Java Lambda Assignments

1. Write an application to perform basic arithmetic operations like add, subtract, multiply & divide. You need to define a functional interface first.

interface operations{

public int operators(int a,int b);

}

public class l1 {

public static void main(String[] args) {

// TODO Auto-generated method stub

operations add = (a,b)->(a+b);

System.out.println(add.operators(2,6));

operations Sub = (a,b)->(a-b);

System.out.println(Sub.operators(6,2));

operations Mul = (a,b)->(a\*b);

System.out.println(Mul.operators(2,6));

operations div = (a,b)->(a/b);

System.out.println(div.operators(6,2));

}

}

1. Write an application using lambda expressions to print Orders having 2 criteria implemented. 1) order price more than 10000 2) order status is ACCEPTED or COMPLETED

import java.util.\*;

class L2Order

{

private int price;

private String status;

public L2Order(int price, String status)

{

this.price=price;

this.status=status;

}

public int getPrice() {

return price;

}

public void setPrice(int price) {

this.price = price;

}

public String getStatus() {

return status;

}

public void setStatus(String status) {

this.status = status;

}

public static ArrayList<L2Order> listofOrders(ArrayList<L2Order> orders)

{

ArrayList<L2Order> listofOrders=new ArrayList();

for(L2Order order:orders)

{

if(order.getPrice()>10000 && (order.getStatus().equals("Accepted")|| order.getStatus().equals("Completed")))

{

System.out.println("inside if");

L2Order ord = new L2Order(order.getPrice(), order.getStatus());

listofOrders.add(ord);

}

}

return listofOrders;

}

public static void main(String[] args) {

// TODO Auto-generated method stub

L2Order o1 = new L2Order(15000,"Accepted");

L2Order o2 = new L2Order(1000,"Completed");

L2Order o3 = new L2Order(50000,"Rejected");

ArrayList<L2Order> list = new ArrayList<>();

list.add(o1);

list.add(o2);

list.add(o3);

System.out.println(list);

System.out.println(listofOrders(list));

}

}