Ex1 - Getting and Knowing your Data

Step 1. Import the necessary libraries

```
In [1]: import pandas as pd
```

Step 2. Read from data.csv and assign it to a variable called users and use the 'user_id' as index

```
In [2]: users = pd.read_csv('data1.csv', index_col='user_id')
```

Step 3. See the first 25 entries

```
In [3]: users.head(25)
```

Out[3]:

	age	gender	occupation	zip_code
user_id				
1	24	M	technician	85711
2	53	F	other	94043
3	23	M	writer	32067
4	24	M	technician	43537
5	33	F	other	15213
6	42	M	executive	98101
7	57	M	administrator	91344
8	36	M	administrator	5201
9	29	M	student	1002
10	53	M	lawyer	90703
11	39	F	other	30329
12	28	F	other	6405
13	47	M	educator	29206
14	45	M	scientist	55106
15	49	F	educator	97301
16	21	M	entertainment	10309
17	30	M	programmer	6355
18	35	F	other	37212
19	40	M	librarian	2138
20	42	F	homemaker	95660
21	26	M	writer	30068
22	25	M	writer	40206
23	30	F	artist	48197
24	21	F	artist	94533
25	39	M	engineer	55107

Step 4. See the last 10 entries

In [4]:

```
users.tail(10)
```

Out[4]:

	age	gender	occupation	zip_code
user_id				
934	61	M	engineer	22902
935	42	M	doctor	66221
936	24	M	other	32789
937	48	M	educator	98072
938	38	F	technician	55038
939	26	F	student	33319
940	32	M	administrator	2215
941	20	M	student	97229
942	48	F	librarian	78209
943	22	M	student	77841

Step 5. What is the number of rows in the dataset?

In [5]:

```
users.shape[0]
```

Out[5]:

```
943
```

Step 6. What is the number of columns in the dataset?

In [6]:

```
users.shape[1]
```

Out[6]:

```
4
```

Step 7. Print the name of all the columns.

In [7]:

```
users.columns
```

Out[7]:

```
Index(['age', 'gender', 'occupation', 'zip_code'], dtype='object')
```

Step 8. Print only the occupation column

In [8]:

```
users.occupation

#or

users['occupation']
```

Out[8]:
```
user_id
1         technician
2              other
3             writer
4         technician
5              other
           ...
939          student
940    administrator
941          student
942        librarian
943          student
Name: occupation, Length: 943, dtype: object
```

Step 9. How many different occupations are in this dataset?

In [9]:
```
users.occupation.value_counts()
```

Out[9]:
```
student          196
other            105
educator          95
administrator     79
engineer          67
programmer        66
librarian         51
writer            45
executive         32
scientist         31
artist            28
technician        27
marketing         26
entertainment     18
healthcare        16
retired           14
lawyer            12
salesman          12
none               9
doctor             7
homemaker          7
Name: occupation, dtype: int64
```

In [10]:
```
#value_counts() which returns the count of unique elements
users.occupation.value_counts().count()
# or users.occupation.nunique()
```

Out[10]:
```
21
```

Step 10. What is the most frequent occupation?

In [11]:
```
#Because "most" is asked
users.occupation.value_counts().head(1)
```

Out[11]:
```
student    196
Name: occupation, dtype: int64
```

Step 11. Summarize the DataFrame.

In [12]:
```
users.describe() #Notice: by default, only the numeric columns are returned.
users.info()
```

Out[12]:

	age
count	943.000000
mean	34.051962
std	12.192740
min	7.000000
25%	25.000000
50%	31.000000
75%	43.000000
max	73.000000

Step 12. Summarize only the occupation column

In [13]:

```
users.occupation.describe()
```

Out[13]:

```
count           943
unique           21
top         student
freq            196
Name: occupation, dtype: object
```

Step 13. What is the mean age of users?

In [14]:

```
users.age.mean()
```

Out[14]:

```
34.05196182396607
```

Step 14. What is the age with least occurrence?

In [15]:

```
users.age.value_counts().tail() #7, 10, 11, 66 and 73 years -> only 1 occurr
```

Out[15]:

```
7      1
66     1
10     1
11     1
73     1
Name: age, dtype: int64
```