**Advance JAVA**

**Throwable:**

* It is a class present in “java.lang” package.
* It is possible to throw the object of those classes which comes under the Throwable class.
* Throwable class has two main subclasses.

🡪ERROR

🡪Exception

**Error:**

* Error is a runtime error which cannot be handled.
* If an error occurs, program will get terminated abnormally.

EX:

**🡪Stack Overflow Error**

**🡪System crash**

**🡪JVM crash**

**Exception:**

* It is a runtime Error which can be handled.
* It is a class present in “java.lang”package.
* Whenever an unexpected (or) undefined event occurs then there is suitable object of Exception is not handled then program will get terminated abnormally.

🡪There are two types of Exception

🡪Checked Exceptions

🡪Unchecked Exceptions.

**How to Handle Exception:**

**Exception Handled:**

**Syntax:**

try {

//Exception Throwing Statement

}

Catch(Type of Exception variable)

{

//Solution for Exception

}

* If there is any exception then try block will terminate and catch block catch the Exception and code inside catch block is executed.
* If there is no Exception then catch block will be Skipped.

How to Know Type Of Exception:

🡪By Learning

🡪By Experience

🡪By reading Java Document file.

Default handler / Generalized Handler:

try{

//Exception Throwing Statements

}

Catch(Exception variable) //UPCASTING

{

//Sollution }

* Catch block with Exception type can catch any Exception.
* It is not preferred to use for every Exceptions.

**Multiple catch block:**

* A single try can have multiple catch block.

Syntax:

try {

//Multiple exception throwing statements

}

catch(TypeOfException variable)

{

}

Catch(TypeOfException variable)

{

}

………………

Catch(TypeOfException variable)

{

}

* Whenever there is a multiple exception throwing statements and they are related to each other, then we must use single try and multiple catch block.
* Outcome of single try block is always only one exception at a time.
* Out of multiple catch blocks only one catch block executed at a time.

**Rules to provide multiple catch block:**

* If there is a catch block with Exception type ,then it must be placed at the last position otherwise it leads to “Unreachable catch block error”.
* For Unreachable catch block compiler gives error.

**Unreachable catch block:**

* Those catch blocks which are placed after catch block with Exception type are known as “Unreachable catch block”.
* They are called as Unreachable because they never get executed as every Exception handled by catch block with Exception type.

**JAVA 8 FEATURES:**

try {

c=a/b;

arr[c]=48;

}

Catch(ArithmeticException | ArrayIndexOutOfBoundsException variable)

{

System.out.println(“your operation failed”);

}

**Exception Propagation:**

* Whenever there is an exception in a method, it searches for the handler, if there is no handler ,then the method execution terminated and object of exception travel to calling method.
* This passing of Exception from method to calling method is known as Exception Propagation.

**NOTE:**

* Every Exception is handled by the JVM at last if programmer didn’t handle the exception, but programmer program will get terminated.

**printStackTrace():**

* It is a method of Throwable class used to get information on the exception like Type of the Exception, cause message and location of Exception where it occurred.

**getMessage():**

* It is also a method of Throwable class.
* It is used to get only the cause message.

**How to Create an Exception:**

Class Mohan extends RuntimeException

{

Mohan()

{

}

Mohan(String msg) {

Super(msg);

}

}

|  |  |
| --- | --- |
| **Checked Exception** | **Unchecked exception** |
| * All checked Exceptions directly extends Exception class. * These Exceptions are frequently occurring Exceptions. * These Exceptions occur whenever java program deals with outside resource. * Compiler takes extra care for checked exceptions throwing statements, it forces the programmer to take “try catch block”, if programmer doesn’t provide “try catch block”, then it gives compile time error. * These are called checked Exception as compiler checks for the handler. | * These are the Exceptions which extends RuntimeException class. * These are rarely occurring Exceptions. * These Exceptions occurs when there is logical mistake in program. * Compiler will not force the programmer to provide handler. * These are called unchecked Exception as compiler will not check for handler. |