2.13 Finally Block

This section will guide you to:

- Implement **finally{}** block

This guide has two subsections, namely:

2.13.1 Writing code for **finally{}** block

2.13.2 Pushing code to GitHub repository

Step 2.13.1: Writing code for **finally{}** block

- Case 1: Without **Exception**

- Write the code below in a Java file and run it as a Java application:

```
class TestFinallyBlock{
        public static void main(String args[]){
        try{
        int number=25/5;
        System.out.println(number);
         }
        catch(NullPointerException e)
         {
                System.out.println(e);
         }
        finally
                {
System.out.println("The Execution of final block always happen ");
                }
        System.out.println("after final the rest of the code....");
         }
        }
```

- Output:

```
5
The Execution of final block always happen
after final the rest of the code....the code...
```

- Case 2: With **Exception**

  - Write the code below in a Java file and run it as a Java application**:**

```
publicclass TestFinallyBlock1 {

    publicstaticvoid main(String args[]){
            try{
            int number=5/0;
            System.out.println(number);
             }
            catch(NullPointerException e)
             {
                    System.out.println(e);
             }
            finally
             {
    System.out.println("finally block is always executed");
             }
            System.out.println("then rest of the code...");
             }
}
```

Output:

```
finally block is always executed
Exception in thread "main" java.lang.ArithmeticException: / by zeroat
java1.TestFinallyBlock1.main(TestFinallyBlock1.java:7)
```

- Case 3: **Exception** handled

- Write the code below in a Java file and run it as a Java application:

```
publicclass TestFinallyBlock2{
        publicstaticvoid main(String args[]){
        try{
        int number=25/0;
        System.out.println(number);
          }
        catch(ArithmeticException e)
          {
                System.out.println(e);
                  }
        finally
          {
    System.out.println("finally block is always executed");
                  }
        System.out.println("rest of the code...");
          }
}
```

- Output:

```
java.lang.ArithmeticException: / by zero
finally block is always executed
rest of the code...
```

Step 2.13.2: Pushing the code to your GitHub repositories

Open your command prompt and navigate to the folder where you have created your files.

cd java_program

Initialize your repository using the following command:

git init

Add all the files to your git repository using the following command:

git add .

Commit the changes using the following command:

git commit . -m “Changes have been committed.”

Push the files to the folder you initially created using the following command:

git push -u origin master