```
{
 "cell_type": "markdown",
 "metadata": {
  "id": "CU48hgo4Owz5"
 },
 "source": [
  "## 1. Split this string"
 ]
},
{
 "cell_type": "code",
 "execution_count": null,
 "metadata": {
  "colab": {
   "background_save": true,
   "base_uri": "https://localhost:8080/"
  },
  "id": "s07c7JK7Oqt-",
  "outputId": "ba1fc876-62eb-420b-bb98-343aef0c49c9"
 },
 "outputs": [
  {
```

```
      "name": "stdout",
      "output_type": "stream",
      "text": [
        "['Hi', 'there', 'Sam!']\n"
      ]
    }
   ],
   "source": [
    "s = \"Hi there Sam!\"\n",
    "print(s.split())"
   ]
  },
  {
   "cell_type": "markdown",
   "metadata": {
    "id": "GH1QBn8HP375"
   },
   "source": [
    "## 2. Use .format() to print the following string. \n",
    "\n",
    "### Output should be: The diameter of Earth is 12742 kilometers."
   ]
```

```
},
{
 "cell_type": "code",
 "execution_count": null,
 "metadata": {
  "colab": {
   "base_uri": "https://localhost:8080/"
  },
  "id": "_ZHoml3kPqic",
  "outputId": "f42c5fb9-202f-4e26-b1fb-dc7589540a95"
 },
 "outputs": [
  {
   "name": "stdout",
   "output_type": "stream",
   "text": [
    "The diameter of the Earth is 12742 kilometers\n"
   ]
  }
 ],
 "source": [
  "planet = \"Earth\"\n",
  "diameter = 12742\n",
```

```
      "print(\"The diameter of the {} is {} kilometers\".
format(planet,diameter))"
     ]
    },
    {
     "cell_type": "code",
     "execution_count": null,
     "metadata": {
      "id": "HyRyJv6CYPb4"
     },
     "outputs": [],
     "source": []
    },
    {
     "cell_type": "markdown",
     "metadata": {
      "id": "KE74ZEwkRExZ"
     },
     "source": [
      "## 3. In this nest dictionary grab the word \"hello\""
     ]
    },
    {
```

```
"cell_type": "code",
"execution_count": null,
"metadata": {
  "colab": {
    "base_uri": "https://localhost:8080/"
  },
  "id": "fcVwbCc1QrQI",
  "outputId": "5ab54153-416a-4995-f990-84d3b4032b58"
},
"outputs": [
  {
    "name": "stdout",
    "output_type": "stream",
    "text": [
      "hello\n"
    ]
  }
],
"source": [
  "d =
{'k1':[1,2,3,{'tricky':['oh','man','inception',{'target':[1,2,3,'hello']}]}]}\n",
  "print(d['k1'][3]['tricky'][3]['target'][3])"
]
```

```
},
{
 "cell_type": "code",
 "execution_count": null,
 "metadata": {
  "id": "MvbkMZpXYRaw"
 },
 "outputs": [],
 "source": []
},
{
 "cell_type": "markdown",
 "metadata": {
  "id": "bw0vVp-9ddjv"
 },
 "source": [
  "# Numpy"
 ]
},
{
 "cell_type": "code",
 "execution_count": null,
 "metadata": {
```

```
    "id": "LLiE_TYrhA1O"
   },
   "outputs": [],
   "source": [
    "import numpy as np"
   ]
  },
  {
   "cell_type": "markdown",
   "metadata": {
    "id": "wOg8hinbgx30"
   },
   "source": [
    "## 4.1 Create an array of 10 zeros? \n",
    "## 4.2 Create an array of 10 fives?"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": null,
   "metadata": {
    "colab": {
     "base_uri": "https://localhost:8080/"
```

```
    },
    "id": "NHrirmgCYXvU",
    "outputId": "14f972dd-4892-4383-a403-ebc334209103"
   },
   "outputs": [
    {
     "name": "stdout",
     "output_type": "stream",
     "text": [
      "An array of 10 zeros:\n",
      "[0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]\n"
     ]
    }
   ],
   "source": [
    "import numpy as np\n",
    "array=np.zeros(10)\n",
    "print(\"An array of 10 zeros:\")\n",
    "print(array)"
   ]
  },
  {
   "cell_type": "code",
```

```
"execution_count": null,
"metadata": {
  "colab": {
    "base_uri": "https://localhost:8080/"
  },
  "id": "e4005lsTYXxx",
  "outputId": "2ab231bf-e0f4-4369-c983-ad7bc5222bed"
},
"outputs": [
  {
    "name": "stdout",
    "output_type": "stream",
    "text": [
      "An array of 10 fives:\n",
      "[5. 5. 5. 5. 5. 5. 5. 5. 5. 5.]\n"
    ]
  }
],
"source": [
  "import numpy as np\n",
  "array=np.ones(10)*5\n",
  "print(\"An array of 10 fives:\")\n",
  "print(array)"
```

```
  ]
 },
 {
  "cell_type": "markdown",
  "metadata": {
   "id": "gZHHDUBvrMX4"
  },
  "source": [
   "## 5. Create an array of all the even integers from 20 to
35"
  ]
 },
 {
  "cell_type": "code",
  "execution_count": null,
  "metadata": {
   "colab": {
    "base_uri": "https://localhost:8080/"
   },
   "id": "oAI2tbU2Yag-",
   "outputId": "0a38a18a-3688-4db7-d946-92f9cff38527"
  },
  "outputs": [
```

```
   {
    "name": "stdout",
    "output_type": "stream",
    "text": [
     "Array of all the even integers from 20 to 35\n",
     "[20 22 24 26 28 30 32 34]\n"
    ]
   }
  ],
  "source": [
   "import numpy as np\n",
   "array=np.arange(20,35,2)\n",
   "print(\"Array of all the even integers from 20 to 35\")\n",
   "print(array) "
  ]
 },
 {
  "cell_type": "markdown",
  "metadata": {
   "id": "NaOM308NsRpZ"
  },
  "source": [
```

```
    "## 6. Create a 3x3 matrix with values ranging from 0 to 8"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": null,
   "metadata": {
    "colab": {
     "base_uri": "https://localhost:8080/"
    },
    "id": "tOlEVH7BYceE",
    "outputId": "640f8269-32e9-48d1-a169-f2e2f6e46d0e"
   },
   "outputs": [
    {
     "data": {
      "text/plain": [
       "array([[0, 1, 2],\n",
       "       [3, 4, 5],\n",
       "       [6, 7, 8]])"
      ]
     },
```

```
      "execution_count": 5,
      "metadata": {},
      "output_type": "execute_result"
    }
  ],
  "source": [
    "import numpy as np\n",
    "np.arange(0,9).reshape((3,3))"
  ]
},
{
  "cell_type": "markdown",
  "metadata": {
    "id": "hQ0dnhAQuU_p"
  },
  "source": [
    "## 7. Concatenate a and b \n",
    "## a = np.array([1, 2, 3]), b = np.array([4, 5, 6])"
  ]
},
{
  "cell_type": "code",
  "execution_count": null,
```

```
"metadata": {
  "colab": {
    "base_uri": "https://localhost:8080/"
  },
  "id": "rAPSw97aYfE0",
  "outputId": "94eb1991-3ae3-4a1d-a706-064be3810153"
},
"outputs": [
  {
    "name": "stdout",
    "output_type": "stream",
    "text": [
      "[1 2 3 4 5 6]\n"
    ]
  }
],
"source": [
  "import numpy as np\n",
  "a= np.array([1,2,3])\n",
  "b= np.array([4,5,6])\n",
  "c= np.concatenate((a,b),axis= None)\n",
  "print(c)"
]
```

```
},
{
 "cell_type": "markdown",
 "metadata": {
  "id": "dlPEY9DRwZga"
 },
 "source": [
  "# Pandas"
 ]
},
{
 "cell_type": "markdown",
 "metadata": {
  "id": "ijoYW51zwr87"
 },
 "source": [
  "## 8. Create a dataframe with 3 rows and 2 columns"
 ]
},
{
 "cell_type": "code",
 "execution_count": null,
 "metadata": {
```

```
  "colab": {
    "base_uri": "https://localhost:8080/",
    "height": 143
  },
  "id": "T5OxJRZ8uvR7",
  "outputId": "762fd71f-a98d-486d-f8b6-8414dfe95c8b"
 },
 "outputs": [
  {
   "data": {
     "text/html": [
      "\n",
      "  <div id=\"df-8943a84f-fedf-4c9e-a747-
6e00d4b26e02\">\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>Name</th>\n",
"      <th>Age</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>0</th>\n",
"      <td>sri</td>\n",
"      <td>10</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>1</th>\n",
"      <td>maha</td>\n",
```

```
"      <td>15</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>2</th>\n",
"      <td>priya</td>\n",
"      <td>14</td>\n",
"    </tr>\n",
"  </tbody>\n",
"</table>\n",
"</div>\n",
"      <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-8943a84f-fedf-4c9e-a747-6e00d4b26e02')\"\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-8943a84f-fedf-
4c9e-a747-6e00d4b26e02 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-
8943a84f-fedf-4c9e-a747-6e00d4b26e02');\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",
          "  "
        ],
        "text/plain": [
          "    Name  Age\n",
          "0    sri   10\n",
          "1   maha   15\n",
          "2  priya   14"
        ]
      },
      "execution_count": 3,
      "metadata": {},
      "output_type": "execute_result"
    }
```

```
  ],
  "source": [
    "import pandas as pd\n",
    "data = [['sri', 10], ['maha', 15], ['priya', 14]]\n",
    "df = pd.DataFrame(data, columns=['Name', 'Age'])\n",
    "df\n"
  ]
},
{
  "cell_type": "code",
  "execution_count": null,
  "metadata": {
    "id": "xNpI_XXoYhs0"
  },
  "outputs": [],
  "source": []
},
{
  "cell_type": "markdown",
  "metadata": {
    "id": "UXSmdNclyJQD"
  },
  "source": [
```

```
    "## 9. Generate the series of dates from 1st Jan, 2023 to 10th Feb, 2023"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": null,
   "metadata": {
    "colab": {
     "base_uri": "https://localhost:8080/"
    },
    "id": "dgyC0JhVYl4F",
    "outputId": "ca0a6d0d-403e-4153-f1de-e3cc615bfca9"
   },
   "outputs": [
    {
     "name": "stdout",
     "output_type": "stream",
     "text": [
      "2023-01-01 00:00:00\n",
      "2023-01-02 00:00:00\n",
      "2023-01-03 00:00:00\n",
      "2023-01-04 00:00:00\n",
```

"2023-01-05 00:00:00\n",

"2023-01-06 00:00:00\n",

"2023-01-07 00:00:00\n",

"2023-01-08 00:00:00\n",

"2023-01-09 00:00:00\n",

"2023-01-10 00:00:00\n",

"2023-01-11 00:00:00\n",

"2023-01-12 00:00:00\n",

"2023-01-13 00:00:00\n",

"2023-01-14 00:00:00\n",

"2023-01-15 00:00:00\n",

"2023-01-16 00:00:00\n",

"2023-01-17 00:00:00\n",

"2023-01-18 00:00:00\n",

"2023-01-19 00:00:00\n",

"2023-01-20 00:00:00\n",

"2023-01-21 00:00:00\n",

"2023-01-22 00:00:00\n",

"2023-01-23 00:00:00\n",

"2023-01-24 00:00:00\n",

"2023-01-25 00:00:00\n",

"2023-01-26 00:00:00\n",

"2023-01-27 00:00:00\n",

```
        "2023-01-28 00:00:00\n",
        "2023-01-29 00:00:00\n",
        "2023-01-30 00:00:00\n",
        "2023-01-31 00:00:00\n",
        "2023-02-01 00:00:00\n",
        "2023-02-02 00:00:00\n",
        "2023-02-03 00:00:00\n",
        "2023-02-04 00:00:00\n",
        "2023-02-05 00:00:00\n",
        "2023-02-06 00:00:00\n",
        "2023-02-07 00:00:00\n",
        "2023-02-08 00:00:00\n",
        "2023-02-09 00:00:00\n",
        "2023-02-10 00:00:00\n"
       ]
      }
     ],
     "source": [
      "import pandas as pd\n",
      "abc = pd.date_range(start ='01-01-2023',end='02-10-
2023', freq ='24H')\n",
      "for val in abc:\n",
      "    print(val)"
```

```
  ]
 },
 {
  "cell_type": "markdown",
  "metadata": {
   "id": "ZizSetD-y5az"
  },
  "source": [
   "## 10. Create 2D list to DataFrame\n",
   "\n",
   "lists = [[1, 'aaa', 22],\n",
   "         [2, 'bbb', 25],\n",
   "         [3, 'ccc', 24]]"
  ]
 },
 {
  "cell_type": "code",
  "execution_count": null,
  "metadata": {
   "id": "_XMC8aEt0llB"
  },
  "outputs": [],
  "source": [
```

```
      "lists = [[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]]"
     ]
    },
    {
     "cell_type": "code",
     "execution_count": null,
     "metadata": {
      "colab": {
       "base_uri": "https://localhost:8080/",
       "height": 143
      },
      "id": "knH76sDKYsVX",
      "outputId": "138d513e-7027-4aea-dac6-15048166ea3b"
     },
     "outputs": [
      {
       "data": {
        "text/html": [
         "\n",
         "  <div id=\"df-3d43efd4-b6f5-46f3-b2f9-
258855078c05\">\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>sl.no</th>\n",
"      <th>name</th>\n",
"      <th>age</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
```

```
"    <tr>\n",
"      <th>0</th>\n",
"      <td>1</td>\n",
"      <td>aaa</td>\n",
"      <td>22</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>1</th>\n",
"      <td>2</td>\n",
"      <td>bbb</td>\n",
"      <td>25</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>2</th>\n",
"      <td>3</td>\n",
"      <td>ccc</td>\n",
"      <td>24</td>\n",
"    </tr>\n",
"  </tbody>\n",
"</table>\n",
"</div>\n",
```

```
"      <button class=\"colab-df-convert\"
onclick=\"convertToInteractive('df-3d43efd4-b6f5-46f3-b2f9-
258855078c05')\"\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-3d43efd4-b6f5-
46f3-b2f9-258855078c05 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-
3d43efd4-b6f5-46f3-b2f9-258855078c05');\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",
        "  "
       ],
       "text/plain": [
        "   sl.no name  age\n",
        "0      1  aaa   22\n",
        "1      2  bbb   25\n",
        "2      3  ccc   24"
       ]
      },
      "execution_count": 1,
      "metadata": {},
      "output_type": "execute_result"
     }
    ],
    "source": [
     "import pandas as pd\n",
     "lists = [[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]]\n",
     "df = pd.DataFrame(lists, columns=['sl.no',
'name','age'])\n",
     "df"
    ]
   }
```

```
 ],
 "metadata": {
  "colab": {
   "collapsed_sections": [],
   "provenance": []
  },
  "kernelspec": {
   "display_name": "Python 3",
   "name": "python3"
  },
  "language_info": {
   "name": "python"
  }
 },
 "nbformat": 4,
 "nbformat_minor": 0
}
```