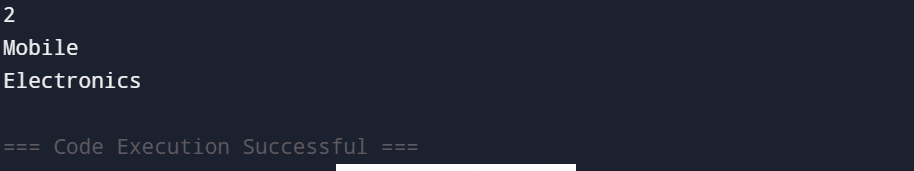
Data Structures & Algorithms

# Exercise 2: E-commerce Platform Search Function

Code:

public class Main {  
 class Product {  
 int id;  
 String name;  
 String category;  
  
 Product(int id, String name, String category) {  
 this.id = id;  
 this.name = name;  
 this.category = category;  
 }  
  
 public void printDetails() {  
 System.out.println(id);  
 System.out.println(name);  
 System.out.println(category);  
 }  
 }  
  
 void searchByName(Product[] products, String name) {  
 boolean found = false;  
 for (Product p : products) {  
 if (p.name.equalsIgnoreCase(name)) {  
 p.printDetails();  
 found = true;  
 break;  
 }  
 }  
 }  
  
 void example() {  
 Product[] items = {  
 new Product(1, "Laptop", "Electronics"),  
 new Product(2, "Mobile", "Electronics"),  
 new Product(3, "Shoes", "Fashion")  
 };  
  
 searchByName(items, "Mobile");  
 }  
  
 public static void main(String[] args) {  
 new Main().example();  
 }  
}

Output:



# Exercise 7: Financial Forecasting

Code:

public class Main {  
 int calculate(int amount, int rate, int years) {  
 if (years == 0) {  
 return amount;  
 }  
 return calculate(amount, rate, years - 1) \* (1 + rate);  
 }  
  
 void run() {  
 int Amount = 1000;  
 int Rate = 2;  
 int Years = 4;  
  
 int result = calculate(Amount, Rate, Years);  
  
 System.out.println("No of years:" + Years);  
 System.out.println("Results: Rs." + result);  
 }  
  
 public static void main(String[] args) {  
 new Main().run();  
 }  
}

Output:

