1. **import** java.util.Scanner;

@FunctionalInterface

**interface** Airthematic {

**int** operations(**int** a, **int** b);

}

**public** **class** Airth\_op {

**public** **static** **void** main(String[] args) {

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

System.***out***.println("Enter two arguments");

**int** n = scn.nextInt();

**int** m = scn.nextInt();

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

System.***out***.println("Addition: "+add.operations(n,m));

Airthematic sub = (a,b) -> (a-b);

System.***out***.println("Subtraction: "+sub.operations(n,m));

Airthematic mul = (a,b) -> (a\*b);

System.***out***.println("Multiplication: "+mul.operations(n,m));

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

System.***out***.println("Division: "+div.operations(n,m));

}

1. **public** **class** orders {

**private** **int** order\_id;

**private** **int** price;

**private** String status;

**public** orders(**int** order\_id, **int** price, String status) {

**this**.order\_id = order\_id;

**this**.price = price;

**this**.status = status;

}

**public** **int** getPrice() {

**return** price;

}

**public** **void** setPrice(**int** price) {

**this**.price = price;

}

**public** **int** getOrder\_id() {

**return** order\_id;

}

**public** **void** setOrder\_id(**int** order\_id) {

**this**.order\_id = order\_id;

}

**public** String getStatus() {

**return** status;

}

**public** **void** setStatus(String status) {

**this**.status = status;

}

**public** String toString() {

**return** "Order: " + order\_id + " - price = " + price + " , status = " + status;

}

}

**import** java.util.Arrays;

**import** java.util.List;

**import** java.util.function.Consumer;

**import** java.util.function.Predicate;

**public** **class** Print\_orders {

**public** **static** **void** main(String[] args) {

List<orders> order = Arrays.*asList*(

**new** orders(1, 4000, "INPROCESS"),

**new** orders(2, 10000, "ACCEPTED"),

**new** orders(3, 10000, "INPROCESS"),

**new** orders(4, 7000, "COMPLETED"),

**new** orders(5, 10000, "COMPLETED"),

**new** orders(6, 50000, "ACCEPTED"),

**new** orders(7, 60000, "COMPLETED"),

**new** orders(8, 15000, "ACCEPTED"),

**new** orders(9, 10000, "INPROCESS"),

**new** orders(10, 20000, "COMPLETED")

);

System.***out***.println("\n\nPrinting the orders which price is greater 10000 and status is ACCEPTED/COMPLETED ");

*Orderlist*(order, o->(o.getPrice()> 10000 && (o.getStatus()== "ACCEPTED"|| o.getStatus() == "COMPLETED")),o->System.***out***.println(o) );

}

**private** **static** **void** Orderlist(List<orders> order, Predicate<orders> predicate, Consumer<orders> consumer) {

**for**(orders o : order) {

**if**(predicate.test(o))

consumer.accept(o);

}

}

}

1. import java.util.function.Consumer;

import java.util.function.Function;

import java.util.function.Predicate;

import java.util.function.Supplier;

public class LambdaInterfaces {

public static void main(String[] args) {

//consumer functional interface

String str = "Consumer Interface";

Consumer<String>displayConsumer = a->System.out.println(a);

displayConsumer.accept(str.toUpperCase());

//Predicate functional interface

Predicate<String> displaypredicate= p ->str.length() > 10;

System.out.println("Predicate functional interface: "+displaypredicate.test(str));

//Function functional interface

Function<Integer,Double>val = a ->a / 5.0;

System.out.println("Function functional interface: "+val.apply(37));

//Supplier functional interface

Supplier<Float>suppval = () ->Math.max(18.99f, 19.9f);

System.out.println("Supplier functional interface: "+suppval.get());

}

}

1. import java.util.ArrayList;

public class Oddlength {

public static void main(String[] args) {

// TODO Auto-generated method stub

ArrayList<String> words =new ArrayList<String>();

words.add("Hello");

words.add("Welcome");

words.add("Computer");

words.add("System");

words.add("Assignment");

words.add("Collections");

words.removeIf(w->(w.length()%2==0));

words.forEach(System.out::println);

}

}

1. import java.util.Arrays;

import java.util.List;

import java.util.function.Consumer;

public class AppendResult {

public static void main(String[] args) {

// TODO Auto-generated method stub

StringBuilder str=new StringBuilder();

List<String> names =Arrays.asList(

new String("What"), new String("Else"),

new String("Look"), new String("Like"),

new String("Dust"), new String("Or"),

new String("Not"),new String("Edible")

);

for(String n : names) {

str.append(n.charAt(0));

}

//prints the first letters of all the string in the list

printstring(str,c->System.out.println(str));

}

private static void printstring(StringBuilder str,Consumer consumer) {

// TODO Auto-generated method stub

if(str!=null) {

consumer.accept(str);

}

}

}

1. import java.util.Arrays;

import java.util.List;

import java.util.function.UnaryOperator;

class replace implements UnaryOperator<String>{

public String apply(String str) {

return str.toUpperCase();

}

}

public class Unaryopt {

public static void main(String[] args) {

// TODO Auto-generated method stub

List<String> names =Arrays.asList(

new String("What"), new String("Else"),

new String("Look"), new String("Like"),

new String("Dust"), new String("Or"),

new String("Not"),new String("Edible")

);

System.out.println("list before replace operation: "+names);

names.replaceAll(new replace());

System.out.println("Contents of the list after replace operation: \n"+names);

}

}

1. import java.util.Map;

import java.util.TreeMap;

public class Convert {

public static void main(String[] args) {

// TODO Auto-generated method stub

StringBuilder str=new StringBuilder();

Map<Integer,String> map = new TreeMap<>();

map.put(1, "Hello");

map.put(2, "Guys");

map.put(3, "How");

map.put(4, "Are");

map.put(5, "You");

map.put(6, "Doing");

map.put(7, "In");

map.put(8, "Training");

for(Map.Entry<Integer,String>entry:map.entrySet()) {

Integer key = entry.getKey();

String c = entry.getValue();

str.append(key + c);

}

//print result string

System.out.println(str);

}

}

1. import java.util.ArrayList;

import java.util.List;

public class Threadlist {

public static void main(String[] args) {

// TODO Auto-generated method stub

List<Integer> num=new ArrayList<Integer>(){{

add(11);

add(55);

add(37);

add(95);

add(99);

}};

Thread mylambda = new Thread(()->System.out.println(num));

mylambda.run();

}

}