---------- Workshop #2 ----------

- This workshop includes marked tasks that comprise 16% of your final mark in this module.
- To complete the tasks, you need to apply data preprocessing methods discussed in Lecture 2. However, you may need to research to find solutions to some tasks.
- You can duplicate the code and report cells if you need more than one code/report cell for your solution

Tasks

TASK 2.1: Download the adult dataset from Canvas, import it into Jupyter Notebook, and complete the following steps (Completing the report cell is required):

a. Write a code to show how many Null values each column contains. Report the columns that contain Null values in the report cell (Hint: use the isna().sum() method to show the number of the Null values) (NOTE: Nan values are also considered as Null values) (1%).

b. Change the display setting and print the first 100 rows. Find columns that contain wrong data and report them in the report cell (1%).

c. We cannot inspect the entire data by eyes for columns containing wrong data. Explain in the report cell what method we could use to find all columns which contain wrong data (2%)

In [1]:
```
############# WRITE YOUR CODE IN THIS CELL (IF APPLICABLE)####################
import pandas as pd
import numpy as np

# importing our datasets
data = pd.read_csv('adult.csv')

#...A....TOTAL NULL VALUES IN EACH COLUMNS
print("\n TOTAL NULL VALUES")
print (data.isna().sum())
```

```
#....B....print the first 100 ROWS and change the display setting
pd.set_option('display.max_rows',100)

print("\n FIRST 100 ROWS")
print(data.head(100))
```

```
 TOTAL NULL VALUES
age                 0
workclass         963
fnlwgt              0
education           0
education-num       0
marital-status      0
occupation        966
relationship        0
race                0
sex                 0
capital-gain        0
capital-loss        0
hours-per-week      0
native-country    274
income              0
dtype: int64

 FIRST 100 ROWS
    age         workclass  fnlwgt     education  education-num  \
0    39         State-gov   77516     Bachelors             13
1    50  Self-emp-not-inc   83311     Bachelors             13
2    38           Private  215646       HS-grad              9
3    53           Private  234721          11th              7
4    28           Private  338409     Bachelors             13
5    37           Private  284582       Masters             14
6    49           Private  160187           9th              5
7    52  Self-emp-not-inc  209642       HS-grad              9
8    31           Private   45781       Masters             14
9    42           Private  159449     Bachelors             13
10   37           Private  280464  Some-college             10
11   30         State-gov  141297     Bachelors             13
12   23           Private  122272     Bachelors             13
13   32           Private  205019    Assoc-acdm             12
14   40           Private  121772     Assoc-voc             11
15   34           Private  245487       7th-8th              4
16   25  Self-emp-not-inc  176756       HS-grad              9
17   32           Private  186824       HS-grad              9
18   38           Private   28887          11th              7
19   43  Self-emp-not-inc  292175       Masters             14
20   40           Private  193524     Doctorate             16
21   54           Private  302146       HS-grad              9
22   35       Federal-gov   76845           9th              5
23   43           Private  117037          11th              7
24   59           Private  109015       HS-grad              9
25   56         Local-gov  216851     Bachelors             13
26   19           Private  168294       HS-grad              9
27   54                 ?  180211  Some-college             10
28   39           Private  367260       HS-grad              9
29   49           Private  193366       HS-grad              9
30   23         Local-gov  190709    Assoc-acdm             12
31   20           Private  266015  Some-college             10
32   45           Private  386940     Bachelors             13
33   30       Federal-gov   59951  Some-college             10
34   22         State-gov  311512  Some-college             10
35   48           Private  242406          11th              7
36   21           Private  197200  Some-college             10
