NIELIT Gorakhpur

Course Name: A Level (2nd Sem) **Subject: JAVA**
Topic :Exception Handling in Java **Date: 07-04-20**

Exception

An exception is an unwanted or unexpected event, which occurs during the execution of a program i.e at run time that disrupts or stops the normal flow of the program. When an exception occurs program execution gets terminated.

Why Exception occurs?

There can be various reasons that can cause a program to throw exception. For example: Opening a non-existing file in your program, Network connection problem, invalid input data provided by user or accessing the table in database which is not created.

Exception Handling

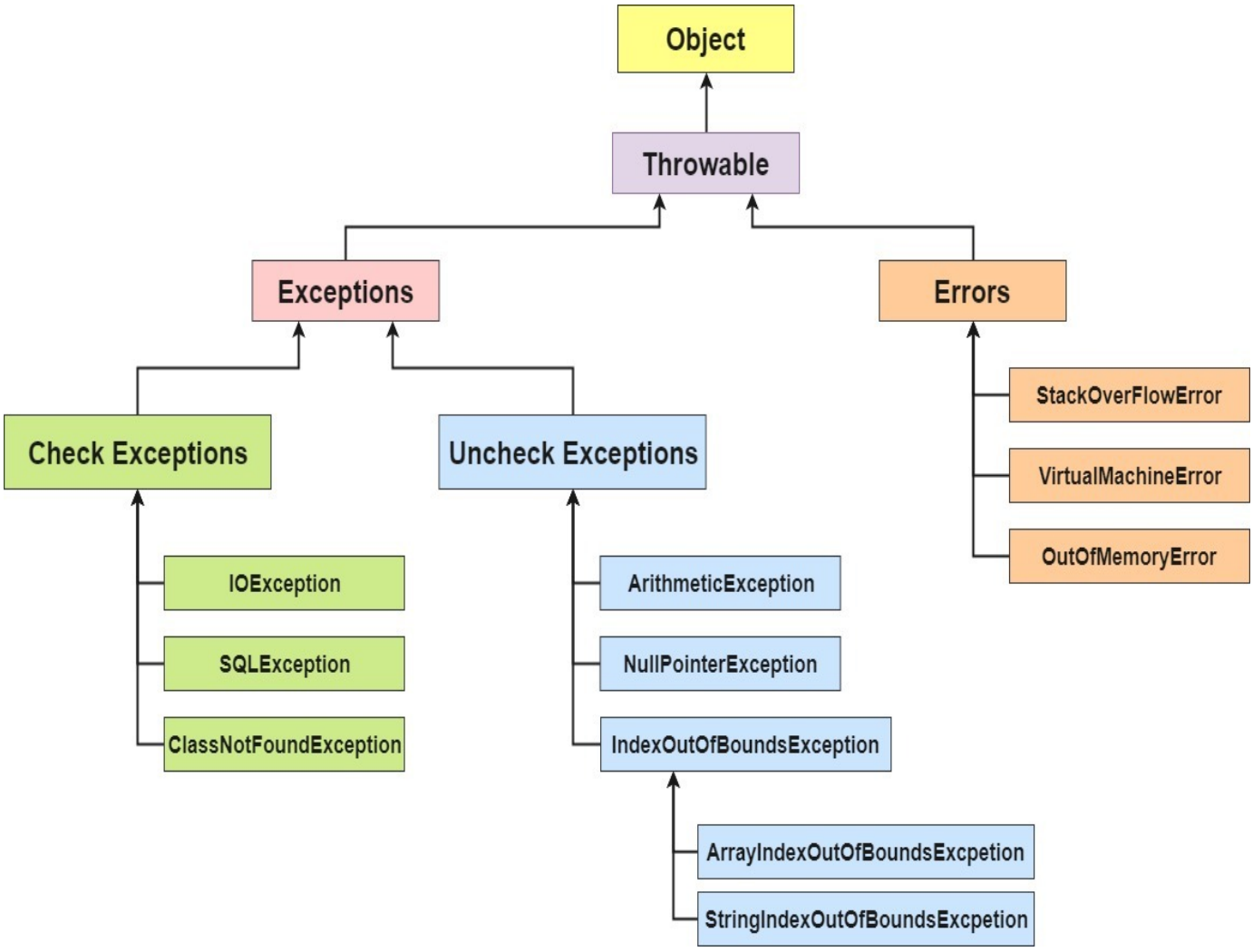
Exception Handling is a mechanism to handle runtime errors such as ClassNotFoundException, IOException, SQLException, RemoteException, etc. By handling the exceptions we can provide a meaningful message to the user about the issue.

Advantages of exception handling

Exception handling ensures that the flow of the program doesn't break when an exception occurs. By handling we make sure that all the statements execute and the flow of program doesn't break.
One of the important purposes of exception handling in Java is to continue program execution after an exception is caught and handled.
The execution of a Java program does not terminate when an exception occurs. Once the exception is resolved, program execution continues till completion.

Java Exception classes:

The java.lang.Throwable class is the root class of Java Exception hierarchy which is inherited by two subclasses: **Exception** and **Error.**



Difference between error and exception

Errors indicate that something very severe occured .The error can not be recovered by the programmer.

Exceptions are events that occur in the code. A programmer can handle such conditions and take necessary corrective actions.

Types of Java Exceptions

1) Checked Exception

All exceptions other than Runtime Exceptions are known as Checked exceptions as the compiler checks them during compilation to see whether the programmer has handled them or not. If these exceptions are not handled/declared in the program, you will get compilation error. For example, SQLException, IOException, ClassNotFoundException etc.

2) Unchecked Exception

Runtime Exceptions are also known as Unchecked Exceptions. These exceptions are not checked at compile-time so compiler does not check whether the programmer has handled them or not but it's the responsibility of the programmer to handle these exceptions and provide a safe exit.For example:ArithmeticException,NullPointerException,ArrayIndexOutOfBoundsException etc.

Problem without Exception Handling

```
class E
{
public static void main(String[] args)
{
int a=10, b=0;
int c=a/b;
System.out.println("Denominator must not be zero");
}
}
```

Here arithmeticException will be occured and the program will be terminated and the compiler will inform about this exception on command prompt.

Exercise:

1. What is exception in java and how java handles it?
2. Discuss the different types of exceptions.