1)

**package** lambdaFunctions;

**public** **class** LambdaExample1 {

**interface** Arithmetic {

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

}

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

Arithmetic addition = (**int** a, **int** b) -> (a + b);

System.***out***.println("Addition = " + addition.operation(5, 5));

Arithmetic subtraction = (**int** a, **int** b) -> (a - b);

System.***out***.println("Subtraction = " + subtraction.operation(10, 3));

Arithmetic multiplication = (**int** a, **int** b) -> (a \* b);

System.***out***.println("Multiplication = " + multiplication.operation(5, 6));

Arithmetic division = (**int** a, **int** b) -> (a / b);

System.***out***.println("Division = " + division.operation(15, 5));

}

}

2)

package lambda;

import java.util.ArrayList;

import java.util.Arrays;

import java.util.List;

import java.util.function.Consumer;

public class MainOrder {

public static void main(String[] args) {

List<Order> or=new ArrayList<>(Arrays.asList(

new Order(1,5000,"Accepted"),

new Order(2,15000,"Completed"),

new Order(3,26000,"Pending"),

new Order(4,10000,"Accepted"),

new Order(5,32000,"Accepted")

));

for(Order o:or)

System.out.println(o);

System.out.println("Orders having two criteria");

or.stream()

.forEach(i -> {

if(i.getPrice()>10000 && (i.getStatus()=="Accepted" || i.getStatus()=="Completed"))

System.out.println(i);

});

}

}

3)

**package** lambda;

**import** java.util.Arrays;

**import** java.util.List;

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

**public** **class** PredicateEx {

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

{

List<String> names = Arrays.*asList*("Java","JavaCore","Python","JavaString","CSharp");

Predicate<String> p = (s)->s.startsWith("J");

**for** (String st:names)

{

**if** (p.test(st))

System.***out***.println(st);

}

}

}

}

**package** lambda;

**import** java.util.function.Supplier;

**public** **class** SupplierEx {

**static** String *product* = "Android";

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

Supplier<Boolean> boolSupplier = () -> *product*.length() == 10;

Supplier<Integer> intSupplier = () -> *product*.length() - 2;

Supplier<String> supplier = () -> *product*.toUpperCase();

System.***out***.println(boolSupplier.get());

System.***out***.println(intSupplier.get());

System.***out***.println(supplier.get());

}

}

**package** lambda;

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

**public** **class** ConsumerEx {

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

{

Consumer<Integer> display = a -> System.***out***.println(a);

display.accept(10);

}

}

**package** lambda;

**import** java.util.function.Function;

**public** **class** FunctionEx {

**static** String show(String message){

**return** "Hello "+message;

}

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

Function<String, String> fun = FunctionEx::*show*;

System.***out***.println(fun.apply("Peter"));

}

}

4)

package lambdaFunctions;

import java.util.ArrayList;

import java.util.Arrays;

import java.util.List;

public class RemoveOdd {

public static void main(String[] args) {

List<Integer> numbers = new ArrayList<>(Arrays.asList(1,2,3,4,5,6,7,8,9,10));

numbers.removeIf( number -> number%2 != 0 );

System.out.println(numbers);

}

}

5)

**package** lambda;

**import** java.util.Arrays;

**import** java.util.List;

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

**public** **class** FirstLetter {

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

{

List<String> strings = Arrays.*asList*("RAJ", "AIRA", "ISHAN", "NAVEEN", "TOM");

StringBuilder sb = **new** StringBuilder();

Consumer<String> consumer = c -> sb.append(c.charAt(0));

strings.forEach(s -> consumer.accept(s));

System.***out***.println("String: " + sb);

}

}

6)

package lambdaFunctions;

import java.util.ArrayList;

import java.util.function.UnaryOperator;

class Op implements UnaryOperator<String> {

public String apply(String str) {

return str.toUpperCase();

}

}

public class Test {

public static void main(String[] args) throws CloneNotSupportedException {

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

list.add("Java");

list.add("JavaScript");

System.out.println("Contents of the list: "+list);

list.replaceAll(new Op());

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

}

}

7)

package lambda;

import java.util.HashMap;

import java.util.Map;

import java.util.function.BiConsumer;

public class Map7 {

public static void main(String[] args) {

Map<Integer, String> words = new HashMap<Integer, String>();

words.put(1, "Raj");

words.put(2, "Navya");

words.put(3, "Aishaa");

words.put(4, "Wiz");

words.put(5, "vaish");

words.put(6, "Ram");

words.put(7, "Param");

words.put(8, "\_\_");

StringBuilder str = new StringBuilder();

BiConsumer<Integer,String> consumer = (key, value) -> str.append(key + value);

words.entrySet().forEach(e -> consumer.accept(e.getKey(), e.getValue()));

System.out.println("Final string : " + str);

}

}

8)

**package** lambda;

**import** java.util.ArrayList;

**import** java.util.List;

**public** **class** Lambda8 {

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

List<Integer> numlist = **new** ArrayList<>();

numlist.add(1);

numlist.add(2);

numlist.add(3);

numlist.add(4);

numlist.add(5);

numlist.add(6);

Thread thread = **new** Thread(() -> numlist.forEach( num -> System.***out***.println(num)));

System.***out***.println("Number list is: ");

thread.run();

}

}