**Exception.**

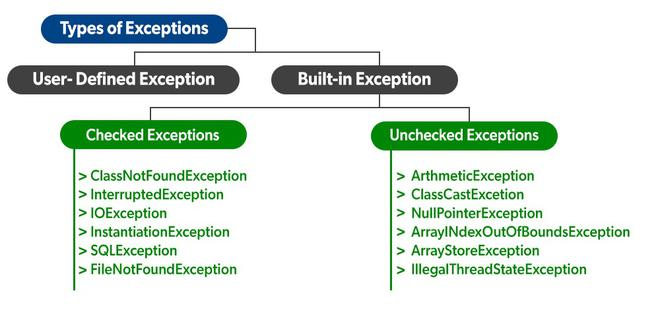
An exception is an abnormal condition that arises in a code sequence at run time. In other words, an exception is a run-time error.

Exceptions can be caught and handled by the program.

When an exception occurs within a method, it creates an object. This object is called the exception object. It contains information about the exception, such as the name and description of the exception and the state of the program when the exception occurred.



**Types of Exceptions**



Exceptions can be categorized in two ways:

**Built-in Exceptions**

Checked Exceptions: Checked exceptions are called compile-time exceptions because these exceptions are checked at compile-time by the compiler.

Unchecked Exceptions: The unchecked exceptions are just opposite to the checked exceptions. The compiler will not check these exceptions at compile time

**User-Defined Exceptions**

Sometimes, the built-in exceptions in Java are not able to describe a certain situation. In such cases, users can also create exceptions, which are called ‘user-defined Exceptions’.

Protect the code that can potentially throw an exception. -- try

Specify what needs to happen on exception. - catch

Specify what needs to happen no matter what. - Finally.

**Syntax:**

try{

// Piece of code that can potentially throw exception

}catch(Exception e){

// Handle the exception

}finally{

// Wrap up/ clean up.

}

try{

// Piece of code that can potentially throw exception

}catch(Exception e1){

// Handle the exception

}catch(Exception e2){

// Handle the exception

}finally{

// Wrap up/ clean up.

}