Rajalakshmi Engineering College

Name: Suria Prakash
Email: 241501225@rajalakshmi.edu.in
Roll no: 241501225
Phone: 8056422252
Branch: REC
Department: AI & ML - Section 4
Batch: 2028
Degree: B.E - AI & ML

Scan to verify results

2024_28_III_OOPS Using Java Lab

REC_Week 12_Java_Lamba Expressions_MCQ

Attempt : 1
Total Mark : 10
Marks Obtained : 9

Section 1 : MCQ

1. Can a lambda expression have more than one parameter?

Answer

Yes, it can have multiple parameters

Status : Correct ***Marks : 1/1***

2. Which of the following interfaces is NOT a functional interface in Java?

Answer

Iterable

Status : Correct ***Marks : 1/1***

3. What is a lambda expression in Java?

Answer

A way to define anonymous methods

Status : Correct ***Marks : 1/1***

4. What is the return type of a lambda expression in Java?

Answer

The return type is inferred from the context

Status : Correct ***Marks : 1/1***

5. What is the syntax for a basic lambda expression in Java?

Answer

(parameters) -> expression

Status : Correct ***Marks : 1/1***

6. Which functional interface in Java takes two arguments and returns a result?

Answer

BiFunction

Status : Correct ***Marks : 1/1***

7. Which functional interface is commonly used with lambda expressions in Java?

Answer

Runnable

Status : Correct ***Marks : 1/1***

8. Can a lambda expression in Java have a body with multiple statements?

Answer

Yes, if the statements are enclosed in curly braces

Status : Correct ***Marks : 1/1***

9. Can a lambda expression in Java have a body with multiple statements?

Answer

Yes, if the statements are enclosed in curly braces

Status : Correct ***Marks : 1/1***

10. Which of the following is a valid lambda expression in Java?

Answer

(x) -> x * 2

Status : Wrong ***Marks : 0/1***

Rajalakshmi Engineering College

Name: Suria Prakash
Email: 241501225@rajalakshmi.edu.in
Roll no: 241501225
Phone: 8056422252
Branch: REC
Department: AI & ML - Section 4
Batch: 2028
Degree: B.E - AI & ML

Scan to verify results

2024_28_III_OOPS Using Java Lab

2028_REC_OOPS using Java_Week 12_Q1

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

Sabrina is working on a project that involves analyzing a set of numbers. In her exploration, she encounters scenarios where extracting even numbers and finding their sum is essential.

Create a program that calculates the sum of even numbers from a given array of integers using a lambda expression.

Input Format

The first line of input consists of an integer N, representing the size of the array.

The second line consists of N space-separated integers, representing the elements of the array.

Output Format

The output prints the sum of the even integers from the array.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 3
29 37 45

Output: 0

Answer

```
import java.util.*;
import java.util.stream.*;

public class Main {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      int n = sc.nextInt();
      int[] arr = new int[n];

      for(int i = 0; i < n; i++) {
         arr[i] = sc.nextInt();
      }

      int sum = Arrays.stream(arr)
                  .filter(num -> num % 2 == 0)
                  .sum();

      System.out.println(sum);
      sc.close();
   }
}
```

Status : Correct ***Marks : 10/10***

Rajalakshmi Engineering College

Name: Suria Prakash
Email: 241501225@rajalakshmi.edu.in
Roll no: 241501225
Phone: 8056422252
Branch: REC
Department: AI & ML - Section 4
Batch: 2028
Degree: B.E - AI & ML

Scan to verify results

2024_28_III_OOPS Using Java Lab

2028_REC_OOPS using Java_Week 12_Q2

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

Alex is learning about Java's functional interfaces and lambda expressions.

He wants to write a simple program that prints the square of each number in an array using a predefined functional interface.

Help Alex complete this task using the Consumer functional interface.

Input Format

- The first line contains an integer N, the number of elements in the array.
- The second line contains N space-separated integers.

Output Format

- Print the squares of all elements in the array, separated by a space.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 4
1 2 3 4
Output: 1 4 9 16

Answer

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

public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        int[] arr = new int[n];

        for(int i = 0; i < n; i++) {
            arr[i] = sc.nextInt();
        }

        Consumer<Integer> printSquare = num -> System.out.print(num * num + " ");

        for(int num : arr) {
            printSquare.accept(num);
        }

        System.out.println();
        sc.close();
    }
}
```

Status : Correct ***Marks : 10/10***

Rajalakshmi Engineering College

Name: Suria Prakash
Email: 241501225@rajalakshmi.edu.in
Roll no: 241501225
Phone: 8056422252
Branch: REC
Department: AI & ML - Section 4
Batch: 2028
Degree: B.E - AI & ML

Scan to verify results

2024_28_III_OOPS Using Java Lab

2028_REC_OOPS using Java_Week 12_Q3

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

In the mystical realm of programming, there exists a magical incantation to reveal hidden words.

Elara, the skilled enchantress, wishes to summon a word using her spell and then reverse its characters to uncover its enchanted reflection.

Write a program that uses the predefined functional interface Supplier<String> and a lambda expression to:

Supply (generate) a string, and

Display its reversed form.

Input Format

No input is required from the user.

The string must be supplied internally using a Supplier<String>.

Output Format

Print the reversed version of the supplied string.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: Wizard!!
Output: !!draziW

Answer

```
import java.util.function.Supplier;
import java.util.Scanner;

public class Main {
   public static void main(String[] args) {
            Scanner sc = new Scanner(System.in);
      String input = sc.nextLine();
      // Using Supplier functional interface with lambda expression
      Supplier<String> wordSupplier = () -> input;

      // Get the string from supplier
      String word = wordSupplier.get();

      // Reverse the string
      String reversed = new StringBuilder(word).reverse().toString();

      // Display the reversed string
      System.out.println(reversed);
   }
}
```

Status : Correct ***Marks : 10/10***