCsAYyK2RlHqK6I0Q6bRTYG1gNIqFfObbUXVCoHtaV+IVzl4BMFqdmBryY0Lg
vbHOaEBpbYMKsKCI7OudFkqvygPbaLSGYqFx9HYXcNeVi7D2soWR4kg1gzdGKs5jEgXRyrPFnmJBW4/+zts
bmt41HS8omrF/
dBDKWyVrRras07pdQDqozYBU8DrbGF3NYE5PEQN9JUzv6vhuKBACsHDO+7XPJ0iZFrQvjQs6voaytH7CtcA
qFrJdvTwbcE95pFIPaAfU1nJksR2oNRhrBvckNtw3GtMo3yUCwLaXj0YePKahRu94Y3Y7r5/
Iom7fa1gsFnJ4d2wik/
lsZQQm76udRSeNd+WmTK0IEXDVOaVI25o2GhWd2BXZG2o0mKh7ArQSWRyp+Y2PLCTSozohMikkgNZWGR4br
eLxneVMOwys27TX6fU6fkZhG8CNmUqxkEeOkHjxFMO0GjU1cXbFnWtbZ8fPKICasMiiHrcZRLEb3HXlIiyZ
```

```
OzO1vDBMlrHZ36JdCO0dieh+InqTiH7qSVtHRGUi2iX/LvH8dhsRHSCi/
US03JN+kUw7QESrPemnE9EOmb6RiKbJ9Ony+wH5+3xXhQ4sZ5oXbyGijJFWbtwVGpG0k+ChBtOu2LTtbwG4
KCD9HiHEEvn3DAAQ0dkArgWwUJ7zN0SUJ6I8gK8BuBjA2QCuk8cCwF/Ja/
02gGMAbpLpNwE4JtPvkcelRqes3HRJIyferSDIuQUx7UqooBBC/
DMA22HjFQAeFkK8J4R4BcABAOfKvwNCiJ8LIY4DeBjAFUREAC4A8Jg8/wEAA55rPSA/Pwbg4/
J4psMo9RRxz4ol1ntLMwzjliSz5c8R0QtSNdUr00oAvDu5H5JpuvQPAhgRQoz50iddS/
7+ljx+CkR0MxENEdHQkSNHYhVmxrT29HxqB8ojFQz0lXDdeXObnRWGaQlcL7SMKyi+DuAMAEsAHAZwt7Mcx
UAIcZ8Qol8I0T97dvRYJ4PDZVTYcyfTDA6XsXVfvEEAw3QSOdQWWroklnusEOKX6jMRfQPAU/
JrGYB32HeqTIMm/dcAeoioS84avMerax0ioi4AM+XxztmweT/
rlzNOoza1Z5hWJ59CaPVYMwoiOsXz9RMAlEfUJgDXSo+l0wGcCeDHAJ4HcKb0cJqGmsF7k6ht2L0VwNXy/
BsBfM9zrRvl56sBbBEpbfDdriuz2wkWEgxjR3VcOI/
3FDqjIKKHAHwMwIeI6BCAtQA+RkRLUHN8eRXAnwKAEGIvET0C4EUAYwA+K4QYl9f5HIDNAPIA7hdCqKWDXw
TwMBHdCWAYwDdl+jcB/AMRHUDNmH5t4tIGkEakRYZhmGbiOoYdpTRIb
xr9/f1iaGjI+vi4m5gzDMNkFbUHfBSIaKcQoj/
ot45fI5Sl6LEMwzBJKeTJ+T4fHS8o0t6LmWEYppF05chp5FiABUVDdthiGIZpFJXqhHPba8cLioG+Eqal4E
7GMAzTLFx7PXW8oBgcLuN4RmP2MwzDxMG17bXjBUUa+8syDMM0E9e2144XFOz1xDBMu3H+WdFDGZnoeEHBX
k8Mw7Qbj+8sOzVod7ygYK8nhmHajUp13KlaveMFxUBfiUOMMwzTdrhUq3e8oACAT3zE7eIUhmGYZuNSrd7x
gmJwuIzHd3JgQIZh2gfXYTw6XlBs2LyfQ1gzDNNWuA7j0fGCgt1jGYZpNypVt1uxdbygYPdYhmEYMx0vKNg
9lmEYxkzHC4qBvhJ6uwvNzgbDMIwzeopu+7SOFxQAcOnvnBJ+EMMwTIvwh4vd9mksKAA8/
cLhZmeBYRjGGVv3HXF6vY4XFIPDZRwbrTY7GwzDMM7gMOOOuf3Jvc3OAsMwjFNmso3CLTybYBim3SDHm3Z2
vKBgGIZpN0YcD4A7XlC4diNjGIZpNrzDnWPWXb6w2VlgGIZxiuuFxB0vKFwGzmIYhmk2Nyyd57xf63hBAQA
ljvfEMEybcOfAIufXZEEBjvfEMEz74HKvbAULCtTUTye/f1qzs8EwDJOYVY/tdi4sWFCgJoF/
+Zvjzc4GwzBMYqrjAhs273d6zVBBQUT3E9GbRPRTT9osInqOiF6S//
fKdCKirxLRASJ6gYg+4jnnRnn8S0R0oyf9HCLaI8/5KlFtqYjuHq4ZHC7jtif2pHFphmGYptCMEB7fAnCRL
201gB8IIc4E8AP5HQAuBnCm/LsZwNeBWqcPYC2A8wCcC2Ctp+P/
OoBPe867KOQeTuGtUBmGaTcavo5CCPHPAI76kq8A8ID8/ACAAU/
6t0WN7QB6iOgUAMsBPCeEOCqEOAbgOQAXyd8+IITYLoQQAL7tu1bQPZzCW6EyDNNOFPKUmXUUJwshVGzuXw
A4WX4uATjoOe6QTDOlHwpIN91jCkR0MxENEdHQkSPRwuvyVqgMw7QTG65enL11FHImIBzkJfY9hBD3CSH6h
RD9s2fPjnTtVcsXoFjIJ80iwzBM00ljsR0QX1D8UqqNIP9/
U6aXAcz1HHeqTA4RDzQAAB2ASURBVDOlnxqQbrqHUwb6SrjrSvcLVBiGYRrJmb81I5XFdkB8QbEJgPJcuhH
A9zzpn5LeT0sBvCXVR5sBXEhEvdKIfSGAzfK3t4loqfR2+pTvWkH3cM5AXwk3LJ2X1uUZhmFS56U330HfHc
82Z8EdET0E4EcAFhDRISK6CcB6AH9ARC8B+H35HQCeAfBzAAcAfAPAZwBACHEUwJcAPC//
7pBpkMf8vTznZQDfl+m6e6TCnQOLkHccw51hGKaRHBut4rYn9jgXFlRT/7cP/
f39YmhoKNa5fXc8yxsZMQzT8pR6iti2+oJI5xDRTiFEf9BvvDLbAwsJJkvwBJeJS5n3zE4P19sHMkwSfveM
Wchzo8wMhQ7WTbOg8NBmWjimxfnJ629hnBtlJiAg5UUA2YYFhSQNTwGGSQKHlskOAkB1onMlRVezM5AF1gz
uwYPbX292NhiGYZwwY5rbRcQdP6MYHC7jwe2vd/KskklAsZBru5X9nauJbx/+8hNuF951vKDYsHk/
CwkmNledcyqmd7XHa1TqKaLUU+T3IQZZawOZi/XU6nD0WCYJG58/
iJFKe7hVr1q+gN+HmLw3NtHsLKRKxwsKjh7LJKE63h7jb0JtFNrTXWh2VpiEpOFR3fGCgqPHmmE//
s7gd8+YBYBdxNuB689zH7eu4wWFih7L/eFUCnnCdefNRSHHD6fd2fbyUcxf/
XTbqNE6hRwA9XrmiXDD0nmpRJDteEGh4JHUVMbHBZ5+4XBH+48zrU+pp+jcXTQrTAD4wEkF9BQLGBcC39n+
eioRZHkdBWqeT8xUJsDxr5jGQ3C/CPqd4+27eNE/
Czw2WsWqx3YDcOf9xDMKZMvzqbuQQ3eBq4XpPHqKBdy7YglmFt0a1F0HyGsFquPC6QCYZxSoeT5loTEVCzm
8+KWL69/P/h/
fx2i1vd3u2p1SRtqWnxwBWdMovv1ubSTcLp5kSegpFhLbi1wOgHnoCuD8s6Lts50WlerEJN1iJWUhkWcjdZ
00DPZx9gRoFFnsiidE+u7GjXRaKSbQDLhwKnDp+s+CAsDWfUcC04PCM6TdzrzTxbTXeOQAVnMB6O0u4H0nu
Z1cF/KEVcsX1K/
fDEw126nOG40qNwGYNWN6Y24WgLf9uYB7CeinaO9WJ3DXlYtQ6imCgIaENyiPVLBmcA+Wrd8SW2VBqE1de7
sL9XwHdVbVCZH6rKUV6J7W5dxov+Lfz60bEtdetrApexmYapbXx6TLnJ5i02yfvd0FbLh6sdMwHmyjgN5GM
aeniIG+0qQHnqQDt+U7CSLZ6tQdp69+OvB4l4KPKHjEloYXi0veGKkgT+R074eHdhxE/
2mzJrWfDZv3442RCub0FHH+WbPx+M5yU0KJFwt5XHVOacr9CzkCKFurzQs5QiFPLWWrKxbyWLV8ATZs3m/
VV3QXcpheyGNktIo5PUWMHh+LPXBJS93JMwoEr85WlR10bFYXoOnyPDhcRq4BI8gsqTOilHZOT9H5BkHjQh
g3uX/6hcNN22/
irisXof+0WZMC2fV2F7Dhk4ux4erFk2bQy+SK7WZQ6iliwycX48UvXYx7Vyyp5yvL5Kj2fAf6SqFRHwjADU
vn4cUvXYzhv7gQr6y/FNtWX4C1ly2MXc607K0ksvR2O6C/
v18MDQ1FPm9wuFwf8c0sFkAEjIxW65+PjVadjIxdeDME0dtdwNrLFk6Zbg4Ol3HbE3uauglOKeEoKSpqxPz
QjoOhAqBYyOOuKxdpR39J67zUU8Sq5Qti14Fr7ySVn1WP7p6ykNLbhvzvw/
Gx8dij+jjPkAC8sv7SKemDw2XcsnFXrHxEoZADxkXt2ROAXI4wblERPcUCdq29sP5d16/
MkfWgUw+tGdwTS7OQZEZBRDuFEP2Bv7GgmEzaHesNS+elsv9FDsDM7sKURtgIVRlg7gyWnTELP37lWGorvK
flCcelusTf2fnrspAjvO+krknPCQDWbdo7RYA3U2WmOvSVG3c5zcOMaXkIIbSdvk4tRQCul+EhVOdn0650z
```

9Dm2ZZ8nWmUdzOJKvHeFUsmqQmjDHIIwD3y/
LJHpekviw1qQ7UopdAJWKtzWVDY06iOtRH0dhcysbLa9qXNATipkEukj1YzBG/n4n3p/S9rFmZcCn/
egea0R119qU7Qm7+
+O57VtrGwereZXXufie2z8HaWpvzp8vTOe2OxBzU9xQLeG5sIbE9B9asjijD2ktaMgo3ZkrgV45o8ET48ux
svvflO4mu5UpclxXZkNwGgd8Z0vPfWu7FHg5XqONZt2lt/
Gf3OCH42bN4fWUi4Mvp2F3LonTFdK8QAJFJbxUX37AVqz8ubxxFDJ3z3NYu175Tq0GqC+gWt912lOl6/
p60XUbcnrtPayxZaP79iIY/jY+OxhUSxkAeRfq/
zSnUctz5yIrRG0CAGCJ7d2t7fpUusFzZm48SostlColjI4+5rFuO5L3wMNyyd58SFsdlCIipvjFRw3XlzE1
1jpFK1DooW1YVRGViV0TcJ07ry2Lb6groRM0igqejGjXRnNd2rPFLBsvVb6s9Xt9anp1jQGnSndmjmsqk6s
l1XNOqJ6xT2/
PJEdcP9XVcuijyb9Z9vEpzACSeHNYN76n2OQO25rnp0N1Y9tju2DdN2thIHFhSIN6oMI+qDVQ1NVfSdA4vw
8l2X4Ial7mPLZwFd1zCnp4g7BxYlFpS2cW6iLmpUo/
6BvhK2rb4gkbAYqVSxbP0WnL766Umdr5+BvhImGqQiLhbyuO68ucauuzxSqXt0BQkCwomyAaivRQJqHauaJ
agRddi7p+rIdu+YoDrVzZImhDAK6jD859u0p0p1HA/
tODil3NUJEXuWmifCyo27jO0oCSwokE5QwCjjkhuWztM21Id2HHSXKQmheXv8qtHX9UvnTXEzLuROrCZVgv
LV9ZfGEpa2dRp14yolgAaHy4ntBwRMGlGa3GnDOqDuQi5wUd80i4V+SiCrwcqdA4tw/
dJ5RmHhVQl5BYFX1anKBJxwK1cdthpBhz0/7+zDey+CPkSG10VUaQt0+J9r1FX0/
vNtVT9J3LGD6mVcCKt2FBcWFGj+dqhP7T6sHVm69u8Hai+yaY/
ftGYxpZ5iffTVf9qsqS1e0zPdObAo8n4CtnXq7+jCeGOkYqWqDJsMBdmOVOfrZ3C4jHfeGzNeb7Q6AQhMWo
1/74ol+Ne/
vAT3rlhizIcSyN7Byp0Di3CPXLugQwlj7+wqqEzrNu3Fuk17p+j+qxPC+Jz8s2zvvV5Zf6k2RIY3JI9pxhK
k04+yij7o/IG+kpWwiTtbLuQJ1y+dVxeWQdfRtaMksKBAbRTgUgNcLOQjjUxGKlXtyNLUoG5YOs/
5Nq4lqfpxHZ/I/1Jt2Lx/yjTbFBp5NMJ+AgTzyE7NBpRgBoBtqy/
Aq+svDS33nJ5iqLqkWMgbFx+aQsH4bQBKKPn11kFrPqsTAt3TuqaoUgb6StoO3yRQw9Rr/
nN1s7iRSlWrdxcCgTaMe1csCZxle+tOJ6i9+TDNLK86p4QNm/dPGqAN9JWs7E8kzw/
SAqy9bKHxvTqvTjMmNaFOwcW1YWlTiXpWkvCggK1FyLpuF06wtRHQWGNxYR3RKBrUGrLwyijYS89xYLWyD
g4XMa/vWsewUaht7swZWSoa8i69J4IgktAv2GLdzYQJJhN9aZUY6aXUNW/rk6UEDPVmTdPOqGkc8xRgsY/
O40SfcCP7blxZ+aV6vgU9ZffhXnZ+i2Yv/
pprNy4q153OgRQL7vJ2P74znKo6k83TBPQBxP1q8j8cdeUei/
OYOwtn8DVlc+1liSReywRvQrgNwDGAYwJIfqJaBaAjQDmA3gVwDVCiGNERAC+AuASAKMA/
lgI8RN5nRsBrJGXvVMI8YBMPwfAtwAUATwD4PMipYUfSfcNECJ4oYvX/W3+B4v4l5ePWgkl1Rmp/W/
VKuM81faxVule90/bVavFQh7rLl84JX/
eRXouFsfpVosD5vhaQUSpdVMnHNTxKvXIpBW0CFiURua8e10+daoi5WIa5vaqBgtRR4bK7gFMthH4402pVc
IrN+6q50cnXINiVQWtRwlTj5kYF6IufEzrXGybgSp70OJBnRuragfedRCm+5nqJswtGwh23S0W8jipkNOu/
QiyiQRdw7WbbKIFd1JQ9AshfuVJ+2sAR4UQ64loNYBeIcQXiegSAP8VNUFxHoCvCCHOk4JlCEA/
avWyE8A5Urj8GMB/A7ADNUHxVSHE9015ShLCI4m/uu1CF/
+Sft2UPO7CGZ2BNU+ECSFCQwcAtQCCulZRklExveEIcpqFVd4y+H3Gg4LimRYkmfLkJWxRk+11dOtPdCE51
H0BhLYjtSDMZu2ObtFa0MIuU569bSmorUdZDObH5aJFf16TOgyo5+dfIe1ixbuLAHxBaylMebvXt+BRd404
9djoBXdXAPiY/PwAgB8C+KJM/7acEWwnoh4iOkUe+5wQ4qjM7HMALiKiHwL4gBBiu0z/
NoABAEZBEZegUdP8Dxax7eWjoedGkeDekcay9VsCBUWYjt2EqROzbTxhI2YvplmMGnH5O5LySAWP7yzjqnN
K2LrviFUDt9mFME8UWk7b3Qx1L6oq0/
SuXL083pnTsvVbrN09VVswdYZBQkI3I7TR2atzgkbT3sVgUXDpXu7Pq82MyqQNUM/
PP2NJurjW+84n6aiDZh66vKm1KTbXcE1SQSEAPEtEAsDfCSHuA3CyEOKw/
P0XAE6Wn0sAvL6eh2SaKf1QQHpqBIUU1xFlhO4nbCSpdOxxGqCNmiAM2+msreuhrmPauu+I9YgsTFVjIwyT
qkcAYGaxMCUf73pUVGEdW9BztFl9rWtvNuE+bI3OajGY/
7phuDSc+vMaJtiLhTzOP2u2VUwkr0vv+WfNDgy6N2NaHu8EOE70FAuYMb1ritpu3aa9eOf4WN0xI0jdFwVd
G/
WuTYk7Y0hCUkHxe0KIMhH9FoDniGif90chhJBCJFWI6GYANwPAvHnuXDtNL8Dd18TbGMQm0Feppxg4Crdtg
ElHGLbCxtb1UPccyyMVnL766VBh5jXqKhVCT4RInOoaQZ2xMijaxgOqjk+N4+PtgEwdmy4wnPd5685VC7tM
2Ap4Ux69ZbElzp7zQWFQbIWoUrGp57lh8/5ItgtAb4gu5HMoFjDlGa67PDjQZJBGQM3OVm7cpbXnRA3dEbQ
2pZHCIpGgEEKU5f9vEtE/
AjgXwC+J6BQhxGGpWnpTHl4G4HXhOVWmlXFCVaXSfyjTTw04Pigf9wG4D6jZKJKUyYvuBdBNAcMYHC6HCgn
vpiemDskVulmL+lO/Bxk9TYLUO7o3dSRejxNgauP3v5hKhaBeXFt0Qq17WpexM/ITNNoETjyLuKq/
MDWUjReLrYAPm8FEnSHYxqPyz4ps8hpkhFchz8sjFdz+5N5IQf8ItTalK+NblWo9+mtQvmzVbN6Fhbc9sQd
Drx3F1n1HUB6pTFmUuOrR3ZFih6XRD4QRW1AQ0QwAOSHEb+TnCwHcAWATgBsBrJf/
f0+esgnA54joYdSM2W9JYbIZwP8kol553IUAbhNCHCWit4loKWrG7E8B+N9x8xsH3Uuv9MNRCRv5eEecK0P
0/i4Im7WE/
W6yZXgbsU1Homv8SQSmVwiabA5BHWzUEXKOqO6H77+W6hSXrd8Sqg5M6sViM5tUv9/6yO5AG0hU18qBvhKG
Xjtq3P+jkKfA7TltOjvvoMW/j0bU6MjK6yzKrpZe4rx/ler4pAGi/

```
wnF8TBs9DarSWYUJwP4x5rXK7oAfFcI8U9E9DyAR4joJgCvAbhGHv8Mah5PB1Bzj/
0TAJAC4UsAnpfH3aEM2wA+gxPusd9HSoZsHXH1/bpRepj/
vVdfH9V9NA5hnXDY77admv85hhmKw9KAmtDydsx+bD1x/
IZlhW5krwsj7dfvm1w8TTMoFzYmG9T1XLhWDg6X8fjOsjGKwIxpXYnLsGHzfm2n6p8BFgt544zpnhVLYpU9
ziACsHfrtaXR0SRiCwohxM8BLA5I/zWAjwekCwCf1VzrfgD3B6QPAfh3cfMYFdNuVCoOv1r8Y7u/
gbdT0DUyv5eTzqBVyBFGj49Z6fVtCFv0FvZ7lE7N7+1lKwRNL6ZJV2ujIjB1DDpj58I578feN34TeG1Xs6J
GeLGo+6j8JRFKNs/av1AsDqaBlrJZeMuhs/
nM8cx4bcru7xcKeWrYvuK2tpy04f0oJCYjlVfP6PX91y1o0nUKOl349Uvn1c/
XjYS7CzlUJ0R9qu3CqKXrhGcWC1i2fot2FOTt0ON0alHUKya1lamzDZua54m0IRgAvbEzbMFk1FlRs7xYXG
KjBtGNgKN49oU5CgR50JnamU3bDeoXCjmKtClY2J4wuv1NlNs1kP4MMwwWFJKwUZEKDeyfXvs7K9Mo3GYUo
8vHe2NCe++wa+oI6oQLOcI7x8e03hcuRjNRZyIAQtdq+AlTEYwLgcd3ltF/
2qxIgiZsHBl1VmQj8F0tqAq6blzPOi82LqxBbSbq/
XV7fRfyFHj9JDMmkwu7iqnVPa1Lq55UrrS6xaV+z62wfDZ7IMFboUpsV+wG4d16UadWsV3FGScffn1slAV2
awb3TAoPclIhp/XsibPvryuiPldbG4Xu/
DgrgnXP3SYvUcqRZBW1l6Rt1ZRHf0cYlNc49x8cLk9yIzWFiYmLTX2pbWFt6yYtYe8S3grVgrhGKnWuIqnX
ii4fulAOaiMYL1G8grxGyHEhtEKCgMThCpIQ9bkmMaDr7mdSIZg6RJt1Erp8pOkmnXR9iyLuyD1qYEh1ryT
ltumwbTdTimujc00jhBALComtL7gfwuSNUpIaCHUdoi64WRJf+CihF5q9Z0ec5xrXgK673/
lnzcbGHx+cpPoo5AgbPqlffOl/
iXW6bV0+4nSmtiRZ3+InTkfYCM8+L7aqriir6xvleKDDlfowDBYUkqCFPUThftoCmKLrTtJ4TB1i/
2mzpqSbPDvCsO1smuFlEUSS5xpnphfk5rrxed+Og4aNTIJe4kKOpnjNmPKRZmeaZH2LCxoV+VRhOzuLs7q+
WTRqYS4LCg9BHZGNrtp1xeg6RF163JfNtPLca4zL0osRFxeuoKbNlmxddKsTItLzTbMzTaqec33/
tNuaSdXmxUVgzUaR5ozTCwuKEGxVUo1eKalI8rKZVp5n7YVwQVI1QdSX0hQmYtfaC63umXZnmkQ95/
r+aWNax+RdvNloAZaERqnvWFCE4G80ur0XmqnDj/uytdILkQV0L6U/fEfY8XFCZDSiThqtCmo0q5YH7/
Wgwnr43VFb4T1oVJ2xoLDA22h07opZf5nCgv+1Cs10M9TNLnXhuVut440ycGgFd08/
A30l7XqcJHtTxMXFM2zUYI8FRURacRTeKM+ItGl2OdQ9goLpBdmpWrGt2Awcml0PSTC5mTcSl8+wEYM9FhQ
xaLVReKM8I9ImC+UY6IsW2bfV2ooNWaiHuOgCF5oCGqZBqz1DFhQdQKM8I9LGVTmSTvkb7f+fFNdqolZuT7
ptU0sNrrtWe4a5ZmeASR9dB5bVjk2Hi3KoKX9ZuoOqKf/
gcOCeWIGsWr4AxUJ+UlpWbQ8uyuunldtTVuqu1Z4hC4oOICsvR1JclMM05bdloK+Eu65chFJPEYTaaDSLPv
aAm/L6aeX2lJW6a7VnyKqnDqAVjapBuCiHqyl/
q9ge0lBxtHp7ykLdtdozZEHRIWTh5XBB0nK0mn0hKWmVt13aUzNppWfIqiemo2i1KX9SOq28TDrwjILpKJo
55W/
GIrVWU3Ew2YQ3LmKYBpDmBkQM4wLeuKiDacVQC+1Iqy2wYhgvLCjamFYOtdBuuPY+4gEA00jYmN3GpOFDz8
TD5QKrNBbRMYwJFhRtTKuFCWhnXHof8QCAaTQsKNqYVgsT0M64XBHMAwCm0bCNoo1ptf0Q2h1XC6w6bdEg0
3x4RtHGZCWuDeMWXkTHNBqeUbQ5rRQmgLGDF9ExjYYFBcO0IDwAcAu7G5thQcEwTEfD643CYRsFwzAdDbsb
h5P5GQURXQTgKwDyAP5eCLHe9T2u/8aPsO3lo64vyzBMC1MeqWD+6qebnY1YvLr+UqfXy/
SMgojyAL4G4GIAZwO4jojOdnkPFhIMw7QbrgVcpgUFgHMBHBBC/
FwIcRzAwwCucHkDFhIMwzBmsi4oSgAOer4fkmmTIKKbiWiIiIaOHDnSsMwxDMN0AlkXFFYIIe4TQvQLIfpn
z57d7OwwDMO0FVkXFGUAcz3fT5Vpzlh2xiyXl2MYhmk7si4ongdwJhGdTkTTAFwLYJPLGzz46Y+ysGAYpq1
w7fWUafdYIcQYEX0OwGbU3GPvF0LsdX2fBz/
9UdeXZBiGaRsyLSgAQAjxDIBnmp0PhmGYTiXrqieGYRimybCgYBiGYYywoGAYhmGMsKBgGIZhjJAQotl5cA
oRHQHwWszTPwTgVw6z0wpwmTsDLnNnkKTMpwkhAlcst52gSAIRDQkh+pudj0bCZe4MuMydQVplZtUTwzAMY
4QFBcMwDGOEBcVk7mt2BpoAl7kz4DJ3BqmUmW0UDMMwjBGeUTAMwzBGWFAwDMMwRlhQSIjoIiLaT0QHiGh1
s/
MTFyKaS0RbiehFItpLRJ+X6bOI6Dkiekn+3yvTiYi+Ksv9AhF9xHOtG+XxLxHRjc0qky1ElCeiYSJ6Sn4/
