```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score
import matplotlib.pyplot as plt

data=pd.read_csv('/content/drive/MyDrive/archive
(5)/ecommerce_product_dataset.csv')

df=pd.DataFrame(data)

from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
x = df[['Sales','ProductID']]
y = df['Price']

import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsRegressor # Use
KNeighborsRegressor for regression
from sklearn.metrics import accuracy_score
import matplotlib.pyplot as plt

# Load the data (assuming this part is already done)
data = pd.read_csv('/content/drive/MyDrive/archive
(5)/ecommerce_product_dataset.csv')
df = pd.DataFrame(data)

# Select features and target variable
x = df[['Sales', 'ProductID']]
y = df['Price']

# Create and fit the KNeighborsRegressor model
k = 3
knn = KNeighborsRegressor(n_neighbors=k)  # Change to
KNeighborsRegressor
knn.fit(x, y)
```

```
KNeighborsRegressor(n_neighbors=3)
```

```
# prompt: create a cose for printing the output for predicting price
using sales and prouductid as per user input

def predict_price(sales, product_id):
  """
  Predicts the price of a product based on its sales and product ID.
```

```
  Args:
    sales: The number of sales for the product.
    product_id: The ID of the product.

  Returns:
    The predicted price of the product.
  """
  input_data = np.array([[sales, product_id]])
  predicted_price = knn.predict(input_data)[0]
  return predicted_price

# Get user input for sales and product ID
sales = float(input("Enter the number of sales: "))
product_id = int(input("Enter the product ID: "))

# Predict the price
predicted_price = predict_price(sales, product_id)

# Print the predicted price
print(f"Predicted price: {predicted_price}")
```

```
Enter the sales value: 466
Enter the product ID: 1
Predicted price of the product: 252.45666666666668

/usr/local/lib/python3.10/dist-packages/sklearn/base.py:439:
UserWarning: X does not have valid feature names, but
KNeighborsRegressor was fitted with feature names
  warnings.warn(
```