

Umar Ali

BS Software Engineering Fall 2022 Red
B22F0088SE094@fecid.paf-iast.edu.pk
umarrali.pk@gmail.com

16 January 2025

Lab 1**Software Construction and Development**

Topic:

Java Basics

Submitted to:

Dr Nabeel Ahmed

Ms Zainab

Problem #1:

Find the largest number from 3 numbers.

Solution:

Code:

```
1  public class largest {  
    Run main | Debug main  
2  public static void main(String[] args) {  
3      int num1 = 10;  
4      int num2 = 25;  
5      int num3 = 15;  
6  
7      int largest;  
8  
9      if (num1 >= num2 && num1 >= num3)  
10         largest = num1;  
11     else if (num2 >= num1 && num2 >= num3)  
12         largest = num2;  
13     else  
14         largest = num3;  
15  
16     System.out.println("The largest number among " + num1 + ", " + num2 + " and " + num3 + " is: " + largest);  
17 }  
18 }
```

Output:

```
The largest number among 10, 25 and 15 is: 25  
[Done] exited with code=0 in 0.962 seconds
```

Explanation:

This code compares the first number with the second and third and updates the value of the variable as per the condition. If the condition is not satisfied, then it jumps to the next and in the same scenario it finds out the largest value.

Problem #2:

Program to reverse a string.

Solution:

Code:

```
1  public class reverse {  
    Run main | Debug main  
2      public static void main(String[] args) {  
3          String str = "Hello World";  
4          String reversed = new StringBuilder(str).reverse().toString();  
5          /*  
6              for (int i = str.length() - 1; i >= 0; i--) {  
7                  //why we cant do reversed+=str[i]?  
8                  /*we can not do this with above method because  
9                      in java strings are immutable and we can not change the value of string once it is created  
10                     reversed+=str.charAt(i);  
11                 }  
12             */  
13         }  
14         System.out.println("The Original string is: "+str);  
15         System.out.println("Reversed string: " + reversed);  
16     }  
17 }  
18 }
```

Output:

```
The Original string is: Hello World  
Reversed string: dlrow olleH  
  
[Done] exited with code=0 in 0.836 seconds
```

Explanation:

This code uses a pre built function from StringBuilder which helps us to reverse the string

There is also a second solution present in the code in a commented area in which we can manually reverse the string by traversing from end towards start and by appending every character from end to start at our new variable called reversed in the code.

Problem #3:

Program to calculate the sum of digits of numbers

Solution:

Code:

```
public class sum{
    Run main | Debug main
    public static void main(String[] args){
        int number=4245;
        int sum=0;
        while(number>0){
            int modl=10;
            sum=sum+number%modl;
            number=number/10;
            System.out.println(sum);
        }

        System.out.println("The sum of digits of number is: "+sum);
    }
}
```

Output:

```
5
9
11
15
The sum of digits of number is: 15
[Done] exited with code=0 in 0.78 seconds
```

Explanation:

At each and every iteration the code prints the updated results. I have used a variable named `modl` whose value is 10 by default. At every iteration it extract the last digit of number and after that the number is divided by 10 to eliminate that index and we continue repeating the process until or unless our number is greater then 0.

Problem #4:

Multiplication Tables of given number

Solution:

Code:

```
import java.util.Scanner;

public class table{
    Run main | Debug main
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter the number to print the table: ");
        int num = scanner.nextInt();
        for(int i=1;i<=10;i++){
            System.out.println(num+" X "+i+" = "+num*i);
            scanner.close();
        }
    }
}
```

Output:

```
Enter the number to print the table:
9
9 X 1 = 9
9 X 2 = 18
9 X 3 = 27
9 X 4 = 36
9 X 5 = 45
9 X 6 = 54
9 X 7 = 63
9 X 8 = 72
9 X 9 = 81
9 X 10 = 90
```

The loop iterates for 10 times from 1 to 10 and at every iteration it multiplies the input value to i integer of loop which result in the answer and with help of some formatting in `System.out.println()` statement we get the above output

Problem #5:

Find Smallest Number from given 3 numbers.

Solution:

Code:

```
public class smallest {  
    Run main | Debug main  
    public static void main(String[] args) {  
        int num1 = 10;  
        int num2 = 25;  
        int num3 = 15;  
  
        int smallest;  
  
        if (num1 <= num2 && num1 <= num3)  
            smallest = num1;  
        else if (num2 <= num1 && num2 <= num3)  
            smallest = num2;  
        else  
            smallest = num3;  
  
        System.out.println("The Smallest number among " + num1 + ", " + num2 + " and " + num3 + " is: " + smallest);  
    }  
}
```

Output:

```
The Smallest number among 10, 25 and 15 is: 10
```

```
[Done] exited with code=0 in 0.811 seconds
```

Explanation:

Logic of this code is similar to the first task but the only magic in this is that we have flipped the comparison operators in our if and else if statements which results in inverse results and ultimately helps us to find the smallest numbers from given 3 numbers.