nYh2yLJtlKHqQUTT5fcD8vf5nmvcJtP3E9Hy5pTEDiLqIaLHiGgfEf2MiD7a7vVMRCtlu/
4pET1ERCe1Wz0T0f1E9CYR/dST5qxeiegcItojz/
kqEVFopoQQHf+HWgjzlwF8GMA0ALsBnN3sfMUsyykAPiI/vx/AvwI4G8BfA1gt01cD+Cv5+RIA
3wdAAJYC2CHTZwH4ufy/V37ubXb5Qsr+BQDfBfCU/P4IgGvl578F8F/
k588A+Fv5+VoAG+Xns2XdTwdwumwT+WaXy1DeBwD8Z/l5GoCedq5n1LZBfgVA0VO/
f9xu9QzgPwD4CICfetKc1SuAH8tjSZ57cWiemv1QsvAH4KMANnu+3wbgtmbny1HZvgfgDwDsB3CKTDsFwH7
5+e8AXOc5fr/8/ToAf+dJn3Rc1v5Q2/3wBwAuAPCUfAl+BaDLX8eo7W/yUfm5Sx5H/
nr3Hpe1PwAzZadJvvS2rWcpKA7Kzq9L1vPydqxnAPN9gsJJvcrf9nnSJx2n+2PVUw3VABWHZFpLI6fafQB2
ADhZCHFY/vQLACfLz7qyt9ozuRfAfwcwIb9/
EMCIEGJMfvfmv142+ftb8vhWKvPpAI4A+L9S3fb3RDQDbVzPQogygP8F4HUAh1Grt51o73pWuKrXkvzsTzf
CgqJNIaL3AXgcwC1CiLe9v4naUKJt/
KKJ6A8BvCmE2NnsvDSQLtTUE18XQvQBeAc1lUSdNqznXgBXoCYk5wCYAeCipmaqCTSjXllQ1CgDmOv5fqpM
a0mIqICakHhQCPGETP4lEZ0ifz8FwJsyXVf2VnomywBcTkSvAngYNfXTVwD0EJHaxdGb/
3rZ5O8zAfwarVXmQwAOCSF2yO+PoSY42rmefx/AK0KII0IKoAnUKv7dq5nhat6LcvP/nQjLChqPA/
gTOk9MQ01w9emJucpFtKD4ZsAfiaE+LLnp00AlOfDjajZLlT6p6T3xFIAb8kp7mYAFxJRrxzJXSjTMocQ4j
```

YhxKlCiPmo1d0WIcT1ALYCuFoe5i+zehZXy+OFTL9WesucDuBM1Ax/
mUMI8QsAB4logUz6OIAX0cb1jJrKaSkRdct2rsrctvXswUm9yt/
eJqKl8hl+ynMtPc022mTlDzXvgX9FzQPiz5udnwTl+D3UpqUvANgl/y5BTTf7AwAvAfh/
AGbJ4wnA12S59wDo91zrPwE4IP/+pNllsyz/x3DC6+nDqHUABwA8CmC6TD9Jfj8gf/+w5/w/
l89iPyy8QZpc1iUAhmRdD6Lm3dLW9QzgdgD7APwUwD+g5rnUVvUM4CHUbDBV1GaON7msVwD98vm9DOD/
wOcQEfTHITwYhmEYI6x6YhiGYYywoGAYhmGMsKBgGIZhjLCgYBiGYYywoGAYhmGMsKBgGIZhjLCgYBiGYYz
8f/GBltohlT2oAAAAAElFTkSuQmCC\n"
          },
          "metadata": {
            "needs_background": "light"
          }
        }
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "sns.stripplot(y=df['Balance'])"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 269
        },
        "id": "7JZmnrI1Upkl",
        "outputId": "31092dff-8f48-4b5b-c629-2262bd6fdc21"
      },
      "execution_count": "null",
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "<matplotlib.axes._subplots.AxesSubplot at 0x7fed290c8790>"
            ]
          },
          "metadata": {},
          "execution_count": 15
        },
        {
          "output_type": "display_data",
          "data": {
            "text/plain": [
              "<Figure size 432x288 with 1 Axes>"
            ],
            "image/png":
"iVBORw0KGgoAAAANSUhEUgAAAZcAAADrCAYAAACl8dsDAAAABHNCSVQICAgIfAhkiAAAAAlwSFlzAAALEg
AACxIB0t1+/
AAAADh0RVh0U29mdHdhcmUAbWF0cGxvdGxpYiB2ZXJzaW9uMy4yLjIsIGh0dHA6Ly9tYXRwbG90bGliLm9y
Zy+WH4yJAAAgAElEQVR4nO3dd3jV5fn48fd9TvZOyIKEkABhb0JAwa2Iq7hL/
bmp2qqttnbZZbWt0mWrbV39OlrurV1iwwVVBACyF5hJkBIIIGE7PH8/
jifhLNzAgknCffrunLLlnOez7nCRc+fZYoxBKaWU6ky2Y2YAeglFKq99HkopRStpjNpclFKKdXpNLkopZTqdJpc
lFJKdTpNLkoppTpdSLAD6C6Sk5NN2sMnQqkeZeXKlQe3cm0Z2RQUFAQ7DKWU6lkUtMSnu5ZpcLNnZ2RQUFAQ7DKWU6lFEZLe3cm0
WU0op1ek0uSillOp0mlyUUkp1Ok0uSimlOp0mF6WUUp2uy5KLiPQXkY9FZKOIbBCBCRe6zyX4nIXhH5yvq62O
ma+0WkUES2iMiFTuUzrLJCEfmJU3mOi8Hxplb8mImFWebj1vtA6nt1VP6dSvdH2sqN8f+5X3Pz8ct5Zsy/
Y4ageqCuHIjcB9xljVolILLBSROZbx/
5sjPmj88kiMgyYCYwE+gGLRGSOZbx/
5sjPmj88kiMgKYBYwE+gELRGSIdfjvwAVAMbBCRN42xmwEfmfd61URdEfmfd61UReQqYDTxpfa8wxgwkVnWeV/
vwp9VqV6jtqGZrz+9jINH6wH4ZEsZ4SE2po9MD3JkqifpspspqLMWa/
MWaV9boK2ARkeLlkBvCqMabBGLMTKAAmWVuBMWaHMaYRa9CTyakATyra9CY8wOY0wD8CowU0BBd43br+HTjO0e9p0ONfHTjPOl8pYp1Y5

lOw61JZZW767dH6RoVE91UvpcrGap8cCXVtHdIrJWRJ4TkUSrLAMocrqs2CrzVd4HOGyMaXIrd7mXdfyIdb
57XLeLSIGIFJSVlZ3Qz6hUb5GRGBlQmVL+dHlyEZEY4A3gXmNMJY5mq0HAOGA/
8KeujsEXY8wzxpg8Y0xeSorH6gVKnZKGpMXyzWk5tNb1h6U73ivVEV26/
IuIhOJILC8bY94EMMYccDr+D+Bd6+1eoL/T5ZlWGT7KDwEJIhJi1U6cz2+9V7GIhADx1vlKqQD8/
NIR3Dw1m4rqRkZlxKGtyqqjunK0mADPApuMMY86lfd1Ou0KYL31+m1gljXSKwfIBZYDK4Bca2RYGI5O/
7eNMQb4GLjauv4m4C2ne91kvb4aWGSdr5QKUGZiFKMz4zWxqOPSlTWXqcANwDoR+coq+ynwDREZBxhgF3AH
gDFmg4jMBTbiGGl2lzGmGUBE7gbmAXbgOWPMBut+PwZeFZHfAKtxJDOs7/8UkUKgHEdCUkopdZKI/
kHvkJeXZ3RVZKWU6hgRWWmMyXMv1xn6SimlOp3u56LUSXbwaD0PvbORlbsrGJ+VwAOXjSQlNjzYYSnVqTS5
KHWS/
ej1tSzaXArA3sO1VNc38fwt+UGOSqnOpc1iSp1kS7a5TthdvO1gkCJRqutoclHqJBveN87tfWyQIlGq62hy
Ueoke+TK0QxMjgZgYHI0c64cE+SIlOp82uei1Ek2sl88C+87i/
LqBpKiw3SSouqVNLkoFQQiQp8YHSGmei9tFlNKKdXpNLkopZTqdJpclFJKdTpNLkoppTqdJhellFKdTpOLU
p2spcWwuaSSI7WNwQ5FqaDRochKdaJdB6u5+fnl7DpUQ0SojYe+NoprJ/Vv/
8Ig+3D9fj7depBh6bHMyu9PeIg92CGpHk6Ti1Kd6A8fbWHXoRoA6hpb+NU7G7h4TF9iwrvvr9rzn+/
kwXc2tr1fsaucv103IYgRqd5Am8WU6kR7rMTSqqahmbKq+iBFE5h/f7nH5f376/
ZzpEab9NSJ0eSiVCeaMSrd5f2w9FhyrHXEuqvYCNdaVViIjdAQXZJGnZjuW1dXqgf61lmDsIkwf2MJA1Ni+
P4FQ7rsWSt2lVNZ28jUwclEhB5/H8m95w/hmy8V0NDUAsB3zs0lKkw/
GtSJEWNMsGPoFvLy8kxBQUGww1CqXcYYbntpJQs2HQAgMzGSN799OqlxEcd9z5IjdSzbcYih6bEeWwIo5Y+
IrDTG5LmXa7OYUj3MlzvL2xILQHFFLS8t3R3w9f9ctpupcxZx+iMLeeHznQCkx0dw+fgMTSyq02jdV6kTcL
S+ic+2lZEWF8H4rESXY03NLbyyfA8rd1eQl53EN/KzsNtOvC/
jsJfO9rfX7KWsqp7bzxrIoJQYn9eu3F3OL/
63vu39r97ZyPC+cUwe2Mfj3P8UFPHrdzdSVdfEGbnJPHXDRG0uUwHT/
ylKHaftZUe59qmlHKpuAODavEx+f/XYtuO/
fncjL1o1iv99tY+dB6v5xaUjTvi5Zw1JoW98BPuP1LWV7SmvZU95EQs2HeDTH53jMvT5ow0lPDp/
K1V1TQxM8Rxc8OXOco/ksreihh+9vpbWRvPF2w7ywFsb+MM1Yz2uV8obTS5KdYAxhi93llPT0MR7a/
e3JRaAuQXF3H7mIAanOmoOrxUUuVz72vI9zJ6WQ2psOCH242+Rjgyz8+adp/
PCF7tYsPEA28uq244dqm5gydYyBqfG8PD7mygsq6a4vKYtSew9XOtxvzGZ8R5lrxUU494b+1nhweOOWZ16N
LkoFaCWFsPNL6xg8dYyAKLCPEdoOS/5khAZRknjsdpFXVMLp89ZRGpsOI/
NGs9pgzybogLVNz6S+y8aTmSonb8s2OZyLDk2nFteWEFxhWciARiVEceugzUYY5h9xkDOHprqcc6w9FiPsp
w+3XtItepetENfqQBU1zexpPBgW2IBxwRJZ4NTY4iPPPb32hm5yS7Hm1ocdYHSqnp+9MYaOmOk5k2nZbskg
qsnZhIXEeozsQDMmpTF2gems/ZXF/
ocKn3hyHRG9TvWuR8bEcKjX9cmMRU4rbko5cfOg9V855VVrN9bSVqc57bEF41KJzYilFV7KtheepTzH11Mf
k4Sv7l8FG+sKvZ536LyWmobm4kKC6GhqYV/f7mbdXsrOX1QH66ckIGI747/
DfuO8Nv3NrH7UA0zRqXzvzunsmH/EeIjQxmcGkt1fRPRYXaqnZJfqF0wBq6ckMHXJ/
WnvqmFJdvK6BMTxsQBSR7PsNuEd74zjRW7KqhtbOb0QX0IPYGmPHXq0eSilB8/fXMd6/
dWAnCgsh67TWi2aiBhdhvfPnsQP31zHYWlR9uuWb6znKc+3U6Ln4pJfk5S28ir+99c15aI3lhVzN7DtXz3v
Ny2c2samthRVk1uWgw2EW59YQUHKh1Lyjz72U5iwkP4nlMNJDo8hDlXjea+uWtoaHYEMTkniSf+30TiIkMp
rqjh6ieXUlLpaLK7ZExf/u5lLTERIT/HM/EoFQhNLkr5sX7fEZf3zS2G2dNyqG1sZtak/
pRXN7B+X6XHdS0thqgwu0vT2eScJMqq6hmSFktNQxPDfvEBQ1JjPZ7x6vI9bcnlky2lfOeV1VTVNdEnOoz7
Lx7ellhaLdlW5pJcAGIiQtsSC8BnhYf4cH0JozPj+dHra9oSC8B7a/dz+xmHGds/oYP/
Okr5pslFKT9OH9SHeRuOTViclJ3oMpz4vbX7vV534cg0rp3Un99/uIVD1fVcPaE/
3z1vMCLCva+u5sMNjpFXa/cewX3qS1xkaNvrn/9vPVV1TYBjJNhzn+/
0aPKqrG2ktqGZSKcBBu4LaAIs31XOz/
+3nobmFo9juveM6mxd1ogqIv1F5GMR2SgiG0TkHqs8SUTmi8g263uiVS4i8riIFIrIWhGZ4HSvm6zzt4nIT
U7lE0VknXXN42I1VPt6hlId9fAVo7l4dDp9osM4d1gqf5k13uX4OcNSiAj1/DWKCg/l9EHJ/
O+uqSz50bncc35uWz/Kil0VLue6N58lRDmSS1NzC/
vchg7vPlTNry8f5ZKQCsuqGfvgPKY8vJC5KxzDn88dlkpYyLG4bAKVtQ1eE8uAPlFM8TKJUqkT0ZU9dE3Af
caYEcAU4C4RGQH8BFhojMkFFlrvAS4Ccq2v24EnwZEogAeAyUA+8IBTsngSuM3puhlWua9nKNUhCVFhPD5r
PCt/
cQHP3TyJjIRIl+NRYSHMmpTlUibif9ju+CzX5qcQt6rLsh3llFXWE2K3cf7wNJdj1fXNPPD2Bo+E1NBsKKm
s48dvrmXbgSr6J0Xx0q35nDsslWmDk3nmhjwyEqM8YrlifAZXjc/kwr8s5pLHl/
DRhhKfcSvVEV2WXIwx+40xq6zXVcAmIAOYCbxonfYicLn1eibwknFYBiSISF/
gQmC+MabcGFMBzAdmWMfijDHLjGNM50tu9/
L2DKUC9uf5Wxn9q3mMefAjHv1oCwBriw8zb0MJ1fVNbed97/
whTBzg+HsnzG7jB9OHktXH84O81S8vHUFi1LGmr2YvPf+Ltjia4v547VhunZpDmNNIraq6Juw+BpMZAyt3O
2pGUwb24bmbJ/Gvb07m/
BFp3Hx6NknRYW3nnpGbzFlDknl0wVZ2Hqxmw75K7nx5FUXlnk1qSnXUSelzEZFsYDzwJZBmjGltqC4BWv80
ywCcpzQXW2X+you9lOPnGUoFZMm2Mh5beGxy4uOLCllbfIRPrHkuyTFhzL3jNAamxBBiFy4YkUZ6bAQXjEz
j8vEZbC87ypKtZeSmxTJ1sOt8l+1l1VQ4rQ/

```
mb7ZLXEQoP714GC98sdOlPDzUTqjd5rWvxNf+MQP6RPPxfWczf9MBkqJDyRuQRP7DC1zOaWoxLN1+iP5Jvp
OjUoHo8uQiIjHAG8C9xphK5/H7xhgjIl265r+/
Z4jI7Tia4MjKyvJ2ijpFrS0+4lH2idMEyoNHG3j60x387uox3PHPlW1Lo7y3fj9rig/
zz6W72yZN3jo1h19edmwQQE1DE+6EY0mmX3wEM0b1bTsWYrdx4ch0Plh/
rMlq5rh+nDssldteWulxr799XOh1IUqA+KhQrp6YCcDcgj3UNXr2wYzopysjqxPXpclFREJxJJaXjTFvWsU
HRKSvMWa/1bRVapXvBfo7XZ5ple0FznYr/8Qqz/Ryvr9nuDDGPAM8A479XI7rh1S90pSB7c/
vqKhpoKi8xmPNrbkFRW2JBeClpbu457xc3l23j+c/30VDUzPuDHDJ6L6M7R/
PVRMyiXcaMQbwh2vGkpUUxeqiw0zJSeLOcwYTardx0SjXpAOwZNtBGppaXDr0vamu94wjMtTOqAzHWmPby4
6ypaSK/JwkkmM8J5Aq5U9XjhYT4FlgkzHmUadDbwOtI75uAt5yKr/
RGjU2BThiNW3NA6aLSKLVkT8dmGcdqxSRKdazbnS7l7dnKBWQiQOSuOPMgX7PuTavP9HhIR4d8k1uI7JajO
HzwjJ+9t/1FJYeZU+596VZctNiuP3MQfSJCaemoYmVuyuoqnM0e8WEh3D/xcOZe8dpfH/
6UCJC7dhtwpPXT/SoaeQkR7ebWAC+Pqm/x3nX5Dn+XvvH4h2c96dPufPlVUz67QLu/
vcq6ho9k5FSvnRlzWUqcAOwTkS+ssp+CswB5orIbGA3cK117H3gYqAQqAFuATDGlIvIr4EV1nkPGWPKrdd3
Ai8AkcAH1hd+nqFUwM4dlsrTi3e4lA1Pj2VURjxfG9ePM3JTALj9zIE88cn2tnPqm1wrwZePy2DNXs9mNmc
2oW1k2BOfFPLn+VtpbHZMxPzrN8aTmxpLdUMjNQ3NjOgb7zKn5Y9Xj+WOfxVQVF5Lamw4c64c3XbMGMN/
VhazZNtBhqXHcuvUnLZro8JCeO+70/jhf9ZScqSOC4an8qMLh1HX2MyfF2x1uge8u3Y/CVGh/ObyY/
dWyh/d5tii2xwrdy0thpl//5x1VmIItQtz7zjNY1MwgPP+9InL0vetBqVE8+E9Z/D+
+hLuefUrj+MAiVGhXDE+g1umZrNhXyXf+tcql+MRoTaXvpHIUBuTspMYl5XIjVOyeOjdTSzYdID+iVHcOi2
bt9fsY//hOi4b248Qu/Cnj44ligtHpvH0Da470tY1NnPf3DV8sH4/0eEhfOfcwTzy/maPgQapseEs/
9n5fv/N1KnH1zbHmlwsmlyUN/
sO13LNU0vb9kGJiwjh2kn9iY8MZbE1Guze83O55sml7PYxhHdUvzgiQm0crm1yWYPMm9TYMEqrGvye4yw8x
EZ907HEI+KoabRKig6jvNr1fkvvP5e+8ZEUlh7FGMPCzaXM+WBzu8+anJPEa3ecFnBs6tTgK7no8i9K+fHG
ymKXDbYq65r4vyXHhgWv2FXBpv2VXJOXyR+dagjOvK095ktHEgvgkljANbEAXgcP3Pr8CpqNYesBR6JLjQ2
ss97bqtBK+aJraCvlR1FF+xMKV+85zKM+EsvxcJ4gmRQV6vvEACRGel6/
qaSqLbGAY3+ZQLyzdr9OsFQB0+SilB/
uI8F88ZwtcvxaFzOeNjiZOreaSWg78USFuu6OWXS4zseZrkb0jfW6RpozY3SBSxU4TS5K+fDljkP8e3lR+y
cGKCLExpC0GAJLV44962vddrtsNsbvL23NcQ4X3rS/yuuESmdRYTb8ryeg1DGaXJTyorG5hV+
+td6j3AacNSTluO45NTeZqydkdujjOd2tn6PFdG4tqVUgMdU0tHDrCwU0ellZWSl3mlyU8uLBdzaw5YDnyK
4W4FtnDeLS0ek+r/W1Q/GiTaUs31Xu/aAP+yvrmTq4D2fkJrd/
coD6xkcQG3F8Y3lKq+rZvL+q02JRvZcmF6W8eGPlXp/H7vhnAe+u8700va/R/
QZYvvNQwP04rdYWH+G3V4wi1NdSyB0wICmSvvERbRuQdVRkqJ0BybqopWqfJhelvIgOt/
s8VnmcH8yOa5t5++6p5Od437/O2y9kVV0TM/68mEvH9CM3NYbkmDASvIwCC8Tu8lpW7TncoWtal/
uPDrMz56rRxEWc2Ag2dWrQ5KKUF/YO1i46omBXBSPS4zy2N44MFV6anU+al3knNY0t/
Hf1Xv5wzRj+e+dUPrj3DFJiwjzO6wqtu1dWNzSzSZvEVIB0hr5FZ+irVkfrmxj1wLygPT8+MoQjtd5rR85L
wdht4nWjsa4UYoOVP59O/AnOv1G9h68Z+lpzUcpNVKidvvERQXu+r8QCuAwXPtmJBaCp5VhNRil/
NLko5cZmEx65cjSR7UwqbOVrdFhvNDYznpQAl4tRpzZNLkp5cfbQVP5925SAJjyeSi3Lt7Wzx41SrTS5KOX
D+KxE/
nDNWLL7RB336KzeJCLU1qnzbVTvpslFKR+e+nQ7f5i3GRHh+ilZwQ4n6OoaW3jwnY3BDkP1EJpclPLijZVF
zPlgMwcq69l5sJq/fby9/YtOAe+v3R/sEFQPoclFKS/
+snBbsEPolpy3V1bKH00uSnlRciSwpepPNfZTaWicOiGaXJTyoqn5FBoC1gHVDce3pL869WhyUcqLKG3+8W
pyTlKwQ1A9hCYXpbwY2z8h2CF0T6I1OhUYTS5KeVFYqgs0erN8Z0WwQ1A9hCYXpdzUNDRRWtUQ7DC6pRrtc
1EB0uSilJvHFugwZF86utGZOnVpclHKzcrdHduK+FRi108MFSD9r6KUmyFpscEOoduKCNFRdCowmlyUcnPL
1Jxgh9BthYXoR4YKjP5PUcpNblosZ+rqv15FaHJRAdL/
KUq5aWpuYcUuHXLrzV5dFkcFSJOLUm4+KzxIbaMOufUmCDsrqx6qy5KLiDwnIqUist6p7FcisldEvrK+LnY
6dr+IFIrIFhG50Kl8hlVWKCI/
cSrPEZEvrfLXRCTMKg+33hdax7O76mdU6lR0oFJrL6p9XVlzeQGY4aX8z8aYcdbX+wAiMgKYBYy0rnlCROw
iYgf+DlwEjAC+YZ0L8DvrXoOBCmC2VT4bqLDK/2ydp1TApg3W/hZ/
kqLDgh2C6gG6LLkYYxYDgU4YmAm8aoypN8bsBAqBfOur0BizwxjTALwKzBQRAc4FXreufxG43OleL1qvXwf
Os85XKiAl+pe5TzaBUJ3sogIQjP8ld4vIWqvZLNEqywCKnM4ptsp8lfcBDhtjmtzKXe5lHT9ine9BRG4XkQ
IRKSgrKzvxn0z1CtFhIcEOodtqMVBT3xjsMFQPcLKTy5PAIGAcsB/
400l+vgtjzDPGmDxjTF5KSkowQ1HdSI125vv13jrd6li1L+DkIiLTROQW63WKiHR4ppkx5oAxptkY0wL8A0
ezF8BeoL/TqZlWma/yQ0CCiIS4lbvcyzoeb52vVECSorRPwZ9cXcFABSCg5CIiDwA/
Bu63ikKBf3X0YSLS1+ntFUDrSLK3gVnWSK8cIBdYDqwAcq2RYWE4Ov3fNsYY4GPgauv6m4C3nO51k/
X6amCRdb5SAVm8VZtI/Tlco81iqn2BNi5fAYwHVgEYY/aJiN8/
X0TkFeBsIFlEioEHgLNFZBxggF3AHdb9NojIXGAj0ATcZYxptu5zNzAPsAPPGWM2WI/
4MfCqiPwGWA08a5U/C/
xTRApxDCiYFeDPqJSDbojlV9/4yGCHoHqAQJNLgzHGiDh+60Qkur0LjDHf8FL8rJey1vN/C/zWS/
n7wPteyndwrFnNubwOuKa9+JTyZW3xkWCH0G2F2WFoujaLqfYF2ucyV0SextHPcRuwAEefiVK9TpNOQ/
epsRm0lVkFIqDkYoz5I445I28AQ4FfGmP+2pWBKRUssyZlBTuEbsugNTsVmICaxaxO9iXGmPnW+0gRyTbG7
```

```
OrK4JQKhpzkaKLDbFQ3tAQ7lG6psk479FX7Am0W+w/g/
JvWbJUp1evUNzVrYvFjfFZi+yepU16gySXEWn4FAOu1TgZQvdIrX+4JdgjdWky4rmCg2hdocikTka+1vhGR
mcDBrglJqeD6fLvOuVXqRAX6J8i3gJdF5G+A4Fi768Yui0qpIEqPDQ92CEr1eAElF2PMdmCKiMRY7492aVR
KBVFdk/a3+FNcUUNmYlSww1DdXKCjxcKBq4BsIKR1BXtjzENdFplSQdI/SWeg+1OvyVcFINBmsbdwLF2/
EqjvunCUCj5dmNG/
gcntLtChVMDJJdMY421XSaV6nX2Ha4MdQre2fm8lozPjgx2G6uYCHS32hYiM7tJIlOom6hp0Pxd/
EqJCgx2C6gECrblMA24WkZ04msUEMMaYMV0WmVJBMmGAThL0p3+Sduar9gWaXC7q0iiU6kZ0vxLf7BLsCFR
PEehQ5N0AIpIKRHRpREoFWYrOc/
EpSmfnqwAFuhPl10RkG7AT+BTHRl8fdGFcSgXNroM1wQ6h2zpa10Rjsw5FVu0LtEP/
18AUYKsxJgc4D1jWZVEpFUQRYYH+Wpx6dCcXFahAf4sajTGHAJuI2IwxHwN5XRiXUkEzVOe5+LV4a2mwQ1A
9QKDJ5bC19MtiHGuMPQZUd11YSgVPqva5+LV6z+Fgh6B6gECTy0ygFvge8CGwHbisq4JSKpg+XF8S7BC6td
nTBgY7BNUDBDpazLmW8mIXxaJUt/D5dt1NwpcZI9NIjNatnFT7/NZcRKRKRCq9fFWJSOXJClKpkyk/
OynYIXRb4aH2YIegegi/
NRdjjPZsqlNOWpxO5fJl+U7dSE0FpkMzotwnURpjdD9Y1esM7xcX7BC6rSO6eoEKkE6iVMqNboTlW01jCw2
6n4sKgE6iVMpNaVVdsEPotqLDbITqAmMqADqJUik3K3aWBzuEbisy1E7rTrRK+RNon4v7JMpSdBKl6qXeWb
s/2CF0WzW6140KUEcmUdagkyjVKSC7j/
a5+KS1FhWggJKLMabaGNNijGkC3gP+ajWTKdXr3DJ1QLBD6LZqGprZXnY02GGoHqC9SZRTROQTEXlTRMaLy
HpgPXBARGa0c+1zIlJqXdNaliQi80Vkm/U90SoXEXlcRApFZK2ITHC65ibr/
G0icpNT+UQRWWdd87hYDcG+nqFUoF5aqiPs/amubwp2CKoHaK/
m8jfgYeAVYBHwTWNMOnAm8Eg7174AuCegnwALjTG5wELrP
Th2usy1vm4HngRHogAeACYD+cADTsniSeA2p+tmtPMMpQKytaQq2CF0a2uLjwQ7BNUDtJdcQowxHxlj/
gOUGGOWARhjNrd3Y2PMYsB92M1Mjq1N9iJwuVP5S8ZhGZAgIn2BC4H5xphyY0wFMB+YYR2LM8YsM8YY4CW3
e3l7hlIBsdm0X8GfMh2qrQLQXnJxni1V63bsePYNSjPGtA7FKQHSrNcZQJHTecVWmb/yYi/l/
p6hVEDqG3WSoD+zJmUFOwTVA7SXXMa2LlQJjHFeuBIYfSIPtmocXbqxXXvPEJHbRaRARArKysq6MhTVgwxJ
iwl2CN1a34TIYIegegC/ycUYYzfGxBljYo0xIdbr1vehx/
G8A1aTFtb31i3t9gL9nc7LtMr8lWd6Kff3DG8/3zPGmDxjTF5KSspx/DiqNxqVER/
sEJTq8U72ZuFvA60jvm4C3nIqv9EaNTYFOGI1bc0DpotIotWRPx2YZx2rtEazCXCj2728PUOpgHyxXUfZ+6
K9USpQHVoVuSNE5BXgbCBZRIpxjPqaA8wVkdnAbuBa6/
T3gYuBQhyTNW8BMMaUi8ivgRXWeQ8ZY1oHCdyJY0RaJI5FNFsX0vT1DKUCkhKrm2H50qXt2KpXEUe3hMrLy
zMFBQXBDkN1AyVHapnyyKJgh9Ft7ZpzSbBDUN2IiKw0xnisNXmym8WU6vbS4yOJj+yySr1SpwRNLkq5mb/
xAEdqdRa6UidCk4tSbl74YlewQ+i24iK0RqcCo8lFKTd948ODHUK3lRStgx1UYDS5KOXmhim6KrIvu8trdO
FKFRBNLkq5qazTD09fjIGmZh1hqtqnyUUpN0N1+Ref7ALxUcezOIc61WhyUcrN85/
vDnYI3ZbWWVSgNLko5WZPeXWwQ+i2WjS7qABpclHKTVK0jhZT6kRpclHKzSWj04MdQrc1sl9csENQPYQmF6
XchIXagx1Ct/X8zZOCHYLqITS5KOVmRF/
969ybayZmkhoXEewwVA+hyUUpNxGhdhIidbitu20HqoIdgupBNLko5cXEAQnBDqHb+ar4CDsP6kg6FRhNLk
p5MaDPyZlIGWrrWXs7NrW0BDsE1UNoclHKi/
5JJ6dvobGHTRwZkBQd7BBUD6HJRSkvjOlZNYqTIQfZg2cAABLjSURBVCrMTm1Dc7DDUD2EJhelvFi6/
WCn3KeHtXr5VdPQzCMfbAp2GKqH0OSilJtlOw4xf1Npp9zLdEKrV4jAjJFpJ36jTrBiV3mwQ1A9hCYXpdyU
HKnrtHt1Ro9Kk4HrJg9gcGrwV2ueOCAx2CGoHkKTi1JuUmM7d22xUPuJt419ufMgR2oaOyGaEzMxS5OLCow
mF6XcHKjqvJoLQGMnbK7131X7KDta3wnRnJhPt5YFOwTVQ2hyUcrN1EHJhId0r1+No91ka+E4XblABah7/
QYp1Q2kxkXwwi35RHSjBJPQDXZ/FIEfXjg02GGoHqL7/PYo1Y2cNqgP47K6xxIwNoE95bUduqYrRkB/
Y1IWfWJ0rxsVGE0uSnnx8pe7Wbajewy7PZ5J/
J097z87KZIfaK1FdYAmF6W8eOHzXZ12r5AumEkZYhOiw1z3nbEJRIR0zazNlNgIkqLDuuTeqnfS5KKUF8kx
nfNBmpEQQVMXrB/W1GKodluKpcXAkHTve9Ekx4S1m+T8HV2xu4Ifv762o2GqU5gmF6W8+OGMYV4/
jN1LYsN971ppA/Ye7txhzf6MyYznZxcPx9s4hOr6Jp9JbnJOEk/8vwn0i/ffn/
JaQRFF5TWdEao6BQQluYjILhFZJyJfiUiBVZYkIvNFZJv1PdEqFxF5XEQKRWStiExwus9N1vnbROQmp/
KJ1v0LrWt70QpP6mSYkJXIt88e5FFuc0o4oTbhT9eO4+bTs73e42QsTp8YFUqU1Ty2tvgIcz7czAu35HucV
9voGU1seAgXjkwjJTac7aVHKalsfx5NjS5cqQIUEsRnn2OMcV4d8CfAQmPMHBH5ifX+x8BFQK71NRl4Epgs
IknAA0Aejv7LlSLytjGmwjrnNuBL4H1gBvDByfmxVG+R3cdzefnmFkNcRAiPXDmaSdlJpMZFMH1kOjedns2
izQf49buds7DjRSPTqGtqIT+nD3//
uNDnPJeMxEjW761se796z2GO1jfTLz6CfW7L2ITZhQZrQmdGQiSjM+P4cP0BAN5lf7sx2W3C0PTY4/2R1Cm
mOzWLzQRetF6/CFzuVP6ScVgGJIhIX+BCYL4xptxKKPOBGdaxOGPMMmOMAV5yupdSAbt0bF/
6J0Z6lFfWNTFxQJLLfvI5ydGMyYwP+N7t/eJ9bVwGz9+Sz7fPHsRvLh/
p8zxvCbC6vonrTxvgUd7QbHjwayN44ZZJzLv3DBZs9L84Z5TbgIFbfNTQlPImWDUXA3wkIgZ42hjzDJBmjG
n986kEaF0GNgMocrq22CrzV17spVypDgkPsfPMjXlc9NgSl/KMhEhiI+x8tq2MJz/dzpqiw/
SJDudXXxtBVJg9oKYjf138NoHxTmt4eTs3xAaTsvtgDISH2KhvcjR7pcWFc8HINAT447wtHsOYs/pEc/
bQVFpaDPGRoRyqbvAag90m/P6qMVTWNfFVUQWTspO4emJmuz+XUq2ClVymGWP2ikgqMF9ENjsfNMYYK/
F0KRG5HbgdICsrq6sfp3qggl3lbR/eAgxMiWb2tIFMeWQRVXXHmqqO1tdw6wsF/PKy4Ty2sJDD7Swy6e0/
t12EmIgQHpo5kvT4Y7WibQeOepwbFmJn6Y5DjutswpXjM8hMiuK6/
```

```
CziIkJZuOmAR2IJs9t4+L2NzH5hBUPSYpmYnchHGw543FuA1+84jfHWCsjXTdbfDdVxQWkWM8bstb6XAv8F
8oEDVpMW1vfWOvteoL/T5ZlWmb/yTC/l3uJ4xhiTZ4zJS0lJOdEfS/Uy+4/U8qt3NrbVCgwwY1Q6Ty/
e7pJYWhngoXc20dzBocd2m5CZEMHl4zNYdN9ZzBznWtGeNjjZ4xrn2lFziyE2IoTvXzCkLSkN8NJc1tDcwr
bSaloMbC6pYqGXPWsEuGRM37bEotTxOunJRUSiRSS29TUwHVgPvA20jvi6CXjLev02cKM1amwKcMRqPpsHT
BeRRGtk2XRgnnWsUkSmWKPEbnS6l1IBe21FkUei2FJSRXGF76VYDHhNPP40txiKD9fxxqpi7vvPGs8TBCJD
Hf0fNoHLx/X1OGXVnsMu7wenxnDPebl+l/v3lgQN8O7a/
S6bgq3eU8FTn27ni07anVOdGoJRc0kDPhORNcBy4D1jzIfAHOACEdkGnG+9B8dorx1AIfAP4E4AY0w58Gtg
hfX1kFWGdc7/WddsR0eKqQ56fOE2/rJgm0f5OcNSmTEq/YTuHebnA/
+TLWXc9fJKl9Fhv3xrA7WNjppKi4EVuyo8rttSUulR9r0LhrDs/
vOYPtz7Lpb+Vn7esPcI4FgG54onvmDOB5u57h9f8teFnv8mSnlz0vtcjDE7gLFeyg8B53kpN8BdPu71HPCc
l/ICYNQJB6tOWS9+scvlvQj8YPoQrsvPYua4DPrFR/
DiF7tpaO74bJarJmSyYHMpZVXe55W8t66EusYWnr15EgB7DrlOXCz1cp3Nx+z7PjHh3HvBEJbuOESVlbBsA
pkJkezxUwN7fFEhZw5J4YmPt7uUP714B3eeMxh7Fyxpo3qX7jQUWaluIyLUbea9gRmj+iIixISH8LNLRvDS
7HwyEhxDlTMSIjkzN4WEyPb/
Xntj9V4mZCXwp2vHMufK0V6brhZuLuVonWNQgHtNKS0uwmOI9K1Tc3w+b0S/OD7+4dn8/
qoxvHRrPtsfvrjdfVnKqxv428eFtBjXpjP390r5EsxJlEp1W9fkZfCXBYVt7w3w948LmZCVyCvL95ASG05E
iI3oMDvXT85i9hk5XPbXzwPa1KuhqYV5Gw6QHBPOb68YzY6DR3lm8U6P8+559SuevXkSj1w5mhCb8OZqx7g
Ub30+WUlRfp+ZHBPOtZOOjX8pqmh/GZc1RYcpdxuqfOvUHK21qIBoclHKTVVdI68sL/
IoX7mrgjdXeQ483Fp6lM0lVR6JJTEqlAo/
Q5ILrL6T718wlIWbStleVu1yfOHmUjbuq2REvziXCZvevLduP7PyAx8yPDA5mtVFR/
yes+NgNc4VlWsmZuqy+ypg2iymlJsP1pVwwMs6W4eqfa+9tb3Mcy5KewlhvLUZWUSoncdmjfd6TrFVw+gb7
/9euw5WU1nnf26Ns59fOpJEt90t+8VHcP2ULC4enc7tZwzEvQWsdUi2UoHQ5KJUAGaMTCMqzHdFf2S/
eCbnJLW9H9s/
gYevGNW2dH+ITZg9LYfc1BhsAucPT+MnFw1rO39URjx3ui2UmRQdxunWHJdr8jKZ6GfuSVFFLQ+
+vbHtfW1DM098Ush3XlnNfwo8a2ETBySy9P7zeP1bp/
HU9RN46voJLPrB2dwyNYfr8gdw89RsIkJdPx68zbdRyhdtFlPKzeSBSR5ldpuN756Xy8//
t97jWFiIje1lR5k+Io3vnjcYu81GfnYSNpvw2Y/
PZU3RYXKSo9utyfxoxjBG9otnbkERiVGh3HnOYGLCHb+iUWEhvPHt07nz5QLeX+c5qx7g061lba/
veXU1H210nPfOmn2UVtVz1zmDAcdcnQ/W76dfQiQzx/
UjPMQxeOGR9zfx9OIdAKTHRfCbmaN4adluyqsbuGZif67J0+VfVOA0uSjlxlun/
KHqeq6fMoBBKdF877Wv2panjw6zU93QzP4jdby4dDd2m41fXjai7bqIUDuTB/
YJ+NmXjOnLJWM8J0m2unhUP5/JZXhfx4rFR2oamb/
J9ZzXVxZz1zmDWbr9EDc8+2Xb3i7vrNnHP2dPpqi8hmeW7Gg7v6SyjuW7ynn77mkBx66UM00uSrkZnh7HoJ
Rolw72S8f0A+C0Qcks++n57DpYzcGj9Vz91FKXaz/dWgqMoKsMSfO+5P2w9Fge/
Jpj9eTwUBvRYSEuSbKPtUXxi1/
sctk0bMm2g2wuqaSuscWjj8XXPBylAqF9Lkq5sdmEf31zMtdNzuKM3GQeuXI0109xXcI+OzmaMZkJbR/
arYb52Ga4s3y8xXM9sIdmjuTDe89kYEoM4Kgt/WD6EFq3yIsKs3PfdMcoL2/
DiENsQm19E2mxrjtRXjlBm8HU8dOai1Je9I2P5OErRvs9JyzExh+vGcsPX1/LwaP1jOwXx/0XD/
N7TaBaWgxbDlSRHhdBolMCa/SyIsBbq/
eRGhvOjFHHmtNunprDucPS2FxSSX5OEglRjnvcOi2HBZsOtI38mj4ijWc/
29k29DrEJkzOSeKG0wa43E+pjhKjM24ByMvLMwUFBcEOQ3UThaVHeWNVMTHhIXx9Un/
qm1r46ZvrWL2ngvycJH57xWjSrA76xuYWKmoaSI3132EfqOKKGm58djk7DlYTZrfx04uHcbM1A3/
f4VoufnyJ1yX9f3fVaL4+6dhcl6P1TfxtUSFrig6Tn5PEnecMIjzETlF5DR9tPEBGQgTD+sZyzh8/
dWkSmzEynadumNgpP4vq/
URkpTEmz71cay5KuSksreKyv37etljkK8v3kB4XQcFux6THBZtKaWpZ27ZXfajd1mmJBeAvC7ax46Cjv6eh
uYWH39/MzHEZJEaH0S8hkve/ewbPfbaT//vMdVb/ayuKXJLLD/+zhg/WlwCwdMchyo7W8/
AVo+mfFMXsaTltP6v735c1je1vdqZUe7TPRSk3cwuK2xILOJZbaU0srZZZG3V1BfeFKhuaW9h7+NiSL/
0SIrnznMGEuPWfJEa5Np/
N21Dicvy9tftxNzg1lilOQ69F4HrdHEx1Ak0uSrnxWLQSyO7junZXXEQon23rmv1N3BeqDAuxcelfP+PqJ7
+gqNyReJKiw1wmXcaGh/Dd83Lb3ofabaS7zavJdFvsstVzN0/
il5eO4Jap2bx2+2lMH3liWwooBdrn0kb7XFSrA5V1zPzb55RU1gGO2ey/
uXwU981dw8b9rvumPDZrnMfOkSfKGMPzn+/iw/UlbC2tculfmTY4mX99c3Lb+y0lVew8WM1pg/
oQ77bS8YKNB7jn1dVUNzQTHxnKMzdM7NCcG6UC4avPRZOLRZOLclZV18iCTQeIDgvhnGGphNptFJXXcMbvP
3Y5Lz87ibnfOq1LYmhqbmHwz1z3uYsJD2H9gxcGfI+quka2lR5leHockWGeNTKlTpR26CvVAbERoVwx3nWe
R0SoHbtNXLYHjgrvug/
sELuNCVkJLlsY5+d4Lk3jT2xEKBOyfK9JplRX0T4XpQKUEhvOTadlt72PCLVxt7VeV1f589fHcdrAPkSF2T
l7aEq7c2+U6i60WcyizWIqUCt3V7DzYDVn5ia3uxhlq/qmZj7eXIYInDM0lTA/+9cr1ZNos5hSnWTigES/
y9+7q6pr5MonvmBbqWPPl+F943jj26f5XcJfqZ5O/3xSqou99dW+tsQCsGl/
pdc5J12tsLSKRZsPUB3AVsxKnSj900mpLubtw9zbsv5d6Q/
```

```
zNvP3j7cDjjkyr9w2haHp3ldYVqozaM1FqS522dh+xEUc+zsuMSrU754tna20so6nPj22V0t5dQOPL9rmcs
6H6/dzyeNLuODRT3n5y90nLTbVe2nNRaku1i8hkne+M41XlhdhE/
hGflanrkXWnsO1jS7DpwEOHT22V0th6VHu+vfqtnN+9t/
1ZPeJZqpua6xOgCYXpU6CAX2i+clFnbMcf0cNSYtlbGY8a4qPtJVdPbF/2+svth/
0SD6Lt5ZpclEnRJOLUqeAF2/N59nPdrL7UA0XjUrnotHHmuWG9/
Xc4MxbmVIdoclFqVNAQlRY226U7iZlJ/GdcwfzzOIdNLcYrsnL5LKx/
U5yhKq30eSilOK+6UO58+zBNBtDTLh+LKgTp/+LlPKjrrHZ6xL8vZEubKk6kyYXpbwoKq/
hu6+uZvWew+SmxvDoteMYnRkf7LCU6jF0notSXvzirfWstlYj3lZ6lHtfWx3kiJTqWXptzUVEZgCPAXbg/
4wxc4IckupB1jkN2wXYXlZN9k/eC1I0wRct8PTNE5g29ORN/
lQ9W6+suYiIHfg7cBEwAviGiIwIblSqpyguq+RQdUOww+hWqg1c//
yqUzrBqo7plckFyAcKjTE7jDENwKvAzCDHpHqIaX9aEuwQurVvPr8s2CGoHqC3JpcMoMjpfbFV5kJEbheRA
hEpKCsrO2nBKdWTLdpyKNghqB6gtyaXgBhjnjHG5Blj8lJSUoIdjlI9ws8uyg12CKoH6K3JZS/
Q3+l9plWmVLt2zbkk2CF0a7PPGhLsEFQP0FtHi60AckUkB0dSmQVcF9yQVE+iCUapE9Mrk4sxpklE7gbm4R
iK/
JwxZkOQw1JKqVNGr0wuAMaY94H3gx2HUkqdinprn4tSSqkg0uSilFKq02lyUUop1ek0uSillOp0Yoxp/
6xTgIiUAbuDHYdSXiQDB4MdhFI+DDDGeMxC1+SiVDcnIgXGmLxgx6FUR2izmFJKqU6nyUUppVSn0+SiVPf3
TLADUKqjtM9FKaVUp9Oai1JKqU6nyUUppVSn0+SilFKq02lyUUop1ek0uSillOp0/x/
R7sbR0WMrDQAAAABJRU5ErkJggg==\n"
          },
          "metadata": {
            "needs_background": "light"
          }
        }
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "plt.hist(df['Balance'])"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 369
        },
        "id": "QlRPX3ftUyfY",
        "outputId": "4060536b-0e2b-4b7b-885b-4d8f110610d0"
      },
      "execution_count": "null",
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "(array([3.623e+03, 6.900e+01, 3.600e+02, 1.173e+03, 2.081e+03,
1.747e+03,\n",
              "        7.290e+02, 1.860e+02, 3.000e+01, 2.000e+00]),\n",
              " array([     0.   ,  25089.809,  50179.618,  75269.427, 100359.236,\
n",
              "        125449.045, 150538.854, 175628.663, 200718.472, 225808.281,\
n",
              "        250898.09 ]),\n",
              " <a list of 10 Patch objects>)"
            ]
          },
          "metadata": {},
          "execution_count": 17
        },
        {
```

"output_type": "display_data",
"data": {
"text/plain": [
"<Figure size 432x288 with 1 Axes>"
],
"image/png":
"iVBORw0KGgoAAAANSUhEUgAAAYAAAAD4CAYAAADlwTGnAAAABHNCSVQICAgIfAhkiAAAAAlwSFlzAAALEgAACxIB0t1+/
AAAADh0RVh0U29mdHdhcmUAbWF0cGxvdGxpYiB2ZXJzaW9uMy4yLjIsIGh0dHA6Ly9tYXRwbG90bGliLm9yZy+WH4yJAAATdElEQVR4nO3db4xd9X3n8fen5k+qJiqmTC2vbaJoyqbKMY4DyBSNjIPisuubFKTDzgJRoyqbKMY4DyBSNjIPisubsLIQ+tcvoisuBrtGyC/yGF7vkXQmJlbl
iubsLUiJ1q5r2AWmySGS1CRLalMYR3pAqG0KbRFgJXeoSpCoPAgxdx2CIy0AcYcvBk5iQRtGyC/vdB/
fn1cWZ8dyxZ+ba83u/pKt77vf8zjm/n89lPpw/995UFZKk/
vzKuDsgSRoPA0CSOmUASFKnDABJ6pQBIEmdumDcHTidyy67rDZu3DjubkjSeeXpp5/+cVVNzNfunA6A6jRs
MjU1Ne5uSNJ5JckPR2nnnKSBJ6pQBIEmdMgAkqVMGgCR1ygcCQpE4ZAJLUKQNAkjplAEhSpwwASerUOf1J4LO
1cde3xrLdw3/aCzblaSF8AhAkjplAEhSpwwASerUvAGQ5G1J+N/qeG1JnkbkjSeeXpp5/+cVVNzNfunA6AjRs3
MjU1Ne5uSNJ5JckPR2nnKSBJ6pQBIEmdMgAkqVMGgCR1ygcCQpE4ZAJLUKQNAkjplAEhSpwwASerUOf1J4LO
1cde3xrLdw3d/aCzblaSF8AhAkjplAEhSpwwASerUvAGQ5G1JnkzyvSQHk/
xpq38xyQ+S7G+Pza2eJPclmUryTJKrp61rR5KX2uvH2ZJ0ZMCbadwWhVRYsMomsvrf0D4+fvwyQDbVaPYCtpbTbxH40hmSaK8Bb1Sr4PTuKSSrJQgqLyGkoCqZgDqLSQQESkA1IYGs3JXqeaNkj8fALFFAcxK9Gio4lcaSn75wTR9Ta0rXSE3GDZuHhmLBzY6sTgDd6IhyJqm4ic5hGQCxXhEyHpBr4z/oC+Yz1DSupM5eGWKZFlNdH4ZfhaWnyxD+RdQpK6IdQrO2GKf7z1ZTpFqLQg==
Ly9vLA96jSL3AB8qS33XeCSJGuB64B9VXWi/
dHfB2w9u+5Lks7USNcAkqxKsh84zuCP+BNt1l3tNM89SS5OLH2m1ueqnbmtnkqkkUzMzMwscjiRp
VCMFQFW9WVWbgfXA1Un+BXAH8FvAvwIuBf5oMTpUVburarKqJicm5v1FM0nSGVrQXUBV9VPgcWBrVR1rp3l
eB/4bcHVrdhTYMLTMY+labqy5JGoNR7gKaSHJJm/
5V4IPA99t5fZIEuBF4ti2yF7i5i7Q10LfBaVR0DHgW2DJFmdZDWwpdUkSWMwyl1Aa4cKxBgPVZvAE6pQBkaavKrKrmWKHuV
23lqpDVbW7qiaranJiYmKpNiJ3VvQXUBV9VPgceA9wCVLmiz1gNH2/RRYANm/RRYANAm//
rwE+G67MsI0laZqPcBTSR5R5JI2/avB4HnGQTBTSR5JI2/avB4HnGQTBR1qzHcDDbXpve02b/
+2qqlbf3u4SuhzY7i6GxNcfy3w4B4BAd6K1NxgTWpGsAPhXMyiJI4mgRxaA3/o6ACmw/Hf2b/
xFkmngBIM7f6iqg0keAp4D3g3Buq6o3F3c4kqRRzRsAVr6J3BqWXcr2bW3ZbFqWAvuJWAe4W+b7JuPW1O3J7dGy3Uk8SEhSAdggmeDBgWhQf2L
Msa67gLsW3k1J0mLzk8CS1Ck+CS1CkDQJ3Z6ZQBIEmdMgAkqVMGGCR1ygCQpE4ZAJLUKQNAkjplAEhSpwwAS
KnDABJ6pQBIEmdMgAkqVMGGCR1ygCQpE4ZAJLUKQNAkjplAEhSpwwASerUvAGQZEOSx5M8l+Rgko+1+ieSH
E2yvz22DS1zR5LR5LpJIeSXDdU39pq00l2Lc2QJEmjuCEnm8GCEKm8Af1BVfe
kHcDTSfa2je6+n/1v5+lAjQ1+ue2jS6nLm+7JMr0J8CFgmjkfDfR6tl4nCqdKKZRtmMpaiJN5B6V9xdTqNjZs
m+Sft9mfAz4IHAHmQGHASrK3qp4GqkbjqjYvGzQ1tTnBUQTcsByYkXoxNPEnaAeeDoQxLBzo4Ciw6/EBl2b9
9CkhnLNyxJ0nxGOQX0xBQcZAHEVY0A1MWPpY9hhyP8DbFtDozexSQwOm0fRdGt6kqmgYAWt9b2Kd8hs8dsm7fhIV
V2Sj4nmFuDDA8WZQYWmAVSq8Qrfs2ya0qgaf+F2eGtDHIJtQ8ShnMBa5GtiPUvQ/lg2Ay83vu+ZWuW==
g2AqjpWVX/
fpv8ReB5YB9wAPNCaPQDc2KZvAL5UA98FLkmyFrgO2FdVJ6rqVWAfsHVRRyNJGtmCLgIn2QhcCTwBrKmqY2
3Wj4A1bXod8PLQYkdaba76qdvYmWQqydTMzMxCuidJWoCRAyDJ24GvAR+vqp8Nz6uqAmoxOlRVu6tqsqomJ
yYmFmOVkqRZjBQASS5k8Mf/y1X19VZ+pZ3aoT0fb/JWbd7vnAfJjOB4/cV/Ih7QxZ0psnHaHlX4KTHWI3LsW/Adebpf+ZZeSL4+A4MmKhiw+aBYoOHfDIFhN/
k2QE8PFS/ud0NdC3wjtVV9CiwJcnqdvBsF9Lu0C+0CpHjPxJX/Qdh+vo/SuxJfQiQTbTzFKfkemhOQWb0JtH4+nRW4yYCJ+ktx2eqRZxoB7geNkV6YYJ60sqQHNJ2HoczTSHPB8MIctPjSxTqGy4tBDfzDLd==
A7we8AzSfa32h8DdwMPJbkV+CHw0TbvEWAbMA38ArgFoKpOJPkU8FRr98mqOrEoo5AkLdi8AVBV3wEyx+wP
zNK+gNvmWNceYM9COihJWhp+FYQkdcoAkKROGQCS1CkDQJI6ZQBIUqcMAEnqlAEgSZ0yACSpUwaAJHXKn4S
UztC4fo7Sn6LUYvEIQJI6ZQBIUqcMAEnqlAEgSZ0yACSpUwaAJHXKAJCkThkAktQpA0CSOmUASFKn5g2AJH
uSHE/
y7FDtE0mOJtnfHtuG5t2RZDrJoSTXDdW3ttp0kl2LPxRJ0kKMcgTwReAknfHtuG5t2RZDrJoSTXDdW3ttp0kl2LPxRJ0kKMcgTwRWDrLPV7qmpzezwCkOQKYDvw7rbMny
VZlWQV8ZCQV8DngeuAK4Hsc1kuBWTXpVpI0JvN+FWVfs/3V7
V2SjSOu7wbgwap6HfhkN6jZvuqpeAkjyYGVpr7HIJAfaKaLVrbYOeHmozZFWm6v+S5Ls

TDKVZGpmZuYsuidJOp0zDYD7gHcBm4FjwGcWq0NVtbuqJqtqcmJiYrFWK0k6xRn9HkBVvXJyOskXgG+2l0e
BDUNN17cap6lLksbgjI4AkqwdevlH4OQdQnuB7UkuTnI5sAl4EngK2JTk8iQXMbhQvPfMuy1JOlvzHgEk+Q
rwPuCyJEeAO4H3JdkMFHAY+H2AqjqY5CEGF3ffAG6rqjfbem4HHgVWAXuq6uCij0aSNLJR7gK6aZby/
adpfxdw1yz1R4BHFtQ7SdKS8ZPAktQpA0CSOmUASFKnDABBJ6pQBIEmdMgAkqVMGgCR1ygCQpE4ZAJLUKQNA
kjplAEhSpwwASeqUASBJnTqjH4SRTrVx17fG3QVJC+QRgCR1ygCQpE4ZAJLUKQNAkjplAEhSp+YNgCR7khx
P8uxQ7dIk+5K80n7XDOqTmBJFcNLbOjtX8hyY6lGY4kaVSjHAF8Edh6Sm0X8FhVbQIea68Brgc2tc
dO4D4YBAZwJ3ANcDVw5fBCr4eckAsh+5eutccB8+Nbq/b+ZL/XTrg/ZTGpmVcjHm+5t6yOn6
jlUJEkLaMzvQawpqqOtekfAWva9Drg5aF2R1ptrrokaUzO+iJwVcPyb
sAr7dQO7fl4qx8FNgy1W99qc9V/
SVXtrqrJqpcmJg4w+5JkuZZpgGwFzh5J88OOh+s3tbqBrgdfaqaJHgS1JVreLv1taTZI0JvN+GVySrwD
vAy5LcoTB3Tx3AwlXR1vwRYBswDfCuAWgqk4k+RTwVGv3ax6yao69TsSsYLjrkHvY0XlmnMqarl2I4k6RzjV0FIUqcMAE
nq1IoMgPP96yaSHE7yySBIYsSHMBAEl7yTJL9SaZa7dIk+5K80J5Xt3qS3NvGeiDJVUPr2dHav5Bkx1D9X7b1T7dlM4Yx7klyP
MmzQ7UlH+Nc2xjjmD+R5Gjb1/uTbBuad0cr/6Ek1w3vZ31/
uTbBuad0fr/6Ek1w3VZ31/
t5srnmj1r7YbLUhycXs93eZvXKbxbkjyeJLnkhxM8rFWX7H7+TRjPjf3/cAu8EdLgK+B1wx7n4
tcAyHgctOqf1nYFeb3gV8uk1vA8a4WctAC1xJcJf1E3TyHYaqAJJHaqzmcK6hvcO6Gsuzr95YVCJ/
C1wFPLucY5xxrG2Mc8yeA/zhL2yvae/di4PL2Mc8yeA/zhL2yvae/di4PL2nl51uvc38BCwvU3/OfDv2/R/
AP68TW8HvrpM410LXNWm3wH8QxvxvXit3PpxnzObmfl/U/
+mXaAe8BHh16fQdwx7j7tcAxHOaXA+AQsHboTXaoTX8euOnUdsBNwOeH6p9vtbXA94fqv9Fcfqb2m3zOPcyFv/
GC75GOfaxhjHPNcfhre8bxncOfeeud7f7Q/gj4ELWv3/
tzu5bJu+oLXLGPb3wwy+B2zF7+dZxnxO7ueVeApotq+bWDemvpypAv4mydMZfDUGwJqqOtamfwSsadNzjfd
09SOz1M8FyzjHGucc3A/
r8QAWAneW1VXAdcDtyX53eGZmAgL9jTK7yYjo/
5nNzPKzEAzvuvm6iqo+dq+Xhrd4d4T1dfP0v9XLCMc5xxrG2NRVa9U9U9X+BLzDY17
DwMf8EuCTJBafU37KuNv/XW/sll+RCB8n8IVa/
lFb2fZxvzubqfV2IAnNdfN5HkkiTvODkNXAs8y2AM01f278uY+nNmwHT/K84AkK9h2cUFZKhP8sFf1b
Bc9Z39/t/3IfBz7Slj/13+/kmD8CfLu1X1Lt3/5+4Pmq+uzQrQ+uzQrBW7n+ca8zm7n8dxYWQZLrxsY3D1/
UXgT8bdnwX2/Z0Mrvh/Dzh4sv8Mzu9BrwA/C1wAtPtcctQfbK9AV8E/
ivjuSD4FQaHwv+YvxnMW5djjHPNtY4xj/os2pgPtP+C1Q+3/pPX/EEN3as31/
m7vnSfbv8VfAhe3+tva6+k2/53LNN73Mjj1cgDY3x7bVvJ+PsSFKnVuIpEnSCAwASWqUASBJnTIAJKlTBoAkdcoAkKROGQCS1Kn/B2PCY/Bb/4SKAAAAAElFTkSuQmCC\n"
          },
          "metadata": {
            "needs_background": "light"
          }
        }
      ]
    },
    {
      "cell_type": "code",
      "source": [

```
        "#split data"
      ],
      "metadata": {
        "id": "uMk-JgFuU4iC"
      },
      "execution_count": "null",
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "x=df.iloc[:,0:12].values\n",
        "y=df.iloc[:,12:14].values"
      ],
      "metadata": {
        "id": "WC67FUjPU-wm"
      },
      "execution_count": "null",
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "x"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "QY-xPbBgVU92",
        "outputId": "0ba73df4-1abb-4c1a-9894-3d11288946db"
      },
      "execution_count": "null",
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "array([[1.0000000e+00, 1.5634602e+07, 1.1150000e+03, ...,
1.0000000e+00,\n",
              "        1.0000000e+00, 1.0000000e+00],\n",
              "       [2.0000000e+00, 1.5647311e+07, 1.1770000e+03, ...,
1.0000000e+00,\n",
              "        0.0000000e+00, 1.0000000e+00],\n",
              "       [3.0000000e+00, 1.5619304e+07, 2.0400000e+03, ...,
3.0000000e+00,\n",
              "        1.0000000e+00, 0.0000000e+00],\n",
              "       ...,\n",
              "       [9.9980000e+03, 1.5584532e+07, 1.5700000e+03, ...,
1.0000000e+00,\n",
              "        0.0000000e+00, 1.0000000e+00],\n",
              "       [9.9990000e+03, 1.5682355e+07, 2.3450000e+03, ...,
2.0000000e+00,\n",
              "        1.0000000e+00, 0.0000000e+00],\n",
              "       [1.0000000e+04, 1.5628319e+07, 2.7510000e+03, ...,
1.0000000e+00,\n",
              "        1.0000000e+00, 0.0000000e+00]])"
            ]
          },
```

```
        "metadata": {},
        "execution_count": 20
      }
    ]
  },
  {
    "cell_type": "code",
    "source": [
      "y"
    ],
    "metadata": {
      "colab": {
        "base_uri": "https://localhost:8080/"
      },
      "id": "YKMLVugxVWkS",
      "outputId": "13dbaaba-067a-4c23-9be1-4f09b3a73b27"
    },
    "execution_count": "null",
    "outputs": [
      {
        "output_type": "execute_result",
        "data": {
          "text/plain": [
            "array([[1.0134888e+05, 1.0000000e+00],\n",
            "       [1.1254258e+05, 0.0000000e+00],\n",
            "       [1.1393157e+05, 1.0000000e+00],\n",
            "       ...,\n",
            "       [4.2085580e+04, 1.0000000e+00],\n",
            "       [9.2888520e+04, 1.0000000e+00],\n",
            "       [3.8190780e+04, 0.0000000e+00]])"
          ]
        },
        "metadata": {},
        "execution_count": 21
      }
    ]
  },
  {
    "cell_type": "code",
    "source": [
      "from sklearn.model_selection import train_test_split"
    ],
    "metadata": {
      "id": "Xe74DUadVZef"
    },
    "execution_count": "null",
    "outputs": []
  },
  {
    "cell_type": "code",
    "source": [
      "xtrain, xtest, ytrain, ytest =
train_test_split(x,y,test_size=0.3,random_state=0)"
    ],
    "metadata": {
      "id": "imxXGVmXVovW"
    },
    "execution_count": "null",
    "outputs": []
```

```
    },
    {
      "cell_type": "code",
      "source": [
        "xtrain"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "dKh70SnOWUXr",
        "outputId": "4367c18f-96da-49a5-f7f4-b99c0b3dc636"
      },
      "execution_count": "null",
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "array([[7.6820000e+03, 1.5633608e+07, 2.5900000e+02, ...,
2.0000000e+00,\n",
              "        1.0000000e+00, 1.0000000e+00],\n",
              "       [9.0320000e+03, 1.5742323e+07, 1.6400000e+02, ...,
2.0000000e+00,\n",
              "        1.0000000e+00, 0.0000000e+00],\n",
              "       [3.6920000e+03, 1.5760244e+07, 1.3040000e+03, ...,
1.0000000e+00,\n",
              "        0.0000000e+00, 1.0000000e+00],\n",
              "       ...,\n",
              "       [3.2650000e+03, 1.5574372e+07, 1.2020000e+03, ...,
2.0000000e+00,\n",
              "        1.0000000e+00, 0.0000000e+00],\n",
              "       [9.8460000e+03, 1.5664035e+07, 2.1220000e+03, ...,
2.0000000e+00,\n",
              "        1.0000000e+00, 1.0000000e+00],\n",
              "       [2.7330000e+03, 1.5592816e+07, 2.6780000e+03, ...,
1.0000000e+00,\n",
              "        1.0000000e+00, 0.0000000e+00]])"
            ]
          },
          "metadata": {},
          "execution_count": 24
        }
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "xtest"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "KJ_QFNAiWV6V",
        "outputId": "ad3e963c-b846-49b3-b9de-1d04cec62adb"
      },
      "execution_count": "null",
      "outputs": [
```

```
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "array([[9.3950000e+03, 1.5615753e+07, 2.6910000e+03, ...,
1.0000000e+00,\n",
              "        1.0000000e+00, 1.0000000e+00],\n",
              "       [8.9900000e+02, 1.5654700e+07, 8.4600000e+02, ...,
1.0000000e+00,\n",
              "        1.0000000e+00, 0.0000000e+00],\n",
              "       [2.3990000e+03, 1.5633877e+07, 1.8570000e+03, ...,
1.0000000e+00,\n",
              "        1.0000000e+00, 1.0000000e+00],\n",
              "       ...,\n",
              "       [9.3080000e+03, 1.5680405e+07, 2.0890000e+03, ...,
2.0000000e+00,\n",
              "        1.0000000e+00, 1.0000000e+00],\n",
              "       [8.3950000e+03, 1.5597983e+07, 3.3600000e+02, ...,
1.0000000e+00,\n",
              "        1.0000000e+00, 1.0000000e+00],\n",
              "       [5.2340000e+03, 1.5591286e+07, 2.4530000e+03, ...,
1.0000000e+00,\n",
              "        1.0000000e+00, 1.0000000e+00]])"
            ]
          },
          "metadata": {},
          "execution_count": 25
        }
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "ytest"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "7mGyI5JsWZvS",
        "outputId": "65d6f3a7-e12d-42be-dbd8-ac093775d296"
      },
      "execution_count": "null",
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "array([[1.9285267e+05, 0.0000000e+00],\n",
              "       [1.2870210e+05, 1.0000000e+00],\n",
              "       [7.5732250e+04, 0.0000000e+00],\n",
              "       ...,\n",
              "       [1.6740029e+05, 0.0000000e+00],\n",
              "       [7.0849470e+04, 0.0000000e+00],\n",
              "       [3.3759410e+04, 1.0000000e+00]])"
            ]
          },
          "metadata": {},
          "execution_count": 26
```

```
        }
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "ytrain"
      ],
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "HSM2IcWRWfxm",
        "outputId": "c7588d22-a74d-4cd8-ce9e-45812b721bc0"
      },
      "execution_count": "null",
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "array([[5.5796830e+04, 1.0000000e+00],\n",
              "       [1.9823020e+04, 0.0000000e+00],\n",
              "       [1.3848580e+04, 0.0000000e+00],\n",
              "       ...,\n",
              "       [1.8142987e+05, 0.0000000e+00],\n",
              "       [1.4875016e+05, 0.0000000e+00],\n",
              "       [1.1885526e+05, 1.0000000e+00]])"
            ]
          },
          "metadata": {},
          "execution_count": 27
        }
      ]
    },
    {
      "cell_type": "code",
      "source": [],
      "metadata": {
        "id": "2ZDNd8rPWiLJ"
      },
      "execution_count": "null",
      "outputs": []
    }
  ]
}
```