Phase 2

Prediction task is to determine whether a person makes over 50K a year.

In [3]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings

warnings.filterwarnings("ignore")

%matplotlib inline
```

Load clean dataset

In [4]:
```
df=pd.read_csv("data_adult_eda.csv")
```

In [5]:
```
df
```

Out[5]:

	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male	2174	0
0	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male	0	0
1	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male	0	0
2	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male	0	0
3	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female	0	0
4	37	Private	284582	Masters	14	Married-civ-spouse	Exec-managerial	Wife	White	Female	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...
32555	27	Private	257302	Assoc-acdm	12	Married-civ-spouse	Tech-support	Wife	White	Female	0	0
32556	40	Private	154374	HS-grad	9	Married-civ-spouse	Machine-op-inspct	Husband	White	Male	0	0
32557	58	Private	151910	HS-grad	9	Widowed	Adm-clerical	Unmarried	White	Female	0	0
32558	22	Private	201490	HS-grad	9	Never-married	Adm-clerical	Own-child	White	Male	0	0
32559	52	Self-emp-inc	287927	HS-grad	9	Married-civ-spouse	Exec-managerial	Wife	White	Female	15024	0

32560 rows × 15 columns

AS IT CAN BE SEEN THAT COLUMNS ARE NOT IN A RIGHT FORMAT SO FIRST RENAME ALL THE COLUMNS

In [6]:
```
df.columns
```

Out[6]:
```
Index(['39', ' State-gov', '77516', ' Bachelors', '13', ' Never-married',
       ' Adm-clerical', ' Not-in-family', ' White', ' Male', '2174', '0', '40',
       ' United-States', ' <=50K'],
      dtype='object')
```

In [7]:
```
df.columns = ['Age', 'workclass', 'fnlwgt', 'education','education-num','marital-statu
```

In [8]:
```
df.columns
```

Out[8]:
```
Index(['Age', 'workclass', 'fnlwgt', 'education', 'education-num',
       'marital-status', 'occupation', 'relationship', 'race', 'sex',
       'capital-gain', 'capital-loss', 'hours-per-week', 'native-country',
       'makes over'],
      dtype='object')
```

In [9]:
```
df
```

Out[9]:

	Age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	
0	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	I
1	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	I
2	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	I
3	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Fe
4	37	Private	284582	Masters	14	Married-civ-spouse	Exec-managerial	Wife	White	Fe
...	...	...	...	...	...	...	...	...	...	
32555	27	Private	257302	Assoc-acdm	12	Married-civ-spouse	Tech-support	Wife	White	Fe
32556	40	Private	154374	HS-grad	9	Married-civ-spouse	Machine-op-inspct	Husband	White	I
32557	58	Private	151910	HS-grad	9	Widowed	Adm-clerical	Unmarried	White	Fe
32558	22	Private	201490	HS-grad	9	Never-married	Adm-clerical	Own-child	White	I
32559	52	Self-emp-inc	287927	HS-grad	9	Married-civ-spouse	Exec-managerial	Wife	White	Fe

32560 rows × 15 columns

In [10]:
```
df_copy_new=df.copy()
```

In [11]:
```
df_copy_new.head()
```

Out[11]:

	Age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	sex
0	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male
1	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male
2	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male
3	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female
4	37	Private	284582	Masters	14	Married-civ-spouse	Exec-managerial	Wife	White	Female

In [12]:
```
df_copy_new.tail()
```

Out[12]:

	Age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	
32555	27	Private	257302	Assoc-acdm	12	Married-civ-spouse	Tech-support	Wife	White	Fe
32556	40	Private	154374	HS-grad	9	Married-civ-spouse	Machine-op-inspct	Husband	White	
32557	58	Private	151910	HS-grad	9	Widowed	Adm-clerical	Unmarried	White	Fe
32558	22	Private	201490	HS-grad	9	Never-married	Adm-clerical	Own-child	White	
32559	52	Self-emp-inc	287927	HS-grad	9	Married-civ-spouse	Exec-managerial	Wife	White	Fe

In [13]:
```
df_copy_new.shape
```

Out[13]: (32560, 15)

In [14]:
```
df_copy_new.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 32560 entries, 0 to 32559
Data columns (total 15 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   Age             32560 non-null  int64
 1   workclass       32560 non-null  object
 2   fnlwgt          32560 non-null  int64
 3   education       32560 non-null  object
 4   education-num   32560 non-null  int64
 5   marital-status  32560 non-null  object
 6   occupation      32560 non-null  object
 7   relationship    32560 non-null  object
 8   race            32560 non-null  object
 9   sex             32560 non-null  object
 10  capital-gain    32560 non-null  int64
 11  capital-loss    32560 non-null  int64
 12  hours-per-week  32560 non-null  int64
 13  native-country  32560 non-null  object
 14  makes over      32560 non-null  int64
dtypes: int64(7), object(8)
memory usage: 3.7+ MB
```

In [15]:
```
df_copy_new.describe()
```

Out[15]:

	Age	fnlwgt	education-num	capital-gain	capital-loss	hours-per-week	makes over
count	32560.000000	3.256000e+04	32560.000000	32560.000000	32560.000000	32560.000000	32560.0
mean	38.581634	1.897818e+05	10.080590	1077.615172	87.306511	40.437469	50000.0
std	13.640642	1.055498e+05	2.572709	7385.402999	402.966116	12.347618	0.0
min	17.000000	1.228500e+04	1.000000	0.000000	0.000000	1.000000	50000.0
25%	28.000000	1.178315e+05	9.000000	0.000000	0.000000	40.000000	50000.0
50%	37.000000	1.783630e+05	10.000000	0.000000	0.000000	40.000000	50000.0
75%	48.000000	2.370545e+05	12.000000	0.000000	0.000000	45.000000	50000.0
max	90.000000	1.484705e+06	16.000000	99999.000000	4356.000000	99.000000	50000.0

In [16]:
```
df_copy_new.columns
```

Out[16]:
```
Index(['Age', 'workclass', 'fnlwgt', 'education', 'education-num',
       'marital-status', 'occupation', 'relationship', 'race', 'sex',
       'capital-gain', 'capital-loss', 'hours-per-week', 'native-country',
       'makes over'],
      dtype='object')
```

Duplicate value of Entire dataset

In [17]:
```
df_copy_new[df_copy_new.duplicated()]
```

Out[17]:

	Age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	
4880	25	Private	308144	Bachelors	13	Never-married	Craft-repair	Not-in-family	White	
5103	90	Private	52386	Some-college	10	Never-married	Other-service	Not-in-family	Asian-Pac-Islander	
9170	21	Private	250051	Some-college	10	Never-married	Prof-specialty	Own-child	White	F
11630	20	Private	107658	Some-college	10	Never-married	Tech-support	Not-in-family	White	F
13083	25	Private	195994	1st-4th	2	Never-married	Priv-house-serv	Not-in-family	White	F
15058	21	Private	243368	Preschool	1	Never-married	Farming-fishing	Not-in-family	White	
17039	46	Private	173243	HS-grad	9	Married-civ-spouse	Craft-repair	Husband	White	
18554	30	Private	144593	HS-grad	9	Never-married	Other-service	Not-in-family	Black	
18697	19	Private	97261	HS-grad	9	Never-married	Farming-fishing	Not-in-family	White	
21317	19	Private	138153	Some-college	10	Never-married	Adm-clerical	Own-child	White	F
21489	19	Private	146679	Some-college	10	Never-married	Exec-managerial	Own-child	Black	
21874	49	Private	31267	7th-8th	4	Married-civ-spouse	Craft-repair	Husband	White	
22299	25	Private	195994	1st-4th	2	Never-married	Priv-house-serv	Not-in-family	White	F
22366	44	Private	367749	Bachelors	13	Never-married	Prof-specialty	Not-in-family	White	F
22493	49	Self-emp-not-inc	43479	Some-college	10	Married-civ-spouse	Craft-repair	Husband	White	
22760	39	Private	138192	Bachelors	13	Married-civ-spouse	Craft-repair	Husband	White	
25871	23	Private	240137	5th-6th	3	Never-married	Handlers-cleaners	Not-in-family	White	
26312	28	Private	274679	Masters	14	Never-married	Prof-specialty	Not-in-family	White	
28229	27	Private	255582	HS-grad	9	Never-married	Machine-op-inspct	Not-in-family	White	F

	Age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	
28521	42	Private	204235	Some-college	10	Married-civ-spouse	Prof-specialty	Husband	White	
28845	39	Private	30916	HS-grad	9	Married-civ-spouse	Craft-repair	Husband	White	
29156	38	Private	207202	HS-grad	9	Married-civ-spouse	Machine-op-inspct	Husband	White	
30844	46	Private	133616	Some-college	10	Divorced	Adm-clerical	Unmarried	White	F
31992	19	Private	251579	Some-college	10	Never-married	Other-service	Own-child	White	
32403	35	Private	379959	HS-grad	9	Divorced	Other-service	Not-in-family	White	F

In [18]:
```
df_copy_new.columns
```

Out[18]:
```
Index(['Age', 'workclass', 'fnlwgt', 'education', 'education-num',
       'marital-status', 'occupation', 'relationship', 'race', 'sex',
       'capital-gain', 'capital-loss', 'hours-per-week', 'native-country',
       'makes over'],
      dtype='object')
```

Duplicate value of perticular column

In [19]:
```
df_copy_new[df_copy_new.duplicated('fnlwgt')]
```

Out[19]:

	Age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	
442	44	Private	116632	Some-college	10	Married-civ-spouse	Prof-specialty	Husband	White	I
459	44	Private	116632	Assoc-acdm	12	Never-married	Farming-fishing	Own-child	White	I
593	23	Private	117789	Bachelors	13	Never-married	Adm-clerical	Own-child	White	Fe
679	29	Local-gov	92262	HS-grad	9	Never-married	Protective-serv	Own-child	White	I
718	34	Private	217460	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	I
...	...	...	...	...	...	...	...	...	...	
32554	22	Private	310152	Some-college	10	Never-married	Protective-serv	Not-in-family	White	I
32556	40	Private	154374	HS-grad	9	Married-civ-spouse	Machine-op-inspct	Husband	White	I
32557	58	Private	151910	HS-grad	9	Widowed	Adm-clerical	Unmarried	White	Fe
32558	22	Private	201490	HS-grad	9	Never-married	Adm-clerical	Own-child	White	I
32559	52	Self-emp-inc	287927	HS-grad	9	Married-civ-spouse	Exec-managerial	Wife	White	Fe

10913 rows × 15 columns

Drop 'fnlwgt' column

In [20]:
```
df_copy_new=df_copy_new.drop_duplicates(subset=['fnlwgt'],keep="first")
```

In [21]:
```
df_copy_new
```

Out[21]:

	Age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	
0	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	
1	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	
2	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	
3	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Fe
4	37	Private	284582	Masters	14	Married-civ-spouse	Exec-managerial	Wife	White	Fe
...	...	...	...	...	...	...	...	...	...	
32541	72	?	129912	HS-grad	9	Married-civ-spouse	?	Husband	White	
32548	43	State-gov	255835	Some-college	10	Divorced	Adm-clerical	Other-relative	White	Fe
32550	32	Private	34066	10th	6	Married-civ-spouse	Handlers-cleaners	Husband	Amer-Indian-Eskimo	
32551	43	Private	84661	Assoc-voc	11	Married-civ-spouse	Sales	Husband	White	
32555	27	Private	257302	Assoc-acdm	12	Married-civ-spouse	Tech-support	Wife	White	Fe

21647 rows × 15 columns

In [22]:
```
df_copy_new.shape
```

Out[22]: (21647, 15)

In [23]:
```
df_copy_new.head()
```

Out[23]:

	Age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	sex
0	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male
1	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male
2	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male
3	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female
4	37	Private	284582	Masters	14	Married-civ-spouse	Exec-managerial	Wife	White	Female

In [24]:
```
df_copy_new.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 21647 entries, 0 to 32555
Data columns (total 15 columns):
 #   Column          Non-Null Count  Dtype 
---  ------          --------------  ----- 
 0   Age             21647 non-null  int64 
 1   workclass       21647 non-null  object
 2   fnlwgt          21647 non-null  int64 
 3   education       21647 non-null  object
 4   education-num   21647 non-null  int64 
 5   marital-status  21647 non-null  object
 6   occupation      21647 non-null  object
 7   relationship    21647 non-null  object
 8   race            21647 non-null  object
 9   sex             21647 non-null  object
 10  capital-gain    21647 non-null  int64 
 11  capital-loss    21647 non-null  int64 
 12  hours-per-week  21647 non-null  int64 
 13  native-country  21647 non-null  object
 14  makes over      21647 non-null  int64 
dtypes: int64(7), object(8)
memory usage: 2.6+ MB
```

Segregate on the basis numerical and categorical data

In [25]:
```
numeric_features=[feature for feature in  df_copy_new.columns if df_copy_new[feature].
```

In [26]:
```
categoric_features=[feature for feature in df_copy_new.columns if df_copy_new[feature]
```

In [27]:
```
numeric_features
```

Out[27]:
```
['Age',
 'fnlwgt',
 'education-num',
 'capital-gain',
 'capital-loss',
 'hours-per-week',
 'makes over']
```

In [28]:
```
categoric_features
```

Out[28]:
```
['workclass',
 'education',
 'marital-status',
 'occupation',
 'relationship',
 'race',
 'sex',
 'native-country']
```

In [29]:
```
df_copy_new.head()
```

Out[29]:

	Age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	sex
0	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male
1	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male
2	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male
3	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female
4	37	Private	284582	Masters	14	Married-civ-spouse	Exec-managerial	Wife	White	Female

In [30]:
```
df_copy_new.columns
```

Out[30]:
```
Index(['Age', 'workclass', 'fnlwgt', 'education', 'education-num',
       'marital-status', 'occupation', 'relationship', 'race', 'sex',
       'capital-gain', 'capital-loss', 'hours-per-week', 'native-country',
       'makes over'],
      dtype='object')
```

Checking the counts of value of category present in specific categorical column

In [31]:
```
df_copy_new['sex'].value_counts()
```

Out[31]:
```
 Male      13585
 Female     8062
Name: sex, dtype: int64
```

Percentage of counts of value of category present in specific categorical column

In [32]:
```
df_copy_new['sex'].value_counts(normalize=True)*100
```

Out[32]:
```
 Male      62.756964
 Female    37.243036
Name: sex, dtype: float64
```

Check Percentage of counts of values of category for all the categorical columns

In [33]:
```
for col in categoric_features:
    print(f"{col},{df_copy_new[col].value_counts(normalize=True)*100}")
    print("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~")
```

