# DATA HANDLING USING PANDAS – I
PROGRAMS

Write a Pandas program to multiple and divide two Pandas Series.
Sample Series: [2, 4, 8, 10], [1, 3, 7, 9]

```
import pandas as pd
ds1 = pd.Series([2, 4, 8, 10])
ds2 = pd.Series([1, 3, 7, 9])
print("Multiply two Series:")
ds = ds1 * ds2
print(ds)
print("Divide Series1 by Series2:")
ds = ds1 / ds2
print(ds)
```

Write a Pandas program to convert a dictionary to a Pandas series.
Sample dictionary: d1 = {'a': 100, 'b': 200, 'c':300}

```
import pandas as pd
d1 = {'a': 100, 'b': 200, 'c':300}
print("Original dictionary:")
print(d1)
new_series = pd.Series(d1)
print("Converted series:")
print(new_series)
```

Write a Pandas program to sort a given Series.
400, 300.12,100, 200

```python
import pandas as pd
s = pd.Series([400, 300.12,100, 200])
print("Original Data Series:")
print(s)
new_s = pd.Series(s).sort_values()
print(new_s)
```

Write a Pandas program to change the order of index of a given series.
Original Data Series:
A    1
B    2
C    3
dtype: int64
Data Series after changing the order of index:
B    2
A    1
C    3
dtype: int64

```python
import pandas as pd
s = pd.Series(data = [1,2,3], index = ['A', 'B', 'C'])
print("Original Data Series:")
print(s)
s = s.reindex(index = ['B','A','C'])
print("Data Series after changing the order of index:")
print(s)
```

Write a Pandas program to get the items which are not common of two given series.

```
import pandas as pd
import numpy as np
sr1 = pd.Series([1, 2, 3])
sr2 = pd.Series([2, 3, 6])
print("Original Series:")
print("sr1:")
print(sr1)
print("sr2:")
print(sr2)
print("\nItems of a given series not present in another given series:")
sr11 = pd.Series(np.union1d(sr1, sr2))
sr22 = pd.Series(np.intersect1d(sr1, sr2))
result = sr11[~sr11.isin(sr22)]
print(result)
```

Write a Pandas program to create and display a DataFrame from a specified dictionary with index labels.

```
import pandas as pd
import numpy as np
exam_data  = {'name': ['Manish', 'Dhiraj'],
      'score': [12.5, 9]}
labels = ['NAME', 'SCORE']
df = pd.DataFrame(exam_data , index=labels)
print(df)
```

## Write a Pandas program to get the first 3 rows of a given DataFrame.

```
import pandas as pd
import numpy as np
exam_data  = {'name': ['Manish', 'Dhiraj','Man', 'Dhir'],
      'score': [12.5, 91,2.5, 9]}
df = pd.DataFrame(exam_data )
print("First three rows of the data frame:")
print(df.iloc[:3])   #print(df.head(3))
```

## Write a Pandas program to count the number of rows and columns of a DataFrame.

```
import pandas as pd
import numpy as np
exam_data  = {'name': ['Manish', 'Dhiraj','Man', 'Dhir'],
      'score': [12.5, 91,2.5, 9]}
df = pd.DataFrame(exam_data )
total_rows=len(df.axes[0])
total_cols=len(df.axes[1])
print("Number of Rows: "+str(total_rows))
print("Number of Columns: "+str(total_cols))
```

## Write a Pandas program to select the rows the score is between 15 and 20 (inclusive)

```python
import pandas as pd
import numpy as np
exam_data  = {'name': ['Manish', 'Dhiraj','Man', 'Dhir'],
       'score': [12.5, 91,20.5, 19]}
df = pd.DataFrame(exam_data )
print("Rows where score between 15 and 20 (inclusive):")
print(df[df['score'].between(15, 20)])
```

## Write a Pandas program to sort the DataFrame first by 'name' in descending order, then by 'score' in ascending order.

```python
import pandas as pd
import numpy as np
exam_data  = {'name': ['Manish', 'Dhiraj','Man', 'Dhir'],
       'score': [12.5, 91,20.5, 19]}
df = pd.DataFrame(exam_data )
result_sort=df.sort_values(by=['name', 'score'], ascending=[True, True])
print("Sort the data frame first by 'name' in descending order, then by 'score' in ascending order:")
print(result_sort)
```

## Write a Pandas program to change the name 'Manish' to 'Anish' in name column of the data frame.

```
import pandas as pd
import numpy as np
exam_data  = {'name': ['Manish', 'Dhiraj','Man', 'Dhir'],
      'score': [12.5, 91,20.5, 19]}
df = pd.DataFrame(exam_data )
df['name'] = df['name'].replace('Manish', 'Anish')
print(df)
```

## Write a Pandas program to insert a new column in existing DataFrame.

```
import pandas as pd
import numpy as np
exam_data  = {'name': ['Manish', 'Dhiraj','Man', 'Dhir'],
      'score': [12.5, 91,20.5, 19]}
df = pd.DataFrame(exam_data )
medium = ['english','hindi','hindi','english']
df['medium'] = medium
print("\nNew DataFrame after inserting the 'medium' column")
print(df)
```

Write a Pandas program to rename columns of a given DataFrame.

```
import pandas as pd
import numpy as np
exam_data  = {'name': ['Manish', 'Dhiraj','Man', 'Dhir'],
      'score': [12.5, 91,20.5, 19]}
df = pd.DataFrame(exam_data )
df = df.rename(columns={'name': 'NAME', 'score': 'SCORE'})
print("New DataFrame after renaming columns:")
print(df)
```

Write a Pandas program to delete DataFrame row(s) based on given column value/condition.

```
import pandas as pd
import numpy as np
exam_data  = {'name': ['Manish', 'Dhiraj','Man', 'Dhir'],
      'score': [12.5, 91,20.5, 19]}
df = pd.DataFrame(exam_data )
df = df[df.score >= 20]
print("New DataFrame")
print(df)
```

## Write a Pandas program to combining two series into a DataFrame.

```python
import pandas as pd
import numpy as np
s1 = pd.Series(['100', '200', '400'])
s2 = pd.Series(['10', '20', '40'])
print("Data Series:")
print(s1)
print(s2)
df = pd.concat([s1, s2], axis=1)
print("New DataFrame combining two series:")
print(df)
```

## Write a Pandas program to get the specified row value of a given DataFrame.

```python
import pandas as pd
import numpy as np
s1 = pd.Series(['100', '200', '400'])
s2 = pd.Series(['10', '20', '40'])
df = pd.concat([s1, s2], axis=1)
print("Value of Row4")
print(df.iloc[2])
```

Write a Pandas program to insert a given column at a specific column index in a DataFrame.

```python
import pandas as pd
import numpy as np
s1 = pd.Series(['100', '200', '400'])
s2 = pd.Series(['10', '20', '40'])
df = pd.concat([s1, s2], axis=1)
new_col = [1, 2, 3]
# insert the said column at the beginning in the DataFrame
idx = 0
df.insert(loc=idx, column='1', value=new_col)
print("\nNew DataFrame")
print(df)
```