In [1]:

```
import numpy as np
import pandas as pd
```

In [2]:

```
df= pd.read_csv("claimants sample.csv")
df
```

Out[2]:

	CASENUM	SEX	INSUR	SEATBELT	AGE	LOSS	ATTORNEY
0	5	0	1.0	0	50.0	34.940	0
1	3	1	0.0	0	18.0	0.891	1
2	66	0	1.0	0	5.0	0.330	1
3	70	1	1.0	1	31.0	0.037	0
4	96	0	1.0	0	30.0	NaN	1
5	97	1	1.0	0	35.0	0.309	0
6	10	0	NaN	0	9.0	3.538	0
7	36	1	NaN	0	34.0	4.881	0
8	51	1	1.0	0	60.0	0.874	1
9	55	1	1.0	0	NaN	0.350	1

In [3]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 7 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   CASENUM   10 non-null     int64  
 1   SEX       10 non-null     int64  
 2   INSUR     8 non-null      float64
 3   SEATBELT  10 non-null     int64  
 4   AGE       9 non-null      float64
 5   LOSS      9 non-null      float64
 6   ATTORNEY  10 non-null     int64  
dtypes: float64(3), int64(4)
memory usage: 688.0 bytes
```

Check for missing values in each column

In [4]:

```
df.isnull().sum()
```

Out[4]:

```
CASENUM     0
SEX         0
INSUR       2
SEATBELT    0
AGE         1
LOSS        1
ATTORNEY    0
dtype: int64
```

- **Empty cells can potentially give you a wrong result when you analyze data.**

Dealing the Missing Values

1. Remove the rows that contain missing values

In [5]:

```
df1 = df.dropna()
```

In [6]:

```
df1
```

Out[6]:

	CASENUM	SEX	INSUR	SEATBELT	AGE	LOSS	ATTORNEY
0	5	0	1.0	0	50.0	34.940	0
1	3	1	0.0	0	18.0	0.891	1
2	66	0	1.0	0	5.0	0.330	1
3	70	1	1.0	1	31.0	0.037	0
5	97	1	1.0	0	35.0	0.309	0
8	51	1	1.0	0	60.0	0.874	1

- **We should not remove >5% of original data.**
- **This is usually OK, for very big data sets, and removing a few rows will not have a big impact on the result.**

2. Replace the nan values

```
- Mean
- Median
- Mode
- fill with some value
```

- Continous Variables ---> AGE,LOSS ---> Replace with either Mean or Median for contionus data
- Discrete Variables ---> INSUR ---> Mode is used for discrete data

1. fillna() using pandas

In [7]:

```
df["AGE"].mean()
```

Out[7]:

```
30.22222222222222
```

In [8]:

```
df["AGE"].fillna(30.222, inplace=True)
df
```

Out[8]:

	CASENUM	SEX	INSUR	SEATBELT	AGE	LOSS	ATTORNEY
0	5	0	1.0	0	50.000	34.940	0
1	3	1	0.0	0	18.000	0.891	1
2	66	0	1.0	0	5.000	0.330	1
3	70	1	1.0	1	31.000	0.037	0
4	96	0	1.0	0	30.000	NaN	1
5	97	1	1.0	0	35.000	0.309	0
6	10	0	NaN	0	9.000	3.538	0
7	36	1	NaN	0	34.000	4.881	0
8	51	1	1.0	0	60.000	0.874	1
9	55	1	1.0	0	30.222	0.350	1

or

In [9]:

```
df['AGE'].fillna(df["AGE"].mean(),inplace=True)
df
```

Out[9]:

	CASENUM	SEX	INSUR	SEATBELT	AGE	LOSS	ATTORNEY
0	5	0	1.0	0	50.000	34.940	0
1	3	1	0.0	0	18.000	0.891	1
2	66	0	1.0	0	5.000	0.330	1
3	70	1	1.0	1	31.000	0.037	0
4	96	0	1.0	0	30.000	NaN	1
5	97	1	1.0	0	35.000	0.309	0
6	10	0	NaN	0	9.000	3.538	0
7	36	1	NaN	0	34.000	4.881	0
8	51	1	1.0	0	60.000	0.874	1
9	55	1	1.0	0	30.222	0.350	1

In [10]:

```python
df['LOSS'].fillna(df["LOSS"].median(),inplace=True)
df
```

Out[10]:

	CASENUM	SEX	INSUR	SEATBELT	AGE	LOSS	ATTORNEY
0	5	0	1.0	0	50.000	34.940	0
1	3	1	0.0	0	18.000	0.891	1
2	66	0	1.0	0	5.000	0.330	1
3	70	1	1.0	1	31.000	0.037	0
4	96	0	1.0	0	30.000	0.874	1
5	97	1	1.0	0	35.000	0.309	0
6	10	0	NaN	0	9.000	3.538	0
7	36	1	NaN	0	34.000	4.881	0
8	51	1	1.0	0	60.000	0.874	1
9	55	1	1.0	0	30.222	0.350	1

In [11]:

```python
mode = df["INSUR"].mode()[0]

df['INSUR'].fillna(mode,inplace=True)
df
```

Out[11]:

	CASENUM	SEX	INSUR	SEATBELT	AGE	LOSS	ATTORNEY
0	5	0	1.0	0	50.000	34.940	0
1	3	1	0.0	0	18.000	0.891	1
2	66	0	1.0	0	5.000	0.330	1
3	70	1	1.0	1	31.000	0.037	0
4	96	0	1.0	0	30.000	0.874	1
5	97	1	1.0	0	35.000	0.309	0
6	10	0	1.0	0	9.000	3.538	0
7	36	1	1.0	0	34.000	4.881	0
8	51	1	1.0	0	60.000	0.874	1
9	55	1	1.0	0	30.222	0.350	1

SimpleImputer() using SKlearn

In [12]:

```python
df= pd.read_csv("claimants sample.csv")
df
```

Out[12]:

	CASENUM	SEX	INSUR	SEATBELT	AGE	LOSS	ATTORNEY
0	5	0	1.0	0	50.0	34.940	0
1	3	1	0.0	0	18.0	0.891	1
2	66	0	1.0	0	5.0	0.330	1
3	70	1	1.0	1	31.0	0.037	0
4	96	0	1.0	0	30.0	NaN	1
5	97	1	1.0	0	35.0	0.309	0
6	10	0	NaN	0	9.0	3.538	0
7	36	1	NaN	0	34.0	4.881	0
8	51	1	1.0	0	60.0	0.874	1
9	55	1	1.0	0	NaN	0.350	1

In [13]:

```python
from sklearn.impute import SimpleImputer
```

In [14]:

```python
mean_imputer = SimpleImputer(strategy='mean')

df["AGE"] = mean_imputer.fit_transform(df[["AGE"]])

df
```

Out[14]:

	CASENUM	SEX	INSUR	SEATBELT	AGE	LOSS	ATTORNEY
0	5	0	1.0	0	50.000000	34.940	0
1	3	1	0.0	0	18.000000	0.891	1
2	66	0	1.0	0	5.000000	0.330	1
3	70	1	1.0	1	31.000000	0.037	0
4	96	0	1.0	0	30.000000	NaN	1
5	97	1	1.0	0	35.000000	0.309	0
6	10	0	NaN	0	9.000000	3.538	0
7	36	1	NaN	0	34.000000	4.881	0
8	51	1	1.0	0	60.000000	0.874	1
9	55	1	1.0	0	30.222222	0.350	1

In [15]:

```
median_imputer = SimpleImputer(strategy='median')
df["LOSS"] = median_imputer.fit_transform(df[["LOSS"]])
df
```

Out[15]:

	CASENUM	SEX	INSUR	SEATBELT	AGE	LOSS	ATTORNEY
0	5	0	1.0	0	50.000000	34.940	0
1	3	1	0.0	0	18.000000	0.891	1
2	66	0	1.0	0	5.000000	0.330	1
3	70	1	1.0	1	31.000000	0.037	0
4	96	0	1.0	0	30.000000	0.874	1
5	97	1	1.0	0	35.000000	0.309	0
6	10	0	NaN	0	9.000000	3.538	0
7	36	1	NaN	0	34.000000	4.881	0
8	51	1	1.0	0	60.000000	0.874	1
9	55	1	1.0	0	30.222222	0.350	1

In [16]:

```
mode_imputer = SimpleImputer(strategy='most_frequent')
df["INSUR"] = mode_imputer.fit_transform(df[["INSUR"]])
df
```

Out[16]:

	CASENUM	SEX	INSUR	SEATBELT	AGE	LOSS	ATTORNEY
0	5	0	1.0	0	50.000000	34.940	0
1	3	1	0.0	0	18.000000	0.891	1
2	66	0	1.0	0	5.000000	0.330	1
3	70	1	1.0	1	31.000000	0.037	0
4	96	0	1.0	0	30.000000	0.874	1
5	97	1	1.0	0	35.000000	0.309	0
6	10	0	1.0	0	9.000000	3.538	0
7	36	1	1.0	0	34.000000	4.881	0
8	51	1	1.0	0	60.000000	0.874	1
9	55	1	1.0	0	30.222222	0.350	1

```
In [17]: df= pd.read_csv("claimants sample.csv")
         df.isnull().sum()
```

```
Out[17]: CASENUM     0
         SEX         0
         INSUR       2
         SEATBELT    0
         AGE         1
         LOSS        1
         ATTORNEY    0
         dtype: int64
```

```
In [18]: df['AGE'].fillna(df["AGE"].mean(),inplace=True)
         df['LOSS'].fillna(df["LOSS"].median(),inplace=True)
         df['INSUR'].fillna(df["INSUR"].mode()[0],inplace=True)

         df.isnull().sum()
```

```
Out[18]: CASENUM     0
         SEX         0
         INSUR       0
         SEATBELT    0
         AGE         0
         LOSS        0
         ATTORNEY    0
         dtype: int64
```