Checked vs Unchecked Exception

Exceptions

In Java exceptions (which differs from the C exception) there are two types of exceptions.

Checked exceptions

Checked exceptions are exceptions that are checked at compile time. If some code in a method throws a checked exception, then the method must either handle the exception by surrounding it with try-catch or has to throw the exception to make the caller of the method handle it.

Exception such as **FileNotFoundException** are checked exception that you must handle.

Checked exceptions are sub-class of Exception that aren't sub-class of RuntimeException

Unchecked exceptions

Unchecked exceptions are the opposite that aren't checked by the compiler, therefore, you are not forced to handle it with try-catch. It is up to the programmer to handle it or throw it.

Exception such as **NullPointerException** are unchecked.

This is why you can use Integer.parseInt without surrounding it with try-catch clause.

Unchecked exceptions are sub-class of RuntimeException

> Fun fact: RuntimeException is actually a sub-class of Exception, however, it is treated differently by the compiler to be unchecked exception.

Revision #1
Created 2023-04-14 12:46:28 UTC by Tamarine
Updated 2023-04-14 13:40:09 UTC by Tamarine