Read the following data set: https://archive.ics.uci.edu/ml/machine-learning-databases/adult/adult.data (https://archive.ics.uci.edu/ml/machine-learning-databases/adult/adult.data) Task:

1. Create an sqlalchemy engine using a sample from the data set

In [1]:

```
import pandas as pd
import sqlalchemy as sqy
from sqlalchemy import  create_engine,Column,Integer,String,Float
from sqlalchemy.ext.declarative import declarative_base
from sqlalchemy.orm import sessionmaker

#reading the dataSet
df = pd.read_csv('https://archive.ics.uci.edu/ml/machine-learning-databases/adult/adult.data')

# List of Column Names from the Adultnames file
ColumnNames= ['age','workclass','fnlwgt','education','educationNum','maritalStatus','occupation','relationship','race','sex','capitalGain','capitalLoss','hoursperweek','nativeCountry','salperyear']
df.columns = ColumnNames

# Creating a sqlalchemy engine
engine = create_engine('sqlite:///:memory:', echo=False)
print(sqy.__version__)

# Constructing the Base Class from declarative to create table from it
Base = declarative_base()

class Adult(Base):
    __tablename__ = 'adult'

    id = Column(Integer, primary_key=True)
    age = Column(Integer)
    workclass = Column(String)
    fnlwgt = Column(Integer)
    education = Column(String)
    educationNum = Column(Integer)
    maritalStatus = Column(String)
    occupation = Column(String)
    relationship = Column(String)
    race = Column(String)
    sex = Column(String)
    capitalGain = Column(Integer)
    capitalLoss = Column(Integer)
    hoursperweek = Column(Integer)
    nativeCountry = Column(String)
    salperyear = Column(String)

    def __repr__(self):
        return "<Adult(age='%d', workclass='%s', fnlwgt='%d',education='%s', educationNum='%d',maritalStatus ='%s',occupation ='%s',relationship ='%s',race ='%s',sex ='%s',capitalGain ='%d',capitalLoss ='%d',hoursperweek ='%d',nativeCountry ='%s',salperyear ='%s')>" %(
    self.age, self.workclass,self.fnlwgt,self.education,self.educationNum,self.maritalStatus,self.occupation,self.relationship,self.race,self.sex,self.capitalGain,self.capitalLoss,self.hoursperweek,self.nativeCountry,self.salperyear)

# creating the Table in the Engine, i.e memory
Base.metadata.create_all(engine)

#binding the engine to the session
SessionMaker = sessionmaker(bind=engine)

session = SessionMaker()
```

```
# adding the data in the session , by first converting first 5 records to dictio
nary and then accessing individiuallyusing addall to add multiple rows

dict1 = (df.head().to_dict('index'))
for x in dict1.values():
    temp = Adult(**x)
    session.add(temp)

# Commit flushes all the new data in session to the Database in memmory
session.commit()

print("Following are the records from the adult table in memory retrieved and pr
inted :\n",'-'*80)
# now retrieving the records from the session through query and printing
for row in session.query(Adult).all():
    print(row)
    print('-'*80)
```

```
1.2.11
Following are the records from the adult table in memory retrieved a
nd printed :
 --------------------------------------------------------------------
-------------
<Adult(age='50', workclass=' Self-emp-not-inc', fnlwgt='83311',educa
tion=' Bachelors', educationNum='13',maritalStatus =' Married-civ-sp
ouse',occupation =' Exec-managerial',relationship =' Husband',race
=' White',sex =' Male',capitalGain ='0',capitalLoss ='0',hoursperwee
k ='13',nativeCountry =' United-States',salperyear =' <=50K')>
--------------------------------------------------------------------
------------
<Adult(age='38', workclass=' Private', fnlwgt='215646',education=' H
S-grad', educationNum='9',maritalStatus =' Divorced',occupation =' H
andlers-cleaners',relationship =' Not-in-family',race =' White',sex
=' Male',capitalGain ='0',capitalLoss ='0',hoursperweek ='40',native
Country =' United-States',salperyear =' <=50K')>
--------------------------------------------------------------------
------------
<Adult(age='53', workclass=' Private', fnlwgt='234721',education=' 1
1th', educationNum='7',maritalStatus =' Married-civ-spouse',occupati
on =' Handlers-cleaners',relationship =' Husband',race =' Black',sex
=' Male',capitalGain ='0',capitalLoss ='0',hoursperweek ='40',native
Country =' United-States',salperyear =' <=50K')>
--------------------------------------------------------------------
------------
<Adult(age='28', workclass=' Private', fnlwgt='338409',education=' B
achelors', educationNum='13',maritalStatus =' Married-civ-spouse',oc
cupation =' Prof-specialty',relationship =' Wife',race =' Black',sex
=' Female',capitalGain ='0',capitalLoss ='0',hoursperweek ='40',nati
veCountry =' Cuba',salperyear =' <=50K')>
--------------------------------------------------------------------
------------
<Adult(age='37', workclass=' Private', fnlwgt='284582',education=' M
asters', educationNum='14',maritalStatus =' Married-civ-spouse',occu
pation =' Exec-managerial',relationship =' Wife',race =' White',sex
=' Female',capitalGain ='0',capitalLoss ='0',hoursperweek ='40',nati
veCountry =' United-States',salperyear =' <=50K')>
--------------------------------------------------------------------
------------
```

2. Write two basic update queries

In [2]:

```
print("Records having  50 years aged  \n",'-'*80)
for adt in session.query(Adult).filter(Adult.age == 50):
    print(adt)
    print('-'*80)

# Updating The Hoursperweek for 50 years old
session.query(Adult).filter(Adult.age == 50).update({'hoursperweek' : 20})
session.commit()

print("After Updating hoursperweek to 20 and fetching records from DB \n",'-'*80
)
for adt in session.query(Adult).filter(Adult.age == 50):
    print(adt)
    print('-'*80)
```

```
************ Query 1 ***********

Records having  50 years aged
 --------------------------------------------------------------------------
-------------
<Adult(age='50', workclass=' Self-emp-not-inc', fnlwgt='83311',educa
tion=' Bachelors', educationNum='13',maritalStatus =' Married-civ-sp
ouse',occupation =' Exec-managerial',relationship =' Husband',race
=' White',sex =' Male',capitalGain ='0',capitalLoss ='0',hoursperwee
k ='13',nativeCountry =' United-States',salperyear =' <=50K')>
--------------------------------------------------------------------------
------------
After Updating hoursperweek to 20 and fetching records from DB
 --------------------------------------------------------------------------
-------------
<Adult(age='50', workclass=' Self-emp-not-inc', fnlwgt='83311',educa
tion=' Bachelors', educationNum='13',maritalStatus =' Married-civ-sp
ouse',occupation =' Exec-managerial',relationship =' Husband',race
=' White',sex =' Male',capitalGain ='0',capitalLoss ='0',hoursperwee
k ='20',nativeCountry =' United-States',salperyear =' <=50K')>
--------------------------------------------------------------------------
------------
```

In [3]:

```
print("Checking if record exists where workclass is not Private \n",'-'*80)
for adt in session.query(Adult).filter(Adult.workclass != ' Private'):
    print(adt)
    print('-'*80)

session.query(Adult).filter(Adult.workclass != ' Private').update({'workclass' :
 ' Private'})
session.commit()

print("After Updating Workclass to Private and fetching from DB \n",'-'*80)
for adt in session.query(Adult).all():
    print(adt)
    print('-'*80)
```

```
Checking if record exists where workclass is not Private
 ------------------------------------------------------------------
-------------
<Adult(age='50', workclass=' Self-emp-not-inc', fnlwgt='83311',educa
tion=' Bachelors', educationNum='13',maritalStatus =' Married-civ-sp
ouse',occupation =' Exec-managerial',relationship =' Husband',race
=' White',sex =' Male',capitalGain ='0',capitalLoss ='0',hoursperwee
k ='20',nativeCountry =' United-States',salperyear =' <=50K')>
------------------------------------------------------------------
------------
After Updating Workclass to Private and fetching from DB
 ------------------------------------------------------------------
-------------
<Adult(age='50', workclass=' Private', fnlwgt='83311',education=' Ba
chelors', educationNum='13',maritalStatus =' Married-civ-spouse',occ
upation =' Exec-managerial',relationship =' Husband',race =' White',
sex =' Male',capitalGain ='0',capitalLoss ='0',hoursperweek ='20',na
tiveCountry =' United-States',salperyear =' <=50K')>
------------------------------------------------------------------
------------
<Adult(age='38', workclass=' Private', fnlwgt='215646',education=' H
S-grad', educationNum='9',maritalStatus =' Divorced',occupation =' H
andlers-cleaners',relationship =' Not-in-family',race =' White',sex
=' Male',capitalGain ='0',capitalLoss ='0',hoursperweek ='40',native
Country =' United-States',salperyear =' <=50K')>
------------------------------------------------------------------
------------
<Adult(age='53', workclass=' Private', fnlwgt='234721',education=' 1
1th', educationNum='7',maritalStatus =' Married-civ-spouse',occupati
on =' Handlers-cleaners',relationship =' Husband',race =' Black',sex
=' Male',capitalGain ='0',capitalLoss ='0',hoursperweek ='40',native
Country =' United-States',salperyear =' <=50K')>
------------------------------------------------------------------
------------
<Adult(age='28', workclass=' Private', fnlwgt='338409',education=' B
achelors', educationNum='13',maritalStatus =' Married-civ-spouse',oc
cupation =' Prof-specialty',relationship =' Wife',race =' Black',sex
=' Female',capitalGain ='0',capitalLoss ='0',hoursperweek ='40',nati
veCountry =' Cuba',salperyear =' <=50K')>
------------------------------------------------------------------
------------
<Adult(age='37', workclass=' Private', fnlwgt='284582',education=' M
asters', educationNum='14',maritalStatus =' Married-civ-spouse',occu
pation =' Exec-managerial',relationship =' Wife',race =' White',sex
=' Female',capitalGain ='0',capitalLoss ='0',hoursperweek ='40',nati
veCountry =' United-States',salperyear =' <=50K')>
------------------------------------------------------------------
------------
```

3.Write two delete queries

In [4]:

```
print("Checking if record exists where workclass is not Private \n",'-'*80)
for adt in session.query(Adult).filter(Adult.workclass != ' Private'):
    print(adt)
    print('-'*80)

session.query(Adult).filter(Adult.workclass != ' Private').delete()
session.commit()

print("After Deleting and fetching from DB , check if  record now exists \n",'-'
*80)
for adt in session.query(Adult).all():
    print(adt)
    print('-'*80)
```

```
Checking if record exists where workclass is not Private
 --------------------------------------------------------------------------------
After Deleting and fetching from DB , check if  record now exists
 --------------------------------------------------------------------------------
<Adult(age='50', workclass=' Private', fnlwgt='83311',education=' Ba
chelors', educationNum='13',maritalStatus =' Married-civ-spouse',occ
upation =' Exec-managerial',relationship =' Husband',race =' White',
sex =' Male',capitalGain ='0',capitalLoss ='0',hoursperweek ='20',na
tiveCountry =' United-States',salperyear =' <=50K')>
--------------------------------------------------------------------------------
<Adult(age='38', workclass=' Private', fnlwgt='215646',education=' H
S-grad', educationNum='9',maritalStatus =' Divorced',occupation =' H
andlers-cleaners',relationship =' Not-in-family',race =' White',sex
=' Male',capitalGain ='0',capitalLoss ='0',hoursperweek ='40',native
Country =' United-States',salperyear =' <=50K')>
--------------------------------------------------------------------------------
<Adult(age='53', workclass=' Private', fnlwgt='234721',education=' 1
1th', educationNum='7',maritalStatus =' Married-civ-spouse',occupati
on =' Handlers-cleaners',relationship =' Husband',race =' Black',sex
=' Male',capitalGain ='0',capitalLoss ='0',hoursperweek ='40',native
Country =' United-States',salperyear =' <=50K')>
--------------------------------------------------------------------------------
<Adult(age='28', workclass=' Private', fnlwgt='338409',education=' B
achelors', educationNum='13',maritalStatus =' Married-civ-spouse',oc
cupation =' Prof-specialty',relationship =' Wife',race =' Black',sex
=' Female',capitalGain ='0',capitalLoss ='0',hoursperweek ='40',nati
veCountry =' Cuba',salperyear =' <=50K')>
--------------------------------------------------------------------------------
<Adult(age='37', workclass=' Private', fnlwgt='284582',education=' M
asters', educationNum='14',maritalStatus =' Married-civ-spouse',occu
pation =' Exec-managerial',relationship =' Wife',race =' White',sex
=' Female',capitalGain ='0',capitalLoss ='0',hoursperweek ='40',nati
veCountry =' United-States',salperyear =' <=50K')>
--------------------------------------------------------------------------------
```

In [6]:

```
print("Checking if record exists where education Number is other than 13 \n",'-'
*80)
for adt in session.query(Adult).filter(Adult.educationNum == 13).all():
    print(adt)
    print('-'*80)

session.query(Adult).filter(Adult.educationNum == 13).delete()
session.commit()

print("After Deleting the record , fetching from DB , if filtered record exists
\n",'-'*80)
for adt in session.query(Adult).all():
    print(adt)
    print('-'*80)
```

```
Checking if record exists where education Number is other than 13
 --------------------------------------------------------------------------
-------------
After Deleting the record , fetching from DB , if filtered record ex
ists
 --------------------------------------------------------------------------
-------------
<Adult(age='38', workclass=' Private', fnlwgt='215646',education=' H
S-grad', educationNum='9',maritalStatus =' Divorced',occupation =' H
andlers-cleaners',relationship =' Not-in-family',race =' White',sex
=' Male',capitalGain ='0',capitalLoss ='0',hoursperweek ='40',native
Country =' United-States',salperyear =' <=50K')>
--------------------------------------------------------------------------
------------
<Adult(age='53', workclass=' Private', fnlwgt='234721',education=' 1
1th', educationNum='7',maritalStatus =' Married-civ-spouse',occupati
on =' Handlers-cleaners',relationship =' Husband',race =' Black',sex
=' Male',capitalGain ='0',capitalLoss ='0',hoursperweek ='40',native
Country =' United-States',salperyear =' <=50K')>
--------------------------------------------------------------------------
------------
<Adult(age='37', workclass=' Private', fnlwgt='284582',education=' M
asters', educationNum='14',maritalStatus =' Married-civ-spouse',occu
pation =' Exec-managerial',relationship =' Wife',race =' White',sex
=' Female',capitalGain ='0',capitalLoss ='0',hoursperweek ='40',nati
veCountry =' United-States',salperyear =' <=50K')>
--------------------------------------------------------------------------
------------
```

4. Write two filter queries

In [7]:

```
print("To See those who have studied till masters  \n",'-'*80)
for adt in session.query(Adult).filter(Adult.education.like('%Masters%')):
    print(adt)
    print('-'*80)
```

```
To See those who have studied till masters
 --------------------------------------------------------------------------
-------------
<Adult(age='37', workclass=' Private', fnlwgt='284582',education=' M
asters', educationNum='14',maritalStatus =' Married-civ-spouse',occu
pation =' Exec-managerial',relationship =' Wife',race =' White',sex
=' Female',capitalGain ='0',capitalLoss ='0',hoursperweek ='40',nati
veCountry =' United-States',salperyear =' <=50K')>
--------------------------------------------------------------------------
------------
```

In [8]:

```
print("To See how many people have studied masters \n",'-'*80)
#print(session.query(Adult).filter(Adult.nativeCountry.like('%United-States%')).
group_by(Adult.sex).count(Adult.sex))
print(session.query(Adult).filter(Adult.education.like('%Masters%')).count())
print('-'*80)
```

```
To See how many people have studied masters
 --------------------------------------------------------------------------
-------------
1
--------------------------------------------------------------------------
------------
```

5. Write two function queries

In [10]:

```
from sqlalchemy import func
print("To show the gender distribution for country  \n",'-'*80)
print(session.query(Adult.nativeCountry,Adult.sex,func.count('*')).group_by(Adul
t.nativeCountry,Adult.sex).all())
print('-'*80)
```

```
To show the gender distribution for country
 --------------------------------------------------------------------------
-------------
[(' United-States', ' Female', 1), (' United-States', ' Male', 2)]
--------------------------------------------------------------------------
------------
```

In [11]:

```
print("To See how many people are married \n",'-'*80)
print(session.query(Adult.maritalStatus,func.count('*')).group_by(Adult.maritalS
tatus).all())
print('-'*80)
```

```
To See how many people are married
 --------------------------------------------------------------------------
-------------
[(' Divorced', 1), (' Married-civ-spouse', 2)]
--------------------------------------------------------------------------
------------
```

In [ ]:

```

```