```
workclass,Private             69.099644
Self-emp-not-inc     7.576108
?                    6.564420
Local-gov            6.361159
State-gov            4.213055
Self-emp-inc         3.169030
Federal-gov          2.951910
Without-pay          0.046196
Never-worked         0.018478
Name: workclass, dtype: float64
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
education, HS-grad         32.401718
 Some-college    22.534300
 Bachelors       15.475586
 Masters          4.905992
 Assoc-voc        4.180718
 11th             4.102185
 10th             3.256802
 Assoc-acdm       3.145932
 7th-8th          2.235876
 9th              1.769298
 Prof-school      1.570656
 12th             1.358156
 Doctorate        1.219569
 5th-6th          1.113318
 1st-4th          0.577447
 Preschool        0.152446
Name: education, dtype: float64
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
marital-status, Married-civ-spouse       43.202291
 Never-married            33.413406
 Divorced                 14.681018
 Widowed                   3.825010
 Separated                 3.469303
 Married-spouse-absent     1.339678
 Married-AF-spouse         0.069294
Name: marital-status, dtype: float64
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
occupation,Prof-specialty       12.126392
Adm-clerical         12.029381
Exec-managerial      11.775304
Craft-repair         11.650575
Sales                11.419596
Other-service        11.119324
?                     6.582898
Machine-op-inspct     6.347300
Transport-moving      4.596480
Handlers-cleaners     4.217675
Farming-fishing       3.025823
Tech-support          2.633159
Protective-serv       1.921744
Priv-house-serv       0.535871
Armed-Forces          0.018478
Name: occupation, dtype: float64
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
relationship, Husband           37.219938
 Not-in-family     26.345452
```

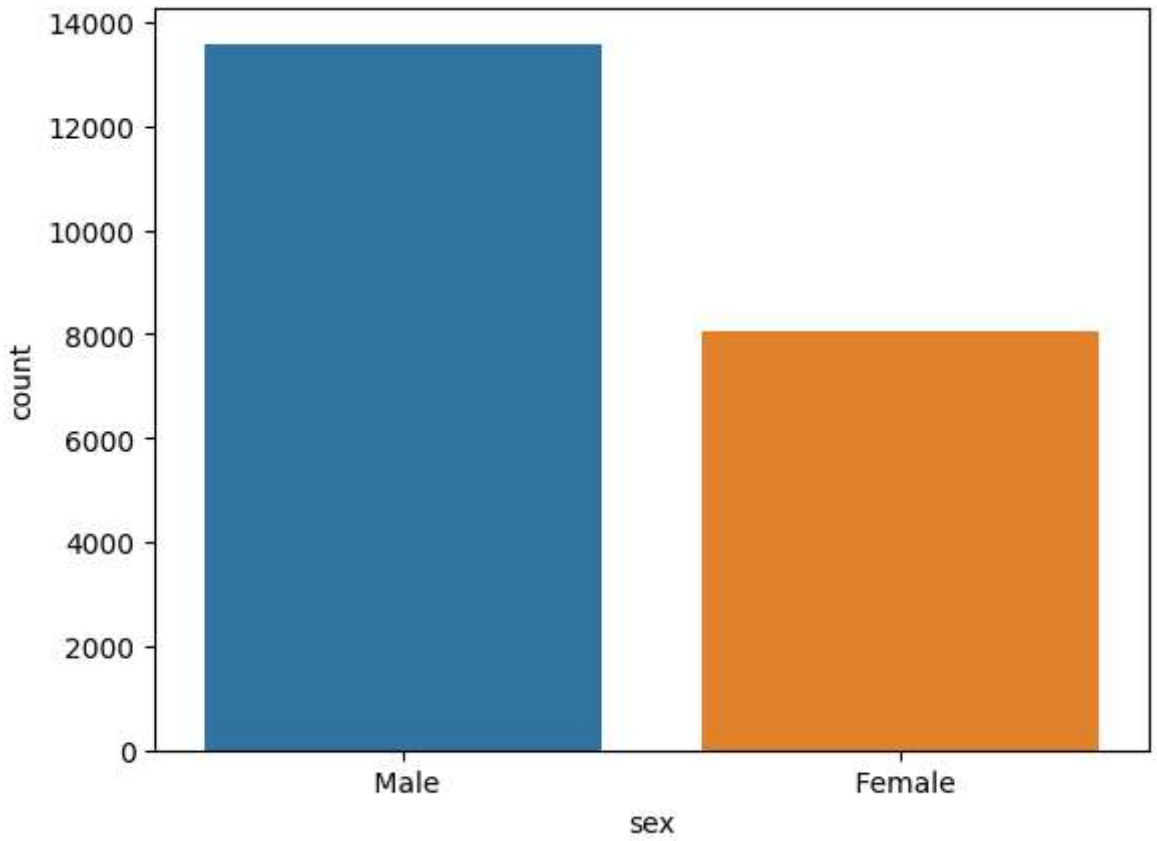
```
 Own-child         16.173142
 Unmarried         11.895413
 Wife               5.330993
 Other-relative     3.035063
Name: relationship, dtype: float64
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
race, White                 83.577401
 Black                 11.729108
 Asian-Pac-Islander     2.563866
 Amer-Indian-Eskimo     1.122557
 Other                  1.007068
Name: race, dtype: float64
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
sex, Male      62.756964
 Female    37.243036
Name: sex, dtype: float64
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
native-country,United-States                 89.864646
Mexico                         2.129625
?                              1.635331
Philippines                    0.425001
Germany                        0.425001
Puerto-Rico                    0.420382
Canada                         0.383425
Cuba                           0.360327
El-Salvador                    0.351088
England                        0.272555
Dominican-Republic             0.254077
Jamaica                        0.249457
India                          0.221740
Columbia                       0.217120
Japan                          0.198642
South                          0.198642
Guatemala                      0.189403
Vietnam                        0.184783
Poland                         0.170924
China                          0.170924
Italy                          0.161685
Haiti                          0.161685
Taiwan                         0.138587
Peru                           0.120109
Portugal                       0.110870
Nicaragua                      0.110870
Iran                           0.106250
Ecuador                        0.097011
France                         0.078533
Greece                         0.073913
Laos                           0.069294
Ireland                        0.055435
Outlying-US(Guam-USVI-etc)     0.055435
Trinadad&Tobago                0.050815
Thailand                       0.050815
Scotland                       0.046196
Cambodia                       0.046196
Honduras                       0.046196
Yugoslavia                     0.036957
Hong                           0.032337
```

```
Hungary                         0.023098
Holand-Netherlands              0.004620
Name: native-country, dtype: float64
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

Data Visualization by count plot as univariate analysis

In [34]:
```
sns.countplot(x=df_copy_new['sex'])
```

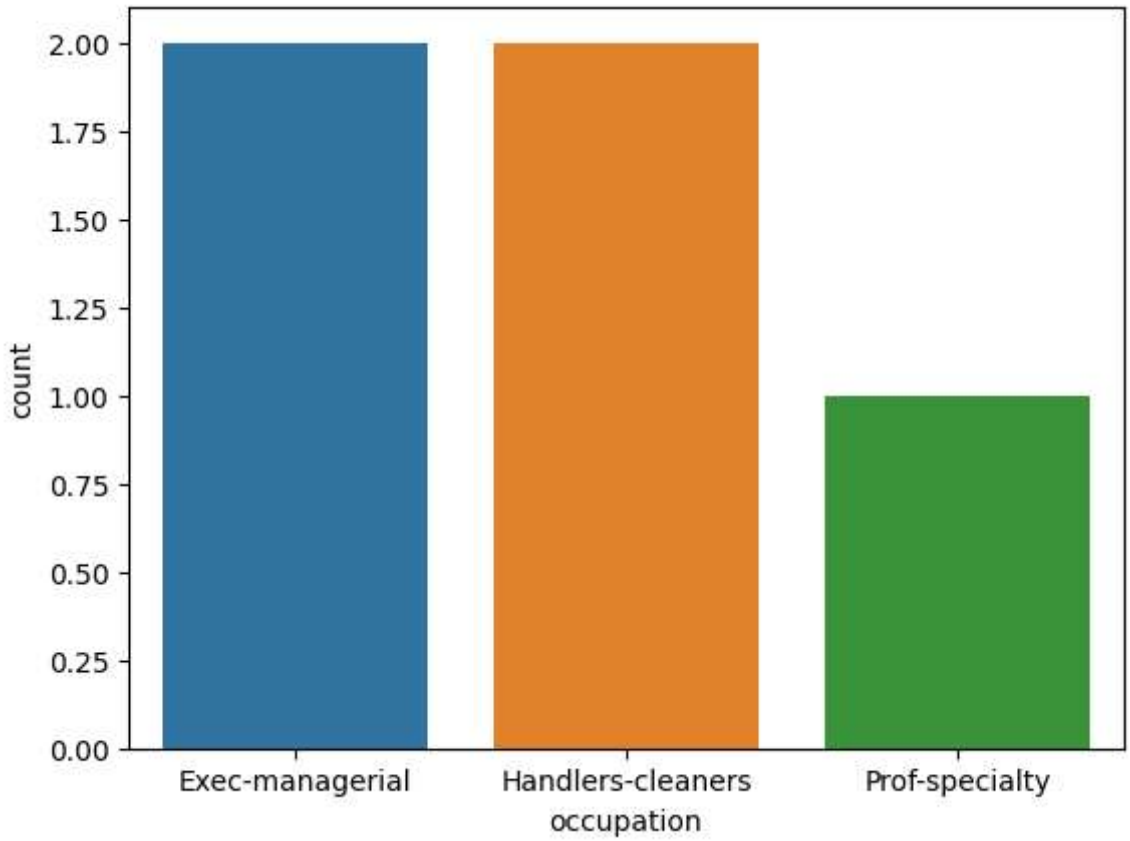
Out[34]: <AxesSubplot: xlabel='sex', ylabel='count'>



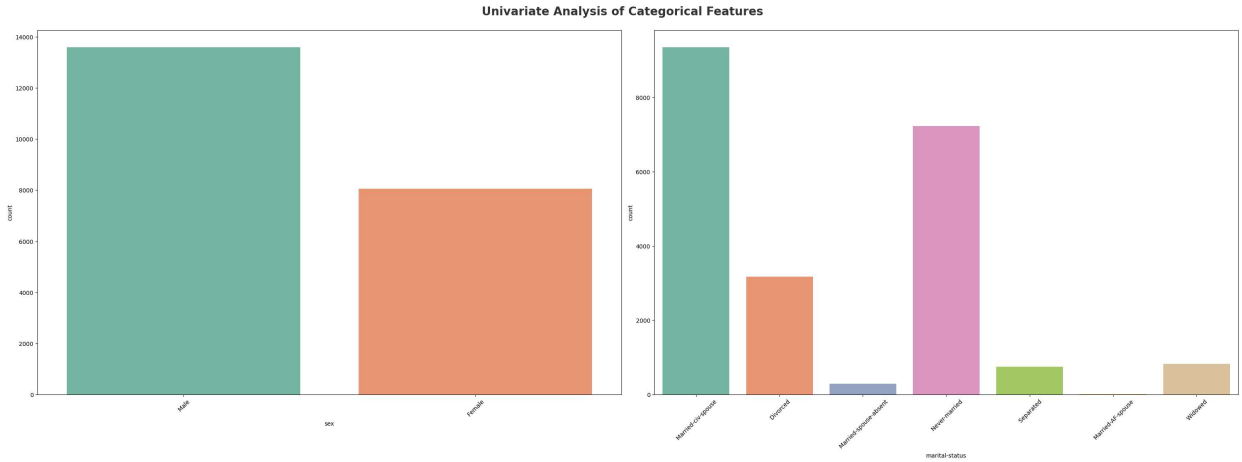
CONCLUSION---------MALE COUNT IS MORE THAN FEMALE COUNT

In [35]:
```
sns.countplot(x=df_copy_new.head()['occupation'])
```

Out[35]: <AxesSubplot: xlabel='occupation', ylabel='count'>



In [36]:
```
# categorical columns
plt.figure(figsize=(30, 20))
plt.suptitle('Univariate Analysis of Categorical Features', fontsize=20, fontweight='b
alpha=0.8, y=1.)
category = ['sex','marital-status']
for i in range(0, len(category)):
    plt.subplot(2, 2, i+1)
    sns.countplot(x=df_copy_new[category[i]],palette="Set2")
    plt.xlabel(category[i])
    plt.xticks(rotation=45)
    plt.tight_layout()
```



CONCLUSION----Married-civ-spouse have higher number in Marital-status column

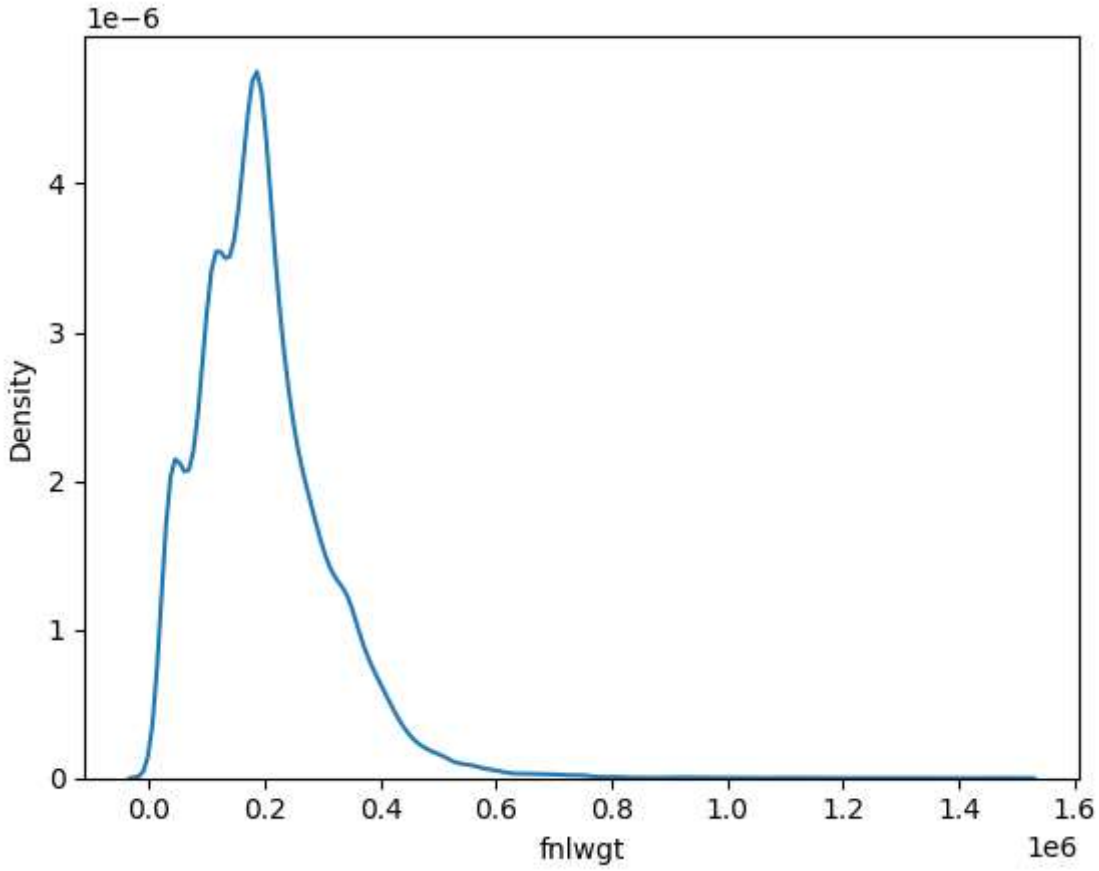
Analysis on Numeric Feature

In [37]:
```
numeric_features
```

Out[37]:
```
['Age',
 'fnlwgt',
 'education-num',
 'capital-gain',
 'capital-loss',
 'hours-per-week',
 'makes over']
```

In [38]:
```
sns.kdeplot(df_copy_new['fnlwgt'])
```

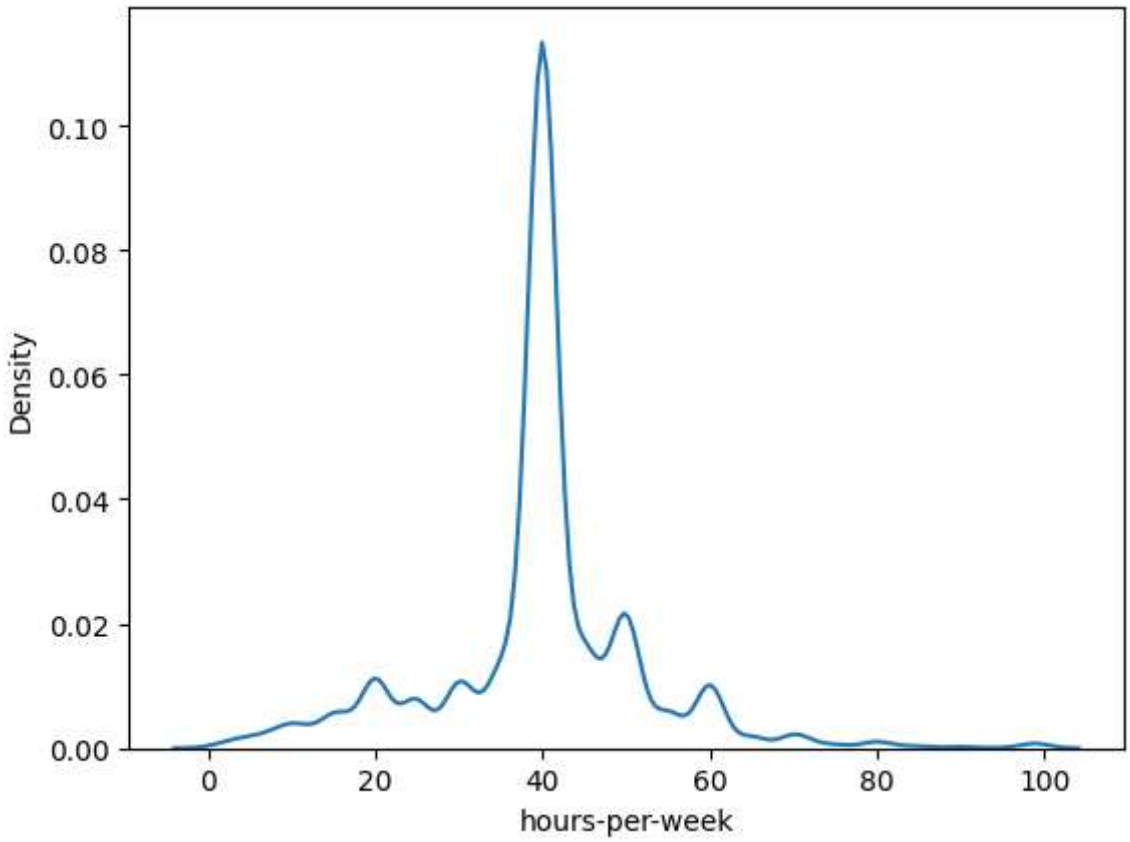
Out[38]:
```
<AxesSubplot: xlabel='fnlwgt', ylabel='Density'>
```



CONCLUSION------Distribution is Right Skewed

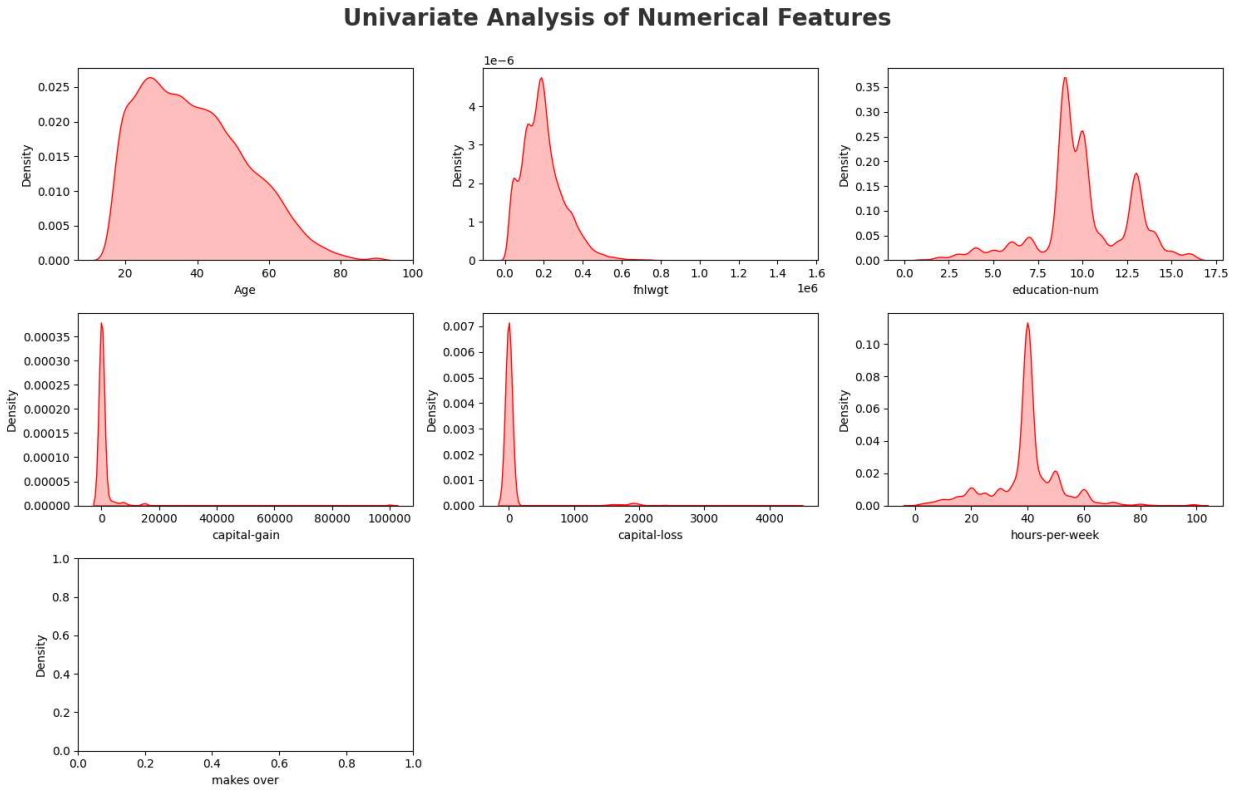
In [39]:
```
sns.kdeplot(df_copy_new['hours-per-week'])
```

Out[39]:
```
<AxesSubplot: xlabel='hours-per-week', ylabel='Density'>
```



Conclution:For Univariate if the variable is categorical do count plot and if variable is Numerical do distribution plot.

In [40]:
```
plt.figure(figsize=(15, 15))
plt.suptitle('Univariate Analysis of Numerical Features', fontsize=20, fontweight='bol
alpha=0.8, y=1.)
for i in range(0, len(numeric_features)):
    plt.subplot(5, 3, i+1)
    sns.kdeplot(x=df_copy_new[numeric_features[i]],shade=True, color='r')
    plt.xlabel(numeric_features[i])
    plt.tight_layout()
```



Univariate analysis-----Most popular category in 'marital-status' feature

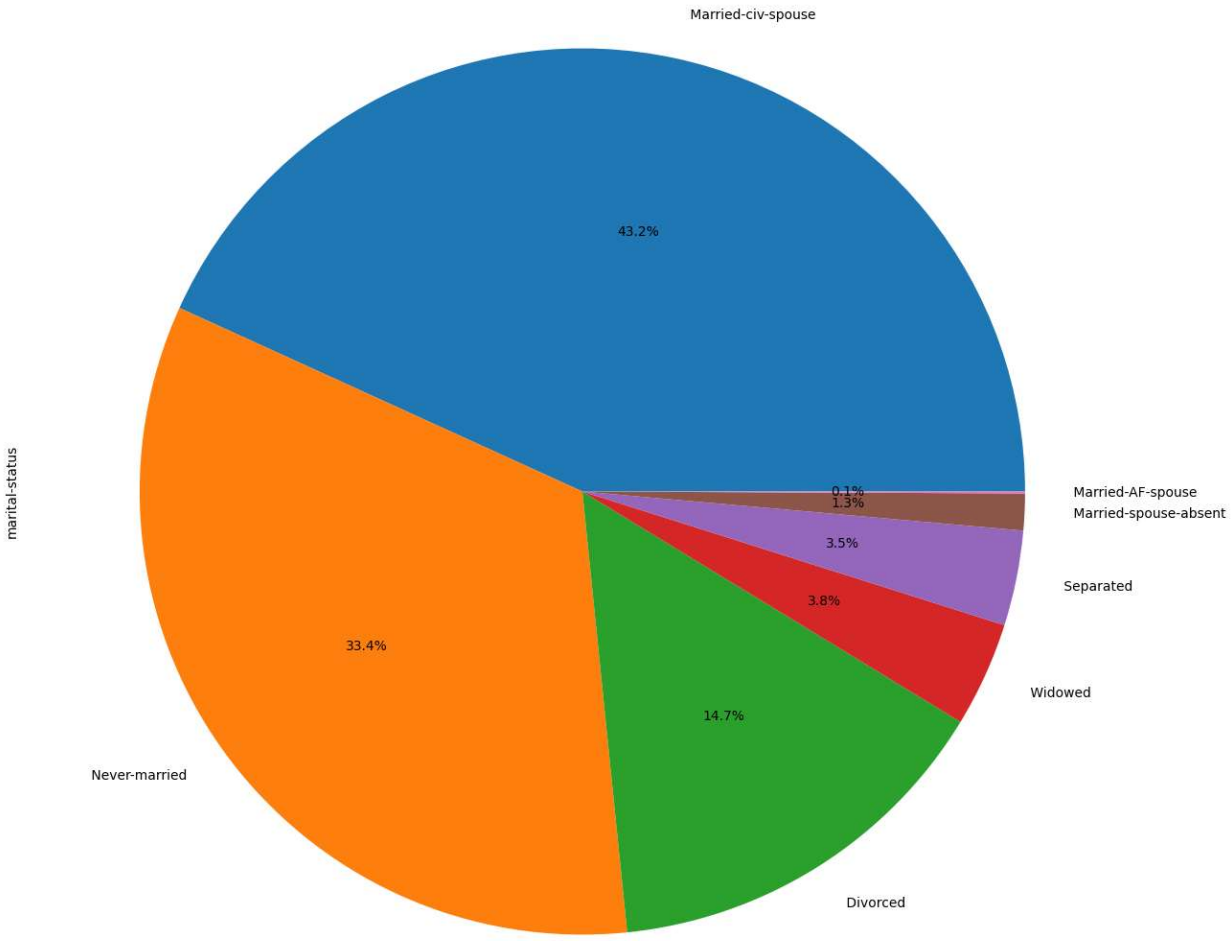
In [41]:
```
df_copy_new['marital-status'].value_counts()
```

Out[41]:
```
Married-civ-spouse       9352
Never-married            7233
Divorced                 3178
Widowed                   828
Separated                 751
Married-spouse-absent     290
Married-AF-spouse          15
Name: marital-status, dtype: int64
```

value count of 'marital-status' column in percentage

In [42]:
```
df_copy_new['marital-status'].value_counts().plot.pie(y=df_copy_new['marital-status'],
```

Out[42]:
```
<AxesSubplot: ylabel='marital-status'>
```



Conclusion-----Married-civ-spouse is most popular category as it is having 43.2% counts.

Top Ten Counts of Column 'marital-status' (Univariate Analysis)

In [43]:
```
df_copy_new.head()
```

Out[43]:

	Age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	sex
0	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male
1	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male
2	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male
3	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female
4	37	Private	284582	Masters	14	Married-civ-spouse	Exec-managerial	Wife	White	Female

In [44]:
```
df_copy_new.columns
```

Out[44]:
```
Index(['Age', 'workclass', 'fnlwgt', 'education', 'education-num',
       'marital-status', 'occupation', 'relationship', 'race', 'sex',
       'capital-gain', 'capital-loss', 'hours-per-week', 'native-country',
       'makes over'],
      dtype='object')
```

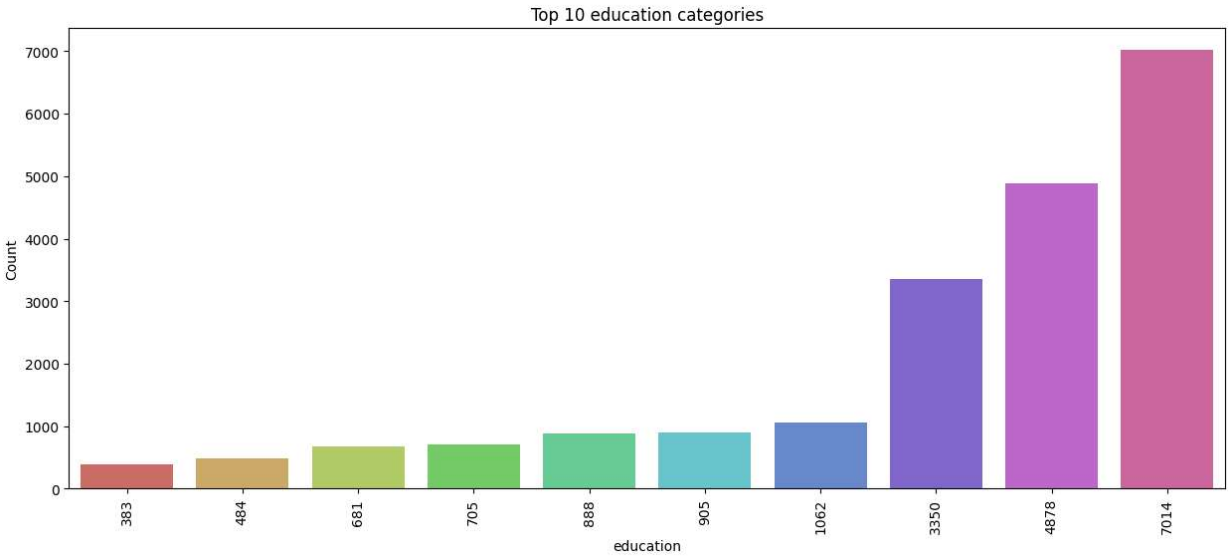
In [45]:
```
df_cat1=df_copy_new['education'].value_counts()[:10]
```

In [46]:
```
df_cat1
```

Out[46]:
```
 HS-grad         7014
 Some-college    4878
 Bachelors       3350
 Masters         1062
 Assoc-voc        905
 11th             888
 10th             705
 Assoc-acdm       681
 7th-8th          484
 9th              383
Name: education, dtype: int64
```

In [47]:
```
category1 = pd.DataFrame(df_copy_new['education'].value_counts())
category1.rename(columns = {'education':'Count'},inplace=True)
```

In [48]:
```
plt.figure(figsize=(15,6))
sns.barplot(x=df_cat1, y ='Count',data = category1[:10],palette='hls')
plt.title('Top 10 education categories')
plt.xticks(rotation=90)
plt.show()
```



CONCLUSION------EDUCATION OF MAXIMUM PEOPLE ARE HS-GRADE ONLY.

Bivariate Data Analysis(Taking Two Features 'education' and 'makes over' for analysing the data)

Which category in column'education' have highest number of 'makes over' members

In [49]:
```
df_cat_make=df_copy_new.groupby(['education' ])['makes over'].sum().sort_values(ascenc
```

In [50]:
```
df_cat_make
```

Out[50]:

	education	makes over
0	HS-grad	350700000
1	Some-college	243900000
2	Bachelors	167500000
3	Masters	53100000
4	Assoc-voc	45250000
5	11th	44400000
6	10th	35250000
7	Assoc-acdm	34050000
8	7th-8th	24200000
9	9th	19150000
10	Prof-school	17000000
11	12th	14700000
12	Doctorate	13200000
13	5th-6th	12050000
14	1st-4th	6250000
15	Preschool	1650000

In [51]:
```
df_cat1_make=df_cat_make.head(10)
```

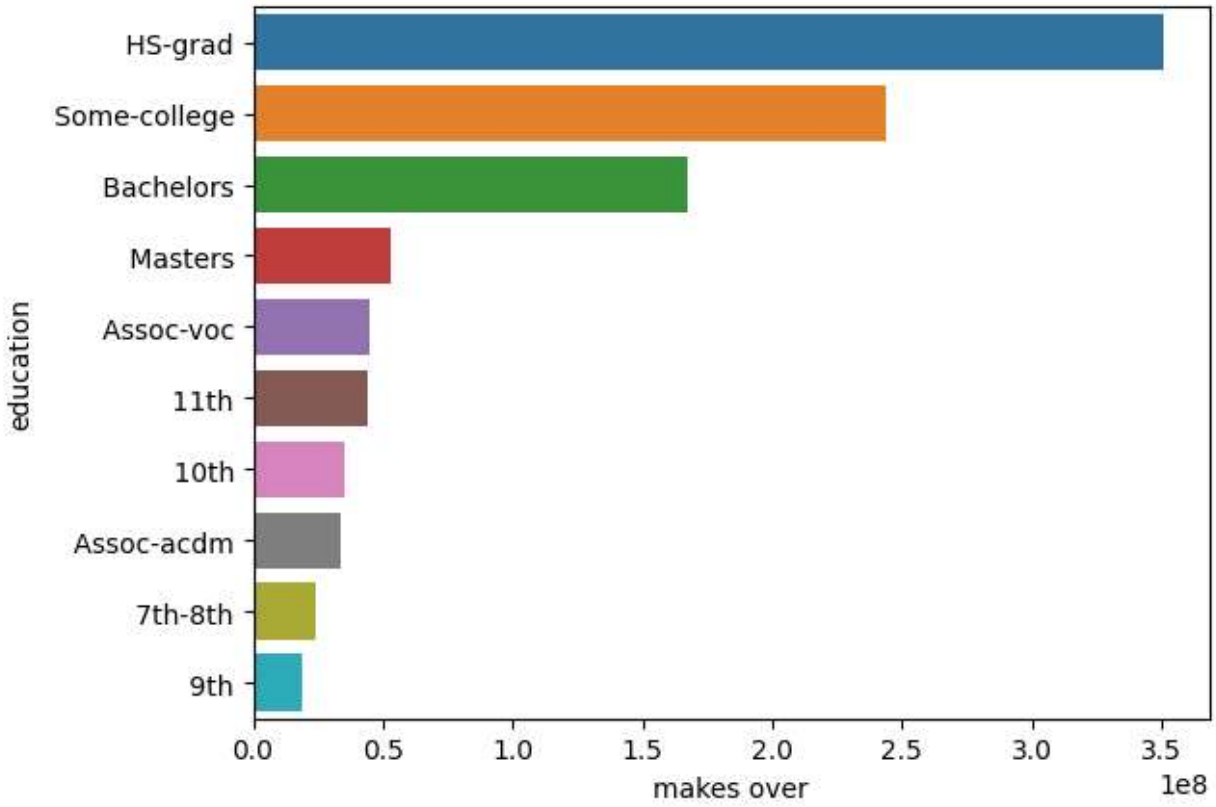
In [52]:
```
df_cat1_make
```

Out[52]:

	education	makes over
0	HS-grad	350700000
1	Some-college	243900000
2	Bachelors	167500000
3	Masters	53100000
4	Assoc-voc	45250000
5	11th	44400000
6	10th	35250000
7	Assoc-acdm	34050000
8	7th-8th	24200000
9	9th	19150000

In [53]:
```
sns.barplot(x='makes over',y='education',data=df_cat1_make)
```

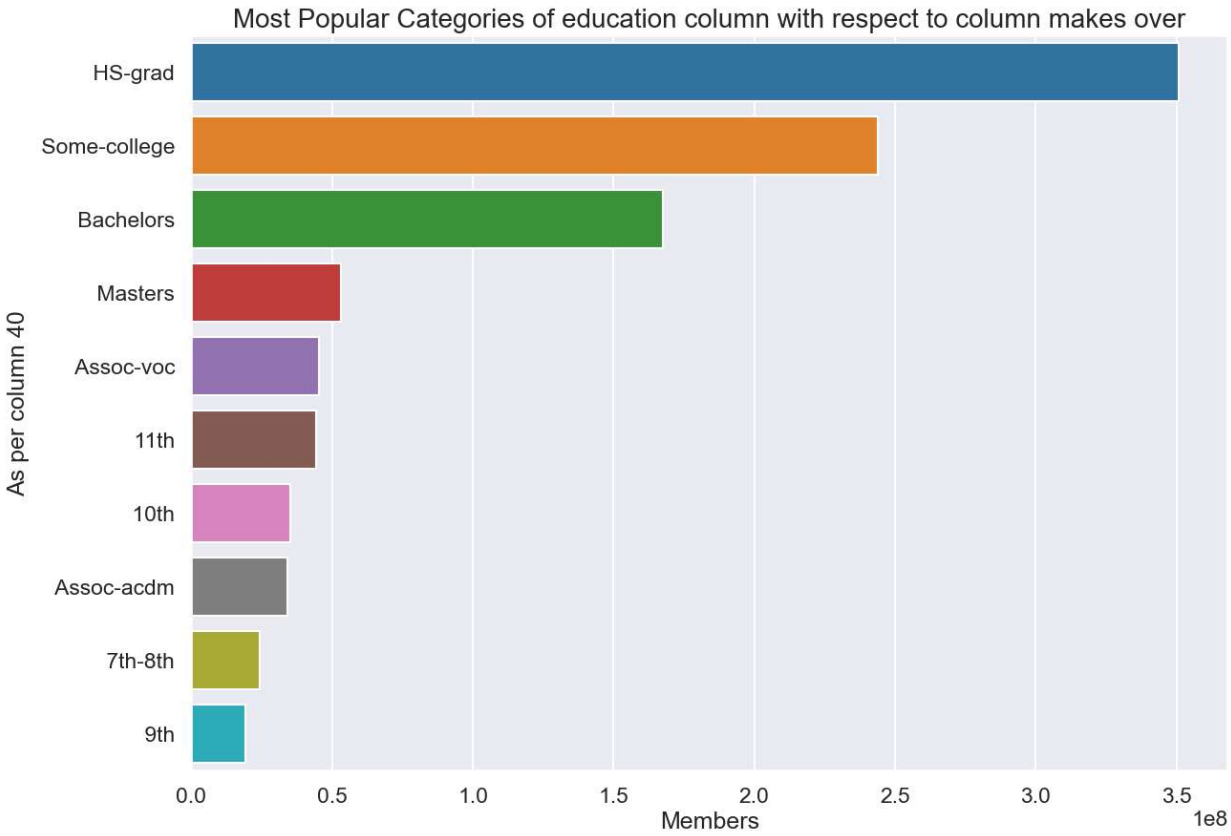
Out[53]:
```
<AxesSubplot: xlabel='makes over', ylabel='education'>
```



CONCLUSION---------People of HS-grad having higher makes over in a year.

In [54]:
```
plt.figure(figsize = (14,10))
sns.set_context("talk")
sns.set_style("darkgrid")
ax = sns.barplot(x = 'makes over' , y = 'education' , data = df_cat1_make )
ax.set_xlabel("Members")
ax.set_ylabel('As per column 40')
ax.set_title("Most Popular Categories of education column with respect to column makes
```

Out[54]:
```
Text(0.5, 1.0, 'Most Popular Categories of education column with respect to column ma
kes over')
```



Multivariate Analysis using more than 2 features 'education', 'occupation' and 'makes over'.

In [55]:
```
df_copy_new.head()
```

Out[55]:

	Age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	sex
0	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male
1	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male
2	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male
3	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female
4	37	Private	284582	Masters	14	Married-civ-spouse	Exec-managerial	Wife	White	Female

In [56]:
```
df_copy_new.columns
```

Out[56]:
```
Index(['Age', 'workclass', 'fnlwgt', 'education', 'education-num',
       'marital-status', 'occupation', 'relationship', 'race', 'sex',
       'capital-gain', 'capital-loss', 'hours-per-week', 'native-country',
       'makes over'],
      dtype='object')
```

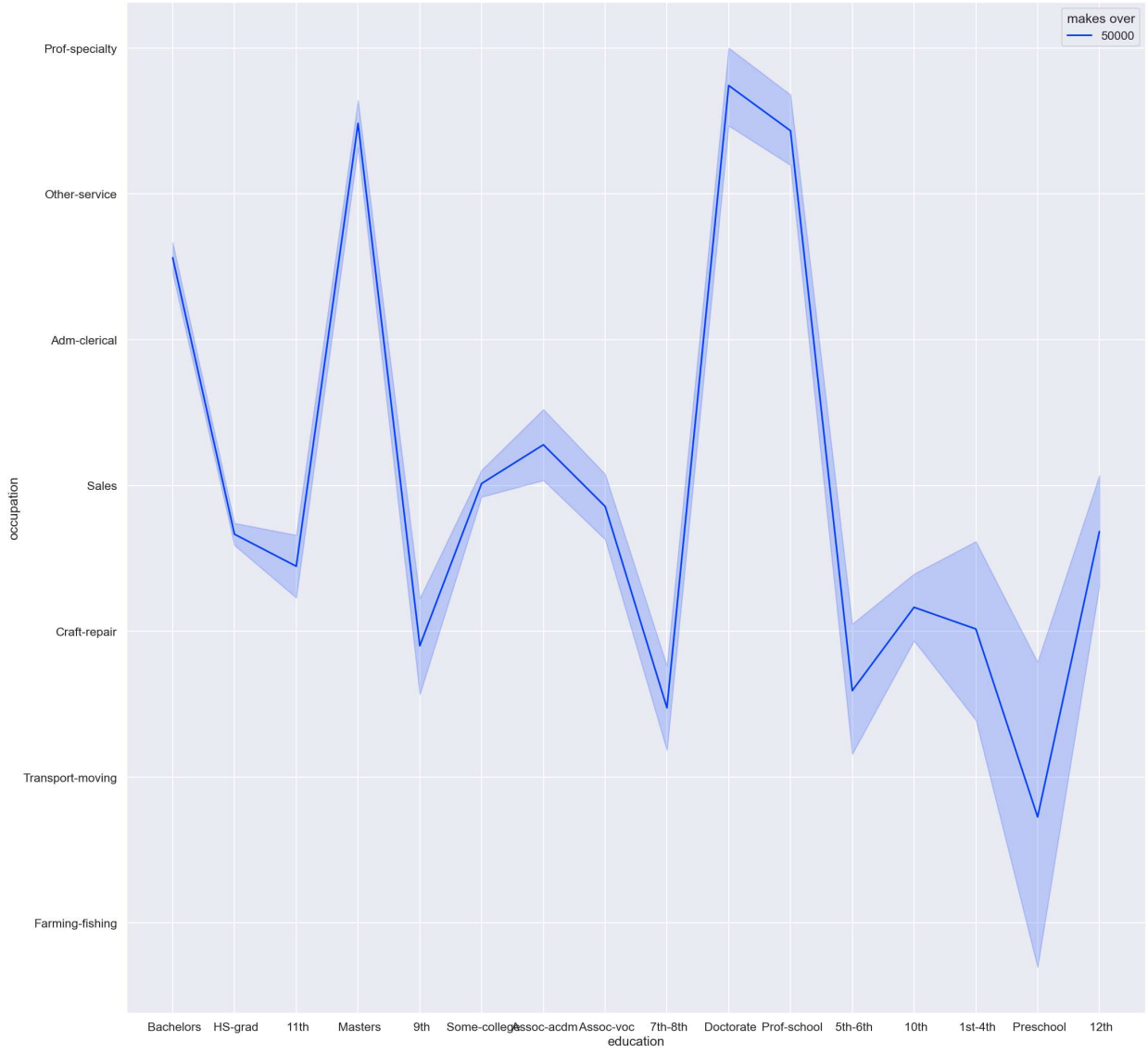
In [115…
```
df_copy_new.head()
```

Out[115]:

	Age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	sex
0	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male
1	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male
2	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male
3	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female
4	37	Private	284582	Masters	14	Married-civ-spouse	Exec-managerial	Wife	White	Female

In [121…
```
fig, ax = plt.subplots(figsize=(25, 25))
sns.lineplot(x='education', y='occupation',data=df_copy_new, palette='bright',hue='mak
```

Out[121]:
```
<AxesSubplot: xlabel='education', ylabel='occupation'>
```



Conclusion----------less educated people yearly make over is less.

In [137…
```
df_copy_new.head().groupby(['Age','education-num'])["sex"].sum().sort_values().reset_i
```

Out[137]:

	Age	education-num	sex
0	28	13	Female
1	37	14	Female
2	38	9	Male
3	50	13	Male
4	53	7	Male

Analysis based on sex with respect to Age and makeover respectively(Bivariate Analysis)

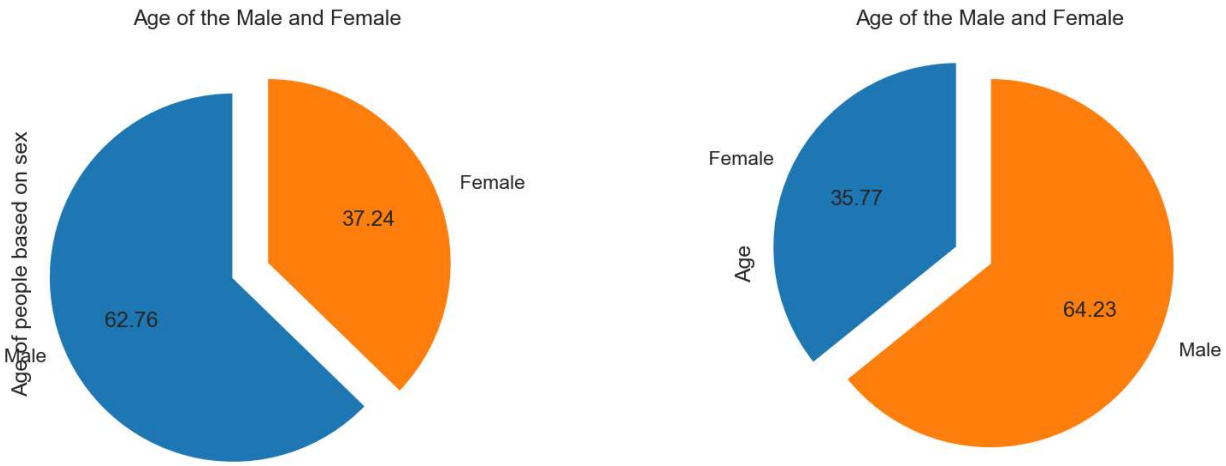
In [150…
```
df_copy_new.groupby('sex').agg({'Age':sum})
```

Out[150]:

sex	Age
Female	302475
Male	543054

In [151…
```
fig, ax = plt.subplots(1,2,figsize=(20,7))
df_copy_new.value_counts('sex').plot.pie(y='sex',startangle=90, explode=(0.2,0),
title='Age of the Male and Female', legend=False, autopct='%.2f',
ax=ax[0])
ax[0].set(ylabel='Age of people based on sex')
df_copy_new.groupby('sex').agg({'Age':sum}).plot.pie(y='Age', startangle=90,
explode=(0.2,0), title='Age of the Male and Female',
legend=False, autopct='%.2f', ax=ax[1])
```

Out[151]: <AxesSubplot: title={'center': 'Age of the Male and Female'}, ylabel='Age'>



Conclusion----Age of Male is Higher than Female

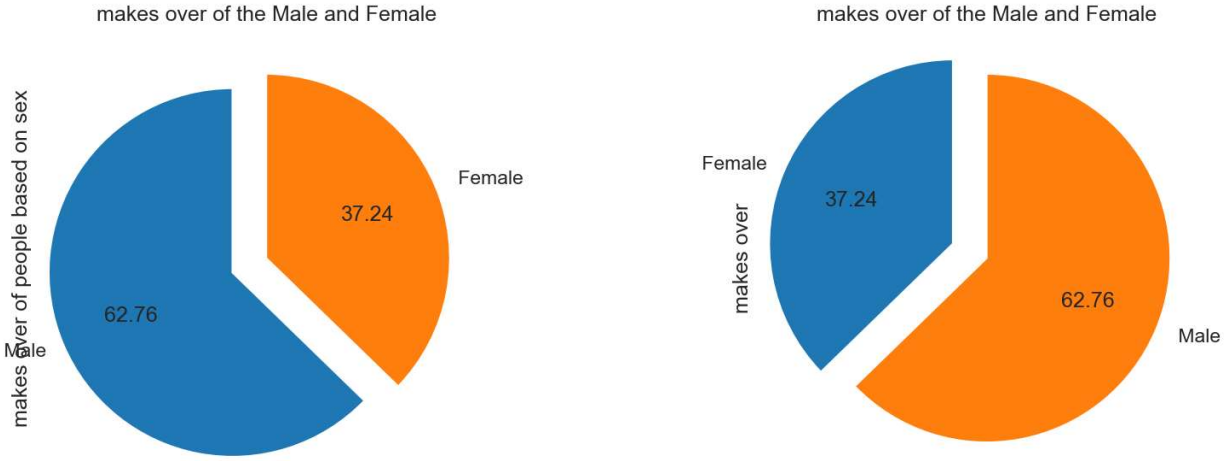
In [152…
```
df_copy_new.groupby('sex').agg({'makes over':sum})
```

Out[152]:

sex	makes over
Female	403100000
Male	679250000

In [153…
```
fig, ax = plt.subplots(1,2,figsize=(20,7))
df_copy_new.value_counts('sex').plot.pie(y='sex',startangle=90, explode=(0.2,0),
title='makes over of the Male and Female', legend=False, autopct='%.2f',
ax=ax[0])
ax[0].set(ylabel='makes over of people based on sex')
df_copy_new.groupby('sex').agg({'makes over':sum}).plot.pie(y='makes over', startangle
explode=(0.2,0), title='makes over of the Male and Female',
legend=False, autopct='%.2f', ax=ax[1])
```

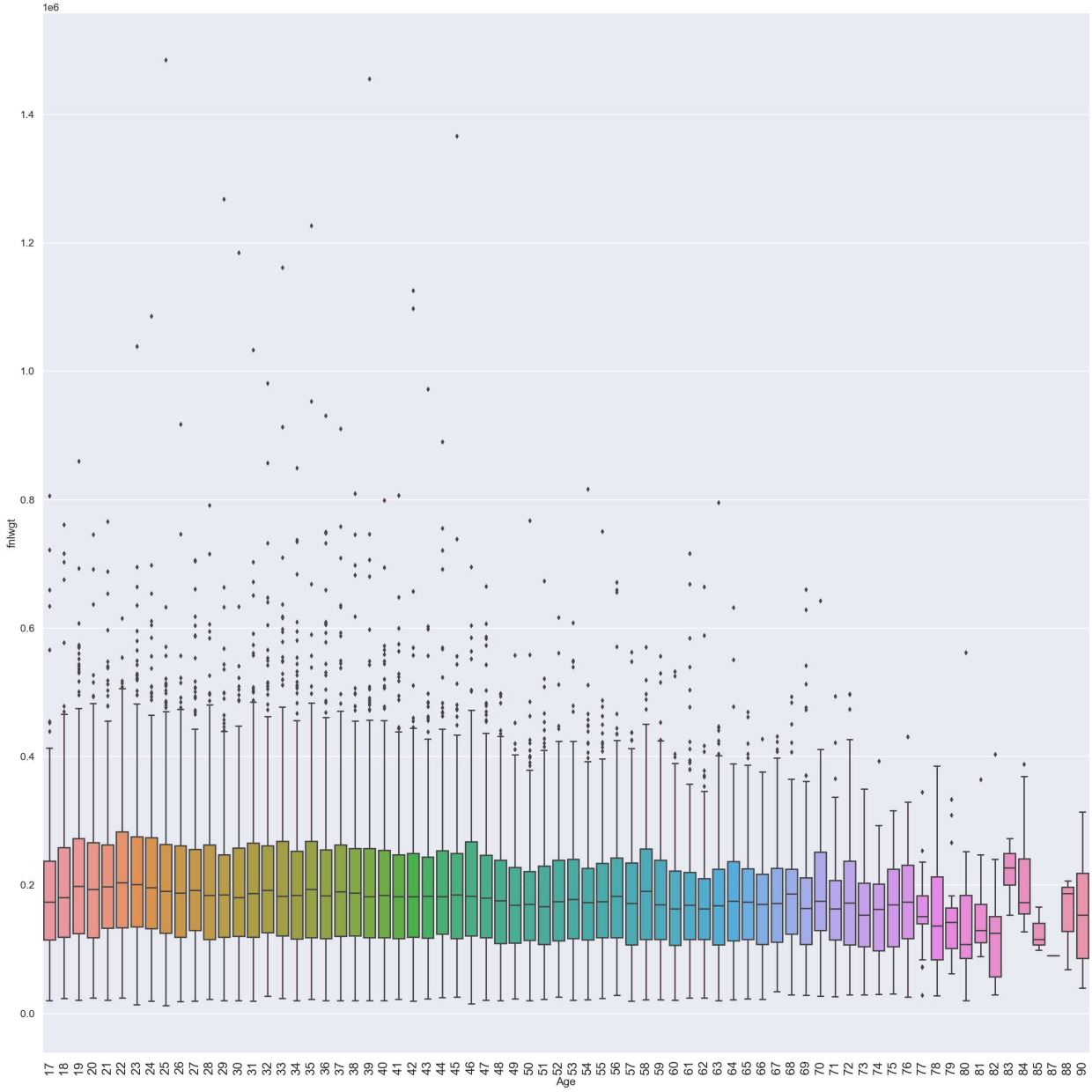
Out[153]: <AxesSubplot: title={'center': 'makes over of the Male and Female'}, ylabel='makes over'>



Conclusion-------Make over wise Male also have more values.

In [160…

```
plt.figure(figsize=(30,30))
sns.boxplot(x="Age",y="fnlwgt",data=df_copy_new)
plt.xticks(size=20,rotation=90)
plt.show()
```



Conclusion----Highest Age is 90 and Median is some around between 0.1 to 0.2

In [ ]: