

## Installation of Eclipse

Eclipse is a free open source platform Integrated Development Environment (IDE) with the help of which applications are made using the Java programming language and others such as C/C++, PERL, Python, Ruby, etc.

- Steps to download Eclipse IDE on windows

1. Open browser and navigate to '<http://www.eclipse.org>'
2. Click on 'Download' to start downloading the installer.
3. Click on 'Download x86\_64'
4. Click on 'Download'. This will download the .exe file.
5. In the file explorer, run 'eclipse-inst-jre-1.8-win64.exe'  
This will install Eclipse.
6. Click on 'Eclipse IDE for Java Developers'
7. Click on 'Install'
8. Click on 'Create a new Java project'.  
Now, you can start programming.

Experiment - 1

Aim: Generate a random number up to 100 and print whether it is prime or not.

```

import java.util.Random;
public class Main {
    public static void main(String[] args) {
        Random rand = new Random();
        int randomNum = rand.nextInt(2, 100);
        boolean flag = true;

        for (int i = 2; i <= Math.sqrt(randomNum); i++) {
            if (randomNum % i == 0) {
                flag = false;
                break;
            }
        }
    }
}

```

```

if (flag) System.out.printf("%d is a prime number", randomNum);
else System.out.printf("%d is not a prime number", randomNum);

```

$$\begin{array}{r} \textcircled{1} \\ 5 | 324 \end{array}$$

Experiment - 2

- Aim: A. Design a program to generate first 10 terms of Fibonacci Series

```
public class Main {
    public static void main (String [] args) {
        int n = 10; System.out.print ("Fibonacci sequence of to 10:");
        int a = 0, b = 1, c, i;
        System.out.print (a + " ");
        for (i = 0; i < n; i++) {
            c = a + b;
            a = b;
            b = c;
            System.out.print (c + " ");
        }
    }
}
```

~~(a)  $b = a + b$~~

B. Design a program to find factorial of a number using Recursion.

```
import java.util.Scanner;
public class Main {
    public static void main (String [] args) {
        Scanner input = new Scanner (System.in);
        System.out.print ("Enter integer for factorial:");
        int n = input.nextInt ();
        System.out.printf ("Factorial of %d is %d", factorial (n), factorial (n));
    }
}
```

```
static int factorial (int n) {
    if (n == 1) {
        return 1;
    }
    return n * factorial (n - 1);
}
```

Experiment -3

Aim: Find the average and sum of array of  $n$  numbers entered by user.

```

import java.util.Scanner;
public class Experiment3 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the number of elements:");
        int n = scanner.nextInt();
        int[] arr = new int[n];
        for (int i = 0; i < n; i++) {
            System.out.print("Enter element " + i + ": ");
            arr[i] = scanner.nextInt();
        }
        int sum = 0;
        for (int i = 0; i < n; i++) {
            sum += arr[i];
        }
        System.out.printf("The sum and average of %d and %d is %.2f", sum, n, (double) sum / n);
    }
}
  
```

(2)  
12/10/2024

Expt. No. \_\_\_\_\_

Date 5/3/2024

Page No. \_\_\_\_\_

### Experiment - 4

Aim: Create a class to find out the area and perimeter of rectangle.

import java.util.Scanner;

public class Experiment4 {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        System.out.print("Enter length: ");

        double length = input.nextDouble();

        System.out.print("Enter breadth: ");

        double breadth = input.nextDouble();

        Rectangle rectangle = new Rectangle(length, breadth);

        System.out.printf("The perimeter of the rectangle  
                               is %.2f", rectangle.perimeter());

        System.out.printf("The area of the rectangle is  
                               %.2f", rectangle.area());

}

}

Class Rectangle {

    double length;

    double breadth;

    Rectangle(double length, double breadth) {

        this.length = length;

        this.breadth = breadth;

}

8.7.0

Teacher's Signature \_\_\_\_\_

Expt. No. \_\_\_\_\_

Date \_\_\_\_\_

Page No. \_\_\_\_\_

double area ( )  $\frac{1}{2}$

return the length  $\times$  the breadth;

3

double perimeter ( )  $\frac{1}{2}$

return  $2 \cdot 0 \times (\text{the length} + \text{the breadth})$ ;

3

5

Q  
10/09/14

Teacher's Signature \_\_\_\_\_

## Experiment-5

Aim: Design a class that performs String operations  
 (Equal, Reverse the String, change case)

```
import java.util.Scanner;
```

```
public class Experiment5 {
```

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

```
        Scanner input = new Scanner (System.in);
```

```
        System.out.print ("Enter your string: ");
```

```
        String userString = input.next();
```

```
        System.out.print ("Enter string for comparison: ");
```

```
        String comparisonString = input.next();
```

```
        StringOperations op = new StringOperations (userString);
```

```
        System.out.println ("Reversed: " + op.reverse());
```

```
        System.out.println ("Equal: " + comparisonString
```

```
            + " ? " + op.equals (comparisonString));
```

```
        System.out.println ("Case changed: " +
```

```
            op.changeCase());
```

3

```
class StringOperations {
```

```
    String string;
```

```
    StringOperations (String string) {this.string = string;}
```

P.T.O.

String reverse() {

    String reversed = "";

    for (int i = this.string.length() - 1; i >= 0; -i) {

        reversed = reversed + this.string.charAt(i);

    }

    return reversed;

}

boolean equals(String other) {

    return this.string.equals(other);

}

String changeCase() {

    String output = "";

    for (int i = 0; i < this.string.length(); ++i) {

        char c = this.string.charAt(i);

        if (Character.isUpperCase(c)) {

            output = output + Character.toLowerCase(c);

        }

        else {

            output = output + Character.toUpperCase(c);

        }

    }

    return output;

}

(D)  
19/03/24

## Experiment - 6

Aim: To demonstrate the use of final keyword with data members, function and class.

// Final Method  
package final;

public class FinalMethod {

final void displayMessage() {

System.out.println("This is a final method.");

public static void main(String[] args) {  
FinalMethod obj = new FinalMethod();  
obj.displayMessage();

}

// Final Class

package final;

public class FinalClassExample {

public static void main(String[] args) {  
System.out.println("This is a final class.");

}

// Final Members  
package final;

public class FinalDataMembers {

final int finalNumber = 42;

public static void main (String [] args) {

FinalDataMembers obj = new FinalDataMembers();

System.out.println ("Final Number: " + obj.finalNumber);

}

Q  
11/03/20

Experiment - 7

Aim: To demonstrate the use of Keywords try, catch, finally, throws and handles.

```

package exception;
import java.util.Scanner;
public class Main {
    public static void divide(float a, float b)
        throws ArithmeticException {
        if (b == 0) {
            throw new ArithmeticException("Can't divide
                by zero");
        }
        float result = a / b;
        System.out.println("The output of " + a + " / "
            + b + " is " + result);
    }
}

public static void main (String args) {
    Scanner input = new Scanner (System.in);
    System.out.println ("Enter first number:");
    float a = input.nextFloat();
    System.out.println ("Enter second number:");
    float b = input.nextFloat();
    try {
        divide (a, b);
    } catch (ArithmaticException e) {
        e.printStackTrace();
    }
    finally {
        System.out.println ("Finishing ...");
    }
}

```

Experiment - 8

Aim: To design a program to demonstrate multithreading using Thread class.

package thread;

```
public class MultiThread {
    public static void main(String[] args) {
        Thread thread1 = new MyThread("Thread 1");
        Thread thread2 = new MyThread("Thread 2");
        thread1.start();
        thread2.start();
    }
}
```

```
class MyThread extends Thread {
    private String ThreadName;
    public MyThread(String name) {
        ThreadName = name;
    }
}
```

```
public void run() {
    System.out.println("Thread " + ThreadName +
        " is running");
    for (int i = 1; i <= 5; i++) {
        System.out.println(ThreadName + " " +
            ":" + i);
    }
}
```

Page No. \_\_\_\_\_

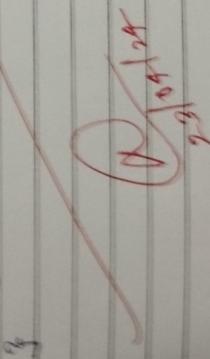
Date \_\_\_\_\_

Page No. \_\_\_\_\_

say q  
Tinread & sleep (1000);  
q

catch (Tinread has been & ) q  
System.out.println ("Tinread " + TinreadName  
+ " is now asleep.");  
q

System.out.println ("Tinread " + TinreadName  
+ " is sleeping.");  
q

✓  
  
D. S. A.

Teacher's Signature \_\_\_\_\_