```python
# train a Gaussian Naive Bayes classifier on the training set
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score
from sklearn.model_selection import cross_val_score
from sklearn.metrics import classification_report


# instantiate the model
gnb = GaussianNB()
#5-fold cross validation
scores = cross_val_score(gnb, X_train, y_train, cv = 5,
scoring='accuracy')
print('5 Cross validation score of GaussianNB model:{}'.format(scores))
print(" ")

# fit the model
gnb.fit(X_train, y_train)
print(" ")
y_pred1 = gnb.predict(X_train)
print(" ")
y_pred = gnb.predict(X_test)


print('Model accuracy score of GaussianNB model: {0:0.4f}'.
format(accuracy_score(y_test, y_pred)))
print('Training accuracy score of GaussianNB model: {0:0.4f}'.
format(accuracy_score(y_train, y_pred1)))
print('The Classification Report of GaussianNB
model\n\n'+(classification_report(y_test, y_pred)))
```

```
5 Cross validation score of GaussianNB model:[0.79886381 0.79821867
0.80190418 0.7948402
9 0.80666462]
Model accuracy score of GaussianNB model: 0.8009
Training accuracy score of GaussianNB model: 0.8001
The Classification Report of GaussianNB model
precision recall f1-score support
<=50K. 0.82 0.95 0.88 12435
>50K. 0.66 0.32 0.43 3846
accuracy 0.80 16281
macro avg 0.74 0.64 0.66 16281
weighted avg 0.78 0.80 0.77 16281
```

```python
# train a DecisionTreeClassifier on the training set

from sklearn.tree import DecisionTreeClassifier

# instantiate the model

clf = DecisionTreeClassifier()

#5-fold cross validation
scores = cross_val_score(clf, X_train, y_train, cv = 5,
scoring='accuracy')
print('5 Cross validation score of DecisionTreeClassifier
model:{}'.format(scores))

# fit the model
clf.fit(X_train, y_train)
y_pred1 = clf.predict(X_train)
y_pred = clf.predict(X_test)

print('Model accuracy score: {0:0.4f}'. format(accuracy_score(y_test,
y_pred)))
print('Training accuracy score: {0:0.4f}'. format(accuracy_score(y_train,
y_pred1)))
print('The Classification Report of DecisionTreeClassification
model\n\n'+classification_report(y_test, y_pred))
```

```
5 Cross validation score of DecisionTreeClassifier model:[0.84216183
0.84566953 0.851812
04 0.85150491 0.85457617]
Model accuracy score: 0.8519
Training accuracy score: 0.8521
The Classification Report of DecisionTreeClassification model
precision recall f1-score support
<=50K. 0.87 0.95 0.91 12435
>50K. 0.77 0.53 0.63 3846
accuracy 0.85 16281
macro avg 0.82 0.74 0.77 16281
weighted avg 0.84 0.85 0.84 16281
```

```python
# train a LogisticRegression on the training set


from sklearn.linear_model import LogisticRegression
```

```
# instantiate the model

lr= LogisticRegression( solver='lbfgs',max_iter = 700)

#5-fold cross validation
scores = cross_val_score(lr, X_train, y_train, cv = 5, scoring='accuracy')
print('5 Cross validation score of Logistic Regression
model:{}'.format(scores))

# fit the model
lr.fit(X_train, y_train)
y_pred1 = lr.predict(X_train)
y_pred = lr.predict(X_test)

print('Model accuracy score: {0:0.4f}'. format(accuracy_score(y_test,
y_pred)))
print('Training accuracy score: {0:0.4f}'. format(accuracy_score(y_train,
y_pred1)))
print('The Classification Report of Logistic Regression
model\n\n'+classification_report(y_test, y_pred))
```

```
5 Cross validation score of Logistic Regression model:[0.82250883 0.8240172
0.8264742
0.82324939 0.82877764]
Model accuracy score: 0.8262
Training accuracy score: 0.8253
The Classification Report of Logistic Regression model
precision recall f1-score support
<=50K. 0.85 0.94 0.89 12435
>50K. 0.71 0.45 0.55 3846
accuracy 0.83 16281
macro avg 0.78 0.70 0.72 16281
weighted avg 0.81 0.83 0.81 16281
```