**Exception** means "exceptional condition" and is an occurrence that alters the normal program flow.

Every exception is an instance of a class that has class Exception.

**Error** - unusual situations that are not caused by program errors (JVM running out of memory). Generally, application won't be able to recover from an Error.

If your code does not handle them (and it usually won't), it will still compile with no trouble.

When an Error or a subclass of Error is thrown, it's unchecked.

Error can be thrown using the throw keyword. You're not required to handle them.

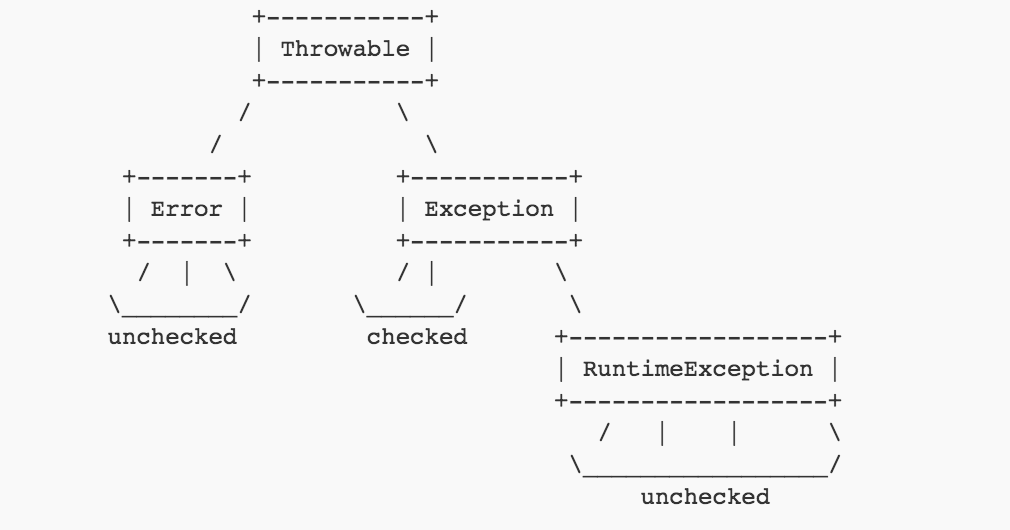
Example - OutOfMemoryError, VirtualMachineError

**Checked** are exceptions which are checked at compile time.

Example - FileNotFoundException, IOException, ClassNotFoundException

**Unchecked** are the exceptions that are not checked at compiled time, exceptions under Error & RuntimeException are unchecked exceptions.

Example - ArithmeticException, NullPointerException, ArrayIndexOutOfBound



**Throw and Throws**

throw keyword is used to explicitly throw Exception from any method or static block

throws keyword, used in method declaration, denoted which Exception can possible be thrown by this method.

|  |  |
| --- | --- |
| **throw** | **throws** |
| throw keyword is used to explicitly throw an exception | throws keyword is used to declare an exception |
| throw is used within the method | throws is used with the method signature |
| Cannot throw multiple exceptions | can declare multiple exceptions for e.g.  public void method()throws IOException,SQLException |