**Algorithms and Data Structures**

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

**Product class:**

package algo;

public class Product {

public int proId;

public String proName;

public String category;

public Product(int id,String name,String cate) {

proId=id;

proName=name;

category=cate;

}

public void display() {

System.*out*.println("The product name is " + proName + " id is "+proId + " and category is " + category);

}

}

**Main class:**

package algo;

import java.util.\*;

public class Main {

public static void main(String[] args) {

Scanner sc= new Scanner(System.*in*);

//Linear search

Product[] products = new Product[3];

products[0]= new Product(18,"mouse","zebronics");

products[1]= new Product(2,"pendrive","drive");

products[2]= new Product(26,"keyboard","dell");

System.*out*.println("enter");

int id=sc.nextInt();

int n2=sc.nextInt();

int flag=0;

for(int i=0;i<3;i++) {

if(products[i].proId == id) {

flag=1;

System.*out*.println("found");

products[i].display();

break;

}

}

if(flag==0)

System.*out*.println("not found");

flag=0;

//Binary Search

Product[] products2 = new Product[5];

products2[0]= new Product(21,"watch","titan");

products2[1]= new Product(22,"bag","luis");

products2[2]= new Product(26,"makeup","dazller");

products2[3]= new Product(29,"shoes","bata");

products2[4]= new Product(33,"saree","silk");

int lo=0,hi=products2.length-1;

while(lo<=hi) {

int mid=(lo+hi)/2;

if(products2[mid].proId==n2) {

flag=1;

System.*out*.println("found");

products2[mid].display();

break;

}

else if(products2[mid].proId<n2)

lo=mid+1;

else

hi=mid-1;

}

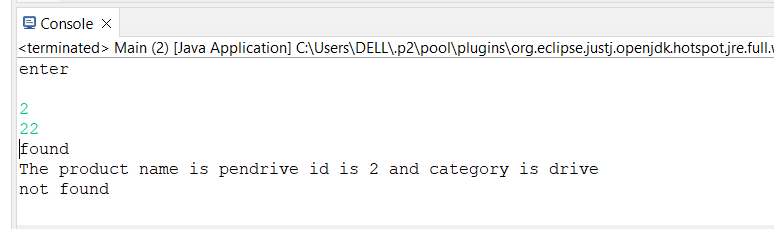
if(flag==0)

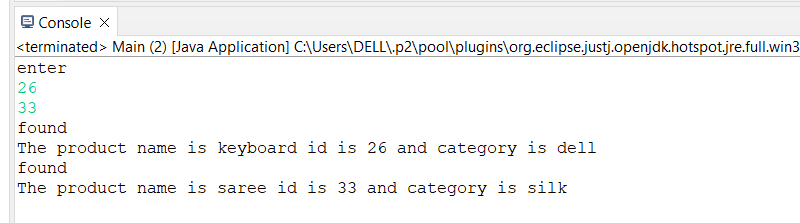
System.*out*.println("not found");

}

}

**OUTPUT**:





**Exercise-7: Financial Forecasting**

**Forecast:**

package algo;

public class Forecast {

public static double futureVal(double principal, double rate, int years) {

if (years == 0) {

return principal;

}

return *futureVal*(principal \* (1 + rate), rate, years - 1);

}

public static void main(String[] args) {

double initialInvestment = 1000.0;

double annualRate = 0.05;

int numYears = 5;

double futureVal = *futureVal*(initialInvestment, annualRate, numYears);

System.*out*.printf("Future Value after %d years: %.2f%n", numYears, futureVal);

}

}

**OUTPUT:**

