What is an Exception in Java ?

An exception is an event, which occurs during the execution of a program, that disrupts the normal flow of the program's instructions. The term exception is shorthand for the phrase "exceptional event." Using exceptions helps us:

- Separate out error handling code from rest of the code
- Errors can be propagated up the call stack and only handled by interested methods

What does Throwing an Exception Mean ?

When an error occurs within a method, the method creates an object and hands it off to the runtime system. The object, called an exception object, contains information about the error, including its type and the state of the program when the error occurred. Creating an exception object and handing it to the runtime system is called throwing an exception.

What does Throwing an Exception Mean ?

When an error occurs within a method, the method creates an object and hands it off to the runtime system. The object, called an exception object, contains information about the error, including its type and the state of the program when the error occurred. Creating an exception object and handing it to the runtime system is called throwing an exception.

What are Checked Exceptions?

These are exceptional conditions that a well-written application should anticipate and recover from. Checked exceptions are subject to catch-or-specify requirement.

What are Unchecked Exceptions?

Exceptions not subject to the catch or specify requirement are collectively called unchecked exceptions.

- RuntimeException and all its subclasses
- Error and all its subclasses

What are Runtime Exceptions?

These are exceptional conditions that are internal to the application, and that the application usually cannot anticipate or recover from. These usually indicate programming bugs, such as logic errors or improper use of an API.

Handling Multiple Exceptions

Multiple types of exceptions thrown by a snippet of code can be handled by multiple catch clauses followed by the try block

What is an Exception in Java ?

An exception is an event, which occurs during the execution of a program, that disrupts the normal flow of the program's instructions. The term exception is shorthand for the phrase "exceptional event." Using exceptions helps us:

- Separate out error handling code from rest of the code
- Errors can be propagated up the call stack and only handled by interested methods

What does Throwing an Exception Mean ?

When an error occurs within a method, the method creates an object and hands it off to the runtime system. The object, called an exception object, contains information about the error, including its type and the state of the program when the error occurred. Creating an exception object and handing it to the runtime system is called throwing an exception.

What does Throwing an Exception Mean ?

When an error occurs within a method, the method creates an object and hands it off to the runtime system. The object, called an exception object, contains information about the error, including its type and the state of the program when the error occurred. Creating an exception object and handing it to the runtime system is called throwing an exception.

What are Checked Exceptions?

These are exceptional conditions that a well-written application should anticipate and recover from. Checked exceptions are subject to catch-or-specify requirement.

What are Unchecked Exceptions?

Exceptions not subject to the catch or specify requirement are collectively called unchecked exceptions.

- RuntimeException and all its subclasses
- Error and all its subclasses

What are Runtime Exceptions?

These are exceptional conditions that are internal to the application, and that the application usually cannot anticipate or recover from. These usually indicate programming bugs, such as logic errors or improper use of an API.

Handling Multiple Exceptions

Multiple types of exceptions thrown by a snippet of code can be handled by multiple catch clauses followed by the try block