Exception

Exception is an error message. When user input is error, jvm won’t understand the language and give an exception as an output.

Exception Handling

Exception handling means handling the error using the Try and Catch block. When user does something wrong while writing the code, the whole program will shut down. So, to skip the inappropriate termination we use exception handling.

Try means here we try to use code and if we still have error, we try to catch them with exception, so that it executes the remaining part of the program.

There will be always catch after try block.

Two Types

1. Un-checked Exception: The exception which are unknown to the compiler.
2. Checked Exception: The exception which are known to the compiler and already checked exception.

Different Type of “try & catch” block

1. Single try & catch   
   -----------------------------  
   try{  
   riskycode;  
   }

catch(ExceptionClass reference){  
message  
}

1. Nested try catch  
   ---------------------------------  
   try{  
   riskycode;  
   }

try{  
riskycode;  
}

catch(ExceptionClass reference){  
message  
}

catch(ExceptionClass reference){  
message  
}

1. Try with multiple catch blocks  
   -------------------------------------------  
     
   try{  
   riskycode;  
   }

catch(ExceptionClass reference){  
message  
}

catch(ExceptionClass reference){  
message  
}

1. Frequent use of Try with catch  
   ----------------------------------------------  
   try{  
   riskycode;  
   }

catch(ExceptionClass reference){  
message  
}

Finally

Finally is the block which deallocate the memory or close the connection.

Finally can only be written with catch.

Finally is a block which will execute respective exception. If we have exception or we don’t finally will execute.

Finally is the last block of statement per class.

##Whenever we write the try catch, we cannot have instruction in between try and catch.

Throw

Throw is the keyword used to generate the user defined exception.

Programmer throw the exception object and jvm catch that exception object using the throw keyword.

Main feature:: To handover our created exception to the jvm manually.

Throws

throws exception is the keyword which can handle only checker exception.

To handle the checked exception, we can use throws and try catch.

To handle the un-checked exception, we can use try catch.

1) Checked Exception

The classes which directly inherit Throwable class except RuntimeException and Error are known as checked exceptions e.g. IOException, SQLException etc. Checked exceptions are checked at compile-time.

2) Unchecked Exception

The classes which inherit RuntimeException are known as unchecked exceptions e.g. ArithmeticException, NullPointerException, ArrayIndexOutOfBoundsException etc. Unchecked exceptions are not checked at compile-time, but they are checked at runtime.

3) Error

Error is irrecoverable e.g. OutOfMemoryError, VirtualMachineError, AssertionError etc.

|  |  |
| --- | --- |
|  | **Description** |
| try | The "try" keyword is used to specify a block where we should place exception code. The try block must be followed by either catch or finally. It means, we can't use try block alone. |
| catch | The "catch" block is used to handle the exception. It must be preceded by try block which means we can't use catch block alone. It can be followed by finally block later. |
| finally | The "finally" block is used to execute the important code of the program. It is executed whether an exception is handled or not. |
| throw | The "throw" keyword is used to throw an exception. |
| throws | The "throws" keyword is used to declare exceptions. It doesn't throw an exception. It specifies that there may occur an exception in the method. It is always used with method signature. |