```
{
 "cells": [
  {
   "cell_type": "markdown",
   "id": "0340c2e9",
   "metadata": {},
   "source": [
    "## Assignment 2\n",
    "### Setup"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 2,
   "id": "1aaa2b41",
   "metadata": {},
   "outputs": [],
   "source": [
    "import sqlite3\n",
    "import pandas as pd\n",
    "import numpy as np\n",
    "from sklearn.model_selection import train_test_split\n",
    "\n",
    "con = sqlite3.connect(\"database.sqlite\")\n",
    "\n",
    "# helper function to run SQL queries\n",
    "def sql(q, params=None):\n",
    "    return pd.read_sql_query(q, con, params=params or [])\n"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 3,
   "id": "b75a55e1",
   "metadata": {},
   "outputs": [
    {
     "name": "stdout",
     "output_type": "stream",
     "text": [
      "\n",
      "League (first 3 rows)\n"
     ]
    },
    {
     "data": {
      "text/html": [
       "<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>id</th>\n",
       "      <th>country_id</th>\n",
       "      <th>name</th>\n",
       "    </tr>\n",
       "  </thead>\n",
       "  <tbody>\n",
       "    <tr>\n",
       "      <th>0</th>\n",
       "      <td>1</td>\n",
       "      <td>1</td>\n",
       "      <td>Belgium Jupiler League</td>\n",
       "    </tr>\n",
       "    <tr>\n",
       "      <th>1</th>\n",
       "      <td>1729</td>\n",
       "      <td>1729</td>\n",
       "      <td>England Premier League</td>\n",
       "    </tr>\n",
       "    <tr>\n",
       "      <th>2</th>\n",
       "      <td>4769</td>\n",
       "      <td>4769</td>\n",
       "      <td>France Ligue 1</td>\n",
       "    </tr>\n",
       "  </tbody>\n",
       "</table>\n",
       "</div>"
      ],
      "text/plain": [
       "     id  country_id                    name\n",
       "0     1           1  Belgium Jupiler League\n",
       "1  1729        1729  England Premier League\n",
       "2  4769        4769          France Ligue 1"
      ]
     },
     "metadata": {},
     "output_type": "display_data"
    },
    {
     "name": "stdout",
     "output_type": "stream",
     "text": [
      "\n",
      "Match (first 3 rows)\n"
     ]
    },
    {
     "data": {
      "text/html": [
       "<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>id</th>\n",
"      <th>country_id</th>\n",
"      <th>league_id</th>\n",
"      <th>season</th>\n",
"      <th>stage</th>\n",
"      <th>date</th>\n",
"      <th>match_api_id</th>\n",
"      <th>home_team_api_id</th>\n",
"      <th>away_team_api_id</th>\n",
"      <th>home_team_goal</th>\n",
"      <th>...</th>\n",
"      <th>SJA</th>\n",
"      <th>VCH</th>\n",
"      <th>VCD</th>\n",
"      <th>VCA</th>\n",
"      <th>GBH</th>\n",
"      <th>GBD</th>\n",
"      <th>GBA</th>\n",
"      <th>BSH</th>\n",
"      <th>BSD</th>\n",
"      <th>BSA</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>2008/2009</td>\n",
"      <td>1</td>\n",
"      <td>2008-08-17 00:00:00</td>\n",
"      <td>492473</td>\n",
"      <td>9987</td>\n",
"      <td>9993</td>\n",
"      <td>1</td>\n",
"      <td>...</td>\n",
"      <td>4.0</td>\n",
"      <td>1.65</td>\n",
"      <td>3.40</td>\n",
"      <td>4.50</td>\n",
"      <td>1.78</td>\n",
"      <td>3.25</td>\n",
"      <td>4.00</td>\n",
"      <td>1.73</td>\n",
"      <td>3.40</td>\n",
"      <td>4.20</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>1</th>\n",
"      <td>2</td>\n",
"      <td>1</td>\n",
"      <td>1</td>\n",
"      <td>2008/2009</td>\n",
"      <td>1</td>\n",
"      <td>2008-08-16 00:00:00</td>\n",
"      <td>492474</td>\n",
"      <td>10000</td>\n",
"      <td>9994</td>\n",
"      <td>0</td>\n",
"      <td>...</td>\n",
"      <td>3.8</td>\n",
"      <td>2.00</td>\n",
"      <td>3.25</td>\n",
"      <td>3.25</td>\n",
"      <td>1.85</td>\n",
"      <td>3.25</td>\n",
"      <td>3.75</td>\n",
       "      <td>1.91</td>\n",
       "      <td>3.25</td>\n",
       "      <td>3.60</td>\n",
       "    </tr>\n",
       "    <tr>\n",
       "      <th>2</th>\n",
       "      <td>3</td>\n",
       "      <td>1</td>\n",
       "      <td>1</td>\n",
       "      <td>2008/2009</td>\n",
       "      <td>1</td>\n",
       "      <td>2008-08-16 00:00:00</td>\n",
       "      <td>492475</td>\n",
       "      <td>9984</td>\n",
       "      <td>8635</td>\n",
       "      <td>0</td>\n",
       "      <td>...</td>\n",
       "      <td>2.5</td>\n",
       "      <td>2.35</td>\n",
       "      <td>3.25</td>\n",
       "      <td>2.65</td>\n",
       "      <td>2.50</td>\n",
       "      <td>3.20</td>\n",
       "      <td>2.50</td>\n",
       "      <td>2.30</td>\n",
       "      <td>3.20</td>\n",
       "      <td>2.75</td>\n",
       "    </tr>\n",
       "  </tbody>\n",
       "</table>\n",
       "<p>3 rows × 115 columns</p>\n",
       "</div>"
      ],
      "text/plain": [
       "   id  country_id  league_id     season  stage                 date  \\\n",
       "0   1           1          1  2008/2009      1  2008-08-17 00:00:00   \n",
       "1   2           1          1  2008/2009      1  2008-08-16 00:00:00   \n",
       "2   3           1          1  2008/2009      1  2008-08-16 00:00:00   \n",
       "\n",
       "   match_api_id  home_team_api_id  away_team_api_id  home_team_goal  ...  SJA
\\\n",
       "0       492473              9987              9993               1  ...  4.0
\n",
       "1       492474             10000              9994               0  ...  3.8
\n",
       "2       492475              9984              8635               0  ...  2.5
\n",
       "\n",
       "    VCH   VCD   VCA   GBH   GBD   GBA   BSH   BSD   BSA  \n",
       "0  1.65  3.40  4.50  1.78  3.25  4.00  1.73  3.40  4.20  \n",
       "1  2.00  3.25  3.25  1.85  3.25  3.75  1.91  3.25  3.60  \n",
       "2  2.35  3.25  2.65  2.50  3.20  2.50  2.30  3.20  2.75  \n",
       "\n",
       "[3 rows x 115 columns]"
      ]
     },
     "metadata": {},
     "output_type": "display_data"
    },
    {
     "name": "stdout",
     "output_type": "stream",
     "text": [
      "\n",
      "Team (first 3 rows)\n"
     ]
    },
{
 "data": {
  "text/html": [
   "<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>id</th>\n",
   "      <th>team_api_id</th>\n",
   "      <th>team_fifa_api_id</th>\n",
   "      <th>team_long_name</th>\n",
   "      <th>team_short_name</th>\n",
   "    </tr>\n",
   "  </thead>\n",
   "  <tbody>\n",
   "    <tr>\n",
   "      <th>0</th>\n",
   "      <td>1</td>\n",
   "      <td>9987</td>\n",
   "      <td>673</td>\n",
   "      <td>KRC Genk</td>\n",
   "      <td>GEN</td>\n",
   "    </tr>\n",
   "    <tr>\n",
   "      <th>1</th>\n",
   "      <td>2</td>\n",
   "      <td>9993</td>\n",
   "      <td>675</td>\n",
   "      <td>Beerschot AC</td>\n",
   "      <td>BAC</td>\n",
   "    </tr>\n",
   "    <tr>\n",
   "      <th>2</th>\n",
   "      <td>3</td>\n",
   "      <td>10000</td>\n",
   "      <td>15005</td>\n",
   "      <td>SV Zulte-Waregem</td>\n",
   "      <td>ZUL</td>\n",
   "    </tr>\n",
   "  </tbody>\n",
   "</table>\n",
   "</div>"
  ],
  "text/plain": [
   "   id  team_api_id  team_fifa_api_id    team_long_name team_short_name\n",
   "0   1         9987               673          KRC Genk             GEN\n",
   "1   2         9993               675      Beerschot AC             BAC\n",
   "2   3        10000             15005  SV Zulte-Waregem             ZUL"
  ]
 },
 "metadata": {},
 "output_type": "display_data"
},
{
 "name": "stdout",
 "output_type": "stream",
 "text": [
  "\n",
  "Team_Attributes (first 3 rows)\n"
 ]
},
{
 "data": {
  "text/html": [
   "<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>id</th>\n",
   "      <th>team_fifa_api_id</th>\n",
   "      <th>team_api_id</th>\n",
   "      <th>date</th>\n",
   "      <th>buildUpPlaySpeed</th>\n",
   "      <th>buildUpPlaySpeedClass</th>\n",
   "      <th>buildUpPlayDribbling</th>\n",
   "      <th>buildUpPlayDribblingClass</th>\n",
   "      <th>buildUpPlayPassing</th>\n",
   "      <th>buildUpPlayPassingClass</th>\n",
   "      <th>...</th>\n",
   "      <th>chanceCreationShooting</th>\n",
   "      <th>chanceCreationShootingClass</th>\n",
   "      <th>chanceCreationPositioningClass</th>\n",
   "      <th>defencePressure</th>\n",
   "      <th>defencePressureClass</th>\n",
   "      <th>defenceAggression</th>\n",
   "      <th>defenceAggressionClass</th>\n",
   "      <th>defenceTeamWidth</th>\n",
   "      <th>defenceTeamWidthClass</th>\n",
   "      <th>defenceDefenderLineClass</th>\n",
   "    </tr>\n",
   "  </thead>\n",
   "  <tbody>\n",
   "    <tr>\n",
   "      <th>0</th>\n",
   "      <td>1</td>\n",
   "      <td>434</td>\n",
   "      <td>9930</td>\n",
   "      <td>2010-02-22 00:00:00</td>\n",
   "      <td>60</td>\n",
   "      <td>Balanced</td>\n",
   "      <td>NaN</td>\n",
   "      <td>Little</td>\n",
   "      <td>50</td>\n",
   "      <td>Mixed</td>\n",
   "      <td>...</td>\n",
   "      <td>55</td>\n",
       "      <td>Normal</td>\n",
       "      <td>Organised</td>\n",
       "      <td>50</td>\n",
       "      <td>Medium</td>\n",
       "      <td>55</td>\n",
       "      <td>Press</td>\n",
       "      <td>45</td>\n",
       "      <td>Normal</td>\n",
       "      <td>Cover</td>\n",
       "    </tr>\n",
       "    <tr>\n",
       "      <th>1</th>\n",
       "      <td>2</td>\n",
       "      <td>434</td>\n",
       "      <td>9930</td>\n",
       "      <td>2014-09-19 00:00:00</td>\n",
       "      <td>52</td>\n",
       "      <td>Balanced</td>\n",
       "      <td>48.0</td>\n",
       "      <td>Normal</td>\n",
       "      <td>56</td>\n",
       "      <td>Mixed</td>\n",
       "      <td>...</td>\n",
       "      <td>64</td>\n",
       "      <td>Normal</td>\n",
       "      <td>Organised</td>\n",
       "      <td>47</td>\n",
       "      <td>Medium</td>\n",
       "      <td>44</td>\n",
       "      <td>Press</td>\n",
       "      <td>54</td>\n",
       "      <td>Normal</td>\n",
       "      <td>Cover</td>\n",
       "    </tr>\n",
       "    <tr>\n",
       "      <th>2</th>\n",
       "      <td>3</td>\n",
       "      <td>434</td>\n",
       "      <td>9930</td>\n",
       "      <td>2015-09-10 00:00:00</td>\n",
       "      <td>47</td>\n",
       "      <td>Balanced</td>\n",
       "      <td>41.0</td>\n",
       "      <td>Normal</td>\n",
       "      <td>54</td>\n",
       "      <td>Mixed</td>\n",
       "      <td>...</td>\n",
       "      <td>64</td>\n",
       "      <td>Normal</td>\n",
       "      <td>Organised</td>\n",
       "      <td>47</td>\n",
       "      <td>Medium</td>\n",
       "      <td>44</td>\n",
       "      <td>Press</td>\n",
       "      <td>54</td>\n",
       "      <td>Normal</td>\n",
       "      <td>Cover</td>\n",
       "    </tr>\n",
       "  </tbody>\n",
       "</table>\n",
       "<p>3 rows × 25 columns</p>\n",
       "</div>"
      ],
      "text/plain": [
       "   id  team_fifa_api_id  team_api_id                 date  buildUpPlaySpeed  
\\\n",
       "0   1               434         9930  2010-02-22 00:00:00                60   
```

```
\n",
       "1   2               434         9930  2014-09-19 00:00:00                52
\n",
       "2   3               434         9930  2015-09-10 00:00:00                47
\n",
       "\n",
       "  buildUpPlaySpeedClass  buildUpPlayDribbling buildUpPlayDribblingClass  \\\n",
       "0              Balanced                   NaN                    Little   \n",
       "1              Balanced                  48.0                    Normal   \n",
       "2              Balanced                  41.0                    Normal   \n",
       "\n",
       "   buildUpPlayPassing buildUpPlayPassingClass  ... chanceCreationShooting  \\\n",
       "0                  50                   Mixed  ...                     55   \n",
       "1                  56                   Mixed  ...                     64   \n",
       "2                  54                   Mixed  ...                     64   \n",
       "\n",
       "   chanceCreationShootingClass chanceCreationPositioningClass  \\\n",
       "0                       Normal                      Organised   \n",
       "1                       Normal                      Organised   \n",
       "2                       Normal                      Organised   \n",
       "\n",
       "   defencePressure defencePressureClass  defenceAggression  \\\n",
       "0               50               Medium                 55   \n",
       "1               47               Medium                 44   \n",
       "2               47               Medium                 44   \n",
       "\n",
       "  defenceAggressionClass defenceTeamWidth  defenceTeamWidthClass  \\\n",
       "0                  Press               45                 Normal   \n",
       "1                  Press               54                 Normal   \n",
       "2                  Press               54                 Normal   \n",
       "\n",
       "  defenceDefenderLineClass  \n",
       "0                    Cover  \n",
       "1                    Cover  \n",
       "2                    Cover  \n",
       "\n",
       "[3 rows x 25 columns]"
      ]
     },
     "metadata": {},
     "output_type": "display_data"
    },
    {
     "name": "stdout",
     "output_type": "stream",
     "text": [
      "\n",
      "Player (first 3 rows)\n"
     ]
    },
    {
     "data": {
      "text/html": [
       "<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>id</th>\n",
       "      <th>player_api_id</th>\n",
       "      <th>player_name</th>\n",
       "      <th>player_fifa_api_id</th>\n",
       "      <th>birthday</th>\n",
       "      <th>height</th>\n",
       "      <th>weight</th>\n",
       "    </tr>\n",
       "  </thead>\n",
       "  <tbody>\n",
       "    <tr>\n",
       "      <th>0</th>\n",
       "      <td>1</td>\n",
       "      <td>505942</td>\n",
       "      <td>Aaron Appindangoye</td>\n",
       "      <td>218353</td>\n",
       "      <td>1992-02-29 00:00:00</td>\n",
       "      <td>182.88</td>\n",
       "      <td>187</td>\n",
       "    </tr>\n",
       "    <tr>\n",
       "      <th>1</th>\n",
       "      <td>2</td>\n",
       "      <td>155782</td>\n",
       "      <td>Aaron Cresswell</td>\n",
       "      <td>189615</td>\n",
       "      <td>1989-12-15 00:00:00</td>\n",
       "      <td>170.18</td>\n",
       "      <td>146</td>\n",
       "    </tr>\n",
       "    <tr>\n",
       "      <th>2</th>\n",
       "      <td>3</td>\n",
       "      <td>162549</td>\n",
       "      <td>Aaron Doran</td>\n",
       "      <td>186170</td>\n",
       "      <td>1991-05-13 00:00:00</td>\n",
       "      <td>170.18</td>\n",
       "      <td>163</td>\n",
       "    </tr>\n",
       "  </tbody>\n",
       "</table>\n",
       "</div>"
      ],
      "text/plain": [
       "   id  player_api_id         player_name  player_fifa_api_id  \\\n",
       "0   1         505942  Aaron Appindangoye              218353   \n",
       "1   2         155782     Aaron Cresswell              189615   \n",
       "2   3         162549         Aaron Doran              186170   \n",
       "\n",
       "              birthday  height  weight  \n",
       "0  1992-02-29 00:00:00  182.88     187  \n",
       "1  1989-12-15 00:00:00  170.18     146  \n",
       "2  1991-05-13 00:00:00  170.18     163  "
      ]
     },
     "metadata": {},
     "output_type": "display_data"
    },
    {
     "name": "stdout",
     "output_type": "stream",
     "text": [
      "\n",
      "Player_Attributes (first 3 rows)\n"
     ]
    },
    {
     "data": {
      "text/html": [
       "<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>id</th>\n",
       "      <th>player_fifa_api_id</th>\n",
       "      <th>player_api_id</th>\n",
       "      <th>date</th>\n",
       "      <th>overall_rating</th>\n",
       "      <th>potential</th>\n",
       "      <th>preferred_foot</th>\n",
       "      <th>attacking_work_rate</th>\n",
       "      <th>defensive_work_rate</th>\n",
       "      <th>crossing</th>\n",
       "      <th>...</th>\n",
       "      <th>vision</th>\n",
       "      <th>penalties</th>\n",
       "      <th>marking</th>\n",
       "      <th>standing_tackle</th>\n",
       "      <th>sliding_tackle</th>\n",
       "      <th>gk_diving</th>\n",
       "      <th>gk_handling</th>\n",
       "      <th>gk_kicking</th>\n",
       "      <th>gk_positioning</th>\n",
       "      <th>gk_reflexes</th>\n",
       "    </tr>\n",
       "  </thead>\n",
       "  <tbody>\n",
       "    <tr>\n",
       "      <th>0</th>\n",
       "      <td>1</td>\n",
       "      <td>218353</td>\n",
       "      <td>505942</td>\n",
       "      <td>2016-02-18 00:00:00</td>\n",
       "      <td>67</td>\n",
       "      <td>71</td>\n",
       "      <td>right</td>\n",
       "      <td>medium</td>\n",
       "      <td>medium</td>\n",
       "      <td>49</td>\n",
       "      <td>...</td>\n",
       "      <td>54</td>\n",
       "      <td>48</td>\n",
       "      <td>65</td>\n",
       "      <td>69</td>\n",
       "      <td>69</td>\n",
       "      <td>6</td>\n",
       "      <td>11</td>\n",
       "      <td>10</td>\n",
       "      <td>8</td>\n",
       "      <td>8</td>\n",
       "    </tr>\n",
       "    <tr>\n",
       "      <th>1</th>\n",
       "      <td>2</td>\n",
       "      <td>218353</td>\n",
       "      <td>505942</td>\n",
       "      <td>2015-11-19 00:00:00</td>\n",
       "      <td>67</td>\n",
       "      <td>71</td>\n",
       "      <td>right</td>\n",
       "      <td>medium</td>\n",
       "      <td>medium</td>\n",
       "      <td>49</td>\n",
       "      <td>...</td>\n",
       "      <td>54</td>\n",
       "      <td>48</td>\n",
       "      <td>65</td>\n",
       "      <td>69</td>\n",
       "      <td>69</td>\n",
       "      <td>6</td>\n",
       "      <td>11</td>\n",
       "      <td>10</td>\n",
       "      <td>8</td>\n",
       "      <td>8</td>\n",
       "    </tr>\n",
       "    <tr>\n",
       "      <th>2</th>\n",
       "      <td>3</td>\n",
       "      <td>218353</td>\n",
       "      <td>505942</td>\n",
       "      <td>2015-09-21 00:00:00</td>\n",
       "      <td>62</td>\n",
       "      <td>66</td>\n",
       "      <td>right</td>\n",
       "      <td>medium</td>\n",
       "      <td>medium</td>\n",
       "      <td>49</td>\n",
       "      <td>...</td>\n",
       "      <td>54</td>\n",
       "      <td>48</td>\n",
       "      <td>65</td>\n",
       "      <td>66</td>\n",
       "      <td>69</td>\n",
       "      <td>6</td>\n",
       "      <td>11</td>\n",
       "      <td>10</td>\n",
       "      <td>8</td>\n",
       "      <td>8</td>\n",
       "    </tr>\n",
       "  </tbody>\n",
       "</table>\n",
       "<p>3 rows × 42 columns</p>\n",
       "</div>"
      ],
      "text/plain": [
       "   id  player_fifa_api_id  player_api_id                 date  overall_rating  
\\\n",
       "0   1              218353         505942  2016-02-18 00:00:00              67   
\n",
       "1   2              218353         505942  2015-11-19 00:00:00              67   
\n",
       "2   3              218353         505942  2015-09-21 00:00:00              62   
```

```
\n",
       "\n",
       "   potential preferred_foot attacking_work_rate defensive_work_rate  crossing  
\\\n",
       "0         71          right              medium              medium        49   
\n",
       "1         71          right              medium              medium        49   
\n",
       "2         66          right              medium              medium        49   
\n",
       "\n",
       "   ...  vision  penalties  marking  standing_tackle  sliding_tackle  \\\n",
       "0  ...      54         48       65               69              69   \n",
       "1  ...      54         48       65               69              69   \n",
       "2  ...      54         48       65               66              69   \n",
       "\n",
       "   gk_diving  gk_handling  gk_kicking  gk_positioning  gk_reflexes  \n",
       "0          6           11          10               8            8  \n",
       "1          6           11          10               8            8  \n",
       "2          6           11          10               8            8  \n",
       "\n",
       "[3 rows x 42 columns]"
      ]
     },
     "metadata": {},
     "output_type": "display_data"
    }
   ],
   "source": [
    "for t in [\"League\", \"Match\", \"Team\", \"Team_Attributes\", \"Player\",
\"Player_Attributes\"]:\n",
    "    try:\n",
    "        print(f\"\\n{t} (first 3 rows)\")\n",
    "        display(sql(f\"select * from {t} limit 3\"))\n",
    "    except Exception as e:\n",
    "        print(f\"{t}: {e}\")\n"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": null,
   "id": "65b708fa",
   "metadata": {},
   "outputs": [],
   "source": []
  },
  {
   "cell_type": "code",
   "execution_count": null,
   "id": "21c67355",
   "metadata": {},
   "outputs": [],
   "source": []
  },
  {
   "cell_type": "markdown",
   "id": "0ffbb1d9",
   "metadata": {},
   "source": [
    "### 1. Filter for England and Spain leagues."
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 4,
   "id": "151b7be9",
   "metadata": {},
   "outputs": [
    {
     "data": {
      "text/html": [
       "<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>league_id</th>\n",
       "      <th>name</th>\n",
       "    </tr>\n",
       "  </thead>\n",
       "  <tbody>\n",
       "    <tr>\n",
       "      <th>0</th>\n",
       "      <td>1729</td>\n",
       "      <td>England Premier League</td>\n",
       "    </tr>\n",
       "    <tr>\n",
       "      <th>1</th>\n",
       "      <td>21518</td>\n",
       "      <td>Spain LIGA BBVA</td>\n",
       "    </tr>\n",
       "  </tbody>\n",
       "</table>\n",
       "</div>"
      ],
      "text/plain": [
       "   league_id                    name\n",
       "0       1729  England Premier League\n",
       "1      21518         Spain LIGA BBVA"
      ]
     },
     "metadata": {},
     "output_type": "display_data"
    }
   ],
   "source": [
    "leagues = sql(\"\"\"\n",
    "    select id as league_id, name\n",
    "    from league\n",
    "    where name in ('Spain LIGA BBVA', 'England Premier League')\n",
    "\"\"\")\n",
    "display(leagues)"
   ]
  },
  {
   "cell_type": "markdown",
   "id": "7b344835",
   "metadata": {},
   "source": [
    "### 2. List statistics for your favorite team from the Premier League and La Liga"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 5,
   "id": "ba8663ad",
   "metadata": {},
   "outputs": [],
   "source": [
    "matches = sql(\"\"\"\n",
    "    select\n",
    "        m.id,\n",
    "        m.season,\n",
    "        l.name as league_name,\n",
    "        t1.team_long_name as home_team,\n",
    "        t2.team_long_name as away_team,\n",
    "        m.home_team_goal,\n",
    "        m.away_team_goal\n",
    "    from match m\n",
    "    join league l on l.id = m.league_id\n",
    "    join team t1 on t1.team_api_id = m.home_team_api_id\n",
    "    join team t2 on t2.team_api_id = m.away_team_api_id\n",
    "    where l.name = 'Spain LIGA BBVA'\n",
    "\"\"\")\n"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 6,
   "id": "e9bdcf03",
   "metadata": {},
   "outputs": [],
   "source": [
    "home = matches[[\"league_name\", \"season\", \"home_team\", \"home_team_goal\",
\"away_team_goal\"]].rename(\n",
    "    columns={\"home_team\": \"team\", \"home_team_goal\": \"gf\",
\"away_team_goal\": \"ga\"}\n",
    ")\n",
    "away = matches[[\"league_name\", \"season\", \"away_team\", \"away_team_goal\",
\"home_team_goal\"]].rename(\n",
    "    columns={\"away_team\": \"team\", \"away_team_goal\": \"gf\",
\"home_team_goal\": \"ga\"}\n",
    ")"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 7,
   "id": "2bc769ba",
   "metadata": {},
   "outputs": [
    {
     "data": {
      "text/html": [
       "<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>League</th>\n",
"      <th>Season</th>\n",
"      <th>Team</th>\n",
"      <th>Games</th>\n",
"      <th>Wins</th>\n",
"      <th>Draws</th>\n",
"      <th>Losses</th>\n",
"      <th>GF</th>\n",
"      <th>GA</th>\n",
"      <th>Pts</th>\n",
"      <th>GD</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>145</th>\n",
"      <td>Spain LIGA BBVA</td>\n",
"      <td>2015/2016</td>\n",
"      <td>Levante UD</td>\n",
"      <td>38</td>\n",
"      <td>8</td>\n",
"      <td>8</td>\n",
"      <td>22</td>\n",
"      <td>37</td>\n",
"      <td>70</td>\n",
"      <td>32</td>\n",
"      <td>-33</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>146</th>\n",
"      <td>Spain LIGA BBVA</td>\n",
"      <td>2015/2016</td>\n",
"      <td>Málaga CF</td>\n",
"      <td>38</td>\n",
"      <td>12</td>\n",
"      <td>12</td>\n",
"      <td>14</td>\n",
"      <td>38</td>\n",
"      <td>35</td>\n",
"      <td>48</td>\n",
"      <td>3</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>147</th>\n",
"      <td>Spain LIGA BBVA</td>\n",
"      <td>2015/2016</td>\n",
"      <td>RC Celta de Vigo</td>\n",
"      <td>38</td>\n",
"      <td>17</td>\n",
"      <td>9</td>\n",
"      <td>12</td>\n",
"      <td>51</td>\n",
"      <td>59</td>\n",
"      <td>60</td>\n",
"      <td>-8</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>148</th>\n",
"      <td>Spain LIGA BBVA</td>\n",
"      <td>2015/2016</td>\n",
"      <td>RC Deportivo de La Coruña</td>\n",
"      <td>38</td>\n",
"      <td>8</td>\n",
    "      <td>18</td>\n",
    "      <td>12</td>\n",
    "      <td>45</td>\n",
    "      <td>61</td>\n",
    "      <td>42</td>\n",
    "      <td>-16</td>\n",
    "    </tr>\n",
    "    <tr>\n",
    "      <th>149</th>\n",
    "      <td>Spain LIGA BBVA</td>\n",
    "      <td>2015/2016</td>\n",
    "      <td>RCD Espanyol</td>\n",
    "      <td>38</td>\n",
    "      <td>12</td>\n",
    "      <td>7</td>\n",
    "      <td>19</td>\n",
    "      <td>40</td>\n",
    "      <td>74</td>\n",
    "      <td>43</td>\n",
    "      <td>-34</td>\n",
    "    </tr>\n",
    "    <tr>\n",
    "      <th>150</th>\n",
    "      <td>Spain LIGA BBVA</td>\n",
    "      <td>2015/2016</td>\n",
    "      <td>Rayo Vallecano</td>\n",
    "      <td>38</td>\n",
    "      <td>9</td>\n",
    "      <td>11</td>\n",
    "      <td>18</td>\n",
    "      <td>52</td>\n",
    "      <td>73</td>\n",
    "      <td>38</td>\n",
    "      <td>-21</td>\n",
    "    </tr>\n",
    "    <tr>\n",
    "      <th>151</th>\n",
    "      <td>Spain LIGA BBVA</td>\n",
    "      <td>2015/2016</td>\n",
    "      <td>Real Betis Balompié</td>\n",
    "      <td>38</td>\n",
    "      <td>11</td>\n",
    "      <td>12</td>\n",
    "      <td>15</td>\n",
    "      <td>34</td>\n",
    "      <td>52</td>\n",
    "      <td>45</td>\n",
    "      <td>-18</td>\n",
    "    </tr>\n",
    "    <tr>\n",
    "      <th>152</th>\n",
    "      <td>Spain LIGA BBVA</td>\n",
    "      <td>2015/2016</td>\n",
    "      <td>Real Madrid CF</td>\n",
    "      <td>38</td>\n",
    "      <td>28</td>\n",
    "      <td>6</td>\n",
    "      <td>4</td>\n",
    "      <td>110</td>\n",
    "      <td>34</td>\n",
    "      <td>90</td>\n",
    "      <td>76</td>\n",
    "    </tr>\n",
    "    <tr>\n",
    "      <th>153</th>\n",
    "      <td>Spain LIGA BBVA</td>\n",
    "      <td>2015/2016</td>\n",
"      <td>Real Sociedad</td>\n",
"      <td>38</td>\n",
"      <td>13</td>\n",
"      <td>9</td>\n",
"      <td>16</td>\n",
"      <td>45</td>\n",
"      <td>48</td>\n",
"      <td>48</td>\n",
"      <td>-3</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>154</th>\n",
"      <td>Spain LIGA BBVA</td>\n",
"      <td>2015/2016</td>\n",
"      <td>Real Sporting de Gijón</td>\n",
"      <td>38</td>\n",
"      <td>10</td>\n",
"      <td>9</td>\n",
"      <td>19</td>\n",
"      <td>40</td>\n",
"      <td>62</td>\n",
"      <td>39</td>\n",
"      <td>-22</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>155</th>\n",
"      <td>Spain LIGA BBVA</td>\n",
"      <td>2015/2016</td>\n",
"      <td>SD Eibar</td>\n",
"      <td>38</td>\n",
"      <td>11</td>\n",
"      <td>10</td>\n",
"      <td>17</td>\n",
"      <td>49</td>\n",
"      <td>61</td>\n",
"      <td>43</td>\n",
"      <td>-12</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>156</th>\n",
"      <td>Spain LIGA BBVA</td>\n",
"      <td>2015/2016</td>\n",
"      <td>Sevilla FC</td>\n",
"      <td>38</td>\n",
"      <td>14</td>\n",
"      <td>10</td>\n",
"      <td>14</td>\n",
"      <td>51</td>\n",
"      <td>50</td>\n",
"      <td>52</td>\n",
"      <td>1</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>157</th>\n",
"      <td>Spain LIGA BBVA</td>\n",
"      <td>2015/2016</td>\n",
"      <td>UD Las Palmas</td>\n",
"      <td>38</td>\n",
"      <td>12</td>\n",
"      <td>8</td>\n",
"      <td>18</td>\n",
"      <td>45</td>\n",
"      <td>53</td>\n",
"      <td>44</td>\n",
"      <td>-8</td>\n",
"    </tr>\n",
"    <tr>\n",
        "      <th>158</th>\n",
        "      <td>Spain LIGA BBVA</td>\n",
        "      <td>2015/2016</td>\n",
        "      <td>Valencia CF</td>\n",
        "      <td>38</td>\n",
        "      <td>11</td>\n",
        "      <td>11</td>\n",
        "      <td>16</td>\n",
        "      <td>46</td>\n",
        "      <td>48</td>\n",
        "      <td>44</td>\n",
        "      <td>-2</td>\n",
        "    </tr>\n",
        "    <tr>\n",
        "      <th>159</th>\n",
        "      <td>Spain LIGA BBVA</td>\n",
        "      <td>2015/2016</td>\n",
        "      <td>Villarreal CF</td>\n",
        "      <td>38</td>\n",
        "      <td>18</td>\n",
        "      <td>10</td>\n",
        "      <td>10</td>\n",
        "      <td>44</td>\n",
        "      <td>35</td>\n",
        "      <td>64</td>\n",
        "      <td>9</td>\n",
        "    </tr>\n",
        "  </tbody>\n",
        "</table>\n",
        "</div>"
       ],
       "text/plain": [
        "              League     Season                       Team  Games  Wins  \\\n",
        "145  Spain LIGA BBVA  2015/2016                 Levante UD     38     8   \n",
        "146  Spain LIGA BBVA  2015/2016                  Málaga CF     38    12   \n",
        "147  Spain LIGA BBVA  2015/2016           RC Celta de Vigo     38    17   \n",
        "148  Spain LIGA BBVA  2015/2016  RC Deportivo de La Coruña     38     8   \n",
        "149  Spain LIGA BBVA  2015/2016               RCD Espanyol     38    12   \n",
        "150  Spain LIGA BBVA  2015/2016             Rayo Vallecano     38     9   \n",
        "151  Spain LIGA BBVA  2015/2016        Real Betis Balompié     38    11   \n",
        "152  Spain LIGA BBVA  2015/2016             Real Madrid CF     38    28   \n",
        "153  Spain LIGA BBVA  2015/2016              Real Sociedad     38    13   \n",
        "154  Spain LIGA BBVA  2015/2016     Real Sporting de Gijón     38    10   \n",
        "155  Spain LIGA BBVA  2015/2016                   SD Eibar     38    11   \n",
        "156  Spain LIGA BBVA  2015/2016                 Sevilla FC     38    14   \n",
        "157  Spain LIGA BBVA  2015/2016              UD Las Palmas     38    12   \n",
        "158  Spain LIGA BBVA  2015/2016                Valencia CF     38    11   \n",
        "159  Spain LIGA BBVA  2015/2016              Villarreal CF     38    18   \n",
        "\n",
        "     Draws  Losses   GF  GA  Pts  GD  \n",
        "145      8      22   37  70   32 -33  \n",
        "146     12      14   38  35   48   3  \n",
        "147      9      12   51  59   60  -8  \n",
        "148     18      12   45  61   42 -16  \n",
        "149      7      19   40  74   43 -34  \n",
        "150     11      18   52  73   38 -21  \n",
        "151     12      15   34  52   45 -18  \n",
        "152      6       4  110  34   90  76  \n",
        "153      9      16   45  48   48  -3  \n",
        "154      9      19   40  62   39 -22  \n",
        "155     10      17   49  61   43 -12  \n",
        "156     10      14   51  50   52   1  \n",
        "157      8      18   45  53   44  -8  \n",
        "158     11      16   46  48   44  -2  \n",
        "159     10      10   44  35   64   9  "
       ]
      },
     "metadata": {},
     "output_type": "display_data"
    }
   ],
   "source": [
    "all_matches = pd.concat([home, away], ignore_index=True)\n",
    "\n",
    "def result(row):\n",
    "    if row[\"gf\"] > row[\"ga\"]:\n",
    "        return \"win\"\n",
    "    elif row[\"gf\"] < row[\"ga\"]:\n",
    "        return \"loss\"\n",
    "    else:\n",
    "        return \"draw\"\n",
    "\n",
    "all_matches[\"Result\"] = all_matches.apply(result, axis=1)\n",
    "all_matches[\"Pts\"] = all_matches[\"Result\"].map({\"win\": 3, \"draw\": 1,
\"loss\": 0})\n",
    "\n",
    "team_stats = (\n",
    "    all_matches.groupby([\"league_name\", \"season\", \"team\"], as_index=False)\n",
    "    .agg(\n",
    "        Games=(\"Result\", \"count\"),\n",
    "        Wins=(\"Result\", lambda s: (s == \"win\").sum()),\n",
    "        Draws=(\"Result\", lambda s: (s == \"draw\").sum()),\n",
    "        Losses=(\"Result\", lambda s: (s == \"loss\").sum()),\n",
    "        GF=(\"gf\", \"sum\"),\n",
    "        GA=(\"ga\", \"sum\"),\n",
    "        Pts=(\"Pts\", \"sum\"),\n",
    "    )\n",
    ")\n",
    "\n",
    "team_stats[\"GD\"] = team_stats[\"GF\"] - team_stats[\"GA\"]\n",
    "\n",
    "team_stats = team_stats.rename(columns={\n",
    "    \"league_name\": \"League\",\n",
    "    \"season\": \"Season\",\n",
    "    \"team\": \"Team\"\n",
    "})\n",
    "\n",
    "display(team_stats.tail(15))"
   ]
  },
  {
   "cell_type": "markdown",
   "id": "e0444866",
   "metadata": {},
   "source": [
    "### 3. Select team attributes."
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 8,
   "id": "25418cdc",
   "metadata": {},
   "outputs": [
    {
     "data": {
      "text/html": [
       "<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>id</th>\n",
        "      <th>team_fifa_api_id</th>\n",
        "      <th>team_api_id</th>\n",
        "      <th>date</th>\n",
        "      <th>buildUpPlaySpeed</th>\n",
        "      <th>buildUpPlaySpeedClass</th>\n",
        "      <th>buildUpPlayDribbling</th>\n",
        "      <th>buildUpPlayDribblingClass</th>\n",
        "      <th>buildUpPlayPassing</th>\n",
        "      <th>buildUpPlayPassingClass</th>\n",
        "      <th>...</th>\n",
        "      <th>chanceCreationShooting</th>\n",
        "      <th>chanceCreationShootingClass</th>\n",
        "      <th>chanceCreationPositioningClass</th>\n",
        "      <th>defencePressure</th>\n",
        "      <th>defencePressureClass</th>\n",
        "      <th>defenceAggression</th>\n",
        "      <th>defenceAggressionClass</th>\n",
        "      <th>defenceTeamWidth</th>\n",
        "      <th>defenceTeamWidthClass</th>\n",
        "      <th>defenceDefenderLineClass</th>\n",
        "    </tr>\n",
        "  </thead>\n",
        "  <tbody>\n",
        "    <tr>\n",
        "      <th>0</th>\n",
        "      <td>40</td>\n",
        "      <td>1861</td>\n",
        "      <td>9865</td>\n",
        "      <td>2010-02-22 00:00:00</td>\n",
        "      <td>65</td>\n",
        "      <td>Balanced</td>\n",
        "      <td>NaN</td>\n",
        "      <td>Little</td>\n",
        "      <td>45</td>\n",
        "      <td>Mixed</td>\n",
        "      <td>...</td>\n",
        "      <td>70</td>\n",
        "      <td>Lots</td>\n",
        "      <td>Organised</td>\n",
        "      <td>70</td>\n",
        "      <td>High</td>\n",
        "      <td>65</td>\n",
        "      <td>Press</td>\n",
        "      <td>30</td>\n",
        "      <td>Narrow</td>\n",
        "      <td>Cover</td>\n",
        "    </tr>\n",
        "    <tr>\n",
        "      <th>1</th>\n",
        "      <td>41</td>\n",
        "      <td>1861</td>\n",
        "      <td>9865</td>\n",
        "      <td>2011-02-22 00:00:00</td>\n",
        "      <td>55</td>\n",
        "      <td>Balanced</td>\n",
"      <td>NaN</td>\n",
"      <td>Little</td>\n",
"      <td>53</td>\n",
"      <td>Mixed</td>\n",
"      <td>...</td>\n",
"      <td>69</td>\n",
"      <td>Lots</td>\n",
"      <td>Organised</td>\n",
"      <td>58</td>\n",
"      <td>Medium</td>\n",
"      <td>68</td>\n",
"      <td>Double</td>\n",
"      <td>67</td>\n",
"      <td>Wide</td>\n",
"      <td>Cover</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>2</th>\n",
"      <td>42</td>\n",
"      <td>1861</td>\n",
"      <td>9865</td>\n",
"      <td>2012-02-22 00:00:00</td>\n",
"      <td>42</td>\n",
"      <td>Balanced</td>\n",
"      <td>NaN</td>\n",
"      <td>Little</td>\n",
"      <td>70</td>\n",
"      <td>Long</td>\n",
"      <td>...</td>\n",
"      <td>40</td>\n",
"      <td>Normal</td>\n",
"      <td>Organised</td>\n",
"      <td>37</td>\n",
"      <td>Medium</td>\n",
"      <td>53</td>\n",
"      <td>Press</td>\n",
"      <td>45</td>\n",
"      <td>Normal</td>\n",
"      <td>Cover</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>3</th>\n",
"      <td>43</td>\n",
"      <td>1861</td>\n",
"      <td>9865</td>\n",
"      <td>2013-09-20 00:00:00</td>\n",
"      <td>46</td>\n",
"      <td>Balanced</td>\n",
"      <td>NaN</td>\n",
"      <td>Little</td>\n",
"      <td>65</td>\n",
"      <td>Mixed</td>\n",
"      <td>...</td>\n",
"      <td>40</td>\n",
"      <td>Normal</td>\n",
"      <td>Organised</td>\n",
"      <td>37</td>\n",
"      <td>Medium</td>\n",
"      <td>53</td>\n",
"      <td>Press</td>\n",
"      <td>45</td>\n",
"      <td>Normal</td>\n",
"      <td>Cover</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>4</th>\n",
"      <td>44</td>\n",
     "      <td>1861</td>\n",
     "      <td>9865</td>\n",
     "      <td>2014-09-19 00:00:00</td>\n",
     "      <td>46</td>\n",
     "      <td>Balanced</td>\n",
     "      <td>41.0</td>\n",
     "      <td>Normal</td>\n",
     "      <td>65</td>\n",
     "      <td>Mixed</td>\n",
     "      <td>...</td>\n",
     "      <td>40</td>\n",
     "      <td>Normal</td>\n",
     "      <td>Organised</td>\n",
     "      <td>37</td>\n",
     "      <td>Medium</td>\n",
     "      <td>53</td>\n",
     "      <td>Press</td>\n",
     "      <td>45</td>\n",
     "      <td>Normal</td>\n",
     "      <td>Cover</td>\n",
     "    </tr>\n",
     "    <tr>\n",
     "      <th>5</th>\n",
     "      <td>45</td>\n",
     "      <td>1861</td>\n",
     "      <td>9865</td>\n",
     "      <td>2015-09-10 00:00:00</td>\n",
     "      <td>46</td>\n",
     "      <td>Balanced</td>\n",
     "      <td>41.0</td>\n",
     "      <td>Normal</td>\n",
     "      <td>65</td>\n",
     "      <td>Mixed</td>\n",
     "      <td>...</td>\n",
     "      <td>40</td>\n",
     "      <td>Normal</td>\n",
     "      <td>Organised</td>\n",
     "      <td>37</td>\n",
     "      <td>Medium</td>\n",
     "      <td>53</td>\n",
     "      <td>Press</td>\n",
     "      <td>45</td>\n",
     "      <td>Normal</td>\n",
     "      <td>Cover</td>\n",
     "    </tr>\n",
     "    <tr>\n",
     "      <th>6</th>\n",
     "      <td>89</td>\n",
     "      <td>448</td>\n",
     "      <td>8315</td>\n",
     "      <td>2010-02-22 00:00:00</td>\n",
     "      <td>60</td>\n",
     "      <td>Balanced</td>\n",
     "      <td>NaN</td>\n",
     "      <td>Little</td>\n",
     "      <td>45</td>\n",
     "      <td>Mixed</td>\n",
     "      <td>...</td>\n",
     "      <td>55</td>\n",
     "      <td>Normal</td>\n",
     "      <td>Organised</td>\n",
     "      <td>30</td>\n",
     "      <td>Deep</td>\n",
     "      <td>70</td>\n",
     "      <td>Double</td>\n",
     "      <td>35</td>\n",
     "      <td>Normal</td>\n",
"      <td>Cover</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>7</th>\n",
"      <td>90</td>\n",
"      <td>448</td>\n",
"      <td>8315</td>\n",
"      <td>2011-02-22 00:00:00</td>\n",
"      <td>70</td>\n",
"      <td>Fast</td>\n",
"      <td>NaN</td>\n",
"      <td>Little</td>\n",
"      <td>69</td>\n",
"      <td>Long</td>\n",
"      <td>...</td>\n",
"      <td>57</td>\n",
"      <td>Normal</td>\n",
"      <td>Organised</td>\n",
"      <td>48</td>\n",
"      <td>Medium</td>\n",
"      <td>67</td>\n",
"      <td>Double</td>\n",
"      <td>55</td>\n",
"      <td>Normal</td>\n",
"      <td>Cover</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>8</th>\n",
"      <td>91</td>\n",
"      <td>448</td>\n",
"      <td>8315</td>\n",
"      <td>2012-02-22 00:00:00</td>\n",
"      <td>42</td>\n",
"      <td>Balanced</td>\n",
"      <td>NaN</td>\n",
"      <td>Little</td>\n",
"      <td>59</td>\n",
"      <td>Mixed</td>\n",
"      <td>...</td>\n",
"      <td>59</td>\n",
"      <td>Normal</td>\n",
"      <td>Organised</td>\n",
"      <td>37</td>\n",
"      <td>Medium</td>\n",
"      <td>60</td>\n",
"      <td>Press</td>\n",
"      <td>68</td>\n",
"      <td>Wide</td>\n",
"      <td>Cover</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>9</th>\n",
"      <td>92</td>\n",
"      <td>448</td>\n",
"      <td>8315</td>\n",
"      <td>2013-09-20 00:00:00</td>\n",
"      <td>46</td>\n",
"      <td>Balanced</td>\n",
"      <td>NaN</td>\n",
"      <td>Little</td>\n",
"      <td>38</td>\n",
"      <td>Mixed</td>\n",
"      <td>...</td>\n",
"      <td>37</td>\n",
"      <td>Normal</td>\n",
"      <td>Free Form</td>\n",
"      <td>37</td>\n",
       "      <td>Medium</td>\n",
       "      <td>60</td>\n",
       "      <td>Press</td>\n",
       "      <td>68</td>\n",
       "      <td>Wide</td>\n",
       "      <td>Cover</td>\n",
       "    </tr>\n",
       "  </tbody>\n",
       "</table>\n",
       "<p>10 rows × 25 columns</p>\n",
       "</div>"
      ],
      "text/plain": [
       "   id  team_fifa_api_id  team_api_id                 date  buildUpPlaySpeed
\\\n",
       "0  40              1861         9865  2010-02-22 00:00:00                65
\n",
       "1  41              1861         9865  2011-02-22 00:00:00                55
\n",
       "2  42              1861         9865  2012-02-22 00:00:00                42
\n",
       "3  43              1861         9865  2013-09-20 00:00:00                46
\n",
       "4  44              1861         9865  2014-09-19 00:00:00                46
\n",
       "5  45              1861         9865  2015-09-10 00:00:00                46
\n",
       "6  89               448         8315  2010-02-22 00:00:00                60
\n",
       "7  90               448         8315  2011-02-22 00:00:00                70
\n",
       "8  91               448         8315  2012-02-22 00:00:00                42
\n",
       "9  92               448         8315  2013-09-20 00:00:00                46
\n",
       "\n",
       "  buildUpPlaySpeedClass  buildUpPlayDribbling buildUpPlayDribblingClass  \\\n",
       "0              Balanced                   NaN                    Little   \n",
       "1              Balanced                   NaN                    Little   \n",
       "2              Balanced                   NaN                    Little   \n",
       "3              Balanced                   NaN                    Little   \n",
       "4              Balanced                  41.0                    Normal   \n",
       "5              Balanced                  41.0                    Normal   \n",
       "6              Balanced                   NaN                    Little   \n",
       "7                  Fast                   NaN                    Little   \n",
       "8              Balanced                   NaN                    Little   \n",
       "9              Balanced                   NaN                    Little   \n",
       "\n",
       "   buildUpPlayPassing buildUpPlayPassingClass  ... chanceCreationShooting  \\\n",
       "0                  45                   Mixed  ...                     70   \n",
       "1                  53                   Mixed  ...                     69   \n",
       "2                  70                    Long  ...                     40   \n",
       "3                  65                   Mixed  ...                     40   \n",
       "4                  65                   Mixed  ...                     40   \n",
       "5                  65                   Mixed  ...                     40   \n",
       "6                  45                   Mixed  ...                     55   \n",
       "7                  69                    Long  ...                     57   \n",
       "8                  59                   Mixed  ...                     59   \n",
       "9                  38                   Mixed  ...                     37   \n",
       "\n",
       "   chanceCreationShootingClass chanceCreationPositioningClass  \\\n",
       "0                         Lots                      Organised   \n",
       "1                         Lots                      Organised   \n",
       "2                       Normal                      Organised   \n",
       "3                       Normal                      Organised   \n",
       "4                       Normal                      Organised   \n",
       "5                       Normal                      Organised   \n",
       "6                       Normal                      Organised   \n",
       "7                       Normal                      Organised   \n",
       "8                       Normal                      Organised   \n",
       "9                       Normal                      Free Form   \n",
       "\n",
       "   defencePressure defencePressureClass  defenceAggression  \\\n",
       "0               70                 High                 65   \n",
       "1               58               Medium                 68   \n",
       "2               37               Medium                 53   \n",
       "3               37               Medium                 53   \n",
       "4               37               Medium                 53   \n",
       "5               37               Medium                 53   \n",
       "6               30                 Deep                 70   \n",
       "7               48               Medium                 67   \n",
       "8               37               Medium                 60   \n",
       "9               37               Medium                 60   \n",
       "\n",
       "  defenceAggressionClass defenceTeamWidth  defenceTeamWidthClass  \\\n",
       "0                  Press               30                 Narrow   \n",
       "1                 Double               67                   Wide   \n",
       "2                  Press               45                 Normal   \n",
       "3                  Press               45                 Normal   \n",
       "4                  Press               45                 Normal   \n",
       "5                  Press               45                 Normal   \n",
       "6                 Double               35                 Normal   \n",
       "7                 Double               55                 Normal   \n",
       "8                  Press               68                   Wide   \n",
       "9                  Press               68                   Wide   \n",
       "\n",
       "  defenceDefenderLineClass  \n",
       "0                    Cover  \n",
       "1                    Cover  \n",
       "2                    Cover  \n",
       "3                    Cover  \n",
       "4                    Cover  \n",
       "5                    Cover  \n",
       "6                    Cover  \n",
       "7                    Cover  \n",
       "8                    Cover  \n",
       "9                    Cover  \n",
       "\n",
       "[10 rows x 25 columns]"
      ]
     },
     "metadata": {},
     "output_type": "display_data"
    }
   ],
   "source": [
    "team_attributes_spain = sql(\"\"\"\n",
    "    select *\n",
    "    from team_attributes ta\n",
    "    where ta.team_api_id in (\n",
    "        select distinct t.team_api_id\n",
    "        from team t\n",
    "        join match m on t.team_api_id in (m.home_team_api_id,
m.away_team_api_id)\n",
    "        join league l on l.id = m.league_id\n",
    "        where l.name = 'Spain LIGA BBVA'\n",
    "    )\n",
    "\"\"\")\n",
    "\n",
    "display(team_attributes_spain.head(10))"
   ]
  },
  {
   "cell_type": "markdown",
   "id": "26dcd994",
   "metadata": {},
   "source": [
    "### 4. Select player attributes."
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 9,
   "id": "2ba50605",
   "metadata": {},
   "outputs": [
    {
     "data": {
      "text/html": [
       "<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>id</th>\n",
       "      <th>player_fifa_api_id</th>\n",
       "      <th>player_api_id</th>\n",
       "      <th>date</th>\n",
       "      <th>overall_rating</th>\n",
       "      <th>potential</th>\n",
       "      <th>preferred_foot</th>\n",
       "      <th>attacking_work_rate</th>\n",
       "      <th>defensive_work_rate</th>\n",
       "      <th>crossing</th>\n",
       "      <th>...</th>\n",
       "      <th>vision</th>\n",
       "      <th>penalties</th>\n",
       "      <th>marking</th>\n",
       "      <th>standing_tackle</th>\n",
       "      <th>sliding_tackle</th>\n",
       "      <th>gk_diving</th>\n",
       "      <th>gk_handling</th>\n",
       "      <th>gk_kicking</th>\n",
       "      <th>gk_positioning</th>\n",
       "      <th>gk_reflexes</th>\n",
       "    </tr>\n",
       "  </thead>\n",
       "  <tbody>\n",
       "    <tr>\n",
       "      <th>0</th>\n",
       "      <td>243</td>\n",
       "      <td>183853</td>\n",
       "      <td>40719</td>\n",
       "      <td>2016-04-28 00:00:00</td>\n",
       "      <td>70</td>\n",
       "      <td>71</td>\n",
       "      <td>left</td>\n",
       "      <td>medium</td>\n",
"      <td>medium</td>\n",
"      <td>64</td>\n",
"      <td>...</td>\n",
"      <td>62</td>\n",
"      <td>67</td>\n",
"      <td>36</td>\n",
"      <td>31</td>\n",
"      <td>37</td>\n",
"      <td>14</td>\n",
"      <td>13</td>\n",
"      <td>9</td>\n",
"      <td>15</td>\n",
"      <td>10</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>1</th>\n",
"      <td>244</td>\n",
"      <td>183853</td>\n",
"      <td>40719</td>\n",
"      <td>2016-04-14 00:00:00</td>\n",
"      <td>70</td>\n",
"      <td>72</td>\n",
"      <td>left</td>\n",
"      <td>medium</td>\n",
"      <td>medium</td>\n",
"      <td>64</td>\n",
"      <td>...</td>\n",
"      <td>62</td>\n",
"      <td>67</td>\n",
"      <td>36</td>\n",
"      <td>31</td>\n",
"      <td>37</td>\n",
"      <td>14</td>\n",
"      <td>13</td>\n",
"      <td>9</td>\n",
"      <td>15</td>\n",
"      <td>10</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>2</th>\n",
"      <td>245</td>\n",
"      <td>183853</td>\n",
"      <td>40719</td>\n",
"      <td>2016-01-21 00:00:00</td>\n",
"      <td>70</td>\n",
"      <td>72</td>\n",
"      <td>left</td>\n",
"      <td>medium</td>\n",
"      <td>medium</td>\n",
"      <td>64</td>\n",
"      <td>...</td>\n",
"      <td>62</td>\n",
"      <td>67</td>\n",
"      <td>36</td>\n",
"      <td>31</td>\n",
"      <td>37</td>\n",
"      <td>14</td>\n",
"      <td>13</td>\n",
"      <td>9</td>\n",
"      <td>15</td>\n",
"      <td>10</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>3</th>\n",
"      <td>246</td>\n",
"      <td>183853</td>\n",
"      <td>40719</td>\n",
"      <td>2015-12-17 00:00:00</td>\n",
"      <td>71</td>\n",
"      <td>73</td>\n",
"      <td>left</td>\n",
"      <td>medium</td>\n",
"      <td>medium</td>\n",
"      <td>64</td>\n",
"      <td>...</td>\n",
"      <td>62</td>\n",
"      <td>67</td>\n",
"      <td>36</td>\n",
"      <td>31</td>\n",
"      <td>37</td>\n",
"      <td>14</td>\n",
"      <td>13</td>\n",
"      <td>9</td>\n",
"      <td>15</td>\n",
"      <td>10</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>4</th>\n",
"      <td>247</td>\n",
"      <td>183853</td>\n",
"      <td>40719</td>\n",
"      <td>2015-11-12 00:00:00</td>\n",
"      <td>71</td>\n",
"      <td>73</td>\n",
"      <td>left</td>\n",
"      <td>medium</td>\n",
"      <td>medium</td>\n",
"      <td>64</td>\n",
"      <td>...</td>\n",
"      <td>62</td>\n",
"      <td>67</td>\n",
"      <td>36</td>\n",
"      <td>31</td>\n",
"      <td>37</td>\n",
"      <td>14</td>\n",
"      <td>13</td>\n",
"      <td>9</td>\n",
"      <td>15</td>\n",
"      <td>10</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>5</th>\n",
"      <td>248</td>\n",
"      <td>183853</td>\n",
"      <td>40719</td>\n",
"      <td>2015-11-06 00:00:00</td>\n",
"      <td>71</td>\n",
"      <td>73</td>\n",
"      <td>left</td>\n",
"      <td>medium</td>\n",
"      <td>medium</td>\n",
"      <td>64</td>\n",
"      <td>...</td>\n",
"      <td>62</td>\n",
"      <td>67</td>\n",
"      <td>36</td>\n",
"      <td>31</td>\n",
"      <td>37</td>\n",
"      <td>14</td>\n",
"      <td>13</td>\n",
"      <td>9</td>\n",
"      <td>15</td>\n",
"      <td>10</td>\n",
"    </tr>\n",
    "    <tr>\n",
    "      <th>6</th>\n",
    "      <td>249</td>\n",
    "      <td>183853</td>\n",
    "      <td>40719</td>\n",
    "      <td>2015-10-02 00:00:00</td>\n",
    "      <td>71</td>\n",
    "      <td>74</td>\n",
    "      <td>left</td>\n",
    "      <td>medium</td>\n",
    "      <td>medium</td>\n",
    "      <td>64</td>\n",
    "      <td>...</td>\n",
    "      <td>62</td>\n",
    "      <td>67</td>\n",
    "      <td>36</td>\n",
    "      <td>31</td>\n",
    "      <td>37</td>\n",
    "      <td>14</td>\n",
    "      <td>13</td>\n",
    "      <td>9</td>\n",
    "      <td>15</td>\n",
    "      <td>10</td>\n",
    "    </tr>\n",
    "    <tr>\n",
    "      <th>7</th>\n",
    "      <td>250</td>\n",
    "      <td>183853</td>\n",
    "      <td>40719</td>\n",
    "      <td>2015-09-25 00:00:00</td>\n",
    "      <td>71</td>\n",
    "      <td>74</td>\n",
    "      <td>left</td>\n",
    "      <td>medium</td>\n",
    "      <td>medium</td>\n",
    "      <td>64</td>\n",
    "      <td>...</td>\n",
    "      <td>62</td>\n",
    "      <td>67</td>\n",
    "      <td>36</td>\n",
    "      <td>31</td>\n",
    "      <td>37</td>\n",
    "      <td>14</td>\n",
    "      <td>13</td>\n",
    "      <td>9</td>\n",
    "      <td>15</td>\n",
    "      <td>10</td>\n",
    "    </tr>\n",
    "    <tr>\n",
    "      <th>8</th>\n",
    "      <td>251</td>\n",
    "      <td>183853</td>\n",
    "      <td>40719</td>\n",
    "      <td>2015-09-21 00:00:00</td>\n",
    "      <td>70</td>\n",
    "      <td>73</td>\n",
    "      <td>left</td>\n",
    "      <td>medium</td>\n",
    "      <td>medium</td>\n",
    "      <td>64</td>\n",
    "      <td>...</td>\n",
    "      <td>62</td>\n",
    "      <td>67</td>\n",
    "      <td>36</td>\n",
    "      <td>31</td>\n",
    "      <td>37</td>\n",
    "      <td>14</td>\n",
    "      <td>13</td>\n",
    "      <td>9</td>\n",
    "      <td>15</td>\n",
    "      <td>10</td>\n",
    "      <td>14</td>\n",
         "      <td>13</td>\n",
         "      <td>9</td>\n",
         "      <td>15</td>\n",
         "      <td>10</td>\n",
         "    </tr>\n",
         "    <tr>\n",
         "      <th>9</th>\n",
         "      <td>252</td>\n",
         "      <td>183853</td>\n",
         "      <td>40719</td>\n",
         "      <td>2015-03-06 00:00:00</td>\n",
         "      <td>70</td>\n",
         "      <td>75</td>\n",
         "      <td>left</td>\n",
         "      <td>medium</td>\n",
         "      <td>medium</td>\n",
         "      <td>65</td>\n",
         "      <td>...</td>\n",
         "      <td>64</td>\n",
         "      <td>66</td>\n",
         "      <td>35</td>\n",
         "      <td>30</td>\n",
         "      <td>36</td>\n",
         "      <td>13</td>\n",
         "      <td>12</td>\n",
         "      <td>8</td>\n",
         "      <td>14</td>\n",
         "      <td>9</td>\n",
         "    </tr>\n",
         "  </tbody>\n",
         "</table>\n",
         "<p>10 rows × 42 columns</p>\n",
         "</div>"
        ],
        "text/plain": [
         "    id  player_fifa_api_id  player_api_id                 date  \\\n",
         "0  243              183853          40719  2016-04-28 00:00:00   \n",
         "1  244              183853          40719  2016-04-14 00:00:00   \n",
         "2  245              183853          40719  2016-01-21 00:00:00   \n",
         "3  246              183853          40719  2015-12-17 00:00:00   \n",
         "4  247              183853          40719  2015-11-12 00:00:00   \n",
         "5  248              183853          40719  2015-11-06 00:00:00   \n",
         "6  249              183853          40719  2015-10-02 00:00:00   \n",
         "7  250              183853          40719  2015-09-25 00:00:00   \n",
         "8  251              183853          40719  2015-09-21 00:00:00   \n",
         "9  252              183853          40719  2015-03-06 00:00:00   \n",
         "\n",
         "   overall_rating  potential preferred_foot attacking_work_rate  \\\n",
         "0              70         71           left              medium   \n",
         "1              70         72           left              medium   \n",
         "2              70         72           left              medium   \n",
         "3              71         73           left              medium   \n",
         "4              71         73           left              medium   \n",
         "5              71         73           left              medium   \n",
         "6              71         74           left              medium   \n",
         "7              71         74           left              medium   \n",
         "8              70         73           left              medium   \n",
         "9              70         75           left              medium   \n",
         "\n",
         "  defensive_work_rate  crossing  ...  vision  penalties  marking  \\\n",
         "0              medium        64  ...      62         67       36   \n",
         "1              medium        64  ...      62         67       36   \n",
         "2              medium        64  ...      62         67       36   \n",
         "3              medium        64  ...      62         67       36   \n",
         "4              medium        64  ...      62         67       36   \n",
         "5              medium        64  ...      62         67       36   \n",
         "6              medium        64  ...      62         67       36   \n",
       "7              medium        64  ...      62         67       36   \n",
       "8              medium        64  ...      62         67       36   \n",
       "9              medium        65  ...      64         66       35   \n",
       "\n",
       "   standing_tackle  sliding_tackle  gk_diving  gk_handling  gk_kicking  \\\n",
       "0               31              37         14           13           9   \n",
       "1               31              37         14           13           9   \n",
       "2               31              37         14           13           9   \n",
       "3               31              37         14           13           9   \n",
       "4               31              37         14           13           9   \n",
       "5               31              37         14           13           9   \n",
       "6               31              37         14           13           9   \n",
       "7               31              37         14           13           9   \n",
       "8               31              37         14           13           9   \n",
       "9               30              36         13           12           8   \n",
       "\n",
       "   gk_positioning  gk_reflexes  \n",
       "0              15           10  \n",
       "1              15           10  \n",
       "2              15           10  \n",
       "3              15           10  \n",
       "4              15           10  \n",
       "5              15           10  \n",
       "6              15           10  \n",
       "7              15           10  \n",
       "8              15           10  \n",
       "9              14            9  \n",
       "\n",
       "[10 rows x 42 columns]"
      ]
     },
     "metadata": {},
     "output_type": "display_data"
    }
   ],
   "source": [
    "matches = sql(\"\"\"\n",
    "    select *\n",
    "    from match m\n",
    "    join league l on l.id = m.league_id\n",
    "    where l.name = 'Spain LIGA BBVA'\n",
    "\"\"\")\n",
    "\n",
    "player_cols = [col for col in matches.columns if 'home_player_' in col or
'away_player_' in col]\n",
    "player_ids = pd.unique(matches[player_cols].values.ravel())\n",
    "player_ids = [pid for pid in player_ids if pd.notnull(pid)]\n",
    "\n",
    "ids_str = ','.join(str(int(pid)) for pid in player_ids)\n",
    "query = f\"select * from player_attributes where player_api_id in ({ids_str}) limit
100\"\n",
    "player_attributes_spain = sql(query)\n",
    "\n",
    "display(player_attributes_spain.head(10))"
   ]
  },
  {
   "cell_type": "markdown",
   "id": "859e219c",
   "metadata": {},
   "source": [
    "### 5. Implement your own gradient descent code for Logistic Regression and SVM."
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 20,
   "id": "aec7eec9",
   "metadata": {},
   "outputs": [],
   "source": [
    "df = sql(\"select * from player_attributes where attacking_work_rate in
('high','medium') limit 1000\")\n",
    "df = df.dropna()\n",
    "\n",
    "features = ['overall_rating', 'potential', 'shot_power', 'stamina', 'strength']\n",
    "x = df[features].values.astype(float)\n",
    "y = (df['attacking_work_rate'] == 'high').astype(int).values  # 1=high, 0=medium\n",
    "\n",
    "x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3,
random_state=42, stratify=y)"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 22,
