## Ex3 - Getting and Knowing your Data

This time we are going to pull data directly from the internet. Special thanks to: <a href="https://github.com/justmarkham">https://github.com/justmarkham</a> for sharing the dataset and materials.

Step 1. Import the necessary libraries

import pandas as pd

data = pd.read\_csv('https://raw.githubusercontent.com/thieu1995/csv-files/main/data/pandas/u.user', sep='|')

Step 3. Assign it to a variable called users and use the 'user\_id' as index

users = data.set\_index('user\_id')

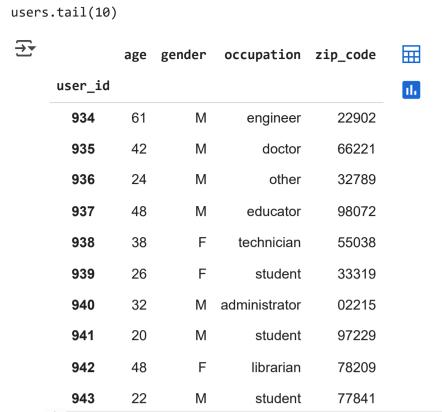
Step 4. See the first 25 entries

users.head(25)

|         | age | gender | occupation    | zip_code |
|---------|-----|--------|---------------|----------|
| user_id |     |        |               |          |
| 1       | 24  | М      | technician    | 85711    |
| 2       | 53  | F      | other         | 94043    |
| 3       | 23  | М      | writer        | 32067    |
| 4       | 24  | М      | technician    | 43537    |
| 5       | 33  | F      | other         | 15213    |
| 6       | 42  | М      | executive     | 98101    |
| 7       | 57  | М      | administrator | 91344    |
| 8       | 36  | М      | administrator | 05201    |
| 9       | 29  | М      | student       | 01002    |
| 10      | 53  | М      | lawyer        | 90703    |
| 11      | 39  | F      | other         | 30329    |
| 12      | 28  | F      | other         | 06405    |
| 13      | 47  | М      | educator      | 29206    |
| 14      | 45  | М      | scientist     | 55106    |
| 15      | 49  | F      | educator      | 97301    |
| 16      | 21  | М      | entertainment | 10309    |
| 17      | 30  | М      | programmer    | 06355    |
| 18      | 35  | F      | other         | 37212    |
| 19      | 40  | М      | librarian     | 02138    |
| 20      | 42  | F      | homemaker     | 95660    |
| 21      | 26  | М      | writer        | 30068    |
| 22      | 25  | М      | writer        | 40206    |
| 23      | 30  | F      | artist        | 48197    |
| 24      | 21  | F      | artist        | 94533    |
| 25      | 39  | М      | engineer      | 55107    |

Next steps: Generate code with users View recommended plots New interactive sheet

## Step 5. See the last 10 entries



```
users.shape[0]

→ 943
```

Step 7. What is the number of columns in the dataset?

```
users.shape[1]

→ 4
```

users.index

Step 8. Print the name of all the columns.

```
users.columns

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

→ Step 9. How is the dataset indexed?

```
☐ Index([ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, ... 934, 935, 936, 937, 938, 939, 940, 941, 942, 943], dtype='int64', name='user_id', length=943)
```

## Step 11. Print only the occupation column

users['occupation']  $\overline{\mathbf{x}}$ occupation user\_id 1 technician 2 other 3 writer technician 5 other 939 student 940 administrator 941 student 942 librarian 943 student 943 rows × 1 columns dtvpe: object

→ Step 12. How many different occupations are in this dataset?

```
users['occupation'].nunique()

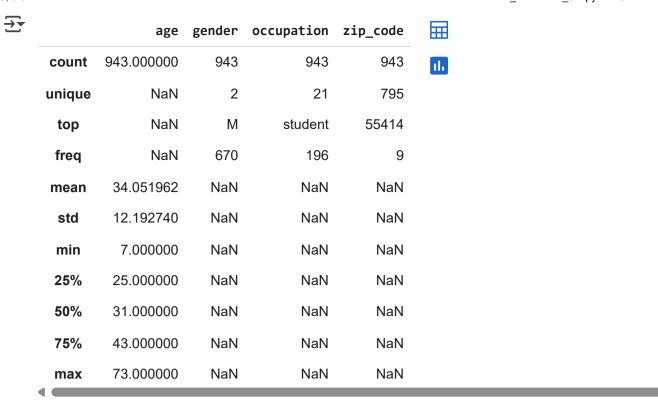
→ 21
```

Step 14. Summarize the DataFrame.

```
users.info()
    <class 'pandas.core.frame.DataFrame'>
     Index: 943 entries, 1 to 943
     Data columns (total 4 columns):
                  Non-Null Count Dtype
     # Column
     0
                     943 non-null
                                     int64
         age
         gender
                     943 non-null
                                     object
         occupation 943 non-null
                                     object
         zip_code
                     943 non-null
                                     object
     dtypes: int64(1), object(3)
     memory usage: 36.8+ KB
```

→ Step 15. Summarize all the columns

```
users.describe(include='all')
```



## → Step 16. Summarize only the occupation column

users['occupation'].value\_counts()

| s[ occupation ] | ·varac_ |
|-----------------|---------|
|                 | count   |
| occupation      |         |
| 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       |
| homemaker       | 7       |
| doctor          | 7       |
| dtvne: int64    |         |
|                 |         |

```
float(users['age'].mean())

34.05196182396607
```

Step 18. What is the age with least occurrence?

int(users['age'].value\_counts().idxmin())



Start coding or generate with AI.