OUTPUTS

Complete all following Task:

• Dataset for the Task: "bank.csv"

1. Load the provided dataset and import in pandas DataFrame.

2. Check info of the DataFrame and identify following:

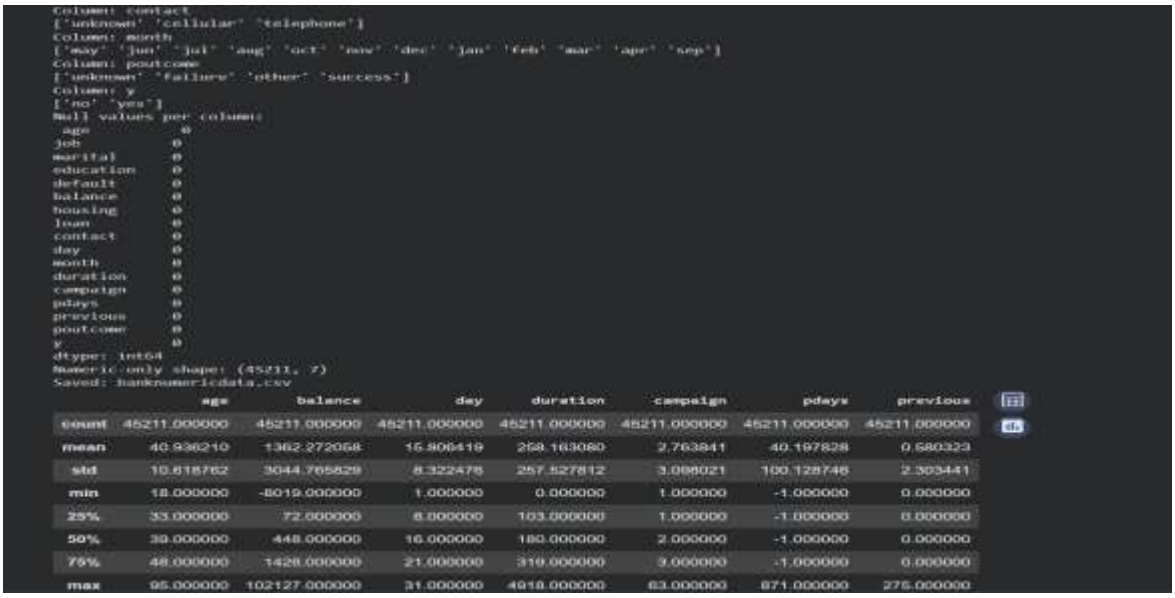
(a) columns with dtypes=object

(b) unique values of those columns.

(c) check for the total number of null values in each column.

3. Drop all the columns with dtypes object and store in new DataFrame, also write the DataFrame in ".csv" with name "banknumericdata.csv"

4. Read "banknumericdata.csv" and Find the summary statistics.



	age	balance	day	duration	campaign	pdays	previous
count	45211.000000	45211.000000	45211.000000	45211.000000	45211.000000	45211.000000	45211.000000
mean	40.936210	1362.272058	15.806419	258.163080	2.763841	40.197828	0.580323
std	10.618762	3044.765829	8.322476	257.527812	3.098021	100.128746	2.303441
min	18.000000	-8019.000000	1.000000	0.000000	1.000000	-1.000000	0.000000
25%	33.000000	72.000000	8.000000	103.000000	1.000000	-1.000000	0.000000
50%	39.000000	448.000000	16.000000	180.000000	2.000000	-1.000000	0.000000
75%	48.000000	1428.000000	21.000000	319.000000	3.000000	-1.000000	0.000000
max	95.000000	102127.000000	31.000000	4918.000000	63.000000	871.000000	275.000000

Problem 2- Data Imputations:

Complete all the following Task:

• Dataset for the Task: "medical_student.csv"

1. Load the provided dataset and import in pandas DataFrame.

2. Check info of the DataFrame and identify column with missing (null) values.

3. For the column with missing values fill the values using various techniques we discussed above. Try to explain why did you select the particular methods for particular column.

4. Check for any duplicate values present in Dataset and do necessary to manage the duplicate items.

{Hint: dataset.duplicated.sum()

```
MEDICAL DATASET SHAPE: (200000, 13)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200000 entries, 0 to 199999
Data columns (total 13 columns):
 #   Column          Non-Null Count   Dtype
---  ------          --------------   -----
 0   Student ID      180000 non-null  float64
 1   Age             180000 non-null  float64
 2   Gender          180000 non-null  object
 3   Height          180000 non-null  float64
 4   Weight          180000 non-null  float64
 5   Blood Type      180000 non-null  object
 6   BMI             180000 non-null  float64
 7   Temperature     180000 non-null  float64
 8   Heart Rate      180000 non-null  float64
 9   Blood Pressure  180000 non-null  float64
 10  Cholesterol     180000 non-null  float64
 11  Diabetes        180000 non-null  object
 12  Smoking         180000 non-null  object
dtypes: float64(9), object(4)
memory usage: 19.8+ MB

Missing values:
 Student ID        20000
Age                20000
Gender             20000
Height             20000
Weight             20000
Blood Type         20000
BMI                20000
Temperature        20000
Heart Rate         20000
Blood Pressure     20000
Cholesterol        20000
Diabetes           20000
Smoking            20000
dtype: int64

After imputation:
 Student ID        0
Age                0
Gender             0
Height             0
Weight             0
Blood Type         0
BMI                0
Temperature        0
Heart Rate         0
```

```
Blood Pressure     0
Cholesterol        0
Diabetes           0
Smoking            0
dtype: int64
Duplicates: 12879
Shape after removing duplicates: (187121, 13)
Saved: medical_student_cleaned.csv
```

Problem- 3:

The ’Embarked’ column in the Titanic dataset contains categorical data representing the ports of embarka tion:

• ’C’ for Cherbourg

• ’Q’ for Queenstown

• ’S’ for Southampton

Task:

1. Use one-hot encoding to convert the ’Embarked’ column into separate binary columns (’Embarked C’, ’Embarked Q’, ’Embarked S’).

2. Add these new columns to the original DataFrame.

3. Drop the original ’Embarked’ column.

4. Print the first few rows of the modified DataFrame to verify the changes

```
TITANIC SHAPE: (891, 12)

First class count: (216, 7)

Fare Stats for Pclass 1:
Mean: 84.1546875
Median: 60.287499999999994
Min: 0.0
Max: 512.3292

Null Age count: 30
Shape after dropping null Age: (186, 7)
```

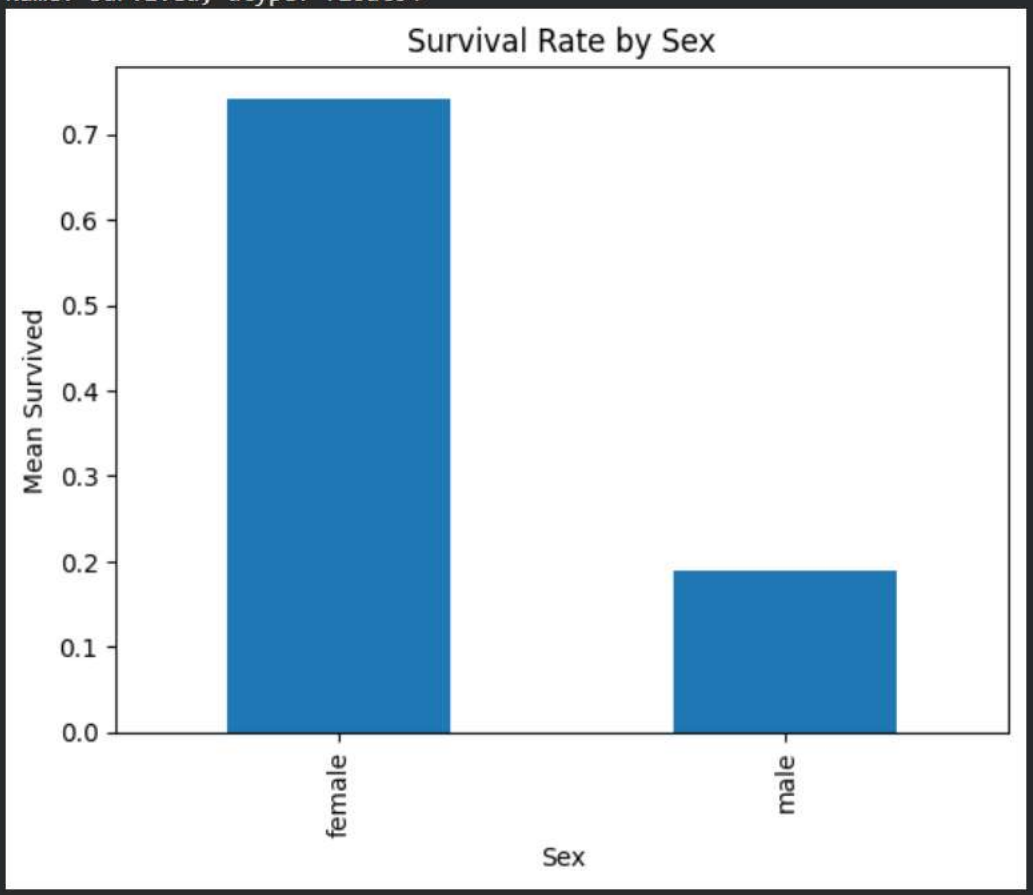
```
Saved: titanic_with_onehot_embarked.csv
```

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

Problem- 4:

Compare the mean survival rates (’Survived’) for the different groups in the ’Sex’ column. Draw a visual ization to show how the survival distributions vary by gender

```
Survival rate by sex:
 Sex
female    0.742038
male      0.188908
Name: Survived, dtype: float64
```

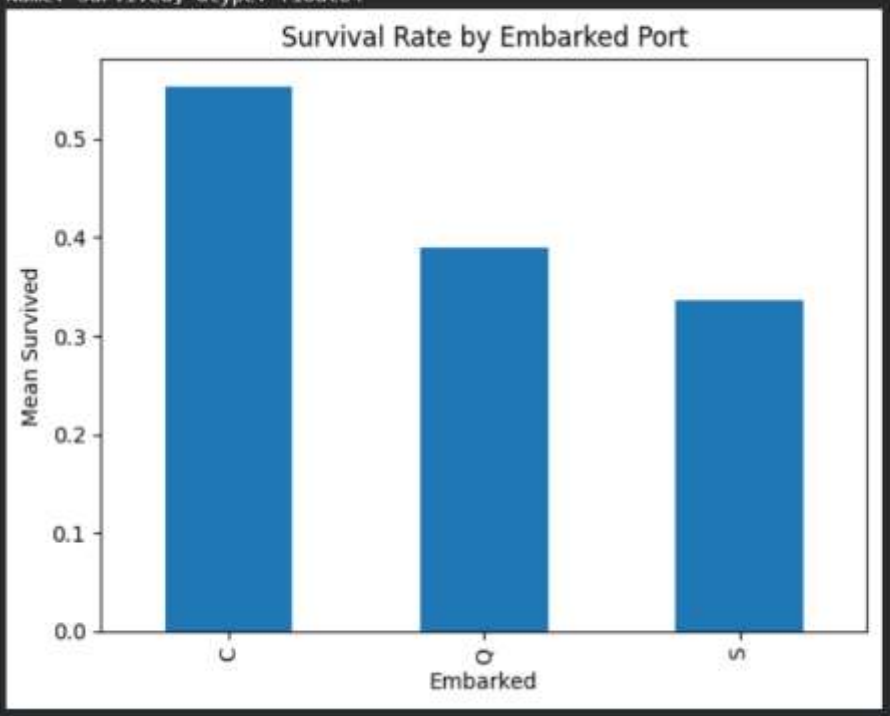


Problem- 5:

Draw a visualization that breaks your visualization from Exercise 3 down by the port of embarkation (’Em

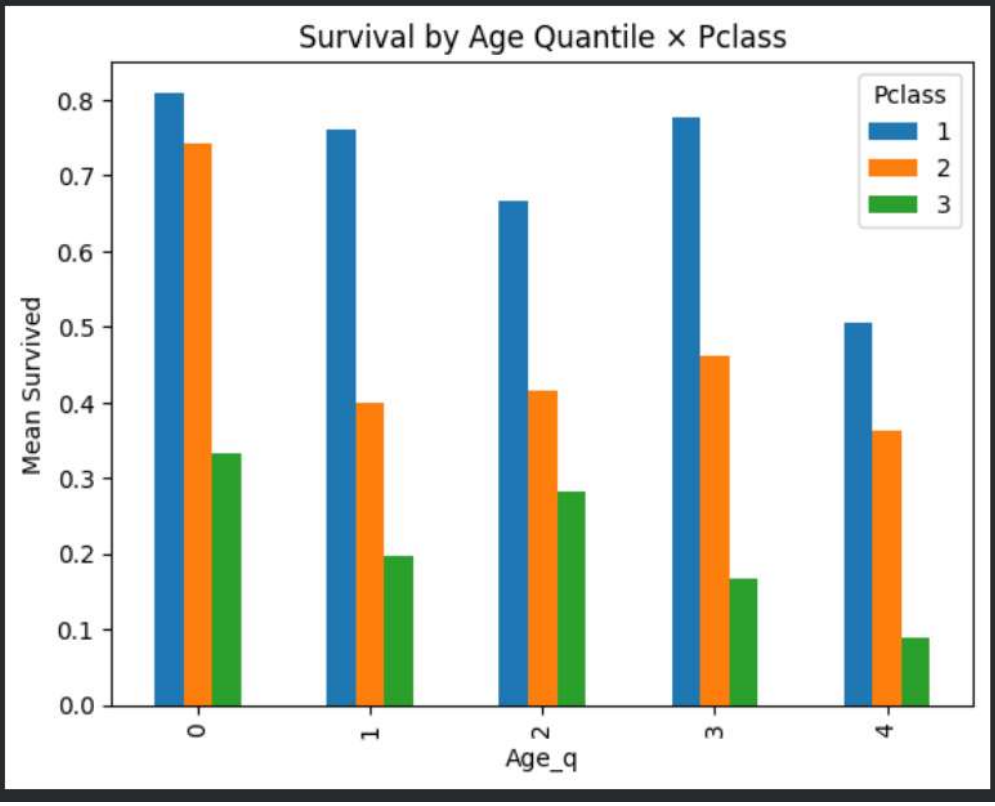
barked’). In this instance, compare the ports ’C’ (Cherbourg), ’Q’ (Queenstown), and ’S’ (Southampton).

```
Survival rate by Embarked:
 Embarked
C    0.553571
Q    0.389610
S    0.336957
Name: Survived, dtype: float64
```



Problem- 6{Optional}:

Show how the survival rates (’Survived’) vary by age group and passenger class (’Pclass’). Break up the ’Age’ column into five quantiles in your DataFrame, and then compare the means of ’Survived’ by class and age group. Draw a visualization using a any plotting library to represent this graphically


Survival by Age Quantile × Pclass
Pclass
1
2
3
Mean Survived
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0
1
2
3
4
Age_q