Team id : PNT2022TMID46283
Date : 15 November 2022

What is exception handling?

Exception handling is the process of responding to unwanted or unexpected events when a computer program runs. Exception handling deals with these events to avoid the program or system crashing, and without this process, exceptions would disrupt the normal operation of a program.

Exceptions occur for numerous reasons, including invalid user input, code errors, device failure, the loss of a network connection, insufficient memory, to run an application, a memory conflict with another program, a program attempting to divide by zero or a user attempting to open files that are unavailable.

When an exception occurs, specialized programming language constructs, interrupt

hardware mechanisms or operating system interprocess communication facilities handle the exception.

How is exception handling used?

If a program has a lot of statements and an exception happens halfway through its execution, the statements after the exception do not execute, and the program crashes. Exception handling helps ensure this does not happen when an exception occurs.

How exception handling works

try block

Detects and thows exceptions

EXCEPTION OBJECT

catch block

Catches and handles
exceptions

The try block detects and throws any found exceptions to the catch blocks, which then handles them.

What are the types of exceptions?

Exceptions can come in the following two exception classes:

1. Checked exceptions. Also called *compile-time exception*s, the compiler checks these exceptions during the compilation process to confirm if the exception is being handled by the programmer. If not, then a compilation error displays on the system. Checked exceptions include SQLException and

ClassNotFoundException.

2. Unchecked exceptions. Also called *runtime exceptions*, these exceptions occur during program execution.

These exceptions are not checked at compile time, so the programmer is responsible for handling these exceptions. Unchecked exceptions do not give compilation errors. Examples of unchecked exceptions include NullPointerException and IllegalArgumentException.

Examples of exception handling

The following are examples of exceptions:

- SQLException is a checked exception that occurs while executing queries on a database for Structured Query Language syntax.

ClassNotFoundException is a checked exception that occurs when the required class is not found - either due to a

command-line error, a missing CLASS file or an issue with the classpath.

IllegalStateException is an unchecked exception that occurs when an environment's state does not match the operation being executed.

- IllegalArgumentException is an unchecked exception that occurs when an incorrect argument is passed to a method.

- NullPointerException is an unchecked exception that occurs when a user tries to access an object using a reference variable that is null or empty.