In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
```

In [2]:
```
import warnings
warnings.filterwarnings('ignore')
```

In [4]:
```
df = pd.read_csv('census_income.csv')
df.head
```

Out[4]:
```
<bound method NDFrame.head of        Age          Workclass  Fnlwgt    Education  Education_num  \
0       50   Self-emp-not-inc   83311    Bachelors             13
1       38            Private  215646      HS-grad              9
2       53            Private  234721         11th              7
3       28            Private  338409    Bachelors             13
4       37            Private  284582      Masters             14
...    ...                ...     ...          ...            ...
32555   27            Private  257302   Assoc-acdm             12
32556   40            Private  154374      HS-grad              9
32557   58            Private  151910      HS-grad              9
32558   22            Private  201490      HS-grad              9
32559   52       Self-emp-inc  287927      HS-grad              9

            Marital_status          Occupation    Relationship    Race  \
0       Married-civ-spouse     Exec-managerial         Husband   White
1                 Divorced   Handlers-cleaners   Not-in-family   White
2       Married-civ-spouse   Handlers-cleaners         Husband   Black
3       Married-civ-spouse      Prof-specialty            Wife   Black
4       Married-civ-spouse     Exec-managerial            Wife   White
...                    ...                 ...             ...     ...
32555   Married-civ-spouse        Tech-support            Wife   White
32556   Married-civ-spouse   Machine-op-inspct         Husband   White
32557              Widowed        Adm-clerical       Unmarried   White
32558        Never-married        Adm-clerical       Own-child   White
32559   Married-civ-spouse     Exec-managerial            Wife   White

           Sex  Capital_gain  Capital_loss  Hours_per_week  Native_country  \
0         Male             0             0              13   United-States
1         Male             0             0              40   United-States
2         Male             0             0              40   United-States
3       Female             0             0              40            Cuba
4       Female             0             0              40   United-States
...        ...           ...           ...             ...             ...
32555   Female             0             0              38   United-States
32556     Male             0             0              40   United-States
32557   Female             0             0              40   United-States
32558     Male             0             0              20   United-States
32559   Female         15024             0              40   United-States

       Income
0       <=50K
1       <=50K
2       <=50K
3       <=50K
4       <=50K
...       ...
32555   <=50K
32556    >50K
32557   <=50K
32558   <=50K
32559    >50K

[32560 rows x 15 columns]>
```

In [5]:
```
df.shape
```

Out[5]:
```
(32560, 15)
```

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

Out[6]:
```
Age                int64
Workclass         object
Fnlwgt             int64
Education         object
Education_num      int64
Marital_status    object
Occupation        object
Relationship      object
Race              object
Sex               object
Capital_gain       int64
Capital_loss       int64
Hours_per_week     int64
Native_country    object
Income            object
dtype: object
```

In [7]:
```
df.isnull().sum()
```

Out[7]:
```
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 [8]:
```
df.nunique()
```

Out[8]:
```
Age                  73
Workclass             9
Fnlwgt            21647
Education            16
Education_num        16
Marital_status        7
Occupation           15
Relationship          6
Race                  5
Sex                   2
Capital_gain        119
Capital_loss         92
Hours_per_week       94
Native_country       42
Income                2
dtype: int64
```

In [9]:
```
df.describe().T
```

Out[9]:

| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| **Age** | 32560.0 | 38.581634 | 13.640642 | 17.0 | 28.0 | 37.0 | 48.0 | 90.0 |
| **Fnlwgt** | 32560.0 | 189781.814373 | 105549.764924 | 12285.0 | 117831.5 | 178363.0 | 237054.5 | 1484705.0 |
| **Education_num** | 32560.0 | 10.080590 | 2.572709 | 1.0 | 9.0 | 10.0 | 12.0 | 16.0 |
| **Capital_gain** | 32560.0 | 1077.615172 | 7385.402999 | 0.0 | 0.0 | 0.0 | 0.0 | 99999.0 |
| **Capital_loss** | 32560.0 | 87.306511 | 402.966116 | 0.0 | 0.0 | 0.0 | 0.0 | 4356.0 |
| **Hours_per_week** | 32560.0 | 40.437469 | 12.347618 | 1.0 | 40.0 | 40.0 | 45.0 | 99.0 |

In [21]:
```
df.columns
```

Out[21]:
```
Index(['Age', 'Workclass', 'Fnlwgt', 'Education', 'Education_num',
       'Marital_status', 'Occupation', 'Relationship', 'Race', 'Sex',
       'Capital_gain', 'Capital_loss', 'Hours_per_week', 'Native_country',
       'Income'],
      dtype='object')
```

In [22]:
```
df['Sex'].value_counts()
```

Out[22]:
```
 Male      21789
 Female    10771
Name: Sex, dtype: int64
```

In [23]:
```
df['Native_country'].value_counts()
```

Out[23]:
```
 United-States                 29169
 Mexico                          643
 ?                               583
 Philippines                     198
 Germany                         137
 Canada                          121
 Puerto-Rico                     114
 El-Salvador                     106
 India                           100
 Cuba                             95
 England                          90
 Jamaica                          81
 South                            80
 China                            75
 Italy                            73
 Dominican-Republic               70
 Vietnam                          67
 Guatemala                        64
 Japan                            62
 Poland                           60
 Columbia                         59
 Taiwan                           51
 Haiti                            44
 Iran                             43
 Portugal                         37
 Nicaragua                        34
 Peru                             31
 France                           29
 Greece                           29
 Ecuador                          28
 Ireland                          24
 Hong                             20
 Cambodia                         19
 Trinadad&Tobago                  19
 Laos                             18
 Thailand                         18
 Yugoslavia                       16
 Outlying-US(Guam-USVI-etc)       14
 Honduras                         13
 Hungary                          13
 Scotland                         12
 Holand-Netherlands                1
Name: Native_country, dtype: int64
```

In [12]:
```
df['Workclass'].value_counts()
```

Out[12]:
```
 Private             22696
 Self-emp-not-inc     2541
 Local-gov            2093
 ?                    1836
 State-gov            1297
 Self-emp-inc         1116
 Federal-gov           960
 Without-pay            14
 Never-worked            7
Name: Workclass, dtype: int64
```

In [18]:
```
df['Race'].value_counts()
```

Out[18]:
```
 White                 27815
 Black                  3124
 Asian-Pac-Islander     1039
 Amer-Indian-Eskimo      311
 Other                   271
Name: Race, dtype: int64
```

In [20]:
```
df['Marital_status'].value_counts()
```

Out[20]:
```
 Married-civ-spouse       14976
 Never-married            10682
 Divorced                  4443
 Separated                 1025
 Widowed                    993
 Married-spouse-absent      418
 Married-AF-spouse           23
Name: Marital_status, dtype: int64
```

In [24]:
```
df['Occupation'].value_counts()
```

Out[24]:
```
 Prof-specialty       4140
 Craft-repair         4099
 Exec-managerial      4066
 Adm-clerical         3769
 Sales                3650
 Other-service        3295
 Machine-op-inspct    2002
 ?                    1843
 Transport-moving     1597
 Handlers-cleaners    1370
 Farming-fishing       994
 Tech-support          928
 Protective-serv       649
 Priv-house-serv       149
 Armed-Forces            9
Name: Occupation, dtype: int64
```

In [29]:
```
df['Income'].value_counts()
```

Out[29]:
```
 <=50K    24719
 >50K      7841
Name: Income, dtype: int64
```

In [31]:
```
df['Education'].value_counts()
```

Out[31]:
```
 HS-grad         10501
 Some-college     7291
 Bachelors        5354
 Masters          1723
 Assoc-voc        1382
 11th             1175
 Assoc-acdm       1067
 10th              933
 7th-8th           646
 Prof-school       576
 9th               514
 12th              433
 Doctorate         413
 5th-6th           333
 1st-4th           168
 Preschool          51
Name: Education, dtype: int64
```

In [33]:
```
df['Workclass'] = df['Workclass'].replace('?', 'Private')
df['Occupation'] = df['Occupation'].replace('?', 'Prof-specialty')
df['Native_country'] = df['Native_country'].replace('?', 'United-States')
```

In [40]:
```
df.head()
```

Out[40]:

| | Age | Workclass | Fnlwgt | Education | Education_num | Marital_status | Occupation | Relationship | Race | Sex | Capital_gain | Capital_loss | Hours_per_week | Native_ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 50 | Self-emp-not-inc | 83311 | Bachelors | 13 | Married-civ-spouse | Exec-managerial | Husband | White | Male | 0 | 0 | 13 | Unite |
| **1** | 38 | Private | 215646 | HS-grad | 9 | Divorced | Handlers-cleaners | Not-in-family | White | Male | 0 | 0 | 40 | Unite |
| **2** | 53 | Private | 234721 | 11th | 7 | Married-civ-spouse | Handlers-cleaners | Husband | Black | Male | 0 | 0 | 40 | Unite |
| **3** | 28 | Private | 338409 | Bachelors | 13 | Married-civ-spouse | Prof-specialty | Wife | Black | Female | 0 | 0 | 40 | |
| **4** | 37 | Private | 284582 | Masters | 14 | Married-civ-spouse | Exec-managerial | Wife | White | Female | 0 | 0 | 40 | Unite |

In [35]:
```
df.Education= df.Education.replace(['Preschool', '1st-4th', '5th-6th', '7th-8th', '9th','10th', '11th', '12th'], 'school')
df.Education = df.Education.replace('HS-grad', 'high school')
df.Education = df.Education.replace(['Assoc-voc', 'Assoc-acdm', 'Prof-school', 'Some-college'], 'higher')
df.Education = df.Education.replace('Bachelors', 'undergrad')
df.Education = df.Education.replace('Masters', 'grad')
df.Education= df.Education.replace('Doctorate', 'doc')
```

In [36]:
```
df['Marital_status']= df['Marital_status'].replace(['Married-civ-spouse', 'Married-AF-spouse'], 'married')
df['Marital_status']= df['Marital_status'].replace(['Never-married'], 'not-married')
df['Marital_status']= df['Marital_status'].replace(['Divorced', 'Separated','Widowed',
                                                   'Married-spouse-absent'], 'other')
```

In [37]:
```
df.Income = df.Income.replace('<=50K', 0)
df.Income = df.Income.replace('>50K', 1)
```

In [39]:
```
df.head()
```

Out[39]:

| | Age | Workclass | Fnlwgt | Education | Education_num | Marital_status | Occupation | Relationship | Race | Sex | Capital_gain | Capital_loss | Hours_per_week | Native_ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 50 | Self-emp-not-inc | 83311 | Bachelors | 13 | Married-civ-spouse | Exec-managerial | Husband | White | Male | 0 | 0 | 13 | Unite |
| **1** | 38 | Private | 215646 | HS-grad | 9 | Divorced | Handlers-cleaners | Not-in-family | White | Male | 0 | 0 | 40 | Unite |
| **2** | 53 | Private | 234721 | 11th | 7 | Married-civ-spouse | Handlers-cleaners | Husband | Black | Male | 0 | 0 | 40 | Unite |
| **3** | 28 | Private | 338409 | Bachelors | 13 | Married-civ-spouse | Prof-specialty | Wife | Black | Female | 0 | 0 | 40 | |
| **4** | 37 | Private | 284582 | Masters | 14 | Married-civ-spouse | Exec-managerial | Wife | White | Female | 0 | 0 | 40 | Unite |

In [42]:
```python
df['Marital_status'].value_counts()
```

Out[42]:
```
 Married-civ-spouse       14976
 Never-married            10682
 Divorced                  4443
 Separated                 1025
 Widowed                    993
 Married-spouse-absent      418
 Married-AF-spouse           23
Name: Marital_status, dtype: int64
```

In [43]:
```python
df['Education'].value_counts()
```

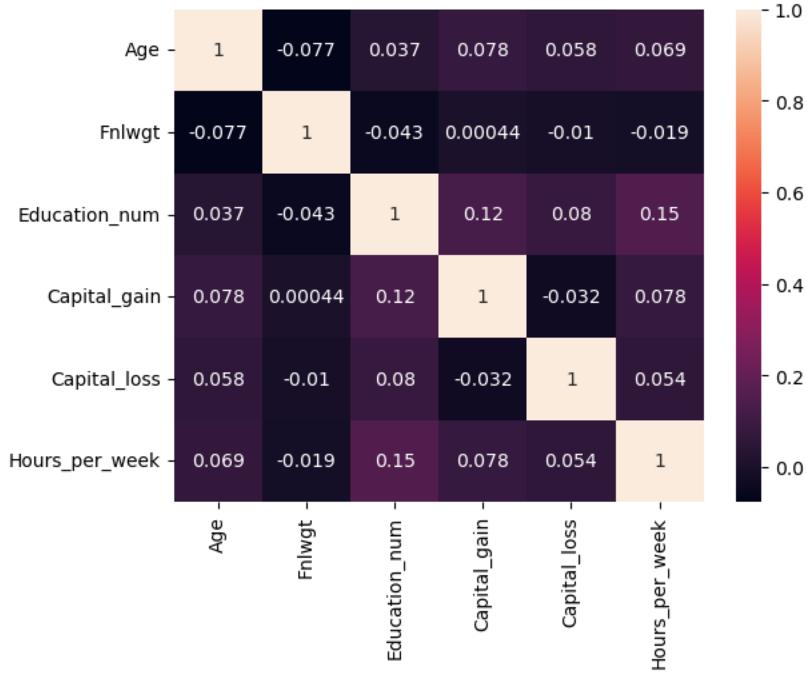
Out[43]:
```
 HS-grad         10501
 Some-college     7291
 Bachelors        5354
 Masters          1723
 Assoc-voc        1382
 11th             1175
 Assoc-acdm       1067
 10th              933
 7th-8th           646
 Prof-school       576
 9th               514
 12th              433
 Doctorate         413
 5th-6th           333
 1st-4th           168
 Preschool          51
Name: Education, dtype: int64
```

In [44]:
```python
df.corr()
```

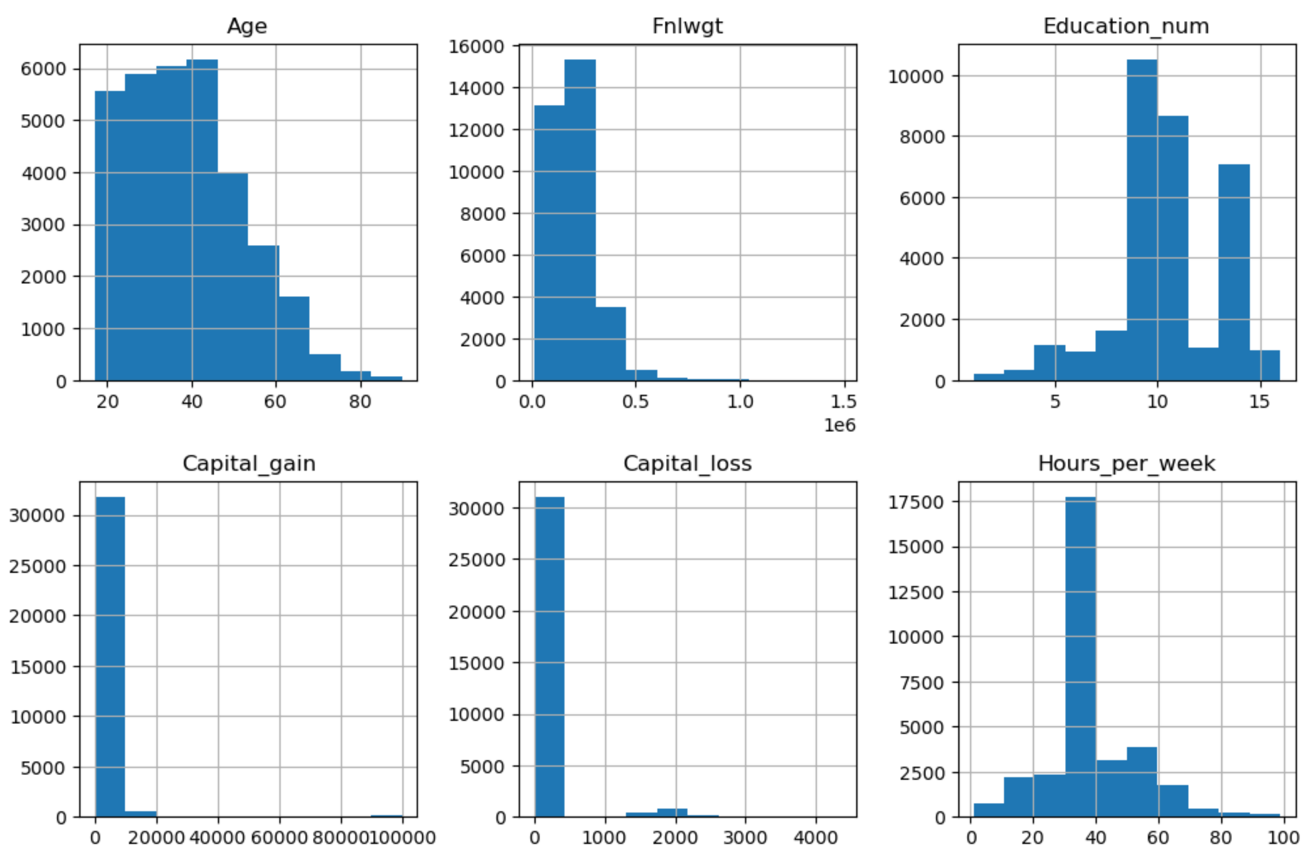
Out[44]:

| | Age | Fnlwgt | Education_num | Capital_gain | Capital_loss | Hours_per_week |
|---|---|---|---|---|---|---|
| **Age** | 1.000000 | -0.076646 | 0.036527 | 0.077674 | 0.057775 | 0.068756 |
| **Fnlwgt** | -0.076646 | 1.000000 | -0.043159 | 0.000437 | -0.010259 | -0.018770 |
| **Education_num** | 0.036527 | -0.043159 | 1.000000 | 0.122627 | 0.079932 | 0.148127 |
| **Capital_gain** | 0.077674 | 0.000437 | 0.122627 | 1.000000 | -0.031614 | 0.078409 |
| **Capital_loss** | 0.057775 | -0.010259 | 0.079932 | -0.031614 | 1.000000 | 0.054256 |
| **Hours_per_week** | 0.068756 | -0.018770 | 0.148127 | 0.078409 | 0.054256 | 1.000000 |

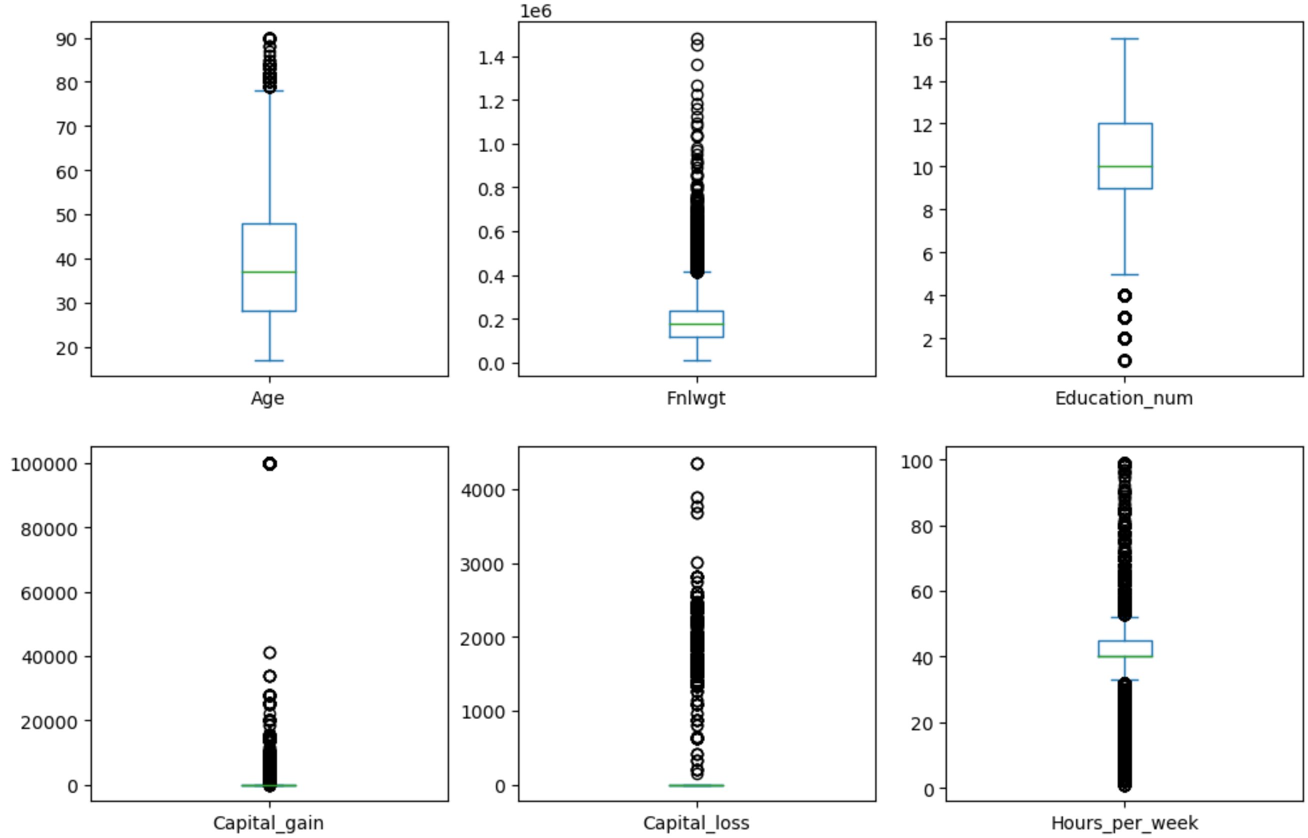
In [45]:
```python
sns.heatmap(df.corr(), annot=True);
```



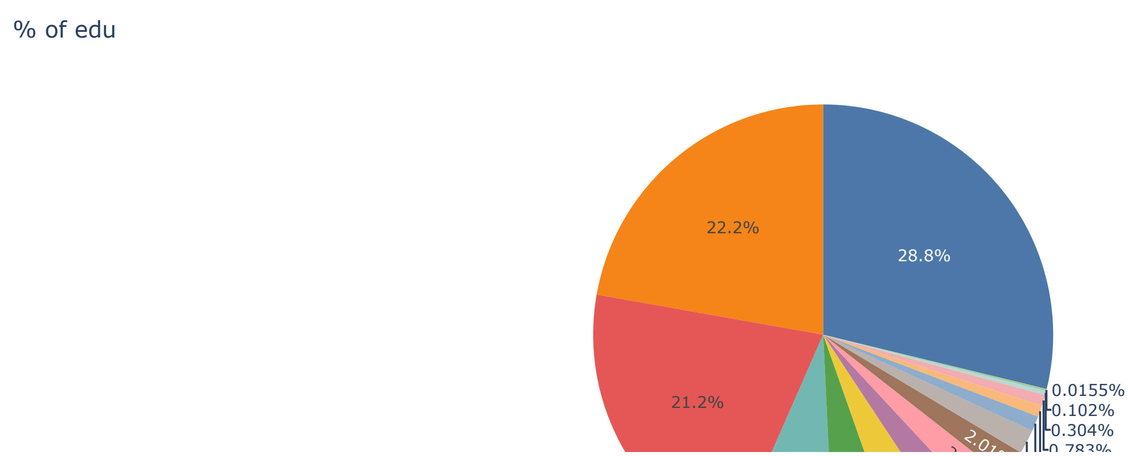

In [46]:
```
df.hist(figsize=(12,12), layout=(3,3), sharex=False);
```




In [47]:
```
df.plot(kind='box', figsize=(12,12), layout=(3,3), sharex=False, subplots=True);
```




In [51]:
```
px.pie(df, values='Education_num', names='Education', title='% of edu',
      color_discrete_sequence = px.colors.qualitative.T10)
```




In [53]:
```
X= df.drop(['Income'], axis=1)
y = df['Income']
```

In [54]:
```
from sklearn.preprocessing import StandardScaler, LabelEncoder
```

In [55]:
```
df1= df.copy()
df1= df1.apply(LabelEncoder().fit_transform)
df1.head()
```

Out[55]:

| | Age | Workclass | Fnlwgt | Education | Education_num | Marital_status | Occupation | Relationship | Race | Sex | Capital_gain | Capital_loss | Hours_per_week | Native_cou |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 33 | 6 | 2925 | 9 | 12 | 2 | 4 | 0 | 4 | 1 | 0 | 0 | 12 | |
| **1** | 21 | 4 | 14085 | 11 | 8 | 0 | 6 | 1 | 4 | 1 | 0 | 0 | 39 | |
| **2** | 36 | 4 | 15335 | 1 | 6 | 2 | 6 | 0 | 2 | 1 | 0 | 0 | 39 | |
| **3** | 11 | 4 | 19354 | 9 | 12 | 2 | 10 | 5 | 2 | 0 | 0 | 0 | 39 | |
| **4** | 20 | 4 | 17699 | 12 | 13 | 2 | 4 | 5 | 4 | 0 | 0 | 0 | 39 | |

In [57]:
```
ss= StandardScaler().fit(df1.drop('Income', axis=1))
```

In [58]:
```
X= ss.transform(df1.drop('Income', axis=1))
y= df['Income']
```

In [59]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=40)
```

In [60]:
```
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score

lr = LogisticRegression()

model = lr.fit(X_train, y_train)
prediction = model.predict(X_test)

print("Acc on training data: {:,.3f}".format(lr.score(X_train, y_train)))
print("Acc on test data: {:,.3f}".format(lr.score(X_test, y_test)))
```

```
Acc on training data: 0.824
Acc on test data: 0.825
```

In [61]:
```
from sklearn.ensemble import RandomForestClassifier

rfc = RandomForestClassifier()

model1 = rfc.fit(X_train, y_train)
prediction1 = model1.predict(X_test)

print("Acc on training data: {:,.3f}".format(rfc.score(X_train, y_train)))
print("Acc on test data: {:,.3f}".format(rfc.score(X_test, y_test)))
```

```
Acc on training data: 1.000
Acc on test data: 0.862
```

In [62]:
```
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report
```

In [63]:
```
print(confusion_matrix(y_test, prediction1))
```

```
[[6935  477]
 [ 875 1481]]
```

In [64]:
```
print(classification_report(y_test, prediction1))
```

```
              precision    recall  f1-score   support

       <=50K       0.89      0.94      0.91      7412
        >50K       0.76      0.63      0.69      2356

    accuracy                           0.86      9768
   macro avg       0.82      0.78      0.80      9768
weighted avg       0.86      0.86      0.86      9768
```

In [69]:
```
print('Precision =' , 6935/(6935+875))
```

```
Precision = 0.8879641485275288
```

In [70]:
```
print('Recall =' , 6935/(6935+477))
```

```
Recall = 0.9356449001618996
```

In [71]:
```
print('Precision = ', 1481/(1481+477))
```

```
Precision =  0.7563840653728294
```

In [72]:
```
print('Recall= ', 1481/(1481+875))
```

```
Recall=  0.6286078098471987
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: