The Java *throws*keyword is used to declare an exception. It gives an information to the programmer that there may occur an exception so it is better for the programmer to provide the exception handling code so that normal flow can be maintained.

Exception Handling is mainly used to handle the checked exceptions. If there occurs any unchecked exception such as *NullPointerException*, it is programmers fault that he is not performing checkup before the code is used.

**The syntax of Java throws Keyword**

return\_type method\_name() throws exception\_class\_name{

//method code

}

We declare only checked exception using a *throws* keyword. Let's see an example to demonstrate the usage of a *throws* keyword.

Basically, whenever exception arises there two cases, either you should handle the exception using [**try/catch**](http://www.javaguides.net/2018/08/java-trycatch-block.html) or you declare the exception i.e. specifying *throws*with the method.

**throws Keyword Example**

In this example, the *exceptionWithoutHandler(), exceptionWithoutHandler1() and exceptionWithoutHandler2()*methods uses *throws*keyword to declare exception.

public class ExceptionHandlingWorks {

public static void main(String[] args) {

exceptionHandler();

}

private static void exceptionWithoutHandler() throws IOException {

try (BufferedReader reader = new BufferedReader(new FileReader(new File("/invalid/file/location")))) {

int c;

// Read and display the file.

while ((c = reader.read()) != -1) {

System.out.println((char) c);

}

}

}

private static void exceptionWithoutHandler1() throws IOException {

exceptionWithoutHandler();

}

private static void exceptionWithoutHandler2() throws IOException {

exceptionWithoutHandler1();

}

private static void exceptionHandler() {

try {

exceptionWithoutHandler2();

} catch (IOException e) {

System.out.println("IOException caught!");

}

}

}