```
{"nbformat":4, "nbformat minor":0, "metadata": { "colab": { "provenance
":[], "collapsed sections":[]}, "kernelspec": { "name": "python3", "dis
play name": "Python
3"}, "language info": { "name": "python" } }, "cells": [{ "cell type": "mar
kdown", "source": ["# Basic
Python"], "metadata": {"id": "McSxJAwcOdZ1"}}, {"cell type": "markdown
", "source": ["## 1. Split this
string"], "metadata": {"id": "CU48hqo40wz5"}}, {"cell type": "code", "s
ource":["s = \"Hi there Sam!
\""], "metadata": { "id": "s07c7JK7Oqt-"}, "execution count": null, "out
puts":[]},{"cell type":"code","source":["s = \"Hi there Sam!
\"\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": "cbd762d5-1291-45
71-
c807-9c252fdd79b9"}, "execution_count":null, "outputs":[{"output_ty
pe":"stream", "name": "stdout", "text": ["['Hi', 'there',
'Sam!']\n"]}]}, {"cell type": "markdown", "source": ["## 2.
Use .format() to print the following string. \n","\n","### Output
should be: The diameter of Earth is 12742
kilometers."], "metadata": {"id": "GH1QBn8HP375"}}, {"cell type": "cod
e", "source": ["planet = \"Earth\"\n", "diameter =
12742"], "metadata": {"id": " ZHoml3kPqic"}, "execution count":null, "
outputs":[]},{"cell type":"code", "source":["planet = \"Earth
\"\n", "diameter = 1\overline{2}742\n", "print('The diameter of {one} is {two}
kilometers.'
\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-40
2a-b519-
b243d2875068"}, "execution count":null, "outputs":[{"output type":"
stream", "name": "stdout", "text": ["The diameter of Earth is 12742
kilometers.\n"]}]},{"cell type":"markdown","source":["## 3. In
this nest dictionary grab the word \"hello
\""], "metadata": { "id": "KE74ZEwkRExZ"}}, { "cell type": "code", "sourc
e":["d =
{'k1':[1,2,3,{'tricky':['oh','man','inception',{'target':[1,2,3,'
hello']}]})\n"], "metadata": { "id": "fcVwbCc1QrQI"}, "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", "t
imestamp":1663257420943, "user tz":-330, "elapsed":922, "user":{"dis
playName":"Sneka
M", "userId": "15402403260744541280"}}, "outputId": "9776b471-
ad31-4606-
f4de-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", "so
urce":["import numpy as
np"], "metadata": { "id": "LLiE TYrhA10" }, "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", "s
ource":["import numpy as np\n", "array=np.zeros(10)\n", "print(\"An
array of 10 zeros:
\")\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": "986fa55c-
a8bf-4a58-
fb9f-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(\"An array of 10 fives:
\") \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": "322bf618-
fe52-442b-
fd7e-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", "sourc
e":["import numpy as np
\n", "array=np.arange(20,36,2)\n", "print(\"Array of all the even
integers from 20 to 35
\")\n", "print(array)"], "metadata": { "id": "oAI2tbU2Yaq-", "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
a861-2f362d631131"}, "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": "tOlE}
```

```
VH7BYceE", "colab": { "base uri": "https://localhost:8080/" }, "executi
onInfo":{"status":"ok","\overline{tamp}":1663259117617,"user tz":-330,"
elapsed":600,"user":{"displayName":"Sneka
M", "userId": "15402403260744541280" } }, "outputId": "2bc68292-
aeaa-468a-
cfd9-22a7adf50b27"}, "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 \sqrt{n}, "## a = np.array([1, 2, 3]), b =
np.array([4, 5,
6])"],"metadata":{"id":"hQ0dnhAQuU p"}},{"cell type":"code","sour
ce":["import numpy as np\n","a = np.array([1, \overline{2}, 3])\n","b =
np.array([4, 5,
6])\n", "np.concatenate((a,b))"], "metadata":{"id": "rAPSw97aYfE0","
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
c2-9b48-
ab4b4b9f1826"}, "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": "dlPEY9DRwZga"}}, {"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": "T50xJRZ8uvR7"}, "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-e773-
cd8be90998de"}, "execution count":null, "outputs":[{"output type":"
execute result", "data": {"text/plain": [" Name Age\n", "0
10\n", "\overline{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","
tr th {\n","
                 vertical-align: top;
\n'', " \n'', " .dataframe thead th \n'', "
                  \n","</style>\n","
align: right; \n","
class=\"dataframe\">\n"," <thead>\n","
                                       <tr style=\"text-
Name\n","
<th>Age\n"," </tr>\n"," </thead>\n"," <tbody>\n","
john\n","
10\n"," \n","
                               \n","
                                             1\n","
```

```
\n","
            2\n","
                                jessi\n","
13\n","
                 \n","
\n","\n","</div>\n","
                                      <button class=\"colab-
df-convert\" onclick=\"convertToInteractive('df-
ecd2c03f-44ee-475e-a762-d1cc60ae66ed') \"\n","
                                                      title=
\"Convert this dataframe to an interactive table.\"\n","
\"http://www.w3.org/2000/svq\" height=\"24px\"viewBox=\"0 0 24 24
\"\n","
            width=\"24px\">\n"," <path d=\"M0 0h24v24H0V0z\"
fill=\"none\"/>\n","
                   <path d=\"M18.56 5.441.94 2.06.94-2.06</pre>
2.06-.94-2.06-.94-.94-2.06-.94 2.06-2.06.94zm-11 1L8.5
8.51.94-2.06 2.06-.94-2.06-.94L8.5 2.51-.94 2.06-2.06.94zm10
101.94 2.06.94-2.06 2.06-.94-2.06-.94-2.06-.94 2.06-2.06.94z
\"/><path d=\"M17.41 7.961-1.37-1.37c-.4-.4-.92-.59-1.43-.59-.52
0-1.04.2-1.43.59L10.3 9.451-7.72 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-.5917.78-7.78
2.81-2.81c.8-.78.8-2.07 0-2.86zM5.41 20L4 18.5917.72-7.72 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;
                     .colab-df-convert {\n"," background-
\n"," }\n","\n","
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"," </style>\n","\n","
                                     <script>\n","
const buttonEl =\n","
                            document.querySelector('#df-
ecd2c03f-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-a762-
d1cc60ae66ed'); \n","
                           const dataTable =\n","
await google.colab.kernel.invokeFunction('convertToInteractive',
\n","
                                                      [key],
                if (!dataTable) return; \n", "\n","
{});\n","
const docLinkHtml = 'Like what you see? Visit the ' +\n","
'<a target=\" blank\"</pre>
href=https://colab.research.google.com/notebooks/data table.ipynb
>data table notebook</a>'\n","
interactive tables.';\n","
                                      + ' to learn more about
                                element.innerHTML = '';\n","
dataTable['output type'] = 'display_data';\n","
```

```
google.colab.output.renderOutput(dataTable, element); \n","
const docLink = document.createElement('div'); \n","
docLink.innerHTML = docLinkHtml; \n", "
element.appendChild(docLink); \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", "sou
rce":["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": "1VgloaTbamw0", "executionInfo": { "status": "ok"
"timestamp":1663261106862, "user tz":-330, "elapsed":22, "user": { "d
isplayName": "Sneka
M", "userId": "15402403260744541280"}}, "colab": {"base uri": "https:/
/localhost:8080/"},"outputId":"ab9b267d-5a9d-418f-
d8ab-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', '2023-01-10', '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 DataFramen', "n', "lists = [[1, 'aaa', 22],
               [2, 'bbb', 25], \n","
                                               [3, 'ccc',
241]"],"metadata":{"id":"ZizSetD-
y5az"}}, {"cell type":"code", "source":["lists = [[1, 'aaa', 22],
[2, 'bbb', 25], [3, 'ccc',
24]]"], "metadata": {"id": " XMC8aEt0llB"}, "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": "2e84b271-
d522-4778-
abe4-5bd8309bd25c"}, "execution count":null, "outputs": [{"output ty
pe":"stream", "name": "stdout", "text":["
               2\n","0 [1, aaa, 22] [2, bbb, 25] [3, ccc,
24]\n"]}]}]
```