```
{
  "cells": [
    {
      "cell_type": "markdown",
      "source": [
        "#Assignment_2"
      ],
      "metadata": {
        "id": "T0hdBrFkczpO"
      },
      "id": "T0hdBrFkczpO"
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "id": "ca866dae",
      "metadata": {
        "id": "ca866dae"
      },
      "outputs": [],
      "source": [
        "import pandas as pd\n",
        "import numpy as np\n",
        "import plotly.express as px"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "id": "de81505e",
      "metadata": {
        "id": "de81505e"
      },
      "outputs": [],
      "source": [
        "df=pd.read_csv(\"/content/Churn_Modelling.csv\")"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "id": "33350f68",
      "metadata": {
        "id": "33350f68",
        "outputId": "8d9ec9a9-7e66-4675-dc77-67ecf87c6730",
        "colab": {
          "base_uri": "https://localhost:8080/"
        }
      },
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "RowNumber          False\n",
              "CustomerId         False\n",
              "Surname            False\n",
```

```
                "CreditScore             False\n",
                "Geography               False\n",
                "Gender                  False\n",
                "Age                     False\n",
                "Tenure                  False\n",
                "Balance                 False\n",
                "NumOfProducts           False\n",
                "HasCrCard               False\n",
                "IsActiveMember          False\n",
                "EstimatedSalary         False\n",
                "Exited                  False\n",
                "dtype: bool"
              ]
            },
            "metadata": {},
            "execution_count": 3
          }
        ],
        "source": [
          "df.isnull().any()"
        ]
      },
      {
        "cell_type": "code",
        "execution_count": null,
        "id": "10779451",
        "metadata": {
          "id": "10779451",
          "outputId": "92d54998-58b7-4de2-d20c-6047d69925ed",
          "colab": {
            "base_uri": "https://localhost:8080/"
          }
        },
        "outputs": [
          {
            "output_type": "execute_result",
            "data": {
              "text/plain": [
                "RowNumber                0\n",
                "CustomerId               0\n",
                "Surname                  0\n",
                "CreditScore              0\n",
                "Geography                0\n",
                "Gender                   0\n",
                "Age                      0\n",
                "Tenure                   0\n",
                "Balance                  0\n",
                "NumOfProducts            0\n",
                "HasCrCard                0\n",
                "IsActiveMember           0\n",
                "EstimatedSalary          0\n",
                "Exited                   0\n",
                "dtype: int64"
              ]
            },
            "metadata": {},
            "execution_count": 4
```

```
      }
    ],
    "source": [
      "df.isnull().sum()"
    ]
  },
  {
    "cell_type": "code",
    "execution_count": null,
    "id": "36bcf6e9",
    "metadata": {
      "id": "36bcf6e9",
      "outputId": "369c7f26-cbe2-493c-d924-b53f8cd7d323",
      "colab": {
        "base_uri": "https://localhost:8080/",
        "height": 145
      }
    },
    "outputs": [
      {
        "output_type": "execute_result",
        "data": {
          "text/plain": [
            "   RowNumber  CustomerId   Surname  CreditScore Geography  Gender  Age  \\\n",
            "0          1    15634602  Hargrave          619    France  Female   42  \n",
            "\n",
            "   Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  EstimatedSalary  \\\n",
            "0       2       0.0              1          1               1        101348.88  \n",
            "\n",
            "   Exited  \n",
            "0       1  "
          ],
          "text/html": [
            "\n",
            "  <div id=\"df-ab6bbd35-ffb7-4563-80c3-f276e4f81ae9\">\n",
            "    <div class=\"colab-df-container\">\n",
            "      <div>\n",
            "<style scoped>\n",
            "    .dataframe tbody tr th:only-of-type {\n",
            "        vertical-align: middle;\n",
            "    }\n",
            "\n",
            "    .dataframe tbody tr th {\n",
            "        vertical-align: top;\n",
            "    }\n",
            "\n",
            "    .dataframe thead th {\n",
            "        text-align: right;\n",
            "    }\n",
            "</style>\n",
            "<table border=\"1\" class=\"dataframe\">\n",
            "  <thead>\n",
            "    <tr style=\"text-align: right;\">\n",
```

```
                "      <th></th>\n",
                "      <th>RowNumber</th>\n",
                "      <th>CustomerId</th>\n",
                "      <th>Surname</th>\n",
                "      <th>CreditScore</th>\n",
                "      <th>Geography</th>\n",
                "      <th>Gender</th>\n",
                "      <th>Age</th>\n",
                "      <th>Tenure</th>\n",
                "      <th>Balance</th>\n",
                "      <th>NumOfProducts</th>\n",
                "      <th>HasCrCard</th>\n",
                "      <th>IsActiveMember</th>\n",
                "      <th>EstimatedSalary</th>\n",
                "      <th>Exited</th>\n",
                "    </tr>\n",
                "  </thead>\n",
                "  <tbody>\n",
                "    <tr>\n",
                "      <th>0</th>\n",
                "      <td>1</td>\n",
                "      <td>15634602</td>\n",
                "      <td>Hargrave</td>\n",
                "      <td>619</td>\n",
                "      <td>France</td>\n",
                "      <td>Female</td>\n",
                "      <td>42</td>\n",
                "      <td>2</td>\n",
                "      <td>0.0</td>\n",
                "      <td>1</td>\n",
                "      <td>1</td>\n",
                "      <td>1</td>\n",
                "      <td>101348.88</td>\n",
                "      <td>1</td>\n",
                "    </tr>\n",
                "  </tbody>\n",
                "</table>\n",
                "</div>\n",
                "      <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-
ab6bbd35-ffb7-4563-80c3-f276e4f81ae9')\"\n",
                "              title=\"Convert this dataframe to an interactive table.\"\n",
                "              style=\"display:none;\">\n",
                "        \n",
                "  <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0
0 24 24\"\n",
                "       width=\"24px\">\n",
                "    <path d=\"M0 0h24v24H0V0z\" fill=\"none\"/>\n",
                "    <path d=\"M18.56 5.44l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94
2.06-2.06.94zm-11 1L8.5 8.5l.94-2.06 2.06-.94-2.06-.94L8.5 2.5l-.94 2.06-2.06.94zm10 10l.94
2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94 2.06-2.06.94z\"/><path d=\"M17.41 7.96l-1.37-
1.37c-.4-.4-.92-.59-1.43-.59-.52 0-1.04.2-1.43.59L10.3 9.45l-7.72 7.72c-.78.78-.78 2.05 0
2.83L4 21.41c.39.39.9.59 1.41.59.51 0 1.02-.2 1.41-.59l7.78-7.78 2.81-2.81c.8-.78.8-2.07 0-
2.86zM5.41 20L4 18.59l7.72-7.72 1.47 1.35L5.41 20z\"/>\n",
                "  </svg>\n",
                "      </button>\n",
                "      \n",
                "  <style>\n",
```

```
                "       .colab-df-container {\n",
                "         display:flex;\n",
                "         flex-wrap:wrap;\n",
                "         gap: 12px;\n",
                "       }\n",
                "\n",
                "       .colab-df-convert {\n",
                "         background-color: #E8F0FE;\n",
                "         border: none;\n",
                "         border-radius: 50%;\n",
                "         cursor: pointer;\n",
                "         display: none;\n",
                "         fill: #1967D2;\n",
                "         height: 32px;\n",
                "         padding: 0 0 0 0;\n",
                "         width: 32px;\n",
                "       }\n",
                "\n",
                "       .colab-df-convert:hover {\n",
                "         background-color: #E2EBFA;\n",
                "             box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n",
                "         fill: #174EA6;\n",
                "       }\n",
                "\n",
                "       [theme=dark] .colab-df-convert {\n",
                "         background-color: #3B4455;\n",
                "         fill: #D2E3FC;\n",
                "       }\n",
                "\n",
                "       [theme=dark] .colab-df-convert:hover {\n",
                "         background-color: #434B5C;\n",
                "         box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",
                "         filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",
                "         fill: #FFFFFF;\n",
                "       }\n",
                "   </style>\n",
                "\n",
                "         <script>\n",
                "           const buttonEl =\n",
                "                             document.querySelector('#df-ab6bbd35-ffb7-4563-80c3-
f276e4f81ae9 button.colab-df-convert');\n",
                "           buttonEl.style.display =\n",
                "             google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
                "\n",
                "           async function convertToInteractive(key) {\n",
                "                 const element = document.querySelector('#df-ab6bbd35-ffb7-
4563-80c3-f276e4f81ae9');\n",
                "             const dataTable =\n",
                "                                                                    await
google.colab.kernel.invokeFunction('convertToInteractive',\n",
                "                                                                [key], {});\n",
                "             if (!dataTable) return;\n",
                "\n",
                "             const docLinkHtml = 'Like what you see? Visit the ' +\n",
                "                                                    '<a      target=\"_blank\"
href=https://colab.research.google.com/notebooks/data_table.ipynb>data                table
```

```
notebook</a>'\n",
                    "                    + ' to learn more about interactive tables.';\n",
                    "                element.innerHTML = '';\n",
                    "                dataTable['output_type'] = 'display_data';\n",
                    "                await google.colab.output.renderOutput(dataTable, element);\n",
                    "                const docLink = document.createElement('div');\n",
                    "                docLink.innerHTML = docLinkHtml;\n",
                    "                element.appendChild(docLink);\n",
                    "              }\n",
                    "            </script>\n",
                    "        </div>\n",
                    "    </div>\n",
                    "    "
                ]
            },
            "metadata": {},
            "execution_count": 5
        }
    ],
    "source": [
        "df.head(1)"
    ]
},
{
    "cell_type": "code",
    "execution_count": null,
    "id": "d5a1d429",
    "metadata": {
        "id": "d5a1d429",
        "outputId": "419c8200-88c1-4b52-d3e0-9186a62700cb",
        "colab": {
            "base_uri": "https://localhost:8080/"
        }
    },
    "outputs": [
        {
            "output_type": "execute_result",
            "data": {
                "text/plain": [
                    "0                1\n",
                    "1                2\n",
                    "2                3\n",
                    "3                4\n",
                    "4                5\n",
                    "              ...   \n",
                    "9995         9996\n",
                    "9996         9997\n",
                    "9997         9998\n",
                    "9998         9999\n",
                    "9999        10000\n",
                    "Name: RowNumber, Length: 10000, dtype: int64"
                ]
            },
            "metadata": {},
            "execution_count": 6
        }
    ],
```

```
    "source": [
      "df['RowNumber']"
    ]
  },
  {
    "cell_type": "code",
    "execution_count": null,
    "id": "51273e9c",
    "metadata": {
      "id": "51273e9c",
      "outputId": "94fe4177-0cf3-4391-ddab-ffcf9ba40c95",
      "colab": {
        "base_uri": "https://localhost:8080/"
      }
    },
    "outputs": [
      {
        "output_type": "execute_result",
        "data": {
          "text/plain": [
            "(10000, 14)"
          ]
        },
        "metadata": {},
        "execution_count": 7
      }
    ],
    "source": [
      "df.shape"
    ]
  },
  {
    "cell_type": "code",
    "execution_count": null,
    "id": "c80f5acb",
    "metadata": {
      "id": "c80f5acb",
      "outputId": "36590cfa-5624-4757-e4c5-b06d3e431ddf",
      "colab": {
        "base_uri": "https://localhost:8080/",
        "height": 424
      }
    },
    "outputs": [
      {
        "output_type": "execute_result",
        "data": {
          "text/plain": [
            "         CustomerId       Surname\n",
            "0          15634602     Hargrave\n",
            "1          15647311         Hill\n",
            "2          15619304         Onio\n",
            "3          15701354         Boni\n",
            "4          15737888     Mitchell\n",
            "...             ...          ...\n",
            "9995       15606229     Obijiaku\n",
            "9996       15569892    Johnstone\n",
```

```
    "9997       15584532              Liu\n",
    "9998       15682355    Sabbatini\n",
    "9999       15628319         Walker\n",
    "\n",
    "[10000 rows x 2 columns]"
  ],
  "text/html": [
    "\n",
    "    <div id=\"df-644e99de-bcd9-4f89-af30-ad1c5e17504a\">\n",
    "      <div class=\"colab-df-container\">\n",
    "        <div>\n",
    "<style scoped>\n",
    "      .dataframe tbody tr th:only-of-type {\n",
    "            vertical-align: middle;\n",
    "      }\n",
    "\n",
    "      .dataframe tbody tr th {\n",
    "            vertical-align: top;\n",
    "      }\n",
    "\n",
    "      .dataframe thead th {\n",
    "            text-align: right;\n",
    "      }\n",
    "</style>\n",
    "<table border=\"1\" class=\"dataframe\">\n",
    "    <thead>\n",
    "      <tr style=\"text-align: right;\">\n",
    "        <th></th>\n",
    "        <th>CustomerId</th>\n",
    "        <th>Surname</th>\n",
    "      </tr>\n",
    "    </thead>\n",
    "    <tbody>\n",
    "      <tr>\n",
    "        <th>0</th>\n",
    "        <td>15634602</td>\n",
    "        <td>Hargrave</td>\n",
    "      </tr>\n",
    "      <tr>\n",
    "        <th>1</th>\n",
    "        <td>15647311</td>\n",
    "        <td>Hill</td>\n",
    "      </tr>\n",
    "      <tr>\n",
    "        <th>2</th>\n",
    "        <td>15619304</td>\n",
    "        <td>Onio</td>\n",
    "      </tr>\n",
    "      <tr>\n",
    "        <th>3</th>\n",
    "        <td>15701354</td>\n",
    "        <td>Boni</td>\n",
    "      </tr>\n",
    "      <tr>\n",
    "        <th>4</th>\n",
    "        <td>15737888</td>\n",
    "        <td>Mitchell</td>\n",
```

```
"      </tr>\n",
"      <tr>\n",
"        <th>...</th>\n",
"        <td>...</td>\n",
"        <td>...</td>\n",
"      </tr>\n",
"      <tr>\n",
"        <th>9995</th>\n",
"        <td>15606229</td>\n",
"        <td>Obijiaku</td>\n",
"      </tr>\n",
"      <tr>\n",
"        <th>9996</th>\n",
"        <td>15569892</td>\n",
"        <td>Johnstone</td>\n",
"      </tr>\n",
"      <tr>\n",
"        <th>9997</th>\n",
"        <td>15584532</td>\n",
"        <td>Liu</td>\n",
"      </tr>\n",
"      <tr>\n",
"        <th>9998</th>\n",
"        <td>15682355</td>\n",
"        <td>Sabbatini</td>\n",
"      </tr>\n",
"      <tr>\n",
"        <th>9999</th>\n",
"        <td>15628319</td>\n",
"        <td>Walker</td>\n",
"      </tr>\n",
"    </tbody>\n",
"</table>\n",
"<p>10000 rows × 2 columns</p>\n",
"</div>\n",
"          <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-
644e99de-bcd9-4f89-af30-ad1c5e17504a')\"\n",
"                  title=\"Convert this dataframe to an interactive table.\"\n",
"                  style=\"display:none;\">\n",
"            \n",
"    <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0
0 24 24\"\n",
"           width=\"24px\">\n",
"      <path d=\"M0 0h24v24H0V0z\" fill=\"none\"/>\n",
"        <path d=\"M18.56 5.44l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94
2.06-2.06.94zm-11 1L8.5 8.5l.94-2.06 2.06-.94-2.06-.94L8.5 2.5l-.94 2.06-2.06.94zm10 10l.94
2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94 2.06-2.06.94z\"/><path d=\"M17.41 7.96l-1.37-
1.37c-.4-.4-.92-.59-1.43-.59-.52 0-1.04.2-1.43.59L10.3 9.45l-7.72 7.72c-.78.78-.78 2.05 0
2.83L4 21.41c.39.39.9.59 1.41.59.51 0 1.02-.2 1.41-.59l7.78-7.78 2.81-2.81c.8-.78.8-2.07 0-
2.86zM5.41 20L4 18.59l7.72-7.72 1.47 1.35L5.41 20z\"/>\n",
"    </svg>\n",
"          </button>\n",
"          \n",
"    <style>\n",
"      .colab-df-container {\n",
"        display:flex;\n",
"        flex-wrap:wrap;\n",
```

```
                "           gap: 12px;\n",
                "       }\n",
                "\n",
                "       .colab-df-convert {\n",
                "           background-color: #E8F0FE;\n",
                "           border: none;\n",
                "           border-radius: 50%;\n",
                "           cursor: pointer;\n",
                "           display: none;\n",
                "           fill: #1967D2;\n",
                "           height: 32px;\n",
                "           padding: 0 0 0 0;\n",
                "           width: 32px;\n",
                "       }\n",
                "\n",
                "       .colab-df-convert:hover {\n",
                "           background-color: #E2EBFA;\n",
                "               box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n",
                "           fill: #174EA6;\n",
                "       }\n",
                "\n",
                "       [theme=dark] .colab-df-convert {\n",
                "           background-color: #3B4455;\n",
                "           fill: #D2E3FC;\n",
                "       }\n",
                "\n",
                "       [theme=dark] .colab-df-convert:hover {\n",
                "           background-color: #434B5C;\n",
                "           box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",
                "           filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",
                "           fill: #FFFFFF;\n",
                "       }\n",
                "   </style>\n",
                "\n",
                "           <script>\n",
                "               const buttonEl =\n",
                "                               document.querySelector('#df-644e99de-bcd9-4f89-af30-
ad1c5e17504a button.colab-df-convert');\n",
                "               buttonEl.style.display =\n",
                "                   google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
                "\n",
                "               async function convertToInteractive(key) {\n",
                "                   const element = document.querySelector('#df-644e99de-bcd9-
4f89-af30-ad1c5e17504a');\n",
                "                   const dataTable =\n",
                "                                                                       await
google.colab.kernel.invokeFunction('convertToInteractive',\n",
                "                                                                   [key], {});\n",
                "                   if (!dataTable) return;\n",
                "\n",
                "                   const docLinkHtml = 'Like what you see? Visit the ' +\n",
                "                                                       '<a      target=\"_blank\"
href=https://colab.research.google.com/notebooks/data_table.ipynb>data                table
notebook</a>'\n",
                "                       + ' to learn more about interactive tables.';\n",
                "                   element.innerHTML = '';\n",
```

```
                "                dataTable['output_type'] = 'display_data';\n",
                "                await google.colab.output.renderOutput(dataTable, element);\n",
                "                const docLink = document.createElement('div');\n",
                "                docLink.innerHTML = docLinkHtml;\n",
                "                element.appendChild(docLink);\n",
                "              }\n",
                "            </script>\n",
                "        </div>\n",
                "    </div>\n",
                "    "
              ]
            },
            "metadata": {},
            "execution_count": 8
          }
        ],
        "source": [
          "df[df.columns[1:3]]"
        ]
      },
      {
        "cell_type": "code",
        "execution_count": null,
        "id": "122bc6a0",
        "metadata": {
          "id": "122bc6a0",
          "outputId": "88467dbd-dbca-42f1-87ec-2d996647360f",
          "colab": {
            "base_uri": "https://localhost:8080/",
            "height": 424
          }
        },
        "outputs": [
          {
            "output_type": "execute_result",
            "data": {
              "text/plain": [
                "          CustomerId      Surname  CreditScore Geography  Gender  Age\n",
                "0           15634602     Hargrave          619      France  Female   42\n",
                "1           15647311         Hill          608       Spain  Female   41\n",
                "2           15619304         Onio          502      France  Female   42\n",
                "3           15701354         Boni          699      France  Female   39\n",
                "4           15737888     Mitchell          850       Spain  Female   43\n",
                "...              ...          ...          ...         ...     ...  ...\n",
                "9995        15606229     Obijiaku          771      France    Male   39\n",
                "9996        15569892    Johnstone          516      France    Male
35\n",
                "9997        15584532          Liu          709      France  Female   36\n",
                "9998        15682355    Sabbatini          772     Germany    Male   42\n",
                "9999        15628319       Walker          792      France  Female   28\n",
                "\n",
                "[10000 rows x 6 columns]"
              ],
              "text/html": [
                "\n",
                "  <div id=\"df-f9eaa05b-4820-4848-9f4e-e2bf33162c95\">\n",
                "    <div class=\"colab-df-container\">\n",
```

```
"           <div>\n",
"<style scoped>\n",
"       .dataframe tbody tr th:only-of-type {\n",
"               vertical-align: middle;\n",
"       }\n",
"\n",
"       .dataframe tbody tr th {\n",
"               vertical-align: top;\n",
"       }\n",
"\n",
"       .dataframe thead th {\n",
"               text-align: right;\n",
"       }\n",
"</style>\n",
"<table border=\"1\" class=\"dataframe\">\n",
"    <thead>\n",
"       <tr style=\"text-align: right;\">\n",
"           <th></th>\n",
"           <th>CustomerId</th>\n",
"           <th>Surname</th>\n",
"           <th>CreditScore</th>\n",
"           <th>Geography</th>\n",
"           <th>Gender</th>\n",
"           <th>Age</th>\n",
"       </tr>\n",
"    </thead>\n",
"    <tbody>\n",
"       <tr>\n",
"           <th>0</th>\n",
"           <td>15634602</td>\n",
"           <td>Hargrave</td>\n",
"           <td>619</td>\n",
"           <td>France</td>\n",
"           <td>Female</td>\n",
"           <td>42</td>\n",
"       </tr>\n",
"       <tr>\n",
"           <th>1</th>\n",
"           <td>15647311</td>\n",
"           <td>Hill</td>\n",
"           <td>608</td>\n",
"           <td>Spain</td>\n",
"           <td>Female</td>\n",
"           <td>41</td>\n",
"       </tr>\n",
"       <tr>\n",
"           <th>2</th>\n",
"           <td>15619304</td>\n",
"           <td>Onio</td>\n",
"           <td>502</td>\n",
"           <td>France</td>\n",
"           <td>Female</td>\n",
"           <td>42</td>\n",
"       </tr>\n",
"       <tr>\n",
"           <th>3</th>\n",
"           <td>15701354</td>\n",
```

```
"          <td>Boni</td>\n",
"          <td>699</td>\n",
"          <td>France</td>\n",
"          <td>Female</td>\n",
"          <td>39</td>\n",
"        </tr>\n",
"        <tr>\n",
"          <th>4</th>\n",
"          <td>15737888</td>\n",
"          <td>Mitchell</td>\n",
"          <td>850</td>\n",
"          <td>Spain</td>\n",
"          <td>Female</td>\n",
"          <td>43</td>\n",
"        </tr>\n",
"        <tr>\n",
"          <th>...</th>\n",
"          <td>...</td>\n",
"          <td>...</td>\n",
"          <td>...</td>\n",
"          <td>...</td>\n",
"          <td>...</td>\n",
"          <td>...</td>\n",
"          <td>...</td>\n",
"        </tr>\n",
"        <tr>\n",
"          <th>9995</th>\n",
"          <td>15606229</td>\n",
"          <td>Obijiaku</td>\n",
"          <td>771</td>\n",
"          <td>France</td>\n",
"          <td>Male</td>\n",
"          <td>39</td>\n",
"        </tr>\n",
"        <tr>\n",
"          <th>9996</th>\n",
"          <td>15569892</td>\n",
"          <td>Johnstone</td>\n",
"          <td>516</td>\n",
"          <td>France</td>\n",
"          <td>Male</td>\n",
"          <td>35</td>\n",
"        </tr>\n",
"        <tr>\n",
"          <th>9997</th>\n",
"          <td>15584532</td>\n",
"          <td>Liu</td>\n",
"          <td>709</td>\n",
"          <td>France</td>\n",
"          <td>Female</td>\n",
"          <td>36</td>\n",
"        </tr>\n",
"        <tr>\n",
"          <th>9998</th>\n",
"          <td>15682355</td>\n",
"          <td>Sabbatini</td>\n",
"          <td>772</td>\n",
"          <td>Germany</td>\n",
```

```
                    "          <td>Male</td>\n",
                    "          <td>42</td>\n",
                    "        </tr>\n",
                    "        <tr>\n",
                    "          <th>9999</th>\n",
                    "          <td>15628319</td>\n",
                    "          <td>Walker</td>\n",
                    "          <td>792</td>\n",
                    "          <td>France</td>\n",
                    "          <td>Female</td>\n",
                    "          <td>28</td>\n",
                    "        </tr>\n",
                    "      </tbody>\n",
                    "</table>\n",
                    "<p>10000 rows  脳  6 columns</p>\n",
                    "</div>\n",
                    "          <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-
f9eaa05b-4820-4848-9f4e-e2bf33162c95')\"\n",
                    "                  title=\"Convert this dataframe to an interactive table.\"\n",
                    "                  style=\"display:none;\">\n",
                    "            \n",
                    "    <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0
0 24 24\"\n",
                    "           width=\"24px\">\n",
                    "        <path d=\"M0 0h24v24H0V0z\" fill=\"none\"/>\n",
                    "          <path d=\"M18.56 5.44l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94
2.06-2.06.94zm-11 1L8.5 8.5l.94-2.06 2.06-.94-2.06-.94L8.5 2.5l-.94 2.06-2.06.94zm10 10l.94
2.06.94-2.06 2.06-.94-.94-.94-2.06-.94 2.06-2.06.94z\"/><path d=\"M17.41 7.96l-1.37-
1.37c-.4-.4-.92-.59-1.43-.59-.52 0-1.04.2-1.43.59L10.3 9.45l-7.72 7.72c-.78.78-.78 2.05 0
2.83L4 21.41c.39.39.9.59 1.41.59.51 0 1.02-.2 1.41-.59l7.78-7.78 2.81-2.81c.8-.78.8-2.07 0-
2.86zM5.41 20L4 18.59l7.72-7.72 1.47 1.35L5.41 20z\"/>\n",
                    "    </svg>\n",
                    "          </button>\n",
                    "          \n",
                    "    <style>\n",
                    "      .colab-df-container {\n",
                    "        display:flex;\n",
                    "        flex-wrap:wrap;\n",
                    "        gap: 12px;\n",
                    "      }\n",
                    "\n",
                    "      .colab-df-convert {\n",
                    "        background-color: #E8F0FE;\n",
                    "        border: none;\n",
                    "        border-radius: 50%;\n",
                    "        cursor: pointer;\n",
                    "        display: none;\n",
                    "        fill: #1967D2;\n",
                    "        height: 32px;\n",
                    "        padding: 0 0 0 0;\n",
                    "        width: 32px;\n",
                    "      }\n",
                    "\n",
                    "      .colab-df-convert:hover {\n",
                    "        background-color: #E2EBFA;\n",
                    "            box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px
rgba(60, 64, 67, 0.15);\n",
```

```
                    "          fill: #174EA6;\n",
                    "      }\n",
                    "\n",
                    "      [theme=dark] .colab-df-convert {\n",
                    "          background-color: #3B4455;\n",
                    "          fill: #D2E3FC;\n",
                    "      }\n",
                    "\n",
                    "      [theme=dark] .colab-df-convert:hover {\n",
                    "          background-color: #434B5C;\n",
                    "          box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",
                    "          filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",
                    "          fill: #FFFFFF;\n",
                    "      }\n",
                    "    </style>\n",
                    "\n",
                    "          <script>\n",
                    "            const buttonEl =\n",
                    "                              document.querySelector('#df-f9eaa05b-4820-4848-9f4e-
e2bf33162c95 button.colab-df-convert');\n",
                    "            buttonEl.style.display =\n",
                    "              google.colab.kernel.accessAllowed ? 'block' : 'none';\n",
                    "\n",
                    "            async function convertToInteractive(key) {\n",
                    "                const element = document.querySelector('#df-f9eaa05b-4820-
4848-9f4e-e2bf33162c95');\n",
                    "              const dataTable =\n",
                    "                                                                                await
google.colab.kernel.invokeFunction('convertToInteractive',\n",
                    "                                                                    [key], {});\n",
                    "              if (!dataTable) return;\n",
                    "\n",
                    "              const docLinkHtml = 'Like what you see? Visit the ' +\n",
                    "                                                                    '<a      target=\"_blank\"
href=https://colab.research.google.com/notebooks/data_table.ipynb>data                table
notebook</a>'\n",
                    "                + ' to learn more about interactive tables.';\n",
                    "              element.innerHTML = '';\n",
                    "              dataTable['output_type'] = 'display_data';\n",
                    "              await google.colab.output.renderOutput(dataTable, element);\n",
                    "              const docLink = document.createElement('div');\n",
                    "              docLink.innerHTML = docLinkHtml;\n",
                    "              element.appendChild(docLink);\n",
                    "            }\n",
                    "          </script>\n",
                    "        </div>\n",
                    "    </div>\n",
                    "    "
                ]
            },
            "metadata": {},
            "execution_count": 9
        }
    ],
    "source": [
        "df[df.columns[1:7]]"
    ]
```

```
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "id": "c3de7a7e",
      "metadata": {
        "id": "c3de7a7e"
      },
      "outputs": [],
      "source": [
        "#we need not to perfrom descriptive statistic among the dataset because we dont have any null values in the dataset\n",
        "#we dont have any null values to perform handling missing values"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "id": "c4053f3f",
      "metadata": {
        "id": "c4053f3f",
        "outputId": "1cff808e-21f4-48d4-ce37-a817c3caccc8",
        "colab": {
          "base_uri": "https://localhost:8080/"
        }
      },
      "outputs": [
        {
          "output_type": "execute_result",
          "data": {
            "text/plain": [
              "<function plotly.express._chart_types.histogram(data_frame=None, x=None, y=None, color=None, pattern_shape=None, facet_row=None, facet_col=None, facet_col_wrap=0, facet_row_spacing=None, facet_col_spacing=None, hover_name=None, hover_data=None, animation_frame=None, animation_group=None, category_orders=None, labels=None, color_discrete_sequence=None, color_discrete_map=None, pattern_shape_sequence=None, pattern_shape_map=None, marginal=None, opacity=None, orientation=None, barmode='relative', barnorm=None, histnorm=None, log_x=False, log_y=False, range_x=None, range_y=None, histfunc=None, cumulative=None, nbins=None, text_auto=False, title=None, template=None, width=None, height=None)>"
            ]
          },
          "metadata": {},
          "execution_count": 12
        }
      ],
      "source": [
        "px.histogram"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": null,
      "id": "1836ed87",
      "metadata": {
        "id": "1836ed87",
        "outputId": "2cecb74f-ac8f-4811-f4a9-463ef1c16500",
```

```
      "colab": {
        "base_uri": "https://localhost:8080/"
      }
    },
    "outputs": [
      {
        "output_type": "stream",
        "name": "stdout",
        "text": [
          "[[1 15634602 'Hargrave' ... 1 1 101348.88]\n",
          " [2 15647311 'Hill' ... 0 1 112542.58]\n",
          " [3 15619304 'Onio' ... 1 0 113931.57]\n",
          " ...\n",
          " [9998 15584532 'Liu' ... 0 1 42085.58]\n",
          " [9999 15682355 'Sabbatini' ... 1 0 92888.52]\n",
          " [10000 15628319 'Walker' ... 1 0 38190.78]]\n"
        ]
      }
    ],
    "source": [
      "X = df.iloc[:, :-1].values\n",
      "print(X)"
    ]
  },
  {
    "cell_type": "code",
    "execution_count": null,
    "id": "545a22fb",
    "metadata": {
      "id": "545a22fb",
      "outputId": "169355d3-bf3e-4df4-d521-8e085f5e5e7b",
      "colab": {
        "base_uri": "https://localhost:8080/"
      }
    },
    "outputs": [
      {
        "output_type": "stream",
        "name": "stdout",
        "text": [
          "[1 0 1 ... 1 1 0]\n"
        ]
      }
    ],
    "source": [
      "Y = df.iloc[:, -1].values\n",
      "print(Y)"
    ]
  },
  {
    "cell_type": "code",
    "execution_count": null,
    "id": "b3354453",
    "metadata": {
      "id": "b3354453"
    },
    "outputs": [],
```

```
            "source": []
        }
    ],
    "metadata": {
        "kernelspec": {
            "display_name": "Python 3 (ipykernel)",
            "language": "python",
            "name": "python3"
        },
        "language_info": {
            "codemirror_mode": {
                "name": "ipython",
                "version": 3
            },
            "file_extension": ".py",
            "mimetype": "text/x-python",
            "name": "python",
            "nbconvert_exporter": "python",
            "pygments_lexer": "ipython3",
            "version": "3.9.12"
        },
        "colab": {
            "provenance": [],
            "collapsed_sections": []
        }
    },
    "nbformat": 4,
    "nbformat_minor": 5
}
```