Day_36_Exception_Handling_Basic

11 April 2022 07:04 PM

Program with exceptions and errors and how skips the flow of execution?

Use below program by Removing Try Catch block.

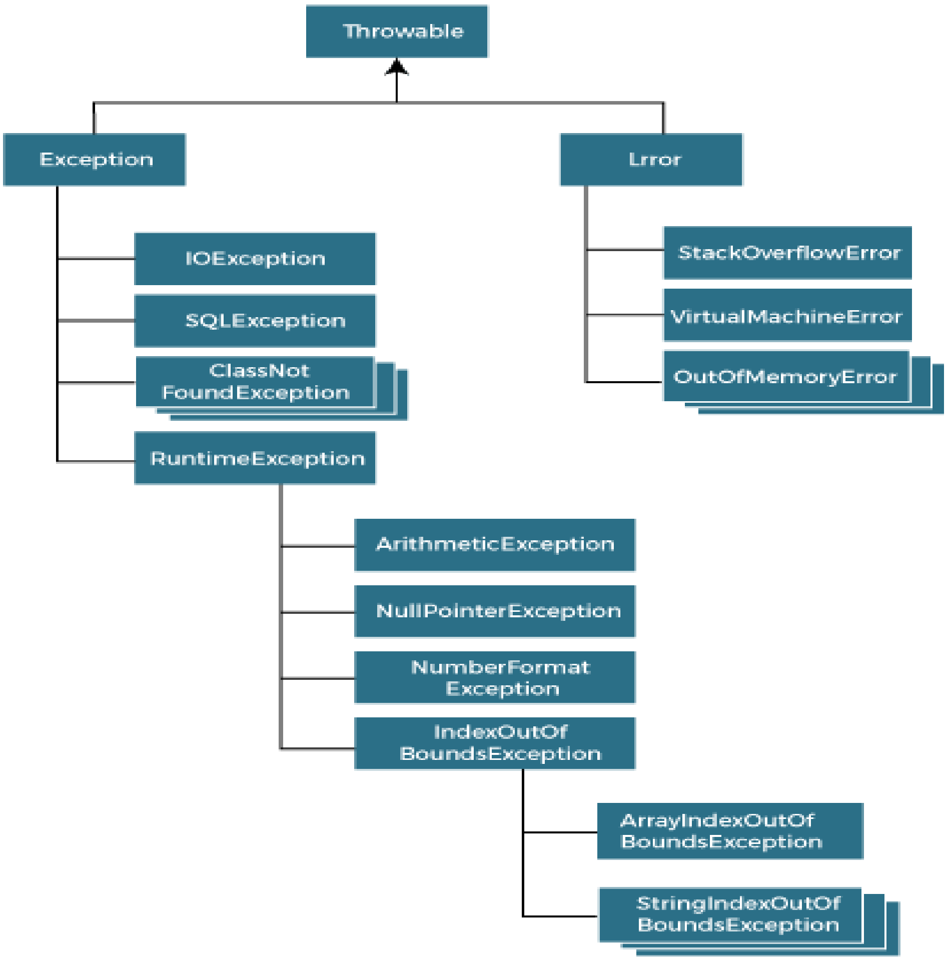
What is Exception ?

Dictionary Meaning: Exception is an abnormal condition.
In Java, an exception is an event that disrupts the normal flow of the program. It is an object which is thrown at runtime.

What is Exception Handling?

The **Exception Handling in Java** is one of the powerful *mechanism to handle the runtime errors* so that the normal flow of the application can be maintained, runtime errors such as ClassNotFoundException, IOException, SQLException, RemoteException,ArrayIndexOutOFBOundExceptio, / by Zero etc.

Hierarchy of Java Exception classes?



Types of Java Exceptions?

There are mainly two types of exceptions: checked and unchecked. An error is considered as the unchecked exception. However, according to Oracle, there are three types of exceptions namely:

1. Checked Exception
2. Unchecked Exception
3. Error

1) Checked Exception :-

The classes that directly inherit the Throwable class except RuntimeException and Error are known as checked exceptions. For example, IOException, SQLException, etc. Checked exceptions are checked at compile-time. And compiler force to handel it.

2) Unchecked Exception :-

The classes that inherit the RuntimeException are known as unchecked exceptions. For example, ArithmeticException, NullPointerException, ArrayIndexOutOfBoundsException, etc. Unchecked exceptions are not checked at compile-time, but they are checked at runtime.

3) Error :-

Error is irrecoverable. Some example of errors are OutOfMemoryError, VirtualMachineError, AssertionError etc.

Java Exception Handling Example.

```
public class WithoughtExceptionHand {


    public void CalculateDivision(int b) {

        int num = 20;
        try {

            System.out.println("Division is -"+ 20/b);

        } catch(Exception e) {
            System.out.println("Exception occured :-" +e);
        }
    }


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

            WithoughtExceptionHand obj = new WithoughtExceptionHand();
        obj.CalculateDivision(10);
        obj.CalculateDivision(20);

        obj.CalculateDivision(0);

        obj.CalculateDivision(5);
        obj.CalculateDivision(2);
        obj.CalculateDivision(1);

        } catch(Exception e) {
            System.out.println(e);
        }
        int[] arr = new int[2];
        arr[0] = 10;
        arr[1] = 20;

        for(int i=0; i <= 10; i++) {
            System.out.println("Numbers - "+i);
        }
        try {

        System.out.println("Array Element - " +arr[0]);
        System.out.println("Array Element - "+arr[1]);
        System.out.println("Array Element - "+arr[2]);

        } catch(Exception e) {
            System.out.println(e);
        }
        for(int i=11; i <= 20; i++) {
            System.out.println("Numbers - " +i);
        }

    }

}
```

Runtime Error Program. (StackOverflowError)

```
public class ErrorTest {
    void m1() {
        System.out.println("In m1 method");
        m2();
    }
    void m2() {
        System.out.println("In m2 method");
        m1();
    }
    public static void main(String[] args) {
        ErrorTest et = new ErrorTest();
        et.m1();
    }
}
```