Data Wrangling

In [ ]:
```
import numpy as np
import pandas as pd
import seaborn as sns
```

In [ ]:
```
kashti = sns.load_dataset('titanic')
k1 = kashti
k2 = kashti
k3 = kashti
k1.shape
# k4 = kashti
# k4
```

Out[ ]:
```
(891, 15)
```

Dealing with Missing Values

- In a data set values are missing either, N/A, NaN, 0 or empty cell.

In [ ]:
```
# simple operation on column (Math Operatore)
k1 = kashti
(k1['age'] + 1).head()
```

Out[ ]:
```
0    23.0
1    39.0
2    27.0
3    36.0
4    36.0
Name: age, dtype: float64
```

In [ ]:
```
# give the shape of the date set
k1.shape
# it find the null value in all data set
k1.isna().sum()
```

Out[ ]:
```
survived         0
pclass           0
sex              0
age            177
sibsp            0
parch            0
fare             0
embarked         2
class            0
who              0
adult_male       0
deck           688
embark_town      2
alive            0
alone            0
dtype: int64
```

In [ ]:
```
# isnull is same like isna
k1.isnull().sum()
```

Out[ ]:
```
survived         0
pclass           0
sex              0
age            177
sibsp            0
parch            0
fare             0
embarked         2
class            0
who              0
adult_male       0
deck           688
embark_town      2
alive            0
alone            0
dtype: int64
```

In [ ]:
```
# Use dropna to drop all null values
# k3 = k1.dropna()
# k3.shape
# k3

# 'dropna' will drop all the null values from null 'deck' column
# and also reduce other null values in the data set
k1.dropna(subset=['deck'], axis=0, inplace=True)
k1.shape
k1.isnull().sum()
```

Out[ ]:
```
survived         0
pclass           0
sex              0
age             19
sibsp            0
parch            0
fare             0
embarked         2
class            0
who              0
adult_male       0
deck             0
embark_town      2
alive            0
alone            0
dtype: int64
```

In [ ]:
```
k1.describe()
k1.shape
```

Out[ ]:
```
(203, 15)
```

In [ ]:
```
k1.dropna(subset=['age'], axis=0, inplace=True)
```

In [ ]:
```
k1.isnull().sum()
k1
```

Out[ ]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	d
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	d
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	
6	0	1	male	54.0	0	0	51.8625	S	First	man	True	
10	1	3	female	4.0	1	1	16.7000	S	Third	child	False	
11	1	1	female	58.0	0	0	26.5500	S	First	woman	False	
...	...	...	...	...	...	...	...	...	...	...	...	
871	1	1	female	47.0	1	1	52.5542	S	First	woman	False	
872	0	1	male	33.0	0	0	5.0000	S	First	man	True	
879	1	1	female	56.0	0	1	83.1583	C	First	woman	False	
887	1	1	female	19.0	0	0	30.0000	S	First	woman	False	
889	1	1	male	26.0	0	0	30.0000	C	First	man	True	

184 rows × 15 columns

In [ ]:

Replace Missing values with average of that column

In [ ]:
```
# finding average (mean) of column
mean = k3['age'].mean()  # 29.69
mean
```

Out[ ]:
```
35.77945652173913
```

In [ ]:
```
# now replace all the Nan values in a column with this mean
k3['age'] = k3['age'].replace(np.nan, mean)

# After replacing all null values with mean of colum
k3.isnull().sum()
```

Out[ ]:
```
survived         0
pclass           0
sex              0
age              0
sibsp            0
parch            0
fare             0
embarked         2
class            0
who              0
adult_male       0
deck           688
embark_town      2
alive            0
alone            0
dtype: int64
```

In [ ]:
```
k3.isnull().sum()
```

Out[ ]:
```
survived          0
pclass            0
sex               0
age               0
sibsp             0
parch             0
fare              0
embarked          2
class             0
who               0
adult_male        0
deck            688
embark_town       2
alive             0
alone             0
dtype: int64
```

Assignment

Remove deck and embark_town

In [ ]:
```
k5 = sns.load_dataset('titanic')

k5.head()
# Remove deck colums
k5 = k5.drop('deck', axis = 1)
k5.head()
```

Out[ ]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	emb
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	Sou
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	C
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	Sou
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	Sou
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	Sou

In [ ]:
```
k5.drop('embark_town', axis=1)
k5.head()
```

Out[ ]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	emb
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	Sou
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	C
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	Sou
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	Sou
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	Sou

In [ ]: