In [1]:
```
#Major Prjoect Submission : IPL Score Prediction by : Milind Dalakoti @ milinddalako
```

In [2]:
```
import pandas as pd
df=pd.read_csv("ipl2017.csv")
```

In [3]:
```
df.head()
#Checking for null values
df.isna().sum()
```

Out[3]:
```
mid               0
date              0
venue             0
bat_team          0
bowl_team         0
batsman           0
bowler            0
runs              0
wickets           0
overs             0
runs_last_5       0
wickets_last_5    0
striker           0
non-striker       0
total             0
dtype: int64
```

In [4]:
```
y=df["total"]
#dropping total [Target column ] and unwnated columns
x=df.drop(["total","bowl_team","bat_team","date"],axis=1)

x.head()
```

Out[4]:

	mid	venue	batsman	bowler	runs	wickets	overs	runs_last_5	wickets_last_5	striker	s
0	1	M Chinnaswamy Stadium	SC Ganguly	P Kumar	1	0	0.1	1	0	0	
1	1	M Chinnaswamy Stadium	BB McCullum	P Kumar	1	0	0.2	1	0	0	
2	1	M Chinnaswamy Stadium	BB McCullum	P Kumar	2	0	0.2	2	0	0	
3	1	M Chinnaswamy Stadium	BB McCullum	P Kumar	2	0	0.3	2	0	0	
4	1	M Chinnaswamy Stadium	BB McCullum	P Kumar	2	0	0.4	2	0	0	

In [5]:
```
# #finding Number of unique values in each column
len(x["venue"].unique())
```

Out[5]:
```
35
```

In [6]:
```
len(x["batsman"].unique().sum())
```

Out[6]:
```
3988
```

In [7]:
```
len(x["bowler"].unique().sum())
```

Out[7]:
```
3241
```

In [8]:
```
type(x)
```

Out[8]:
```
pandas.core.frame.DataFrame
```

In [9]:
```
from sklearn.preprocessing import  LabelEncoder
encoder=LabelEncoder()
```

In [10]:
```
encoder.fit(x["venue"])
x["venue"]=encoder.transform(x["venue"])
encoder.fit(x["batsman"])
x["batsman"]=encoder.transform(x["batsman"])
encoder.fit(x["bowler"])
x["bowler"]=encoder.transform(x["bowler"])
```

In [11]:
```
#Splitting the data into train and test
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=26)
```

In [12]:
```
#Scaling Data
from sklearn.preprocessing import MinMaxScaler
scaler=MinMaxScaler()
#Fitting and Transforming data
x_train=scaler.fit_transform(x_train)
x_test=scaler.transform(x_test)
```

In [13]:
```
#Using Random Forest Regressor
from sklearn.ensemble import RandomForestRegressor
model=RandomForestRegressor()
model.fit(x_train,y_train)
```

Out[13]:
```
RandomForestRegressor()
```

In [14]:
```
#Finding Accuracy
model.score(x_test,y_test)*100
```

Out[14]:
```
93.90109041991124
```

In [15]:
```
data = pd.DataFrame({"mid":[1],"venue":[14],"batsman":[328],"bowler":[96],"runs":[38
data=scaler.transform(data)
model.predict(data)
```

Out[15]: array([222.04])