1 Importing library and data

In [2]:
```
import pandas as pd
us = pd.read_csv('users.csv')
```

2 Check the data type

In [3]:
```
us.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 495 entries, 0 to 494
Data columns (total 7 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   USERID      495 non-null    object
 1   ACTIVE      495 non-null    bool  
 2   CREATEDATE  495 non-null    object
 3   LASTLOGIN   433 non-null    object
 4   ROLE        495 non-null    object
 5   SOURCE      447 non-null    object
 6   STATE       439 non-null    object
dtypes: bool(1), object(6)
memory usage: 23.8+ KB
```

In [4]:
```
us.head()
```

Out[4]:

	USERID	ACTIVE	CREATEDATE	LASTLOGIN	ROLE	SOURCE	STATE
0	5ff1e194b6a9d73a3a9f1052	True	2021-01-03 10:24:04	2021-01-03 10:25:37	consumer	Email	WI
1	5ff1e194b6a9d73a3a9f1052	True	2021-01-03 10:24:04	2021-01-03 10:25:37	consumer	Email	WI
2	5ff1e194b6a9d73a3a9f1052	True	2021-01-03 10:24:04	2021-01-03 10:25:37	consumer	Email	WI
3	5ff1e1eacfcf6c399c274ae6	True	2021-01-03 10:25:30	2021-01-03 10:25:30	consumer	Email	WI
4	5ff1e194b6a9d73a3a9f1052	True	2021-01-03 10:24:04	2021-01-03 10:25:37	consumer	Email	WI

In [5]:
```
# Check the type of each feature in the trans table
print(us.dtypes)
```

```
USERID        object
ACTIVE          bool
CREATEDATE    object
LASTLOGIN     object
ROLE          object
SOURCE        object
STATE         object
dtype: object
```

3 The number of null value of each column

In [6]:
```
# Count the number of null value of each column
num_missing = us.isnull().sum()
print(num_missing)
```

```
USERID         0
ACTIVE         0
CREATEDATE     0
LASTLOGIN     62
ROLE           0
SOURCE        48
STATE         56
dtype: int64
```

In [8]:
```
# Calculate the percentage of null values in each column
null_percentages = (us.isnull().sum() / len(us)) * 100

print(null_percentages)
```

```
USERID         0.000000
ACTIVE         0.000000
CREATEDATE     0.000000
LASTLOGIN     12.525253
ROLE           0.000000
SOURCE         9.696970
STATE         11.313131
dtype: float64
```

According to the result above, we can find that the percentage of missing value in the dataset is relativly small. Therefore, it might not affect analysis results.

4 Percentage of duplicate value in 'USERID' column

In [11]:
```
# Check the number of duplicate value in 'userId' column
duplicate_var = us['USERID'].duplicated().sum()
print(duplicate_var)
```

```
283
```

In [15]:
```
# Calculate the percentage of duplicate value in 'userId' column
duplicates_num = (us['USERID'].count() - us['USERID'].nunique())
percentage_dup = (duplicates_num / us['USERID'].count()) * 100
print("The percentage of dupplicate value in 'USERID' column is: ",
                                                    percentage_dup)
```

```
The percentage of dupplicate value in 'USERID' column is:  57.17171717171718
```