**1.Ecommerce Search Function**

package Ecom;

import java.util.\*;

public class Main {

public static void main(String[] args) {

Product[] products = {

new Product(101, "Laptop", "Electronics"),

new Product(205, "Shirt", "Clothing"),

new Product(150, "Watch", "Accessories"),

new Product(303, "Mobile", "Electronics")

};

// Sort array for binary search

Arrays.*sort*(products, Comparator.*comparingInt*(p -> p.productId));

int searchId = 150;

// Linear Search

Product result1 = SearchProduct.*linearSearch*(products, searchId);

System.*out*.println("Linear Search:");

System.*out*.println(result1 != null ? result1.productName : "Product not found");

// Binary Search

Product result2 = SearchProduct.*binarySearch*(products, searchId);

System.*out*.println("Binary Search:");

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

}

}

**package** Ecom;

**public** **class** Product {

**int** productId;

String productName;

String category;

**public** Product(**int** productId, String productName, String category) {

**this**.productId = productId;

**this**.productName = productName;

**this**.category = category;

}

}

**package** Ecom;

**public** **class** SearchProduct {

**public** **static** Product linearSearch(Product[] products, **int** targetId) {

**for** (Product product : products) {

**if** (product.productId == targetId) {

**return** product;

}

}

**return** **null**;

}

// Binary search - array must be sorted by productId

**public** **static** Product binarySearch(Product[] products, **int** targetId) {

**int** low = 0, high = products.length - 1;

**while** (low <= high) {

**int** mid = (low + high) / 2;

**if** (products[mid].productId == targetId) {

**return** products[mid];

} **else** **if** (products[mid].productId < targetId) {

low = mid + 1;

} **else** {

high = mid - 1;

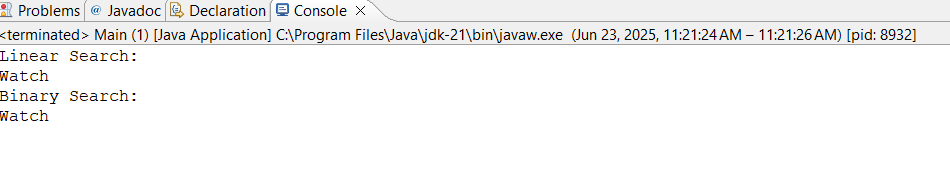
}

}

**return** **null**;

}

}



**2.Financial Forecasting**

Public class FinancialForecast{

public static double calculateFutureValue(double presentValue, double rate, int years) {  
 if (years == 0) {  
 return presentValue;  
 }  
 return (1 + rate) \* calculateFutureValue(presentValue, rate, years - 1);  
}  
  
public static void main(String[] args) {  
 double presentValue = 10000.0; // Starting money  
 double annualGrowthRate = 0.08; // 8%  
 int years = 5;  
  
 double futureValue = calculateFutureValue(presentValue, annualGrowthRate, years);  
 System.out.printf("Future Value after %d years: ₹%.2f\n", years, futureValue);  
}

}

