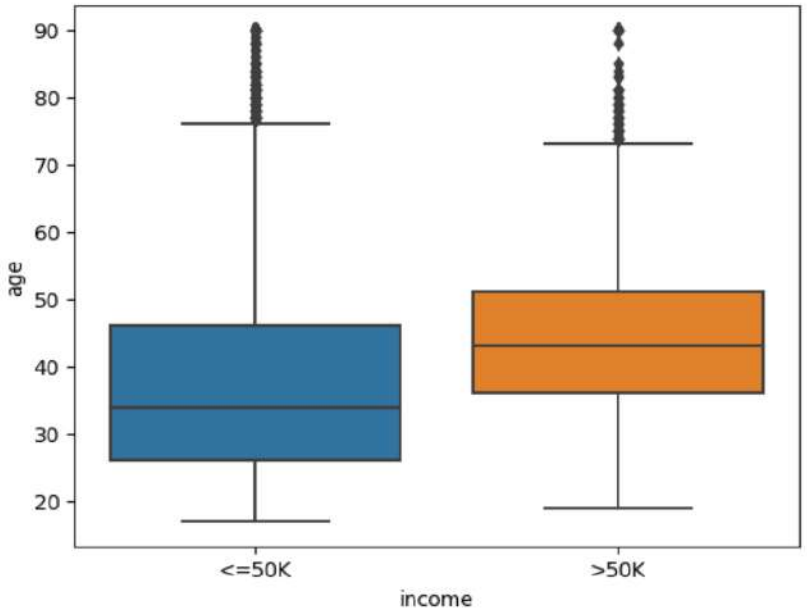
In [56]:
```
data.columns
```

Out[56]:
```
Index(['age', 'workclass', 'fnlwgt', 'education', 'marital-status',
       'occupation', 'relationship', 'race', 'gender', 'hours-per-week',
       'native-country', 'income'],
      dtype='object')
```

In [57]:
```
sns.boxplot(x="income", y="age", data=data)
```

Out[57]:
```
<Axes: xlabel='income', ylabel='age'>
```



REPLACE SALARY VALUES WITH ["<=50K",">50K"] WITH 0 AND 1

In [58]:
```
data.columns
```

Out[58]:
```
Index(['age', 'workclass', 'fnlwgt', 'education', 'marital-status',
       'occupation', 'relationship', 'race', 'gender', 'hours-per-week',
       'native-country', 'income'],
      dtype='object')
```

In [59]:
```
data["income"].unique()
```

Out[59]:
```
array(['<=50K', '>50K'], dtype=object)
```

In [60]:
```
data["income"].value_counts()
```

Out[60]:
```
<=50K    33973
>50K     11202
Name: income, dtype: int64
```

In [64]:
```
#2nd method

data.replace(to_replace=["<=50K",">50K"], value=[0,1], inplace=True)
```

In [66]:
```
data.head(1)
```

Out[66]:

	age	workclass	fnlwgt	education	marital-status	occupation	relationship	race	gender	hours-per-week	native-country	income
0	25	Private	226802	11th	Never-married	Machine-op-inspct	Own-child	Black	Male	40	United-States	0

WHICH WORKCLASS GETTING THE HIGHEST SALARY?

In [67]:
```
data.columns
```

Out[67]:
```
Index(['age', 'workclass', 'fnlwgt', 'education', 'marital-status',
       'occupation', 'relationship', 'race', 'gender', 'hours-per-week',
       'native-country', 'income'],
      dtype='object')
```

In [68]:
```
data.groupby("workclass")["income"].mean().sort_values(ascending=False)
```

Out[68]:
```
workclass
Self-emp-inc        0.554407
Federal-gov         0.390469
Local-gov           0.295161
Self-emp-not-inc    0.279051
State-gov           0.267215
Private             0.217816
Without-pay         0.095238
Name: income, dtype: float64
```

HOW HAS BETTER CHANCE TO GET SALARY GREATER THAN 50K MALSE AND FEMALE?

In [69]:
```
data.columns
```

Out[69]:
```
Index(['age', 'workclass', 'fnlwgt', 'education', 'marital-status',
       'occupation', 'relationship', 'race', 'gender', 'hours-per-week',
       'native-country', 'income'],
      dtype='object')
```

In [70]:
```
data.groupby("gender")["income"].mean().sort_values(ascending=False)
```

Out[70]:
```
gender
Male      0.312609
Female    0.113692
Name: income, dtype: float64
```

CONVERT WORKCLASS COLUMNS DATATYPE TO CATEGORY DATATYPE

In [71]:
```
data.columns
```

Out[71]:
```
Index(['age', 'workclass', 'fnlwgt', 'education', 'marital-status',
       'occupation', 'relationship', 'race', 'gender', 'hours-per-week',
       'native-country', 'income'],
      dtype='object')
```

In [72]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 45175 entries, 0 to 48841
Data columns (total 12 columns):
 #   Column          Non-Null Count  Dtype 
---  ------          --------------  ----- 
 0   age             45175 non-null  int64 
 1   workclass       45175 non-null  object
 2   fnlwgt          45175 non-null  int64 
 3   education       45175 non-null  object
 4   marital-status  45175 non-null  object
 5   occupation      45175 non-null  object
 6   relationship    45175 non-null  object
 7   race            45175 non-null  object
 8   gender          45175 non-null  object
 9   hours-per-week  45175 non-null  int64 
 10  native-country  45175 non-null  object
 11  income          45175 non-null  int64 
dtypes: int64(4), object(8)
memory usage: 4.5+ MB
```

In [73]:
```
data["workclass"]=data["workclass"].astype("category")
```

In [74]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 45175 entries, 0 to 48841
Data columns (total 12 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   age             45175 non-null  int64
 1   workclass       45175 non-null  category
 2   fnlwgt          45175 non-null  int64
 3   education       45175 non-null  object
 4   marital-status  45175 non-null  object
 5   occupation      45175 non-null  object
 6   relationship    45175 non-null  object
 7   race            45175 non-null  object
 8   gender          45175 non-null  object
 9   hours-per-week  45175 non-null  int64
 10  native-country  45175 non-null  object
 11  income          45175 non-null  int64
dtypes: category(1), int64(4), object(7)
memory usage: 4.2+ MB
```

In [ ]: