**AWT & SWING CODE**

import java.awt.\*;

// swing for same code we have to add J in front of every element eg JFrame / JLabel / JTextField

public class LoginForm1 {

public static void main(String[] args) {

Frame f = new Frame("Login Form");

Label l = new Label("Login Form");

l.setBounds(50, 40, 300, 30);

Label l1 = new Label("Username:");

l1.setBounds(50, 90, 80, 30);

TextField t1 = new TextField();

t1.setBounds(150, 90, 200, 30);

Label l2 = new Label("Password:");

l2.setBounds(50, 130, 80, 30);

TextField t2 = new TextField();

t2.setBounds(150, 130, 200, 30);

Button b = new Button("Submit");

b.setBounds(150, 300, 100, 30);

f.add(l);

f.add(l1);

f.add(t1);

f.add(l2);

f.add(t2);

f.add(b);

f.setSize(400, 400);

f.setLayout(null);

f.setVisible(true);

}

}

**Exception Handling**

**Q1. Handling Multiple Exception (OR) Mutiple Catch**

import java.lang.\*;

public class Main {

public static void main(String[] args) {

try {

int[] arr = {1, 2, 3};

System.out.println(arr[5]);

String str = null;

System.out.println(str.length());

String numStr = "abc";

int num = Integer.parseInt(numStr);

System.out.println(num);

int result = 10 / 0;

System.out.println(result);

} catch (ArrayIndexOutOfBoundsException e) {

System.out.println("Error: Array index is out of bounds.");

} catch (NullPointerException e) {

System.out.println("Error: Null pointer exception occurred.");

} catch (NumberFormatException e) {

System.out.println("Error: Invalid number format.");

} catch (ArithmeticException e) {

System.out.println("Error: Division by zero is not allowed.");

}

}

}

Note: Both code are same but here we just added try block

**Q2. Handling Multiple Try Catch (OR) Nested Try Catch**

public class NestedTryCatch {

public static void main(String[] args) {

try {

int[] arr = {1, 2, 3};

System.out.println("Accessing array element: " + arr[5]);

try {

String str = null;

System.out.println("String length: " + str.length());

try {

String numStr = "abc";

int num = Integer.parseInt(numStr);

System.out.println("Converted number: " + num);

} catch (NumberFormatException e) {

System.out.println("Error: Invalid number format in inner try block 2.");

}

} catch (NullPointerException e) {

System.out.println("Error: Null pointer exception in inner try block 1.");

}

} catch (ArrayIndexOutOfBoundsException e) {

System.out.println("Error: Array index is out of bounds in the outer try block.");

}

}

}

Note:Same code but no multiple try block and catch block added all exceptions in single catch block

**Q3. Handling Multiple Exceptions in single Catch**

public class Main {

public static void main(String[] args) {

try {

int[] arr = {1, 2, 3};

System.out.println(arr[5]);

String str = null;

System.out.println(str.length());

String numStr = "abc";

int num = Integer.parseInt(numStr);

System.out.println(num);

int result = 10 / 0;

System.out.println(result);

} catch (ArrayIndexOutOfBoundsException | NullPointerException | NumberFormatException | ArithmeticException e) {

System.out.println("Error: " + e.getMessage());

       }

    }

}

**File Handling**

**Q1. Create a file**

import java.io.File;

import java.io.IOException;

public class Cfile1{

    public static void main(String[] args) throws IOException {

        File f = new File("NewFile.txt");

        if (f.createNewFile()) {

            System.out.println("File " + f.getName() + " is created!");

        } else {

            System.out.println("File already exists!");

        }

    }

}

**Same code but using try catch (OR)**

import java.io.File;

import java.io.IOException;

public class Main {

public static void main(String[] args) {

try {

File f = new File("NewFile.txt");

if (f.createNewFile()) {

System.out.println("File " + f.getName() + " is created!");

} else {

System.out.println("File already exists!");

}

} catch (IOException e) {

System.out.println("An error occurred.");

}

}

}

**Q2. Delete a file**

import java.io.File;

import java.io.IOException;

public class Cfile1{

    public static void main(String[] args) throws IOException {

        File f = new File("NewFile.txt");

        if (f.delete()) {

            System.out.println("File " + f.getName() + " is deleted!");

        } else {

            System.out.println("File already delete!");

        }

    }

}

**Same code but using try catch (OR)**

import java.io.File;

import java.io.IOException;

public class Main {

public static void main(String[] args) {

try {

File f = new File("NewFile.txt");

if (f.delete()) {

System.out.println("File " + f.getName() + " is deleted!");

} else {

System.out.println("File already deleted!");

}

} catch (IOException e) {

System.out.println("An error occurred.");

}

}

}

**Q3.File Attributes**

import java.io.File;

import java.io.IOException;

public class Cfile1{

    public static void main(String[] args) throws IOException {

        File f = new File("NewFile.txt");

        if(f.exists()){

            System.out.println("Name"+f.getName());

            System.out.println("Path"+f.getAbsoluteFile());

            System.out.println("is readable?"+f.canWrite());

            System.out.println("is writeable?"+f.canRead());

            System.out.println("size"+f.length());

        }else{

            System.out.println("File not exists!!");

        }

    }

}

**Q4.Write into a file**

import java.io.File;

import java.io.FileWriter;

import java.io.IOException;

public class Cfile1{

    public static void main(String[] args) throws IOException {

        File f = new File("NewFile.txt");

            FileWriter w = new FileWriter("NewFile.txt");

            w.write("Hello Jay\n");

            w.close();

    }

}

**Same code using try catch (OR)**

import java.io.File;

import java.io.FileWriter;

import java.io.IOException;

public class Cfile1{

    public static void main(String[] args) throws IOException {

        File f = new File("NewFile.txt");

        try {

            FileWriter w = new FileWriter("NewFile.txt");

            w.write("Hello Jay\n");

            w.close();

        } catch (IOException e) {

            System.out.println("Error: " + e.getMessage());

        }

}

}

**Q5. Read a file**

import java.io.BufferedReader;

import java.io.File;

import java.io.FileReader;

import java.io.IOException;

public class Cfile1 {

    public static void main(String[] args) throws IOException {

        File f = new File("NewFile.txt");

        if (f.exists()) {

            FileReader fr = new FileReader(f);

            BufferedReader br = new BufferedReader(fr);

            String line;

            line = br.readLine();

            System.out.println(line);

            br.close();

            fr.close();

        } else {

            System.out.println("File does not exist.");

        }

    }

}