

ExceptionHandling codes:

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

    m1();

    System.out.println("continue");
}
static void m1(){
try{
    System.out.println("try continue");
    System.out.println(10/0);

}
catch(ArithmeticException ae){
    try{
        System.out.println("catch block" +(10/0));
    }
    catch(Exception e){System.out.println("inner catch block");}
    finally{

        System.out.println("Finally block");
    }

}

}
```

User defined exception:

// A Class that represents use-defined exception

```
import java.io.*;
```

```
class MyException extends Exception {  
    public MyException(String s) throws IOException  
    {  
        // Call constructor of parent Exception  
        super(s);  
    }  
}
```

// A Class that uses above MyException

```
public class ExceptionDemo2 {  
    // Driver Program  
    public static void main(String args[]) throws IOException  
    {  
        // Throw an object of user defined exception  
        MyException me=new MyException("sudheer defined");  
        try{  
            throw me;  
            //throw new ArithmeticException("My arithmetic");  
        }  
  
        catch(ArrayIndexOutOfBoundsException e){  
            System.out.println("User Defined Exception1"+e.getMessage());  
        }  
        catch(ArithmeticException ae){
```

```

        System.out.println("User Defined Exception2"+ae.getMessage());}
    catch(Exception e){
        System.out.println("User Defined Exception3"+e.getMessage());}

    }
}

```

// checked Exception:

```

import java.io.*;
class ExceptionDemo3{
    public static void main(String args[]){
        try{
            PrintWriter pw=new PrintWriter("sudheer.txt");
            pw.write("Hello world sudheer");
            pw.close();
        }
        catch(FileNotFoundException fe){

            System.out.println("catch block");
        }

    }

}

```

//throws with inheritance

```

import java.io.*;
class A{

```

```
void sleep()throws ArithmeticException{
```

```
    System.out.println("A is sleeping");
```

```
}
```

```
}
```

```
class B extends A{
```

```
void sleep() {
```

```
    System.out.println("B is sleeping");
```

```
}
```

```
}
```

```
class MainDemo{
```

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

```
    A a;
```

```
    a=new B();
```

```
    a.sleep();
```

```
}
```

```
}
```