

LAB Assignment – 4

Subject : Programming with Java

1. Write a program to calculate factorial number.

```
import java.util.Scanner;

public class Factorial{
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int num = scanner.nextInt();
        int factorial = 1;

        for (int i = 1; i <= num; i++) {
            factorial *= i;
        }

        System.out.println("Factorial of " + num + " is " + factorial);
    }
}

/*
output
Enter a number: 5
Factorial of 5 is 120
*/
```

2. Write a program that generates a Fibonacci sequence up to a given number.

```
import java.util.Scanner;

public class Fibonacci {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int n = scanner.nextInt();

        int first = 0, second = 1;

        System.out.print("Fibonacci sequence: ");
        for (int i = 1; i <= n; i++) {
            System.out.print(first + " ");
            int next = first + second;
            first = second;
        }
    }
}
```

```

        second = next;
    }
}

/*
OUTPUT
Enter a number: 6
Fibonacci sequence: 0 1 1 2 3 5
*/

```

3. Write a program to calculate the power of a number (base raised to an exponent).

```

import java.util.Scanner;

public class Power {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter base: ");
        int base = scanner.nextInt();
        System.out.print("Enter exponent: ");
        int exponent = scanner.nextInt();

        int result = 1;
        for (int i = 1; i <= exponent; i++) {
            result *= base;
        }

        System.out.println(base + " raised to the power of " + exponent + " is " + result);
    }
}

/*
OUTPUT:
Enter base: 2
Enter exponent: 3
2 raised to the power of 3 is 8
*/

```

4. Write a program that demonstrates autoboxing (automatic conversion of primitive types to wrapper classes) and unboxing (automatic conversion of wrapper objects back to primitives).

```

public class AutoboxingUnboxing {
    public static void main(String[] args) {
        // Autoboxing: Primitive to Wrapper class
        int num = 10;
        Integer integerObj = num; // Automatically converted to Integer
object

        // Unboxing: Wrapper class to Primitive
        int num2 = integerObj; // Automatically converted back to int

        System.out.println("Autoboxing: " + integerObj);
        System.out.println("Unboxing: " + num2);
    }
}

/*
OUTPUT
Autoboxing: 10
Unboxing: 10
*/

```

5. Write a program that takes a string as input and outputs the reversed string.

```

import java.util.Scanner;

public class ReverseString {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String input = scanner.nextLine();

        String reversed = new StringBuilder(input).reverse().toString();
        System.out.println("Reversed string: " + reversed);
    }
}

/*
OUTPUT:
Enter a string: hello
Reversed string: olleh
*/

```

6. Write a program that counts the number of vowels and consonants in a given string.

```
import java.util.Scanner;

public class VowelConsonantCounter {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String input = scanner.nextLine();

        int vowels = 0, consonants = 0;
        input = input.toLowerCase();

        for (int i = 0; i < input.length(); i++) {
            char ch = input.charAt(i);
            if (ch >= 'a' && ch <= 'z') {
                if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch ==
'u') {
                    vowels++;
                } else {
                    consonants++;
                }
            }
        }

        System.out.println("Vowels: " + vowels);
        System.out.println("Consonants: " + consonants);
    }
}

/*
OUTPUT
Enter a string: Hello World
Vowels: 3
Consonants: 7
*/
```

7. Write a program to extract specific tokens (e.g., the first or third token) from a string.

Example: "name=ABC;age=30;city=Rajkot" → extract "age=30" (2nd token).

```
public class ExtractToken {
    public static void main(String[] args) {
```

```

String input = "name=ABC;age=30;city=Rajkot";
String[] tokens = input.split(";");

// Extracting 2nd token
String secondToken = tokens[1];
System.out.println("Extracted token: " + secondToken);
}
}
/*
OUTPUT
Extracted token: age=30
*/

```

8. Write a program that reverses a string using a stack. `import`

```

java.util.Stack;

public class ReverseStringWithStack {
    public static void main(String[] args) {
        String input = "Hello";
        Stack<Character> stack = new Stack<>();

        // Push characters to stack
        for (int i = 0; i < input.length(); i++) {
            stack.push(input.charAt(i));
        }

        // Pop characters to form reversed string
        StringBuilder reversed = new StringBuilder();
        while (!stack.isEmpty()) {
            reversed.append(stack.pop());
        }

        System.out.println("Reversed string: " + reversed);
    }
}

/*
OUTPUT
Reversed string: olleH
*/

```

9. Write a program to demonstrate Queue operation.

```
import java.util.LinkedList;
import java.util.Queue;

public class QueueOperations {
    public static void main(String[] args) {
        Queue<Integer> queue = new LinkedList<>();

        // Enqueue operations (Adding elements)
        queue.add(10);
        queue.add(20);
        queue.add(30);

        System.out.println("Queue: " + queue);

        // Dequeue operation (Removing an element)
        System.out.println("Dequeued element: " + queue.poll());

        System.out.println("Queue after dequeue: " + queue);
    }
}
/*
OUTPUT
Queue: [10, 20, 30]
Dequeued element: 10
Queue after dequeue: [20, 30]
*/
```