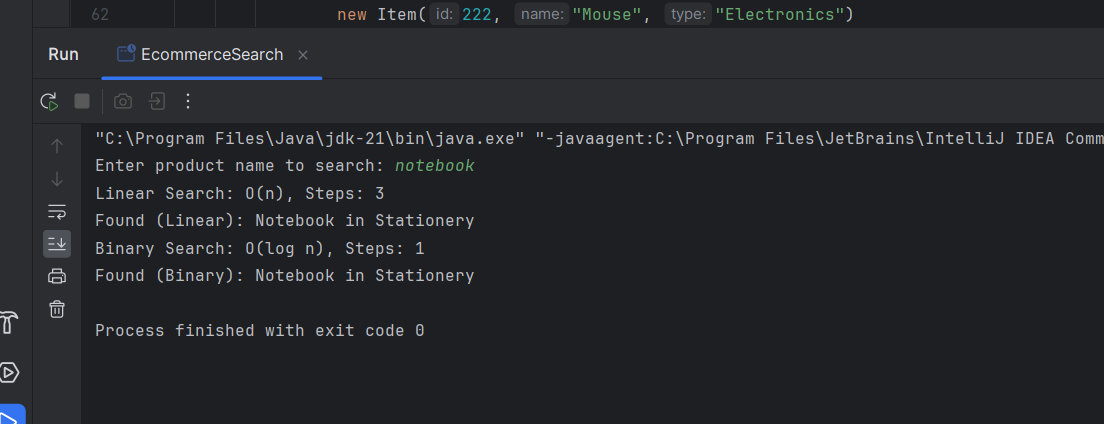
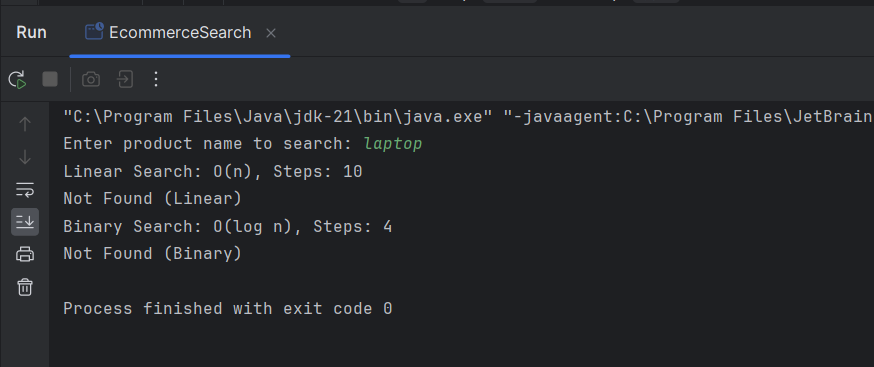
**Exercise 2: E-commerce Platform Search Function**

**Code:**

import java.util.\*;  
  
class Item {  
 int id;  
 String name;  
 String type;  
  
 Item(int id, String name, String type) {  
 this.id = id;  
 this.name = name;  
 this.type = type;  
 }  
}  
class Finder {  
 public static Item findByLinear(Item[] list, String keyword) {  
 int steps = 0;  
 for (Item it : list) {  
 steps++;  
 if (it.name.equalsIgnoreCase(keyword)) {  
 System.*out*.println("Linear Search: O(n), Steps: " + steps);  
 return it;  
 }  
 }  
 System.*out*.println("Linear Search: O(n), Steps: " + steps);  
 return null;  
 }  
  
 public static Item findByBinary(Item[] sortedList, String keyword) {  
 int low = 0, high = sortedList.length - 1;  
 int steps = 0;  
  
 while (low <= high) {  
 steps++;  
 int mid = (low + high) / 2;  
 int check = sortedList[mid].name.compareToIgnoreCase(keyword);  
  
 if (check == 0) {  
 System.*out*.println("Binary Search: O(log n), Steps: " + steps);  
 return sortedList[mid];  
 }  
 if (check < 0) low = mid + 1;  
 else high = mid - 1;  
 }  
  
 System.*out*.println("Binary Search: O(log n), Steps: " + steps);  
 return null;  
 }  
}  
public class EcommerceSearch {  
 public static void main(String[] args) {  
 Item[] items = {  
 new Item(201, "T-Shirt", "Clothing"),  
 new Item(105, "Headphones", "Electronics"),  
 new Item(312, "Notebook", "Stationery"),  
 new Item(408, "Sneakers", "Footwear"),  
 new Item(529, "Watch", "Accessories"),  
 new Item(777, "Sunglasses", "Accessories"),  
 new Item(888, "Shoes", "Footwear"),  
 new Item(999, "Bag", "Travel"),  
 new Item(111, "Camera", "Electronics"),  
 new Item(222, "Mouse", "Electronics")  
 };  
 Scanner scan = new Scanner(System.*in*);  
 System.*out*.print("Enter product name to search: ");  
 String input = scan.nextLine();  
  
 Item resultLinear = Finder.*findByLinear*(items, input);  
 if (resultLinear != null)  
 System.*out*.println("Found (Linear): " + resultLinear.name + " in " + resultLinear.type);  
 else  
 System.*out*.println("Not Found (Linear)");  
  
 Arrays.*sort*(items, Comparator.*comparing*(i -> i.name.toLowerCase()));  
  
 Item resultBinary = Finder.*findByBinary*(items, input);  
 if (resultBinary != null)  
 System.*out*.println("Found (Binary): " + resultBinary.name + " in " + resultBinary.type);  
 else  
 System.*out*.println("Not Found (Binary)");  
 }  
}

**Output**:





**Exercise 7: Financial Forecasting**

**Code:**

import java.util.\*;  
  
public class MoneyPredictor {  
  
 static Map<Integer, Double> *memory* = new HashMap<>();  
  
 public static double predictValue(int year, double base, double rate) {  
 if (year == 0) return base;  
 if (*memory*.containsKey(year)) return *memory*.get(year);  
  
 double past = *predictValue*(year - 1, base, rate);  
 double future = past \* (1 + rate);  
 *memory*.put(year, future);  
  
 return future;  
 }  
  
 public static void main(String[] args) {  
 Scanner input = new Scanner(System.*in*);  
  
 System.*out*.print("Enter starting amount: ");  
 double initial = input.nextDouble();  
  
 System.*out*.print("Enter annual growth rate (e.g. 0.1 for 10%): ");  
 double growth = input.nextDouble();  
  
 System.*out*.print("Enter how many years to forecast: ");  
 int years = input.nextInt();  
  
 System.*out*.println("Year-wise forecast:");  
 for (int i = 0; i <= years; i++) {  
 double amount = *predictValue*(i, initial, growth);  
 System.*out*.printf("Year %d: ₹%.2f%n", i, amount);  
 }  
 }  
}

**Output:**

