```python
from sklearn.linear_model import LinearRegression

model = LinearRegression(fit_intercept=True)

from sklearn.model_selection import train_test_split

Xtrain,Xtest,ytrain,ytest=train_test_split(X,y,train_size=0.7,random_state=1957)

model.fit(Xtrain, ytrain)
print(model.coef_)
print(model.intercept_)

from sklearn.metrics import mean_squared_error, r2_score, mean_absolute_error

y_model = model.predict(Xtest)

from sklearn.model_selection import cross_val_score
scores=cross_val_score(model, X, y,cv=5)

numeric_transformer=Pipeline(steps=[("imputer",
SimpleImputer(strategy='median')),('scaler',StandardScaler())])

categorical_transformer=Pipeline(steps=[('imputer',SimpleImputer(strategy='constant',fill_value=
'missing')),('onehot',OneHotEncoder(handle_unknown='ignore'))])



lr_clf=Pipeline(steps=[('preprocessor',preprocessor),
('regressor',LinearRegression(fit_intercept=True))])



from sklearn.linear_model import SGDRegressor
lr_sgd = Pipeline(steps=[('preprocessor',preprocessor),
                ('regressor', SGDRegressor(random_state=1957))])
lr_sgd.fit(X_train, y_train)
y_model_sgd = lr_sgd.predict(X_test)


from sklearn.tree import DecisionTreeRegressor
ls_dt = Pipeline(steps=[('preprocessor',preprocessor),
                ('regressor', DecisionTreeRegressor(random_state=1957))])
ls_dt.fit(X_train, y_train)
y_model_dt = ls_dt.predict(X_test)
```

```python
from sklearn.ensemble import RandomForestRegressor

ls_rf = Pipeline(steps=[('preprocessor',preprocessor),
                        ('regressor', RandomForestRegressor(random_state=1957))])
ls_rf.fit(X_train, y_train)
y_model_rf = ls_rf.predict(X_test)

From sklearn.metrics import confusion_matrix
confusion_matrix(y_train_5, y_train_pred)
```