

```
In [ ]: #Name:Khushi Chandrashekhar Satpute
#Aim: To Find Unique and Duplicate Value Count in given dataset
#Roll No:43
#Sec:B

In [1]: import pandas as pd

In [3]: import os

In [5]: os.getcwd()

Out[5]: 'C:\\Users\\vasus'

In [9]: os.chdir("C:\\Users\\vasus\\Downloads")

In [13]: data=pd.read_csv("diabetes.csv")

In [15]: data.head()

Out[15]:
   Pregnancies  Glucose  BloodPressure  SkinThickness  Insulin  BMI  DiabetesPedigreeFunction  Age  Outcome
0            6       148             72             35         0   33.6              0.627    50         1
1            1         85             66             29         0   26.6              0.351    31         0
2            8       183             64              0         0   23.3              0.672    32         1
3            1         89             66             23        94   28.1              0.167    21         0
4            0       137             40             35       168   43.1              2.288    33         1

In [17]: data.tail()

Out[17]:
   Pregnancies  Glucose  BloodPressure  SkinThickness  Insulin  BMI  DiabetesPedigreeFunction  Age  Outcome
763           10       101             76             48       180   32.9              0.171    63         0
764            2       122             70             27         0   36.8              0.340    27         0
765            5       121             72             23       112   26.2              0.245    30         0
766            1       126             60              0         0   30.1              0.349    47         1
767            1        93             70             31         0   30.4              0.315    23         0

In [19]: data.info

Out[19]: <bound method DataFrame.info of
0            6       148             72             35         0   33.6
1            1         85             66             29         0   26.6
2            8       183             64              0         0   23.3
3            1         89             66             23        94   28.1
4            0       137             40             35       168   43.1
..          ...      ...             ...          ...      ...
763          10       101             76             48       180   32.9
764           2       122             70             27         0   36.8
765           5       121             72             23       112   26.2
766           1       126             60              0         0   30.1
767           1        93             70             31         0   30.4

   DiabetesPedigreeFunction  Age  Outcome
0            0.627    50         1
1            0.351    31         0
2            0.672    32         1
3            0.167    21         0
4            2.288    33         1
..          ...      ...      ...
763          0.171    63         0
764          0.340    27         0
765          0.245    30         0
766          0.349    47         1
767          0.315    23         0

[768 rows x 9 columns]>

In [21]: data.describe

Out[21]: <bound method NDFrame.describe of
0            6       148             72             35         0   33.6
1            1         85             66             29         0   26.6
2            8       183             64              0         0   23.3
3            1         89             66             23        94   28.1
4            0       137             40             35       168   43.1
..          ...      ...             ...          ...      ...
763          10       101             76             48       180   32.9
764           2       122             70             27         0   36.8
765           5       121             72             23       112   26.2
766           1       126             60              0         0   30.1
767           1        93             70             31         0   30.4

   DiabetesPedigreeFunction  Age  Outcome
0            0.627    50         1
1            0.351    31         0
2            0.672    32         1
3            0.167    21         0
4            2.288    33         1
..          ...      ...      ...
763          0.171    63         0
764          0.340    27         0
765          0.245    30         0
766          0.349    47         1
767          0.315    23         0

[768 rows x 9 columns]>

In [23]: data.isna()

Out[23]:
   Pregnancies  Glucose  BloodPressure  SkinThickness  Insulin  BMI  DiabetesPedigreeFunction  Age  Outcome
0            0         0              0              0         0    0              0.627    50         1
1            0         0              0              0         0    0              0.351    31         0
2            0         0              0              0         0    0              0.672    32         1
3            0         0              0              0         0    0              0.167    21         0
4            0         0              0              0         0    0              2.288    33         1
..          ...      ...             ...          ...      ...
763          0         0              0              0         0    0              0.171    63         0
764          0         0              0              0         0    0              0.340    27         0
765          0         0              0              0         0    0              0.245    30         0
766          0         0              0              0         0    0              0.349    47         1
767          0         0              0              0         0    0              0.315    23         0

768 rows x 9 columns

In [27]: data.isna().any()

Out[27]:
Pregnancies      False
Glucose           False
BloodPressure     False
SkinThickness     False
Insulin           False
BMI               False
DiabetesPedigreeFunction  False
Age              False
Outcome          False
dtype: bool

In [29]: data.isna().sum()

Out[29]:
Pregnancies      0
Glucose           0
BloodPressure     0
SkinThickness     0
Insulin           0
BMI               0
DiabetesPedigreeFunction  0
Age              0
Outcome          0
dtype: int64

In [43]: data['Age'].unique()

Out[43]: array([50, 31, 32, 21, 33, 30, 26, 29, 53, 54, 34, 57, 59, 51, 27, 41, 43,
        22, 38, 60, 28, 45, 35, 46, 56, 37, 48, 40, 25, 24, 58, 42, 44, 39,
        36, 23, 61, 69, 62, 55, 65, 47, 52, 66, 49, 63, 67, 72, 81, 64, 70,
        68], dtype=int64)

In [39]: data['Glucose','Age'].unique()

-----
KeyError                                Traceback (most recent call last)
File ~\anaconda3\lib\site-packages\pandas\core\indexes\base.py:3805, in Index.get_loc(self, key)
    3804 try:
-> 3805     return self._engine.get_loc(casted_key)
    3806 except KeyError as err:

File index.pyx:167, in pandas._libs.index.IndexEngine.get_loc()

File index.pyx:196, in pandas._libs.index.IndexEngine.get_loc()

File pandas\_libs\hashtable_class_helper.pxi:7081, in pandas._libs.hashtable.PyObjectHashTable.get_item()

File pandas\_libs\hashtable_class_helper.pxi:7089, in pandas._libs.hashtable.PyObjectHashTable.get_item()

KeyError: ('Glucose', 'Age')

The above exception was the direct cause of the following exception:

KeyError                                Traceback (most recent call last)
Cell In[39], line 1
----> 1 data['Glucose','Age'].unique()

File ~\anaconda3\lib\site-packages\pandas\core\frame.py:4102, in DataFrame.__getitem__(self, key)
    4100 if self.columns.nlevels > 1:
    4101     return self._getitem_multilevel(key)
-> 4102 indexer = self.columns.get_loc(key)
    4103 if is_integer(indexer):
    4104     indexer = [indexer]

File ~\anaconda3\lib\site-packages\pandas\core\indexes\base.py:3812, in Index.get_loc(self, key)
    3807 if isinstance(casted_key, slice) or (
    3808     isinstance(casted_key, abc.Iterable)
    3809     and any(isinstance(x, slice) for x in casted_key)
    3810 ):
    3811     raise InvalidIndexError(key)
-> 3812     raise KeyError(key) from err
    3813 except TypeError:
    3814     # If we have a listlike key, _check_indexing_error will raise
    3815     # InvalidIndexError. Otherwise we fall through and re-raise
    3816     # the TypeError.
    3817     self._check_indexing_error(key)

KeyError: ('Glucose', 'Age')

In [45]: data['Age'].duplicated()

Out[45]:
0      False
1      False
2      False
3      False
4      False
...
763     True
764     True
765     True
766     True
767     True
Name: Age, Length: 768, dtype: bool

In [47]: data['Age'].duplicated().sum()

Out[47]: 716

In [53]: data['Age'].count()
```

