1.) Identify your problem statement :

Machine Learning

Supervised Learning

Regression

2.) The dataset has the following basic information:

- **Total number of rows:** 1,338
- **Total number of columns:** 6

The columns in the dataset are:

1. **age**
2. **sex**: Categorical
3. **bmi**
4. **children**
5. **smoker**: Categorical
6. **charges**

3.) Mention the pre-processing method :-

Handle Categorical Variables:

- Use one-hot encoding to convert **sex and smoker** columns into numerical format.

Feature Scaling:

- Standardize numerical features.

Splitting the Data:

- Split the dataset into training and testing sets (70% training and 30% testing).

4.) All the research values:

Machine Learning-Regression-R_Score Values

Dataset: Insurance

1.Multiple Linear Regression (R_Score) = **0.7894790349867009**

2.Support Vector Machine-Regression

SVM_Regression (STD)					
SI.NO.	HYPER PARAMETER	LINEAR	RBF	POLY	SIGMOID
1	C=1.0	-0.010102665	-0.083382386	-0.075699656	-0.075429243
2	C=10	0.462468414	-0.032273294	0.038716223	0.039307144
3	C=100	0.628879286	0.320031783	0.617956962	0.527610355
4	C=1000	0.764931174	0.810206485	0.856648768	0.287470695
5	C=2000	0.744041831	0.854776643	0.860557926	-0.593950973
6	C=3000	0.74142366	0.866339395	0.859893008	-2.124419479
7	C=5000	0.74141793	0.874777817	0.859565641	-7.530043238

The SVM-Regression use RBF and C=5000 (hyper parameter)
R_score = **0.874777817**

3.Decision Tree-Regression

Decision_Tree_Regression				
SI.NO.	criterion	splitter	max_features	R2_Score
1	friedman_mse	best	None	0.69588719
2	friedman_mse	random	None	0.71220457
3	friedman_mse	best	sqrt	0.70050015
4	friedman_mse	random	sqrt	0.68980518
5	friedman_mse	best	log2	0.71068497
6	friedman_mse	random	log2	0.74619917
7	squared_error	best	None	0.69633744
8	squared_error	random	None	0.71931098
9	squared_error	best	sqrt	0.60951615
10	squared_error	random	sqrt	0.71931098
11	squared_error	best	log2	0.76738613
12	squared_error	random	log2	0.63676448
13	absolute_error	best	None	0.69016079
14	absolute_error	random	None	0.69549925
15	absolute_error	best	sqrt	0.59047706

16	absolute_error	random	sqrt	0.73845369
17	absolute_error	best	log2	0.719922
18	absolute_error	random	log2	0.59945579
19	poisson	best	None	0.71754799
20	poisson	random	None	0.67895904
21	poisson	best	sqrt	0.75134486
22	poisson	random	sqrt	0.70693081
23	poisson	best	log2	0.75547955
24	poisson	random	log2	0.6743258

The Decision Tree use **criterion=** squared_error , **splitter=** best**, max_features**= log2

R_score =**0.76738613**

4.Random Forest

Random_Forest				
Sl.NO.	**criterion**	***n_estimators***	**max_features**	**R2_Score**
1	squared_error	10	sqrt	0.853671439
2	squared_error	50	sqrt	0.865042286
3	squared_error	100	sqrt	0.868506878
4	squared_error	10	log2	0.844538356
5	squared_error	50	log2	0.865189498
6	squared_error	100	log2	0.870743523
7	squared_error	10	None	0.837679284
8	squared_error	50	None	0.853817835
9	squared_error	100	None	0.854961572
10	absolute_error	10	sqrt	0.864278209
11	absolute_error	50	sqrt	0.870360398
12	absolute_error	100	sqrt	0.872352051
13	absolute_error	10	log2	0.85979989
14	absolute_error	50	log2	0.872264686
15	absolute_error	100	log2	0.868086959
16	absolute_error	10	None	0.845056409
17	absolute_error	50	None	0.855506593
18	absolute_error	100	None	0.854925901
19	friedman_mse	10	sqrt	0.850876844
20	friedman_mse	50	sqrt	0.873260014
21	friedman_mse	100	sqrt	0.873410969
22	friedman_mse	10	log2	0.844100539
23	friedman_mse	50	log2	0.868782416
24	friedman_mse	100	log2	0.869824005
25	friedman_mse	10	None	0.819785501
26	friedman_mse	50	None	0.847562523
27	friedman_mse	100	None	0.853352006
28	poisson	10	sqrt	0.861795051

29	poisson	50	sqrt	0.872890401
30	poisson	100	sqrt	0.870791136
31	poisson	10	log2	0.855738714
32	poisson	50	log2	0.869570412
33	poisson	100	log2	0.872498264
34	poisson	10	None	0.845554692
35	poisson	50	None	0.855578956
36	poisson	100	None	0.854804434

The Random Forest use **criterion=** friedman_mse, **n_estimators**=100, **max_features**= sqrt
R_score =**0.873410969**

5.) The Final Machine Learning best method of Regression:

The SVM-Regression use RBF and C=5000 (hyper parameter)
R_score = **0.874777817**