JAVA LAB PRACTICAL

ASSIGNMENT - 1

Done by – Harsh Saini.

Roll no. – 24/SCA/BCA(AI&ML)/025.

Class – BCA 2-C.

Q1 – Write a program to find the average and the sum of the N numbers using Command Line argument?

```
public class SumAndAverage {
public static void main(String[] args) {
int sum = 0;
int count = args.length;
for (String num : args) {
sum += Integer.parseInt(num);
}
double average = (double) sum / count;
System.out.println("Sum: " + sum);
System.out.println("Average: " + average);
}
}
```

OUTPUT =

```
Sum: 0
Average: NaN

=== Code Execution Successful ===
```

Q2 – Write a program to demonstrate type casting?

```
public class TypeCastingDemo {
public static void main(String[] args) {
int num = 10;
double d = num;
System.out.println("Implicit Casting (int to double): " + d);
double x = 10.5;
int y = (int) x;
System.out.println("Explicit Casting (double to int): " + y);
}
}
```

OUTPUT =

```
Implicit Casting (int to double): 10.0
Explicit Casting (double to int): 10
```

Q3 – Write a program to generate prime numbers between 1 to given number?

```
public class PrimeNumbers {
public static void main(String[] args) {
int n = 200;
System.out.println("Prime numbers between 1 and " + n + " are:");
for (int i = 2; i <= n; i++) {
if (isPrime(i)) {
System.out.print(i + " ");
}
}
}
static boolean isPrime(int num) {
if (num < 2) return false;
for (int i = 2; i * i <= num; i++) {
if (num % i == 0) return false;
}
return true;
```

}

}

OUTPUT =

```
Prime numbers between 1 and 200 are:
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97 101 103
    107 109 113 127 131 137 139 149 151 157 163 167 173 179 181 191 193 197 199
=== Code Execution Successful ===
```

Q4 – Write a program to demonstrate Nested Switch?

```
import java.util.Scanner;

public class NestedSwitchDemo {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.println("Enter department (BCA, Btech): ");

String dept = scanner.next();

System.out.println("Enter year (1-4): ");

int year = scanner.nextInt();

switch (dept.toUpperCase()) {

case "BCA":

switch (year) {

case 1: System.out.println("Subjects: Math, Physics"); break;

case 2: System.out.println("Subjects: Data Structures, OOPs"); break;

case 3: System.out.println("Subjects: DBMS, Networks"); break;

case 4: System.out.println("Subjects: AI, Cloud Computing"); break;

default: System.out.println("Invalid year.");

}

break;

case "Btech":

switch (year) {

case 1: System.out.println("Subjects: Math, CS"); break;

case 2: System.out.println("Subjects: AI, Digital Electronics"); break;

case 3: System.out.println("Subjects: DSA, MS"); break;

case 4: System.out.println("Subjects: WT, AI&ML"); break;

default: System.out.println("Invalid year.");
```

```
}
break;
default:
System.out.println("Invalid department.");
}
scanner.close();
}
}
```

OUTPUT =

```
Enter department (BCA, Btech):
Btech
Enter year (1-4):
1
Invalid department.

=== Code Execution Successful ===
```

Q5 – Write a program to calculate area of circle using Radius?

```
import java.util.Scanner;
public class CircleArea {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter radius: ");
double radius = scanner.nextDouble();
double area = Math.PI * radius * radius;
System.out.println("Area of the circle: " + area);
scanner.close();
}
}
```

OUTPUT =

```
Enter radius: 25
Area of the circle: 1963.4954084936207

=== Code Execution Successful ===
```

Q6 – Write a program to find GCD of two numbers?

```
import java.util.Scanner;
public class GCD {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter first number: ");
int a = scanner.nextInt();
System.out.print("Enter second number: ");
int b = scanner.nextInt();
int gcd = findGCD(a, b);
System.out.println("GCD of " + a + " and " + b + " is: " + gcd);
scanner.close();
}
static int findGCD(int a, int b) {
while (b != 0) {
int temp = b;
b = a % b;
a = temp;
}
return a;
}
}
```

OUTPUT =

```
Enter first number: 12
Enter second number: 52
GCD of 12 and 52 is: 4

=== Code Execution Successful ===
```

Q7 – Write a program to generate pyramid of stars using nested for loops?

```
public class Pyramid {
public static void main(String[] args) {
int rows = 5;
for (int i = 1; i <= rows; i++) {
for (int j = rows - i; j > 0; j--) {
System.out.print(" ");
}
for (int k = 1; k <= (2 * i - 1); k++) {
System.out.print("*");
}
System.out.println();
}
}
}
```

OUTPUT =

```
    *
   ***
  *****
 *******
*********

=== Code Execution Successful ===
```

Q8 – Write a program to reversed pyramid for loops and decrement operator?

```
public class ReversedPyramid {
```

```
public static void main(String[] args) {
int rows = 5;
for (int i = rows; i >= 1; i--) {
for (int j = 0; j < rows - i; j++) {
System.out.print(" ");
}
for (int k = (2 * i - 1); k > 0; k--) {
System.out.print("*");
}
System.out.println();
}
}
}
```

OUTPUT:

```
*********
 *******
  *****
   ***
    *

=== Code Execution Successful ===
```

Q9 – Write a program to find the factorial of a given number using recursion?

```
public class FactorialRecursion {
public static int factorial(int n) {
if (n == 0 || n == 1)
return 1;
return n * factorial(n - 1);
}
public static void main(String[] args) {
```

```
int num = 7;
System.out.println("Factorial of " + num + " is: " + factorial(num));
}
}
```

OUTPUT =

```
Factorial of 7 is: 5040

=== Code Execution Successful ===
```

Q10 – Write a program to design using abstract methods and abstract classes?

```
abstract class Animal
{
  public abstract void animalSound();
   public void sleep()
 {
   System.out.println("Zzz");
 }
}
class Pig extends Animal
{
 public void animalSound()
{
     System.out.println("The pig says: wee wee");
 }
}
class Main {
 public static void main(String[] args)
```

```
  {
    Pig myPig = new Pig();
    myPig.animalSound();
    myPig.sleep();
  }
}
```

```
Output

The pig says: wee wee
Zzz

=== Code Execution Successful ===
```

Q11 – Write a program to count the number of objects created for a class using static member function?

```
class ObjectCounter {
    private static int count = 0;

    public ObjectCounter() {
        count++;
    }

    public static int getObjectCount() {
        return count;
    }

    public static void main(String[] args) {
        ObjectCounter obj1 = new ObjectCounter();
        ObjectCounter obj2 = new ObjectCounter();
        ObjectCounter obj3 = new ObjectCounter();
```

```
        ObjectCounter obj4 = new ObjectCounter();

        System.out.println("Number of objects created: " + ObjectCounter.getObjectCount());
    }
}
```

OUTPUT =

```
Number of objects created: 4

=== Code Execution Successful ===
```

Q12 – Write a program to demonstrate the use of function overloading?

```
class FunctionOverloading {

    public void display(int num) {
        System.out.println("Integer: " + num);
    }

    public void display(String text) {
        System.out.println("String: " + text);
    }

    public void display(int num1, int num2) {
        System.out.println("Sum: " + (num1 + num2));
    }

    public static void main(String[] args) {
        FunctionOverloading obj = new FunctionOverloading();

        obj.display(10);
```

```
        obj.display("Hello");
        obj.display(5, 15);
    }
}
```

OUTPUT =

```
Integer: 10
String: Hello
Sum: 20

=== Code Execution Successful ===
```