

```
In [26]: import pandas as pd
pd.read_csv(r"C:\Users\DELL\Downloads\Heart.csv")
```

```
Out[26]:
```

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	Oldi
0	1	63	1	typical	145	233	1	2	150	0	
1	2	67	1	asymptomatic	160	286	0	2	108	1	
2	3	67	1	asymptomatic	120	229	0	2	129	1	
3	4	37	1	nonanginal	130	250	0	0	187	0	
4	5	41	0	nontypical	130	204	0	2	172	0	
...	...	...	...	...	...	...	...	...	...	...	...
298	299	45	1	typical	110	264	0	0	132	0	
299	300	68	1	asymptomatic	144	193	1	0	141	0	
300	301	57	1	asymptomatic	130	131	0	0	115	1	
301	302	57	0	nontypical	130	236	0	2	174	0	
302	303	38	1	nonanginal	138	175	0	0	173	0	

303 rows × 15 columns

## Storing Dataset in variable 'data'

```
In [27]: data = pd.read_csv(r"C:\Users\DELL\Downloads\Heart.csv")
```

## Displaying Data

```
In [4]: print(data)
```

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR
\									
0	1	63	1	typical	145	233	1	2	150
1	2	67	1	asymptomatic	160	286	0	2	108
2	3	67	1	asymptomatic	120	229	0	2	129
3	4	37	1	nonanginal	130	250	0	0	187
4	5	41	0	nontypical	130	204	0	2	172
..	...	...	...	...	...	...	...	...	...
298	299	45	1	typical	110	264	0	0	132
299	300	68	1	asymptomatic	144	193	1	0	141
300	301	57	1	asymptomatic	130	131	0	0	115
301	302	57	0	nontypical	130	236	0	2	174
302	303	38	1	nonanginal	138	175	0	0	173

  

	ExAng	Oldpeak	Slope	Ca	Thal	AHD
0	0	2.3	3	0.0	fixed	No
1	1	1.5	2	3.0	normal	Yes
2	1	2.6	2	2.0	reversable	Yes
3	0	3.5	3	0.0	normal	No
4	0	1.4	1	0.0	normal	No
..	...	...	...	...	...	...
298	0	1.2	2	0.0	reversable	Yes
299	0	3.4	2	2.0	reversable	Yes
300	1	1.2	2	1.0	reversable	Yes
301	0	0.0	2	1.0	normal	Yes
302	0	0.0	1	NaN	normal	No

[303 rows x 15 columns]

## Displaying first 5 rows of database

In [5]: `print(data.head(5))`

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	\
0	1	63	1	typical	145	233	1	2	150	
1	2	67	1	asymptomatic	160	286	0	2	108	
2	3	67	1	asymptomatic	120	229	0	2	129	
3	4	37	1	nonanginal	130	250	0	0	187	
4	5	41	0	nontypical	130	204	0	2	172	

  

	ExAng	Oldpeak	Slope	Ca	Thal	AHD
0	0	2.3	3	0.0	fixed	No
1	1	1.5	2	3.0	normal	Yes
2	1	2.6	2	2.0	reversable	Yes
3	0	3.5	3	0.0	normal	No
4	0	1.4	1	0.0	normal	No

## Displaying last 5 rows of database

In [7]: `print(data.tail(5))`

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR
\									
298	299	45	1	typical	110	264	0	0	132
299	300	68	1	asymptomatic	144	193	1	0	141
300	301	57	1	asymptomatic	130	131	0	0	115
301	302	57	0	nontypical	130	236	0	2	174
302	303	38	1	nonanginal	138	175	0	0	173

  

	ExAng	Oldpeak	Slope	Ca	Thal	AHD
298	0	1.2	2	0.0	reversable	Yes
299	0	3.4	2	2.0	reversable	Yes
300	1	1.2	2	1.0	reversable	Yes
301	0	0.0	2	1.0	normal	Yes
302	0	0.0	1	NaN	normal	No

## Dimension of Database

```
In [8]: print(data.shape)
```

```
(303, 15)
```

## Detailed Information of dataset

```
In [9]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 15 columns):
#   Column          Non-Null Count  Dtype
---  -
0   Unnamed: 0      303 non-null    int64
1   Age             303 non-null    int64
2   Sex             303 non-null    int64
3   ChestPain       303 non-null    object
4   RestBP          303 non-null    int64
5   Chol            303 non-null    int64
6   Fbs             303 non-null    int64
7   RestECG         303 non-null    int64
8   MaxHR           303 non-null    int64
9   ExAng           303 non-null    int64
10  Oldpeak         303 non-null    float64
11  Slope           303 non-null    int64
12  Ca              299 non-null    float64
13  Thal            301 non-null    object
14  AHD             303 non-null    object
dtypes: float64(2), int64(10), object(3)
memory usage: 35.6+ KB
```

## Displaying Column names

```
In [11]: column_names = list(data.columns.values)
print(column_names)
```

```
['Unnamed: 0', 'Age', 'Sex', 'ChestPain', 'RestBP', 'Chol', 'Fbs', 'RestECG', 'MaxHR', 'ExAng', 'Oldpeak', 'Slope', 'Ca', 'Thal', 'AHD']
```

## Renaming column names

```
In [15]: data.rename(columns={'Age':'AGE', 'Chol':'Cholestrol', 'AHD':'ahd'}, inplace=True)
         column_names = list(data.columns.values)
         print(column_names)
```

['Unnamed: 0', 'AGE', 'Sex', 'ChestPain', 'RestBP', 'Cholestrol', 'Fbs', 'RestECG', 'MaxHR', 'ExAng', 'Oldpeak', 'Slope', 'Ca', 'Thal', 'ahd']

```
In [16]: data.head()
```

```
Out[16]:
```

	Unnamed: 0	AGE	Sex	ChestPain	RestBP	Cholestrol	Fbs	RestECG	MaxHR	ExAng	Oldpeak	Slope	Ca	Thal	ahd
0	1	63	1	typical	145	233	1	2	150	0	0	1	0	1	0
1	2	67	1	asymptomatic	160	286	0	2	108	1	0	1	0	1	0
2	3	67	1	asymptomatic	120	229	0	2	129	1	0	1	0	1	0
3	4	37	1	nonanginal	130	250	0	0	187	0	0	1	0	1	0
4	5	41	0	nontypical	130	204	0	2	172	0	0	1	0	1	0

## Null Values in each column

```
In [7]: count_null = data.isna().sum().sum()
         print(count_null)
```

<bound method NDFrame.\_add\_numeric\_operations.<locals>.sum of Unnamed: 0  
0  
Age 0  
Sex 0  
ChestPain 0  
RestBP 0  
Chol 0  
Fbs 0  
RestECG 0  
MaxHR 0  
ExAng 0  
Oldpeak 0  
Slope 0  
Ca 4  
Thal 2  
AHD 0  
dtype: int64>

## Datatype of each attribute of Dataset

```
In [8]: data.dtypes
```

```
Out[8]: Unnamed: 0      int64
Age      int64
Sex      int64
ChestPain object
RestBP   int64
Chol     int64
Fbs      int64
RestECG  int64
MaxHR    int64
ExAng    int64
Oldpeak  float64
Slope    int64
Ca       float64
Thal     object
AHD      object
dtype: object
```

## Mapping Yes/No to 1/0 in AHD column

```
In [29]: data['AHD'] = data['AHD'].map({'Yes': 1, 'No': 0})
print(data)
```

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR
0	1	63	1	typical	145	233	1	2	150
1	2	67	1	asymptomatic	160	286	0	2	108
2	3	67	1	asymptomatic	120	229	0	2	129
3	4	37	1	nonanginal	130	250	0	0	187
4	5	41	0	nontypical	130	204	0	2	172
..	...	...	...	...	...	...	...	...	...
298	299	45	1	typical	110	264	0	0	132
299	300	68	1	asymptomatic	144	193	1	0	141
300	301	57	1	asymptomatic	130	131	0	0	115
301	302	57	0	nontypical	130	236	0	2	174
302	303	38	1	nonanginal	138	175	0	0	173
..									
...									
0	0	2.3	3	0.0	fixed	0			
1	1	1.5	2	3.0	normal	1			
2	1	2.6	2	2.0	reversable	1			
3	0	3.5	3	0.0	normal	0			
4	0	1.4	1	0.0	normal	0			
..	...	...	...	...	...	...			
298	0	1.2	2	0.0	reversable	1			
299	0	3.4	2	2.0	reversable	1			
300	1	1.2	2	1.0	reversable	1			
301	0	0.0	2	1.0	normal	1			
302	0	0.0	1	NaN	normal	0			

[303 rows x 15 columns]

## Dropping unnamed column

```
In [30]: data = data.drop(columns='Unnamed: 0')
print(data)
```

	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	\
0	63	1	typical	145	233	1	2	150	0	
1	67	1	asymptomatic	160	286	0	2	108	1	
2	67	1	asymptomatic	120	229	0	2	129	1	
3	37	1	nonanginal	130	250	0	0	187	0	
4	41	0	nontypical	130	204	0	2	172	0	
..	...	...	...	...	...	...	...	...	...	
298	45	1	typical	110	264	0	0	132	0	
299	68	1	asymptomatic	144	193	1	0	141	0	
300	57	1	asymptomatic	130	131	0	0	115	1	
301	57	0	nontypical	130	236	0	2	174	0	
302	38	1	nonanginal	138	175	0	0	173	0	

  

	Oldpeak	Slope	Ca	Thal	AHD
0	2.3	3	0.0	fixed	0
1	1.5	2	3.0	normal	1
2	2.6	2	2.0	reversable	1
3	3.5	3	0.0	normal	0
4	1.4	1	0.0	normal	0
..	...	...	...	...	...
298	1.2	2	0.0	reversable	1
299	3.4	2	2.0	reversable	1
300	1.2	2	1.0	reversable	1
301	0.0	2	1.0	normal	1
302	0.0	1	NaN	normal	0

[303 rows x 14 columns]

## Splitting Dataset in Training and Testing

```
In [44]: from sklearn.model_selection import train_test_split
x = data[['Age', 'Sex', 'ChestPain', 'RestBP', 'Chol']]
y = data['AHD']
x_train, x_test, y_train, y_test = train_test_split(x,y,test_size=0.2)
```

## Checking newly split dataset

```
In [41]: x_train.shape
```

```
Out[41]: (242, 5)
```

```
In [42]: x_test.shape
```

```
Out[42]: (61, 5)
```