adult-income-analysis

October 31, 2024

1. IMPORTING LIBRARIES

[188]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

2. IMPORTING THE DATASET.

[189]:
```
data = r"H:\DA. Python\8. Adult Income Analysis\adult.csv"
newdata = pd.read_csv(data)
```

3. CHECKING DATA STRUCTURE

[190]:
```
newdata.shape
```

[190]:
```
(48842, 15)
```

[191]:
```
newdata
```

[191]:
```
       age     workclass  fnlwgt     education  educational-num  \
0       25       Private  226802          11th                7
1       38       Private   89814       HS-grad                9
2       28     Local-gov  336951    Assoc-acdm               12
3       44       Private  160323  Some-college               10
4       18             ?  103497  Some-college               10
…     …            …     …            …                …
48837   27       Private  257302    Assoc-acdm               12
48838   40       Private  154374       HS-grad                9
48839   58       Private  151910       HS-grad                9
48840   22       Private  201490       HS-grad                9
48841   52  Self-emp-inc  287927       HS-grad                9

           marital-status         occupation relationship   race  gender  \
0           Never-married  Machine-op-inspct    Own-child  Black    Male
1      Married-civ-spouse    Farming-fishing      Husband  White    Male
2      Married-civ-spouse    Protective-serv      Husband  White    Male
3      Married-civ-spouse  Machine-op-inspct      Husband  Black    Male
4           Never-married                  ?    Own-child  White  Female
…                     …                 …          …     …      …
```

```
48837  Married-civ-spouse       Tech-support         Wife  White  Female
48838  Married-civ-spouse  Machine-op-inspct      Husband  White    Male
48839             Widowed       Adm-clerical    Unmarried  White  Female
48840       Never-married       Adm-clerical    Own-child  White    Male
48841  Married-civ-spouse    Exec-managerial         Wife  White  Female

       capital-gain  capital-loss  hours-per-week native-country income
0                 0             0              40  United-States  <=50K
1                 0             0              50  United-States  <=50K
2                 0             0              40  United-States   >50K
3              7688             0              40  United-States   >50K
4                 0             0              30  United-States  <=50K
…               …             …              …            …     …
48837             0             0              38  United-States  <=50K
48838             0             0              40  United-States   >50K
48839             0             0              40  United-States  <=50K
48840             0             0              20  United-States  <=50K
48841         15024             0              40  United-States   >50K

[48842 rows x 15 columns]
```

4. DISPLAY TOP 10 ROWS OF DATA

[192]:
```
newdata.head(10)
```

[192]:
```
   age         workclass  fnlwgt     education  educational-num  \
0   25           Private  226802          11th                7
1   38           Private   89814       HS-grad                9
2   28         Local-gov  336951    Assoc-acdm               12
3   44           Private  160323  Some-college               10
4   18                 ?  103497  Some-college               10
5   34           Private  198693          10th                6
6   29                 ?  227026       HS-grad                9
7   63  Self-emp-not-inc  104626   Prof-school               15
8   24           Private  369667  Some-college               10
9   55           Private  104996       7th-8th                4

       marital-status         occupation   relationship   race  gender  \
0       Never-married  Machine-op-inspct      Own-child  Black    Male
1  Married-civ-spouse    Farming-fishing        Husband  White    Male
2  Married-civ-spouse    Protective-serv        Husband  White    Male
3  Married-civ-spouse  Machine-op-inspct        Husband  Black    Male
4       Never-married                  ?      Own-child  White  Female
5       Never-married      Other-service  Not-in-family  White    Male
6       Never-married                  ?      Unmarried  Black    Male
7  Married-civ-spouse     Prof-specialty        Husband  White    Male
8       Never-married      Other-service      Unmarried  White  Female
```

```
9  Married-civ-spouse       Craft-repair        Husband  White    Male

   capital-gain  capital-loss  hours-per-week native-country income
0             0             0              40  United-States  <=50K
1             0             0              50  United-States  <=50K
2             0             0              40  United-States   >50K
3          7688             0              40  United-States   >50K
4             0             0              30  United-States  <=50K
5             0             0              30  United-States  <=50K
6             0             0              40  United-States  <=50K
7          3103             0              32  United-States   >50K
8             0             0              40  United-States  <=50K
9             0             0              10  United-States  <=50K
```

5. DISPLAY LAST 10 ROWS OF DATA

[193]:
```
newdata.tail(10)
```

[193]:
```
       age     workclass  fnlwgt     education  educational-num  \
48832   32       Private   34066          10th                6
48833   43       Private   84661     Assoc-voc               11
48834   32       Private  116138       Masters               14
48835   53       Private  321865       Masters               14
48836   22       Private  310152  Some-college               10
48837   27       Private  257302    Assoc-acdm               12
48838   40       Private  154374       HS-grad                9
48839   58       Private  151910       HS-grad                9
48840   22       Private  201490       HS-grad                9
48841   52  Self-emp-inc  287927       HS-grad                9

           marital-status         occupation   relationship  \
48832  Married-civ-spouse  Handlers-cleaners        Husband
48833  Married-civ-spouse              Sales        Husband
48834       Never-married       Tech-support  Not-in-family
48835  Married-civ-spouse    Exec-managerial        Husband
48836       Never-married    Protective-serv  Not-in-family
48837  Married-civ-spouse       Tech-support           Wife
48838  Married-civ-spouse  Machine-op-inspct        Husband
48839             Widowed       Adm-clerical      Unmarried
48840       Never-married       Adm-clerical      Own-child
48841  Married-civ-spouse    Exec-managerial           Wife

                     race  gender  capital-gain  capital-loss  hours-per-week  \
48832  Amer-Indian-Eskimo    Male             0             0              40
48833               White    Male             0             0              45
48834  Asian-Pac-Islander    Male             0             0              11
48835               White    Male             0             0              40
```

```
48836              White    Male          0        0        40
48837              White  Female          0        0        38
48838              White    Male          0        0        40
48839              White  Female          0        0        40
48840              White    Male          0        0        20
48841              White  Female      15024        0        40

      native-country income
48832  United-States  <=50K
48833  United-States  <=50K
48834         Taiwan  <=50K
48835  United-States   >50K
48836  United-States  <=50K
48837  United-States  <=50K
48838  United-States   >50K
48839  United-States  <=50K
48840  United-States  <=50K
48841  United-States   >50K
```

6. GETTING INFORMATION ABOUT OUR DATASET LIKE TOTAL NUMBER ROWS, TOTAL NUMBER OF COLUMNS, DATATYPES OF EACH COLUMN AND MEMORY REQUIREMENT.

[194]:
```python
newdata.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48842 entries, 0 to 48841
Data columns (total 15 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   age              48842 non-null  int64
 1   workclass        48842 non-null  object
 2   fnlwgt           48842 non-null  int64
 3   education        48842 non-null  object
 4   educational-num  48842 non-null  int64
 5   marital-status   48842 non-null  object
 6   occupation       48842 non-null  object
 7   relationship     48842 non-null  object
 8   race             48842 non-null  object
 9   gender           48842 non-null  object
 10  capital-gain     48842 non-null  int64
 11  capital-loss     48842 non-null  int64
 12  hours-per-week   48842 non-null  int64
 13  native-country   48842 non-null  object
 14  income           48842 non-null  object
dtypes: int64(6), object(9)
memory usage: 5.6+ MB
```

7.FETCH RANDOM SAMPLE FROM THE DATASET (50%)

[254]:
```
# For this we have to use sample method of Pandas

newdata.sample(frac = 0.50)
```

[254]:
```
       age         workclass  fnlwgt     education      marital-status  \
27962   48           Private   33669  Some-college  Married-civ-spouse
40600   44           Private  186916       Masters  Married-civ-spouse
7043    49           Private  116927     Bachelors  Married-civ-spouse
1129    28         Local-gov  134771     Bachelors       Never-married
47088   62  Self-emp-not-inc  224520       HS-grad  Married-civ-spouse
…      …               …       …           …                 …
7905    29           Private   97189     Assoc-voc       Never-married
14722   38       Federal-gov  104236    Assoc-acdm            Divorced
44047   30           Private  193298       HS-grad  Married-civ-spouse
20315   21           Private   61777  Some-college       Never-married
13742   32           Private  234976  Some-college  Married-civ-spouse

              occupation   relationship   race  gender  hours-per-week  \
27962  Transport-moving        Husband  White    Male              60
40600   Exec-managerial        Husband  White    Male              50
7043              Sales        Husband  White    Male              60
1129     Prof-specialty      Own-child  White  Female              55
47088             Sales        Husband  White    Male              90
…                    …            …     …     …               …
7905       Adm-clerical      Own-child  White  Female              40
14722      Adm-clerical      Unmarried  White  Female              40
44047  Transport-moving        Husband  White    Male              45
20315      Craft-repair  Not-in-family  White    Male              70
13742   Exec-managerial           Wife  White  Female              55

      native-country  income  enconded_salary
27962  United-States       0                1
40600  United-States       1                1
7043   United-States       0                1
1129   United-States       0                1
47088  United-States       1                1
…               …     …              …
7905   United-States       0                1
14722  United-States       0                1
44047  United-States       0                1
20315  United-States       0                1
13742  United-States       0                1

[22588 rows x 13 columns]
```

— Here we are getting 50% sample from original dataset

```
[196]: newdata.sample(frac=0.50,random_state=100) #Using random_state will generate␣
 ↪same sequence of the dataset.
```

```
[196]:        age  workclass  fnlwgt     education  educational-num  \
12393   37    Private  110331   Prof-school               15
48701   23    Private   45834     Bachelors               13
17918   28    Private   89718       HS-grad                9
11352   30    Private  351770           9th                5
36198   31    Private  164190          10th                6
…     …        …     …             …             …
48573   41    Private  318046  Some-college               10
47252   41  Local-gov   33658  Some-college               10
33142   69    Private  312653  Some-college               10
2965    21          ?  334593  Some-college               10
32089   34    Private  186269       HS-grad                9

           marital-status         occupation   relationship   race  gender  \
12393  Married-civ-spouse      Other-service           Wife  White  Female
48701       Never-married    Exec-managerial  Not-in-family  White  Female
17918       Never-married              Sales  Not-in-family  White  Female
11352            Divorced      Other-service      Unmarried  White  Female
36198  Married-civ-spouse  Transport-moving         Husband  White    Male
…                     …                …            …    …     …
48573  Married-civ-spouse  Transport-moving         Husband  White    Male
47252  Married-civ-spouse   Protective-serv         Husband  White    Male
33142  Married-civ-spouse              Sales        Husband  White    Male
2965        Never-married                  ?  Not-in-family  White    Male
32089            Divorced       Adm-clerical      Own-child  White    Male

       capital-gain  capital-loss  hours-per-week native-country income
12393             0             0              60  United-States   >50K
48701             0             0              50  United-States  <=50K
17918          2202             0              48  United-States  <=50K
11352             0             0              38  United-States  <=50K
36198             0             0              40  United-States  <=50K
…               …             …             …            …     …
48573             0             0              48  United-States   >50K
47252             0             0              45  United-States   >50K
33142             0             0              25  United-States  <=50K
2965              0             0              40  United-States  <=50K
32089             0             0              40  United-States  <=50K

[24421 rows x 15 columns]
```

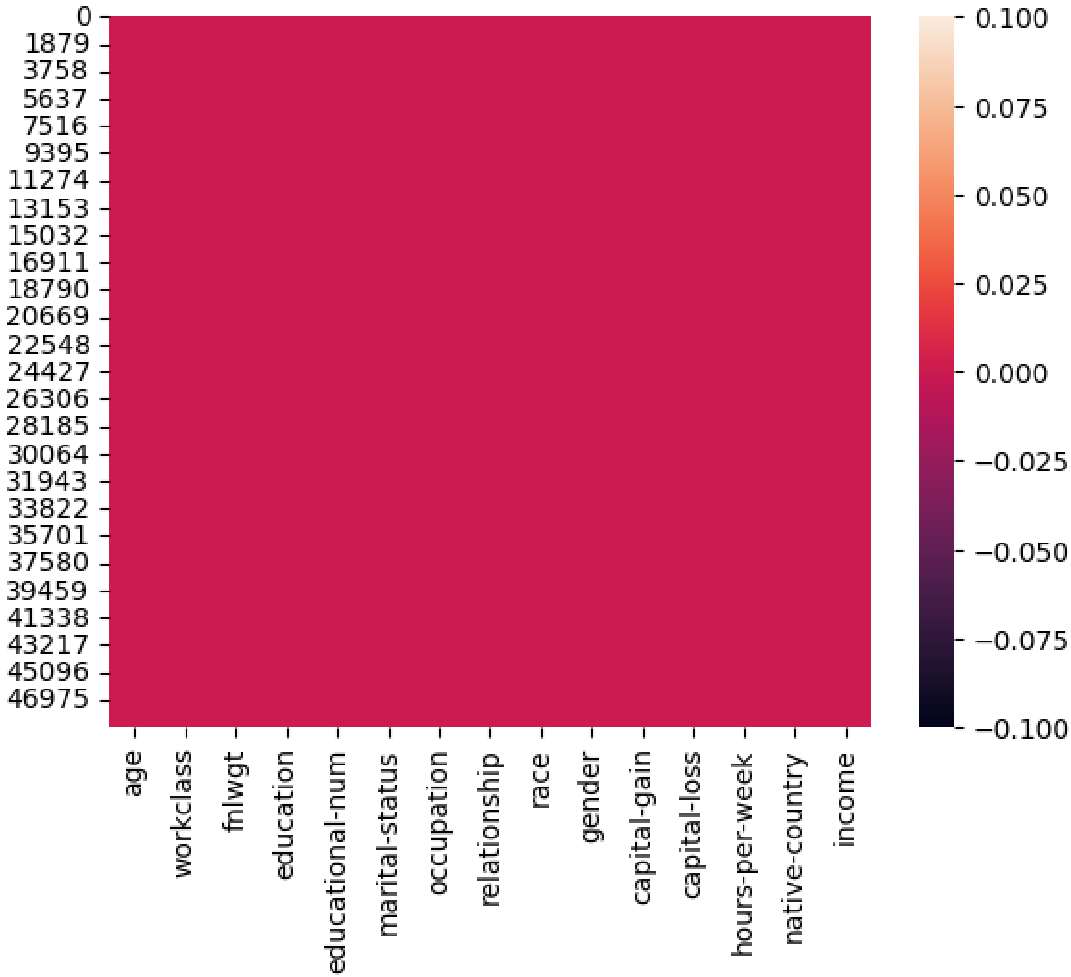
8.CHECK NULL VALUES IN THE DATASET

```
[197]: newdata.isnull().sum(axis=0)
```

[197]:
```
age                0
workclass          0
fnlwgt             0
education          0
educational-num    0
marital-status     0
occupation         0
relationship       0
race               0
gender             0
capital-gain       0
capital-loss       0
hours-per-week     0
native-country     0
income             0
dtype: int64
```

[198]:
```
sns.heatmap(newdata.isnull())
```

[198]:
```
<Axes: >
```



9. PERFORM DATA CLEANING [REPLACE '?' WITH NAN]

[199]:
```
newdata.tail(20)
```

[199]:
```
       age         workclass  fnlwgt     education  educational-num  \
48822   41                 ?  202822       HS-grad                9
48823   72                 ?  129912       HS-grad                9
48824   45         Local-gov  119199    Assoc-acdm               12
48825   31           Private  199655       Masters               14
48826   39         Local-gov  111499    Assoc-acdm               12
48827   37           Private  198216    Assoc-acdm               12
48828   43           Private  260761       HS-grad                9
48829   65  Self-emp-not-inc   99359   Prof-school               15
48830   43         State-gov  255835  Some-college               10
48831   43  Self-emp-not-inc   27242  Some-college               10
48832   32           Private   34066          10th                6
```

48833	43	Private	84661	Assoc-voc	11
48834	32	Private	116138	Masters	14
48835	53	Private	321865	Masters	14
48836	22	Private	310152	Some-college	10
48837	27	Private	257302	Assoc-acdm	12
48838	40	Private	154374	HS-grad	9
48839	58	Private	151910	HS-grad	9
48840	22	Private	201490	HS-grad	9
48841	52	Self-emp-inc	287927	HS-grad	9

	marital-status	occupation	relationship	\
48822	Separated	?	Not-in-family	
48823	Married-civ-spouse	?	Husband	
48824	Divorced	Prof-specialty	Unmarried	
48825	Divorced	Other-service	Not-in-family	
48826	Married-civ-spouse	Adm-clerical	Wife	
48827	Divorced	Tech-support	Not-in-family	
48828	Married-civ-spouse	Machine-op-inspct	Husband	
48829	Never-married	Prof-specialty	Not-in-family	
48830	Divorced	Adm-clerical	Other-relative	
48831	Married-civ-spouse	Craft-repair	Husband	
48832	Married-civ-spouse	Handlers-cleaners	Husband	
48833	Married-civ-spouse	Sales	Husband	
48834	Never-married	Tech-support	Not-in-family	
48835	Married-civ-spouse	Exec-managerial	Husband	
48836	Never-married	Protective-serv	Not-in-family	
48837	Married-civ-spouse	Tech-support	Wife	
48838	Married-civ-spouse	Machine-op-inspct	Husband	
48839	Widowed	Adm-clerical	Unmarried	
48840	Never-married	Adm-clerical	Own-child	
48841	Married-civ-spouse	Exec-managerial	Wife	

	race	gender	capital-gain	capital-loss	hours-per-week	\
48822	Black	Female	0	0	32	
48823	White	Male	0	0	25	
48824	White	Female	0	0	48	
48825	Other	Female	0	0	30	
48826	White	Female	0	0	20	
48827	White	Female	0	0	40	
48828	White	Male	0	0	40	
48829	White	Male	1086	0	60	
48830	White	Female	0	0	40	
48831	White	Male	0	0	50	
48832	Amer-Indian-Eskimo	Male	0	0	40	
48833	White	Male	0	0	45	
48834	Asian-Pac-Islander	Male	0	0	11	
48835	White	Male	0	0	40	

```
48836                White     Male          0          0          40
48837                White   Female          0          0          38
48838                White     Male          0          0          40
48839                White   Female          0          0          40
48840                White     Male          0          0          20
48841                White   Female      15024          0          40

      native-country income
48822  United-States  <=50K
48823  United-States  <=50K
48824  United-States  <=50K
48825  United-States  <=50K
48826  United-States   >50K
48827  United-States  <=50K
48828         Mexico  <=50K
48829  United-States  <=50K
48830  United-States  <=50K
48831  United-States  <=50K
48832  United-States  <=50K
48833  United-States  <=50K
48834         Taiwan  <=50K
48835  United-States   >50K
48836  United-States  <=50K
48837  United-States  <=50K
48838  United-States   >50K
48839  United-States  <=50K
48840  United-States  <=50K
48841  United-States   >50K
```

[200]:
```python
# to find how many columns we have "?"

newdata.isin(["?"]).sum()
```

[200]:
```
age                   0
workclass          2799
fnlwgt                0
education             0
educational-num       0
marital-status        0
occupation         2809
relationship          0
race                  0
gender                0
capital-gain          0
capital-loss          0
hours-per-week        0
native-country      857
```

```
income                  0
dtype: int64
```

[201]:
```
# first we have to replace "?" with "NaN"
# We can drop it with dropna method
```

[202]:
```
newdata.columns
```

[202]:
```
Index(['age', 'workclass', 'fnlwgt', 'education', 'educational-num',
       'marital-status', 'occupation', 'relationship', 'race', 'gender',
       'capital-gain', 'capital-loss', 'hours-per-week', 'native-country',
       'income'],
      dtype='object')
```

[203]:
```
newdata['workclass'] = newdata['workclass'].replace("?", np.nan)
newdata['occupation'] = newdata['occupation'].replace("?", np.nan)
newdata['native-country'] = newdata['native-country'].replace("?", np.nan)
```

[204]:
```
newdata.isin(["?"]).sum()
```

[204]:
```
age                0
workclass          0
fnlwgt             0
education          0
educational-num    0
marital-status     0
occupation         0
relationship       0
race               0
gender             0
capital-gain       0
capital-loss       0
hours-per-week     0
native-country     0
income             0
dtype: int64
```

[205]:
```
newdata.isnull().sum()
```

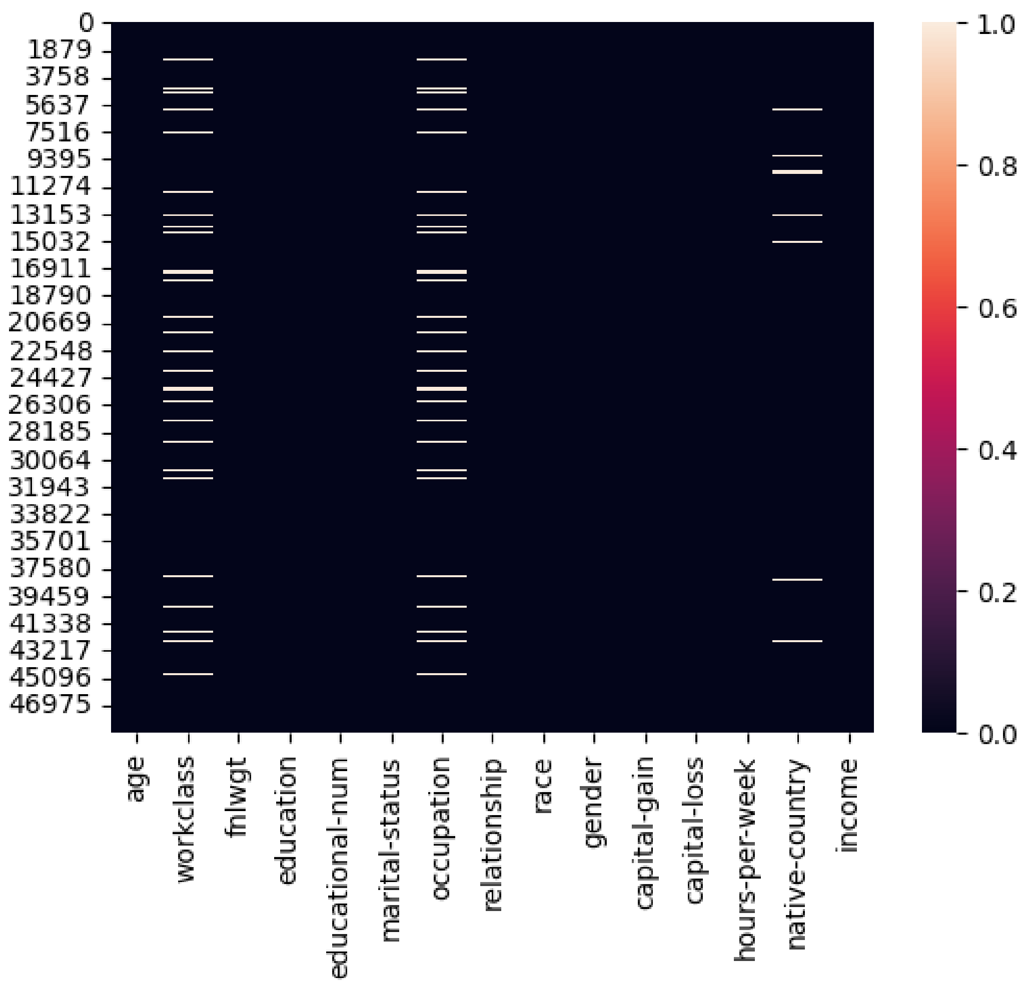
[205]:
```
age                   0
workclass          2799
fnlwgt                0
education             0
educational-num       0
marital-status        0
occupation         2809
relationship          0
```

```
race                 0
gender               0
capital-gain         0
capital-loss         0
hours-per-week       0
native-country     857
income               0
dtype: int64
```

[206]:
```
#LETS VISUALISE NULL VALUES WITH HEATMAP

sns.heatmap(newdata.isnull())
```

[206]: <Axes: >



10. DROP ALL THE MISSING VALUES

[207]:
```
#Lets see all the missing values in percentage

per_value = newdata.isnull().sum()*100 / len(newdata)
print(per_value)
```

```
age                0.000000
workclass          5.730724
fnlwgt             0.000000
education          0.000000
educational-num    0.000000
marital-status     0.000000
occupation         5.751198
relationship       0.000000
race               0.000000
gender             0.000000
capital-gain       0.000000
capital-loss       0.000000
hours-per-week     0.000000
native-country     1.754637
income             0.000000
dtype: float64
```

—5% of value is missing in workclass, occupation and 1% in native-country—

[208]:
```
newdata.dropna(how = 'any', inplace = True)  #use "how" parameter to "=any",␣
↪it'll drop rows with any missing values
```

[209]:
```
newdata.shape
```

[209]:
```
(45222, 15)
```

11. CHECK FOR DUPLICATE DATA AND DROP THEM

[210]:
```
dup = newdata.duplicated().any()
print("Is there any duplicated values?:", dup )
```

```
Is there any duplicated values?: True
```

[211]:
```
newdata = newdata.drop_duplicates()
```

[212]:
```
newdata.shape
```

[212]:
```
(45175, 15)
```

12. GET OVERALL STATISTICS ABOUT THE DATAFRAME

[213]:
```
newdata.describe(include='all')
```

[213]:
```
                 age workclass        fnlwgt education  educational-num  \
count   45175.000000     45175  4.517500e+04     45175     45175.000000
unique           NaN         7           NaN        16              NaN
top              NaN   Private           NaN   HS-grad              NaN
freq             NaN     33262           NaN     14770              NaN
mean       38.556170       NaN  1.897388e+05       NaN        10.119314
std        13.215349       NaN  1.056524e+05       NaN         2.551740
min        17.000000       NaN  1.349200e+04       NaN         1.000000
25%        28.000000       NaN  1.173925e+05       NaN         9.000000
50%        37.000000       NaN  1.783120e+05       NaN        10.000000
75%        47.000000       NaN  2.379030e+05       NaN        13.000000
max        90.000000       NaN  1.490400e+06       NaN        16.000000

            marital-status    occupation relationship   race gender  \
count                45175         45175        45175  45175  45175
unique                   7            14            6      5      2
top     Married-civ-spouse  Craft-repair      Husband  White   Male
freq                 21042          6010        18653  38859  30495
mean                   NaN           NaN          NaN    NaN    NaN
std                    NaN           NaN          NaN    NaN    NaN
min                    NaN           NaN          NaN    NaN    NaN
25%                    NaN           NaN          NaN    NaN    NaN
50%                    NaN           NaN          NaN    NaN    NaN
75%                    NaN           NaN          NaN    NaN    NaN
max                    NaN           NaN          NaN    NaN    NaN

        capital-gain  capital-loss  hours-per-week native-country income
count   45175.000000  45175.000000    45175.000000          45175  45175
unique           NaN           NaN             NaN             41      2
top              NaN           NaN             NaN  United-States  <=50K
freq             NaN           NaN             NaN          41256  33973
mean     1102.576270     88.687593       40.942512            NaN    NaN
std      7510.249876    405.156611       12.007730            NaN    NaN
min         0.000000      0.000000        1.000000            NaN    NaN
25%         0.000000      0.000000       40.000000            NaN    NaN
50%         0.000000      0.000000       40.000000            NaN    NaN
75%         0.000000      0.000000       45.000000            NaN    NaN
max     99999.000000   4356.000000       99.000000            NaN    NaN
```

[214]:
```
newdata.columns
```

[214]:
```
Index(['age', 'workclass', 'fnlwgt', 'education', 'educational-num',
       'marital-status', 'occupation', 'relationship', 'race', 'gender',
       'capital-gain', 'capital-loss', 'hours-per-week', 'native-country',
       'income'],
      dtype='object')
```

[215]:
```
newdata['education'].unique()
```

[215]:
```
array(['11th', 'HS-grad', 'Assoc-acdm', 'Some-college', '10th',
       'Prof-school', '7th-8th', 'Bachelors', 'Masters', '5th-6th',
       'Assoc-voc', '9th', 'Doctorate', '12th', '1st-4th', 'Preschool'],
      dtype=object)
```

[216]:
```
#education column contains str values

#lets check educational-num

newdata['educational-num'].unique()
```

[216]:
```
array([ 7,  9, 12, 10,  6, 15,  4, 13, 14,  3, 11,  5, 16,  8,  2,  1])
```

[217]:
```
# educational number contains int values

# both eduation and educational-num contains similar value

# dropping any one column
```

13. DROP THE COLUMNS EDUCATION-NUM, CAPITAL-GAIN, AND CAPITAL-LOSS

[218]:
```
newdata.columns
```

[218]:
```
Index(['age', 'workclass', 'fnlwgt', 'education', 'educational-num',
       'marital-status', 'occupation', 'relationship', 'race', 'gender',
       'capital-gain', 'capital-loss', 'hours-per-week', 'native-country',
       'income'],
      dtype='object')
```

[219]:
```
newdata = newdata.drop(['educational-num', 'capital-gain', 'capital-loss'],␣
 ↪axis = 1)
```

[220]:
```
newdata.columns
```

[220]:
```
Index(['age', 'workclass', 'fnlwgt', 'education', 'marital-status',
       'occupation', 'relationship', 'race', 'gender', 'hours-per-week',
       'native-country', 'income'],
      dtype='object')
```

UNIVARIATE ANALYSIS

14. WHAT IS THE DISTRIBUTION OF AGE COLUMN?

[221]:
```
newdata.columns
```

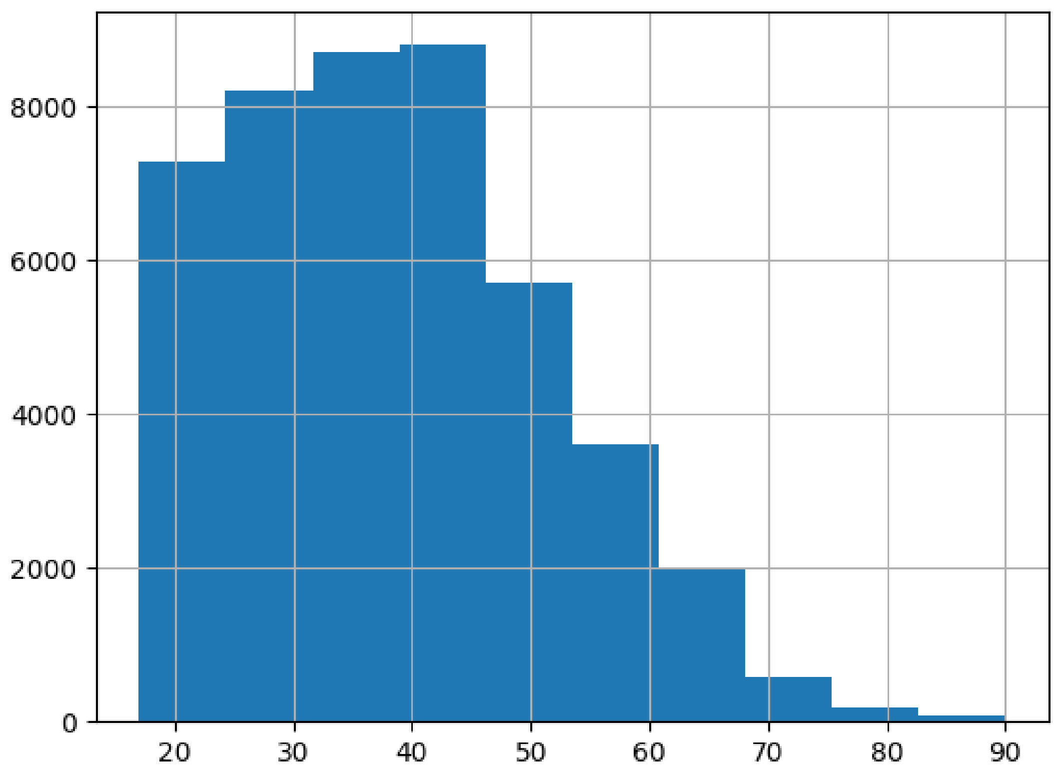
[221]: Index(['age', 'workclass', 'fnlwgt', 'education', 'marital-status',
       'occupation', 'relationship', 'race', 'gender', 'hours-per-week',
       'native-country', 'income'],
      dtype='object')

[222]:
```
newdata['age'].describe()
```

[222]:
```
count    45175.000000
mean        38.556170
std         13.215349
min         17.000000
25%         28.000000
50%         37.000000
75%         47.000000
max         90.000000
Name: age, dtype: float64
```

[223]:
```
newdata['age'].hist()
```

[223]: <Axes: >



CONCLUSION = As we can see most of the age values are from 17 to 48.

15. FIND TOTAL NUMBER OF PERSONS HAVING AGE BETWEEN 17 TO 48 (INCLUSIVE) USING BETWEEN METHOD.

[224]:
```
sum((newdata['age']>=17) & (newdata['age']<=48))
```

[224]:
```
34858
```

[225]:
```
# we can find this using "Between Method" . if we have two or more arguments,␣
↪put inside paranthesis
# use sum function, to find true values.

sum(newdata['age'].between(17,48))
```

[225]:
```
34858
```

16. WHAT IS THE DISTRIBUTION OF WORKCLASS COLUMN?

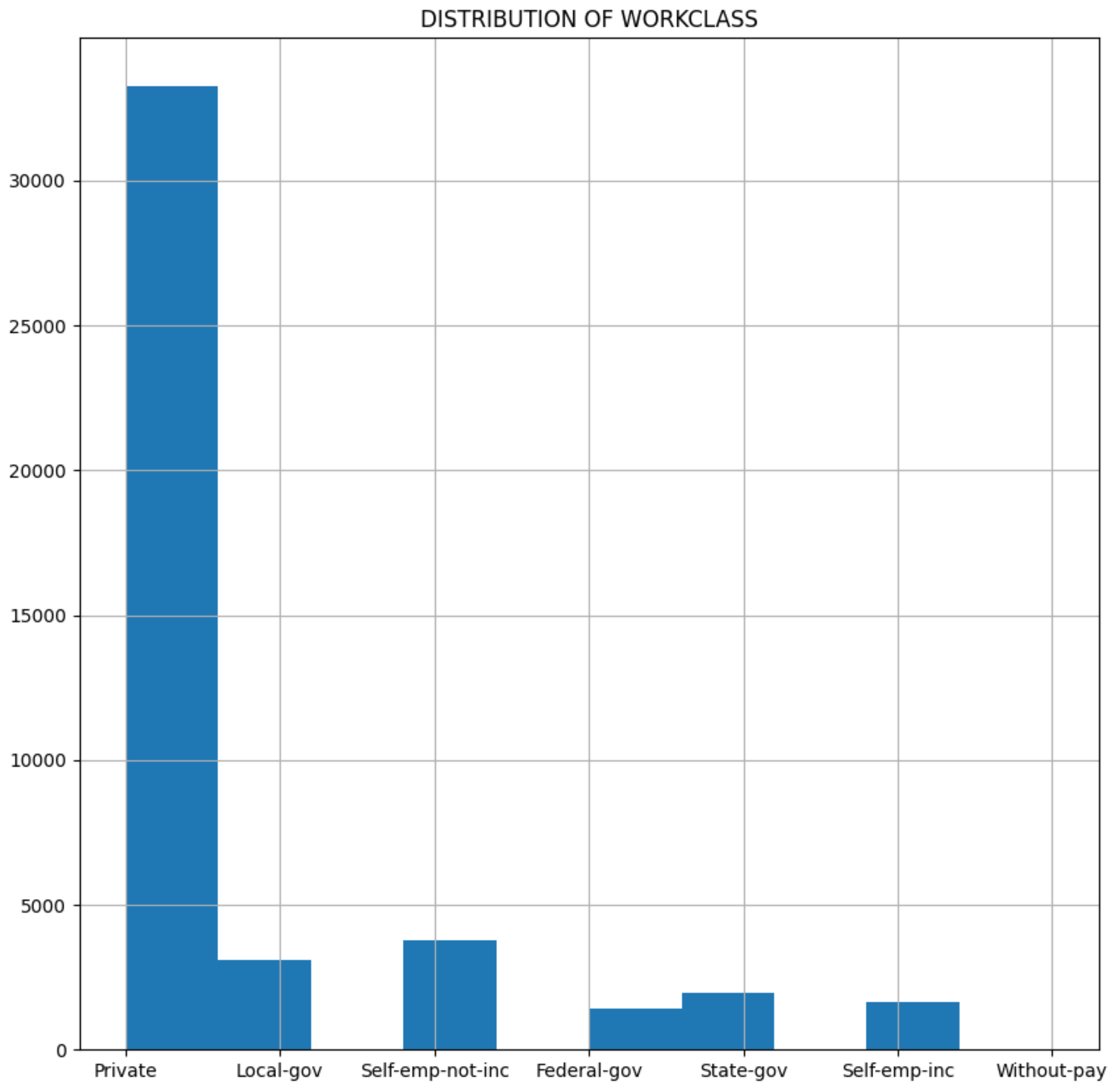
[226]:
```
newdata.columns
```

[226]:
```
Index(['age', 'workclass', 'fnlwgt', 'education', 'marital-status',
       'occupation', 'relationship', 'race', 'gender', 'hours-per-week',
       'native-country', 'income'],
      dtype='object')
```

[227]:
```
newdata['workclass'].describe()
```

[227]:
```
count       45175
unique          7
top       Private
freq        33262
Name: workclass, dtype: object
```

[228]:
```
plt.figure(figsize=(10,10))
newdata['workclass'].hist()
plt.title("DISTRIBUTION OF WORKCLASS")
plt.show()
```



CONCLUSION = This shows that most of them work in private sector job.

17. HOW MANY PERSONS HAVING BACHELORS OR MASTERS DEGREE?

[229]:
```
newdata.columns
```

[229]:
```
Index(['age', 'workclass', 'fnlwgt', 'education', 'marital-status',
       'occupation', 'relationship', 'race', 'gender', 'hours-per-week',
       'native-country', 'income'],
      dtype='object')
```

[230]:
```
filter1 = newdata['education'] == 'Masters'
filter2 = newdata['education'] == 'Bachelors'
```

[231]:
```python
print(filter1.sum())
print(filter2.sum())
```

```
2513
7559
```

[232]:
```python
2513+7559
```

[232]:
```
10072
```

[233]:
```python
len(newdata[filter1 + filter2])
```

[233]:
```
10072
```

[234]:
```python
len(newdata[filter1 | filter2])
```

[234]:
```
10072
```

BIVARIATE ANALYSIS

18. REPLACE INCOME VALUES WITH 0 AND 1

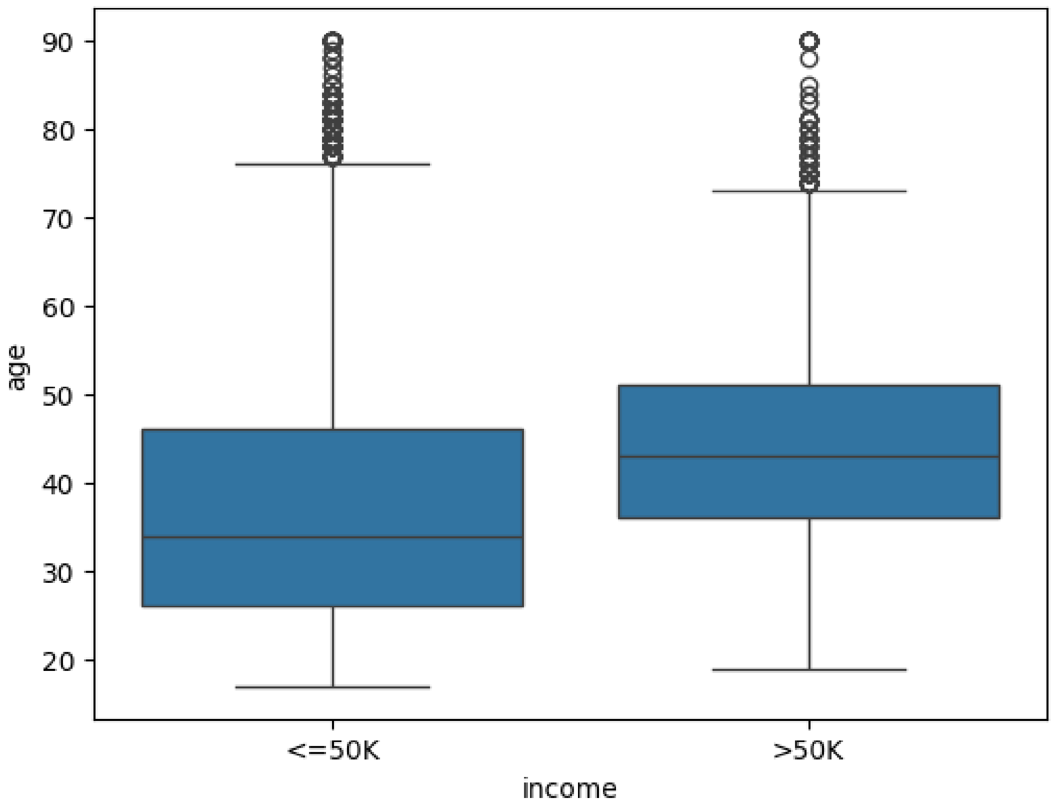
[235]:
```python
newdata.columns
```

[235]:
```
Index(['age', 'workclass', 'fnlwgt', 'education', 'marital-status',
       'occupation', 'relationship', 'race', 'gender', 'hours-per-week',
       'native-country', 'income'],
      dtype='object')
```

[236]:
```python
# Bivariate Analysis is used to find relationship between two variables.
# something as simple as creating scatterplot or boxplot.
```

[237]:
```python
sns.boxplot(x = 'income', y = 'age', data = newdata)
```

[237]:
```
<Axes: xlabel='income', ylabel='age'>
```



19. REPLACE INCOME VALUES [‘<=50K’, ‘>50K’] WITH 0 AND 1

[238]:
```
newdata.columns
```

[238]:
```
Index(['age', 'workclass', 'fnlwgt', 'education', 'marital-status',
       'occupation', 'relationship', 'race', 'gender', 'hours-per-week',
       'native-country', 'income'],
      dtype='object')
```

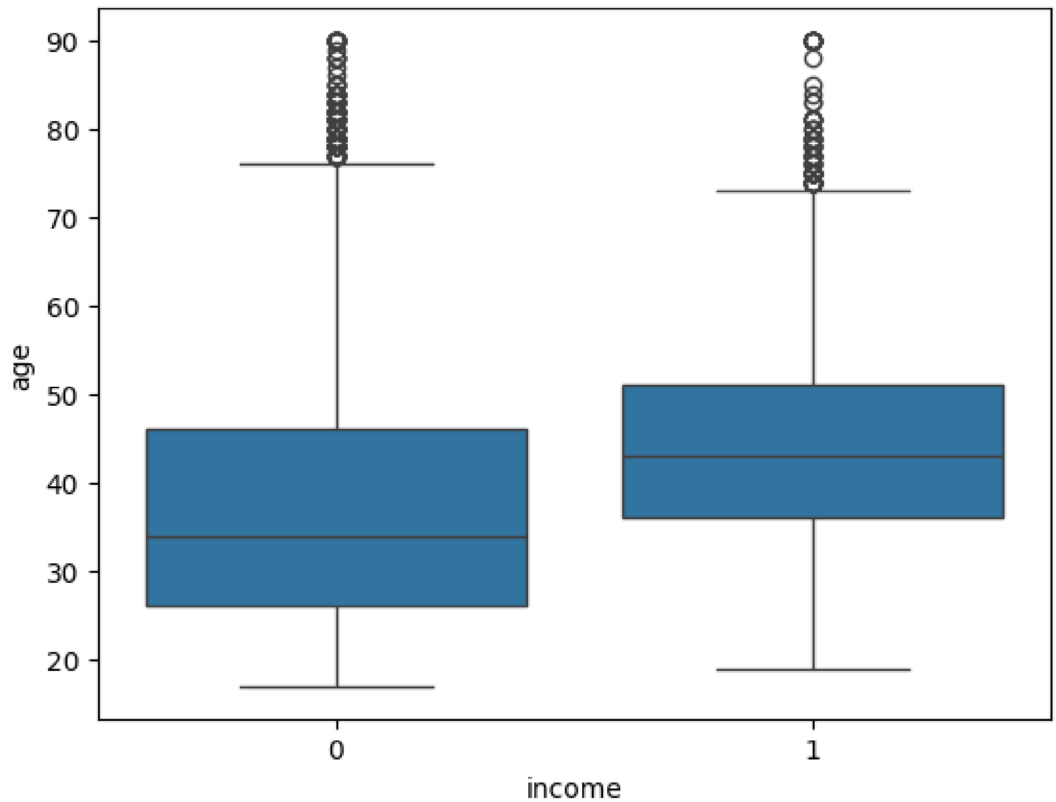
[239]:
```
newdata['income'].unique()
```

[239]:
```
array(['<=50K', '>50K'], dtype=object)
```

[240]:
```
newdata['income'] = newdata['income'].map({'<=50K':0 , '>50K' :1 })
```

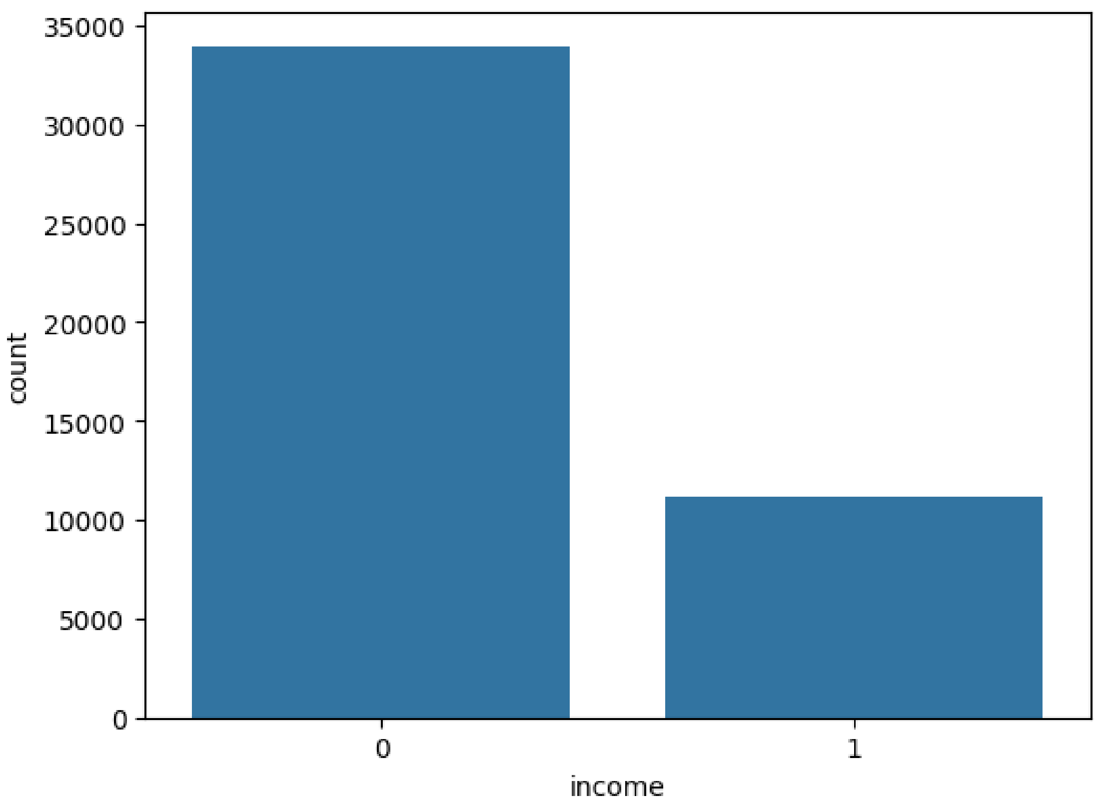
[241]:
```
sns.boxplot(x = 'income', y = 'age', data = newdata)
```

[241]:
```
<Axes: xlabel='income', ylabel='age'>
```



[242]:
```
sns.countplot(x='income', data=newdata)
```

[242]: <Axes: xlabel='income', ylabel='count'>



20. WHICH WORKCLASS GETTING THE HIGHEST SALARY?

[243]:
```
newdata.columns
```

[243]:
```
Index(['age', 'workclass', 'fnlwgt', 'education', 'marital-status',
       'occupation', 'relationship', 'race', 'gender', 'hours-per-week',
       'native-country', 'income'],
      dtype='object')
```

[244]:
```
newdata.groupby('workclass')['income'].mean().sort_values(ascending=False)
```

[244]:
```
workclass
Self-emp-inc        0.554407
Federal-gov         0.390469
Local-gov           0.295161
Self-emp-not-inc    0.279051
State-gov           0.267215
Private             0.217816
Without-pay         0.095238
Name: income, dtype: float64
```

21. WHO HAS BETTER CHANCE TO GET SALARY GREATER THAN 50K MALE OR

FEMALE?

[245]:
```
newdata.columns
```

[245]:
```
Index(['age', 'workclass', 'fnlwgt', 'education', 'marital-status',
       'occupation', 'relationship', 'race', 'gender', 'hours-per-week',
       'native-country', 'income'],
      dtype='object')
```

[246]:
```
def income_data(inc):
    if inc == '<=50k':
        return 0
    else:
        return 1
```

[247]:
```
newdata['enconded_salary'] = newdata['income'].apply(income_data)
```

[248]:
```
newdata.groupby('gender')['enconded_salary'].mean().sort_values(ascending=False)
```

[248]:
```
gender
Female    1.0
Male      1.0
Name: enconded_salary, dtype: float64
```

22. CONVERT WORKCLASS COLUMNS DATATYPE TO CATEGORY DATATYPE

[251]:
```
newdata.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 45175 entries, 0 to 48841
Data columns (total 13 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   age              45175 non-null  int64
 1   workclass        45175 non-null  object
 2   fnlwgt           45175 non-null  int64
 3   education        45175 non-null  object
 4   marital-status   45175 non-null  object
 5   occupation       45175 non-null  object
 6   relationship     45175 non-null  object
 7   race             45175 non-null  object
 8   gender           45175 non-null  object
 9   hours-per-week   45175 non-null  int64
 10  native-country   45175 non-null  object
 11  income           45175 non-null  int64
 12  enconded_salary  45175 non-null  int64
dtypes: int64(5), object(8)
memory usage: 4.8+ MB
```

[252]:
```python
newdata['workclass'] = newdata['workclass'].astype('category')
```

[253]:
```python
newdata.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 45175 entries, 0 to 48841
Data columns (total 13 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   age              45175 non-null  int64
 1   workclass        45175 non-null  category
 2   fnlwgt           45175 non-null  int64
 3   education        45175 non-null  object
 4   marital-status   45175 non-null  object
 5   occupation       45175 non-null  object
 6   relationship     45175 non-null  object
 7   race             45175 non-null  object
 8   gender           45175 non-null  object
 9   hours-per-week   45175 non-null  int64
 10  native-country   45175 non-null  object
 11  income           45175 non-null  int64
 12  enconded_salary  45175 non-null  int64
dtypes: category(1), int64(5), object(7)
memory usage: 4.5+ MB
```