In [1]:
```
# Decision Tree Algorithm ===> Supervised ML Algorithms ===>

# Target data ===>
# Categorical ===> DecisionTreeClassifier()
# Numerical ===> DecisionTreeRegressor
```

In [2]:
```
import numpy as np
import pandas as pd
```

In [3]:
```
df = pd.read_csv("D:\Summer Training Video\ML\covid_toy.csv")
```

In [4]:
```
df
```

Out[4]:

	age	gender	fever	cough	city	has_covid
0	60	Male	103.0	Mild	Kolkata	No
1	27	Male	100.0	Mild	Delhi	Yes
2	42	Male	101.0	Mild	Delhi	No
3	31	Female	98.0	Mild	Kolkata	No
4	65	Female	101.0	Mild	Mumbai	No
...	...	...	...	...	...	...
95	12	Female	104.0	Mild	Bangalore	No
96	51	Female	101.0	Strong	Kolkata	Yes
97	20	Female	101.0	Mild	Bangalore	No
98	5	Female	98.0	Strong	Mumbai	No
99	10	Female	98.0	Strong	Kolkata	Yes

100 rows × 6 columns

In [5]:
```
df = df.dropna()
```

In [6]:
```
from sklearn.preprocessing import LabelEncoder
```

In [7]:
```
lb = LabelEncoder()
```

In [8]:
```
df['gender'] = lb.fit_transform(df['gender'])
df['cough'] = lb.fit_transform(df['cough'])
df['city'] = lb.fit_transform(df['city'])
df['has_covid'] = lb.fit_transform(df['has_covid'])
```

```
C:\Users\yashs\AppData\Local\Temp\ipykernel_18404\1878626996.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)
  df['gender'] = lb.fit_transform(df['gender'])
C:\Users\yashs\AppData\Local\Temp\ipykernel_18404\1878626996.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)
  df['cough'] = lb.fit_transform(df['cough'])
C:\Users\yashs\AppData\Local\Temp\ipykernel_18404\1878626996.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)
  df['city'] = lb.fit_transform(df['city'])
C:\Users\yashs\AppData\Local\Temp\ipykernel_18404\1878626996.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)
  df['has_covid'] = lb.fit_transform(df['has_covid'])
```

In [9]:
```
df.head()
```

Out[9]:

	age	gender	fever	cough	city	has_covid
0	60	1	103.0	0	2	0
1	27	1	100.0	0	1	1
2	42	1	101.0	0	1	0
3	31	0	98.0	0	2	0
4	65	0	101.0	0	3	0

In [10]:
```
x = df.drop(columns = ['has_covid'])
y = df['has_covid']
```

In [11]:
```
from sklearn.model_selection import train_test_split
```

In [12]:
```
x_train , x_test , y_train , y_test = train_test_split(x,y,test_size = 0.2 , random_state = 42
```

In [13]:
```
from sklearn.tree import DecisionTreeClassifier
```

In [14]:
```
dt = DecisionTreeClassifier()
```

In [15]:
```
dt.fit(x_train , y_train)
```

Out[15]:
```
DecisionTreeClassifier
DecisionTreeClassifier()
```

In [16]:
```
y_pred = dt.predict(x_test)
```

In [17]:
```
from sklearn.metrics import accuracy_score
```

In [18]:
```
accuracy_score(y_test , y_pred)
```

Out[18]:
```
0.3888888888888889
```

In [21]:
```
df = pd.read_csv("D:\\Summer Training Video\\ML\\tips.csv")
```

In [22]:
```
df.head()
```

Out[22]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

In [23]:
```
from sklearn.preprocessing import LabelEncoder
```

In [24]:
```
df['sex'] = lb.fit_transform(df['sex'])
df['smoker'] = lb.fit_transform(df['smoker'])
df['day'] = lb.fit_transform(df['day'])
df['time'] = lb.fit_transform(df['time'])
```

In [25]:
```
df.head()
```

Out[25]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	0	0	2	0	2
1	10.34	1.66	1	0	2	0	3
2	21.01	3.50	1	0	2	0	3
3	23.68	3.31	1	0	2	0	2
4	24.59	3.61	0	0	2	0	4

In [27]:
```
x = df.drop(columns = ['total_bill'] , axis = 1)
y = df['total_bill']
```

In [28]:
```
from sklearn.model_selection import train_test_split
```

In [29]:
```
x_train , x_test , y_train , y_test = train_test_split(x,y,test_size = 0.2 , random_state = 42
```

In [30]:
```
from sklearn.tree import DecisionTreeRegressor
```

In [33]:
```
dt = DecisionTreeRegressor()
```

In [34]:
```
dt.fit(x_train , y_train)
```

Out[34]:
```
▾ DecisionTreeRegressor
DecisionTreeRegressor()
```

In [36]:
```
y_pred = dt.predict(x_test)
```

In [38]:
```
from sklearn.metrics import r2_score
```

In [39]:
```
r2_score(y_test , y_pred)
```

Out[39]:
```
0.37640571386610533
```

In [ ]:
```

```