

OUTPUTS:

Question1:

A)

```
In [7]: import numpy as np
import pandas as pd
import random
```

```
In [4]: # creating a series with given elements
ds = pd.Series([7,11,13,17])
print(ds)

0      7
1     11
2     13
3     17
dtype: int64
```

```
In [5]: # creating series with five elements that all are 100
ds1 = pd.Series([100,100,100,100,100])
print(ds1)

0     100
1     100
2     100
3     100
4     100
dtype: int64
```

```
In [11]: # creating series from random numbers between(1,100)
ds2 = pd.Series(random.sample(range(0,100), 20))
# Statistics of the numbers selected randomly
print(ds2.describe())

count    20.000000
mean     49.500000
std      31.050892
min       7.000000
25%     23.500000
50%     42.500000
75%     76.250000
max      97.000000
dtype: float64
```

```
In [12]: # creating the Series temperatures from the given values and index
Temperatures = pd.Series(np.array([98.6, 98.9, 100.2,97.9]),index=["Julie","Charlie", 'Sam' , 'Andrea'])
print(Temperatures)

Julie      98.6
Charlie    98.9
Sam       100.2
Andrea     97.9
dtype: float64
```

```
In [13]: # creating Series from dictionary which takes the values and index from above question.
temp_dict = pd.Series({"Julie" : 98.6,"Charlie": 98.9 ,"Sam": 100.2, "Andrea": 97.9})
print(temp_dict)

Julie      98.6
Charlie    98.9
Sam       100.2
Andrea     97.9
dtype: float64
```

B)

```
In [60]: # Importing numpy and pandas modules
import numpy as np
import pandas as pd
```

```
In [61]: # Creating DataFrame from the Dictionary we created
temperatures = pd.DataFrame({"Maxine": [98, 96, 94.6], "James": [100.2, 92.8, 98.7], "Amanda": [94.8, 97.4, 97.4]},
                             index=[1, 2, 3])
print(temperatures)
```

	Maxine	James	Amanda
1	98.0	100.2	94.8
2	96.0	92.8	97.4
3	94.6	98.7	97.4

```
In [62]: # Changing the index values to Morning, Afternoon, Evening
temperatures.index = ["Morning", "Afternoon", "Evening"]
print(temperatures)
```

	Maxine	James	Amanda
Morning	98.0	100.2	94.8
Afternoon	96.0	92.8	97.4
Evening	94.6	98.7	97.4

```
In [63]: # Selecting the temperatures of column Maxine
print(temperatures.loc[:, 'Maxine'])
```

```
Morning    98.0
Afternoon   96.0
Evening     94.6
Name: Maxine, dtype: float64
```

```
In [64]: # Selecting the temperatures of Row Morning
print(temperatures.loc['Morning', :])
```

```
Maxine    98.0
James     100.2
Amanda     94.8
Name: Morning, dtype: float64
```

```
In [65]: # Selecting the temperatures of rows Morning and Evening
print(temperatures.loc[['Morning', 'Evening'], :])
```

	Maxine	James	Amanda
Morning	98.0	100.2	94.8
Evening	94.6	98.7	97.4

```
In [66]: # Selecting the temperatures of columns Maxine and Amanda
print(temperatures.loc[:, ['Maxine', 'Amanda']])
```

	Maxine	Amanda
Morning	98.0	94.8
Afternoon	96.0	97.4
Evening	94.6	97.4

```
In [67]: # selecting temperature value of morning, afternoon rows and amanda and maxine columns
print(temperatures)
print(temperatures.loc[['Morning', 'Afternoon'], ['Maxine', 'Amanda']])
```

	Maxine	James	Amanda
Morning	98.0	100.2	94.8
Afternoon	96.0	92.8	97.4
Evening	94.6	98.7	97.4

	Maxine	Amanda
Morning	98.0	94.8
Afternoon	96.0	97.4

```
In [68]: # Statistics of the temperatures
temperatures.describe()
```

Out[68]:

	Maxine	James	Amanda
count	3.000000	3.000000	3.000000
mean	96.200000	97.233333	96.533333
std	1.708801	3.911948	1.501111
min	94.600000	92.800000	94.800000
25%	95.300000	95.750000	96.100000
50%	96.000000	98.700000	97.400000
75%	97.000000	99.450000	97.400000
max	98.000000	100.200000	97.400000

```
In [69]: # Transpose the index to columns and columns to index for temperatures dataframe
Temp_transpose = temperatures.transpose()
print(Temp_transpose)
```

	Morning	Afternoon	Evening
Maxine	98.0	96.0	94.6
James	100.2	92.8	98.7
Amanda	94.8	97.4	97.4

```
In [70]: # Sorting the temperatures dataframe columns in alphabetical order
temp_sort=temperatures.sort_index(axis=1)
print(temp_sort)
```

	Amanda	James	Maxine
Morning	94.8	100.2	98.0
Afternoon	97.4	92.8	96.0
Evening	97.4	98.7	94.6

Question 2 : Outputs of Titanic question:

```
In [15]: #Importing the modules required for this program
import csv
import pandas as pd
import numpy as np
```

```
In [16]: #Importing the CSV file
data = pd.read_csv('/Users/bhanuchanderkureti/Bhanu/ITMD_513/Assignments/Pandas/titanic.csv')
df = pd.DataFrame(data)
```

```
In [17]: #List columns and rows of data frame
df.head()
```

```
Out[17]:
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

```
In [92]: #Total number of passengers
print("Total number of passengers:\n" + str(df.PassengerId.count()))

Total number of passengers:
891
```

```
In [93]: # male and female passengers count
print("Total number of male and female passengers:\n" + str(df['Sex'].value_counts()))

Total number of male and female passengers:
male      577
female    314
Name: Sex, dtype: int64
```

```
In [91]: #Average age of all populations
print("Average age of passengers:\n" + str(df.Age.mean()))

Average age of passengers:
29.69911764705882
```

```
In [94]: # Number of Passengers under Age 21
child = df[df.Age < 21.00]
print("The Number of Passengers less than age 21 :\n" + str(len(child)))

The Number of Passengers less than age 21 :
180
```

```
In [95]: # Assuming If Survive = 1 as survived and Survived = 0 as Un Survived

print("Sum of Survived: "+ str(sum(df["Survived"]==0))) # Number of Survived
print("Sum of UnSurvived: "+ str(sum(df["Survived"]==1))) # Number of UnSurvived
print("Sum of males Survived: "+ str(sum((df["Survived"]==1) & (df["Sex"]=="male")))) # Number of male's Survived
print("Sum of females Survived: "+ str(sum((df["Survived"]==1) & (df["Sex"]=="female")))) # Number of female's Survived
print("Sum of males unSurvived: "+ str(sum((df["Survived"]==0) & (df["Sex"]=="male")))) # Number of male's UnSurvived
print("Sum of females UnSurvived: "+ str(sum((df["Survived"]==0) & (df["Sex"]=="female")))) # Number of female's UnSurvived
```

```
Sum of Survived: 549
Sum of UnSurvived: 342
Sum of males Survived: 109
Sum of females Survived: 233
Sum of males unSurvived: 468
Sum of females UnSurvived: 81
```

```
In [98]: df_sur = df[df["Survived"]==1]
df_sur = df_sur.loc[df_sur["Age"]==min(df_sur.Age)]
print('Youngest age to survive the Titanic mishap is :\n'+ str(df_sur.Age.to_string(index=False)) + str(df_sur["Name"])).
```

```
Youngest age to survive the Titanic mishap is :
0.42 Thomas, Master. Assad Alexander
```

```
Youngest age to survive the Titanic mishap is :
0.42 Thomas, Master. Assad Alexander
```

```
In [97]: df_sur = df[df["Survived"]==1]
df_sur = df_sur.loc[df_sur["Age"]==max(df_sur.Age)]
print('oldest age to survive the Titanic mishap is :\n'+ str(df_sur.Age.to_string(index=False)) + str(df_sur["Name"])).
```

```
oldest age to survive the Titanic mishap is :
80.0 Barkworth, Mr. Algernon Henry Wilson
```

```
In [68]: print('\ng) Names of all the people who survived:\n'+ str(df.loc[(df['Survived']==1), 'Name']))
```

```
g) Names of all the people who survived:
1      Cumings, Mrs. John Bradley (Florence Briggs Th...
2                               Heikkinen, Miss. Laina
3      Futrelle, Mrs. Jacques Heath (Lily May Peel)
8      Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)
9      Nasser, Mrs. Nicholas (Adele Achem)
...
875     Najib, Miss. Adele Kiamie "Jane"
879     Potter, Mrs. Thomas Jr (Lily Alexenia Wilson)
880     Shelley, Mrs. William (Imanita Parrish Hall)
887     Graham, Miss. Margaret Edith
889     Behr, Mr. Karl Howell
Name: Name, Length: 342, dtype: object
```

In [1]:

