# Exception Handling:

# What is the difference between Error and Exception?

**Ans**:

Error is a problem in java applications and it will not allow application execution/it will not allow to execute the application.

There are two types of errors in java.

1. Compile time Errors

2 . Runtime Errors

**Compile time errors:**

These are problems identified by the compiler at compilation time.

In general, There are three types of compile time errors.

1. **Lexical Errors**: Mistakes in keywords.

**EX:** int i=10; ----> Valid

nit i=10; ----> Invalid

1. **Syntax Errors**: Gramatical Mistakes or syntactical mistakes.

**EX:** int i=10;---> Valid

i int = 10 ; --> Invalid

1. **Semantic Errors:** Providing operators inbetween incompatible types.

int i=10;

int j=20;

int k=i+j; ----> Valid

**EX:** int i=10;

boolean b=true;

char c= i+b;----> Invalid.

**NOTE:** Some other errors are generated by Compiler when we voilate JAVA rules and regulations injava applications.

**EX:** Unreachable Statements, variable might not have been initialized before its use,possible loss of precision etc…

**2.Runtime Errors:**

These are the problems for which we are unable to provide solutions programatically and these errors are not identified by the compilers and these errors are occurred at runtime.

**EX:**

* InSufficientMainMemory.
* Unavailability of IO Components.
* JVMInternalProblem
* StackOverflowError

# Exception:

Exception is a problem, for which we are able to provide solutions programatically.

**EX:**

* ArithmeticException
* NullPointerException
* ArrayIndexOutOfBoundsException

----

-----

**Exception:** Exception is an unexpected event occured in java applications at runtime , which may beprovided by users while entering dynamic input in java applications, provided by the Database Engines while executing sql queries in Jdbc applications, provided by Network while establishing connection between local machine and remote machine in distributed applications,[CLIENT-SERVER based applications are called distributed applications.].....causes abnormal termination to the java applications is called as EXCEPTION.

In JAVA there are two types of terminations possible:

1. Smooth Termination

Terminating program at end is called as Smooth termination.

1. Abnormal Termination

Terminating program in the middle is called as ABNORMAL TERMIANTION..

* In general, Exceptions are providing abnormal terminations, these abnormal terminations may crash local operating systems , these abnormal terminations may provide hangedout situations to the network.
* And Exception handling mechanisms will provide SMOOTH TERMINATIONS.

To overcome the above problems we have to handle exceptions properly, to handle exceptions properly we have to use "Exception Handling Mechanisms".

* Java is a Robust programming language, because,

Java is having very good memory management system in the form of Heap Memory Management system , it is a dynamic memory management system, it allocates and deallocates memory for the objects at runtime as per the application requirement.

**Java is having very good exception handling mechanisms, because JAVA has provided very good predefined library[Classes] to represent and handle almost all the EXCEPTIONS which are coming in JAVA Applications.**

**Note**: JAVA has provided a seperate predefined class for each and every exception.

There are two types of exceptions in Java:

1. Predefined Exceptions
2. User defined Exceptions

# 1.Predefined Exceptions

These Exceptions are defined by JAVA programming language and provided along with Java software.

There are two types of predefined Exceptions.

1. Checked Exceptions
2. Unchecked Exceptions

# What is the difference between checked exceptions and Unchecked Exceptions?

**Ans:**

Checked Exception is an exception recognized at compilation time by the compiler by monitoring our coding part, but, not occured at compilation time.(guessed by compiler/predicted by compiler).

Unchecked Exceptions are not recognized at compilation time by the compiler, these exceptions are recognized at runtime by the JVM.

**Note:** In Exceptions Arch., RuntimeException and its sub classes and Error and its sub classes are theexamples for Unchecked

Exceptions and all the remaining classes are the examples for Checked Exceptions.

There are two types of Checked Exceptions

1. Pure Checked Exceptions
2. Partially Checked Exceptions

# Q)What is the difference between Pure checked exceptions and Partially Checked Exceptions?

**Ans:**

If any checked exception is having only checked exceptions as sub classes then this exception is called as Pure Checked Exception.

**EX: IOException**

If any Checked Exception contains atleast one sub class as unchecked exception then that Checked Exception is called as Partially Checked Exception.

**EX: Exception ,Throwable**