

Read the following data set: <https://archive.ics.uci.edu/ml/machine-learning-databases/adult/> (<https://archive.ics.uci.edu/ml/machine-learning-databases/adult/>). Rename the columns as per the description from this file: <https://archive.ics.uci.edu/ml/machine-learning-databases/adult/adult.names> (<https://archive.ics.uci.edu/ml/machine-learning-databases/adult/adult.names>) Task: Create a sql db from adult dataset and name it sqladb

1. Select 10 records from the adult sqladb
2. Show me the average hours per week of all men who are working in private sector
3. Show me the frequency table for education, occupation and relationship, separately
4. Are there any people who are married, working in private sector and having a masters degree
5. What is the average, minimum and maximum age group for people working in different sectors
6. Calculate age distribution by country
7. Compute a new column as 'Net-Capital-Gain' from the two columns 'capital-gain' and 'capital-loss'

```
In [1]: # Import all the packages
import pandas as pd
import sqlite3 as sqllite
```

```
In [2]: # Read the data from the given URL and see the top 5 records
df = pd.read_csv('https://archive.ics.uci.edu/ml/machine-learning-databases/adult/
df.head(5)
```

Out[2]:

	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male	2174	0	40
0	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male	0	0	13
1	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male	0	0	40
2	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male	0	0	40
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

```
In [3]: # Rename the columns as per the description.
df.columns = ['age', 'workclass', 'fnlwgt', 'education', 'education-num', 'marita
df = df.infer_objects()

#Strip the Object columns.
df_obj = df.select_dtypes(['object'])
df[df_obj.columns] = df_obj.apply(lambda x: x.str.strip())
```

```
In [4]: # Print after rename.
df.head(5)
```

Out[4]:

	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 [5]: # Assign the Database and Table Names to Local variables.
db_name = 'sqladb.db'
tbl_name = 'adult_names'
```

```
In [6]: # Open Connection to SQLite and insert the data to SQL Lite Table.
# Create a sql db from adult dataset and name it sqladb
con = sqlite3.connect(db_name)
cur = con.cursor()

wildcards = ','.join(['?'] * len(df.columns))
data = [tuple(x) for x in df.values]

cur.execute("drop table if exists %s" % tbl_name)

col_str = "'" + ','.join(df.columns) + "'"
cur.execute("create table %s (%s)" % (tbl_name, col_str))

cur.executemany("insert into %s values(%s)" % (tbl_name, wildcards), data)
con.commit()
```

```
In [7]: # 1. Select 10 records from the adult sqladb
df = pd.read_sql_query("SELECT * FROM adult_names LIMIT 10", con)
df
```

Out[7]:

	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
5	49	Private	160187	9th	5	Married-spouse-absent	Other-service	Not-in-family	Black	Female
6	52	Self-emp-not-inc	209642	HS-grad	9	Married-civ-spouse	Exec-managerial	Husband	White	Male
7	31	Private	45781	Masters	14	Never-married	Prof-specialty	Not-in-family	White	Female
8	42	Private	159449	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male
9	37	Private	280464	Some-college	10	Married-civ-spouse	Exec-managerial	Husband	Black	Male

```
In [8]: # 2. Show me the average hours per week of all men who are working in private sector
average = pd.read_sql_query("SELECT CAST ([hours-per-week] as int) as Hours FROM adult_names WHERE sex = 'M' AND workclass = 'Private'")
print(average.mean())
```

```
Hours    42.221226
dtype: float64
```

In [9]: *# 3. Show me the frequency table for education, occupation and relationship, separately*

```
education = pd.read_sql_query("SELECT education, count(education) as Frequency FROM education")
print(education)
```

	education	Frequency
0	10th	933
1	11th	1175
2	12th	433
3	1st-4th	168
4	5th-6th	333
5	7th-8th	646
6	9th	514
7	Assoc-acdm	1067
8	Assoc-voc	1382
9	Bachelors	5354
10	Doctorate	413
11	HS-grad	10501
12	Masters	1723
13	Preschool	51
14	Prof-school	576
15	Some-college	7291

In [10]:

```
occupation = pd.read_sql_query("SELECT occupation, COUNT(occupation) as Frequency FROM occupation")
print(occupation)
```

	occupation	Frequency
0	?	1843
1	Adm-clerical	3769
2	Armed-Forces	9
3	Craft-repair	4099
4	Exec-managerial	4066
5	Farming-fishing	994
6	Handlers-cleaners	1370
7	Machine-op-inspct	2002
8	Other-service	3295
9	Priv-house-serv	149
10	Prof-specialty	4140
11	Protective-serv	649
12	Sales	3650
13	Tech-support	928
14	Transport-moving	1597

In [11]:

```
relationship = pd.read_sql_query("SELECT relationship, COUNT(relationship) as Frequency FROM relationship")
print(relationship)
```

	relationship	Frequency
0	Husband	13193
1	Not-in-family	8304
2	Other-relative	981
3	Own-child	5068
4	Unmarried	3446
5	Wife	1568

In [12]: *#4. Are there any people who are married, working in private sector and having a*  
 people = pd.read\_sql\_query("SELECT Count(\*) as Count FROM adult\_names where educa  
 people

Out[12]:

	Count
0	540

In [13]: *#5. What is the average, minimum and maximum age group for people working in differ*  
 agegroup = pd.read\_sql\_query("SELECT workclass, avg(age) as Average, min(age) as l  
 agegroup

Out[13]:

	workclass	Average	Min	Max
0	?	40.960240	17	90
1	Federal-gov	42.590625	17	90
2	Local-gov	41.751075	17	90
3	Never-worked	20.571429	17	30
4	Private	36.797585	17	90
5	Self-emp-inc	46.017025	17	84
6	Self-emp-not-inc	44.969697	17	90
7	State-gov	39.436392	17	81
8	Without-pay	47.785714	19	72

In [14]: *#6. Calculate age distribution by country*  
 dfcountry = pd.read\_sql\_query("SELECT [native-country], age, count(\*) as Distribution  
 dfcountry

Out[14]:

	native-country	age	Distribution
0	?	17	2
1	?	18	8
2	?	19	5
3	?	20	10
4	?	21	11
5	?	22	12
6	?	23	6
7	?	24	14
8	?	25	11
9	?	26	18
10	?	27	15
11	?	28	19
12	?	29	12
13	?	30	19
14	?	31	18
15	?	32	17
16	?	33	13
17	?	34	24
18	?	35	18
19	?	36	23
20	?	37	22
21	?	38	20
22	?	39	19
23	?	40	12
24	?	41	22
25	?	42	24
26	?	43	14
27	?	44	10
28	?	45	17
29	?	46	15
...	...	...	...
1251	Vietnam	37	2

	native-country	age	Distribution
1252	Vietnam	38	1
1253	Vietnam	40	1
1254	Vietnam	41	1
1255	Vietnam	43	2
1256	Vietnam	44	3
1257	Vietnam	45	3
1258	Vietnam	46	1
1259	Vietnam	48	1
1260	Vietnam	50	1
1261	Vietnam	51	1
1262	Vietnam	52	1
1263	Vietnam	53	1
1264	Vietnam	54	1
1265	Vietnam	63	1
1266	Vietnam	70	1
1267	Vietnam	73	2
1268	Yugoslavia	20	1
1269	Yugoslavia	22	1
1270	Yugoslavia	25	1
1271	Yugoslavia	29	1
1272	Yugoslavia	31	1
1273	Yugoslavia	35	2
1274	Yugoslavia	36	1
1275	Yugoslavia	40	1
1276	Yugoslavia	41	2
1277	Yugoslavia	43	1
1278	Yugoslavia	45	1
1279	Yugoslavia	56	2
1280	Yugoslavia	66	1

1281 rows × 3 columns

In [15]: *#7 Compute a new column as 'Net-Capital-Gain' from the two columns 'capital-gain'*

```
df = pd.read_sql_query("SELECT * FROM adult_names", con)
df['Net-Capital-Gain'] = df['capital-gain'] - df['capital-loss']
df
```

Out[15]:

	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
4	37	Private	284582	Masters	14	Married-civ-spouse	Exec-managerial	Wife	White