37   19           Private  544091       HS-grad              9
38   31           Private   84154  Some-college             10
39   48  Self-emp-not-inc  265477    Assoc-acdm             12
40   31           Private  507875           9th              5
41   53  Self-emp-not-inc   88506     Bachelors             13
42   24           Private  172987     Bachelors             13
43   49           Private   94638       HS-grad              9
```

```
44   25           Private  289980       HS-grad             9
45   57       Federal-gov  337895     Bachelors            13
46   53           Private  144361       HS-grad             9
47   44           Private  128354       Masters            14
48   41         State-gov  101603     Assoc-voc            11
49   29           Private  271466     Assoc-voc            11
50   25           Private   32275  Some-college            10
51   18           Private  226956       HS-grad             9
52   47           Private   51835   Prof-school            15
53   50       Federal-gov  251585     Bachelors            13
54   47      Self-emp-inc  109832       HS-grad             9
55   43           Private  237993  Some-college            10
56   46           Private  216666       5th-6th             3
57   35           Private   56352     Assoc-voc            11
58   41           Private  147372       HS-grad             9
59   30           Private  188146       HS-grad             9
60   30           Private   59496     Bachelors            13
61   32                 ?  293936       7th-8th             4
62   48           Private  149640       HS-grad             9
63   42           Private  116632     Doctorate            16
64   29           Private  105598  Some-college            10
65   36           Private  155537       HS-grad             9
66   28           Private  183175  Some-college            10
67   53           Private  169846       HS-grad             9
68   49      Self-emp-inc  191681  Some-college            10
69   25                 ?  200681  Some-college            10
70   19           Private  101509  Some-college            10
71   31           Private  309974     Bachelors            13
72   29  Self-emp-not-inc  162298     Bachelors            13
73   23           Private  211678  Some-college            10
74   79           Private  124744  Some-college            10
75   27           Private  213921       HS-grad             9
76   40           Private   32214    Assoc-acdm            12
77   67                 ?  212759          10th             6
78   18           Private  309634          11th             7
79   31         Local-gov  125927       7th-8th             4
80   18           Private  446839       HS-grad             9
81   52           Private  276515     Bachelors            13
82   46           Private   51618       HS-grad             9
83   59           Private  159937       HS-grad             9
84   44           Private  343591       HS-grad             9
85   53           Private  346253       HS-grad             9
86   49         Local-gov  268234       HS-grad             9
87   33           Private  202051       Masters            14
88   30           Private   54334           9th             5
89   43       Federal-gov  410867     Doctorate            16
90   57           Private  249977     Assoc-voc            11
91   37           Private  286730  Some-college            10
92   28           Private  212563  Some-college            10
93   30           Private  117747       HS-grad             9
94   34         Local-gov  226296     Bachelors            13
95   29         Local-gov  115585  Some-college            10
96   48  Self-emp-not-inc  191277     Doctorate            16
97   37           Private  202683  Some-college            10
98   48           Private  171095    Assoc-acdm            12
99   32       Federal-gov  249409       HS-grad             9

           marital-status         occupation    relationship  \
0           Never-married       Adm-clerical   Not-in-family
1      Married-civ-spouse    Exec-managerial         Husband
2                Divorced  Handlers-cleaners   Not-in-family
3      Married-civ-spouse  Handlers-cleaners         Husband
4      Married-civ-spouse     Prof-specialty            Wife
5      Married-civ-spouse    Exec-managerial            Wife
```

6	Married-spouse-absent	Other-service	Not-in-family
7	Married-civ-spouse	Exec-managerial	Husband
8	Never-married	Prof-specialty	Not-in-family
9	Married-civ-spouse	Exec-managerial	Husband
10	Married-civ-spouse	Exec-managerial	Husband
11	Married-civ-spouse	Prof-specialty	Husband
12	Never-married	Adm-clerical	Own-child
13	Never-married	Sales	Not-in-family
14	Married-civ-spouse	Craft-repair	Husband
15	Married-civ-spouse	Transport-moving	Husband
16	Never-married	Farming-fishing	Own-child
17	Never-married	Machine-op-inspct	Unmarried
18	Married-civ-spouse	Sales	Husband
19	Divorced	Exec-managerial	Unmarried
20	Married-civ-spouse	Prof-specialty	Husband
21	Separated	Other-service	Unmarried
22	Married-civ-spouse	Farming-fishing	Husband
23	Married-civ-spouse	Transport-moving	Husband
24	Divorced	Tech-support	Unmarried
25	Married-civ-spouse	Tech-support	Husband
26	Never-married	Craft-repair	Own-child
27	Married-civ-spouse	?	Husband
28	Divorced	Exec-managerial	Not-in-family
29	Married-civ-spouse	Craft-repair	Husband
30	Never-married	Protective-serv	Not-in-family
31	Never-married	Sales	Own-child
32	Divorced	Exec-managerial	Own-child
33	Married-civ-spouse	Adm-clerical	Own-child
34	Married-civ-spouse	Other-service	Husband
35	Never-married	Machine-op-inspct	Unmarried
36	Never-married	Machine-op-inspct	Own-child
37	Married-AF-spouse	Adm-clerical	Wife
38	Married-civ-spouse	Sales	Husband
39	Married-civ-spouse	Prof-specialty	Husband
40	Married-civ-spouse	Machine-op-inspct	Husband
41	Married-civ-spouse	Prof-specialty	Husband
42	Married-civ-spouse	Tech-support	Husband
43	Separated	Adm-clerical	Unmarried
44	Never-married	Handlers-cleaners	Not-in-family
45	Married-civ-spouse	Prof-specialty	Husband
46	Married-civ-spouse	Machine-op-inspct	Husband
47	Divorced	Exec-managerial	Unmarried
48	Married-civ-spouse	Craft-repair	Husband
49	Never-married	Prof-specialty	Not-in-family
50	Married-civ-spouse	Exec-managerial	Wife
51	Never-married	Other-service	Own-child
52	Married-civ-spouse	Prof-specialty	Wife
53	Divorced	Exec-managerial	Not-in-family
54	Divorced	Exec-managerial	Not-in-family
55	Married-civ-spouse	Tech-support	Husband
56	Married-civ-spouse	Machine-op-inspct	Husband
57	Married-civ-spouse	Other-service	Husband
58	Married-civ-spouse	Adm-clerical	Husband
59	Married-civ-spouse	Machine-op-inspct	Husband
60	Married-civ-spouse	Sales	Husband
61	Married-spouse-absent	?	Not-in-family
62	Married-civ-spouse	Transport-moving	Husband
63	Married-civ-spouse	Prof-specialty	Husband
64	Divorced	Tech-support	Not-in-family
65	Married-civ-spouse	Craft-repair	Husband
66	Divorced	Adm-clerical	Not-in-family
67	Married-civ-spouse	Adm-clerical	Wife
68	Married-civ-spouse	Exec-managerial	Husband
69	Never-married	?	Own-child

70	Never-married	Prof-specialty	Own-child	
71	Separated	Sales	Own-child	
72	Married-civ-spouse	Sales	Husband	
73	Never-married	Machine-op-inspct	Not-in-family	
74	Married-civ-spouse	Prof-specialty	Other-relative	
75	Never-married	Other-service	Own-child	
76	Married-civ-spouse	Adm-clerical	Husband	
77	Married-civ-spouse	?	Husband	
78	Never-married	Other-service	Own-child	
79	Married-civ-spouse	Farming-fishing	Husband	
80	Never-married	Sales	Not-in-family	
81	Married-civ-spouse	Other-service	Husband	
82	Married-civ-spouse	Other-service	Wife	
83	Married-civ-spouse	Sales	Husband	
84	Divorced	Craft-repair	Not-in-family	
85	Divorced	Sales	Own-child	
86	Married-civ-spouse	Protective-serv	Husband	
87	Married-civ-spouse	Prof-specialty	Husband	
88	Never-married	Sales	Not-in-family	
89	Never-married	Prof-specialty	Not-in-family	
90	Married-civ-spouse	Prof-specialty	Husband	
91	Divorced	Craft-repair	Unmarried	
92	Divorced	Machine-op-inspct	Unmarried	
93	Married-civ-spouse	Sales	Wife	
94	Married-civ-spouse	Protective-serv	Husband	
95	Never-married	Handlers-cleaners	Not-in-family	
96	Married-civ-spouse	Prof-specialty	Husband	
97	Married-civ-spouse	Sales	Husband	
98	Divorced	Exec-managerial	Unmarried	
99	Never-married	Other-service	Own-child	

	race	sex	capital-gain	capital-loss	hours-per-week	\
0	White	Male	2174	0	40	
1	White	Male	0	0	13	
2	White	Male	0	0	40	
3	Black	Male	0	0	40	
4	Black	Female	0	0	40	
5	White	Female	0	0	40	
6	Black	Female	0	0	16	
7	White	Male	0	0	45	
8	White	Female	14084	0	50	
9	White	Male	5178	0	40	
10	Black	Male	0	0	80	
11	Asian-Pac-Islander	Male	0	0	40	
12	White	Female	0	0	30	
13	Black	Male	0	0	50	
14	Asian-Pac-Islander	Male	0	0	40	
15	Amer-Indian-Eskimo	Male	0	0	45	
16	White	Male	0	0	35	
17	White	Male	0	0	40	
18	White	Male	0	0	50	
19	White	Female	0	0	45	
20	White	Male	0	0	60	
21	Black	Female	0	0	20	
22	Black	Male	0	0	40	
23	White	Male	0	2042	40	
24	White	Female	0	0	40	
25	White	Male	0	0	40	
26	White	Male	0	0	40	
27	Asian-Pac-Islander	Male	0	0	60	
28	White	Male	0	0	80	
29	White	Male	0	0	40	
30	White	Male	0	0	52	
31	Black	Male	0	0	44	

32	White	Male	0	1408	40
33	White	Male	0	0	40
34	Black	Male	0	0	15
35	White	Male	0	0	40
36	White	Male	0	0	40
37	White	Female	0	0	25
38	White	Male	0	0	38
39	White	Male	0	0	40
40	White	Male	0	0	43
41	White	Male	0	0	40
42	White	Male	0	0	50
43	White	Female	0	0	40
44	White	Male	0	0	35
45	Black	Male	0	0	40
46	White	Male	0	0	38
47	White	Female	0	0	40
48	White	Male	0	0	40
49	White	Male	0	0	43
50	Other	Female	0	0	40
51	White	Female	0	0	30
52	White	Female	0	1902	60
53	White	Male	0	0	55
54	White	Male	0	0	60
55	White	Male	0	0	40
56	White	Male	0	0	40
57	White	Male	0	0	40
58	White	Male	0	0	48
59	White	Male	5013	0	40
60	White	Male	2407	0	40
61	White	Male	0	0	40
62	White	Male	0	0	40
63	White	Male	0	0	45
64	White	Male	0	0	58
65	White	Male	0	0	40
66	White	Female	0	0	40
67	White	Female	0	0	40
68	White	Male	0	0	50
69	White	Male	0	0	40
70	White	Male	0	0	32
71	Black	Female	0	0	40
72	White	Male	0	0	70
73	White	Male	0	0	40
74	White	Male	0	0	20
75	White	Male	0	0	40
76	White	Male	0	0	40
77	White	Male	0	0	2
78	White	Female	0	0	22
79	White	Male	0	0	40
80	White	Male	0	0	30
81	White	Male	0	0	40
82	White	Female	0	0	40
83	White	Male	0	0	48
84	White	Female	14344	0	40
85	White	Female	0	0	35
86	White	Male	0	0	40
87	White	Male	0	0	50
88	White	Male	0	0	40
89	White	Female	0	0	50
90	White	Male	0	0	40
91	White	Female	0	0	40
92	Black	Female	0	0	25
93	Asian-Pac-Islander	Female	0	1573	35
94	White	Male	0	0	40
95	White	Male	0	0	50

96		White	Male	0	1902	60
97		White	Male	0	0	48
98		White	Female	0	0	40
99		Black	Male	0	0	40

	native-country	income
0	United-States	<=50K
1	United-States	<=50K
2	United-States	<=50K
3	United-States	<=50K
4	Cuba	<=50K
5	United-States	<=50K
6	Jamaica	<=50K
7	United-States	>50K
8	United-States	>50K
9	United-States	>50K
10	United-States	>50K
11	India	>50K
12	United-States	<=50K
13	United-States	<=50K
14	?	>50K
15	Mexico	<=50K
16	United-States	<=50K
17	United-States	<=50K
18	United-States	<=50K
19	United-States	>50K
20	United-States	>50K
21	United-States	<=50K
22	United-States	<=50K
23	United-States	<=50K
24	United-States	<=50K
25	United-States	>50K
26	United-States	<=50K
27	South	>50K
28	United-States	<=50K
29	United-States	<=50K
30	United-States	<=50K
31	United-States	<=50K
32	United-States	<=50K
33	United-States	<=50K
34	United-States	<=50K
35	Puerto-Rico	<=50K
36	United-States	<=50K
37	United-States	<=50K
38	?	>50K
39	United-States	<=50K
40	United-States	<=50K
41	United-States	<=50K
42	United-States	<=50K
43	United-States	<=50K
44	United-States	<=50K
45	United-States	>50K
46	United-States	<=50K
47	United-States	<=50K
48	United-States	<=50K
49	United-States	<=50K
50	United-States	<=50K
51	?	<=50K
52	Honduras	>50K
53	United-States	>50K
54	United-States	<=50K
55	United-States	>50K
56	Mexico	<=50K
57	Puerto-Rico	<=50K

58	United-States	<=50K
59	United-States	<=50K
60	United-States	<=50K
61	?	<=50K
62	United-States	<=50K
63	United-States	>50K
64	United-States	<=50K
65	United-States	<=50K
66	United-States	<=50K
67	United-States	>50K
68	United-States	>50K
69	United-States	<=50K
70	United-States	<=50K
71	United-States	<=50K
72	United-States	>50K
73	United-States	<=50K
74	United-States	<=50K
75	Mexico	<=50K
76	United-States	<=50K
77	United-States	<=50K
78	United-States	<=50K
79	United-States	<=50K
80	United-States	<=50K
81	Cuba	<=50K
82	United-States	<=50K
83	United-States	<=50K
84	United-States	>50K
85	United-States	<=50K
86	United-States	>50K
87	United-States	<=50K
88	United-States	<=50K
89	United-States	>50K
90	United-States	<=50K
91	United-States	<=50K
92	United-States	<=50K
93	?	<=50K
94	United-States	>50K
95	United-States	<=50K
96	United-States	>50K
97	United-States	>50K
98	England	<=50K
99	United-States	<=50K

############## WRITE YOUR REPORT IN THIS CELL (IF APPLICABLE)#################### A) Thre columns are reported with null values, these includes *Workclass with total number of 963 missing values *Oocuupation = 966 *Native-country = 274 B)Education has some numerical values like 11th, 15th,6th.....instead of characters Workclass could be containing wrong data has some values are replaced with question mark (?) Age also has some value of 17 which could be an underage (wrongdata) given that this is expected to be an adult dataset. C)To view and explore the data set properly we can use the "info" function i.e print(data.info()). This helps in projecting some details that might not be obvios while viewing the head and tail of the datasets,the info function gives infromations on data columns,number of non-null,datatype, that way we know the specific columns that needs more pre-processing also known as data wrangling. we can also use the "display.max" function to view more specific number of rows and columns for better view. Another way is to use the "describe" function which can be used to identify wrong data in numerical variables, or value counts for categorical variables.

TASK 2.2: Complete the following tasks (Completing the report cell is required):

a. Determine whether the columns which contain Null values are numerical, nominal, or ordinal variables and report it in

the report cell (1%).

b. Delete all rows which contain Null values (1%). Then print the number of Null values of all columns using the isna().sum() method.

c. Import the dataset again. Find the mode of the categorical (i.e. nominal or ordinal) columns which contain Null values and use it to fill their Null values (2%)

In [2]:
```
############# WRITE YOUR CODE IN THIS CELL (IF APPLICABLE)####################
#A.determining columns with null values and the variable type
Columns_Null = data.columns[data.isna().any()].tolist()

#The datatype of the above columns with null
for column in Columns_Null:
    data_type = data[column].dtype
    print(f"{column}: {data_type}")
```

```
workclass: object
occupation: object
native-country: object
```

In [3]:
```
#Determining the columns variable type and Null values

print("\n DATATYPE INFROMATIONS")
print(data.info())
print("\n TOTAL NULL")
print (data.isna().sum())
```

```
 DATATYPE INFROMATIONS
<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       47879 non-null  object
 2   fnlwgt          48842 non-null  int64
 3   education       48842 non-null  object
 4   education-num   48842 non-null  int64
 5   marital-status  48842 non-null  object
 6   occupation      47876 non-null  object
 7   relationship    48842 non-null  object
 8   race            48842 non-null  object
 9   sex             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  48568 non-null  object
 14  income          48842 non-null  object
dtypes: int64(6), object(9)
memory usage: 5.6+ MB
None

 TOTAL NULL
age                 0
workclass         963
fnlwgt              0
education           0
education-num       0
marital-status      0
occupation        966
relationship        0
race                0
sex                 0
capital-gain        0
capital-loss        0
hours-per-week      0
native-country    274
income              0
dtype: int64
```

In [4]:

```python
#QUESTION B
#Deleting number of null values and assigning it to a new data
data_dropped = data.dropna()

#printing the total null values of all columns
print(data_dropped.isna().sum())
```

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

In [5]:
```
#QUESTION C
# RE-importing our datasets
REIMPORTED_data = pd.read_csv('adult.csv')


#Selecting the categorical columns with null values
Columns_CN = ['native-country','workclass','occupation']

#fill the above columns with the mode
for column in Columns_CN:
    Mode_replace = REIMPORTED_data[column].mode()[0]
    REIMPORTED_data[column].fillna(Mode_replace,inplace=True)


REIMPORTED_data.isna().sum()
```

Out[5]:
```
age               0
workclass         0
fnlwgt            0
education         0
education-num     0
marital-status    0
occupation        0
relationship      0
race              0
sex               0
capital-gain      0
capital-loss      0
hours-per-week    0
native-country    0
income            0
dtype: int64
```

############## WRITE YOUR REPORT IN THIS CELL (IF APPLICABLE)#################### A)Below are the columns containing NULL values and the type of variables they contain WORKCLASS- 963 -Null values- Nominal Variable OCCUPATION- 966 -Null Values - Nominal Variable NATIVE-COUNTRY - 274 -Null Values - Nominal Variable The above columns falls under the category of nominal variabe as their respective values of groups are without inherent order or ranking. The above variables are all categorical variables B)We deleted all Nullvalues using the "dropna" command C)We reimported our dataset and named it REIMPORTED_data,and replaced all null values with their mode,replacing the null values still maintains the number of rows in the dataset while the dropna command would have cutshort the number of rows in the dataset.

TASK 2.3. Select one of the columns which contains wrong data. Write a code to fill the wrong cells with an appropriate value. You have the freedom to

determine what value you want to use to fill the wrong cells (2%).

In [6]:
```
############# WRITE YOUR CODE IN THIS CELL (IF APPLICABLE)####################
#Select the column containing wrong data
column_with_wrong_data = 'age'

# Replace ages less than 18 with 20
REIMPORTED_data.loc[REIMPORTED_data[column_with_wrong_data]< 18, column_with_wrong_

# Verify the changes
print("Age group below 18 replced with 20:", (REIMPORTED_data[column_with_wrong_dat
```

```
Age group below 18 replced with 20: 0
```

############# WRITE YOUR REPORT IN THIS CELL (IF APPLICABLE)#################### The above line of code can be used to replace wrong data in the age column with 20, the decision was concluded using the basis of adult age group which starts from 18, and individuals below 18 can be grouped as a minor, which could be a wrong data in and ADULT Dataset as we have some variables of 17. Note: The above decision is not based on the average age or most frequently occured.

TASK 2.4: Complete the following tasks:

a. Assume we want to predict the income column as the output. Use the correct method to encode this column (1%).

b. Select one ordinal column and encode it using the appropriate method (1%) .

c. Select one nominal column and encode it using the appropriate method (1%).

d. After encoding the nominal column in the previous step, find a method to append the encoded data to the dataset (2%).

e. Normalise the numerical columns of the dataset (1%)

In [7]:
```
############# WRITE YOUR CODE IN THIS CELL (IF APPLICABLE)####################
#Importing Neccessary
from sklearn import preprocessing
from sklearn.preprocessing import OrdinalEncoder
from sklearn.preprocessing import OneHotEncoder
from sklearn.preprocessing import normalize
from sklearn.preprocessing import MinMaxScaler


#A
#initialize the labelEncoder
le=preprocessing.LabelEncoder()
REIMPORTED_data['coded_income'] = le.fit_transform(REIMPORTED_data['income'])
REIMPORTED_data.head()
```

Out[7]:

	age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	s
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Ma
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Ma
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Ma
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Ma
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Fema

In [8]:
```
#B Ordinal Encoder, Column;Education
encoder = OrdinalEncoder()
REIMPORTED_data['education_encoded'] = encoder.fit_transform(REIMPORTED_data['educa
REIMPORTED_data.head()
```

Out[8]:

	age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	s
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Ma
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Ma
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Ma
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Ma
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Fema

In [9]:
```
#C Nominal Encoder ,column;RACE
encoder = OneHotEncoder(drop='first', sparse = False)
REIMPORTED_data['race']  = encoder.fit_transform(data['race'].values.reshape(-1,1))
REIMPORTED_data.head()
```

```
C:\Users\dessy\anaconda3\Lib\site-packages\sklearn\preprocessing\_encoders.py:972:
FutureWarning: `sparse` was renamed to `sparse_output` in version 1.2 and will be
removed in 1.4. `sparse_output` is ignored unless you leave `sparse` to its defaul
t value.
  warnings.warn(
```

Out[9]:

	age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	sex
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	0.0	Male
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	0.0	Male
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	0.0	Male
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	0.0	Male
4	28	Private	338409	Bachelors	13	Married-civ-	Prof-specialty	Wife	0.0	Female

In [10]:
```
#    QUESTION D

#Encode Race using one-hot encoding
race_encoded = pd.get_dummies(REIMPORTED_data['race'], prefix ='ethnicity')

#concatenate the original dataframe with the encoded column
REIMPORTED_data = pd.concat([REIMPORTED_data,race_encoded], axis =1)

#view the new dataframe
print(REIMPORTED_data.head())
```

```
   age         workclass  fnlwgt  education  education-num  \
0   39         State-gov   77516  Bachelors             13
1   50  Self-emp-not-inc   83311  Bachelors             13
2   38           Private  215646    HS-grad              9
3   53           Private  234721       11th              7
4   28           Private  338409  Bachelors             13

       marital-status         occupation   relationship  race     sex  \
0       Never-married       Adm-clerical  Not-in-family   0.0    Male
1  Married-civ-spouse    Exec-managerial        Husband   0.0    Male
2            Divorced  Handlers-cleaners  Not-in-family   0.0    Male
3  Married-civ-spouse  Handlers-cleaners        Husband   0.0    Male
4  Married-civ-spouse     Prof-specialty           Wife   0.0  Female

   capital-gain  capital-loss  hours-per-week native-country income  \
0          2174             0              40  United-States  <=50K
1             0             0              13  United-States  <=50K
2             0             0              40  United-States  <=50K
3             0             0              40  United-States  <=50K
4             0             0              40           Cuba  <=50K

   coded_income  education_encoded  ethnicity_0.0  ethnicity_1.0
0             0                9.0           True          False
1             0                9.0           True          False
2             0               11.0           True          False
3             0                1.0           True          False
4             0                9.0           True          False
```

In [18]:
```
#E Normalize the numerical columns

# Select the columns you want to normalize
```

```
columns_to_normalize = ['age', 'fnlwgt','education-num','capital-gain','capital-los

# Initialize the MinMaxScaler
scaler = MinMaxScaler()

# Fit and transform the selected columns
REIMPORTED_data[columns_to_normalize] = scaler.fit_transform(REIMPORTED_data[columr

# Print the normalized DataFrame
print("\n NORMALIZED DATASET")
print(REIMPORTED_data.head())
```

```
 NORMALIZED DATASET
        age         workclass    fnlwgt  education  education-num  \
0  0.291667         State-gov  0.044131  Bachelors       0.800000
1  0.444444  Self-emp-not-inc  0.048052  Bachelors       0.800000
2  0.277778           Private  0.137581    HS-grad       0.533333
3  0.486111           Private  0.150486       11th       0.400000
4  0.138889           Private  0.220635  Bachelors       0.800000

       marital-status         occupation   relationship  race     sex  \
0       Never-married       Adm-clerical  Not-in-family   0.0    Male
1  Married-civ-spouse    Exec-managerial        Husband   0.0    Male
2            Divorced  Handlers-cleaners  Not-in-family   0.0    Male
3  Married-civ-spouse  Handlers-cleaners        Husband   0.0    Male
4  Married-civ-spouse     Prof-specialty           Wife   0.0  Female

   capital-gain  capital-loss  hours-per-week native-country income  \
0       0.02174           0.0        0.397959  United-States  <=50K
1       0.00000           0.0        0.122449  United-States  <=50K
2       0.00000           0.0        0.397959  United-States  <=50K
3       0.00000           0.0        0.397959  United-States  <=50K
4       0.00000           0.0        0.397959           Cuba  <=50K

   coded_income  education_encoded  ethnicity_0.0  ethnicity_1.0
0             0                9.0           True          False
1             0                9.0           True          False
2             0               11.0           True          False
3             0                1.0           True          False
4             0                9.0           True          False
```

############ WRITE YOUR REPORT IN THIS CELL (IF APPLICABLE)#################### Overall, encoding is important in many aspects of data processing, storage, and transmission in Python and other programming languages. It allows data to be represented, manipulated, and communicated effectively in a variety of contexts and environments. QUESTION A The income column has four variable type (<=50, >=50k. <=50k.,>=50k) this are four different variables in data ,upon encoding all variables ranges between 0,1,2,and 3. QUESTION B Education is an Ordinal variable as it can be ranked in order of hierarchy, all variables were encoded accordingly QUESTION C We had numerous nomial column in the dataset, i chose race as their is an adsence level of rank to ethnicity,The code's output demonstrates how the 'race' column was transformed into a one-hot encoded representation, with additional binary features representing each race category. QUESTION D After running the code, the output will show the first few rows of the dataframe, with the race column encoded with one-hot encoding.One-hot encoding is a technique for converting categorical variables into numerical representations for machine learning algorithms.It generates binary columns for each category in the original categorical variable, with one indicating the category's presence and zero indicating its absence.using "concat" funtion we append the column race Each distinct category in the 'race' column has been transformed into a binary column (also known as a dummy variable) prefixed by 'ethnicity_,Each row in the dataframe now includes binary indicators for the individual's ethnicity, allowing for further analysis or modelling tasks. QUESTION E Normalisation scales numerical features to a similar range, preventing larger magnitudes from dominating the learning algorithm. This ensures that all features make an equal contribution to the analysis or model training process.In the dataset out numerical columns are 'age', 'fnlwgt','education-num','capital-gain','capital-loss','hours-per-week', and has been normalized into similar numerical grouping for better machine learning understanding and scaling.