**throw Keyword**

throw keyword is used to throw an exception explicitly. Only object of Throwable class or its sub classes can be thrown. Program execution stops on encountering **throw** statement, and the closest catch statement is checked for matching type of exception.

**Syntax :**

**throw** *ThrowableInstance*

**throws Keyword**

Any method capable of causing exceptions must list all the exceptions possible during its execution, so that anyone calling that method gets a prior knowledge about which exceptions to handle. A method can do so by using the **throws** keyword.

**Syntax :**

*type method\_name(parameter\_list)* **throws** *exception\_list*

{

//definition of method

}

**finally clause**

A finally keyword is used to create a block of code that follows a try block. A finally block of code always executes whether or not exception has occurred. Using a finally block, lets you run any cleanup type statements that you want to execute, no matter what happens in the protected code. A finally block appears at the end of catch block.

|  |
| --- |
| finally  {  //code  } |

Even if exception is thrown by the program, which is not handled by catch block, still finally block will get executed.