2.12 Multiple Catch-With Exception Class Hierarchy

This section will guide you to:

- Implement multiple catch-with exception class hierarchy

This guide has three subsections, namely:

2.12.1 Creating a new Java project

2.12.2 Writing code for multiple catch blocks

2.12.3 Pushing the code to your GitHub repository

Step 2.12.1: Creating a new Java project

- Open Eclipse
- Click on File->New->Java Project from the menu bar
- Give the project name as **MultipleCatch** and click **OK**
- Right-click on your project->click on New->click on Class and provide a class name as **multipleCatch** and click **OK**

Step 2.12.2: Writing code for multiple catch blocks

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

```
package io.com;

public class MultipleCatchBlock {

   public static void main(String[] args) {
      try{
         int a[]=new int[5];
         System.out.println(a[10]);
```

```
    }
    catch(ArithmeticException e){
       System.out.println("Arithmetic exception");
    }
    catch(ArrayIndexOutOfBoundsException e){
      System.out.println("ArrayIndexOutOfBounds exception");
    }
    catch(Exception e){
       System.out.println("Parent exception");
    }
    System.out.println("rest of the code");
  }

}
```

Output:

```
Arithmetic Exception
Rest of the code
```

- Write the code given below in a Java file for a nested try-catch block:

```
class Excep1{
 public static void main(String args[]){
try{
try{
System.out.println("going to divide");
intb =39/0;
}catch(ArithmeticException e){System.out.println(e);}

try{
int a[]=newint[5];
a[5]=4;
   }
catch(ArrayIndexOutOfBoundsException e)
{
System.out.println(e);
    }

System.out.println("another statement");
```

```
  }
catch(Exception e)
{
System.out.println("handeled");
}

System.out.println("normal flow..");
 }
 }
```

Output:

```
going to divide
java.lang.ArithmeticException: / by zero
java.lang.ArrayIndexOutOfBoundsException: 5
another statement
normal flow..
```

Step 2.12.3: Pushing the code to your GitHub repositories

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

cd <folder path>

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