

Topics



- What Are Exceptions?
- Types of Exceptions

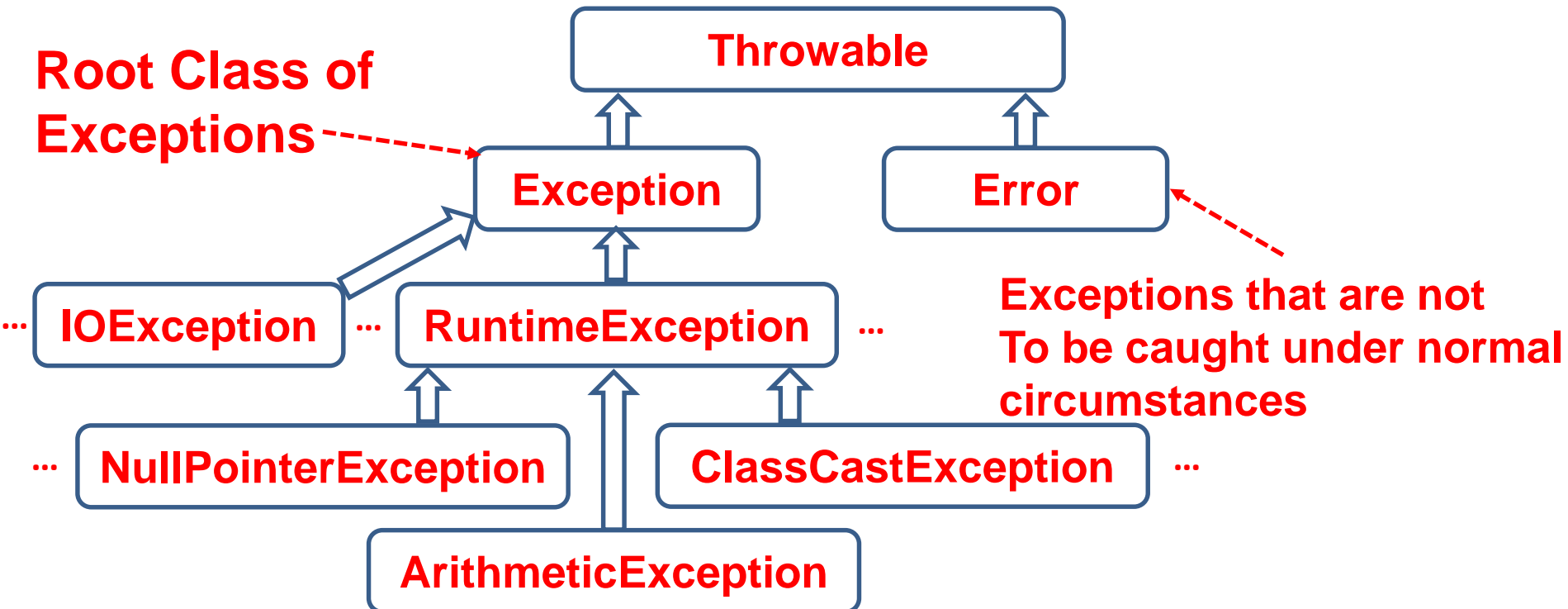
Exceptions



- **Exception** → Abnormal Condition [Occurs at run time]. For Example divide by 0.
- During execution if any exceptional condition occurs then Java Runtime Environment (JRE) i.e. Java Interpreter creates a suitable Exception Object and throws it.
- Every Exception is basically an object belonging to Java's Exception class Hierarchy.
- Exceptions needs to be handled so that appropriate actions can be taken.
- Programmer can also provide exception handling code. However, if there is no exception handling code present during runtime for the thrown exception, then default exception handler will be provided by JRE.
- If thrown Exception is Caught by JRE's Default Exception Handler then the Name of the Exception in string form will be displayed over System.out and the execution of the program will be stopped.

Exception Class Hierarchy

- ❑ **Throwable** is the root class of Exception Hierarchy.
- ❑ **Throwable** has two direct subclasses namely **Exception** and **Error**



Exception Types



1. Checked Exceptions
2. Un-Checked Exceptions

Exception Types :

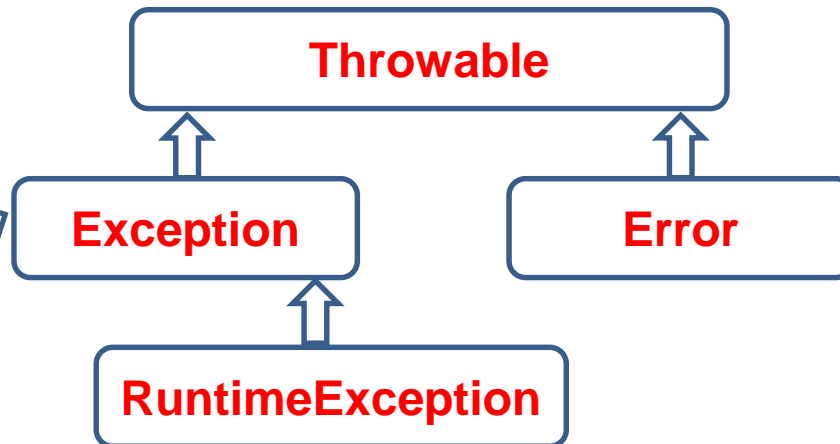
Checked Exceptions



- ❑ Checked at Compile-Time
- ❑ Any Sub-class of Exception Except RuntimeException
- ❑ If a method throws any Checked Exception then Either it has to Handle the Exception or It must specify Exception Using **throws** clause

Checked Exceptions

Any Sub-Class
Of Exception Hierarchy
Except RuntimeException



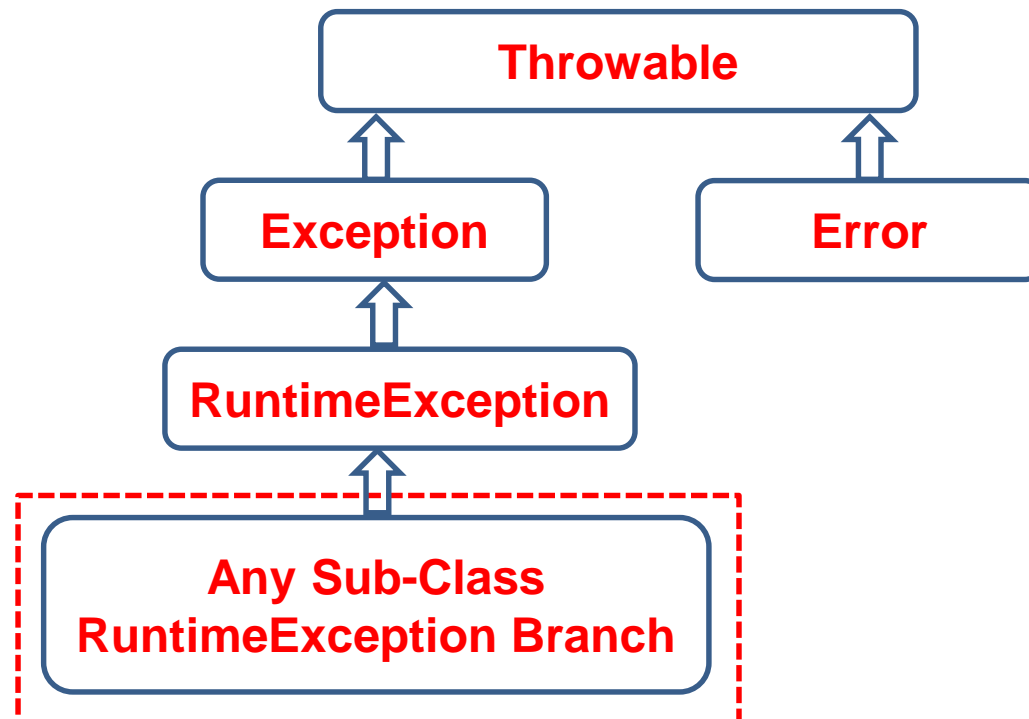
Exception Types :

Un-Checked Exceptions



- ❑ Are Not Checked at Compile-Time. [Automatically Defined For Programs]
- ❑ Any Sub-Class of RuntimeException is by default an Un-Checked Exception

**Un-Checked
Exceptions**



Example 1: Un-Caught Exceptions (Un-Checked Type)



// File Name : ExceptionDemo.java

class ExceptionDemoTest

{

public static void main(String args[])

{

System.out.println("Hello Exceptions");

int d = 0;

int s = 42 / d;

System.out.println("Hello");

}// End of Method

}// End of class

E:\>java ExceptionDemoTest

Hello Exceptions

Exception in thread "main" java.lang.ArithmeticException: / by zero
at ExceptionDemoTest.main(ttj.java:8)

This statement will
results in Exception

Example 2: Un-Caught Exceptions (Un-Checked Type)



```
// File Name : ExceptionDemo.java
class ExceptionDemoTest
{
    public static void main(String args[])
    {
        String s = null;
        System.out.println(s.equals("Java"));
        System.out.println("Hello");
    } // End of Method
} // End of class
```

This statement will
results in Exception

```
E:\>java ExceptionDemoTest
Exception in thread "main" java.lang.NullPointerException
at ExceptionDemoTest.main(ttj.java:7)
```


Example 3: Un-Caught Exceptions (Checked Type)



```
import java.io.*;
class ExceptionDemoTest
{
    public static void main(String args[])
    {
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        String s = br.readLine();
    } // End of Method
} // End of class
```

COMPILATION ERROR

unreported exception java.io.IOException; must be caught or declared to be thrown

```
String s = br.readLine();
```

^

1 error

Example 3: Un-Caught Exceptions (Checked Type) ...



- ❑ If a Method directly throws Any Checked Exception or Calls any other Method which throws a Checked Exception Then
 - Either it has to Handle the Exception OR
 - It Must Specify the Exception Type Using **throws** clause

Method 1: Specifying the Exception Using Throws Clause

```
import java.io.*;
class ExceptionDemoTest
{
    public static void main(String args[]) throws IOException
    {
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        String s = br.readLine();
    } // End of Method
} // End of class
```

Example 3: Un-Caught Exceptions (Checked Type) ...



Method 2: Catch the Exception Using try ... catch

```
import java.io.*;
class ExceptionDemoTest
{
    public static void main(String args[])
    {
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        try
        {
            String s = br.readLine();
        }
        catch(IOException e) {}
    }
} // End of Method
} // End of class
```

Some Built-in Exceptions

Un-Checked Exceptions	Checked Exceptions
<ol style="list-style-type: none">1. <code>ArithmeticException</code>2. <code>ArrayIndexOutOfBoundsException</code>3. <code>ArrayStoreException</code>4. <code>ClassCastException</code>5. <code>NullPointerException</code>6. <code>StringIndexOutOfBoundsException</code>7. <code>NumberFormatException</code>8. <code>IndexOutOfBoundsException</code>	<ol style="list-style-type: none">1. <code>IOException</code>2. <code>CloneNotSupportedException</code>3. <code>ClassNotFoundException</code>

Thank You