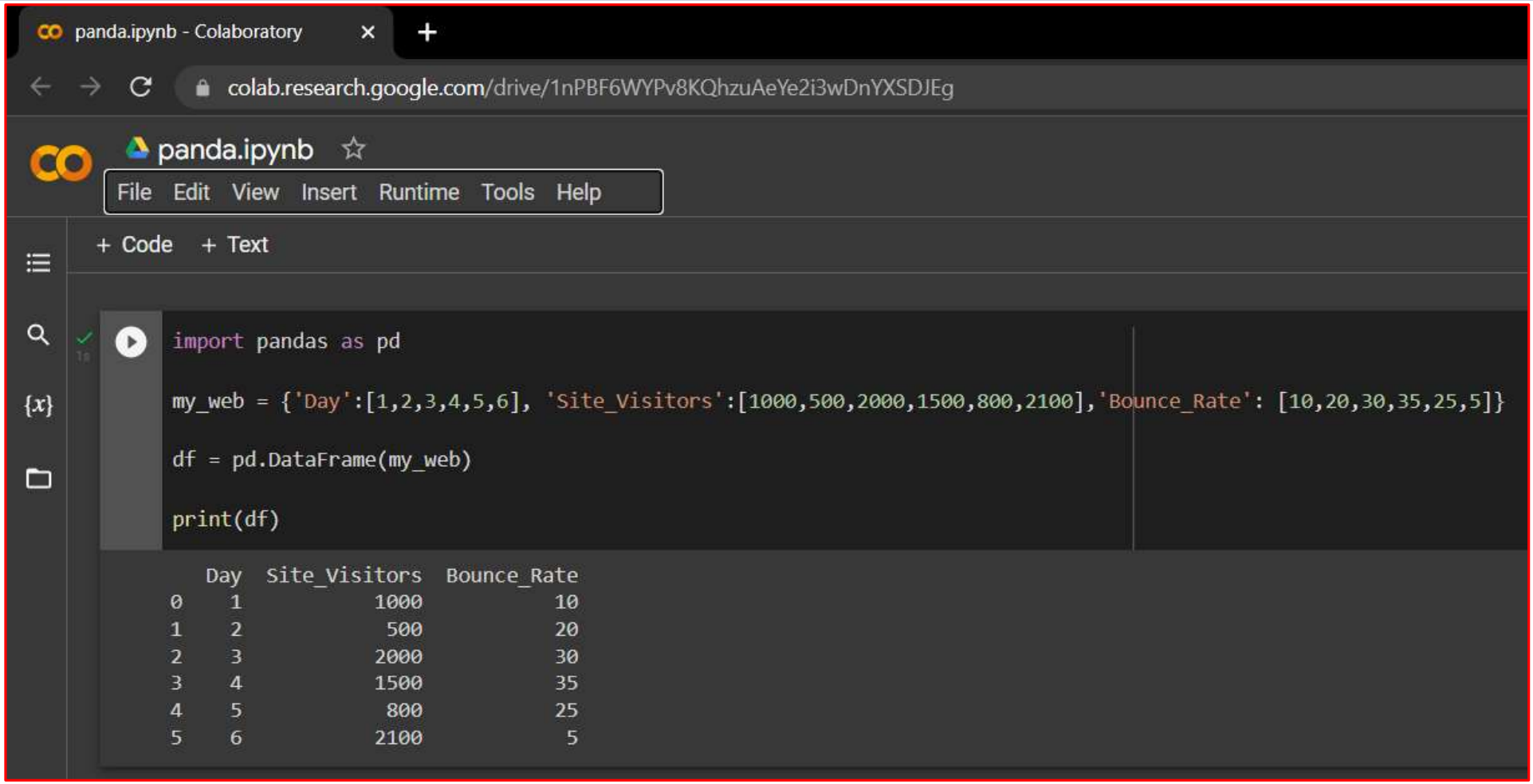
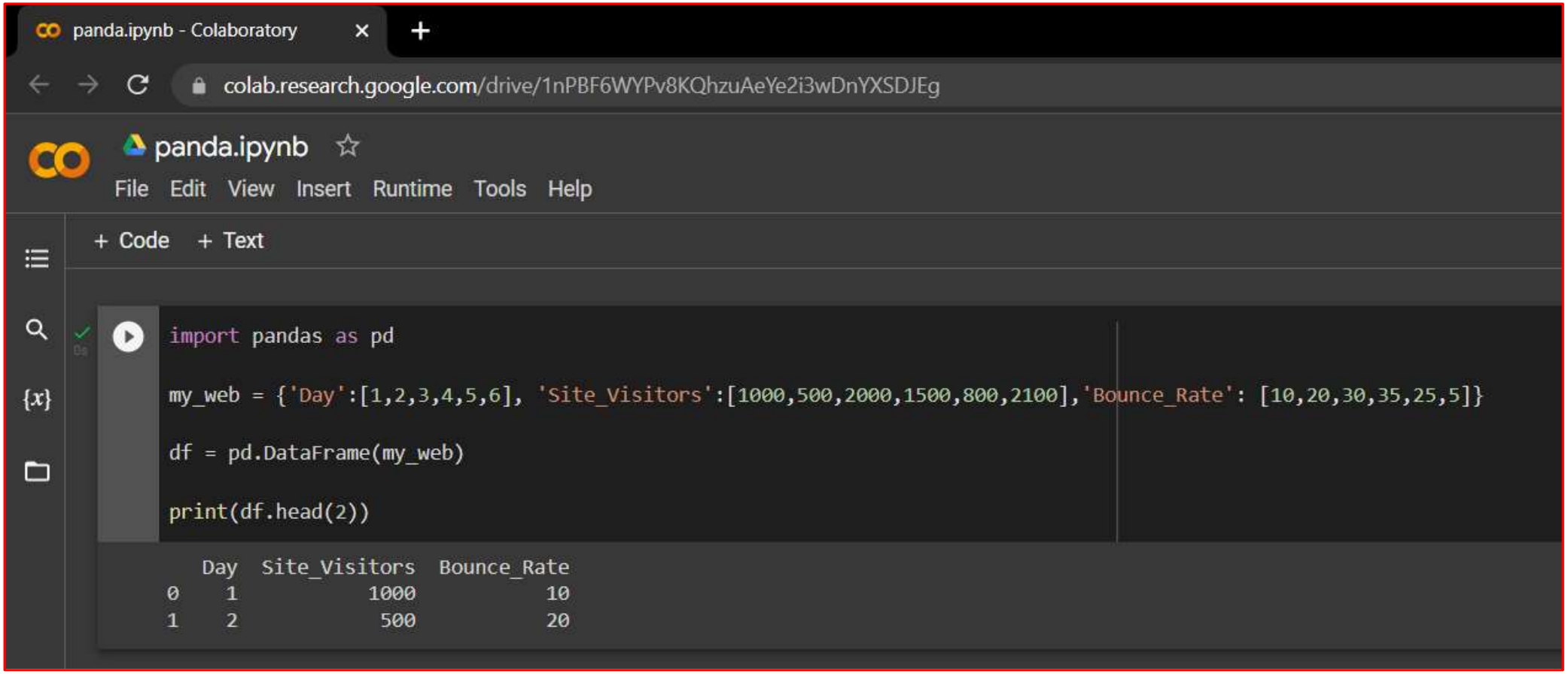
pandas

www.hbpatel.in

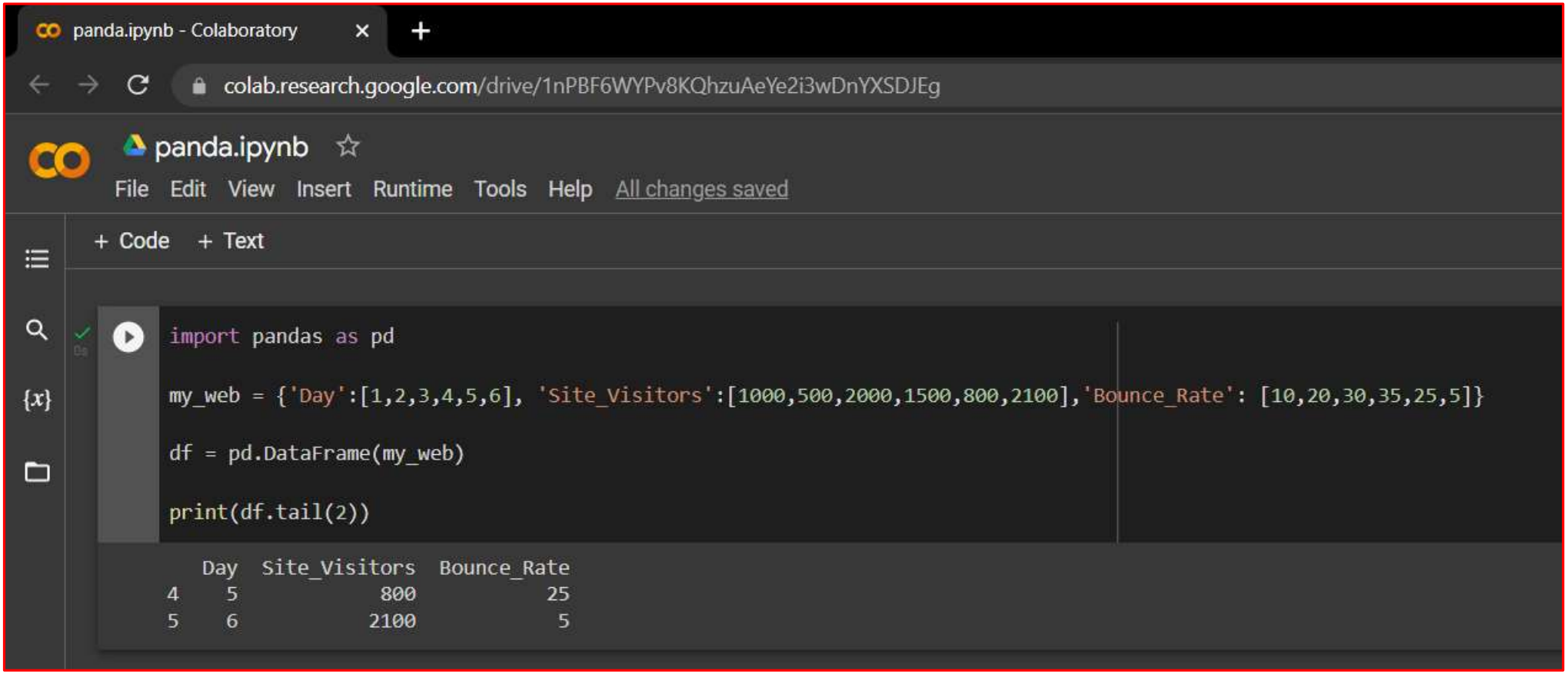


Data Slicing using pandas



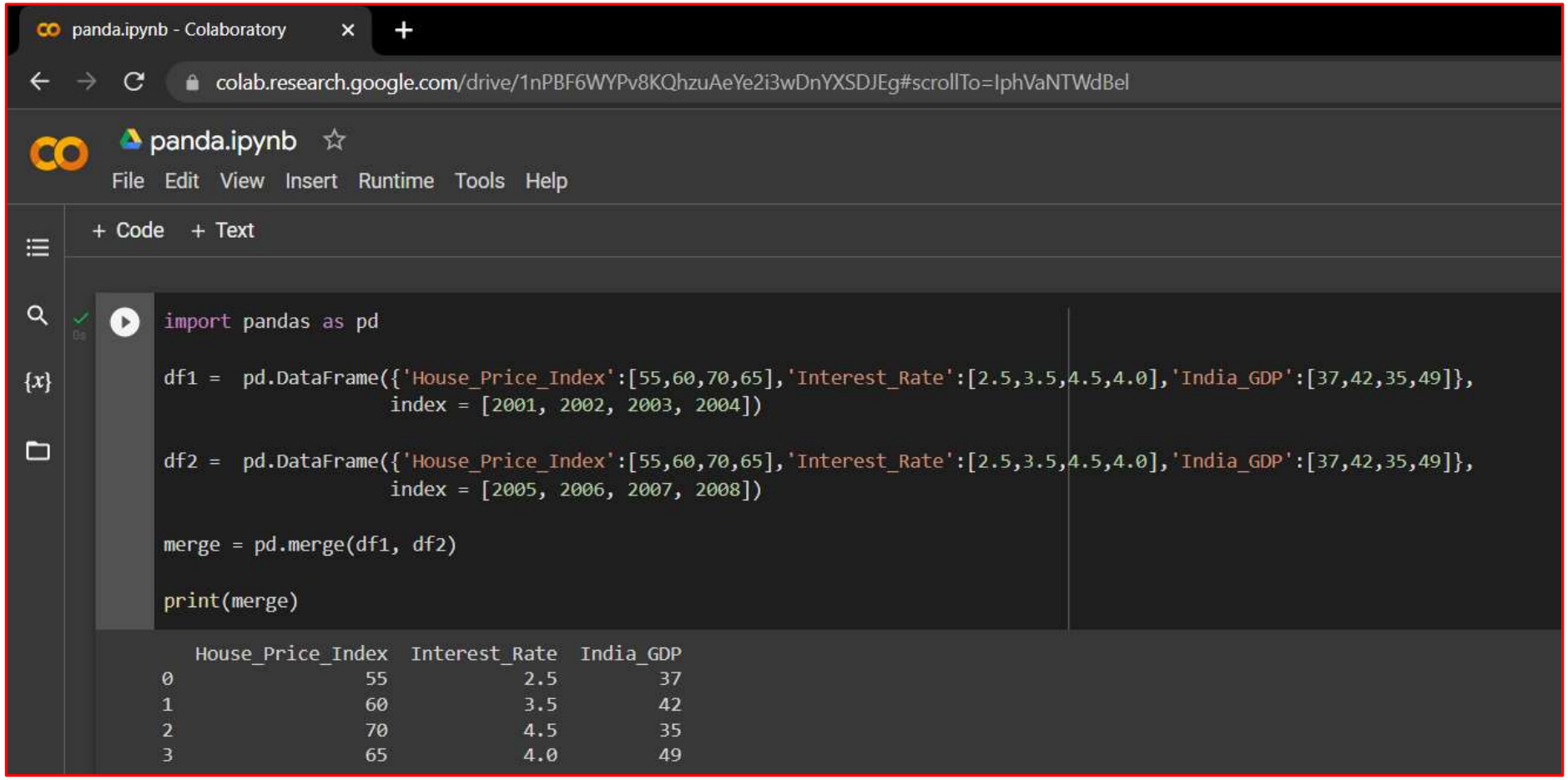
Data Slicing using pandas

www.hbpatel.in



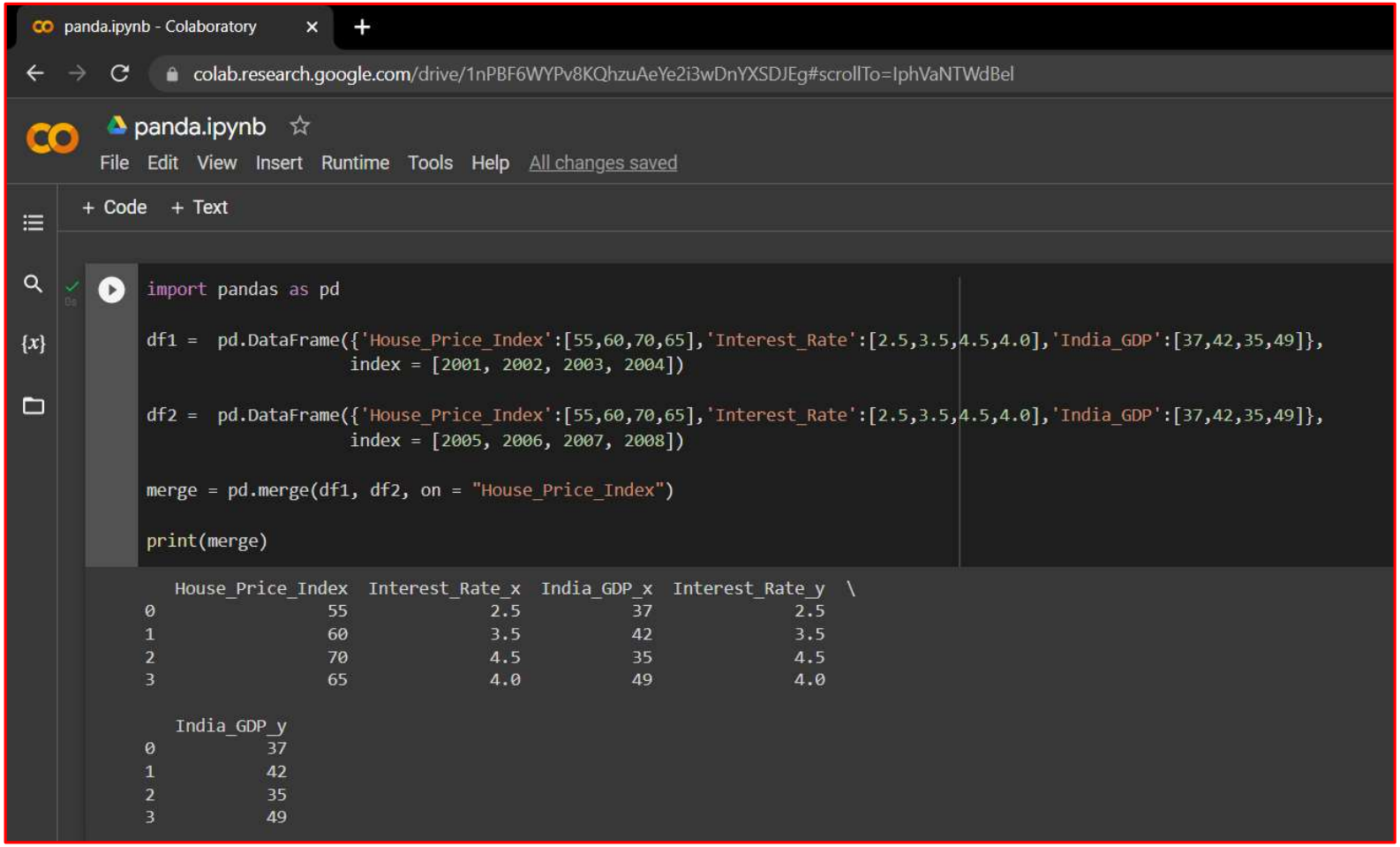
Data Merging using pandas

www.hbpatel.in



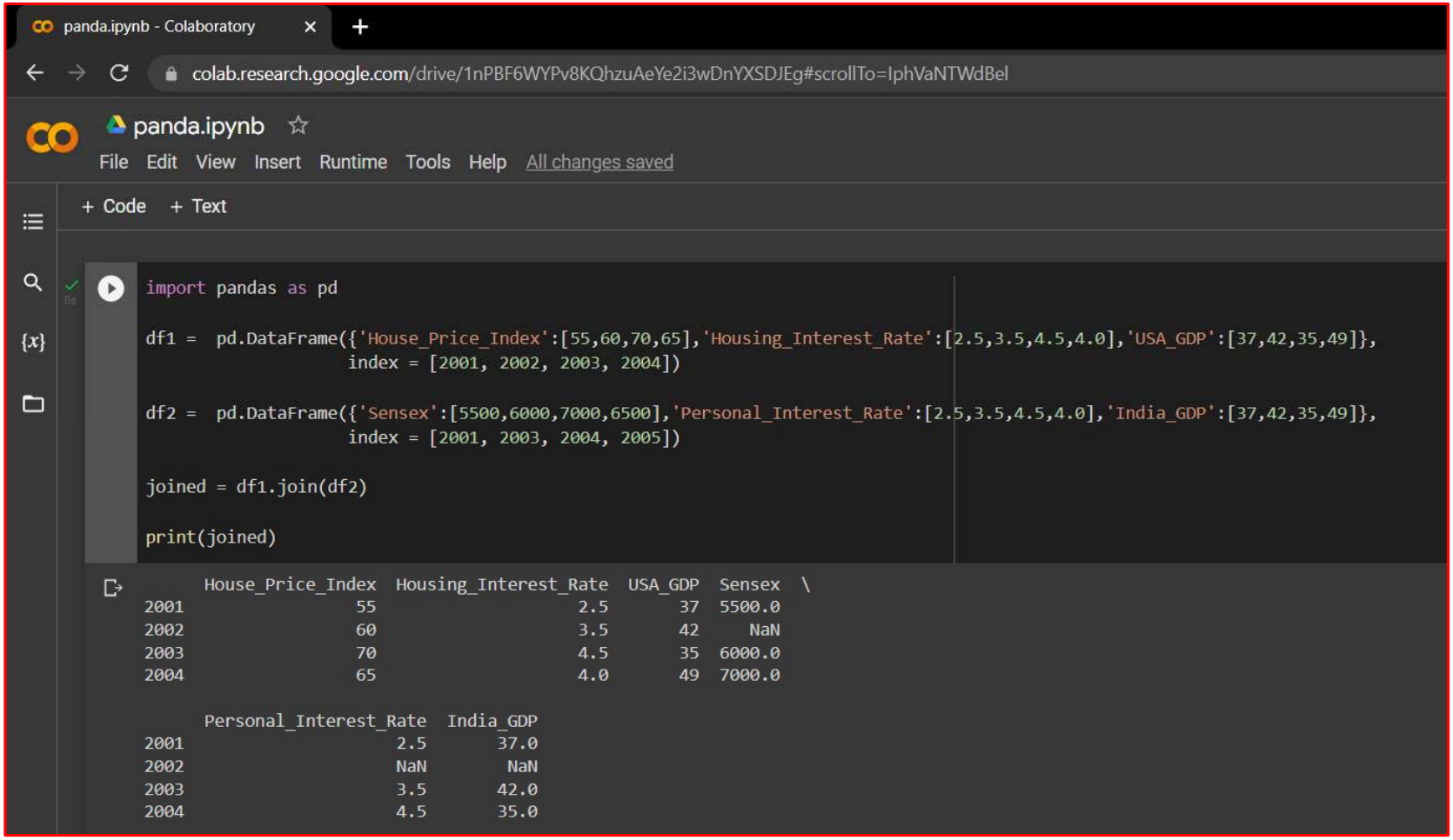
Data Merging using pandas

www.hbpatel.in



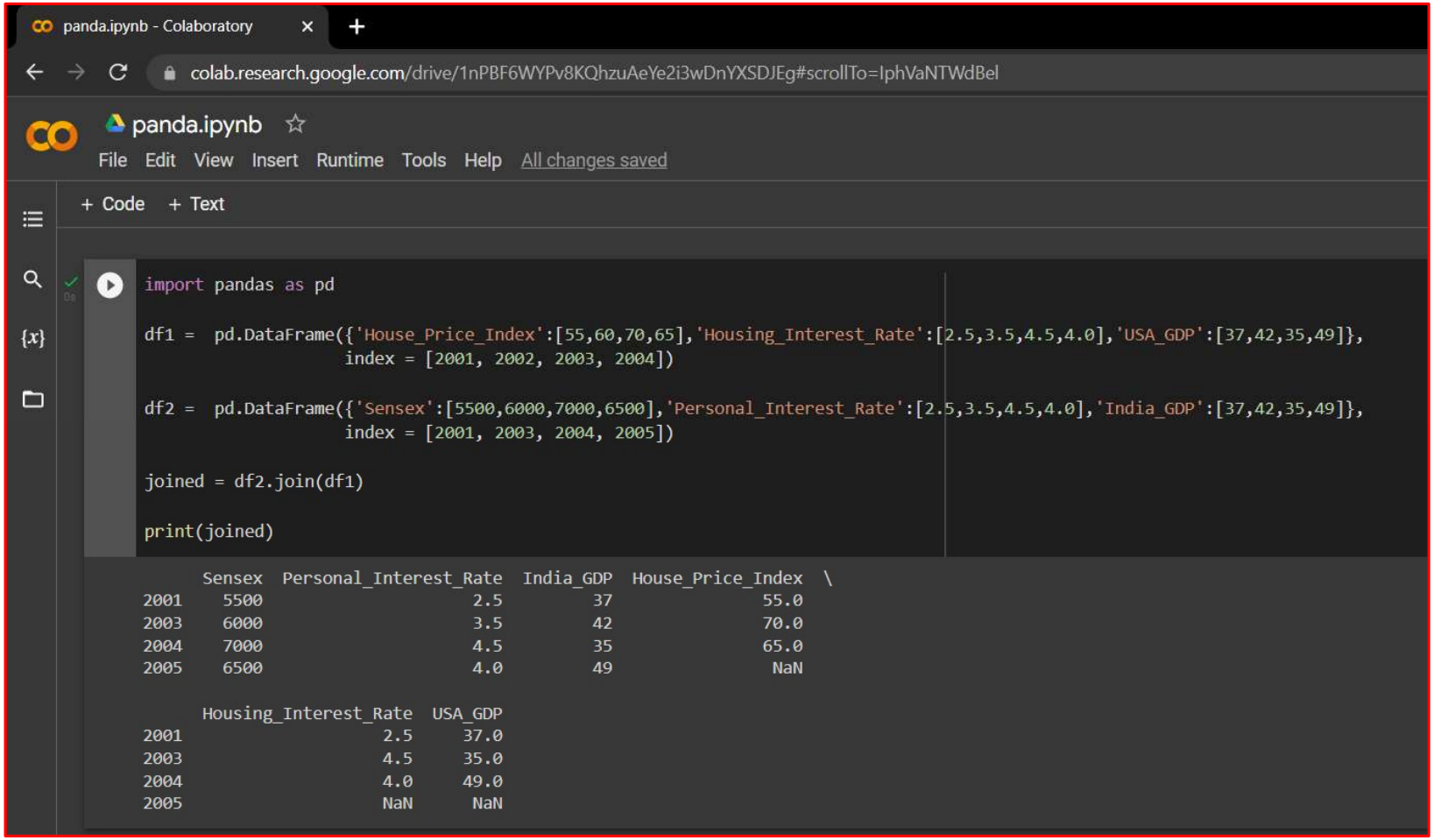
Data Joining using pandas

www.hbpatel.in

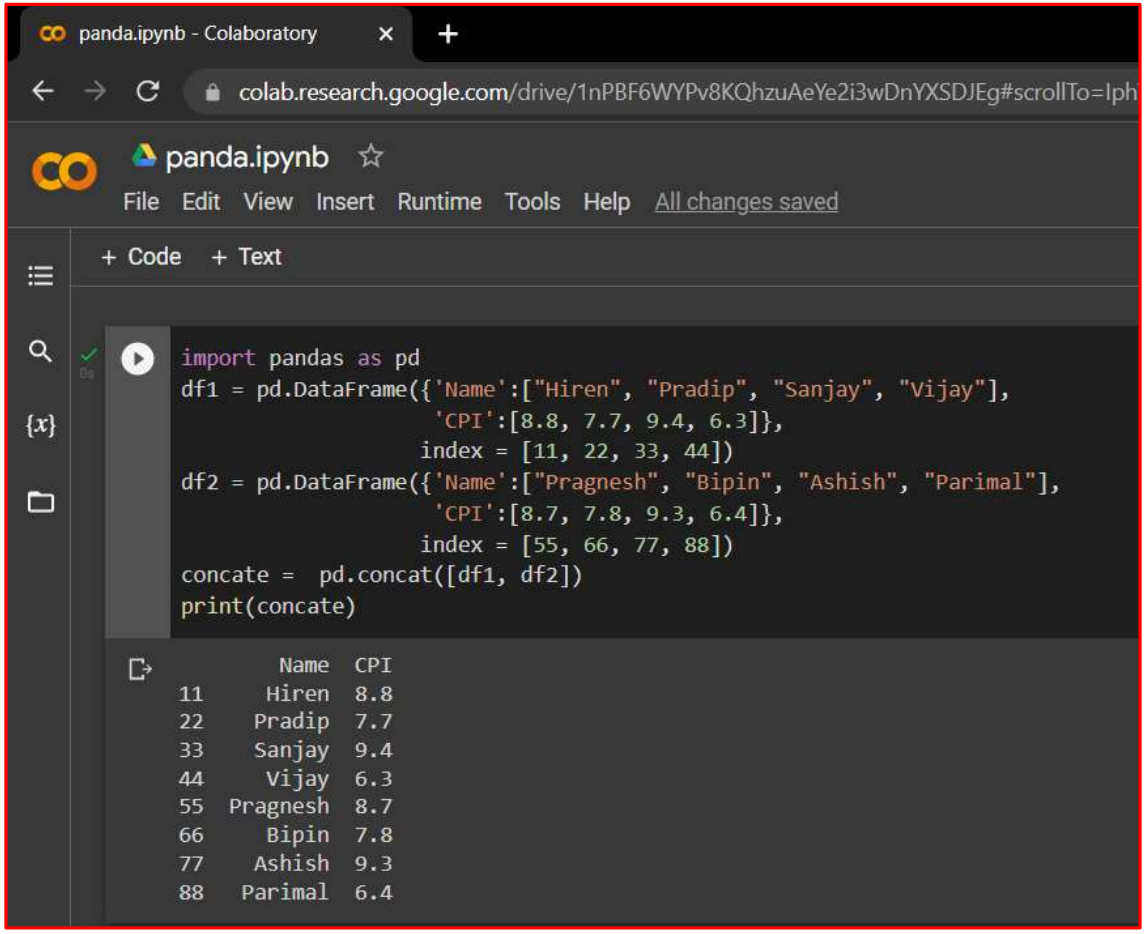


Data Joining using pandas

www.hbpatel.in



Data Concatenation using pandas



Data Correlation using pandas


```
import pandas as pd
data = {
'Name': ['Hiren', 'Sanjay', 'Pradip', 'Vijay', 'Parimal'],
'Age': [46, 43, 36, 39, 28],
'Qualification': [10, 5, 9, 4, 6],
'Income': [1000, 200, 600, 100, 300],
'Height': [5.9, 5.6, 5.8, 5.5, 5.4],
'Weight': [83, 56, 62, 50, 60]}
df = pd.DataFrame(data)
print(df)
print('='*10)
correlation_matrix = df.corr()
print(correlation_matrix)
print('='*10)
correlation_matrix['Income']
```

```
   Name  Age  Qualification  Income  Height  Weight
0   Hiren   46             10    1000     5.9      83
1  Sanjay   43              5     200     5.6      56
2  Pradip   36              9     600     5.8      62
3   Vijay   39              4     100     5.5      50
4  Parimal  28              6     300     5.4      60
==========
                    Age  Qualification    Income    Height    Weight
Age            1.000000       0.241812  0.396521  0.662667  0.419071
Qualification  0.241812       1.000000  0.964003  0.857011  0.874799
Income         0.396521       0.964003  1.000000  0.866126  0.963159
Height         0.662667       0.857011  0.866126  1.000000  0.758207
Weight         0.419071       0.874799  0.963159  0.758207  1.000000
==========
Age              0.396521
Qualification    0.964003
Income           1.000000
Height           0.866126
Weight           0.963159
Name: Income, dtype: float64
```

Principal Component Analysis (PCA) using pandas

PCA is a process of figuring out most important features or principal components that has the most impact on the target variable. (Following program is incomplete)

```
import pandas as pd
from matplotlib import pyplot as plt
import numpy as np

property = {
'Town': ['Ahmedabad', 'Ahmedabad', 'Ahmedabad', 'Ahmedabad', 'Ahmedabad', 'Baroda', 'Baroda', 'Baroda', 'Baroda'],
'ConstructionalArea': [2600, 3000, 3200, 3600, 4000, 2600, 2800, 3300, 3600],
'Bedroom': [2, 3, 3, 4, 4, 2, 3, 4, 4],
'PlotArea': [7500, 9200, 9700, 10500, 11900, 8000, 8500, 10000, 10800],
'TreesNearby': [2, 2, 1, 2, 2, 1, 1, 2, 2],
'Price': [18000000, 22500000, 24000000, 26500000, 31000000, 15000000, 15500000, 18000000, 19800000]
}
```

Principal Component Analysis (PCA) using pandas

PCA is a process of figuring out most important features or principal components that has the most impact on the target variable. (Following program is incomplete)

```
df = pd.DataFrame(property)
print(df)
print('-'*20)
print(df.Town)
print('+'*20)
print(df.ConstructionalArea)
print('='*20)
print(df.keys())
print('@'*20)
print(np.unique(df.Town))
print('#'*20)
df.describe()
```

```
      Town  ConstructionalArea  Bedroom  PlotArea  TreesNearby      Price
0  Ahmedabad                2600        2      7500            2   18000000
1  Ahmedabad                3000        3      9200            2   22500000
2  Ahmedabad                3200        3      9700            1   24000000
3  Ahmedabad                3600        4     10500            2   26500000
4  Ahmedabad                4000        4     11900            2   31000000
5     Baroda                2600        2      8000            1   15000000
6     Baroda                2800        3      8500            1   15500000
7     Baroda                3300        4     10000            2   18000000
8     Baroda                3600        4     10800            2   19800000
--------------------
0    Ahmedabad
1    Ahmedabad
2    Ahmedabad
3    Ahmedabad
4    Ahmedabad
5       Baroda
6       Baroda
7       Baroda
8       Baroda
Name: Town, dtype: object
++++++++++++++++++++
```

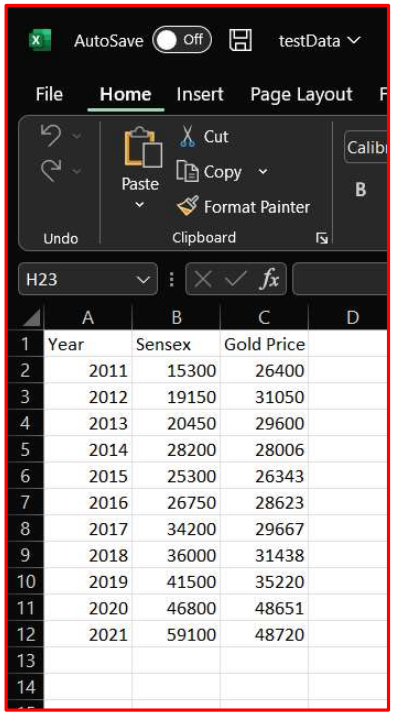
Principal Component Analysis (PCA) using pandas

PCA is a process of figuring out most important features or principal components that has the most impact on the target variable. (Following program is incomplete)

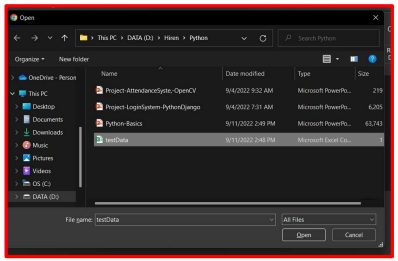
```
df = pd.DataFrame(property)
print(df)
print('-'*20)
print(df.Town)
print('+'*20)
print(df.ConstructionalArea)
print('='*20)
print(df.keys())
print('@'*20)
print(np.unique(df.Town))
print('#'*20)
df.describe()
```

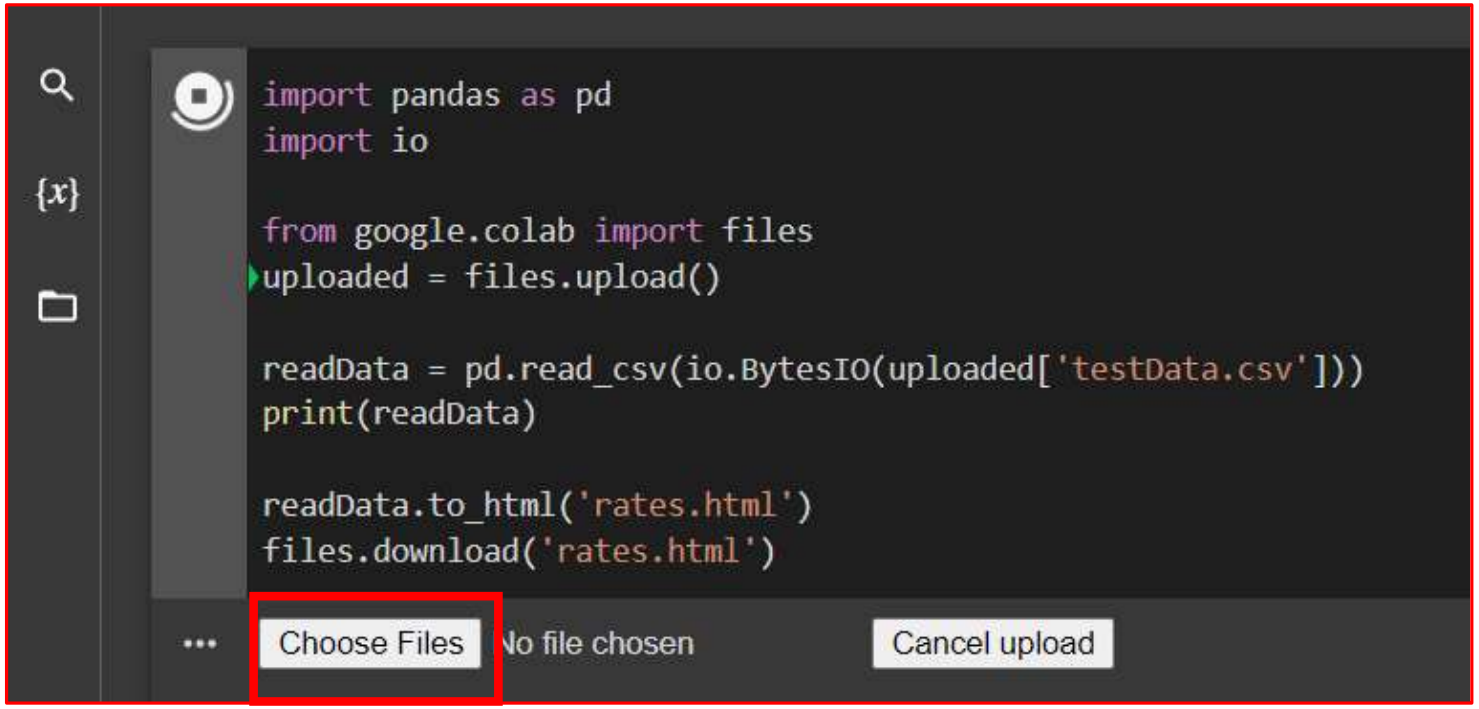
```
++++++++++++++++++++
0    2600
1    3000
2    3200
3    3600
4    4000
5    2600
6    2800
7    3300
8    3600
Name: ConstructionalArea, dtype: int64
====================
Index(['Town', 'ConstructionalArea', 'Bedroom', 'PlotArea', 'TreesNearby',
       'Price'],
      dtype='object')
@@@@@@@@@@@@@@@@@@@@
['Ahmedabad' 'Baroda']
####################
             ConstructionalArea      Bedroom     PlotArea     TreesNearby             Price
count        9.000000                9.000000    9.000000     9.000000                9.000000e+00
mean         3188.888889             3.222222    9566.666667 1.666667                2.114444e+07
std          485.912658              0.833333    1415.9802260.500000                 5.326845e+06
min          2600.000000             2.000000    7500.000000 1.000000                1.500000e+07
25%          2800.000000             3.000000    8500.000000 1.000000                1.800000e+07
50%          3200.000000             3.000000    9700.000000 2.000000                1.980000e+07
75%          3600.000000             4.000000    10500.0000002.000000                2.400000e+07
max          4000.000000             4.000000    11900.0000002.000000                3.100000e+07
```

Convert CSV to HTML using pandas



	A	B	C	D
1	Year	Sensex	Gold Price	
2	2011	15300	26400	
3	2012	19150	31050	
4	2013	20450	29600	
5	2014	28200	28006	
6	2015	25300	26343	
7	2016	26750	28623	
8	2017	34200	29667	
9	2018	36000	31438	
10	2019	41500	35220	
11	2020	46800	48651	
12	2021	59100	48720	
13				
14				





```
import pandas as pd
import io

from google.colab import files
uploaded = files.upload()

readData = pd.read_csv(io.BytesIO(uploaded['testData.csv']))
print(readData)

readData.to_html('rates.html')
files.download('rates.html')
```

Convert CSV to HTML using pandas

www.hbpatel.in

```
import pandas as pd
import io

from google.colab import files
uploaded = files.upload()

readData = pd.read_csv(io.BytesIO(uploaded['testData.csv']))
print(readData)

readData.to_html('rates.html')
files.download('rates.html')
```

Choose Files testData.csv

- **testData.csv**(text/csv) - 225 bytes, last modified: 9/11/2022 - 100% done

```
Saving testData.csv to testData (5).csv
    Year  Sensex  Gold Price
0   2011   15300       26400
1   2012   19150       31050
2   2013   20450       29600
3   2014   28200       28006
4   2015   25300       26343
5   2016   26750       28623
6   2017   34200       29667
7   2018   36000       31438
8   2019   41500       35220
9   2020   46800       48651
10  2021   59100       48720
Downloading "rates.html":
```

rates.html

File | C:/Users/Hiren%20Patel/Downloads/rates.html

	Year	Sensex	Gold Price
0	2011	15300	26400
1	2012	19150	31050
2	2013	20450	29600
3	2014	28200	28006
4	2015	25300	26343
5	2016	26750	28623
6	2017	34200	29667
7	2018	36000	31438
8	2019	41500	35220
9	2020	46800	48651
10	2021	59100	48720

Plotting the Graphs using pandas

www.hbpatel.in

```
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib import style
import io
from google.colab import files

uploaded = files.upload()
readData = pd.read_csv(io.BytesIO(uploaded['testData
.csv']))
print(readData)

readData = readData.set_index(["Year"])

readData.plot(kind='bar')
plt.show()
```