R2 Evaluation for Random Forest regression Algorithm to finalize best model.

1. Algorithm is a learning process to train the model.
2. Evaluation is a scoring process to save the best model.
3. Hyper tuning parameter for example "n_estimator = 100, random_state =0 " means the changing the values of parameter passed to method/function to get best R2 value.

S.NO	HYPERTUNING PARAMETERS	R.VALUE
1.	RandomForestRegressor(criterion = "absolute_error",n_estimators=50, max_features="sqrt")	.8376
2.	RandomForestRegressor(criterion = "absolute_error",n_estimators=100, max_features="sqrt")	.8321
3.	RandomForestRegressor(criterion = "absolute_error",n_estimators=200, max_features="sqrt")	.8406
4.	RandomForestRegressor(criterion = "absolute_error",n_estimators=500, max_features="sqrt")	.8266
5.	RandomForestRegressor(criterion = "absolute_error",n_estimators=1000, max_features="sqrt")	.8131
6.	RandomForestRegressor(criterion = "absolute_error",n_estimators=50, max_features="log2")	.7921
7.	RandomForestRegressor(criterion = "absolute_error",n_estimators=100, max_features="log2")	.8215
8.	RandomForestRegressor(criterion = "absolute_error",n_estimators=200, max_features="log2")	.8000
9.	RandomForestRegressor(criterion = "absolute_error",n_estimators=500, max_features="log2")	.8211
10.	RandomForestRegressor(criterion = "absolute_error",n_estimators=1000, max_features="log2")	.7951
11.	RandomForestRegressor(criterion = "friedman_mse",n_estimators=50, max_features="sqrt")	.8210
12.	RandomForestRegressor(criterion = "friedman_mse",n_estimators=100, max_features="sqrt")	.7775
13.	RandomForestRegressor(criterion = "friedman_mse",n_estimators=200, max_features="sqrt")	.8085
14.	RandomForestRegressor(criterion = "friedman_mse",n_estimators=500, max_features="sqrt")	.7994
15.	RandomForestRegressor(criterion = "friedman_mse",n_estimators=1000, max_features="sqrt")	.8136
16.	RandomForestRegressor(criterion = "friedman_mse",n_estimators=50, max_features="log2")	.7709
17.	RandomForestRegressor(criterion = "friedman_mse",n_estimators=100, max_features="log2")	.7914
18.	RandomForestRegressor(criterion = "friedman_mse",n_estimators=200, max_features="log2")	.8138
19.	RandomForestRegressor(criterion = "friedman_mse",n_estimators=500, max_features="log2")	.8104
20.	RandomForestRegressor(criterion = "friedman_mse",n_estimators=1000, max_features="log2")	.8121
21.	RandomForestRegressor(criterion = "poisson",n_estimators=50, max_features="sqrt")	.8020
22.	RandomForestRegressor(criterion = "poisson",n_estimators=100,	.8068

	max_features="sqrt")	
23.	RandomForestRegressor(criterion = "poisson",n_estimators=200, max_features="sqrt")	.8355
24.	RandomForestRegressor(criterion = "poisson",n_estimators=500, max_features="sqrt")	.7940
25.	RandomForestRegressor(criterion = "poisson",n_estimators=1000, max_features="sqrt")	.8044
26.	RandomForestRegressor(criterion = "poisson",n_estimators=50, max_features="log2")	.7415
27.	RandomForestRegressor(criterion = "poisson",n_estimators=100, max_features="log2")	.8150
28.	RandomForestRegressor(criterion = "poisson",n_estimators=200, max_features="log2")	.8378
29.	RandomForestRegressor(criterion = "poisson",n_estimators=500, max_features="log2")	.8040
30.	RandomForestRegressor(criterion = "poisson",n_estimators=1000, max_features="log2")	.8084

Note: model creation with best r2score below using hypertuning parameter.

```
from sklearn.ensemble import RandomForestRegressor

regressor=RandomForestRegressor(criterion=”squared_error”,
n_estimators=1000,random_state=0)#hypertuning parameters used.
regressor=regressor.fit(X_train,Y_train)
```

r_score =0.9268746676963855