   "id": "04b3f7ac",
   "metadata": {},
   "outputs": [],
   "source": [
    "mu = x_train.mean(axis=0)\n",
    "sd = x_train.std(axis=0) + 1e-8\n",
    "x_train = (x_train - mu) / sd\n",
    "x_test  = (x_test  - mu) / sd\n"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 21,
   "id": "d33c2945",
   "metadata": {},
   "outputs": [],
   "source": [
    "def add_bias(a):\n",
    "    return np.c_[np.ones(a.shape[0]), a]\n",
    "\n",
    "def sigmoid(z):\n",
    "    z = np.clip(z, -500, 500)\n",
    "    return 1.0 / (1.0 + np.exp(-z))"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 23,
   "id": "23f17f56",
   "metadata": {},
   "outputs": [],
   "source": [
    "def logreg_gd(x, y, lr=0.1, n_iter=500, lam=1e-3):\n",
    "    x1 = add_bias(x)\n",
    "    w = np.zeros(x1.shape[1])\n",
    "    hist = []\n",
    "    n = x1.shape[0]\n",
    "    for _ in range(n_iter):\n",
    "        p = sigmoid(x1 @ w)\n",
    "        eps = 1e-12\n",
    "        loss = (-(y*np.log(p+eps) + (1-y)*np.log(1-p+eps)).mean()\n",
    "                + 0.5*lam*np.sum(w[1:]**2)/n)\n",
    "        hist.append(loss)\n",
    "        grad = (x1.T @ (p - y)) / n\n",
    "        grad[1:] += (lam * w[1:]) / n\n",
    "        w -= lr * grad\n",
    "    return w, np.array(hist)\n",
    "\n",
    "def logreg_predict(x, w):\n",
    "    return (sigmoid(add_bias(x) @ w) >= 0.5).astype(int)\n",
    "\n",
    "def svm_gd(x, y, lr=0.01, n_iter=500, lam=1e-3):\n",
    "    y2 = np.where(y > 0, 1.0, -1.0)\n",
    "    x1 = add_bias(x)\n",
    "    w = np.zeros(x1.shape[1])\n",
    "    hist = []\n",
    "    n = x1.shape[0]\n",
    "    for _ in range(n_iter):\n",
    "        m = y2 * (x1 @ w)\n",
    "        viol = m < 1.0\n",
    "        hinge = np.maximum(0.0, 1.0 - m).mean()\n",
    "        loss = hinge + 0.5*lam*np.sum(w[1:]**2)/n\n",
    "        hist.append(loss)\n",
    "        g = -(x1[viol].T @ y2[viol]) / n\n",
    "        g[1:] += (lam * w[1:]) / n\n",
    "        w -= lr * g\n",
    "    return w, np.array(hist)\n",
    "\n",
    "def svm_predict(x, w):\n",
    "    return (add_bias(x) @ w >= 0).astype(int)"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 24,
   "id": "6023730d",
   "metadata": {},
   "outputs": [
    {
     "name": "stdout",
     "output_type": "stream",
     "text": [
      "logreg test acc: 0.823\n",
      "svm    test acc: 0.793\n"
     ]
    }
   ],
   "source": [
    "w_log, hist_log = logreg_gd(x_train, y_train, lr=0.1, n_iter=500, lam=1e-3)\n",
    "pred_log = logreg_predict(x_test, w_log)\n",
    "acc_log = (pred_log == y_test).mean()\n",
    "\n",
    "w_svm, hist_svm = svm_gd(x_train, y_train, lr=0.01, n_iter=500, lam=1e-3)\n",
    "pred_svm = svm_predict(x_test, w_svm)\n",
    "acc_svm = (pred_svm == y_test).mean()\n",
    "\n",
    "print(f'logreg test acc: {acc_log:.3f}')\n",
    "print(f'svm    test acc: {acc_svm:.3f}')"
   ]
  },
  {
   "cell_type": "markdown",
   "id": "bd25b963",
   "metadata": {},
   "source": [
    "### 6. Do your hyperparameter search."
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 25,
   "id": "0996a662",
   "metadata": {},
   "outputs": [],
   "source": [
    "x_tr, x_val, y_tr, y_val = train_test_split(x_train, y_train, test_size=0.2, random_state=0, stratify=y_train)\n",
    "\n",
    "log_lrs  = [0.1, 0.05]\n",
    "log_lams = [0.0, 1e-3]\n",
    "\n",
    "svm_lrs  = [0.03, 0.01]\n",
    "svm_lams = [0.0, 1e-3]"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 26,
   "id": "1313386f",
   "metadata": {},
   "outputs": [],
   "source": [
    "best_log = None # tuning\n",
    "best_log_acc = -1.0\n",
    "for lr in log_lrs:\n",
    "    for lam in log_lams:\n",
    "        w, _ = logreg_gd(x_tr, y_tr, lr=lr, n_iter=300, lam=lam)\n",
    "        acc = (logreg_predict(x_val, w) == y_val).mean()\n",
    "        if acc > best_log_acc:\n",
    "            best_log_acc = acc\n",
    "            best_log = (lr, lam)\n",
    "\n",
    "# tune svm\n",
    "best_svm = None\n",
    "best_svm_acc = -1.0\n",
    "for lr in svm_lrs:\n",
    "    for lam in svm_lams:\n",
    "        w, _ = svm_gd(x_tr, y_tr, lr=lr, n_iter=300, lam=lam)\n",
    "        acc = (svm_predict(x_val, w) == y_val).mean()\n",
    "        if acc > best_svm_acc:\n",
    "            best_svm_acc = acc\n",
    "            best_svm = (lr, lam)"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 27,
   "id": "12dd3c7a",
   "metadata": {},
   "outputs": [
    {
     "name": "stdout",
     "output_type": "stream",
     "text": [
      "best logreg (lr, lam): (0.05, 0.0)  test acc: 0.827\n",
      "best svm    (lr, lam): (0.03, 0.0)  test acc: 0.793\n"
     ]
    }
   ],
   "source": [
    "w_log_best, hist_log_best = logreg_gd(x_train, y_train, lr=best_log[0], n_iter=500, lam=best_log[1])\n",
    "log_test_acc = (logreg_predict(x_test, w_log_best) == y_test).mean()\n",
    "\n",
    "w_svm_best, hist_svm_best = svm_gd(x_train, y_train, lr=best_svm[0], n_iter=500, lam=best_svm[1])\n",
    "svm_test_acc = (svm_predict(x_test, w_svm_best) == y_test).mean()\n",
    "\n",
    "print(\"best logreg (lr, lam):\", best_log, \" test acc:\", round(log_test_acc, 3))\n",
    "print(\"best svm    (lr, lam):\", best_svm, \" test acc:\", round(svm_test_acc, 3))"
   ]
  },
  {
   "cell_type": "markdown",
   "id": "6ee5ffc5",
   "metadata": {},
   "source": [
    "### 7. Calculate your training and test accuracies."
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 28,
   "id": "42769393",
   "metadata": {},
   "outputs": [
    {
     "name": "stdout",
     "output_type": "stream",
     "text": [
      "logreg  train acc: 0.834   test acc: 0.827\n",
      "svm     train acc: 0.791   test acc: 0.793\n"
     ]
    }
   ],
   "source": [
    "train_acc_log = (logreg_predict(x_train, w_log_best) == y_train).mean()\n",
    "test_acc_log  = (logreg_predict(x_test,  w_log_best) == y_test).mean()\n",
    "\n",
    "train_acc_svm = (svm_predict(x_train, w_svm_best) == y_train).mean()\n",
    "test_acc_svm  = (svm_predict(x_test,  w_svm_best) == y_test).mean()\n",
    "\n",
    "print(f\"logreg  train acc: {train_acc_log:.3f}   test acc:
{test_acc_log:.3f}\")\n",
    "print(f\"svm     train acc: {train_acc_svm:.3f}   test acc: {test_acc_svm:.3f}\")"
   ]
  },
  {
   "cell_type": "markdown",
   "id": "efe2e9ac",
   "metadata": {},
   "source": [
    "### Plot the cost function"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 29,
   "id": "fa4a05a0",
   "metadata": {},
   "outputs": [
    {
     "data": {
      "image/png":
```

"iVBORw0KGgoAAAANSUhEUgAAAkAAAAHFCAYAAAAaD0bAAAAAOXRFWHRTb2Z0d2FyZQBNYXRwbG90bGliIHZlcnNp
b24zLjkuMiwgaHR0cHM6Ly9tYXRwbG90bGliLm9yZy8hTgPZAAAACXBIWXMAAA9hAAAPYQGoP6dpAABjoElEQVR4n
O3dd3wUZf4H8M/sJrub3nsntIROEFKkiaI0QUVQFLFwytko6v2op0HvEM9DBAELIhZOuBMBlRqkSwDFBOkdEiA9JJ
u6Kfv8/ghZWZJJAArs7m+zn/XrtK9nZZ2ae+e4u+TDzzIwkhBAgIiIisiEKuTtAREREZGkMQERERGRzGICIiIjI5jA
AERERkc1hACIiIiKbwwBEREREENocBiIiIiGwOAxARERHZHAYgIiIisjkMQGRyy5cvhyRJuHDhglnXXI0kS3nrrrSbN
s2HDhgbnCQ8Px9NPP33H/bJVO3bsgCRJ2LFjh2x9O74ufj555/h7Oy1cvm3Q91u5m3x26PU8////TSc8l7obNYgCiZis5ORnjx49v0jwbNmxAYmJiva+tWbMGs2J7t2RnJyM7t27y7L+devW4ZdffjH7ez
hgwAD07NkT06dPN+t6rM3NvjtEzREDEDVbsbGxCA4ONtnyunWaWaq6uroiNjYWXRzlvxeUcvCAEQWs2zZMnTp0gUajQenp546KGHcPz4
8TrtPvvsM7Rt2xZqtRrR0dH4z3/+g6effeFG7W48BFZaWorXX38dERhNED3377LYCa3c2LFi0yzFv7q
D1UV98hsIKCArz22mto1aoV1Go1fH19MXjwYJw4ceKm2xoeHo6hQ4fi+++/R7du3aDRaz/e87MzMQLL7yA40BgqF
QqREREIDExEVVVVUbLuHTpEkaOXFxe4u7vjiSeewK+//gpJkrB+A/qrP6cCAAAQMGAAAAqKir
wzjvvvoH379lCr1fDx8cEzzzyDnJwco3Vt27YN/fr9rfr1g5eXFxwcHBAAaGopHHnnEKEgtWbIEXbp0gbOzM1xxcXNC+fXuj
vSANHQL74YcfEBcXB0dHR7i4uOC+++5DcnKyUZu33zKYUZu33noLkiCwsvGmtASAlJQUHD"

```
hzA2LFj67y2Z88exMXFQaPRICgoCLNmzcLSpUvrHKatrKzE3/72N/j7+8PR0RF33303Dhw4UO/6hg0bBmdnZ3z22W
e37Js57d+/H8OGDYOXlxc0Gg0iIyMxadIkozZ79uzBgAED4OLiAkdHR8THx2P9+vVGbe70u1Offv36oWPHjti9ezd
iY2Ph4OBgqH91dbVR28Z+Tm/2vWrI1q1bMWDAALi6usLR0REJCQn4+eefjdrUfv5SUlLw8MMPw9XVFW5ubnjyySfr
9EGv1+O9994z9NXX1xdPPfUULl26VGfdmzZtwoABA+Dm5gZHR0dERUVhzpw5ddqdOXMGgwcPhrOzM0JCQvDaa69Bp
9PddLvIBASRiX3xxRcCgDh//rxh2j//+U8BQDz++ONi/fr14quvvhKtWrUSbm5u4tSpU4Z2n3zyiQAgHnnkEfHTTz
+JFStWiLZt24qwsDARFhZmtB4A4s033zQ8f+GFF4Sjo6OYN2+e2L59u/jpp5/Eu+++KxYuXCiEEOLMmTNi5MiRAoB
ITk42PMrLy4UQQoSFhYlx48YZlqfVakWHDh2Ek5OTmD17tti8ebNYvXq1mDhxoti2bdtNaxAWFiYCAgJEq1atxLJl
y8T27dvFgQMHREZGhggJCRFhYWHik08+EVu3bhVvv/22UKvV4umnnzbMX1xcLFq3bi08PT3FokWLxObNm8XkyZNFR
ESEACC++OILQ9tx48YJe3t7ER4eLubMmSN+/vlnsXnzZlFdXS0eeOAB4eTkJBITE0VSUpJYunSpCAoKEtHR0aK0tF
QIIcT58+eFRqMR9913n1i7dq3YsWOHWLFihRg7dqy4evWqEEKIb7/9VgAQr7zyitiyZYvYYunWr+Pjjj8Wrr75q6Mf
27dsFALF9+3bDtBUrVggAYuDAgWLt2rVi1apVIiYmRqhUKrF7925DuzfffFMAEO3atRN///vfRVJSkpg3b55Qq9Xi
mWeeuWmthRBi9uzZQqlUiqKiIqPphw4dEhqNRnTu3FmsXLlS/PDDD2Lw4MEiPDy8zmd03LhxQpIk8cYbb4gtW7aIe
fPmiaCgIOHq6mr0uag1aNAg0b1791v2zVw2bdok7O3tRefOncXy5cvFtm3bxLJly8Rjjz1maLNjxw5hb28vYmJixK
pVq8TatWvFwIEDhSRJYuXKlYZ2d/rdqU/fvn2Fl5eXCAwMFAsWLBCbN28Wr776qgAgXnrpJUO7xn5OhWj4e9WQr7/
+WkiSJEaMGCG+//578eOPP4qhQ4cKpVIptm7damhX+/kLCwsTb7zxhti8ebOYN2+ecHJyEt26dRMVFRWGts8//7wA
IF5++WWxadMm8fHHHwsfHx8REhIicnJyDO2WLl0qJEkS/fr1E//5z3/E1q1bxeLFi8WLL75oaDNu3DihUqlEVFSUe
P/998XWrVvF3//+dyFJkkhMTGxwu8g0GIDI5G4MQFevXhUODg5i8OBDBRu3S0tKEWq0WY8aMEULU/EPo7+8vevXqZd
Tu4sWLwt7e/pYBqGPHjmLEiBE37dtLL70kGsr9Nwag2bNnCwAiKSnppstsaFlKpVKcPHnSaPoLL7wgnJ2dxcWLF42
mv//++wKAOHr0qBBBCiEWLFgkAYuPGjXXmry8AARDLli0zalsbWlavXm00/ddffxUAxOLFi4UQQnz33XcCgEhNTW1w
e15++WXh7u5+022+MQBVV1eLwMBA0alTJ1FdXW1oV1RUJHx9fUV8fLxhWu0foPfee89oomS+++KLQaDRCr9ffdN2DB
g0S7du3rzP90UcfFU5OTkZ/mKqrq0V0dLTRZ/T48eMCgJg8ebRIR/LUBrr4ANGPGDKFQKERxcfFN+2YukZGRIjIyUp
SVlTXYJjY2Vvj6+hoFw6qqKtGxY0cRHBxsqOudfnfq07dvXwFArFu3zmj6X/7yFQKAzfgcZ+ToVo+HtVn5KSEuH
p6SmGDRtmNL26ulp06dJF9OzZ0zCt9vPX0Pv/zTffCCH+/JxcH2KEEGL//v0CgJg+fboQouYz7urqKu6+++6bfnZr
v7v//e9/jaYPHjxYtGvX7pbbSHeGh8DI7JKTk1FWVlbn8lhWFWVlbn8FJISAjuuecew+7okydPIjMzE6NGjTJqFxoaioSEhFuup
2fPnti4cSOmTp2KHTt23PHYYgICYGoqqoyPAYNGgQA2Llzp+
Gni4sLHnjgAaP5H3/88QbX98gjj9RZl7u7O4YNG2a0rq5du8Lf399wqKu7u7eWXX+LcuXN1lt2zZ08
UFBTg8ccfx7p165Cbm3vL7T958iSuXLmCsWPHQqH4858iSuXLmCsWPHQqH4858aZ2dnPPLII9i3b1+dsUoPPvig0fPOnTujvLwc2dnZN13X
lStX4OvrW2f6zp07cc8998Db29swTaFQ1PmMbd++9swTaFQ1PmMbd++HQDwxBNPGE0fNWoU7Ozs6l2nr68v9Ho9MjMzG+yXEMKo9k153
HiY6HqnTp3C2bNn8dxzz0Gj0dTbpqSkBPv378fIkSONzjRSKpUY03YsLl26hJMnTwIw/XenlouLS533dMyYMdDr9d
i1axeAxn9Oa9X3varP3r17kZ+fj3HjxhktV6/X44ESo3kaev9rPx+1P2/8t6xnz56IiooyFu2d+9
eaLVavPjii5Ak6ab9lCQJw4YNq7ONFy9evOU20p1hACKzy8+/eu1PPz+/Ou3qm3ajBQsW4LXXXsPatWvxwP/+
7/+wdu1a9O/fH56enhgxYgRONz59W/3OcxYWfvzxR9jb2xs90Onowhxs90nToAACGYJGXl9ekOjg6OtYZfJyVl
YWCggKoVKo668vMzDSskzIyElu3boWvry9eeuklREZGIjIyEh9++KFhWPjsWyZctw8eJFPPLII/D19UWvXr2QlJ
TU4Pbf6n3X6/W4evWQ0XQvLy+j52q1GgCafavvr7+xtNt7Ozq9OnWrXru1nfdu7cWaf2jX3UjuO
qT+24lJt9Pq9evQohRIP1v367Tf3dqVVf7WtrXLvuxn5Oa9W3PfXJysoCAIwcObLOcufOnQshBPLz8+vtW63a9//G
f6Nu9W9ZY96fWo6OjnU+u2q1GuXl5Y3ZRTLoD9f/XhsiEav+AZGRk1HntypUrhv+d17ar/Yfrejf7X3YtJycnCJYmI
jExEVlZWYb/0Q4bNNuyWg5br4+PjU+/Axsaq739+3t7e6Ny5M/7xj3/UO0/tHyufX6bh
+Li4vh9969e6e6N3796orq7Gb7/9hoULF2LSpEnw8/PDPY489BgB45pln8Mwzz6CkpAS7du3C
rs/xbve8KhQIeHh719q2pvL296/wxq+1DYz5PtX3NzMw0Oous1usqqrK8EftV/3noD6nH9+3Mj
Hx8fALjp59PDwwMKhaLB+gN/9t3U3514aN6t9bc2b8jkF6v+s16d22xYuXIjY2Nh629Wy
Y15f8gKyHwIjlqghsYAPfjgg0bt0tPThVqtFk888YQQ4s7HANVn0qRJAoDRPBoYGWthsYA/fzzz4
3Y6rrLGjJkSJ3p48ePF4GBgSI/P/+m89eOAdqwYYYPR9IbGADk5OdVZxjfffCMAiH379jW5/wUFBQKAeOONNxpss3b
tWgFArF+/XghR/xigoKAg0bVrV6NxEMXFxcLX11ckJCQYptWTOwbh+rI4Q9Q+or8+zzz4rPD0960xv7BigY8eONXkM
0F/+8hfh5eV1y/FJ5hIZGSlat25904hLXl1dLTp16mQ0Bqg+Tfnu1KKI
i7++te/3rLtrcYAff3110IIIU6cOCEAGA3+F0KIAwcOCABixowZhnW7ubmJ03HIMUH3f3dr+kHlxDxCZnbu7O2
bNmoXp06fjqaeewuOPP468vDwkJiZCo9hgzTffBFAzNiMxERFvvPACRo4ciWeffRYFBQVITExEQECA0TiZ+vTq1Qt
Dhw5F586d4eHhgePHj+Prr79GXHwczhatUP6dQIAzJO7F4MGDYJSqUTnzp2hUqnqLG/SpElYtVhg8fjqlTp6Jnz54o
KyvDzp07MXToUPTv37/JtZg9ezaSkpXLiADRs24OOPP0ZwcDDGjRuHDz74AE8++STee
ecdtG7dGhs3bsTmzZsNtbqVxx57DAwYMxceJE9OzzE/b29rh06RK2b9+O4cOH46GHHHsLHH3+Mbdu2YciQIQ
gNDUV5eTmWLVsGAIbxT3/5y1/g4OCAhIQEBAQEIDMzE3PmzIGbmdt7+evUCjw3nvv4YknnsDQoUPxwgsvQKf
T4V//+hcKCgrw7rvvNrl/EnXnXrx+WLVuGU6dOGY0PmTFjBg6dCmTFjBn788UcMGDAAM2bMgIODAz7++GPD2I/aOkZFReHJJ5/E
/PnzYW9vj3vvvRdHjhzB+++/bzhf8/Ca+ogfbt+Eq+q+JaS1atAjDhg1DbGwsJk+ejNDQUKSlpWHz5s1YsWsfIFADOn
Dm477770L9/f7z++utQqVRYvHgx5gm+//dbQd1N/d2p5e3vj+vrvG1eLKNfUMH+lLmLpp5eXnhr3/9K9LLuPAJxJpD6VXbFLr9Vd3/D0p+AlN1AGOOG4JxTixmBQGVVxqdyT6iPwy/zcDm/MGQV/
M/p03l7OyMhQsXYty4ccjPz8dTdUbK1yJo9bOo
Wrl07PP/881i4cCEUCgUGDRqECxcuYNy6tv7vzadfRC2Ixd5LMzZNmdU/0Qd+WmB7PPEq2Rpn8dsGVNIn53pb9iZ9
Jm8PmYHcCYxanob+17506VLRuXNnoVKpHububMsxg+fLjhrKfrffrpp6J169aVXNz0CpVKJtm3bimXLlonhw4eL
XuApk6dKnr06CE8PDyEWq0WrVq1EjFXzynMeNxa5ubmbCNjqdTowfP174+Pgmkgp7nnwmW4qdUCgO9lPrN6IQDQCDpyRLnhg
srhgwZIk6cOHHTGtzsf6o5Tni1VdfFRERE
cLe3l54enqKmJgYMWPGDKMzitLS0sTDz8snJ2dhYuLi3jkkUfEhg0
b6vyvuqH/RQohRGVlpXj//ngfV1bHR
bihRdeEKdPnxZCCJGcnCweeughERYWJtRqtfDy8hJ9+/YV
P/zwg2E5X375pejfv7/w8/MTKpVKBBYGilGjRokWo//vjD0Ka+0+CFqNlT1KtXL6HRaISTk5MYMGCA+OWXX4za3Okeo
MLCQuHs7FznLDIhhLLDIhhNi9e7fo1auXUKvVwt/fX0yZMkUsWbJEABBXr141tNPpdGLSxInCy9tbqDUaERsbK/bu3SvCwsIMn2EhGg5ASUlJIjY2Vmg0GuHk
5CQGDBggfvnll3r7dvLkScNTYqampgq9Xi/69esn/P39hVKpNHoCxsiRI0V+fr4hGJF4w2wYw
Wlh/GM6RYfJ/FF1hVSPk2+GL7z//WPAmM0WQeXATUPSlnAicKYoVubefPoT+NyJfKDuqjWafFb4Wkxv0WcUUbofMNTNWbzYyTXQmUx/o+3KZCcivS9CJkMKDTLT0w27vYQ37/jXKZAlnjKmfPffL0iLSqdUbVlAhj4Iy/pQnmpcy+RUyenLlYW9nqTkVWIyQzI9y6SQfhlxUCHhPiIaASs/9zA9/sPZFf+yrlaTrmmwkEdf5pnkqM7O3+K3fb0i3vnhuHDeUOGtMPOgD/3QmrYdq4ZEnsQ7MP9PfUpe9KqUP4+d3F1YhLLjyf1i3MPQnwbL+90P4Da+BV0ANDC7b0uaCtUwUYMjdvXeHk3zvzQ0WUZVrNtiT4UzJXDpnlnP4xOznI6SsV8cqQsrnbTSCmK+9C++95pXypTAMIj9f9npvUqNHQnTF9t+vQxtbsgbXUnGhrgGZQQXMfoovsIdTQWFt3DaipOwPb6mHoQ/YZZRJIXkHyFdSkx9eVDjYEIDpwGnvJLV+PQeLdgmW22CArbHD7f3tnchjBQoZdRfZIxLhUsHJPnmH47/qTr3PK+0ZnEdqPnC6yu6iplZ+X0lkO9sDVl2/Hq1QGeutIYfBbNrB/eqNLh2v+wgSaeYILEaeKzhXXt99k7ixjq3EYn1vY6Wlz9fB60hLldeXiUXZCBkwwWZKaADdYRq7+Z2ojfR+rlAx8y+8Pd1sOaxcCR+Z8u0kq9XxdpgvNNI8tRzUHPzvptf8qY7ydPd/XPUKZLukf94Ep5mM9E99RJp+l08aKwvyrGkB1e/X/SFetJ+lIfc/Ix3YI9VEW+jT5GNdC3LakyKrKwUvNJy+aU0XR5SnZgqSvQ3YQTkDdYs9bVbGuv9FRn9kc4fhzT2cm/5QHaKMm2uazY8XyFr3OQ0U3b5mbwiCBu4n+QGfSJN2Fxtb1KxCZmxBz9/wgw6GOq2yVgAoz0+lzQlnrw+P/JZ2U1YO4vZjvbNI0/IOeN1Qn0U/zJbgoxTUO2QMx2iTfDsBqtNpt63A8kt+/CNhtYBI8eeM+qM5y9sCoZxSn9DtGYTr9DWd9nDuyvXCDtWyhxn9nVg+lPrsJKmoNLAd9YjqFHFpLUMRqiAmFGxBftbxtdVA5vYcpSD8Rm2E3jebVQ6TewhBgtrAkEnp5enPDDEHr1pjm1amLKrzIk/R/nrHyUnjIUeaBVSdRK6WKrUlWoX3oGDgTfL4tNM2eyADi1BHyiKGZ7HRZI+Ef4WjmcFB0uiVNvnR6+7Np3RWyXuHSdxSnaqIL5m6dqeXZtb8pqVhGbIrCeMbmhUGe8XFLAhz73xmbW2Zt3k+3ivaDjvmIP9QILg2S57RWyMZ0sqSNQF8/yEeQj6N6cPJfI6IbX+fJnwY5vTOmRvNeEEP0sDJmP3fIdzcWVPHx52MEQLfmyZNTfXhs3vbtMRKoVNBj0XTHbbiD5pj6wS4J8qBIk4LVGm8R5L8iBPfsZoVFmhZb6b9g8ghVzAh7yATXhV4MVUhE/gVNUzh9qedaWgzDgGQ0ZvqHFQMIzQSzNkYPeKE6b3ZREqHqu+Ld+2LexlyTy8BXq3gA9m0hRDTJSjdsy2ZcuaiHdMhDT3XuRhQhBnz5N+Pus1+PpI1Fi6PN8PPm4BoSuR3M63Hk9nMZGwzi3AJNKeqP+3IHR7CGR6uUuA+uC6hfaWTgzb2AXg+AIqKTcFjFOh5YJWbdVMFSQAz0V4PMvZDuTR5HjU6/X0AnH5X+aGFZJpCDbXprJ33jEdCOCdY1DpI6cS3FPyPQdu1WgE3FiQG5u4gfHFoJt/0E6l5Mf6F0GaIUHy2aMTNOfbN1BSiMCdJsr3FaHcKh0g9mUmb5bx+J8TZnmKj0hpcoUyeO8pC0jKmKcMGEoIoPz5Q35Tlnx3n7KaS7XAnW6Vt6jC7z2e8ITI9mXCLsNb4U8z3OuZhh4Bw1EZEdT5Uf0HGn1LJpHOpI2Sxa3MUpKdlhBu4KDNu+/1nSKC4k7RAPVDqmBmB1Gxbz1iD9vDEDIwYhmTn3a3/dUHuqEXfQxJ1m4ivpR7D3O7TeVqaN65pbXaeGZpC6Ud2P0tKmFaTRxlgoZ97FRuWaQtkuOqn1pzBTXX8cX+6PQ6DzxeqnwNi9Uh2V1tcJwb8a4lZFd/tpnN7pQR2eZaxWQ8PQs/FAR7SjZ8BO5Fg0cv8jQkA6vNgWGcvJ6OuVY4r5GtQ7Kt2vVmsaxJ5HbU7LRfz7oy0
```

evVqjBw50jB9ypQp6NatG0JCQpCfn48VK1YgKSkJn3/++W2tp6CgAIsWLcLChQsZfoiaMX57yaqo1WpcuHABL774I
vLz8+Ho6IjY2Fh8/PHHhlPFbYWjoyM++OADXLhwATqdDqGhofi///s/zJw5U+6uWSU/Pz+sWLGizqn11dXV+Pvf/4
7MzExIkoTo6Gh8/fXXePLJJ29rPefPn8e0adMwZswYU3SbiGTCQ2BERERkc3ghRCIiIrI5DEBERERkcxiAiIiIyOZ
wEHQ99Ho9rly5AhcXF9kuckZERERNI4RAUVERAgMDb3nBWAagely5cgUhISFyd4OIiIhuQ3p6+i0vF8IAVI/am++l
p6c3eBn826XX65GTkwMfH59G3c6Abg/rbDmstWWwzpbBOluOOWqt1WoREhJy05sJ12IAqkftYS9XV1ezBKDy8nK4u
rryy2VGrLPlsNaWwTpbButsOeasdWOGr/DdJSIiIpvDAEREREQ2R/YAtHjxYkRERECj0SAmJga7d+9usO3TTz8NSZ
LqPG68RcLq1asRHR0NtVqN6OhorFmzxtybQURERM2IrAFo1apVmDRpEmbMmIGUlBT07t0bgwYNQlpaWr3tP/zwQ2R
kZBge6enp8PT0xKOPPmpok5ycjNGjR2Ps2LE4dOgQxo4di1GjRmH//v2W2iwiIiKycrLeC6xXr17o3r07lixZYpgW
FRWFESNGYM6cObecf+3atXj44Ydx/vx5hIWFAQBGjx4NrVaLjRs3Gto98MAD8PDwwLffftuofmm1Wri5uaGwsNAsg
6Czs7Ph6+vLAXZmxDpbDmttGayzZbDOlmOOWjfl77dsZ4FVVFTg4MGDmDp1qtH0gQMHYu/evY1axueff457773XEH
6Amj1AkydPNmp3//33Y/78+Q0uR6fTQafTGZ5rtVoANW+OXq9vVF8aS6/XQwhh8uWSMdbZclhry2CdLYN1thxz1Lo
py5ItAOXm5qK6uhp+fn5G0/38/JCZmXnL+TMyMrBx40b85z//MZqemZnZ5GXOmTMHiYmJdabn5OSgvLz8ln1pCr1e
j8LCQggh+L8LM2KdLYe1tgzW2TJYZ8sxR62Liooa3Vb26wDdeK6+EKJR5+8vX74c7u7uGDFixB0vc9q0aZgyZYrhe
e2FlHx8fMxyCEySJF5ky8xYZ8thrS2DdbYM1tlyzFFrjUbT6LayBSBvb28olco6e2ays7Pr7MG5kRACy5Ytw9ixY6
FSqYxe8/f3b/Iy1Wo11Gp1nekKhcIsXwBJksy2bPoT62w5rLVlsM6WwTpbjqlr3ZTlyPbuqlQqxMTEICkpyWh6UlI
S4uPjbzrvzp07cebMGYz33HN1XouLi6uzzC1btkttxymURERGQ7ZD0ENmXKFIwdOxY9evRAXFwcPv30U6SlpWHChAkA
ag5NXb58GV999ZXRfJ9//jl69eqFjh071lnmxIkTT0adPH8ydOxfDhw/HunXrsHXrVuzZs8ci20RERETWT9YANHr0a
OTl5WH27NnIyMhAx44dsWHDBsNZVXRkZGXWuCVRYWIjVq1fjww8/rHeZ8fHxWLlyJWbOnIlZs2YhMjISq1atQq9evc
y+PURERNQ8yHodIGtlrusACSGQxStZ6NEulMeXzYjX8rAc1toyWGfLYJ0tx2avA2SLdp3OxbhlBxDp7YCkdqF
yd4eIiMhmMd5aUJB7zel5mVrdLVoSERGROTEAWVCguwMAoKRCD21Zpcy9ISIisl0MQBbkqLKDu4MKDl1PNdYsuF5TJ3Bsi
IiLbxQBkYbV7gS5dZQAiIiKSCwOQhdWOA+IeICIiIvkwAFlYkEfNHiAGICIiIvkwAFlY0LVDYJd5CIyIiEg2DEAWZ
ghABeUy94SIiMh2MQBZWPC1Q2BXeAiMiIhINgxAFla7ByivpAKlFVUy94aIiMg2MQBZmKuDPZxUNWXnXiAiIiJ5MA
DJwN9VDYDXAiIiIpILA5AM/F14NWgiIiI5MQDJIMD1WgDiHiAiIiJZMADJoDYA8RAYERGRPBiAZODvUjMGiIfAiIi
I5MEAJAN/HgIjIiKSFQOQDGoHQWcVlaOiSi9zb4iIiGwPA5AMPB3toLZTQAggs5C3xCAiIrI0BiAZSJKEwGtXhL5U
UCpzb4iIiGwPA5BMgtw1AHgmGBERkRwYgGQS7OEIgAOhiYiI5MAAJJOga3eFT7/KQ2BERESWxgAkk9DaAa5DaAJTPAERER
GRpDEAyCfWqOQR2MY8BiIiIyNIYgGQS5lkTgLKLdCirqJa5N0RERLaFAUgm7o4quGpqr0pwMHQRDyf2uy+N/Hk7C4o+s
IAXMwrkbknREREtoUBSEa144C4B4iIiMiyGIBkVDsOiAOhiYiIiz4Qrub4nlIwcAtiIjTjL4a1JtjJIpQQJRHX8Hn6MY
ruDFqjF7Mj6S2dKWjRxFKxbW/WbrTn94BPbFNqmNJ1AKC+pmTUzPA4S1CDwhlb9lCMuDQfvJQSNnFTZQmpFQO3cXA
WlcXNmE6tDnszP6bYS8pH6DZB3OGoShwNNvUIVrz4u1IQvKNQAWbkk5Mt3E8lP2FFfPLllthzKu8trJiSbvK4Hy8b3E

```
bixdffBH5+fmGcUCZmZlNWiYAzJkzB4mJiXWm5+TkoLy8vKmbdlN6vR6FhYUQQtxW4o3ydUB2kQ67j11CmGOVSfvW
ktxpnanxWGvLYJ0tg3W2HHPUuqio8WdJyxaAakmSZPRcCFFnWi29Xg9JkrBixQq4udWMh5k3bx5GjhyJRYsWGfYCN
WWZADBt2jRMmTLF8Fyr1SIkJAQ+Pj5mOQQmSRJ8fHxu6w2Pa1OMnWcLcCq/Cr6+vibtW0typ3WmxmOtLYN1tgzW2X
LMUWuNRtPotrIFIG9vbyiVyjp7ZrKzs+vswakVEBCAoKAgQ/gBasYMCSFw6dIltGnTBv7+/k1aJgCo1Wqo1eo60xU
KhVm+AJIk3fayY64d9kpJL4AkSTcNdrbuTupMTcNaWwbrbBmss+WYutZNWY5s765KpUJMTAySkpKMpiclJSE+Pr7e
eRISEnDlyhUUFxcbpp06dQoKhQLBwcEAgLi4uDrL3LJlS4PLbG46BbnBTiEhp0iHS1fL5O4OERFRsyRrvJ0yZQqWL
l2KZcuW4fjx45g8eTLS0tIwYcIEADWHpp566ilD+zFjxsDLywvPPPMMjh07hl27duGNN97As88+azj8NXHiRGzZsg
Vz587FiRMnMHfuXGzduhWTJk2SYxNNTmOvRHRgzWG5lPQCeTtDRETUTMkagEaPHo358+dj9uzZ6Nq1K3bt2oUNGzY
gLCwMAJCRkWF0TSBnZ2ckJSWhoKAAPXr0wBNPPIFhw4ZhwYIFhjbx8fFYuXIlvvjiC3Tu3BnLly/HqlWr0KtXL4tv
n7l0D/UAAPx+8arMPSEiImqeZL0OkLWy1usA1VqXehkTV6aiS7Ab1r18t0n711LwWh6Ww1pbButsGayz5ch9HSC+u
81Q7R6go1e0vDM8ERHRbWAAaoaCPRzg46JGlV7gMG+MSkRE1GQMQM2QJEnoHuoOAEhJ4zggIiKipmIAaqa6GQZCF8
jbESIiomaIAaiZigmrCUC/XbwKjmMnIiJqGgagZqpTkBtUdgrkFutwIa9U7u4QERE1KwxAzZTGXomuIe4AgAPn8+T
tDBERUTPDANSM9YqouS/Y/vP5MveEiIioeWEAasZ6XgtABxiAiIIimoQBqBnrHuoBpULCpatluFzFzAG6MSERE1FgNQ
M+aktkPHIDcAwK/cC0RERNRoDEDNHMcBERERNR0DUDPXM7x2HBDPBCMiImosBqBmrkd4zQURz+aUILdYJ3NviIiIm
gcGoGbO3VGF9v4uADgOiIiIqLEYgFqA2tPh953jYTAiIqLGYABqAeIjvQAAv5xlACIiImoMBqAWILaVFyQJOJNdjC
xtudzdISIisnoMQC2Au6MKHQNrrge092yuzL0hIiKyfgxALUR862uHwc7wMBgREdGtMAC1EAmR3gCAX87kQghc2+
IiIisGwNQC3FXuCdUSgUyCstxPrdE7u4QERFZNQagFsJBpUT3MHcAPBuMiIjoVhiAWpDaw2B7z3AgNBER0c0wALUg
8a2vBaCzeajWcxwQERFRQxiAWpAuwW5wVtuhKwSR68Uyt0dIiIiq8UA1ILYKRWIu3ZV6F2ncmTuDRERkfViAGph+
rb1AQDsOMkARERE1BAGoBamX7uaAPR72lUUllbK3BsiIiLrxADUwgR7OKK1rzP0AtjDs8GIiIjqxQDUAvUzHAbLlr
knRERE1okBqAXqe+0w2M5TObwtBhEZGecLBXIrtIh+MZRXJ3h4iIyOowALVAajsl4q+dDr/jFA+DERE
R3YgBqIWqPRuMp8MTERHVxQDUQvVr5wsAOHiRp8MTERHdiAGohQrxdERbP2dU6wUPgxEREd2AAagFuy/aDwCw5VQ
zD0hIiKyLgxALdi9UTUBaOfJHFRU6WXuDRERkfVgAGrBugS7wctJhaKKKhy4kC93d4iIiKwGA1ALplBIhoPRbx7Pkr
YfBiIiIDBiAWrjacUBbj2fxqtBERETXMAC1cPGR3nCwVyKjsBzHMgrk7g4REZFVYABq4TT2SvRp6w0A+OlYlsy9ISEVYg
CyAfdF+wMAthzNlLknRERE1kH2ALR48WJERAo9EgJiYGu3fvbrDtjh07IElSnceJEycMbZYvX15vm/LycrW
6N8oXdgoJJzKLcDanWO7uEBERye62AtCXX36J9evX/7W9/7Lj43Hx4sVGL2fVqlWYNGkSpPRGp7aSBuO3gQsAG5AZ4MRERHVYg
3RuDBg1CWlraTec7efIkMjIyDI82bdrRtZaSRJPAnxQ+2xhPkD+OvK7xDThd7EAK7gX/xT/dTNZs3aWgQAZyDI3n7g7OUySdoUTZSAbUb8ayBBRJqBNABNXbbAHvXRG3Lo7vMZ9Cq1CKj3NRGIFg4Jd
r3MlsrTSYXYVp745UwefvzjCl7q31ruLhEREcnitiLXpk2bsGfPHsPzRYsWoWvXrhgzZgyuXr3a6OVOMmTIFS5cuxb
Jly3D8+HFMnjwZaWlpmDBhAoCaQ1NPPfWUof38+fOxdu1anD59GkePHsW0adwevqvPyyAcnJ2ZLXvC5HOkCwWeDncpMlAD9LH5ivLSBiTa/zHNP9/pQ
4SDgwMAIDk5GR999BHee+89+eHt7Y/LkyY1ezrx58/Dcc89h/PjxiIqKwvz58z58xESEoIlS5bcdD5fX1/4+/sbHkqlKlvpAnNgSbzyOLrdJ7Y5QELz8P4t/bd86KhoIAPTYTLLq/n7sQFRNaxhV6nWlY5THDZMbXCCdRuA+x2V3WsRfQMQkIlcahYHDsbs1gCtLj2R0FRasKJhmSQO8RzvROBAUuvODyzNLrU9z7z4WjeQOp9OtbpxlyNBtPn0emzCIQ
h1SZKMXvf392/6RrYwgzvV1GD9YR4GIyIisrudmdLT09G6dWsAwNq1azFy5Eg8//774/zzSEhIQL9+/Rq1jIqKChw8eBB
Tp041mj5w4EDs3bv3pvN269YN5eXliI6OxsyZM9G/f3+j6Uwg3GJp17JkvVB5EJ8q5ETUWv0gbn+rXIRUn8XBEbg3Q9snFZZQ7s70VtdQFWiXG/f
VltztpRer4deb9orKOv1egghTL7cW7k3yhdKhYTjGVqczS5ChLeTRddvaXLV2Rax1pbBOlsG62w55qh1U5Z1WwHI2
dkZeXl5CA0NxZYtWwx7fTQaDcrKyhq1jNzcXFRXV8PPz89oup+fHzIz62+SExAQgKVLlyIuLg46nQ7/+c9/MGDAAOA
zYsQN9+vQBALRv3x7Lly9Hp06doNVq8eHHzIhIQGHDh2qc6is1pw5c5CYmFhnek5oZuHDun1ehQWFkIIAYXCskO
wYoKdcSCtCN/tO4NxPQMsum5Lk7POtoa1tgzW2TJYZ8sxR62Liooa3fa2AtB9992H8ePHo1u1ueOtUkhGIfzBCxbUu9xp0
9CjCw8ObtCxJkoyeCyHqTKvVrl07tGvXzvA8Li4O6enp+Mc//mEIQL1798bt3b0NbRISEhAbE4OFCxdiwYIF9S532rR
6ZhypQphudarRYhISHw8fGBq6trk7bnVvR6PSRJg+Pj8Nid1iQdgS7LxTjjaG+Fl23pclZZ1vDWlsG62wZrL
PlmKPWTRnve1s6lZg5cybsSbaZg5cybS09OxevVqeHl5AQAOHjyIxx9/vEnLuzGACCEaDCX1iRs3Dj5qmyssAMJs3Jzs7u85eoZuJjY3FN9980+Ayduuuu4wRMVvsEmp8TzuWxTh80yrlzmtvE89pdyWGfLjaE82gAAS22zAHk+QY6zYbTeyfZ7hlQQAhbFuS9KM84UfdG/JV6g2lC6ZmpL9UZhjRoNHKXXZ7phtnDrOd
UFQ899xzSE1NNSzT1j3YJRAAsC71Mu8h7vRvQr8NRkNuu2AtAbb7xhOFPq8ODHDeO211zB48GCc03fOaCzNrYwePRrz58/H0/H
7Nmz0bVrV+zatQSbNmxAWFgYACAjI8PomkAVFRV4/fXX0bt3b8fDnM/gpJbNmrj4C+HKhndB9drLYMmbwPTZ4vB3n6/E7XxpDeDh1aqCpkaSxcAH4/n3VY
hyIy5cvY9euXejZs+ftbGqL80DHAKiUgKiUCpzKKsbxDG56d8ag7zIaz28Ktq1x5MgRhIeH46233sKRIf3Xff4f
fff8fgwYMbPIurudBqtXBzc0NhYaFZBkFnZ2fD19eXpm5fD19dXtuPLL3z9GzYfzcILfVth2qAoWfpgbtZQZ1vBWlsG62wZrLP
lmKPWTfn7fVtrVKlUKC0tBQBs3boVAWnoVAwcOBAB4enoa9gyR9RrRNQgA8GPqFej1PAxGRES257YGQd99992GBHmNqQoAcOrUKQ1QHB5u0g2R6/dv7wkVjhyuF5ThwIR+xrbzk7hIREZFF3dYeoI8++gh6vvtGJEgQF1
exR2LhxIx544AGTdpBMT2OvxKCONbNbfGWJd6WebeEBERWd5t7QEKDQ3FTz/9VGf6PkQqnxW/vWVewSfc3aG4uR+jJbzXrxH0ELjK5N7l1jb
G3HuwAtZ3y1jMRERG1ELcVgACguroaa9euxfHjxyFJEqKiojB8+HAolUpTbJ9JrdlRM/NaHJ6u4pPzFLyImNOYu58
I6Hq3FYDOnDmwYH4l/y2EbOBxROSQ2xqEFPU+xEnpsq2AkGHFkcTYCixXobGwY3B3uczSnBxiMZCneHiIjojjEA0S25aOzx
r76KyMhIpKen0dKSgrrSZUr9ZgxGWmhIpKen4EEc5GmqTJBVVy12ovGEHYxwIWyjmGeJyUxCYUCV/YCIPTQ27NDwYwXLdlYYx
M3Lywvvvvvsudu7cabLOkXm18nHGG4rAvg+5RLcneHiIjIYm4rAKnV6npnqzdYkYgmWqjMqX4uQDJu8Qq7BVa6dOtPxpt+
YYRERkM24rAA0dOhTPP/889h/889bfu1pCLhZfUpCpzKKsbxjMYPFiMiIpJJHEbuwGcnZ1x5MgRhIeH46233sKRIf3xAhTwmrZxjAEQcgBAW4eAtGD
BAkRGRiIuLg4ajQYjbQx8fFo3bo15s+fj3+82iMkafyjCiiKNC8zd3R3r1q3DmTNnMHXqVHIm6H0XIoz0cYDQDX5B8x/xZiZ3AsZAHGjhrL9wq6AjlT8cF7
EB0djdatW5u6f2QBo3qE4LuDl/DjoQzMHBoNV4293F0iIiIyq0YHoFvd5X3V6XfXPRWW/2kArDDE2hQfJSHxpx7GB4ex5aVzHS4+Srw2w+tJVC4
xhrfr+McfHhcneJiIjIrBodgFJSUhrVeDxcBSUhrVTpKk2+4MyUOSJDwZG4Y3fzikb/ZdxFNxYXwfiYioRWt0ANq+fbs5+0+0Eye6
h7EN7deAKns4ux/zxvkEpERC3bbQ2CppbHVWOPEd0CAQDf7Lsoc2+IiIjMiwGIHGYA2Hw0EzlFOpl7Q0REZD4
MQGTQMcgN3ULcUaUXWHUgXe7uEBERWQ8DEBl5sncYAGDF/jRU62X2ieb2U8z5R7bKBtnkcxz5t1oPggN3bFEHlYEj7ygu6hbVg58mdQyuxsh0FJSA3aWWgOHnoIeSxEHkLgpGG+xJpz5ZtYsyBd0K7EVsO49VRWODWhpPmmrHQ0Ro
IiMaeyVG96i5P9iyPedl7g0REZF5MABRHePiw7gTR5m2L9ccjlwvl7g4REZHJMQBRHYHuDqhhaZEFMYqMJ0/3U6TLIKNGqBHFC7u2yd0PRQvPH3Ehge7QP
scARPX6S+9WAICf/shARmGZzL0hIiIyLQYgqlenYDf0ivBElV5g+d4LcneHiIjIpBiAqEG1e4H+sz8NxboqmXtDRE
Rk0gxA1KB72vuilbcTz75TyqqvUy94iIiMg0GIDoph7pHgwvxUXS3aOzx
DD4euyN0dIiIik2AAoptyUCnxXO+avUCLtp/h/h7TGIiKhFYACiCiWxobGwY3B3uczSnBxiMZcneHiIjojjEA0S25aOzx
TEI4AOCjbWeg514gWozghIiJq5hiAqFGeiY+As9o0Jgzp17GBcfBgBYuO0MhOBeICIiar4Yg
KjRnk2IgIO9EocvF2LHqRy5u0NERHTbGICo0byc1xgXzT/pFPcCERFRs8UARFWM8IOKqUOHSpEJuPciwQERFRs8UARE3yQt9IOKqUOHSpEJuPciwQER
E1TwxA1CTezmqMv3Z16Pe3nOR1gYiIqFliAKImG9+nFdw7XEmuxjf/ZgwpUs1HCrVrxMBaauNFNTLRHc1YEDrrMmcaEmuxjf/35J7u4QERE1GQMQNZmrxh4eegVbxrYVjr4Mu0w9
6qmqZe0RERNQ0DEB0W56KC4e/qwaXC8qwYl+a3N0hIiJqEgYgui0aeyVeHdAGQM09wop1VTL3iIiIqPEYgOi2Pdoj
GBHeTsgrqcDHO87K3R0iIqJGYwCi22avVGDqoPPYAgE93n0N6fqnMPSIiImocBiC6IwOj/RAf6YWKKj3e3XRC7u4QE
RE1CgMQ3RFJkjBraDMESlhe1wMz5O1FIrxpJ9+bDc
```

9ejejoaKjVakRHR2PNmjXm3gybFhXgisd61twiY/ZPR6HnxRGJiMjKyRqAVq1ahUmTJmHGjBlISUlB7969MWjQIKS
l3fy06pMnTyIjI8PwaNOmjeG15ORkjB49GmPHjsWhQ4cwduxYjBo1Cvv37zf35ti0Kfe1hYvaDkcua7GaF0ckIiIr
J2sAmjdvHp577jmMHz8eUVFRmD9/PkJCQrBkyZKbzufr6wt/f3/DQ6lUGl6bP38+7rvvPkybNg3t27fHtGnTMGDAA
MyfP9/MW2PbvJ3VhtPi5246icKySpl7RERE1DA7uVZcUVGBgwcPYurUqUbTBw4ciL1799503m7duqG8vBwdGYOX
Mm+vfvb3gtOTkZkydPNmp///333zQA6XQ66HQ6w3OtVgsA0Ov10Ov1jd2kRtHr9RBCmHy51mBsbChW/pqGszkl+Ne
mE5g9vINsfWnJdbY2rLVlsM6WwTpbjjlq3ZRlyRaAcnNzUV1dDT8/P6Ppfn5+yMzMrHeegIAAfPrpp4iJiYFOp8PX
X3+NAQMGYMeOHejTpw8AIDMzs0nLBIA5c+YgMTGxzvScnByUl5c3ddNuSq/Xo7CwEEIIKBSyD8EyuSl9gvDS6lNYs
T8N/SMc0cHfSZZ+tPQ6WxPW2jJYZ8tgnS3HHLUuKipqdFvZAlAtSZKMngsh6kyr1a5dO7Rr187wPC4uDunp6Xj//f
cNAaipywSAadOmYcqUKYbnWq0WISEh8PHxgaura5O251b0ej0kSYKPj0+L/HIN8vXFw+dK8H3KZczbdQVr/hoHO6X
lt7Ol19masNaWwTpbButsOeaotUajaXRb2QKQt7c3lEplnT0z2dnZdfbg3ExsbCy++eYbw3N/f/8mL1OtVkOtVteZ
rlAozPIFkCTJbMu2BtOHROHnE9k4ekWLr/en47m7I2TpR0uvszVhrS2DdbYM1tlyTF3rpixHtndXpVIhJiYGSUlJR
tOTkpIQHx/f6OWkQgICDA8DwuLq7OMrds2dKkZdKd8XZWG64QPW/LSWQUlsncIyIiImOyHgKbMmUKxo4dix49ei
AuLg6ffvop0tLSMGHCBAA1h6YuX76Mr776CkDNGV7h4eHo0KEDKioq8M0332D16tVYvXq1YZkTJ05Enz59MHfuXAw
fPhzr1q3D1q1bsWfPHlm20VaN7hGC//2Wjt/TCjBzzREsHdfjpochiYiILEnWADR69Gjk5eVh9uzZyMjIQMeOHbFh
wwaEhYUBADIyMoyuCVRRUYHXX38dly9fhoODAzp06ID169dj8Bhjbx8fFYuXIlZs6ciVmzZiEyMhKrVq1Cr169L
L59tkyhkPDuI50xdMEe/HwiG2tTL+OhbsFyd4uIiAgAIAkheNneG2i1Wri5uaGwsNAsg6Czs7Ph6+trE8eXF20/g3
9tPgk3B3skTe4DX9fGD1C7E7ZWZzmx1pbBOlsG62w55qh1U/5+890ls3qhTyt0CnJDYVklpq85DOZtIiKyBgxAZFZ
2SgXef7QL7JUSth7PxrrUK3J3iYiIiAGIzK+dvwsmXrtNxps/HEVmoWkvLklERNRUDEBkES/0jUTn4JpDYVP+m8o7
xhMRkawYgMgi7JUKzB/dFQ72Suw9m4dPd4dPd5+Tu0tERGTDGIDIYlr5OOO0tB6MBAO9vPonDlwpl7hEREdkqBiCyqFE9Q
jCooz+q9AITV6agtKJK7i4REZENYgAii5IkCXMe7oQANw3O5ZbgzXVH5e4SERHZIAYgsjh3RxXmjeoKhQT87+Al/P
e3dLm7REREnoYBiGQRF+mFKfe1BQDMWnsEx65oZe4RERHZEgYgks2L/Vqjfzsf6Kr0eHHFQWjLK+XuEhER2QgGIJK
NQiHhg9FdEeTugIt5pXjtu1QIIeTuFhERtQmMYIHbhokaVkemaUJ+7nvdEWpUkPaxXJPBOy8dRxeIw3DL9t8pv1etqJcW8kA2zGhTRs6ABrPjuTG+Bu/nUqcfJ0XRAzjmRHT4Yv7AQ4yPQGKi3gI1uPQZ0pfOmBc5E0uWd+kRl5JuARsczCzyJyB6o3+q9Aw6mwTyR6/D7cVGf1a4bTwjwdSg7Cxd8eA8+ke4iYPTZmzsVd9RRPeaLoQfIDs4SlP6C+SaHW2+Z6x7ALCzbtSbQWb5M7iTOy19R1aCGg5v2phC3lAk+5wPFOXBg/4gfDtXdOWq3IFAUzoHJR0Jn54w==

xeu4tcLV5H441HEtvLC0M6BeKCjPzydVHJ3m4hIFgxARC2Qs9oOj/YIwaM9QpBZWI71hzPw46ErSE0vwN6zedh7Ng
+z1h3B3a29MbRzAO6N8oMHwxAR2RAGIKIWzt9Ng+fujsBzd0cgPb8UP/2RgZ/+uIKjV7SGK04rFRLuCvfAwGh/3Bf
thxBPXmyRiFo2BiAiGxLi6Yi/9ovEX/tF4lxOMX76IwMbj2TieIYW+87lY9+5fMz+6Rg6BLpiYLQ/BnbwQ3t/F55N
RkQtDgMQkY1q5eOMVwe0wasD2iA9vxRbjmVhy9FM/HohH0evaHH0ihYfbD2FYA8H3NPeF/3a+SCulTccVLzOEBE1f
wxARIQQT0fDYbL8kgr8fDwLW45lYdepHFy6Woavki/iq+SLUNkp0CvCE/3b1QSiCG8nubtORHRbGICIyIink8owgL
q0ogq/nMnDjpPZ2HEyB5cLygy34pj9ExDq6Yi+bb3R1U+FgW6ecHHgQGoiah4YgIioQY4qO9wX7Yf7ov0ghMCZ7GL
sOJmDHaeyceB8PtLyS/H1vjR8DcDux7PoFuqOuEhvJER6oVuoB1R2vN8yEVknBiAiahRJktDGzwVt/Fzwlz6tUKKr
wt6zedh2Igs7TmQhQ1thuN7Qgp9Pw8FeibsiPBEf6YWESG9EB7pCqeBgaiKyDgxARHRbnNQ1e4cGtPdBdrYvyu2cs
e9cPn45m4fks7nILa7ArlM52HUqBwDg5mCP2Fae6BnhhZ7hnogKcIGdknuIiEgeDEBEZBKhno4I93bGYz1DIYTAya
wi/HImD3vP5GL/+XwUllVi89EsbD6aBQBwUinRPcwDPcM9cVeEJ7qGuPNO9kRkMQxARGRykiShvb8r2vu74rm7I1B
VrccflwuRfDYPv17Ix8ELV1GkqzIMqAYAe6WEzsHu6BFeE4q6hXrwVh1EZDYMQERkdnZKBbqHeqB7qAcAoFovcDKz
CL9eyMeBC/n49Xw+sot0OHjxKg5evIpPdp4DAIR5OaJriDu6hbijW6gHogJcObCaiEyCAYiILE6pkBAd6IroQFeMi
w+HEAJp+aU4cD4fv124it8u5uNsTgku5pXiYl4p1qVeAQCo7BToGOiKriEe6Bbqjq4h7gj2cOCVqomoyRiAiEh2ki
QhzMsJYV5OeLRHCACgsLQShy4VIDW9ACLpV5GaXoCrpZX4Pa0Av6cVAL/UzOvtrEKHQDd0CnJDxyA3dAxyRZA7QxE
R3RwDEBFZJTdHe/Rp64M+bX0AAEIIXMwrRUr6VaSmFSAlvQDHrmiRW1xhuKlrLQ9He3QMcrsuGLki1NORoYiIDBiA
iKhZkCQJ4d5OCPd2wkPdggEA5ZXVOJ6hxZErWhy5VIgjVwpxKqsIV0srjQZYA4CLxg4dA93QIdAV7QNc0d7fBa19n
XnmGZGNYgAiomZLY69Et1APdLs2uBoAdFXVOJVZjMOXawLRkcuFOJFRhKLyKiSfy0PyuTxDW6VCQitvJ7Tzd0HUtV
DUPsAVgW4a7i0iauEYgIioRVHbKdEp2A2dgt0M0yqr9TiVVYQjlwtxPKMIJzK1OJFZhILSSpzOLsbp7GL89EeGob2
Lxq4mDPm7on2AC9r4uqCNrzM8eFo+UYvBAERELZ69UoEOgTVjgmoJIZCl1RnC0ImMmp9nsotRVF5luK3H9bycVGjt
64w2fs5o7eOMNn41h9F8XdTcY0TUzDAAEZFNkiQJ/m4a+Ltp0K+dr2F6RZUeZ3OKrwtGNaHockEZ8koqkHc+H/vP5
xsty0VjVxOMfJ2v/XRBpI8zgjwceP8zIivFAEREdB2VnQJRAa6ICnA1ml6iq8K5nBKczq4JRKezi3E2uxgX8kpQVF
6FlLQCpKQVGC9LqUColyPCvZzQyscJ4V5OiPCu+Z17jYjkxQBERNQITmq7OmOLgJpB1xdyS42C0ZmsYpzPK0FFlR5
nsotxJrsYOG68PEeV0hCIIq6d3Vb7u4ejPcMRkZkxABER3QG1nRLt/F3Qzt/FaHq1XiCjsAznc0twIbcE53JLDL+n
Xy1DaUU1jmVocSxDW2eZLho7hHg4ItTTEaFejgjxdESIuwaOohzunnpoVLwdCNGdYgAiIjIDpUJCsIcjgj0c0buNj
9FrFVV6XLpaivPXQtH568LRlcJyFJVXNRiOJOkoAlw1CPG8FpCuD0kejvB2VnHvEVEjMAAREVmYyk6BVj7OaOXjXO
e18spqXLpairT8UqTllSItv6zm9/wSpOWVorxKjyuF5bhSWF5nMDZQc2gtyN0BQR4OCHR3qPn9uud+LmrYKbkHiYg
BiIjIimjslWjt64ZLWvsaH1PR6PbKysqBwdMelgnKk59eEpOt/ZmjLUVpRbbi2UX2UCgn+rhoEuTsg0F1TE5KuhaNg
95qfTmr+aaCWj59yIqJmQpIk+Lio4efmgJgwjzqv66qqcelqGa4UlOHytZ+XCq49LyhDRkE5qvQCl689b4i7oz0C3
Bzg76qGv5sD/F018HdTw89Vc226Bq4OdjzURs0aAxARUQuhtlMi0scZkfUcWgNqBmbnFOlw+bpQVBuWakNRUXkVCk
orUVBaieMZ9S4GAKCxV1wLRhr4u2rg56ZBwLXnftd++jjzcBtZLwYgIiIboVT8efHH+vYgAYC2vBJXCsqQUViOrML
ymp/acmRqy5FZWPOzoLQS5ZV6XMgrxYW80gbXp5BQs8fKtSYM+bqq4eOshs+Nz13UvCktWZzsAWjx4sX417/+hYyM
DHTo0AHz589H7969bznfL7/8gr59/8gr59+6Jjx45ITU01em316tWYNWsWzp49i8jISPzjH//AQw89ZKYtICJqOVw19nD1t
0d7f9cG25RXViNLe104uiEoZRWWI6tIh2p9ze1GsrS6RqzXDj4uavi6aODjor72u9pomq+LGu68RhKZiKwBaNWqVZ
g0aRIWL16MhIQEfPLJJxg0aBCOHTuG0NDQBucrLCzEU089hQEDBiArK8voteTkZIwePRpvv/02Ro3
Cnj17OKtXL3NvEhFRi6exVyLMywlhXk4NtqnWC+QV62oCkaVaHnKKaR3ZRec3vxTpka2t+VlTpoS2vgra8CmdzSm66
bnulBC8nNbycVfByVsPLSVXzcK6Z5u2sgpeTGp5OKng7q+Gg4p4lqp8khBByrbxXr17o3r07lixZYpgWFRWFESNGY
M6cOQ3O99hjj6FNmzZQKpVYu3at0R6g0aNHQ6vVYuPGjYZpDzzwADw8PPDtt982ql9arRZubm4oLCyEq6vrRZubm4oLCyEq2vD/wu6HX
q9HtnZ2fD19YVCwWPj5sI6Ww5rbRkttc5CCGjLqpBTXI7souuDUt3QdLW0ssnLd1Qpa8KSkxrezip4Ov0ZnLyvhab
asOTuaA97hdQi62yNzPGZbsrfb9n2AFVUVODgwYOYOnWq0fSBAwdi7969Dc73xRdf4OzZs/jmm2/wzjvv1Hk9OTkZ
kydPNpp2//33Y/78+SbpNxERmY4kSXBztIebo32dU/9vVFGlR7Gn1ycC+pMEzLLalAR
ZUepRXVKM0vQ3p+w2e9Xc9JpYSrRgkvFw08HGvCkYfjtYeTveF3d0d7w2vcy9Q8yRaAcnNzUV1dDT8/P6Ppfn5+yM
zMrHee06dPY+rUqdi9ezfs7OrvemZmZpOWCQ6nQ463Z/HqLXamquv6vV66PX6Rm1PY+n1egghTL5cMsY6Ww5rbRm
sM2CnQM2p+a5qADcPS0IIFOuqkFdS8WcwKrkWmIorkFs7/dq0/JIK6AVQUlGNkopqZGgrGt0vjb3CEIpqAtJ1P51q
prs71Px0dbCHm8YOrg72sLfxM+TM8ZluyrJkHwR942A2IUS9A9yqq6sxZswYJCYmom3btiZZZq05c+YgMTGxzvScn
ByUl5ffdF1NpdfrUVhYCCEEd6+aEetsOay1ZbDOt8cRgKMGCNZIgLcagLrednohUKSrRkFpBS7nFqJaqYFWp0dBWc
3YpMKyKhRc+1lo+FmNKr1AeaUeGdcGgjepb/YKuGiUcFMr4aZRwlWjhKtnYIvaDq71/lTCRMHF7USbho7OKuVUCqa/0Bwc3y
mi4qKGt1WtgDk7e0NpVJZZ89MdnZ2nT04tQM1G/fbbb0hPOM1G/fbbb0hJScHLL78M4M4M/0aGdnhy1btuCee+6Bv79/o5dZa9q0aZgy
ZYrhuVarRUhICHx8fMwyBkiSJPj4+PAAx4xYZ8thrS2Ddb49jgAcNUCwRgIAsTK1W480334RarZa7Ky0a62w5rLVlsM6WwTpbjty15iBoIiIisjncA0REREQ2hwGIiIiIbA4DEBEREdkc
BiAiIiKyOQxAFrR48WJERERAo9EgJiYGu3fvlrtLzcquXbswbNgwBAYGQpIkrF271uh1IQTeeustBAYGwAAf369uh1IQTeeustBAYGwsHBAf369
cPRo0eN2uh0Orzyyivw9vaGk5MTHnzwQVy6dMmCW2H95syZg549e8LFxQW+vr4YMWIETTnaVSXz90/Q2sH5m4UQRcLmQP8uYtKsdN20LMpN8l
ZcCC4uLg4bN240vM46m8ecOXMgSRImTZpkmMMZa37m33noLkiQZPfz9/Q2vW12NBVnEypUrhb29vfjss8/EsWPHxMS
JE4WTk5O4ePGi3F1rNjZs2CBmzJghVq9eLQCINWvWGL3+7rvvvChcXF7F69Wpx+PBhMXr0aBEQECC0Wq2hzYQJE0RQ

```
UJBISkoSv//+u+jfv7/o0qWLqKqqsvDWWK/7779ffPHFF+LIkSMiNTVVDBkyRISGhori4mJDG9baNH744Qexfv16c
fLkSXHy5Ekxffp0YW9vL44cOSKEYJ3N4cCBAyI8PFx07txZTJw40TCdtb5zb775pujQoYPIyMgwPLKzsw2vW1uNGY
AspGfPnmLChAlG09q3by+mTp0qU4+atxsDkF6vF/7+/uLdd981TCsvLxdubm7i448/FkIIUVBQIOzt7cXKlSsNbS5
fviwUCoXYtGmTxfre3GRnZwsAYufOnUII1trcPDw8xNKlS1lnMygqKhJt2rQRSUlJom/fvoYAxFqbxptvvim6dOlS
72vWWGMeArOAiooKHDx4EAMHDjSaPnDgQOzdu1emXrUs58+fR2ZmplGN1Wo1+vbta6jxwYMHUVlZadQmMDAQHTt25
PtwE4WFhQAAT09PAKy1uVRXV2PlypUoKSlBXFwc62wGL730EoYMGYJ7773XaDprbTqnT59GYGAgIiIi8Nhjj+HcuX
MArLPGvBmqBeTm5qK6uhp+fn5G0/38/JCZmSlTr1qW2jrWV+OLFy8a2qhUKnh4eNRpw/ehfkIITJkyBXfffTc6duw
IgLU2tcOHDyMuLg7l5eVwdnbGmjVrEB0dbfgHn3U2jZUrV+L333/Wuc1fqZNo1evXvjqq6/Qtm1bZGVl4Z13
3kF8fDyOHj1qlTVmALIgSZKMngsh6kyjO3M7Neb70LCXX34Zf/zxB/bs2VPnNdbaNNq1a4fU1FQUFBRg9erVGDduH
Hbu3Gl4nXW+c+np6Zg4cSK2bNkCjUbTYDvW+s4MGjTI8HunTp0QFxeHyMhIfPnll4iNjQVgXTXmITAL8Pb2hlKprJ
Ngs7Oz66Rhuj21ZxrcrMb+/v6oqKjA1atXG2xDf3rllVfwww8/YPv27QODjZMZ61NS6VSoXXr1ujRowfmzJmDLl2
64MMPP2SdTejgwYPIzs5GTEwM7OzsYGdnh507d2LBggWws7Mz1Iq1Ni0nJyd06tQJp0+ftsrPMwOQBahUKsTExCAp
KcloelJSEuLj42XqVcsSEREBf39/oxpXVFRg586dhhrHxMTA3t7eqE1GRgaOHDnC9+E6Qgi8/PLL+P7777Ft2zZER
EQYvc5am5cQAjqdjnU2oQEDBuDw4cNITU01PHr06IEnnngCqampaNWqFWttBjqdsePH0dAQIB1fp5NPqya6lV7Gv
znn38ujh07JiZNmiScnJzEhQsX5O5as1FUVCRSUlJESkqKACDmzZsnUlJSDJcSePfdd4Wbm5v4/vvvvxeHDh8Xjjz9
e7ymWwcHBYuvWreL3338X99xzD09jvcFf//pX4ebmJnbs2GF0OmtpaamhDWttGtOmTRO7du0S58+fF3/88YeYPn26
UCgUYsuWLUII1tmcrj8LTAjW2hRee+01sWPHDnHu3Dmxb98+MXToUOHi4mL402dtNWYAsqBFixaJsLAwoVKpRPfu3
Q2nFVPjbN++XQCo8xg3bpwQouY0yzfffFP4+/sLtVot+vTpIw4fPmy0jLKyMvHyyyy8LT09P4eDgIIYOHSrS0tJk2B
rrVV+NAYgvvvjC0Ia1No1nn33W8G+Cj4+PGDBggCH8CME6m9ONAYi1vnO11/Wxt7cXgYGB4uGHHxZHjx41vG5tNZa
EEML0+5WIiIiIrBfHABEREZHNYQAiIiIim8MARERERDaHAYiIiIhsDgMQERER2RwGICIiIrI5DEBERERkcxiAiIga
YceOHZAkCQUFBXJ3hYhMgAGIiIiIbA4DEBEREdkcBiAiahaEEPjvvffQqlUrODg4oEuXLvjuu+8A/Hl4av369ejS
Qs0Gg169eqFw4cPGy1j9erV6NChRqNcLDw/Hvf//b6HWdTocpU6YiJPHDnHu3Dmxb98+MXToUOHi4mL4O2dtNWYAsqBFixaJsLAwoVKpRPfu3
dHxMfH4+TJk+bdcCIyCwYgImoWZs6ciS+++AKLFy/G0aNHMXnyZDz55JNGd2t+/fXX8e9//xu//fYbfH198cADDxiuhnzw4EGMHj0ajz32GA4fPoyZM2fi
H7/4+gcGUhkAmnQHXnb9rGnm0imNLBoC4o3/zsgg+BDdn7Ve1Mn7+oT
PorKyEkBNcBk1ahQee+wxHD58GG+99RZmzZqF5cuXG+Z/6qmnsHLixYsADHjx/Hxx9/DGdnZ6N+zJgxA//+97/x
22+/wc7ODs8++6xFtp+ITIs3QyUiq1dSUgJvb29s27YNcXFxhunjx49HaWkpnn/+efTv3x8rV67EmE9aNXJjf5/0t/a
g7G8uXLMWrUKDzxxBPIcnBli1bDPP/7W9/w5o1a3DgwAEhuBu7Sjmh6nBVk4uMMvIxBi+4KyUT94g
Y4PZfsHcc81CPP6OF1RdIjO/B0Vy8wTrDpjG+yZ4qbLdp3xQPBvT0FB5Btwb6/40WafA6fTDGRjJ7yPLtrMJ8Vx/3
fcBkLJNPyHIVzzLLtxKUo9qgbMjShpLtG/cWwjDnQjIYRztwRcu9Zuo0hzJ7yvTnvYnzaR9bBdGBcM0dzj5hTo9c
Qbp1lDpnaz9CBvWgE3gSNhcnOTN7ZXgdJ9dPHR1Mwj7yM2kvmGYcs1OTQmlIuPvnlLnFiRJzrmW2kNdXVLc4MQIV
```

n4mJieLuu++utTZH2bx5s3B1dRVdunQRK1euFNu2bRMrVqwQjz76qHnMjh07hKurq4iNjRXr1q0TGzduFIMGDRKSJ
Im1a9eaxz399NPCw8NDLFiwQGzfvl1899134o033hCLFi0SQghx6tQpMWLECAFA7N271/xTVlZWbX19+vQR/v7+Ij
Q0VLz33nvif//7n3j++ecFAPHss8+axxkMBvHggw8KT09PMW/ePJGamio+/PBDERYWJjp27ChKS0vNYyMiIkRISIh
o06aNWLFihdi+fbvYt29ftTWsXr1aSJIkhg8fLjZs2CC+/fZbMXToUKFWq8XWrVvN41599VXz53LGjBnif//7n1iw
YIHw9PQU3bp1E+Xl5eaxTz31lAAgpkyZYv78BgYGivDwcJGbm2se9+GHHwpJkkTfvn3Fp59+KrZu3SqWLFkiJk+eb
B4zduxY4ebmJmJiYsTbb78ttm7dKl555RUhSZKYN29eta+LnAPDDTUp9Qk3QUFBoqioyLwsKytLqFQqkZycbF72wA
MPiFatWlkFhylTpgitVisuX75ss47KykpRUVEhJkyYILp162a1b19f32rXvdmXX34pAIhDhw5VO+b3338XAMQHH3x
gsbx79+4iNjbW/NgUboYNG2YxburUqQKAeP755y2WDx8+XPj5+dVaY2JioujQoYPV8r/97W/C09PT4h8dg8EgOnbs
aPEepaenCwBi2rRpFuuvWbNGALAZbubMmSNUKpUoKSmptT5HiI6OFtHR0eLatWvVjunZs6fQ6XWoa+yslJ07txZt
GrVShiNRiGEEJ07dxbDhw+vcX/PPvusqM//Nfv06SMAiK+//tpi+T/+8Q0+hUqnE+fPnhRBCfPbZZwKAWL9+vcW4/f
v3CwBiyZIl5mURERFCrVaL48eP17r/q1evCj8/P6vPmsFgEF27dhXdu3c3LzO0FOm+re/08++UQIceNzcnNAEUKIX37
5RQAQs2fPFkIIUVxcLHx8fMS9995r7rEtY8eO0FQDE559/brF88ODBon379rW+RmraeFiKnFa/fv3g7fhwfUFASd
Tofz588DqDrU9MMPP+CRRx6Bh4cHKisrzT+DBw9GWVkZfv75Z/P6X3zxBXr16gUv9+uLjA1dUVy5cvR3p6utW++
/fvj5YtW9Za41133QU3Nzc89dRT+Pjjj3HmzBmrMXfeeSdi2Px0UcfmZelp6dj3759GD9+vNX4oUOHWjyOiYkBAA
wZMsRq+eXLl2s9NHXp0iXodDqr5Tt37kT//v0REBBgXqZSqTBy5EiLcdu3bwcA/P3vf7dYPnLkSLi4uNjcp06ng9F
oRFZWVrV1CSEs3rP6/Nx66ZmJ06cwOnTpzFhwgRotVqbY65evYpffvkFI0aMsLgiR61WY8yYMfjjzzz9x/PhxAED3
7t3x/fffY+bMmdixY4fdzPx9vbGQw89ZLFs9OjRMBqN+PHHHHwEA3333HVq0aIFhw4ZZLFjjh8X6X
bp0Qbt27Wrd9549e3e3D58mWMHTvWYrtGoxEPPvggg9u/fj6tXr1qsU937b/p8mH69+eo5oKp/MTEx5sNde/bsQVFRES
ZPngxJkmqsU5IkDBs2zO1mv4OIOffFcENOy9/f32qZR+PyspKLCwMn3zzyCfbu3Yv9+dCVq1a2aQ2/gwcPk5eUBADZs2IC
RI0ciLCwMn3zyCfbu3Yv9+/dj/PjxKCsrs9pPs9pPSEhInWqMjo7G1q1bodPp8OyyyyI6OhrR0dF49913Lcaee/fu
xR9//AEA+Oijj6DRaPDYY49Zbd/isZubW43Lb/dV/s2vXrtn8Rz4/Px9BQUFWy29dlp+fDwAIDg62WO7i4i4mLzP
QJg3l9NQWDnzp1W71tdfwYMGFDtdk3ngbRq1araraMVeuXIEQwub7HBoaavG633vvPfzzn//j5+WH48O
E4efJktduvC1u9N/XYtO/s7GwUFBTAzc3NqgdZWVnbF1VaVlZXmf7BuZdrfzbNCt4qNjcX+/ftrfwE23Dyjd6vAwEAAsHkSq0nLli2hUq
6tbqemp6vy/tj4uHhYfXZ1Wg0tX7mqemz/d8momagZcuW5v9pP/vsstf9Nnvk/Rd9dDW6zuxMTa
/kd5s969e6N3794wGAw4cOAAAFi1ahKlTpyIoKAiPPvooACxxx5DUlISVq5cif/7v//D6tWrMXz48DrNDt2ugIAAq
3+ogKrwaPpH7ma3zraYNZpF1VaVlZXmf7BuZdrfzbNCt4qNjcX+/ftrfwE23Dyjd6vAwEAAsHkSq0nLli2hUq
mQmZlp9dylS5cA3Kjd09MT8+bNzpw5U7AQCv4Stit+Tn8mwRYkrK2X1EnbQ2KaqM7XYFT3fSmwpU/o1S3ydsGs
o2bOnzTG3hq/q3n9TraZfMzMzrYLLpUuXzPusy/tDxHBDzZaHhwf69kJlQ+NCl0TpyIoKAiPPvooACxxx5DUlISVq5cif/7v//D6tWrMXz48DrNDt2ugIAAq
1VINpVar0aNHD3To0AFr1qzBr7/+ag43LVu2xPDhw7Fq1SrEx8cjKyvL5iEpR+jQjQoQM2btxotbxPnz5ISUlBXl6e+
R8eo9GIL774wmJc3759AQBr1qxBbGysefnnn3+OykYsp+0OyjwOO+sfF7AMeS+x0OAnxaZP39gXOSWZQUp6sUd3EA8WUPR
GkpCRoNBqrMZ6enmujRowc2bNiAt99+G+7u7gCqXv8nn3yCVq1a2TyEAPewHgUzmQjsuG3d3B0hZ6jc5DmvN71eCoHadQmI
7Zt5WbYqLi/HNN99YHJr69NNPoVKpcN999wGwOkS5du1aGAWG9OjRo969qE6XA3QKMHRqE6vXrF3QokULLHDt2DF0mTKnTOTW9/6bP
R//+/QFU/UfinnnvuMY/bv38/0tPTMWfOHH19sWzZMjz66KP1+o8ENR8MN9Ssvfvuu3dvPPPMM
4iMjERxcTFOnTqFb7/9FU2bQMA8yWykydPxogRI3DhwwuWLcO2bdswsZMgQtG7dGmVixYg
UAYODAgRZjx48fj3Xr1mHlHKlClo1aqV1f0O0rdvX6xYsQInTpyw+Ad7zpw5w+Pbbzd9b5n+PbbbbzFgwADMmTMH7u7uwLZsmflcC5W
q6qh3TEwMHn/8cSxcuBQgjhw5grfffhs+Pj429429/nzzzj+jT58dsd5+Gi3T8+iv3DtXjjxYgbwbNgw9e/bEtGnT0LP1a2Rk
ZOB///sf1qxZAwBITk7G/fff8+MHTT1G7HuIV98Ti9GfJDbVJDuZj60X9JkDaT6Lp1a2Rk8/DsW3dBbiGGEtenKOv8HOnxK1bTyx
j4eHh4eAKrOqwKAN998E4mJiVCr1bUGbn9/fzzzzDDPIyMhAu3btkJKSgv/+97945pln0MEMP2N5LQPkvjgv
F44YUX0L17d7i6uuLPP//Tjm2QdM3+ORRx6pd3+8vLywaNEiBikjL2sHu2G9gf9s+IIG07n1SrVTZeIxw6eXWDzQvTHyJHpn38ufyTBU38/dD0WOhIYaU8BZuklJIDbMbw4n
FOhs2bICLiwvvuv/9+HD16FCL/+//DK6du1qPkerffv2eOqp7Bo0SKoVCokJibi3LlzePnllxEeHo5p06ah0h2pa5hPeH6zE
4sSJGDhwIP7xj38gKCgIp06dwm+//YYb333+vDA+6n75+CpI3sBqaJCmBEbrTpTpEAUd0KUMB3v3Kh3ECl3Sf0xIgGhgyU1T76Fcn
wgMDBQJCQni9ddftxj3xhtviMjISCHRaERMTIz473//y9p8wNtlx6Oy7t9VTA5hhGGMaF8hpCDzF7PYoqUHv9w7eUpVXZ9LjPwz
iNNRgMIjw8XAAQc+bMsXredLXUF198YbHc1LP9+/dbLDV/vPVTrYUFhYKLy8v8dZbb1k9t2vXLtGjRw+h0WqFUHNTGXj0yFcfIEpOG0E
wmDFjhnjzzzTcFAFFQUGAep9frxfTp4Vd04VqOpxNarVb07NlT7N7N7N271+b7cerUKX9BUMhtq9Z9Ijx1ulsBjBDG3MB4BnEu0ox6NnQTM
rvjZtWuX6N+/v/D09Btu7u7iZ8+e4ttvz7UYM3PmTBEXydatwmpNBqNaNOmjZg2bZ8rIy8szj9Hr9Li2nWWRiIziMDQB
JJk9Rm/VZ8+fUSnTp3Ejh07Rh07RFxcnNBoNCIkJETMnj1bVFRUiE8FZYaytqKgQb7/9tujatavQXCy8tLdOjQQTz99NPi5M
mT5nEREREFiyJAh9erRzp07xAhQ4Sfn59wdXUVYWFhYsiQIRafQdPn7Odbk2LYsGHCy8tLeHt7i8cee0xkZmZWeGe0xkZ2dbbM9
gMIg333xTtTGVXTri6uoqAgADx+OOPiwsXLljtOytd4d4enoKDw8PP0bFjR/Hmm2+anx87dqzw9PQs/Wn1ly
PpIQQiiSqoioyXjuuefwww8/4OjRo7XOpgwaNAjnzp3DiRMnGrSvl19+GatWrbfvRn5MgRp
Uup1dy5czFv3jzk5ubWeA4VkT3xbw4iqtW//r8eIESPMy5SktCtWzeEh4fj8uXL9WLNmDVJTU7F8+XKUl+dqe/4ssZxLfNjvd0w7zxx5HkQA3KPH2Rm/8d5kPKDD5x/xC8zA2aA/GK5srmlJ4B1+XQ7uyNXSp0rHqvcPBf3Kg5Vmp9/Nll
IG7aegoACLFy/GokWLGGyIqMH4twcR1WrLli2YNm0aDh06hOTkZHTp0gU9e/bExIkTsW3btmrjj6++/jKysLJw8eRJPP/00Kisr8d///hfHjx9HWVkZpk6dCgDQarUIDQ3F7t27kZKSgqqqqgbWHNtwTx1J0AfXSY6Imp
sZs2wTHSjZmOoZ2C7KERiFHS4G0bZcV+WHMtQRtMRTpF+e/vLN/QctzUYqR+Uk3tFLSMtKfDuhNhz6jmORrWMWVtXNEbE5jHFLFuB9+KAZ27Yxk+V5wOFhIEfmZ5TuCsRbEa4qjSOhDhn8QFs47EknfRS4J2iH4W6P9BEv1TPCvTOW+P5
FBcXIzQ01HyT0Oo0u3Bz6dIlheIeHK10GERERNcCFCxdq/eLUZhdm5Vd+HChWpv/d5QRqMRubm5CAwdvDVsOxz
/Jhr+XBPsuDfZaPI3pdVFSE8PDwGr/41qTZhvSgfHx+HhJuysjL4L4+PjwDD44Dsc/yYa/lwT7Lg32WjyN7XZdTSv
juEhERkVNhuCEiIiKnwnBDRERERToXhhoiIiJwKww0RERERE5FYYbIiIicioMN0RERORUGG6IiIjIqTDcEBERkVNhuCE
iIiKnomi4+fHHHzfHzFs2DCEhoZK5Ls+AXDFaj5KFAbtiyLJ8/V3N9tLhPnIGWurl69Bpurl69iq5du+L999v/v/o6T27K+vLvS0jTN9mlbi+bZfW5A7sMdbAtxz81Zm3/wTLY+sXhNP8Y7AnchfKaUtLqQSUL7AZuzi4H9M8VdGCLTyfnoDYzoADDAjlaQnRD+XgHtVPBRACpEoEA==
0/izZ89i8ODB6N27N9LS0jB79mw8//zzZC0S/OTExMRGJiYp3HL1u2DK1bt8bChQsBADExMThw4
ADefvtt/PWvf3VQlXVjNArklehx7nIZdDpFSyEiImrWmtS3gu/duxeDBg2yWPbAAw9g+fLLqKiogKurq9U6er0eer3e/PjG8bnXpZbGVJKAoGm7jTlsKhP3xqjUaX53jMNb3EI9E0jR1jLyanunvPrQatxXU4
3e/LioqAhA1A1TeWGo1Gu9V2Lv8q+r/zI9xdVTj8aiu7bZesGY1GCCHs+v6Rbey1PNhnebDP8nFer+uzrSYVbrKysqz+Z1JQUBAqKytx5coVhISEWK2zYMECzJs3z2r5tm3b4Ovr2yh1bt++vVG2S5bYZ3mwz/Jgn+VhMBiWRFRlGsTfTkoZK3kBRmFtJSb9vR8LGWPJB2QaSW1ZFWRLgWrUdW6a9+IdudwOlVWD1TGarXh/dN8/PYbLG/j9dcQqDzpkLlhfaoHT8DaiAuC3XnfiiKw0iB
UFGSxLCgoCJWVlcpZYTW8+tmGdiCRmZSvRQHs7xIymJfpVPENRjp5ED3YPq6V2AeqPLFUaeBE8awVEzVyFUVvU9fhSNWiwsKzdlc/6Bvu4Lnx6uusWJYXPZfEtedhb8pqdPXdFAaGwQMxVi6LVhKHtNY6A1gmErBDyvcfcu5Me8XnaR5sBLdXhIt0LbuNFNJFvW5Ts6NRXsnVMKcnc2ZOffzVXnXbGDAAsr0b
xDsSey0P9lke7LN8HNHr4uLiO9oo9tUuEGACRJsngshLC53GTWrFlISkoyPy4qKkJ8fDymTZuGPFVcGQB5UsNswnPl9YOUKA9LgL2xgCGXubGD8kHbAtjHgkyYP1o4fe5T2Jtz9zXMj3t5A9JwNYl6F9gfpMIgPq6H9Og7TQBtPl+LcFPMgsWw4E8T9q9BpwVL
PpAp/O267bpBqPRCEmSEBgYyL+gHIy9lgf7LA/2WT6O6LVWq63zWEWPqjRp00b3++ec4fvy45rLAy1+bm5t8/R2M/s3q/btbMiw1BV12MDCcuDTEaR7Vi1ZJ/gaMjNqXGIJKamr
ajsVquUqns/uEO9Nbgan4p8q9W4A7+wXEoSZIc8h6SNfZaHuyzPNhn+di71/XZlqIFMGTCU/oMjeNWVWxoRe+1MLAbJCH2Yef6TTrYpdbAyuvCW+w9YmXG9GCpbUsZfvm7CakhYyzZUV8BrTLhtz7nmVGdn4FI2pfFhIH9HTXj
5vo3cdN5VISqnWF/LSCIiInIURcNNSUkJTp06ZX6cnp6O1NRU+Pn5oXXr1pg1axYyMjKwatUqAMDEiROxePFiJCUl4cknn8TevXuxfPlyrFmzRqmXQEbsIIBTFshUfvwUOLX0pFVn6ttPGBDxBBNYBUGQVFPmvWJTJtV4vX4jkUoHOoxvgFSHB9fVTrbrP0mUfHkBqeDhU2OrECnkpCCbnOhdRWdoDjfmaVjdknNdu9hD
Y8WKFVi+fDlefPFFJcq3EmgON3Y8l4eIIjqRdHDUgcOHEC/fv3Mj/fv37h3QBRQFBsBAwdBfIQwaOmaNj21sxLzJK2Bq1mJuhWiEONTTpAE6ezMSEsAnJ5LSQyAuaIRHzIaJ
jQsXrwYoaGheO+99XS/DNzEFG5yi8sVroSIkj5UjTc9O3b13xCsC0rV66WtanTx/8+uvqDqyq4XTmcMPDUkRERE
ppUufcNHY8LEVERKQ8hhs7Ms/clsPCwFBERkVIYbuYbuwowt6uCnizA0REZFSGG7Azc7m0AuWVvL03ERGhhu7Ki

```
lhxvU1zuaf5UnFRMRESmB4caOVCoJfh5VNxPMKWK4ISIiUgLDjZ0FeF4PN7wcnIiISBEMN3bmfz3c8F43REREymC4
sTN/02Ep3uuGiIhIEQw3dsbDUkRERMpiuLEzHpYiIiJSFsONnd04LMVwQ0REpASGGzszz9zwLsVERESKYLixM9M5N
7kl+hq/8ZyIiIgcg+HGzvw8XAAAFQaBgtIKhashIiJqfhhu7MzNRYUW7jzvhoiISCkMNw5g+gJNXjFFREQkP4YbBw
i4Hm54Iz8iIiL5Mdw4gM4cbjhzQ0REJDeGGwfgYSkiIiLlMNw4QABnboiIiBTDcOMA5sNSvJEfERGR7BhuHMB8WKq
EMzdERERyY7hxgECv6+GmiOGGiIhIbgw3DhDoUxVuivWVuFZuULgaIiKi5oXhxgG8NS7Qula1lldMERERyYvhxgEk
SUIgb+RHRESkCIYbB9F5awHwcnAiIiK5Mdw4CG/kR0REpAyGGwfhYSkiIiJlMNw4yI0b+XHmhoiISE4MNw5iOueGN
/IjIiKSF8ONgwRy5oaIiEgRDDcOEsgvzyQiIlIEw42D6K7fpfjyVT0MRqFwNURERM0Hw42D+HtqoJIAowDyed4NER
GRbBhuHEStkuDvxUNTREREcmO4cSDeyI+IiEh+DDcOxBv5ERERyY/hxoF4Iz8iIiL5Mdw4EG/kR0REJD+GGwfijfy
IiIjkx3DjQDqec0NERCQ7hhsHMt3Ij4eliIiI5MNw40CBXlXn3OQU6SEE71JMREQkB4YbBzLN3OgrjSgqq1S4GiIi
ouZB8XCzZMkSREVFQavVIjY2Frt27apx/OLFixETEwN3d3e0b98eq1atkqnS+tO6quGtdQHAG/kRERHJRdFws27dO
kydOhVz5sxBWloaevfujcTERGRkZNgcv3TpUsyaNQtz587F0aNHMW/ePDz77LP49ttvZa687ngjPyIiInkpGm4WLF
iACRMmYOLEiYiJicHChQsRHh6OpUuX2hy/evVqPP300300xg1ahTatGmDRx99FBMmTMCbb74pc+V1x69gICIikpeLUjs
uLy/HwYMHMXPmTIvlgwYNwp49e2yuo9frodVqLZa5u7tj3759qKiogKurq8119PobwaKoqAgAYDQaYTQab/dlWDAa
jRBCWGw38PqXZ2YXldl9f82VrT6TY7DX8mCf5cE+y8cRva7PthQLN3l5eTAYDAgKCrJYHhQUhKysLJvrPPDAA/jww
w8xfPhw3H333Th48CBWrFiBiooK5OXlISQkxGqd5ORkzJs3z2p5bm4uysrse6jIaDSisLAQQgioVFWTYp5qAwDgfP
YV5OR42nV/zZWtPpNjsNfyYJ/lwT7LxxG9Li4urvNYxcKNiSRJFo+FEFbLTF5++WWVkZWWhZ8+eEEIgKCgI48aNw1t
vvQW1Wm1znVmzZiEpKcn8uKioCOHh4QgMDISPj4/9Xgiq3kxJkhAYGGh+MyODSgDkoMSghk6ns+v+mitbfSbHYK/l
wT7Lg32WjyN6feuRm5ooFm4CAgKgVqutZmlycnKsZnNM3N3dsWLFCvznP/9BdnY2QkJC8MEHH8Db2xsBAQE219FoN
NBoNFbLVSqVQz7ckiRZbDvIt+rNyCvR8w+THd3aZ3Ic9loe7LM82Gf52LvX9dmOYu+um5sbYmZqaarE8NTUVCQ
kJNa7r6uqKVq1aQa1WY+3atRg6dGij/aDefCM/IiIicjxFD0slJSVhzJgxgxiIuLQ3x8PD744ANkZGRg0qRJAKoOKV2
8eNF8L5sTJ05g37596NGjB65cuYIFCxbgyJEj+Pjjj5V8GTUy3cgvh1dLERERyULRcDNq1Cjk5+dj/vz5yMzMROfO
nZGSkoKIiAgAQGZmpsU9vwH/bs2YPIyEiFXkHtTJeCF16rgL7SAI2L7XODiIiIyD4UP
6F48uTJmDx5ss3nVq5cafE4JiYGaWlpMlRlP77urnBTq1BuMCK3WI9WT2ULomIiMipNc4TVZyIJEk33aWYh6aIiI
gcjeFGBgG8SzEREZFsGG5koOPMDRERkWwYbmRg/n6pIn55JhERkaMx3MjAdM5NbglnboiIiByN4UYGOm/eyI+IiEg
uDDcy4Dk3RERE8mG4kUEgr5YiIiKSDcONDExfwZBXooLRKBSuhoiIqHlxgtTaSWUW2CsRbMRHkOaoPtpyBTyMxKn8zlCW4yq
q1oFP083ADw0RURE5GgMNzLhScVERETyYLiRifn7pXgjPyIiodiuJEJb+RHREQkD4YbmfBGfkRERPJguJGJjve6I
SIikgXDjUx4Iz8iIiJ5MNzI5MbVUjyhmIiIyJEYbmSi87l87l+zg1nboiIiByK4UYmpsNSpeUGXNVXKlwNERGR82K4kY
mXxgUebmoAnL0hIiJyJIYbGel4Iz8iIkHY7iREW/kR0RE5HgMNzLijfyIgcj+FGRoH88kwiIiKHY7iREW/kR0R
E5HgMNzLijfyIiIgcj+FGRqYb+XHmhoiIyHEYbmQU6MXDUkRERI7GcCMjnU9VuMm/Wo4KgIiJwTw42M/Dzc
oFZJAID8knKFqyEiInJODDcyUqkkBHi5AeBJxURERI7CcCMz3siPiIjIsRhuZKbjVzAQERE5FMONzMx3KebMDRERk
UMw3MiMN/IjIiJyLIYbmQXyRn5EREQOxXAjM9ON/PjmluURERI7BcCMz0438OHNDRETkGAw3Mgu6flgqp7gMRqNQuB
oiIiLnw3AjM523BpIEVEBgELpfyLsVERET2xnAjM1e1Cv6eVYemsgp5xRQREZG9MdwoIMS36tAUww0REZH9MdwowHT
eTWYRww0REZG9MdwowDRzk82ZGyIiIrtjuFFA8PVwk8lwQ0REZHeKh5slS5YgKioKWq0WsbGx2LVrV43j16xZg65d
u8LDwwMhISF48sknkZ+fL1PV3CvdV+0UHOnEBAESVIikjgC+AOr2xu+NzF/m0ktmvfVRGevvtDbRHXYIGtbjNdVbpG5W
oiIiLnw3AjM523BpIEVBgELpfyLsVERET2xnAjM1e1Cv6eVYemsgp5xRQREZG9MdwoIMS36tAUww0REZH9MdwowHT
eTWYRww0REZG9MdwowDRzk82ZGyIiIrtjuFFA8PVwk8lwQ0REZHeKh5slS5YgKioKWq0WsbGx2LVrV43j16xZg65d
u8LDwwMhISF48sknkZ+fL1O19hF8/bBUNg9LERER2Z2i4WbdunWYOnUq5syZg7S0NPTu3RuJiYgMiwOX737t144
oknMGHCBBw9ehRffPEF9u/fj4kTJ8pc+e25MXNzTeFKiIiInI+i4WbBggWYMGECJk6ciiJiYGCxcuBDh4eFYunSpzf
E///wzIiMj8fzzzyMqKgr33nsvnn76CEDmym+PKdxk85vBiYiI7M5FqR2Xl5fj4MGQYMGYc+ePTb
XSUhIwJw5c5CSkoLExETk50Tgyy+/xJAhQ6rdj16vh16vNz8uKioCULm8RkRydPk3fy0HBuoPIMbHIPHhRJwXvhEJHTGjIYIx82JhUJCTrFw9YlU82B4eq9UZu9grnQPAgvPmaQ9K9uQQy9B2QwBLtuMxDA0tjDZfJ0vyDF7r6DGBSRuU/PR52wKSIEKKOL9SNWWLSYqEvIeuzQRTFAUlIFrE6DkUGvOrcVfGsTaaVa9Rc4ISI+oHoVhSjLJ0vTlJIHVwKFALYjWjQFsE62ZRhOcW/UzH4JnBy0pH20BFiBTWfEHQbEPZMWxwTb0d1wLm0pFVUUlYDlVpETyZtB3iHjA9jk7pfDdzqkSamYXNU2Q3gGqmVQ4jXoBdKmP6N1LTbmhqNRHDKlzAxyHnlYuHg8E3I8SBlCQDeZVbF5WWWtUIVoE3EqyRTyqPDo5LVdtHnMwBk8mTIvWwg42KzJWCbe+bV+bAHH0h3O8GmFsAI4C+L41CN1CQgVH3ZqoCBr5bM2OJdOSKHI7AHU4b47JIFqYmVBJI7LJL2QF1eyzX3nwB6EjAl9UxSBVbM8yh6S+Pz6jv1i8xU3LQz7kH1zQpKqcbWGvG3lJBnf/+Itnge5jJ4XU6RG6m6OaAXNMZdf7oH1nnvN2Pb5RU9JRxTeBOKKCzbtKUo8ZpIdqKuGMwM0d8fbmg4VABNZWizcvYxiKj1YL+FRtT7rPBFl1k1ht+AXFdZXT8Cv/dwGnGQPtlLBIGE1LxVlKQDZtPRuG0n9yDBqIoVHYhYd1Zhb8zQe8bRMh2MjvB3VFdy6+4Xaq0n/+G2+U1W4h8JVHxfBkDQCpCMqbfPC1CaMr1kHZ47djTKo2c8hNOCKgPo6GvUFtHThMkdLqIqY1JGZQNcqT2AEMDw/oQovhzhVCYeJGYIxHXg1+LBgVJgfFQQh3jpfk2FhOLpJXQ3hwwbUVIXVRwr5AV6QMNW5cEJ3dtYAkXh0jLqHMKXk3dsqNrZUdnY9wYp/9x3wL1ErE2jUH8YY0KjcP+dxW9pNg2dJWTf+8dT09cEPs+8gfRcjZPpFPhhsmvY8qe6W0lqS03mH0KcoE8lEoflhG4k5jJ2hr6oXFVh8DKqtWFh7EiwrUQjSfTT0QtBcwLOBx8dDsLAghk5Ns6dvnd/RmmWYoX0W4C3SjHaKeKNFtJWY9whE/V5ECSqUmMsjhBjaAiUtvLkAhH5LqHj8HQi6eCZm5tRm+cQjgtsE1iVqX4j7RBiRpyDfhxbtlKpiAlqRd4fsBqRRVoCzJTkOXjwBdqJeq1iL8EKTNDvdWYGYLgnW+sRGYoXZc84JI5jTjJGnCF+iLX5dPD0QEnyw9G7dn/8VOpvpTQ9++rJhl/2+JB0AhG6mCLjmvAG7Ot2EqmpX0F/kmYuBf/h+cDHjBhBCRAxQFYVdCqpPhpI1yI/g09E9klfB3SdJnlfLQkKXN9YjVzaEdRPGTfxHJJTpm6IJGWl6VkJAi4jDNh0P9AQ4RNNNjF7IuTDvBnUWdmXnrVHZCx+hFbTw0AHr/PP7PA8OfJwN9U76hHsjMtZ+kVOuGXcINBZM5lXEuIJIJ5W/bwdYfYL3lmhPx1VTNpp+TjRM+7rD3vW4v1wg5mK5L4VMKdqK+6WHMnQ9TBeY3N5t6Q5iqnb+mkWfp5ORVrJ5ZXy5VU4bfCDvvqpp0r4IoFwVI8IbBNvSoQMj/iTfObvOlFIYnYe6RPkPpKaBXnGYr0dnbJqbwa9E/2kOa6DL+XOdTAuxYUKVfu+E5+ob3qfFCjl1IlsH42vL4YbHGc5PGtqV+GCJLLFCS27IOzb6mL2Cz1nWbHN1gyMAvPVTa/jv0FpSc05F0bPVAGQnUz9XF5QSh5h9SQYN30XGymBQk1xkzsxwFKAbOLBN6awNPwpINWSKdQUQvmhW7QgQmIVqDX5n9b8wfvM1aHG8qbmw/cvGZVNKVcDZcrIGwUeAOw2Lb1JSHL3PoTHtFcnyLvGJOGcV05nNBA0c30fnZwoH8m2PlSj23nQqqwbnb8p98ghZuRrhuxYeKCXPdpq+e3xeRh6N4NiuMsjrEDcWrN6m3lnnC3TZvxHOJ8/Gx9b+8fs5NBS9L5fK/JUsqL9fZrwNDQSMI2oDOHYqtYLIBiqd3Bb8Y2XYdsT7PhjVd0l0GeBULuEr2cM8T0ZF6pGl+1CEP5xG9SKLX59gCu9jZhJ9zBxiUsRRFR5Q/vUTl1GZPeEJ19EARv28UTq2vcDCuJ4F7gaeUvTW2QC5gGxf+6CVAZIgGsAxAPyJrJe4ecbOoNMCvgQmtQPzOpwR3w5PPgbyXKnPejk1OgmYZtWxrnNa2Fwj8XvPE0cbPWSjhX1EmE0WkN8nBrn69wMJ9UUyLdNJCjX73oQ2QhDdU86FYGOZxAkhGVKZsT9dGHiAO2NbxCchARAebWHdqmCjJmnJs8bW06qJLyAWhLv4oWczK54ALCTPHTYsQN2vfLlI2rVa20F1fFhxrZ+42AZx4vZk7hLYZ0QWVZ1wR/JT44ymSXLhDwCfIjJhPcEXMKi7KGF1TLUbqnmAmGHtdWNd74Op8sVi8AoEKg6HFqkZU1TxP+0xORtA8MVo3ZoNdYCY+3v5UP1z/nBT1JrJPH2ZuAHJ5sHaH6xEAKH9u5vrPU0GLRgxG+SyxzIw5Et/GRLgCWJ0nKLNwiGIhDa01OCPxc9ilQyOS1y2Jxs3sCoKQpmTgBKRyLNDhJSnE6xGYoE2lnNkA50BEhzVeQWKu37jP9mQImPsEhBpMvNMkmL1Xc7nKbPRDkHFi5txLuGIq0BJvLrGaO2PvQgXF3qJn4gGpcDuJ6wBlVkS5E5VfQaf3P1nuhRmbOAXl8ywrZZyPlyeYR0sGnsjLmmVHg2D5Sz5zQ2tMRjk8Ue0C0sOB1MRCs3fkLXVg89ec6oR8KCG5W0LmQOvnGWsKU9O05lFe9Vc5MDDYJJTfKuhUzY91CfoQX8k19KdW5TAWH2ItV9oOuG0Yh8bPRdkLpPRmUZgd3VvU5YJ4mD4PRGLlQ4DYRIjdJ4dcvPC0RYePgPH3wtRn5vPBlIlAoN6ftZVeAP+LRy8fsOXUJcDPLqmiJxlE4P1dK6DeIqjwZ2kNtWCPWaJtuLW9pHxhmjn5PIAGYj9dRXlCMqRfX17ZLqMiYvmhZDHQfWggNWqNVlNCLDPWZp04F7rrVQNxeN7MxB+kqM6UQtl1cfNZqIN1h5G4dWg38PVyjQ+tECwrUH+QTwrfCM+a4rJ1WsT/8Pb2NiSoDTPoJOJ0ijkR1WkrtwrjNTY6ObMgGMpsH2fJ9jGFKngC2/3XG7P8XuL1AFXzvOHnhh83l48sB1otkq1FW8n9KNpvp5BcX3IvMbzJNOmRJAO3RQUIVBlU4VftHYCQPP6+gq1fdbLzbPezUcvE0KmXu47DVAHQkW4SDC8fNRhCq+Dn8GlxAh7YlDIWOrjsSuOf8k9Ew6mX2ecGVLsgiFRjmnEXz6LB4cshqLB9J0U+YwnwuAHXs4/A0w/3MC+Z29a0/VW6CsFpyQU7m6ZvCUYQtLOqHvJPZe8KSfqlj8jXU0hUXEo6ftTKCWZKI1YCtq3Ml8YSH9XZSjuCUZtBHyBNPW8d/Zdqt0SmVb8KBmflL1+lTEGMQm1l7qQqu8qvcvUbCOKbgNMI4LSXcCDqtLVXgjZoG7bVbG/kk5UE3jhJzmpRgDn6Td3q60GqJGKnl8tGpKYi2J3wXCqYI1bXaFsBRTNLH4WrvEh8YbqSpJkhn/0zIvTnXmChb8r9JBHcqT+EgzGo9ngT/jMDknhF72XqkO7yCkqfHQXYqQjFmzD/n1P/j/8Ooh/WiqDkEbUiuNlEgf0+IFDdGSFywY3ZvSmWtDIaVdHAGYEB8vcJ09v6PJ86f5Xp3OYU1EG/XBA/rKCXlTwDLB1g72HLcQfF9uw2WZPlGe2wbz0bL/HI6UR2Ljw28WnP3wYEDaHR9R/UfKbCb6dNfqgVQK0sfzlB2Szs3Lg+z77rIUNCMfqJHsZJI7/2CRJ6nOlkGQZMaVhTwvAyiI5Mtu6Bdm/iNZ1gdCs1BfUl2OjIp8dDPYJu3Yc3RNHhFwNdqyg8sHkJKjC1KZrKaQrJITS8MiWp4fRIdoSMKeztbpxmIhUZB8aLG/d2yfWzDBvWTSXaY6M0fZ2ZmAf8BHDMMIAMmTANUC0N2uEJ2IwHtBYt2sWJsMzF82EmjLqSmbwQdGGspRz72SL8LYZ/qBwjfjnOM2bhYdAwL7ADHGlLPYMpPL49L/tUM+1RR4BibhcNf+kcRxbSlTbN87cV8+Q1Pp+UJyvmJwqVQp3wm95PIaw/mA7P4/sUPchrOk8q+uLo0A7fRCbpYAGZDVpEl5hqQdzp4a8r/PZs4oPWr/uAt1+T6Y+l/1TOfSgy6N5tNSfJRyGaLqPb0H0KHtlyM+zRuxm8t3JM+NRnlb6vPJeOQKFKHnzQxfjSF/g2Aw2T5U4wlQHBXq1tz8MQwmgZQyH4DAj3EqflVBNeuDcB4CONrKTw5Ep3uf7IuZ5kvIIRgEehd0nGFQVoGRpN4zIHw9yD1oUuedelaZffVlh4nhJEgvEvIGl6R9p3lWcefBzXbM5yOxWa7KjDvVmuZOHddLbw3j1fsC9Tt4v1AwfyKvwT1HRxwdLg3IBTqzfXQhwuGz6zwgGlN7k2CZ48w3xxMyHI0RzTTEhZrcAeFAHHFHXiWIIAmL28zYf2/AXYIIj0DN1tn0idZpZ8MWWv+oWICdAPBUNHFs/tUiP+NdDhJwwuMWxdLYxSEDtX8eVEBMzLNq/gSNJAIAG0BvbmcVIv+9rjQsYSP2j0gkeJqfpTKxtoE/qJCCLqzvSXbrfrrnvcDVkqe0UvhaPLDGX7ZEBwpeMMgrIY2ByWFEeJ4UV7hdJhEu8ipVyzbEHrZIzJIV/B8Ui+2T7wrKR9cjG2MBFmTbDuGWQ13jmvTkDkzexpHRHnUUQqSIiKCI8V7N8ILrsAzN7QURtEQ/6WfQJvdIJvhKvSpWC0sjTXUd2KXW6+pRMPZ0rq7X0nmtjpNEpXs5VmmzS3gsqFLsJnzZf2XbLX1wXV8Fu3QlsZVJuvMdjXC7Ivs9gUpqAHAI7VUE5WgXWTzhC0cQlYMHOlqVZLUgtExsMJN+mzdFhBJMDqy7FCNzrqVUy0DsLFTjfdMBwz5bd8dCCB3eyRNGNDRsx7oGKB6Bkf4QCB3cbCdoYbXqWlg8bZPA6jbHNyM1/bN9K0yjz4mrX5PWTKc5xKS6OI1D3kKwLqnL3Lxu8+7kqoHHa+8Fs0bDRVb2YhVtUnP6xsEKdWgGvlYJFgSzwHSMw5yNyLrsU9L1IMDk9rvlvG1FdRbK4g3+oAcWV6Gwk3Dz1qSm/k8zHwxeWdXryTBLKg1w8OiYLsbSw0G6K2+0H0OCTYLvMUH1oGbSgnNCq2kVRPZMnMJLdSpGMzkj+d1nZdFcGWTBjvjJ1FCOj1CMyJCwjBUcfQYBIB4ZHYOoE7kY2mNqDEUvfU7E3sARI3hgUNqNOkwUDn7PUe1PQ6hG7ypv+vpjYNjbAYA1RMn2YKxnCkBNWjbGs5pNM2bs7wlSUrDWf05j5tyMrg1T9QUhtTzGwt7A3IIlqE7M5X0A4z8VHRnA0vc1gYI+t6zDXUDCJqlYDO6aH75vw0I9j3ac+Do/Ya6Jrz3YDsq0eA4xRNG4A66YHEG+5XrYDI1w7Y/XNmMFrjzbNIbMb1vR1+WPY+RDDE5ozg9DzjA7fF11N/DfvIeZFB2jI5yFIf3rjJM1MCHGSuQR/ILyfqMjdd0AGzDuEEHs4B3SVtEOWt1EBpZNrvXZI7D6pKfd9ou6mywvffN2XiAxOgsOjUFqQCuYqzk5F8q/kMh5Vn/1v4PQMTt3eMqV/NXB/5JyVT1Bdzxsiy8jMJjLM0C1ZwKwJE9HQ2WKWkZW3eOmDUZ0NRbFMgzD/2bxnJTVDHSfsldP0o1tu3D/8JjWhEP/DNsU7M9oGK2a9fzbKfKzEMGDJkuxOg1s0oEUWqRdBtjTEC7m9ACGiKZdCjVJDDs3FUt5jHQFnXF9mARfiEcCHSo1WgyCGlDLq/+Uy+gPR/lNJRq0eCHPBtZs4HeJCtIGeozUvq0OyLUkSxHlvU9LKGAx8+qAdIFrpnxa1I8JOtQOVQ2U6Q9ZMq7cr06ke5jggGf4U6xtbtrLCMRMGmwwtHNgtdXPkm6DmuXD/4eCvs7Ppn6pHI1a0u1KLc8dpcLrIVYUBvTTBbZ2qkS8pxJ1xmGjW4iiO8ZZG0W1uSW5eCSpqRbsz3uVqBC2pX+B9Sr3Wv5h9YAuiDqBGgBzY9aV39C5K8yfXdHgVBMKVsQUmJ7ae0nOP2K8ZMgWeHnVZ3vqwFWHY16nHHxj/6BEJvY0vM9ykCaRGUm1lYZUuRyF8A+yIqEmoHERrEv2Fko3eCp3FyK9xjDyjJ0Bgw5Wg2gjIGpL0/J4xkw6CE3rDfVNRrh4Bq21EU3XiMR4kmJJz2hEQeszfLz7lDfeZfNE3wvIRKzIDLJt8NLa7yNWZfrhZLVIBgbSINbg2bgKd3IZ2UUbNnUGz0CxEzjjQGlbzjzUZIn+cIVDrxMMKwHGoTIizQD57h+MzL+09xnLa8PJbJ8O7nQ8ySMZfrNsM7gIDQY1GLSP4Pzvh6mSI5NKaX+k8WCINBiLL1mmjWoDlOEfY+l9BI6mNQ3jkV2fOq7RW4XTKRDXwe1YA+iK2WWAXIQURkwRiyMLDnv3IKzP0pgzJKwbTBH9vPtGsKuOQvyKz0+eZ1IOj0+PvXpXpsCRGBkG1Ab8IeD5vbh4SKqiB0IqkuNnLHCbTMD0Wtn3XmXQ4YxAFuQUwiuEEMLjEKXfFmAVbsoKfRSb9A3LDNTmn3D5yVjtf6i4jo4nKOnAYbUE3Id7vDBH+ZdgB34dsFoJFxYVkWMJhB5AJtqVhjvfIsDkEXTUeVFlKtD+bATsHRzUAQpmLuD8bKO8tCkmmWOzTcfrYOX6XB9V0bLVsLMbKzlxAZv5dJzr4ZM6YsG6SBpFV+VTcyhjUsGmXZVQOljnyxnwcprVH0Py/SMlTUXlTgVBbQJ7L5yJPGo3HaNKMfLdPq9OZ5xIKfd8XNJs/RbSaeXxpuL9zhAq3DEXCzLv+jJ6tkRt0vqIXsMj6y5KcAVjCAX0V3N79F3SROr8vOPPCOB/YGphcqxNXmfrUtAHFRR/zHGMnmkdB76B3OFHvg4GMnHSWbGlP6A7y7EAEBSlajvw0GRbH3VJFMRSJ+u6Mt/vqpXxKDbtPCQsAMcDEbmVl2WwJd0gNHbnXTTxIRH9IYB3WMr8lK+oGp6IzDkBeiRUKaMQMFTvx9rhFHApNZ9IZgWCB4yFH2YNPOSqnJv1VSYD84Lx1DYBmyMEm3t2ZJLlThhxPyUQU0WTh1BrAm+9QZHyPRLjcLSBcMWKHQC6ue1iwiKoBVEtIW55vkz4eOtSYpAFCpM6xZvqs0jcAbt3cAtwVmBJx3P9ytMz64uBUFnZRbi2S7GKCr/GQQ7xBWPECNAjn1BUmmW3MbdmkHWE4bxXzQb8v8/0Xf1fjJ6/C5nBRAtFGf7tf1f0O+kFyz2HvzS9rXvAe+ffmuxgxRaXF8YbzdPQgnT6xhKRZSnmcBJFiQVUL1Wqah3KxLrCBpWkyrovuujvR7OoN/Nwu1CrEXhEH4KdAQLAd4s6OXGxM7dLNKTHvfwTKc3wwzRvUUDUhG9U/rXnnSnJ5SfzOBVqr9TRmANk2OYG4lX+yX/lsHj2ezDzP9wwbjwsiSZmSNr8/6NnDIJZ57DyS8e9pkjoENyyt5q3MA8BNTMM4JMfeLT4CdbazkXZFH0lbDjP1Ns/HXN9DMPIh6rJfNjdoXfN3Si5PWTl/hpgMkW8OV/Tji8NBeWMhBQaUSmchGqUmYC3H8yKAZ8g5QnvQMYbA9Dm5qnMm7OxOzfWEMb3hNKNN6vWv9lkDwbfqhhpMFN4e3BRU2nkBqb3ZURzgHZOQRo+kHd8wvqzMuTzzLpfoJxWO9dwEkjSuoMBgn9E6jM3MVQBnBX9Ij9FGm0UEwgD6gVY2B5QODDsJUAq9P6uBuU9Gou1Z2mEU61WWZyKzWPGpJvOQRaSDQWNALfq9RRgPNnoRBVtvUyMrX3xb/kqZ/yA1nwhLSzkdRYRzExz2PR6FKvRR5NvNSp2LvI+fq1AZTupykEQvd6LPzwCDFTgRNYAuJIEpljROwSEYDMTNSBePrCAOdbcbuj0DjuIWLMiBWcFBE4wcNLvrAb8X32PfnI0aiNv8VkXhwRxX9oD/XTgUlkgrFk+rcT3ycdfNLRBy01gD1QL4KqJlUMY/KsT8IfcFk/5MfpzB37cpKyHWUw4hXzPYu1dwrK4K6OCgQAxtbtBu5RlbFzvwK+YxwBJKHLtFL8ocdYOZIYZnhgPFI9MjMNPD+dGKMtBRtKcYHbs2jWLWZTbj3UKGFxdUbnWN/g8CYZQgkq+Bu3+N/SDDEjlW+lixANlONrLq8hIq1M5sXHRAbOUCnqZZ62VD5i9uxXaXDH4RzUKnVLNHJbq1KsSQH6wvjLKsGEFG6p3dAvh/gcuR2TJUeMX+1PLF7VCpR5rh7iQQYFpP+ISpSxBNrGCxlqyeaeK4axgJYPVTlm6qJeKuqHD8mBo+Wj0mRMzamBaiagCX0tEOuCZyKNKcmPLaGsmIm3yS6qWG3emqDF9xi85SJT0EsVPWKvy2rK7yVjsCqHxNDiIimdDeGKNFdYm01IJVdRPBVCnMwvWmB94DhfJYwbwDjMkaqq1OgL90E8cQGjmnbPfFvrt3DYyKVmhJtkLXRHKzXmx+MkyJ8XIGWi6LPoATLzdTVk3+Hv46Ynoz8ffATKJh/REgTWZTNHIr9XATZjvzMNRXE6DrwP0ybB2TDDOlxcL+LbhuKGr+yCcyE71OrDwDFFWTrYcr4TgV+qOSVEwBJcg6bk+dM/qJMkGRSKPHGIKEQcAX8dRLdqiGtPj9gMoypjp1YFPUq6TkYhvMEFIxCuxKXy+ozCO8OxbFkZMjg91l8/Tpqa4FZ7n6Rt/ePUZ7uI9zGgNFzmQtN3UXnvLEbz25JfqHXm/6RnWewAzk8pPVhfm8wxGj6ogQBnPeh7mJOd+XeGFpCp0PMrNf+tvRuSnuz7R5BjrS/Np97INpHmIGv8z9NtBpvO6zeJlOnO+f4Qt7Qz34ZAzVDv5zKwvEjPdzYfXz/5gDHj2qowjJLzGmvYtSG9hByYqUVdo8uW8ZzABJQNZ8RD49Aoc/u9ohEMF0zqDMBkxPWZNgzWL30dPDb8wVrWFvCVIEIX2fMrEcFw8xAtpY0wqhKIKmU2zzD1p2HBgZQF5Tyx6WAq6FVF9JoWONg3KGk7/TPu5lIVOXBKDr6hdXPVZTkq6JqgQrcRoSSIXxa3o9IjbBcGeWSkQgq9kMUk4NkASDsCRDrv4M/sgQZHVEkcVCHLXxxN0DMGZdvHKk5y3f8KiVxZYgV/Krnej6akyQIE0TpHrS9r+X0IyeF+H9PcbwTJjpBnUlhbxCI+kYWrMVwA+NBnFJCdwkjYVHRq0IOPwNtnk9ghGDSvq4yGCiiUxJh/uOqNzN6YxsprpyHDzG+FcXUi72ZBEJ3HbNV/rbWyhWWe0f8l4Iiy8PdXK1bb9yImQ3H97DK+e6/F2HGVQ2VNkbLvCtJKP4/+5rJ/XxqJwdNvsj/NM5gbdptlMPRN0pUg9oZ64KQFGbvT3h1Bc+YBGmA6FnR+qB0p9iBv4u/Of/ZGAzWUBX4AfC2Fg9V17qtgTjaOZZvAvf6CYTTjfTHFrVn6LvL2W+nomKBOuMt2oX+hfJfo2vACkKGgdi19Ww6KSeFi7zNVwhXLOuUwvWCmbiFy4sdvVGwsV1XjjvInymZeZVjGzvHNVzsUlKTkC3/sDUtiJOCIqAEbw7Y5hwEvrfjw/dIyXvMR8NONnhcP/7PyvmNTiMZdtYD7glifF+G+UnAYKUwxOBbVQLOYjqv1BoOPdUhvT8Ds1p8kVZuzYVrqgpYKAbGjD3KwrJAdB1pm+Yi9vfwzfGBoGhr0KCpWcdIA+nnYcSj+3Q/8TBj+2qrK3lBQ7T28Iw3C8fM6TL3KgPG3qfZ43MXmGMjMkqzS6wQp9CPXSrd7kE89Mhpv/KcIN0gfwBL7mGhxYtgUb8PIhHDUDUjXXLvzUNKDtJtXLXZY8PqP7bqRrzs89o1qK96rT6+pGRQQ8bqdmy8FmPEt1D+48bP2r5SbOGEbq4UaiNE0cGBpBuxMD5AtkzuwlSzZKyVD3u7cQEfbSebgWFY9hdPpAmoJ3PiQiwA8/rbmrk1/tYKPyTDMoS/MOW7z5ceMbsWQbyhs+Mv3ADfKg4NsDMHbAMk5NRy1a2ZnUl5TfRUs6z84RHhICP2btCMyvuUCD/XBlKHWiXuFJLIlMR5BZVjDVuJ1FdY3WVc/QyfrfdaqevSFxWQ4n6UoAr5nX0MgHBvbyOtA4G+Mhzz6IJzN7bE4H7PY44zjBGdBpQ3B5cO6hPkzYNMGXjQh2AWA7UldiX8sqhFFIh3Gfp20IcTnKI8s1RAD4NtGrhHGUCmwS4JaTBh1VUNJcMPfXkT4w9ehgY+dHTi2wlo3tkvzNcFkb2vOf3g3nZEWhHSHLRvI7tUjukZhGWMYqyjEL0D7/qn+OyaYEbnu3+9zEPLfkOnWQMSpNK1tJzqi5QkTmHYb2tqB4nlLPp0iCcqkldxSZRdKHM53kBw0jl6yuHDeeU6eY9ehdMtwnXkI0wiBbJPQQm7MuOfQKIMLrIDzlBrhymQtK8XQebnQPbQAsV3qqbK2qV+xm8cR1K9YF3F0Jt9h5ERZBWJYWqyxcQOm+ELYgxa/rXLQiPHhDqhlT5w86gykzuaUSMA0T7A6dm3TKZUKq+ipVgWUmq6kfOXUWVmODyuqB02lbK1ILMQkV/Ehq4ZNlHvWKNtJuuCyLzI3eMWcjNpjaJxzvTXn8ctPPSeldUzZv2WmpCYTl9zp0klDqSpNNqUnlMI5Eww3KkX9LYmz8qyiAwZ2aVQLw8k6nOZDTwMU1F1OVCPU3J2n2NZwG1n8X+WEa0XfUPsLT7XmaHmHrS+n2SgBzUlg9qYCXrtUqLz7iHIMzqzW38wcVH5Ud5exuo/7I8t5ugJvRZ5vfBGWu6rcb3LkAzOM4CkWpWL2eCvWmw8pdhgMk9oA+a3+UEKvQbGU1z/z8LUnKFk13XqnNu+0uE4SGkUrhMp+KbBsYG0AU8rkqsRoSSGdmdhD4k85/tUj44+jpUK6m4cd3XPydaFJDLWq6zFvrTm7Tbw3MqZ7xGsiPz2q7a0ZTIZbgdI6ubKKKh9MNfCOpm6f5Zgd7mLZr3Y3Tn8RcYtVMS5QF+2ezd9WrlP7dNyX6awU3oOF3k8eZlU+ygzMS8aQtW3nuD+pGZBNwwm1fI/lEYPXq8mJ79MeWf3Xcxd+8Tfx5VBOtnvpKNC0WpvsAyj
```

PRCKPRvue1GI1GCCFua7utr8/cnM+/avf6nIU9+kx1w17Lg32WB/ssH0f0uj7balC4AQCDwYCNGzciPT0dkiQhJiY
GDz/8MNRqdb22I0mSxWMhhNUyW1auXIkWLVpg+PDhNY5LTk62ef+d3NxclJXZ97wWo9GIwsJCCCGgUjVoUgwt3are
vJOZV5CTk2PP8pyGPfpMdcNey4N9lgf7LB9H9Lq4uLjOYxsUbk6dOoXBgwfj4sWLaN++PYQQOHHiBMLDw7Fp0yZER
0fXuo2AgACo1WqrWZqcnByr2ZxbCSGwYsUKjBkzBm5ubjWOnTVrlsXl6UVFRQgPD0dgYCB8fHxqrbM+jEYjJElCYG
Bgg9/MThEC2PUnsq8aodPp7Fqfs7BHn6lu2Gt5sM/yYJ/l44hea7XaOo9tULh5/vnnER0djZ9//hl+fn4AgPz8fDz
++ON4/vnnsWnTplq34ebmhtjYWKSmpuKRRx4xL09NTcXDDz9c47o7d+7EqVOnONMGHChFr3o9FooNForJarVCqHfLgl
SbqtbUf4ewKoOqGYf/iqd7t9prpjr+XBPsuDfZaPvXtdn+00KNzs3LnTItgAgL+/P9544w306tWrzttJSkrCmDFjE
BcXh/j4eHzwwQfIyMjApEmTAFTNuly8eBGrVq2yWG/58uXo0aMHOnfu3JDyGzXTCcUFpRUovFYBX3dXhSsiIiJqWh
oUbjQajc1jXyUlJbUeJrrZqFGjkJ+fj/nz5yMzMxOdO3dGSkqK+eqnzMxMq3veFBYWYv369Xj33XcbUnqj56VxQYC
XG/JKynHhcil8w3yVLomIiKhJaVC4GTp0KJ566iksX74c3bt3BwDs3BwD88ssvmDRpEh566KF6bWvy5MmYPHmyzedWrlxp
tczX19d8A0FnFe7ngbyScmRcLkVnhhsiIqJ6adCBsPfeew/R0dGIj4+HVquFVqtFQkIC2rZti4ULF9q5xOYn4vqhq
fO81w0REVG9NWjmpkWLFvj6669x6t6tQppKenQwiBjh07om3btvaur1lqbbqR32WGGyIiovqqc7ip7du+d+zYYf79gg
ULGlwQAa1NV0wx3BAREdVbncNNWlpancbV5QZ8VDPTzM35y1cVroSIiKjpqXO42b59uyProJtE+FeFm4tXrqG80gg
3F96PgYiIqK74r2YjpPPWwNNNDaMAMjh7Q0REVC8MN42QJEmICqw67+Z0LsMNERFRfTDcNFJtArwAAGcYboiIiOqF
4aaRanN95uZMbonClRARETUtDDeNVJvA6zM3eZy5ISIiqg+Gm0aqTUDVzM1ZhhsiIqJ6YbhppEyHpS5fLUdBabnC1
RARETUdDDeNlIebC0J8tQB4xRQREVF9MNw0YjypmIiIqP4YbhqxqOvn3fCkYiIiorpjuGnEbtzrhjM3REREdcVw04
hF66rCzakchhsiIqK6YrhpxNoFVYWbc/ml0FcaFK6GiIioaWC4acSCfbTw1rjYBS83w0REVEdMdw0YpIk4Y7rszc
nsnloioiIqC4Ybhq5dkHeAICT2cUKV0JERNQ0MNw0cndcDzcnGG6IiIjqhOGmkTOdVHySh6WIiIjqhOGmkTMdljqX
fxVlFbxiioiIqDYMN42czlsDH60LjAI4w++YIiIiqhXDTSMnSdKNk4pzeN4NERFRbRhumgDTScXHsxhuiIiIasNw0
wTEhFSFm/TMIoUrISIiavwYbpqATqE+AICjmcUwGIXC1RARETV+DDdNQFSAJ7SuKlzTG3A+/6rS5RARETV6DDdN
gFolIfr6eTdHeN4NERFRrRhumohOIaaTiosVroSIiKjxY7hpImKuh5ujmZy5ITqqIjvrajM2n+ZjSfb/HC8PTFbHbJV
b9vwzKLxtS5Gsy2j6iRrXLoP2+zRXx+iwGhqp7aPS8NyhzWG0S2LxWvlGnHYf+zYp6AG6RIg13YU1n4VXwazHkySXaVh
QAHL1UqHA1REREjRvDTRPRKdQXAA9NERER1YbhpokwnXdzjOGGiIioRgw3TcSNk4p5p5WIqIiKgmDDdNhOmw1Ln8UhS
XVShcDRERUePFcNNE+Hm6IcRXCwBIz+TN/IiIiKrDcNOE8NAUERFR7RhumpCOvGKKiIioVgw3TUhn3qmYiIioVgw3
TUinsKqZm5PZxdBXGhSuhoiIqHFiuGlCQn21aOSZjU7bOHhikqjwMnsEqXLISIiapQYbpoQSZJw5/XZm9/+LFC2GCIiokaK4
aaJuSu8BQDgUEaBonUhOmEqXLISIiapQYbpoQSZJw5/XZm9/+LFC2GCIiokaK4
aaJuSu8BQDgUEaBonUQERE1Vgw3TYwp3KRdKFC0DiIiosaK4aaJMYWbUzklKLzGOxUTERHdiuGmifH30qC1nwcA4H
eed0NERGRF8XCzZMkSREVFQavVIjY2Frt27apxvF6vx5w5cxAREQGNRoPo6GisWLFCpmobh26tWwDgeTdERES2uCi
583Xr1mHq1KlYsmQJevXqhf/85z9ITEzEsWPH0Lp1a5vrjBw5EtnZ2Vi+fDnatm2LnJwcVFZWyly5su4Kb4GvD13i
eTdEREQ2KBpuFixYgAkTJmDixIkAgIULF+J///J//sfli5diuTkZKvxmzdvxs6dO3HmzBn4+fkBACIjI+UsuVHo1rolA
ODQhQIIISBJksIVERERNR6KHZYqLy/HwYMHMWjQIIvlgwYNp49e2yu88033yAuLg5vvvfUWwsLC0K5dO7z44ou4du
2aHCU3Gh1DfODmosKVUnovL8IOVFFGSxPCgoCFlZWTbXOXPmDHbv3g2tVouvvvKeXl
5mDx5Mi5fvlzteTd6vR56vd78uKio6nuZjEYjEajnV4NzNsUQth9u7dyUQF3tfLFvnNX8MuZfET6ezh0f42NXH0m
9lou7LM82Gf5OKLX9dmWooelAFgdUqnpMIvRaIQkSVizZg18favu1LtgwQKMGDECixcvhru7u9U6ycnJmDdvntXy3
NxclJWV2eEVWNZXWFgIIQRUKsdOinXSabDvHLAz/RL6RWgcuq/GRs4+N3fstTzYZ3mwz/JxRK+Li4vrPFaxcBMQEA
C1Wm01S5OTk2M1m2MSEhKCsLAwc7ABgJiYGgh8OeDgCAwPh4+Njp1dTxRS
+AgMDHf4Hp19nFT7al4Xfs0qh0+kcuq/GRs4+N3fstTzYZ3mwz/JxRK+1Wm2dxyoWbtzc3BAbG4vU1FQ88sgj5uWp
qal4+OGHba7Tq1cvfPHFFygpKYGXlxcA4MSJE1CpVGjVqpXNdTQaDTQa65kNlUrlkA+3JEkO2/bN4iL94KKmgD
BcLyhDu17wOTcnVZ2Kv5cI+y4N9lo+9e12f7Sj67iYlJeHDDz/EihUrkJ6ejmnTpiEjIwOTJk0CUDXr8sQTT5jHjx
49Gv7+/njyySdx7Ngx/Pjjj5gxYwbGjx9v9v85CUM/Nwc8GdrapmsH45e1nhaoiIiBoPRc+5GTVqFPLz8zF//nxkZma
ic+fOSElJQUREBAgMzMTGRkZ5vFeXl5ITU3Fc889h7i4OPj7+2PkyF4/fXXlXoJiuoR5Y+0jAL8ciYfI2Jtz1wR
ERE1N4qfUDx58mRMnjzZ5nMrV60WtahQweUnIZ+fxSZBXa9
8ovIiKiporhpgnz1rqiU6jpvJt8hashIiJqHBhumrgeUVfQ/HzGR6aIiIAhhumrwebfwBAD+f4cwNERERwHDT5P
VoU3W/m7N5V3HhcqnS5RARESmO4aaJ89G64u7xK+80SuwtUQEREpj+HGCfRpHHwiA4YaIiAhguHEK991k
HhYHfdktERM0bw40T6BTqA39QeRKRKQohhsnoFJJ6N3uz7Q0ogmUFzj1MQJL6H1HAAAemiIiImK4cRJ9OgQCAHaf
ygArusv6+fdHLlYhNxivcLVEBEEBERKYfhhpdsnoFJJ6N3uz7Q0ogmUFzj1MQJL6H1HAAAemiIiImK4cRL3tas6NPXjSYYbIiJq
3hhunETv6+fdHLlYhNxivcLVEBERKYfhxkkEemvQKdQHALD7FGdviIio+WK4cSJ9r18S/kN6jsKVEBERKYfhxokMj
AkCAOw4ngt9pUHhaoiIiJTBcONEurZqAZ23BiX6Sn6RJhERNVsMN05EpZIwsGPV7EcqsSyFqyEiIlIGw42Tud8cbr
JhNAqFqyEiIpIfw42TSYj2h6ebGtl
Fevx+sVDpcoiIiGTHcONkNC5q9OugAwB899slhashIiKSH8ONE3r4rjAAwLe
/X4KBh6aIiKiZYbhxQn3aBcLX3RXRXr8ciZf6XKIiIhkxXDjhNxcVBh8ZzA40tDPDRFRETNC8ONkzIdmko5kskb
+hERUbPCcOOkukf6IdhHi+KySuw4zu+aIiKi5oPhxkmpVBIeuisUAPD1oYsKV0NERCQfhhsn9lDsdXQCzNT0HdcqF
K6GiIhIHgw3TqxTA/aBXmhvNKI737nicVERNQ8MNw4MUmS8H87agBs+VLgaIiKieTDcCOLfYcA5Aw5wf+VLgaIiIieTDcOLnh3cLgopLw24UCnM
wuVrocIiIih2O4cXKB3hrz1zF8cZCzN0RE5PwYbpqBv8W2AgBs+PVPViBiMCldDRETkWAw3zUC/DjoEeRrGsMKj
hIiKnx3DTDLiqVXKikW9Udi9ftv6BwNURERI7FcNNMjqLqn6qqpbX9k41LBNYWrISIichyGm2airc4bR9/BoAn1PN0/WCNdQHB41v5Qxx4tjsn8S8zgUi3kr9FSaklcAEfLTJ3kI8/UIIDwJ85/RoKnEp7dEmYkd+fWYfI1ZN8ClJuHBEeFJZJRVQdVpfPn4qHGgKdYtz9tqcDBuSQKNPI9VIh/Xj6OmSqp8y8C0Lb6rXDa9jQ8uabd+Sb9j
hgiIiIHYbhpZh7t3hoqCfj5zGXesZiIiJwSw00zE9rCHQI/aeV7gaIiIi+2O4aYbGJUQCANb/+IcKr1UoWw
wREZGdMdw0Q/HR/mgf5I3ScgM+P8CF6oiIyATDcNEOSJGH8vVRrMrvKbvllyNMb...

UGSEEIoXYScioqK4Ovri8LCQoecUJyTkwOdTtdkj+eezi3BwAU7IQSwNek+tNV5K12SFWfoc1PBXsuDfZYH+ywfR/
S6Pv9+890lC9GBXrg/puqE7iXbTytcDRERUf0x3JCVKf3bAgC+OnQRJ7Lrfl8BIiKixoDhhqx0adUCD3QKghDAgi0
nlC6HiIioXhhuyKbpg9pDkoDNR7Pw+58FSpdDRERUZww3ZFO7IG8MvysMAPA2Z2+IiKgJYbihak0deAdcVBJ+PJGL
PafylC6HiIioThhuqFoR/p4Y3aM1AOCVb46iwsBv0iUiosaP4YZqNP3+9vDzdMOpnBJ89NNZpcshIiKqFcMN1cjXw
xUzEzsAABZuPYnMwmsKV0RERFQzhhuq1Yi7WyE2oiVKyw14fVO60uUQERHViOGGaqVSSZj/cCeoJGDT75n4IT1b6Z
KIiIiqxXBDddIp1BcTe7cBAMz56giKyyoUroiIiMg2hhuqs2kD2yHS3wNZRWV44/s/lC6HiIjIJoYbqjN3NzWS/9I
FALDmlwzsPZ2vcEVERETWGG6oXuKj/c33vpnx5W8o4uEpIiJqZBhuqN5mJXZAuJ87/rxyDXO+OgIhhNIlERERmTHc
UL15a13x7qPdoFZJ+Pa3S1j/60WlSyIiIjJjuKEGubt1S0wbeAcA4JWvj+BMbonCFREREVVhuKEGe6ZvW/SI8kNpu
QETPj6AgtJypUsiIiJiuKGGU6skLBrdDWEt3HE27yqeXn0Q5ZX8ck0iIlIWww3dFp23FsvHxcFL44Jfzl7GzA2/8w
RjIiJSFMMN3bYOwT54f3Q3qCRgw68Xkfz9Hww4RESkGIYbsou+7XVI/sudAIAPfjyD97edUrgiIiJqrhhuyG5G3dM
a/xoSAwB4J/UElu8+q3BFRETUHDHckF1N7N0GU69fIv7ad8ewbOdphSsiIqLmhuGG7O6FAXfguf5tAQBvfP8HFmw5
znNwiIhINgw3ZHeSJGH6oPZ46cH2AID3tp3C/21KZ8AhIiJZMNyQw0zu2xZh3UEAHy4+yymf/Eb74NDREQOx3BDD
jWuVxTe+msXqFUSNvx6EWNX7ENhKb9JnIiIHIfhhhxu5D3hWD42Dp5uauw9k4+/LP0JFy6XKl0WERE5KYYbkkXf9j
p8MSkBwT5anM69iocX/4QtR7OULouIiJwQww3JpmOoDzY+2wudw3xw+Wo5nlp9EP/88neU6CuVLo2IiJwIww3JKth
Xi/XPJODp+9pAkoB1By5gwDs78MWBCzAYeTUVERHdPoYbkp3GRY1Zg2Pw2T96ItzPHdlFesz48ncMeGcHVu09h9Jy
zuQQEVHDMdyQYnq28UfqtD6YldgBvu6uOJdfile+PoqEN7bhrc1/8KRjIiJqEIYbUpTWVY2n+0Rj76z+mPdQJ7T28
0BBaQWW7DiN+/6/7Ri7Yh82H8lChYH3xyEiorpxUboAIgDwcHPB2IRIPN4zAqnHsvDJzxnYfSoPO0/kYueJXAR6az
AyrhX+cncrRPl7KF0uERE1Ygw31KioVRIe7ByCBzuH4FzeVazdfwFfHryA3GI9Fm8/jcXbTyMmxBv92vjgbz09ERX
orXTJRETUyDDcUKMVGeCJmYkdkHR/O2xNz8a6/Rfw06k8pGcWIz2zGEt+uojWfh7o1dYfcRF+iAnxQbTOExoXtdKl
ExGRghhuqNFzc1Fh8J0hGHxnCK5cLcf3RzKx4cB5HLpYgozLpcjYV4rP9l0AALioJLQJ9ERrP0+0aumOsBbuCPbVw
s/TDS093NDS0xUtPdygdWUAIiJyVgw31KS09HTDo/eEo3+EBh6+fjhw/gp+OpWPwxcL8UdmEYrKKnEiuwQnsktq3I
67qxqeGhd4atTwcHOBp5sa7m5qeLq5QOuqgsZFDTcXFdxcVNBc/7Xq91XLNWoVXNQS1CoJKqnqR63C9V8lqFQS1Ne
Xq1SAurrlpsc3LZckQKWSoJIACdd/lap+Ne1Lur4vCbixzk1jpOvrEBE1Rww31GR5aVzQv0MQ+ncIAgAIIXCpsAwn
sorxZ8E1XLxyDRcLriG7qAwFpeW4fLUCBaXlqDQKXKsw4FqFAXk1Z6AmzyIQ3RSAJJuByPSc5TrCaICLiwvUkgRUs
45pGW55LN0SuCz2g+t1XK+1KotJ5t/fvFyCBFNWMz3G9eE3xkm3rGO9HDetK5lrMD1luY+aarG1D1S3rZuXV1MLIH
CttBQeHlegUknW+zBtv477MO+ijq/X4jVYjLF8bTdqv7XX1dRy0xtksS2Lem8ae9NjW3XdOv7mfVque+v7YSJQUFC
IlvkCkqSqtoZbP1fV7d/mPmp53vRnrz5117WGGuurta6611D7ewZIEFByfpzhhpyGJEkIa1F1KKo6QggU6ytRcLUC
V8srUVpeiat6A0rLK1FabsBVfSXKKowoNxihrzSivNIIfaUB5ebf31hWaRQwCgGDUcAoAKNRwCCE+VeDsWp/hluWG
424ab3rz1/fhhDXtyUExPVfjUJAABANuIGzaVvA7d79ufw21yei5kTnrcE3Ezortn+GG2pWJEmCj9YVPlpXpUupN2
EReKoJQMYbj63CEqoC2M3r3DxG4EbwMo0xGI3Iz78M35YtIUkSjDZCmGn8zevXZYzpcdVruxG/qlsOISzHXE97wmr
9G0Gurvu4tcc3b7fq95bLccu+bY25eTkslluPExAovVoKd3d383+Db319Ne0DNy+vw2uGxfJb+nrLvmFr3zXsAzaX
W+/j5tcIi/Wt3xtby2966bbXuWUfptLKKyrg6uoCYZqTuPl5G6+tpm3WWnc1NVT7XB16gBrWq37/5rVqr89in5bLb
/3c1FS3xkXZ2+gx3BA1EeZDSZBqH2wnRqMROZpy6HQtoVLxnp+OYjQakZOTA51Oxz47EPssH1OvlcJ3l4iIiJyK4u
FmyZIliIqKglarRWxsLHbt2lWn9X766Se4uLjgrrvucmyBRERE1KQoGm7WrVuHqVOnYs6cOUhLS0Pv3r2RmJiIjIy
MGtcrLCzEE088gQEDBshUKRERETUVioabBQsWYMKECZg4cSJiYmKwcOFChIeHY+nSpTWu9/TTT2P06OGIj4+XqVIi
IiJqKhQ7obi8vBwHx7EzJkzLZpPDWaYGEjQIe+bsmyPbqv2WzDAcWaS1sWrWaa9jz76CKdn8yn0/PGWcknxtVm/
aZpo9FYdYWInbdLlthn+bDX8mCf5cE+y8cRva7PthQLN3l5eTAY75GAgKCrJYHhQXh8uXLNtc5ceIEZsyYgV27dsHFpW7Tnjt3LmbPni1L/SIiImr4FD+hWJIki9dCCKs+ADAYDBg9ejRmz56Ndu3a1Xn7s2bNQmFhofnPxYsXb7lmIiIiargUO3Lj7+8PtVptdZQmJyfH6mgOABQXF2P//v1IS0vDlClTAABGoxFCCLi4uGDLli3o37+/1XoajQYajcYxkyAiIqIGR7EjN25uboiLi0NqaqpFe2pqKu6++26r/r6+vjh8+DAOHTpk/pOcnIz27dvj0KFDuM+ZKCxERUYMLSnPXAALCmsb//rdOtI7kxMqrzXPZiA/FGb/7Dn1j6uWWmZGaXpFUDXxbjD9GgcvPG+QBBSoT9gkRuAf8h0PMKCAQEHHn3GHCsItKvYGcUZabcM+hmqlfTYn0cOzbhkfqGZvPf7GkWMEHeQQ5ktd8kK5M0Ju8v/h+vanz2Rul/3ImMK8WmmKtU9Ay3aHaKvBWHRbThDA0Fn2MEFd+vn5drfn+T8sn7z9Mmr

99/x+7du62eY5/to3379jh06BAKCgqwfv16jB07Fjt37jQ/zz7fvgsXLuCFF17Ali1boNVqqx3HXt++xMRE8+/vvP
NOxMfHIzo6Gh9//DF69uwJoPH0mYelblNAQADUarVV6szJybFKsNRwpjPya+pzcHAwysvLceXKlWrHUJXnnnsO33z
zDbZv345WrVqZl7PP9uXm5oa2bdsiLi4Oyc nJ6Nq1K95991322Y4OHjyInJwcxMbGwsXFBS4uLti5cyfee+89uLi4
mHvFXtufp6cn7rzzTpw8ebLRfaYZbm6Tm5sbYmNjkZqaarE8NTUVCQkJClXlfKKiohAcHGzR5fLr/LycuzcudPc59jYW
Li6ulqMyczMxJEjR/heXCeEwJQpU7Bhwwzs27/Ymx/uqYu6y+4CM8OleIA8FwIXfUOnXjN3y/ks6P4Br3KhDS3FmG9tpFtnfdxAYvp0+ajbqwnvh1qKQR43BzWuaMjzoi+TZn8ei5QJDqVEKz5Uz3L8xU4jXuuJVg47DvRmpEUT/JTT9jJ/Jyqza+Ft5UMCTQW7f+TxsyfUZU+NS8a/BH3Q8Xb/oxK+9/w+e8n+zeHV8L+rVMMY2qV5U2k+6AZ/gpjJW4nLYGmDL16i2Cp9sjntaLbjZV2cv/fJH8aXn4sduyTPTVjEFgt0h6F1fFj7Vx6aXkNiEqVsfr9wAaLA5TDFg5tZF8EOqzbf1ffBg9ogSGk6+gLyE7q6pivH/yaZW5sMtxf

```
zllaWmoewz7bx6xZs8SPP/4ozp49K37//Xcxe/ZsoVKpxJYtW4QQ7LMj3Xy1lBDstb1Mnz5d7NixQ5w5c0b8/PPPY
ujQocLb29v8b11j6jPDjZ0sXrxYRERECDc3N3H33XebL62lutu+fbsAYPUzduxYIUTVpYavvvqqCA4OFhqNRtx333
3i8OHDFtu4du2amDJlivDz8xPu7u5i6NChIiMjQ4FX0zjZ6i8A8dFHH5nHsM/2MX78ePPfCYGBgWLAgAHmYCME++x
It4Yb9to+TPetcXV1FaGhoeIvf/mLOHr0qPn5xtRnSQgh7DsXRERERKQcnnNDREREToXhhoiIiJwKww0RERE5FYYb
IiIicioMN0REROORUGG6IiIjIqTDcEBERkVNhuCGiZm/Hjh2QJAkFBQVKl0JEdsBwQ0RERE6F4YaIiIicCsMNESlOC
IG33noLbdq0gbu7O7p27Yovv/wSwI1DRps2bULXrl2h1WrRo0cPHD582GIb69evR6dOnaDRaBAZGYl33nnH4nm9Xo
+XXnoJ4eHh0Gg0uOOOO7B8+XKLMQcPHkRcXBw8PDyQkJCA48ePO/aFE5FDMNwQkeL+9a9/4aOPPsLSpUtx9OhRTJs
2DY8//jh27txpHjNjxgy8/fbb2L9/P3Q6HR566CFUVFQAqAolI0eOxKOPorDhw9j7ty5ePnll7Fy5Urz+k888QTW
rl2L9957D+np6Vi2bBm8vLws6pgzZw7eeecdHDhwAC4uLhg/frwsr5+I7ItfnElEirp69SoCAgKwbds2xMfHm5dPn
DgRpaWle0qpp9CvXz+sXbsWo0aNAgBcvnwZrVq1wsqVKzFy5Ej8/e9/R25uLrZs2WJe/6WXXXsKmTZtw9OhRnDhxAu
3bt0dqaioGDhxoVcOOHTvQr18/bN26FQMGDAAApKSkYMiQIbh27Rq0Wq2Du0BE9sSZGyJS1LFjx1BWVob7778fXl5
e5p9Vq1bh9OnT5nE3Bx8/Pz+0b98e6enpAID09HT06tXLYru9evXCyZMnYTAYc0jQIajVavTp06fGWrp06WL+fUhI
CAAgJyfntl8jEcnLRekCiKh5MxqNAIBNmzYhLCzM4jmNRmMRcH4hLCzM4jmNRmMRcG4lSRKAqnN2TL83uXlS2t3dvU61uLq6Wm3bVB8RN
R2cuSEiRXXs2BEajQYZGRlo27atxU94eLh53M8//2z+/ZUrV3DixAl06NDBvI3du3dbbHfPnj1o164d1Go17rzzTh
iNRotzeIjIeXHmhogU5e3tjRdffBHTpk2D0WjEvffei6KiIuzZswdeXl6IiIgAAMyfPx/+/v4ICgrCnDlzEBAQgOH
DhwMApk+fjnvuuQevvfYaRo0ahb179+L999/HkiVLAACRkZEYO3Ysxo8fj/feew9du3bF+fPnkZOTg5EjRyr10onI
QRhuiEhxr732GnQ6HZKTk3HmzBm0aNEC  d999N2bPnm0+LPTGG2/ghRdewMmTJ9G1a1ds2LABbm5uAID        bjEZkbF+TmWbTz3mNjdzMkxNWQBoDxkpfHtl+a8bl++a99/9kt6K1Kg7pYoA7b7jRsMPl+DRVbXk5Em8EEaM4G+
```

```
```

```
   "version": "3.12.7"
  }
 },
 "nbformat": 4,
 "nbformat_minor": 5
}
```