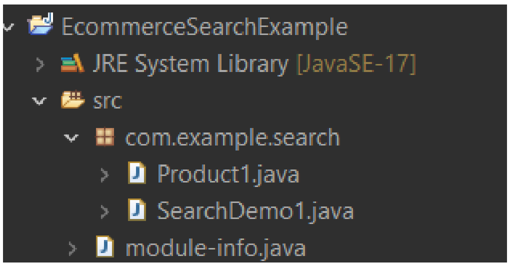
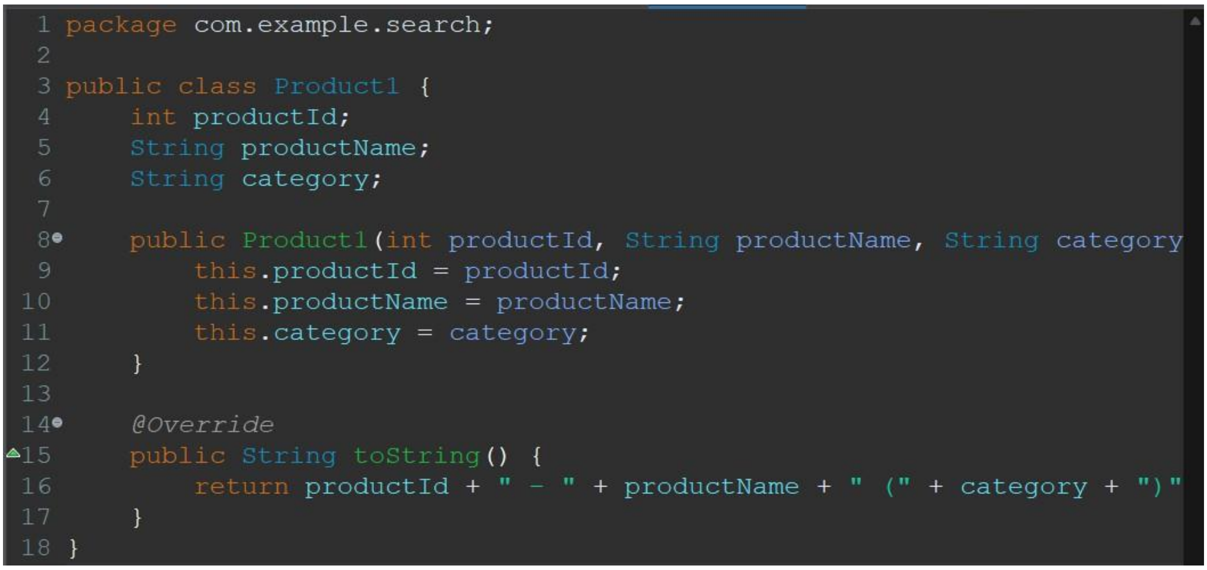
E-commerce Platform Search Function

Flow:



```
EcommerceSearchExample
  JRE System Library [JavaSE-17]
  src
    com.example.search
      Product1.java
      SearchDemo1.java
    module-info.java
```

Product1.java



```java
package com.example.search;

public class Product1 {
    int productId;
    String productName;
    String category;

    public Product1(int productId, String productName, String category
        this.productId = productId;
        this.productName = productName;
        this.category = category;
    }

    @Override
    public String toString() {
        return productId + " - " + productName + " (" + category + ")"
    }
}
```

SearchDemo1.java

package com.example.search;

import java.util.Arrays;

import java.util.Comparator;

import java.util.Scanner;

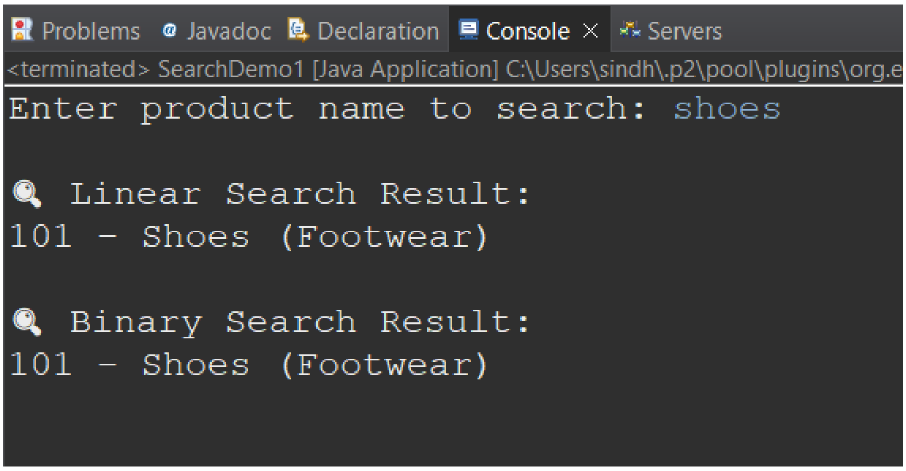
public class SearchDemo1 {

```
public static Product1 linearSearch(Product1[] products, String name) {
    for (Product1 product : products) {
        if (product.productName.equalsIgnoreCase(name)) {
            return product;
        }
    }
    return null;
}


public static Product1 binarySearch(Product1[] products, String name) {
    int left = 0;
    int right = products.length - 1;

    while (left <= right) {
        int mid = (left + right) / 2;
        int cmp = products[mid].productName.compareToIgnoreCase(name);

        if (cmp == 0) return products[mid];
        else if (cmp < 0) left = mid + 1;
        else right = mid - 1;
    }
    return null;
```

```
}

public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);



    Product1[] products = {
        new Product1(101, "Shoes", "Footwear"),
        new Product1(102, "Watch", "Accessories"),
        new Product1(103, "Laptop", "Electronics"),
        new Product1(104, "Phone", "Electronics"),
        new Product1(105, "Shirt", "Clothing")
    };

    System.out.print("Enter product name to search: ");
    String inputName = sc.nextLine();



    Product1 result1 = linearSearch(products, inputName);
    System.out.println("\n Linear Search Result:");
    System.out.println(result1 != null ? result1 : "Product not found");

    Arrays.sort(products, Comparator.comparing(p -> p.productName));
    Product1 result2 = binarySearch(products, inputName);
    System.out.println("\n Binary Search Result:");
```

```
        System.out.println(result2 != null ? result2 : "Product not found");


        sc.close();
    }
}
```

Output:

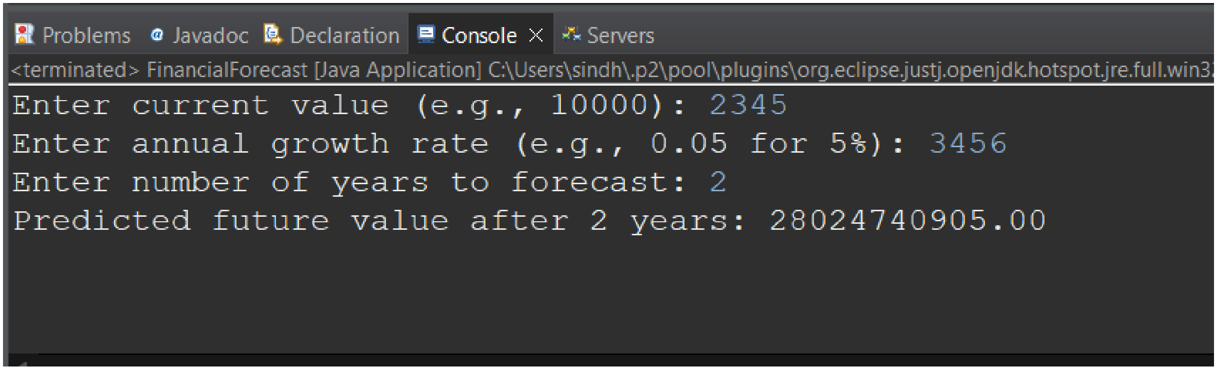


```
<terminated> SearchDemo1 [Java Application] C:\Users\sindh\.p2\pool\plugins\org.e
Enter product name to search: shoes

🔍 Linear Search Result:
101 - Shoes (Footwear)

🔍 Binary Search Result:
101 - Shoes (Footwear)
```

Financial Forecasting
Code: FinancialForecast.java

```
package com.example.forecast;

import java.util.Scanner;

public class FinancialForecast {

    // Recursive method to calculate future value
    public static double forecastFutureValue(double currentValue, double growthRate, int years) {
        if (years == 0) {
            return currentValue;
        }
        return forecastFutureValue(currentValue, growthRate, years - 1) * (1 + growthRate);
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter current value (e.g., 10000): ");
        double currentValue = sc.nextDouble();

        System.out.print("Enter annual growth rate (e.g., 0.05 for 5%): ");
        double growthRate = sc.nextDouble();

        System.out.print("Enter number of years to forecast: ");
        int years = sc.nextInt();

        double futureValue = forecastFutureValue(currentValue, growthRate,
years);

        System.out.printf("Predicted future value after %d years: %.2f\n", years,
futureValue);

        sc.close();
    }
}
```

Output:



```
<terminated> FinancialForecast [Java Application] C:\Users\sindh\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win3
Enter current value (e.g., 10000): 2345
Enter annual growth rate (e.g., 0.05 for 5%): 3456
Enter number of years to forecast: 2
Predicted future value after 2 years: 28024740905.00
```