

```

{"nbformat":4,"nbformat_minor":0,"metadata":{"colab":{"provenance
":[],"collapsed_sections":[],"kernel_spec":{"name":"python3","display_name":"Python
3"},"language_info":{"name":"python"},"cells":[{"cell_type":"markdown","source":["# Basic
Python"],"metadata":{"id":"McSxJAwcOdZ1"},"cell_type":"markdown","source":["## 1. Split this
string"],"metadata":{"id":"CU48hgo4Owz5"},"cell_type":"code","source":["s = \"Hi there Sam!
\\n\""],"metadata":{"id":"s07c7JK7Oqt-"},"execution_count":null,"out
puts":[],"cell_type":"code","source":["s = \"Hi there Sam!
\\n\\n\",x=s.split()\\n\",\"print(x)"],"metadata":{"colab":{"base_uri
":"https://localhost:8080/"},"id":"sTEuQGP-0XuH","executionInfo":{"
status":"ok","timestamp":1663256643954,"user_tz":-330,"elapsed":12,"user":{"displayName":"Sneka
M","userId":"15402403260744541280"},"outputId":"cbd762d5-1291-4571c807-
9c252fdd79b9"},"execution_count":null,"outputs":[{"output_ty
pe":"stream","name":"stdout","text":["[\"Hi\", 'there',
'Sam!']\\n\\n"]}],"cell_type":"markdown","source":["## 2. Use .format() to print the following string.
\\n\\n\\n","### Output should be: The diameter of Earth is 12742
kilometers."],"metadata":{"id":"GH1QBn8HP375"},"cell_type":"code","source":["planet =
\\\"Earth\\\"\\n\",\"diameter = 12742\\n\""],"metadata":{"id":"_ZHoml3kPqic"},"execution_count":null,"
outputs":[],"cell_type":"code","source":["planet = \\\"Earth\\\"\\n\",\"diameter = 12742\\n\",\"print('The
diameter of {one} is {two} kilometers.'
\\n\\n\",\".format(one=planet,two=diameter));"],"metadata":{"id":"HyRyJ
v6CYPb4"},"colab":{"base_uri":"https://localhost:8080/"},"executio
nInfo":{"status":"ok","timestamp":1663256981032,"user_tz":-330,"e
lapsed":987,"user":{"displayName":"Sneka
M","userId":"15402403260744541280"},"outputId":"3784df58-35dc-402a-
b519b243d2875068"},"execution_count":null,"outputs":[{"output_type":"
stream","name":"stdout","text":["The diameter of Earth is 12742
kilometers.\\n\\n"]}],"cell_type":"markdown","source":["## 3. In this nest dictionary grab the word \\\"hello
\\\""],"metadata":{"id":"KE74ZEwkRExZ"},"cell_type":"code","source":["d =
{'k1':[1,2,3,{'tricky':['oh','man','inception',{'target':[1,2,3,'
hello']}]}]}\\n\""],"metadata":{"id":"fcVwbCc1QrQl"},"execution_coun
t":null,"outputs":[],"cell_type":"code","source":["d =
{'k1':[1,2,3,{'tricky':['oh','man','inception',{'target':[1,2,3,'hello']}]}]}\\n\",\"d['k1'][3]['tricky'][3]['target']
[3]"],"metadata":{"id":"TAfKBrx6Or7r"},"colab":{"base_uri":"https:
//localhost:8080/","height":35},"executionInfo":{"status":"ok","timestamp":1663257420943,"user_tz":-
330,"elapsed":922,"user":{"displayName":"Sneka
M","userId":"15402403260744541280"},"outputId":"9776b471ad31-4606f4de-
05e2b99130bb"},"execution_count":null,"outputs":[{"output_ty
pe":"execute_result","data":{"text/plain":["hello"]},"applicatio
n/vnd.google.colaboratory.intrinsic+json":{"type":"string"},"met

```

```
adata":{"execution_count":9}},"cell_type":"markdown","source":["#
Numpy"],"metadata":{"id":"bw0vVp-9ddjv"},"cell_type":"code","source":["import numpy as
np"],"metadata":{"id":"LLiE_TYrhA1O"},"execution_count":null,"out
puts":[]},"cell_type":"markdown","source":["## 4.1 Create an array of 10 zeros? \n","## 4.2 Create an
array of 10 fives?"],"metadata":{"id":"wOg8hinbgx30"},"cell_type":"code","source":["import numpy
as np\n","array=np.zeros(10)\n","print(\nAn array of 10 zeros:
\n)\n","print(array)"],"metadata":{"id":"NHrirmgCYXvU"},"colab":{"
base_uri":"https://localhost:8080/"},"executionInfo":{"status":"o
k","timestamp":1663257753771,"user_tz":-330,"elapsed":511,"user":{"displayName":"Sneka
M","userId":"15402403260744541280"},"outputId":"986fa55ca8bf-4a58fb9f-
20834c6c56d2"},"execution_count":null,"outputs":[{"output_ty
pe":"stream","name":"stdout","text":["An array of 10 zeros: \n","[0. 0. 0. 0. 0. 0. 0. 0. 0.
0.]\n"]}]},"cell_type":"code","source":["import numpy as np\n","array=np.ones(10)*5\n","print(\nAn
array of 10 fives: \n)\n","print(array)"],"metadata":{"id":"e4005lsTYXxx"},"colab":{"
base_uri":"https://localhost:8080/"},"executionInfo":{"status":"o
k","timestamp":1663257811840,"user_tz":-330,"elapsed":465,"user":{"displayName":"Sneka
M","userId":"15402403260744541280"},"outputId":"322bf618fe52-442bfd7e-
6d44658f8d69"},"execution_count":null,"outputs":[{"output_ty
pe":"stream","name":"stdout","text":["An array of 10 fives: \n","[5. 5. 5. 5. 5. 5. 5. 5. 5.
5.]\n"]}]},"cell_type":"markdown","source":["## 5. Create an array of all the even integers from 20 to
35"],"metadata":{"id":"gZHHDUBvrMX4"},"cell_type":"code","source":["import numpy as np
\n","array=np.arange(20,36,2)\n","print(\nArray of all the even integers from 20 to 35
\n)\n","print(array)"],"metadata":{"id":"oAl2tbU2Yag-"},"colab":{"
base_uri":"https://localhost:8080/"},"executionInfo":{"status":"o
k","timestamp":1663257541627,"user_tz":-330,"elapsed":17,"user":{"displayName":"Sneka
M","userId":"15402403260744541280"},"outputId":"82d2ec72-2b37-4f fda861-
2f362d631131"},"execution_count":null,"outputs":[{"output_ty
pe":"stream","name":"stdout","text":["Array of all the even integers from 20 to 35\n","[20 22 24 26 28
30 32 34]\n"]}]},"cell_type":"markdown","source":["## 6. Create a 3x3 matrix with values ranging from
0 to 8"],"metadata":{"id":"NaOM308NsRpZ"},"cell_type":"code","source":["import numpy as np\n","x
= np.arange(0,9).reshape(3,3)\n","print(x)"],"metadata":{"id":"tOIE
VH7BYceE"},"colab":{"base_uri":"https://localhost:8080/"},"executi
onInfo":{"status":"ok","timestamp":1663259117617,"user_tz":-330,"
elapsed":600,"user":{"displayName":"Sneka
M","userId":"15402403260744541280"},"outputId":"2bc68292aeaa-468acfd9-
22a7adf50b27"},"execution_count":null,"outputs":[{"output_ty
pe":"stream","name":"stdout","text":["[[0 1 2]\n"," [3 4 5]\n"," [6 7
8]]\n"]}]},"cell_type":"markdown","source":["## 7. Concatenate a and b \n","## a = np.array([1, 2, 3]), b
= np.array([4, 5, 6])"],"metadata":{"id":"hQ0dnhAQuU_p"},"cell_type":"code","source":["import
```

```

numpy as np\n", "a = np.array([1, 2, 3])\n", "b = np.array([4, 5,
6])\n", "np.concatenate((a,b))", "metadata":{"id":"rAPSw97aYfEO", "
colab":{"base_uri":"https://localhost:8080/"},"executionInfo":{"s
tatus":"ok", "timestamp":1663257630295, "user_tz":-330, "elapsed":46 1, "user":{"displayName":"Sneka
M", "userId":"15402403260744541280"}}, "outputId":"f011096a-7c89-42 c2-
9b48ab4b4b9f1826"},"execution_count":null, "outputs":[{"output_type":
execute_result", "data":{"text/plain":["array([1, 2, 3, 4, 5,
6])"]}, "metadata":{}, "execution_count":11}}, {"cell_type":"markdo wn", "source":["#
Pandas"], "metadata":{"id":"dIPEY9DRwZga"}}, {"cell_type":"markdown ", "source":["## 8. Create a
dataframe with 3 rows and 2 columns"], "metadata":{"id":"ijoYW51zwr87"}}, {"cell_type":"code", "
source":["import pandas as pd \n"], "metadata":{"id":"T5OxJRZ8uvR7"}, "execution_count":null, "out
puts":[]}, {"cell_type":"code", "source":["import pandas as pd
\n", "data=[[\\"john\\",10],[\\"jose\\",17],[\\"jessi
\\",13]]\n", "df=pd.DataFrame(data,columns=['Name','Age'])\n", "df"]
, "metadata":{"id":"xNpI_XXoYhs0", "colab":{"base_uri":"https://loc
alhost:8080/", "height":143}, "executionInfo":{"status":"ok", "times tamp":1663259026289, "user_tz":-
330, "elapsed":988, "user":{"display Name":"Sneka
M", "userId":"15402403260744541280"}}, "outputId":"ec52a97a-7684-4b 4b-
e773cd8be90998de"},"execution_count":null, "outputs":[{"output_type":
execute_result", "data":{"text/plain":[" Name Age\n", "0 john 10\n", "1 jose 17\n", "2 jessi
13"], "text/html":["\n", " <div id=\"df-ecd2c03f-44ee-475e-a762-d1cc60ae66ed\">\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", " textalign: right;\n", " } \n", "</style>\n", "<table
border=\"1\" class=\"dataframe\">\n", " <thead>\n", " <tr style=\"textalign: right;\n">\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>john</td>\n", " <td>10</td>\n", " </tr>\n", " <tr>\n", "
<th>1</th>\n", "

```

```

20z\"/>\n"," </svg>\n"," </button>\n"," \n"," <style>\n"," .colab-df-container {\n","
display:flex;\n"," flex-wrap:wrap;\n"," gap: 12px;\n"," }\n",".colab-df-convert {\n","
backgroundcolor: #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",".colab-dfconvert: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",".colab-df-convert {\n"," background-color: #3B4455;\n","
fill: #D2E3FC;\n"," }\n",".colab-df-convert:hover {\n"," background-color:
#434B5C;\n"," box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n"," filter: dropshadow(0px 1px
2px rgba(0, 0, 0, 0.3));\n"," fill: #FFFFFF;\n"," }\n"," </style>\n"," <script>\n"," const
buttonEl =\n"," document.querySelector('#dfecd2c03f-44ee-475e-a762-d1cc60ae66ed
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-ecd2c03f-44ee-475e-a762d1cc60ae66ed');\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":18}},{"cell_type":"markdown","source":["## 9. Generate the
series of dates from 1st Jan, 2023 to 10th Feb,
2023"],"metadata":{"id":"UXSmdNclyJQD"},"cell_type":"code","source":["import
datetime\n","start=datetime.date(2023,1,1)\n","k=40\n","res=[]\n","for day in range(k):\n","
date=(start+datetime.timedelta(days=day)).isoformat()\n"," res.append(date)\n","print(\"Dates-
\"+str(res)) "],"metadata":{"id":"1Vg1oaTbamwO","executionInfo":{"status":"ok",
"timestamp":1663261106862,"user_tz":-330,"elapsed":22,"user":{"displayName":"Sneka
M","userId":"15402403260744541280"}}, "colab":{"base_uri":"https://
/localhost:8080/"},"outputId":"ab9b267d-5a9d-418fd8ab-
98ebe9db9fd9"},"execution_count":34,"outputs":[{"output_type
":"stream","name":"stdout","text":["Dates-['2023-01-01', '2023-01-02', '2023-01-03', '2023-01-04',
'2023-01-05', '2023-01-06', '2023-01-07', '2023-01-08', '2023-01-09', '2023-01-10', '2023-01-11', '2023-
01-12', '2023-01-13', '2023-01-14', '2023-01-15', '2023-01-16', '2023-01-17', '2023-01-18', '2023-01-19',
'2023-01-20', '2023-01-21', '2023-01-22', '2023-01-23', '2023-01-24', '2023-01-25', '2023-01-26', '2023-
01-27', '2023-01-28', '2023-01-29', '2023-01-30', '2023-01-31', '2023-02-01', '2023-02-02', '2023-02-03',
'2023-02-04', '2023-02-05', '2023-02-06', '2023-02-07', '2023-02-08', '2023-02-

```

```

09']\n"]}},{"cell_type":"markdown","source":["## 10. Create 2D list to DataFrame\n","\n","lists = [[1,
'aaa', 22], \n","    [2, 'bbb', 25],\n","    [3, 'ccc',
24]]"],"metadata":{"id":"ZizSetDy5az"}},{"cell_type":"code","source":["lists = [[1, 'aaa', 22], [2, 'bbb', 25],
[3, 'ccc', 24]]"],"metadata":{"id":"_XMC8aEt0lIB"},"execution_count":null,"o
utputs":[]}, {"cell_type":"code","source":["import pandas as pd \n","lists=[[1, 'aaa', 22],\n","    [2, 'bbb',
25], \n","    [3, 'ccc', 24]]\n","df=pd.DataFrame(data=[lists])\n","print(df)"],"metadata"
":{"id":"knH76sDKYsVX"},"colab":{"base_uri":"https://localhost:8080
/"},"executionInfo":{"status":"ok","timestamp":1663260340354,"use r_tz":-
330,"elapsed":1401,"user":{"displayName":"Sneka
M"},"userId":"15402403260744541280"}}, {"outputId":"2e84b271d522-4778abe4-
5bd8309bd25c"},"execution_count":null,"outputs":[{"output_ty pe":"stream","name":"stdout","text":["
0      1      2\n","0 [1, aaa, 22] [2, bbb, 25] [3, ccc, 24]\n"]}]}}

```