|  |  |
| --- | --- |
| **Throw**  **throw** keyword is used to throw an exception explicitly/forcibly. It can throw only one exception at a time. | **Throws**  **throws** keyword is in the function signature. It is used when the function has some statements that can lead to some exceptions. |
| throw keyword is used to throw an exception explicitly. | throws keyword is used to declare one or more exceptions, separated by commas. |
| Only single exception is thrown by using throw. | Multiple exceptions can be thrown by using throws. |
| throw keyword is used within the method. | throws keyword is used with the method signature. |
| Syntax wise throw keyword is followed by the instance variable. | Syntax wise throws keyword is followed by exception class names. |
| Checked exception cannot be propagated using throw only.Unchecked exception can be propagated using throw. | For the propagation checked exception must use throws keyword followed by specific exception class name. |

**Example:**

* + **throw**:

*filter\_none*

*brightness\_4*

|  |
| --- |
| public class GFG {    public static void main(String[] args)    {     // Use of unchecked Exception     try {      // double x=3/0;      throw new ArithmeticException();     }     catch (ArithmeticException e)     {       e.printStackTrace();     }    }  } |

**Output:**

java.lang.ArithmeticException: / by zero

at UseOfThrow.main(UseOfThrow.java:8)

* + **throws**: =================================================

*filter\_none*

*brightness\_4*

|  |
| --- |
| import java.io.IOException;    public class UseOfThrowAndThrows {        public static void main(String[] args)          throws IOException      {      }  } |

**Output:**

Exception in thread "main" java.io.IOException

at UseOfThrowAndThrows.main(UseOfThrow.java:7)

=============================================================

// throwing multiple exceptions

void Demo() throws ArithmeticException, NullPointerException

{

// Statements where exceptions might occur.

}

=================================================================

The throw and throws are the two keywords used in exception handling. In this article, you will learn about how both of them are different from one another.

**Difference Between throw and throws in Java**

|  |  |
| --- | --- |
| **Throw** | **Throws** |
| This keyword is used for explicitly throwing an exception. | This keyword is used for declaring any exception. |
| Programmers cannot disseminate checked exceptions using the throw keyword. | Programmers can disseminate checked exceptions using throws keyword. |
| An instance trails the throw keyword. | A class trails the throws keyword. |
| You have to use the throw keyword inside any method. | You have to use the throws keyword with any sign of the method. |
| Many exceptions cannot be thrown. | Many exceptions can be declared using the throws. |
| Code Snippet:  Class Example { public static void main(String argu[]) { throw new ArithmeticException("Divided by zero"); } } | Code Snippet:  public void testExc() throws SQLException , IOException |

GEEKS FOR GEEKS

# <https://www.geeksforgeeks.org/difference-between-throw-and-throws-in-java/>

# Difference between throw and throws in Java

**Prerequisite:** [Throw and Throws in Java](https://www.geeksforgeeks.org/throw-throws-java/)

The differences between **throw** and **throws** are:

1. **Point of usage**
   * **throw** keyword is used inside a function. It is used when it is required to throw an Exception logically.

*filter\_none*

*brightness\_4*

|  |
| --- |
| void Demo() throws ArithmeticException, NullPointerException  {  // Statements where exceptions might occur.  throw new ArithmeticException();  } |

* + **throws** keyword is in the function signature. It is used when the function has some statements that can lead to some exceptions.

*filter\_none*

*brightness\_4*

|  |
| --- |
| void Demo()  {  // Statements where exceptions might occur.  } |

1. **Number of exceptions thrown**
   * **throw** keyword is used to throw an exception explicitly. It can throw only one exception at a time.  
     For example:

*filter\_none*

*brightness\_4*

|  |
| --- |
| // throwing only an IOException  throw new IOException(); |

* + **throws** keyword can be used to declare multiple exceptions, separated by comma. Whichever exception occurs, if matched with the declared ones, is thrown automatically then.  
    For example:

*filter\_none*

*brightness\_4*

|  |
| --- |
| // throwing multiple exceptions  void Demo() throws ArithmeticException, NullPointerException  {  // Statements where exceptions might occur.  } |

1. **Syntax**
   * Syntax of **throw** keyword includes the instance of the Exception to be thrown.  
     For example:

*filter\_none*

*brightness\_4*

|  |
| --- |
| // throwing instance of IOException  throw new IOException(); |

* + Syntax of **throws** keyword includes the class names of the Exceptions to be thrown.  
    For example:

*filter\_none*

*brightness\_4*

|  |
| --- |
| // throwing multiple exceptions by class names  void Demo() throws ArithmeticException, NullPointerException  {  // Statements where exceptions might occur.  } |

1. **Propagation of Exceptions**
   * **throw** keyword cannot propagate checked exceptions. It is only used to propagate the unchecked Exceptions that are not checked using throws keyword.  
     For example:

*filter\_none*

*brightness\_4*

|  |
| --- |
| void main() throws IOException  {  // Using throw for ArithmeticException  // as it is unchecked in throws  throw new ArithmeticException();  } |

* + **throws** keyword is used to propagate the checked Exceptions only.  
    For example:

*filter\_none*

*brightness\_4*

|  |
| --- |
| void main() throws IOException  {  } |

1. **Example:**
   * **throw**:

*filter\_none*

*brightness\_4*

|  |
| --- |
| public class GFG {    public static void main(String[] args)    {     // Use of unchecked Exception     try {      // double x=3/0;      throw new ArithmeticException();     }     catch (ArithmeticException e)     {       e.printStackTrace();     }    }  } |

**Output:**

java.lang.ArithmeticException: / by zero

at UseOfThrow.main(UseOfThrow.java:8)

* + **throws**:

*filter\_none*

*brightness\_4*

|  |
| --- |
| import java.io.IOException;    public class UseOfThrowAndThrows {        public static void main(String[] args)          throws IOException      {      }  } |

**Output:**

Exception in thread "main" java.io.IOException

at UseOfThrowAndThrows.main(UseOfThrow.java:7)