1) Identify your problem statement

stage 1 : **Machine Learning**

stage 2 : **Supervised**

stage 3: **Classification**

2) Tell basic info about the dataset (Total number of rows, columns)
   a) Dataset contain 399 rows × 25 columns
   b) 27 input column (age, sex, bmi, children, smoker) and 1 output column (Charges)
   c) **Sg, rbc, pc,pcc,ba,htn,dm,cad,appet,pe and ane** are categorical (Ordinal) column so we have to convert as numerical data

3) Mention the pre-processing method if you're doing any (like converting string to number – nominal data)

Dataset for Before Preprocessing

	age	bp	sg	al	su	rbc	pc	pcc	ba	bgr	...	pcv	wc	rc	htn	dm	cad	appet	pe	ane
0	2.000000	76.459948	c	3.0	0.0	normal	abnormal	notpresent	notpresent	148.112676	...	38.868902	8408.191126	4.705597	no	no	no	yes	yes	no
1	3.000000	76.459948	c	2.0	0.0	normal	normal	notpresent	notpresent	148.112676	...	34.000000	12300.000000	4.705597	no	no	no	yes	poor	no
2	4.000000	76.459948	a	1.0	0.0	normal	normal	notpresent	notpresent	99.000000	...	34.000000	8408.191126	4.705597	no	no	no	yes	poor	no
3	5.000000	76.459948	d	1.0	0.0	normal	normal	notpresent	notpresent	148.112676	...	38.868902	8408.191126	4.705597	no	no	no	yes	poor	yes
4	5.000000	50.000000	c	0.0	0.0	normal	normal	notpresent	notpresent	148.112676	...	36.000000	12400.000000	4.705597	no	no	no	yes	poor	no

Dataset for After Preprocessing

a) Converting string to number – Ordinal Mapping (Label Encoder)

	age	bp	al	su	bgr	bu	sc	sod	pot	hrmo	...	pc_normal	pcc_present	ba_present	htn_yes	dm_yes	cad_ye
0	2.000000	76.459948	3.0	0.0	148.112676	57.482105	3.077356	137.528754	4.627244	12.518156	...	False	False	False	False	False	Fals
1	3.000000	76.459948	2.0	0.0	148.112676	22.000000	0.700000	137.528754	4.627244	10.700000	...	True	False	False	False	False	Fals
2	4.000000	76.459948	1.0	0.0	99.000000	23.000000	0.600000	138.000000	4.400000	12.000000	...	True	False	False	False	False	Fals
3	5.000000	76.459948	1.0	0.0	148.112676	16.000000	0.700000	138.000000	3.200000	8.100000	...	True	False	False	False	False	Fals
4	5.000000	50.000000	0.0	0.0	148.112676	25.000000	0.600000	137.528754	4.627244	11.800000	...	True	False	False	False	False	Fals

4. Develop a good model with good evaluation metric. You can use any machine learning algorithm; you can create many models. Finally, you have to come up with final model..

a) SVM Grid Search

```
The f1_macro value for best parameter {'C': 10, 'gamma': 'auto', 'kernel': 'sigmoid'}: 0.9924946382275899
```

```
print("The confusion Matrix:\n",cm)
```

```
The confusion Matrix:
 [[51  0]
 [ 1 81]]
```

```
print("The report:\n",clf_report)
```

```
The report:
              precision    recall  f1-score   support

       False       0.98      1.00      0.99        51
        True       1.00      0.99      0.99        82

    accuracy                           0.99       133
   macro avg       0.99      0.99      0.99       133
weighted avg       0.99      0.99      0.99       133
```

Accuracy is 0.9924946382275899

b) Descension tree Grid Search

```
The f1_macro value for best parameter {'criterion': 'entropy', 'max_features': 'log2', 'splitter': 'random'}: 0.9551744928358632
```

```
print("The confusion Matrix:\n",cm)
```

```
The confusion Matrix:
 [[50  1]
 [ 5 77]]
```

```
print("The report:\n",clf_report)
```

```
The report:
              precision    recall  f1-score   support

       False       0.91      0.98      0.94        51
        True       0.99      0.94      0.96        82

    accuracy                           0.95       133
   macro avg       0.95      0.96      0.95       133
weighted avg       0.96      0.95      0.96       133
```

Accuracy is 0.812447479

c) Random Forest Grid Search

```
The f1_macro value for best parameter {'criterion': 'gini', 'max_features': 'sqrt', 'n_estimators': 100}: 0.9849624060150376
```

```
print("The confusion Matrix:\n",cm)
```

```
The confusion Matrix:
 [[50  1]
 [ 1 81]]
```

```
print("The report:\n",clf_report)
```

```
The report:
              precision    recall  f1-score   support

       False       0.98      0.98      0.98        51
        True       0.99      0.99      0.99        82

    accuracy                           0.98       133
   macro avg       0.98      0.98      0.98       133
weighted avg       0.98      0.98      0.98       133
```

Accuracy is 0.9849624060150376

Final Result

SVM Grid Search is the best model which gives high Accuracy is 0.9924946382275899 compare to other model