In [2]:
```
import numpy as np
import pandas as pd
import matplotlib as plt
import seaborn as sns
```

QUESTION 1

In [3]:
```
d1={"POLITICIANS":["CHIEF MINISTER","HOME MINISTER","MINISTER","MLA",np.nan]
   ,"SALARIES":["500000","400000",np.nan,"100000","20000"],"LOCATION":["CAPITAL",np

data1=pd.DataFrame(d1,index=[1,2,3,4,5])
data1
```

Out[3]:

| | POLITICIANS | SALARIES | LOCATION |
|---|---|---|---|
| **1** | CHIEF MINISTER | 500000 | CAPITAL |
| **2** | HOME MINISTER | 400000 | NaN |
| **3** | MINISTER | NaN | CITY |
| **4** | MLA | 100000 | TOWN |
| **5** | NaN | 20000 | VILLAGE |

In [4]:
```
data1["POLITICIANS"].fillna("SARPANCH")
```

Out[4]:
```
1    CHIEF MINISTER
2     HOME MINISTER
3          MINISTER
4               MLA
5          SARPANCH
Name: POLITICIANS, dtype: object
```

In [5]:
```
data1.fillna(method='bfill')
```

Out[5]:

| | POLITICIANS | SALARIES | LOCATION |
|---|---|---|---|
| **1** | CHIEF MINISTER | 500000 | CAPITAL |
| **2** | HOME MINISTER | 400000 | CITY |
| **3** | MINISTER | 100000 | CITY |
| **4** | MLA | 100000 | TOWN |
| **5** | NaN | 20000 | VILLAGE |

In [6]:
```
from sklearn.impute import KNNImputer
knn=KNNImputer(n_neighbors=5)
knn.fit_transform(data1[["SALARIES"]])
```

Out[6]:
```
array([[500000.],
       [400000.],
       [255000.],
       [100000.],
       [ 20000.]])
```

In [ ]:
```
```

QUESTION 2

In [ ]:
```
SAMPLING: sampling is the selection of a subset or a statistical sample (termed sam
            of individuals from within a statistical population to estimate charact
            the whole population.
The sampling is 2 types
they are 1) PROBABILITY SAMPLING
         2) NON- PROBABILITY SAMPLING


1)PROBABILITY SAMPLING : It involves random selection, allowing you to make strong
                         statistical inferences about the whole group.

2)NON- PROBABILITY SAMPLING:It involves non-random selection based on convenience c
                            allowing you to easily collect data.
```

In [ ]:
```
```

QUESTION 6

In [ ]:
```
The CLASSIFICATION and REGRESSION are 2 types Algorithms in the Super-vised learnir

CLASSIFICATION : Classification is the process of finding or discovering a model or
                 separating the data into multiple categorical classes i.e. discret
                 In classification, data is categorized under different labels acc
                 the input and then the labels are predicted for the data.


REGRESSION : Regression is the process of finding a model or function for distingui
             continuous real values instead of using classes or discrete values.
             It can also identify the distribution movement depending on the histor
```

In [ ]:
```
```

QUESTION 9

In [8]:
```
import numpy as np
import pandas as pd
```

In [9]:
```
data = [(chr(letter), ord(chr(letter))) for letter in range(65, 91)]
data65=pd.DataFrame(data,columns=["letter","asciivalues"])
```

In [10]:
```
data65
```

Out[10]:

| | letter | asciivalues |
|---|---|---|
| **0** | A | 65 |
| **1** | B | 66 |
| **2** | C | 67 |
| **3** | D | 68 |
| **4** | E | 69 |
| **5** | F | 70 |
| **6** | G | 71 |
| **7** | H | 72 |
| **8** | I | 73 |
| **9** | J | 74 |
| **10** | K | 75 |
| **11** | L | 76 |
| **12** | M | 77 |
| **13** | N | 78 |
| **14** | O | 79 |
| **15** | P | 80 |
| **16** | Q | 81 |
| **17** | R | 82 |
| **18** | S | 83 |
| **19** | T | 84 |
| **20** | U | 85 |
| **21** | V | 86 |
| **22** | W | 87 |
| **23** | X | 88 |
| **24** | Y | 89 |
| **25** | Z | 90 |

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: