Working with Adult Dataset:-

In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

Data Cleaning:-

In [2]:

```
# Reading the dataset:
adult = pd.read_csv(r"C:\Users\lab25\Downloads\adult\adult.data")
adult
```

Out[2]:

	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male
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
...	...	...	...	...	...	...	...	...	...	..
32555	27	Private	257302	Assoc-acdm	12	Married-civ-spouse	Tech-support	Wife	White	Female
32556	40	Private	154374	HS-grad	9	Married-civ-spouse	Machine-op-inspct	Husband	White	Male
32557	58	Private	151910	HS-grad	9	Widowed	Adm-clerical	Unmarried	White	Female
32558	22	Private	201490	HS-grad	9	Never-married	Adm-clerical	Own-child	White	Male
32559	52	Self-emp-inc	287927	HS-grad	9	Married-civ-spouse	Exec-managerial	Wife	White	Female

32560 rows × 15 columns

As the dataset doesnot contain any column's name, therefore we need to use header=None.

In [3]:

```
adult = pd.read_csv(r"C:\Users\lab25\Downloads\adult\adult.data",header=None)
adult
```

Out[3]:

	0	1	2	3	4	5	6	7	8	9
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female
...	...	...	...	...	...	...	...	...	...	...
32556	27	Private	257302	Assoc-acdm	12	Married-civ-spouse	Tech-support	Wife	White	Female
32557	40	Private	154374	HS-grad	9	Married-civ-spouse	Machine-op-inspct	Husband	White	Male
32558	58	Private	151910	HS-grad	9	Widowed	Adm-clerical	Unmarried	White	Female
32559	22	Private	201490	HS-grad	9	Never-married	Adm-clerical	Own-child	White	Male
32560	52	Self-emp-inc	287927	HS-grad	9	Married-civ-spouse	Exec-managerial	Wife	White	Female

32561 rows × 15 columns

Now, we need to give the column's name inorder to work with them.

In [4]:
```
adult.columns = ['age','workclass','fnlwgt',
                  'education','education_num',
                 'marital_status','occupation','relationship',
                 'race','sex','capital_gain','capital_loss',
                 'hours_per_week','native_country','income']
```

In [5]:
```
adult
```

Out[5]:

	age	workclass	fnlwgt	education	education_num	marital_status	occupation	rela
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	
...	...	...	...	...	...	...	...	
32556	27	Private	257302	Assoc-acdm	12	Married-civ-spouse	Tech-support	
32557	40	Private	154374	HS-grad	9	Married-civ-spouse	Machine-op-inspct	
32558	58	Private	151910	HS-grad	9	Widowed	Adm-clerical	U
32559	22	Private	201490	HS-grad	9	Never-married	Adm-clerical	C
32560	52	Self-emp-inc	287927	HS-grad	9	Married-civ-spouse	Exec-managerial	

32561 rows × 15 columns

In [6]:

```python
# checking null values:
adult.isnull().sum()
```

Out[6]:

```
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]:

```
# check the datatypes of each columns:
adult.dtypes
```

Out[8]:

```
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 [9]:

```
# checking unique values:
for i in adult.columns:
    print(f"{i}:\n {adult[i].unique()}\n")
```

```
age:
 [39 50 38 53 28 37 49 52 31 42 30 23 32 40 34 25 43 54 35 59 56 19 20 45
 22 48 21 24 57 44 41 29 18 47 46 36 79 27 67 33 76 17 55 61 70 64 71 68
 66 51 58 26 60 90 75 65 77 62 63 80 72 74 69 73 81 78 88 82 83 84 85 86
 87]

workclass:
 [' State-gov' ' Self-emp-not-inc' ' Private' ' Federal-gov' ' Local-gov'
 ' ?' ' Self-emp-inc' ' Without-pay' ' Never-worked']

fnlwgt:
 [ 77516  83311 215646 ...  34066  84661 257302]

education:
 [' Bachelors' ' HS-grad' ' 11th' ' Masters' ' 9th' ' Some-college'
 ' Assoc-acdm' ' Assoc-voc' ' 7th-8th' ' Doctorate' ' Prof-school'
 ' 5th-6th' ' 10th' ' 1st-4th' ' Preschool' ' 12th']

education_num:
 [13  9  7 14  5 10 12 11  4 16 15  3  6  2  1  8]

marital_status:
 [' Never-married' ' Married-civ-spouse' ' Divorced'
 ' Married-spouse-absent' ' Separated' ' Married-AF-spouse' ' Widowed']

occupation:
 [' Adm-clerical' ' Exec-managerial' ' Handlers-cleaners' ' Prof-specialty'
 ' Other-service' ' Sales' ' Craft-repair' ' Transport-moving'
 ' Farming-fishing' ' Machine-op-inspct' ' Tech-support' ' ?'
 ' Protective-serv' ' Armed-Forces' ' Priv-house-serv']

relationship:
 [' Not-in-family' ' Husband' ' Wife' ' Own-child' ' Unmarried'
 ' Other-relative']

race:
 [' White' ' Black' ' Asian-Pac-Islander' ' Amer-Indian-Eskimo' ' Other']

sex:
 [' Male' ' Female']

capital_gain:
 [ 2174     0 14084  5178  5013  2407 14344 15024  7688 34095  4064  4386
  7298  1409  3674  1055  3464  2050  2176   594 20051  6849  4101  1111
  8614  3411  2597 25236  4650  9386  2463  3103 10605  2964  3325  2580
  3471  4865 99999  6514  1471  2329  2105  2885 25124 10520  2202  2961
 27828  6767  2228  1506 13550  2635  5556  4787  3781  3137  3818  3942
   914   401  2829  2977  4934  2062  2354  5455 15020  1424  3273 22040
  4416  3908 10566   991  4931  1086  7430  6497   114  7896  2346  3418
  3432  2907  1151  2414  2290 15831 41310  4508  2538  3456  6418  1848
  3887  5721  9562  1455  2036  1831 11678  2936  2993  7443  6360  1797
  1173  4687  6723  2009  6097  2653  1639 18481  7978  2387  5060]

capital_loss:
 [   0 2042 1408 1902 1573 1887 1719 1762 1564 2179 1816 1980 1977 1876
 1340 2206 1741 1485 2339 2415 1380 1721 2051 2377 1669 2352 1672  653
 2392 1504 2001 1590 1651 1628 1848 1740 2002 1579 2258 1602  419 2547
 2174 2205 1726 2444 1138 2238  625  213 1539  880 1668 1092 1594 3004
 2231 1844  810 2824 2559 2057 1974  974 2149 1825 1735 1258 2129 2603
 2282  323 4356 2246 1617 1648 2489 3770 1755 3683 2267 2080 2457  155
 3900 2201 1944 2467 2163 2754 2472 1411]

hours_per_week:
 [40 13 16 45 50 80 30 35 60 20 52 44 15 25 38 43 55 48 58 32 70  2 22 56
 41 28 36 24 46 42 12 65  1 10 34 75 98 33 54  8  6 64 19 18 72  5  9 47
 37 21 26 14  4 59  7 99 53 39 62 57 78 90 66 11 49 84  3 17 68 27 85 31
 51 77 63 23 87 88 73 89 97 94 29 96 67 82 86 91 81 76 92 61 74 95]

native_country:
 [' United-States' ' Cuba' ' Jamaica' ' India' ' ?' ' Mexico' ' South'
 ' Puerto-Rico' ' Honduras' ' England' ' Canada' ' Germany' ' Iran'
 ' Philippines' ' Italy' ' Poland' ' Columbia' ' Cambodia' ' Thailand'
 ' Ecuador' ' Laos' ' Taiwan' ' Haiti' ' Portugal' ' Dominican-Republic'
 ' El-Salvador' ' France' ' Guatemala' ' China' ' Japan' ' Yugoslavia'
 ' Peru' ' Outlying-US(Guam-USVI-etc)' ' Scotland' ' Trinadad&Tobago'
 ' Greece' ' Nicaragua' ' Vietnam' ' Hong' ' Ireland' ' Hungary'
 ' Holand-Netherlands']

income:
 [' <=50K' ' >50K']
```

In the above cell 👆 when are checking the unique values, we are encountering some "?"s in some columns like 'workclass', 'occupation' & 'native_country'.

In [10]:

```python
# Finding total no.of '?':
for i in adult.columns:
    print(f"{i}: {sum(adult[i]=='?')}")
```

```
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
```

- So, when we are trying to find the total no.of '?' present in each column, we are being unable to find it.

- This is because our string present in each column have an unnecessary spaces before it, therefore we need to remove those unnecessary spaces.

There are 3 ways to remove the unnecessary leading and trailing spaces from the string values in our dataset.

- 1. using delimiter=' *, *'
- 2. using sep=r'\s*,\s*', engine='python'
- 3. using skipinitialspace=True

1. To remove the unnecessary spaces present at the starting part of the string in each columns, we need to use delimiter=' *, *'

In [12]:

```
adult1 = pd.read_csv(r"C:\Users\lab25\Downloads\adult\adult.data",
                     header=None,
                     delimiter=' *, *')
```

```
C:\Users\lab25\AppData\Local\Temp\ipykernel_4192\1600029040.py:1: ParserWarning: Fal
ling back to the 'python' engine because the 'c' engine does not support regex separ
ators (separators > 1 char and different from '\s+' are interpreted as regex); you c
an avoid this warning by specifying engine='python'.
  adult1 = pd.read_csv(r"C:\Users\lab25\Downloads\adult\adult.data",
```

In [13]:

```
adult1
```

Out[13]:

	0	1	2	3	4	5	6	7	8	9
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female
...	...	...	...	...	...	...	...	...	...	...
32556	27	Private	257302	Assoc-acdm	12	Married-civ-spouse	Tech-support	Wife	White	Female
32557	40	Private	154374	HS-grad	9	Married-civ-spouse	Machine-op-inspct	Husband	White	Male
32558	58	Private	151910	HS-grad	9	Widowed	Adm-clerical	Unmarried	White	Female
32559	22	Private	201490	HS-grad	9	Never-married	Adm-clerical	Own-child	White	Male
32560	52	Self-emp-inc	287927	HS-grad	9	Married-civ-spouse	Exec-managerial	Wife	White	Female

32561 rows × 15 columns

In [14]:

```
adult1.columns = ['age','workclass','fnlwgt',
                  'education','education_num',
                 'marital_status','occupation','relationship',
                 'race','sex','capital_gain','capital_loss',
                 'hours_per_week','native_country','income']
```

Here, if we consider the below cell 👇, we can see the unnecessary spaces have been removed.

In [17]:

```
for i in adult1.columns:
    print(f"{i}:\n {adult1[i].unique()}\n")
```

```
age:
 [39 50 38 53 28 37 49 52 31 42 30 23 32 40 34 25 43 54 35 59 56 19 20 45
 22 48 21 24 57 44 41 29 18 47 46 36 79 27 67 33 76 17 55 61 70 64 71 68
 66 51 58 26 60 90 75 65 77 62 63 80 72 74 69 73 81 78 88 82 83 84 85 86
 87]

workclass:
 ['State-gov' 'Self-emp-not-inc' 'Private' 'Federal-gov' 'Local-gov' '?'
 'Self-emp-inc' 'Without-pay' 'Never-worked']

fnlwgt:
 [ 77516  83311 215646 ...  34066  84661 257302]

education:
 ['Bachelors' 'HS-grad' '11th' 'Masters' '9th' 'Some-college' 'Assoc-acdm'
 'Assoc-voc' '7th-8th' 'Doctorate' 'Prof-school' '5th-6th' '10th'
 '1st-4th' 'Preschool' '12th']

education_num:
 [13  9  7 14  5 10 12 11  4 16 15  3  6  2  1  8]

marital_status:
 ['Never-married' 'Married-civ-spouse' 'Divorced' 'Married-spouse-absent'
 'Separated' 'Married-AF-spouse' 'Widowed']

occupation:
 ['Adm-clerical' 'Exec-managerial' 'Handlers-cleaners' 'Prof-specialty'
 'Other-service' 'Sales' 'Craft-repair' 'Transport-moving'
 'Farming-fishing' 'Machine-op-inspct' 'Tech-support' '?'
 'Protective-serv' 'Armed-Forces' 'Priv-house-serv']

relationship:
 ['Not-in-family' 'Husband' 'Wife' 'Own-child' 'Unmarried' 'Other-relative']

race:
 ['White' 'Black' 'Asian-Pac-Islander' 'Amer-Indian-Eskimo' 'Other']

sex:
 ['Male' 'Female']

capital_gain:
 [ 2174     0 14084  5178  5013  2407 14344 15024  7688 34095  4064  4386
  7298  1409  3674  1055  3464  2050  2176   594 20051  6849  4101  1111
  8614  3411  2597 25236  4650  9386  2463  3103 10605  2964  3325  2580
  3471  4865 99999  6514  1471  2329  2105  2885 25124 10520  2202  2961
 27828  6767  2228  1506 13550  2635  5556  4787  3781  3137  3818  3942
   914   401  2829  2977  4934  2062  2354  5455 15020  1424  3273 22040
  4416  3908 10566   991  4931  1086  7430  6497   114  7896  2346  3418
  3432  2907  1151  2414  2290 15831 41310  4508  2538  3456  6418  1848
  3887  5721  9562  1455  2036  1831 11678  2936  2993  7443  6360  1797
  1173  4687  6723  2009  6097  2653  1639 18481  7978  2387  5060]

capital_loss:
 [   0 2042 1408 1902 1573 1887 1719 1762 1564 2179 1816 1980 1977 1876
 1340 2206 1741 1485 2339 2415 1380 1721 2051 2377 1669 2352 1672  653
 2392 1504 2001 1590 1651 1628 1848 1740 2002 1579 2258 1602  419 2547
 2174 2205 1726 2444 1138 2238  625  213 1539  880 1668 1092 1594 3004
 2231 1844  810 2824 2559 2057 1974  974 2149 1825 1735 1258 2129 2603
 2282  323 4356 2246 1617 1648 2489 3770 1755 3683 2267 2080 2457  155
 3900 2201 1944 2467 2163 2754 2472 1411]

hours_per_week:
 [40 13 16 45 50 80 30 35 60 20 52 44 15 25 38 43 55 48 58 32 70  2 22 56
 41 28 36 24 46 42 12 65  1 10 34 75 98 33 54  8  6 64 19 18 72  5  9 47
 37 21 26 14  4 59  7 99 53 39 62 57 78 90 66 11 49 84  3 17 68 27 85 31
 51 77 63 23 87 88 73 89 97 94 29 96 67 82 86 91 81 76 92 61 74 95]

native_country:
 ['United-States' 'Cuba' 'Jamaica' 'India' '?' 'Mexico' 'South'
 'Puerto-Rico' 'Honduras' 'England' 'Canada' 'Germany' 'Iran'
 'Philippines' 'Italy' 'Poland' 'Columbia' 'Cambodia' 'Thailand' 'Ecuador'
 'Laos' 'Taiwan' 'Haiti' 'Portugal' 'Dominican-Republic' 'El-Salvador'
 'France' 'Guatemala' 'China' 'Japan' 'Yugoslavia' 'Peru'
 'Outlying-US(Guam-USVI-etc)' 'Scotland' 'Trinadad&Tobago' 'Greece'
 'Nicaragua' 'Vietnam' 'Hong' 'Ireland' 'Hungary' 'Holand-Netherlands']

income:
 ['<=50K' '>50K']
```

In [21]:

```python
# Finding total no.of '?':
for i in adult1.columns:
    print(f"{i}: {sum(adult1[i]=='?')}")
```

```
age: 0
workclass: 1836
fnlwgt: 0
education: 0
education_num: 0
marital_status: 0
occupation: 1843
relationship: 0
race: 0
sex: 0
capital_gain: 0
capital_loss: 0
hours_per_week: 0
native_country: 583
income: 0
```

2. To remove the unnecessary spaces present at the starting part of the string in each columns, we can use sep=r'\s*,\s*', engine='python'

In [16]:

```python
adult2 = pd.read_csv(r"C:\Users\lab25\Downloads\adult\adult.data",
                     header=None,sep=r'\s*,\s*', engine='python')
adult2
```

Out[16]:

	0	1	2	3	4	5	6	7	8	9
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female
...	...	...	...	...	...	...	...	...	...	...
32556	27	Private	257302	Assoc-acdm	12	Married-civ-spouse	Tech-support	Wife	White	Female
32557	40	Private	154374	HS-grad	9	Married-civ-spouse	Machine-op-inspct	Husband	White	Male
32558	58	Private	151910	HS-grad	9	Widowed	Adm-clerical	Unmarried	White	Female
32559	22	Private	201490	HS-grad	9	Never-married	Adm-clerical	Own-child	White	Male
32560	52	Self-emp-inc	287927	HS-grad	9	Married-civ-spouse	Exec-managerial	Wife	White	Female

32561 rows × 15 columns

In [18]:

```
for i in adult1.columns:
    print(f"{i}:\n {adult1[i].unique()}\n")
```

```
age:
 [39 50 38 53 28 37 49 52 31 42 30 23 32 40 34 25 43 54 35 59 56 19 20 45
 22 48 21 24 57 44 41 29 18 47 46 36 79 27 67 33 76 17 55 61 70 64 71 68
 66 51 58 26 60 90 75 65 77 62 63 80 72 74 69 73 81 78 88 82 83 84 85 86
 87]

workclass:
 ['State-gov' 'Self-emp-not-inc' 'Private' 'Federal-gov' 'Local-gov' '?'
 'Self-emp-inc' 'Without-pay' 'Never-worked']

fnlwgt:
 [ 77516  83311 215646 ...  34066  84661 257302]

education:
 ['Bachelors' 'HS-grad' '11th' 'Masters' '9th' 'Some-college' 'Assoc-acdm'
 'Assoc-voc' '7th-8th' 'Doctorate' 'Prof-school' '5th-6th' '10th'
 '1st-4th' 'Preschool' '12th']

education_num:
 [13  9  7 14  5 10 12 11  4 16 15  3  6  2  1  8]

marital_status:
 ['Never-married' 'Married-civ-spouse' 'Divorced' 'Married-spouse-absent'
 'Separated' 'Married-AF-spouse' 'Widowed']

occupation:
 ['Adm-clerical' 'Exec-managerial' 'Handlers-cleaners' 'Prof-specialty'
 'Other-service' 'Sales' 'Craft-repair' 'Transport-moving'
 'Farming-fishing' 'Machine-op-inspct' 'Tech-support' '?'
 'Protective-serv' 'Armed-Forces' 'Priv-house-serv']

relationship:
 ['Not-in-family' 'Husband' 'Wife' 'Own-child' 'Unmarried' 'Other-relative']

race:
 ['White' 'Black' 'Asian-Pac-Islander' 'Amer-Indian-Eskimo' 'Other']

sex:
 ['Male' 'Female']

capital_gain:
 [ 2174     0 14084  5178  5013  2407 14344 15024  7688 34095  4064  4386
   7298  1409  3674  1055  3464  2050  2176   594 20051  6849  4101  1111
   8614  3411  2597 25236  4650  9386  2463  3103 10605  2964  3325  2580
   3471  4865 99999  6514  1471  2329  2105  2885 25124 10520  2202  2961
  27828  6767  2228  1506 13550  2635  5556  4787  3781  3137  3818  3942
    914   401  2829  2977  4934  2062  2354  5455 15020  1424  3273 22040
   4416  3908 10566   991  4931  1086  7430  6497   114  7896  2346  3418
   3432  2907  1151  2414  2290 15831 41310  4508  2538  3456  6418  1848
   3887  5721  9562  1455  2036  1831 11678  2936  2993  7443  6360  1797
   1173  4687  6723  2009  6097  2653  1639 18481  7978  2387  5060]

capital_loss:
 [   0 2042 1408 1902 1573 1887 1719 1762 1564 2179 1816 1980 1977 1876
  1340 2206 1741 1485 2339 2415 1380 1721 2051 2377 1669 2352 1672  653
  2392 1504 2001 1590 1651 1628 1848 1740 2002 1579 2258 1602  419 2547
 2174 2205 1726 2444 1138 2238  625  213 1539  880 1668 1092 1594 3004
 2231 1844  810 2824 2559 2057 1974  974 2149 1825 1735 1258 2129 2603
 2282  323 4356 2246 1617 1648 2489 3770 1755 3683 2267 2080 2457  155
 3900 2201 1944 2467 2163 2754 2472 1411]

hours_per_week:
 [40 13 16 45 50 80 30 35 60 20 52 44 15 25 38 43 55 48 58 32 70  2 22 56
 41 28 36 24 46 42 12 65  1 10 34 75 98 33 54  8  6 64 19 18 72  5  9 47
 37 21 26 14  4 59  7 99 53 39 62 57 78 90 66 11 49 84  3 17 68 27 85 31
 51 77 63 23 87 88 73 89 97 94 29 96 67 82 86 91 81 76 92 61 74 95]

native_country:
 ['United-States' 'Cuba' 'Jamaica' 'India' '?' 'Mexico' 'South'
 'Puerto-Rico' 'Honduras' 'England' 'Canada' 'Germany' 'Iran'
 'Philippines' 'Italy' 'Poland' 'Columbia' 'Cambodia' 'Thailand' 'Ecuador'
 'Laos' 'Taiwan' 'Haiti' 'Portugal' 'Dominican-Republic' 'El-Salvador'
 'France' 'Guatemala' 'China' 'Japan' 'Yugoslavia' 'Peru'
 'Outlying-US(Guam-USVI-etc)' 'Scotland' 'Trinadad&Tobago' 'Greece'
 'Nicaragua' 'Vietnam' 'Hong' 'Ireland' 'Hungary' 'Holand-Netherlands']

income:
 ['<=50K' '>50K']
```

3. To remove the unnecessary spaces present at the starting part of the string in each columns, we can also use skipinitialspace=True

In [19]:

```
adult2 = pd.read_csv(r"C:\Users\lab25\Downloads\adult\adult.data",
                     header=None,skipinitialspace=True)
adult2
```

Out[19]:

	0	1	2	3	4	5	6	7	8	9
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female
...	...	...	...	...	...	...	...	...	...	...
32556	27	Private	257302	Assoc-acdm	12	Married-civ-spouse	Tech-support	Wife	White	Female
32557	40	Private	154374	HS-grad	9	Married-civ-spouse	Machine-op-inspct	Husband	White	Male
32558	58	Private	151910	HS-grad	9	Widowed	Adm-clerical	Unmarried	White	Female
32559	22	Private	201490	HS-grad	9	Never-married	Adm-clerical	Own-child	White	Male
32560	52	Self-emp-inc	287927	HS-grad	9	Married-civ-spouse	Exec-managerial	Wife	White	Female

32561 rows × 15 columns

In [20]:

```
for i in adult1.columns:
    print(f"{i}:\n {adult1[i].unique()}\n")
```

```
age:
 [39 50 38 53 28 37 49 52 31 42 30 23 32 40 34 25 43 54 35 59 56 19 20 45
 22 48 21 24 57 44 41 29 18 47 46 36 79 27 67 33 76 17 55 61 70 64 71 68
 66 51 58 26 60 90 75 65 77 62 63 80 72 74 69 73 81 78 88 82 83 84 85 86
 87]

workclass:
 ['State-gov' 'Self-emp-not-inc' 'Private' 'Federal-gov' 'Local-gov' '?'
 'Self-emp-inc' 'Without-pay' 'Never-worked']

fnlwgt:
 [ 77516  83311 215646 ...  34066  84661 257302]

education:
 ['Bachelors' 'HS-grad' '11th' 'Masters' '9th' 'Some-college' 'Assoc-acdm'
 'Assoc-voc' '7th-8th' 'Doctorate' 'Prof-school' '5th-6th' '10th'
 '1st-4th' 'Preschool' '12th']

education_num:
 [13  9  7 14  5 10 12 11  4 16 15  3  6  2  1  8]

marital_status:
 ['Never-married' 'Married-civ-spouse' 'Divorced' 'Married-spouse-absent'
 'Separated' 'Married-AF-spouse' 'Widowed']

occupation:
 ['Adm-clerical' 'Exec-managerial' 'Handlers-cleaners' 'Prof-specialty'
 'Other-service' 'Sales' 'Craft-repair' 'Transport-moving'
 'Farming-fishing' 'Machine-op-inspct' 'Tech-support' '?'
 'Protective-serv' 'Armed-Forces' 'Priv-house-serv']

relationship:
 ['Not-in-family' 'Husband' 'Wife' 'Own-child' 'Unmarried' 'Other-relative']

race:
 ['White' 'Black' 'Asian-Pac-Islander' 'Amer-Indian-Eskimo' 'Other']

sex:
 ['Male' 'Female']

capital_gain:
 [ 2174     0 14084  5178  5013  2407 14344 15024  7688 34095  4064  4386
  7298  1409  3674  1055  3464  2050  2176   594 20051  6849  4101  1111
  8614  3411  2597 25236  4650  9386  2463  3103 10605  2964  3325  2580
  3471  4865 99999  6514  1471  2329  2105  2885 25124 10520  2202  2961
 27828  6767  2228  1506 13550  2635  5556  4787  3781  3137  3818  3942
   914   401  2829  2977  4934  2062  2354  5455 15020  1424  3273 22040
  4416  3908 10566   991  4931  1086  7430  6497   114  7896  2346  3418
  3432  2907  1151  2414  2290 15831 41310  4508  2538  3456  6418  1848
  3887  5721  9562  1455  2036  1831 11678  2936  2993  7443  6360  1797
  1173  4687  6723  2009  6097  2653  1639 18481  7978  2387  5060]

capital_loss:
 [   0 2042 1408 1902 1573 1887 1719 1762 1564 2179 1816 1980 1977 1876
 1340 2206 1741 1485 2339 2415 1380 1721 2051 2377 1669 2352 1672  653
 2392 1504 2001 1590 1651 1628 1848 1740 2002 1579 2258 1602  419 2547
 2174 2205 1726 2444 1138 2238  625  213 1539  880 1668 1092 1594 3004
 2231 1844  810 2824 2559 2057 1974  974 2149 1825 1735 1258 2129 2603
 2282  323 4356 2246 1617 1648 2489 3770 1755 3683 2267 2080 2457  155
 3900 2201 1944 2467 2163 2754 2472 1411]

hours_per_week:
 [40 13 16 45 50 80 30 35 60 20 52 44 15 25 38 43 55 48 58 32 70  2 22 56
 41 28 36 24 46 42 12 65  1 10 34 75 98 33 54  8  6 64 19 18 72  5  9 47
 37 21 26 14  4 59  7 99 53 39 62 57 78 90 66 11 49 84  3 17 68 27 85 31
 51 77 63 23 87 88 73 89 97 94 29 96 67 82 86 91 81 76 92 61 74 95]

native_country:
 ['United-States' 'Cuba' 'Jamaica' 'India' '?' 'Mexico' 'South'
 'Puerto-Rico' 'Honduras' 'England' 'Canada' 'Germany' 'Iran'
 'Philippines' 'Italy' 'Poland' 'Columbia' 'Cambodia' 'Thailand' 'Ecuador'
 'Laos' 'Taiwan' 'Haiti' 'Portugal' 'Dominican-Republic' 'El-Salvador'
 'France' 'Guatemala' 'China' 'Japan' 'Yugoslavia' 'Peru'
 'Outlying-US(Guam-USVI-etc)' 'Scotland' 'Trinadad&Tobago' 'Greece'
 'Nicaragua' 'Vietnam' 'Hong' 'Ireland' 'Hungary' 'Holand-Netherlands']

income:
 ['<=50K' '>50K']
```

Now, after removing the unnecessary spaces using any of those 3 methods as shown above 👆, we can perform the cleaning process of replacing the '?' with some value.

In [21]:
```python
# Checking for the missing values:-
for i in adult1.columns:
    print(f"{i}: {sum(adult1[i]=='?')}")
```

```
age: 0
workclass: 1836
fnlwgt: 0
education: 0
education_num: 0
marital_status: 0
occupation: 1843
relationship: 0
race: 0
sex: 0
capital_gain: 0
capital_loss: 0
hours_per_week: 0
native_country: 583
income: 0
```

Now, we need to replace the '?' with the mean(if a numeric column) or with the top repeated(if a string column) values.

Finding the mean or top repeated values or all the statistical values using describe(include=all')

In [22]:

```
des = adult1.describe(include='all')
des
```

Out[22]:

	age	workclass	fnlwgt	education	education_num	marital_status
count	32561.000000	32561	3.256100e+04	32561	32561.000000	32561
unique	NaN	9	NaN	16	NaN	7
top	NaN	Private	NaN	HS-grad	NaN	Married-civ-spouse
freq	NaN	22696	NaN	10501	NaN	14976
mean	38.581647	NaN	1.897784e+05	NaN	10.080679	NaN
std	13.640433	NaN	1.055500e+05	NaN	2.572720	NaN
min	17.000000	NaN	1.228500e+04	NaN	1.000000	NaN
25%	28.000000	NaN	1.178270e+05	NaN	9.000000	NaN
50%	37.000000	NaN	1.783560e+05	NaN	10.000000	NaN
75%	48.000000	NaN	2.370510e+05	NaN	12.000000	NaN
max	90.000000	NaN	1.484705e+06	NaN	16.000000	NaN

- As 'workclass' is a string column, we can replace the '?'(missing values) with the top repeated values.

- As 'occupation' is also a string column, we can replace the '?'(missing values) with the top repeated values.

- As 'native_country' is also a string column, we can replace the '?'(missing values) with the top repeated values.

In [24]:

```
for i in ['workclass','occupation','native_country']:
    adult1[i].replace('?',des[i][2],inplace=True)
```

```
C:\Users\lab25\AppData\Local\Temp\ipykernel_4192\2683609913.py:2: FutureWarning: Ser
ies.__getitem__ treating keys as positions is deprecated. In a future version, integ
er keys will always be treated as labels (consistent with DataFrame behavior). To ac
cess a value by position, use `ser.iloc[pos]`
  adult1[i].replace('?',des[i][2],inplace=True)
```

In [25]:

```
for i in adult1.columns:
    print(f"{i}: {sum(adult1[i]=='?')}")
```

```
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
```

Data Visualization:-

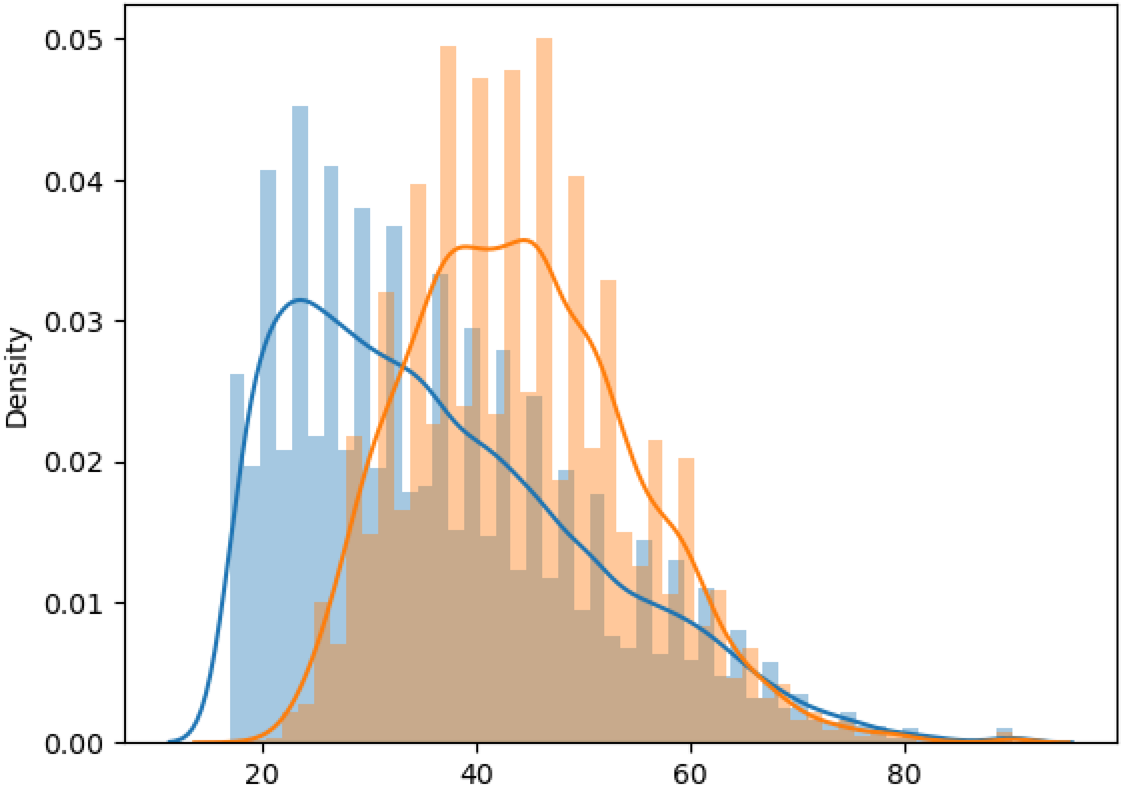
Considering all numeric columns we can plot distplot, boxplot against the target column "income".

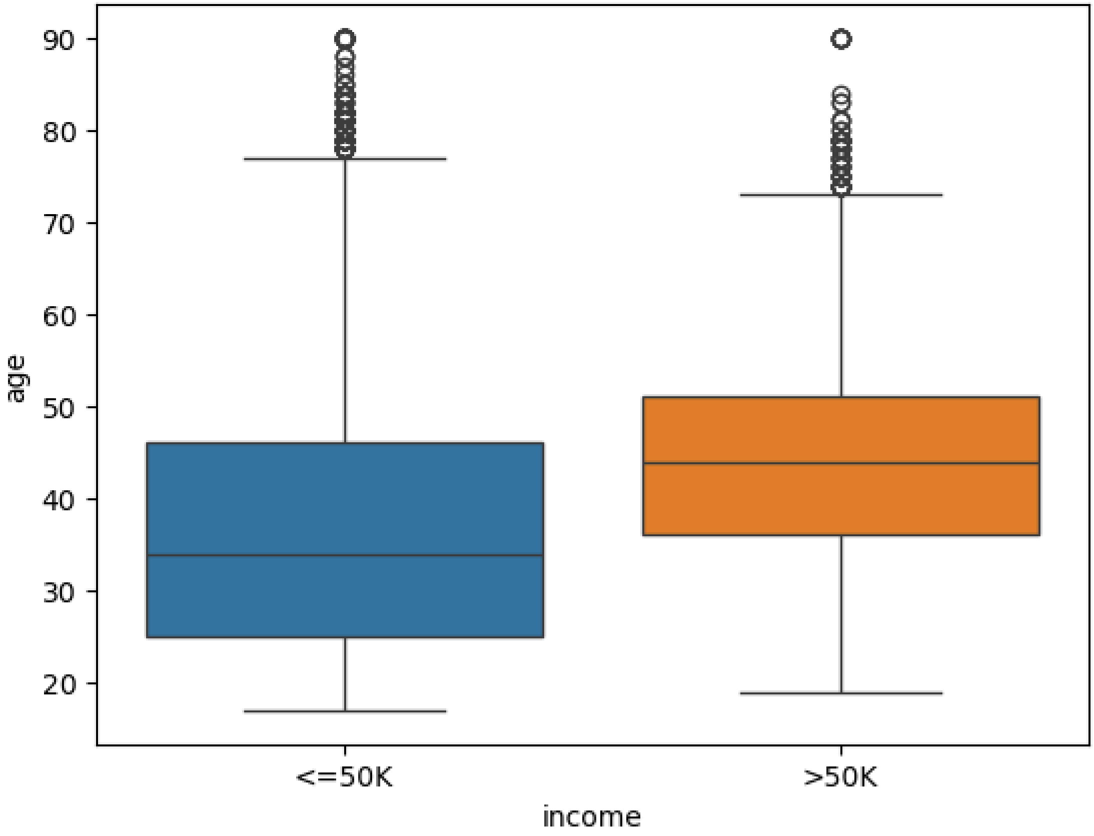
In [27]:

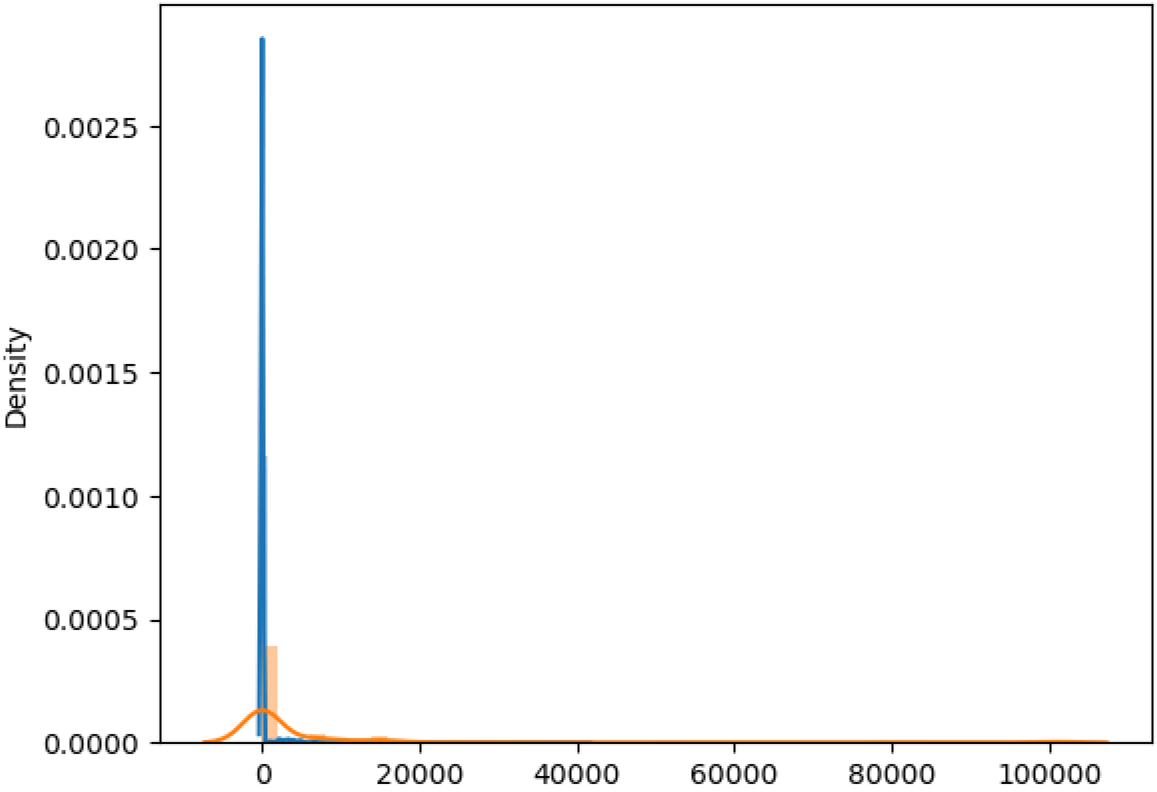
```
import warnings
warnings.filterwarnings('ignore')
```

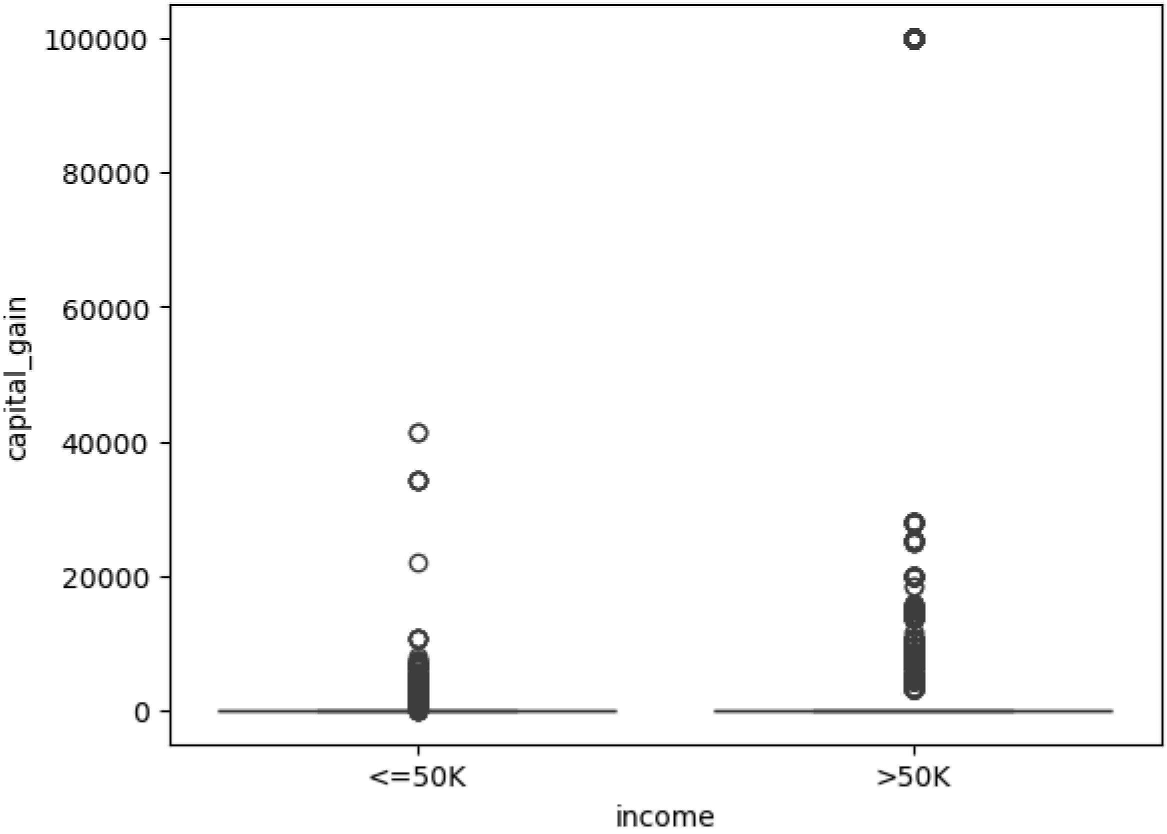
In [30]:

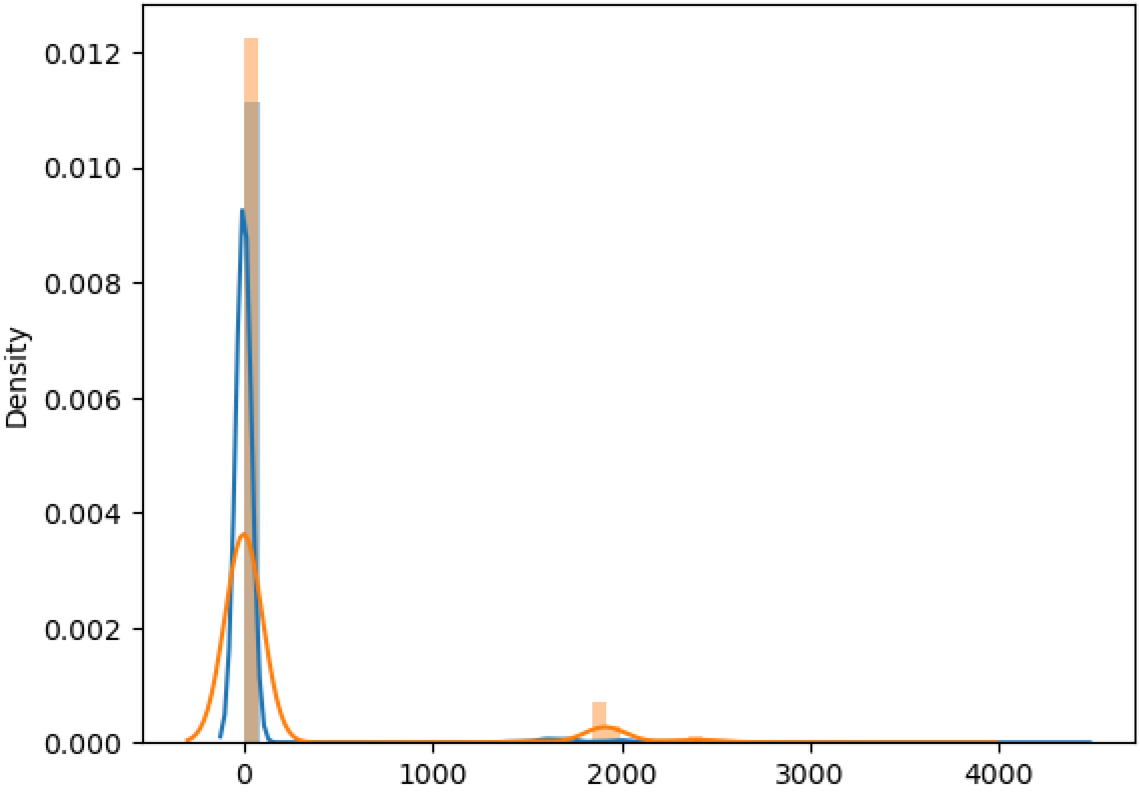
```
for i in ['age','capital_gain','capital_loss','hours_per_week']:
    # distplot
    sns.distplot(x=adult1[i][adult1.income=='<=50K'])
    sns.distplot(x=adult1[i][adult1.income=='>50K'])
    plt.show()

    # boxplot
    sns.boxplot(x=adult1.income,y=adult1[i],hue=adult1.income)
    plt.show()
```

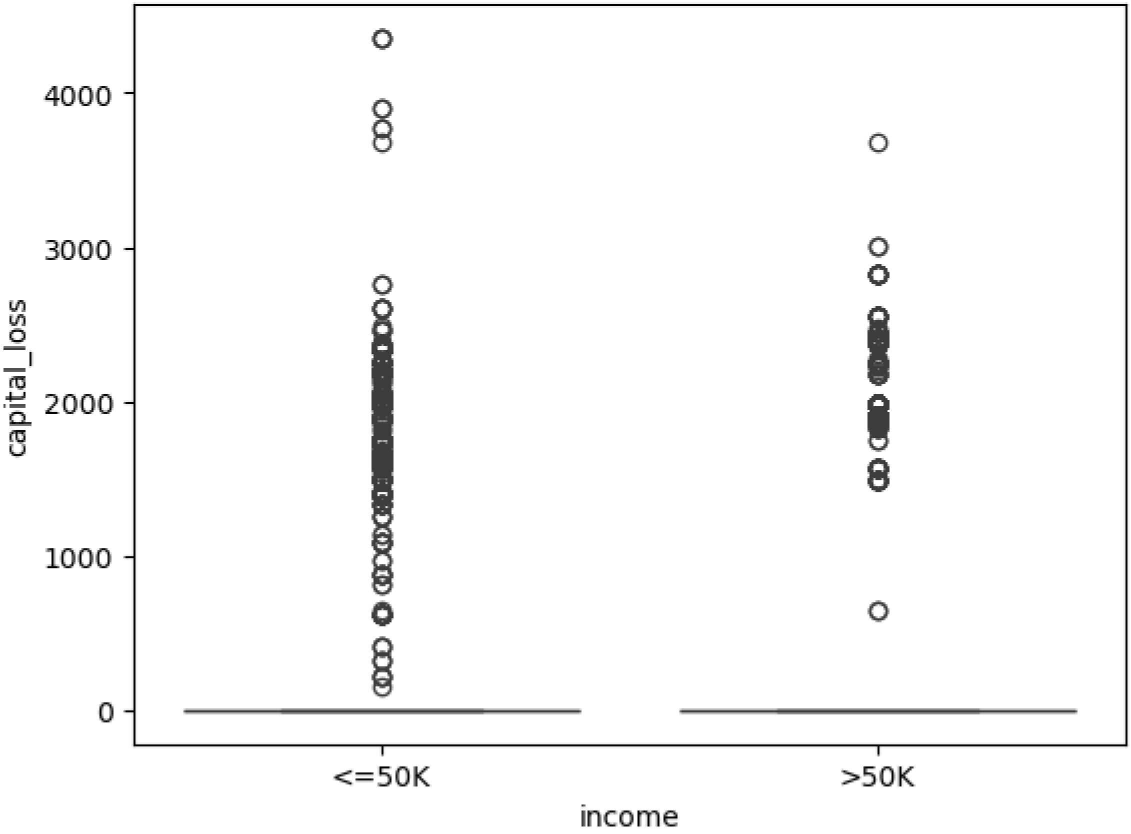

0.05
0.04
0.03
0.02
0.01
0.00
Density
20
40
60
80

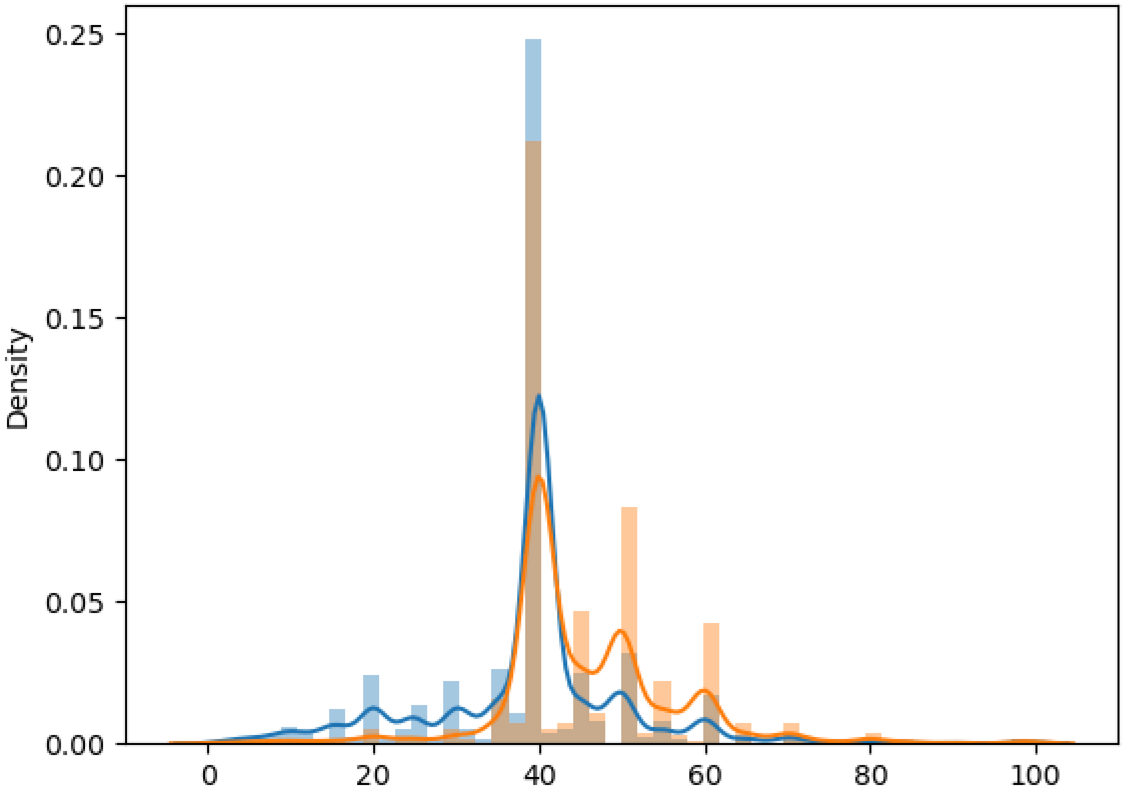

90
80
70
60
50
40
30
20
age
<=50K
>50K
income

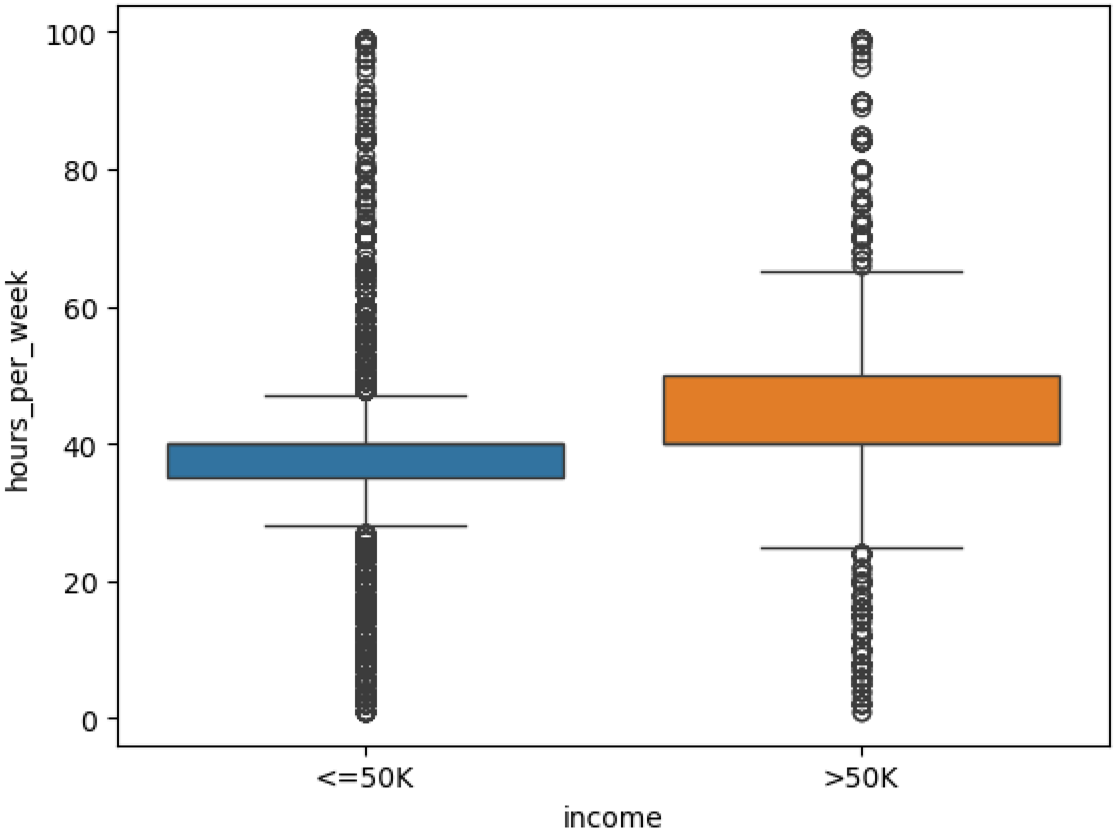

Density
0.0025
0.0020
0.0015
0.0010
0.0005
0.0000
0
20000
40000
60000
80000
100000


capital_gain
100000
80000
60000
40000
20000
0
<=50K
>50K
income

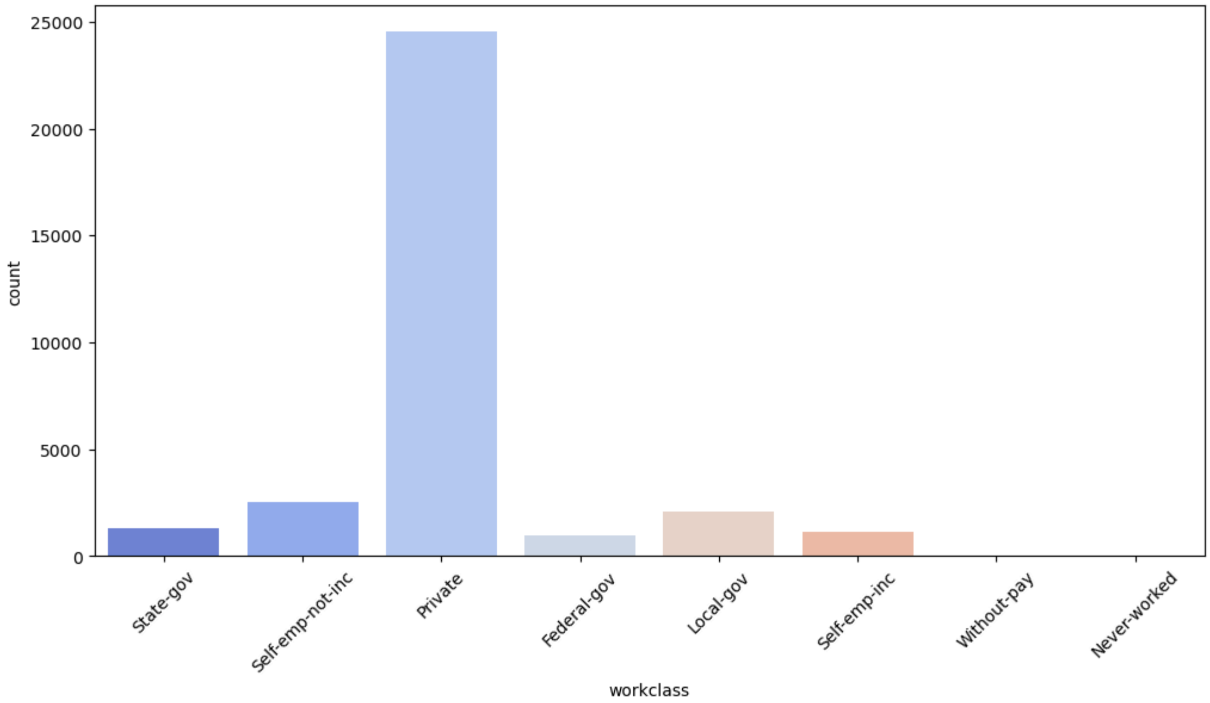

Density
0.000
0.002
0.004
0.006
0.008
0.010
0.012
0
1000
2000
3000
4000


capital_loss
0
1000
2000
3000
4000
<=50K
>50K
income

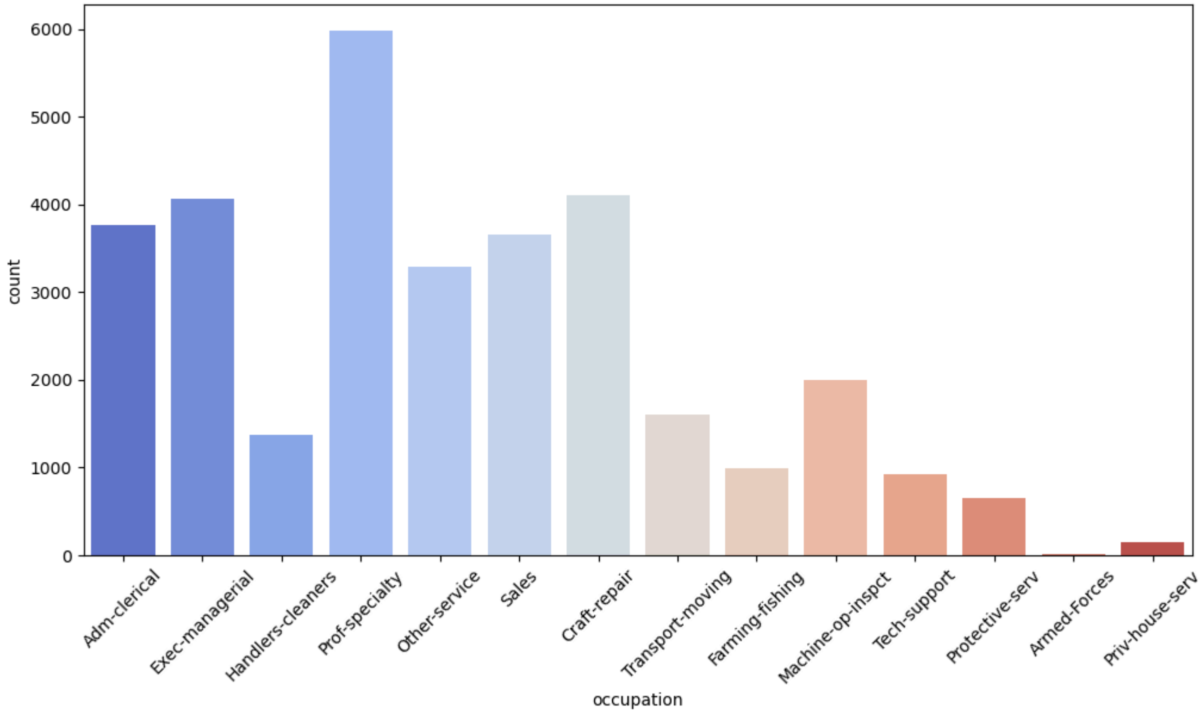




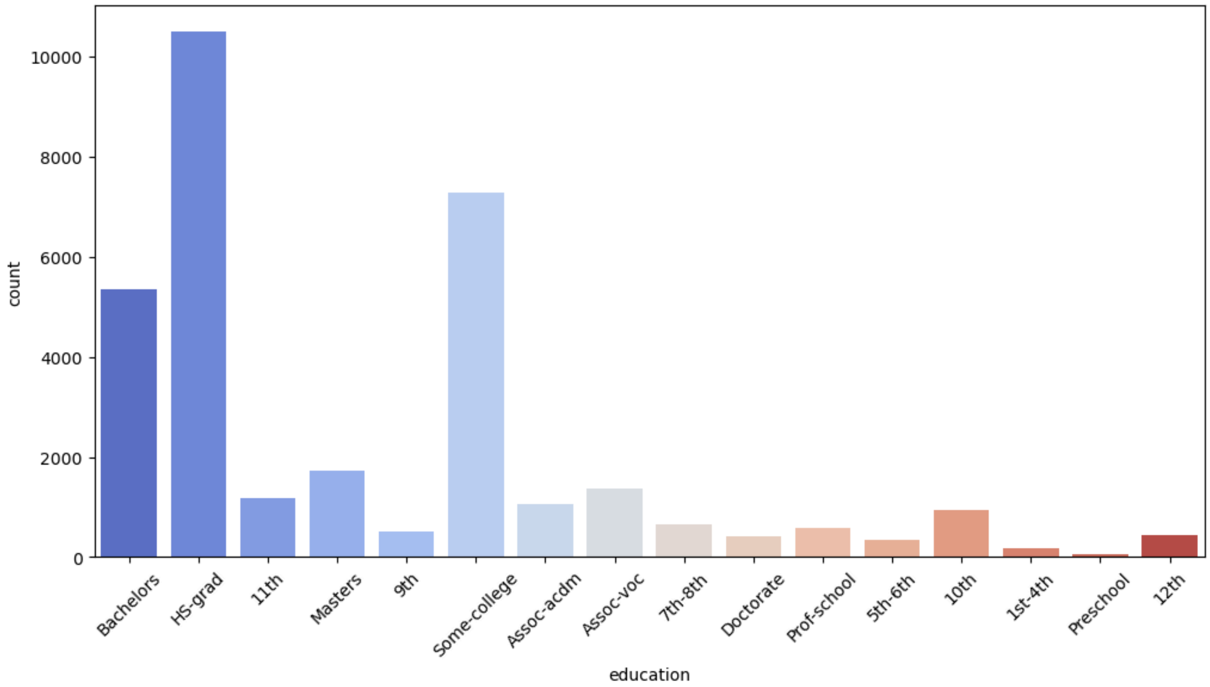
In [39]:
```
# count plot:
plt.figure(figsize=(12, 6))  # Set figure size first
sns.countplot(x='workclass', data=adult1, palette='coolwarm')
plt.xticks(rotation=45)  # Rotate x-axis labels for readability
plt.show()
```



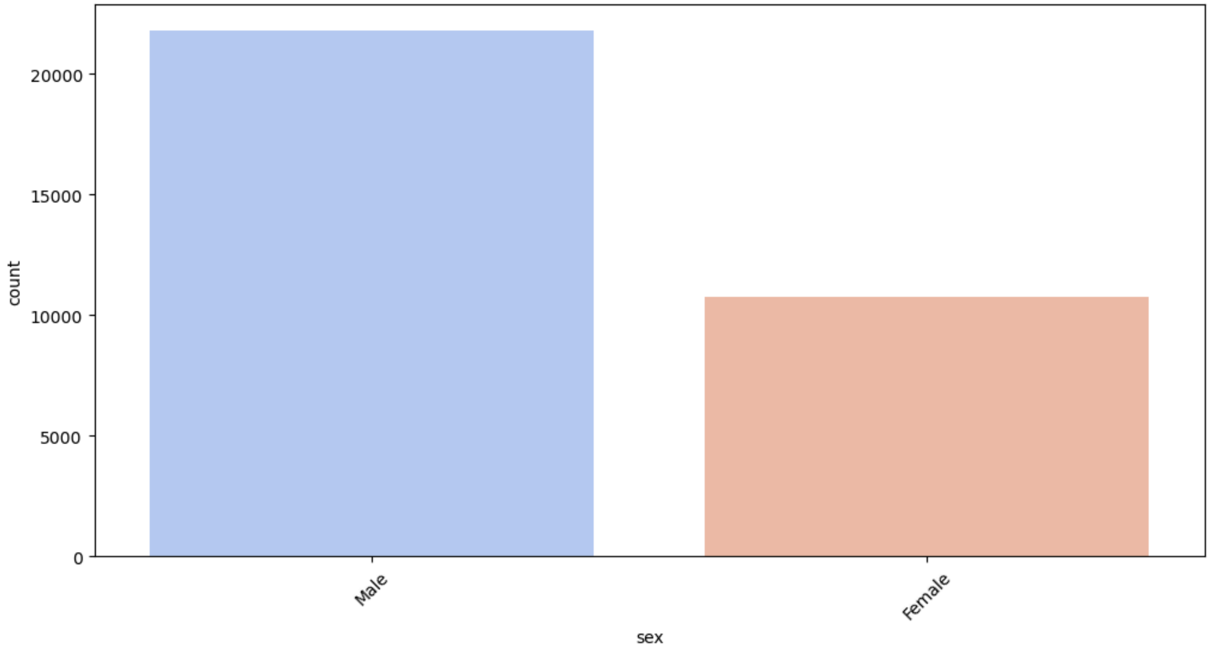
In [40]:
```
# count plot:
plt.figure(figsize=(12, 6))  # Set figure size first
sns.countplot(x='occupation', data=adult1, palette='coolwarm')
plt.xticks(rotation=45)  # Rotate x-axis labels for readability
plt.show()
```



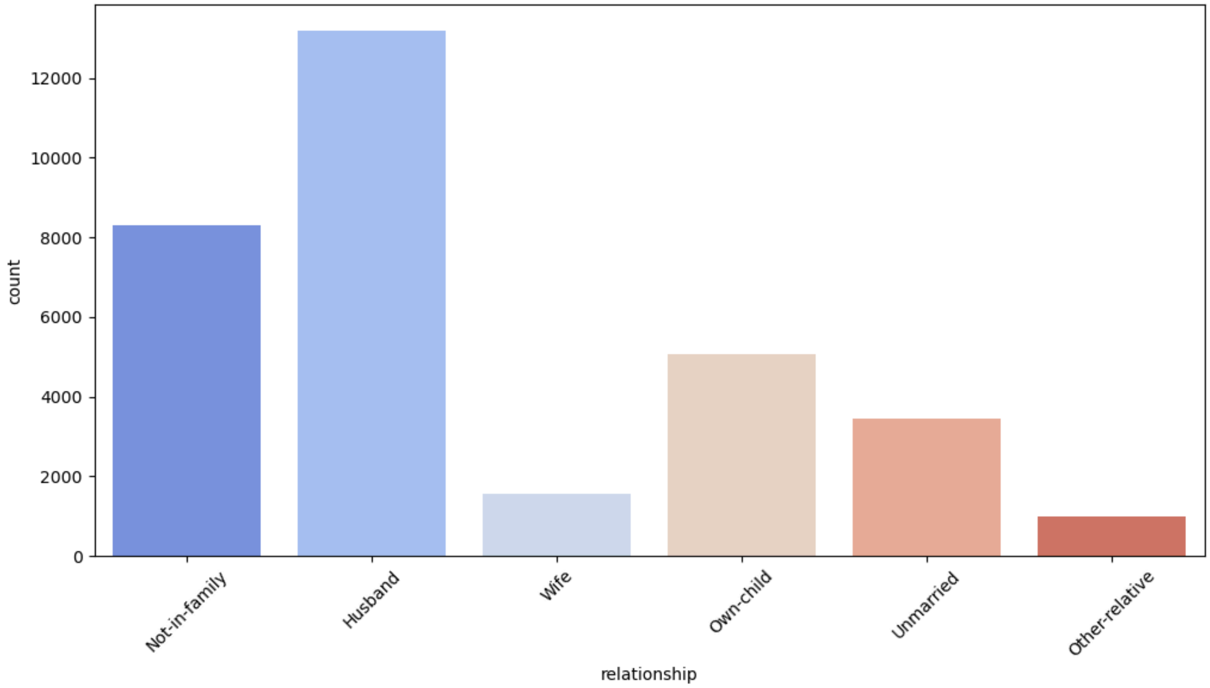
In [41]:
```
# count plot:
plt.figure(figsize=(12, 6))  # Set figure size first
sns.countplot(x='education', data=adult1, palette='coolwarm')
plt.xticks(rotation=45)  # Rotate x-axis labels for readability
plt.show()
```



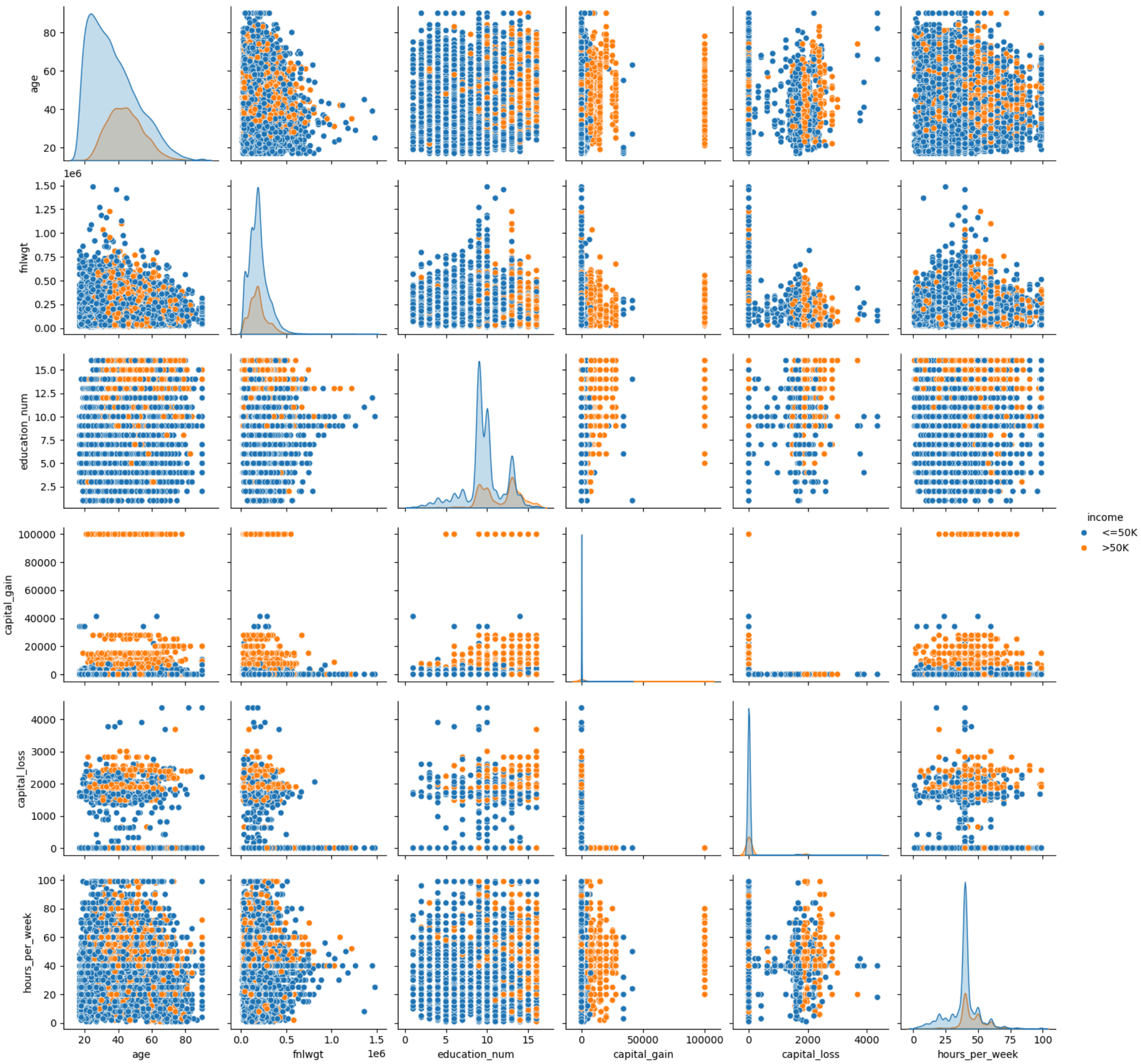
In [42]:
```
# count plot:
plt.figure(figsize=(12, 6))  # Set figure size first
sns.countplot(x='sex', data=adult1, palette='coolwarm')
plt.xticks(rotation=45)  # Rotate x-axis labels for readability
plt.show()
```



In [43]:
```
# count plot:
plt.figure(figsize=(12, 6))  # Set figure size first
sns.countplot(x='relationship', data=adult1, palette='coolwarm')
plt.xticks(rotation=45)  # Rotate x-axis labels for readability
plt.show()
```



In [45]:
```
sns.pairplot(adult1,hue='income')
plt.show()
```


age
80
60
40
20
1e6
fnlwgt
1.50
1.25
1.00
0.75
0.50
0.25
0.00
education_num
15.0
12.5
10.0
7.5
5.0
2.5
capital_gain
100000
80000
60000
40000
20000
0
capital_loss
4000
3000
2000
1000
0
hours_per_week
100
80
60
40
20
0
20 40 60 80
age
0.0 0.5 1.0 1.5
1e6
fnlwgt
0 5 10 15
education_num
0 50000 100000
capital_gain
0 2000 4000
capital_loss
0 25 50 75 100
hours_per_week
income
<=50K
>50K

In [ ]: