

Department: BCA  
Faculty Name: Ashish Kumar  
Subject: JAVA (5<sup>th</sup> sem.)  
Topic: **Java throw keyword**

### Java throw keyword:

- The Java throw keyword is used to explicitly throw an exception.
- We can throw either checked or unchecked exception in java by throw keyword.
- The throw keyword is mainly used to throw custom exception.

### Syntax of java throw keyword:

**throw** exception;

or

**throw** new ExceptionClassName(msg);

### Example:

```
1 public class TestThrow
2 {
3     void validate(int age)
4     {
5         try
6         {
7             if(age<18)
8                 throw new ArithmeticException("not valid");
9             else
10                System.out.println("welcome to vote");
11            }
12            catch(ArithmeticException e)
13            {
14                System.out.println(e);
15            }
16        }
17    }
18    public static void main(String args[])
19    {
20        TestThrow obj=new TestThrow();
21        obj.validate(13);
22        System.out.println("rest of the code...");
23    }
24 }
```

```
C:\Users\ashish\Desktop\java>java TestThrow
Exception in thread "main" java.lang.ArithmeticException: not valid
    at TestThrow.validate(TestThrow.java:6)
    at TestThrow.main(TestThrow.java:13)

C:\Users\ashish\Desktop\java>javac TestThrow.java

C:\Users\ashish\Desktop\java>java TestThrow
java.lang.ArithmeticException: not valid
rest of the code...

C:\Users\ashish\Desktop\java>
```

## Example:

The screenshot shows a Notepad++ editor with a Java file named `TestAge.java`. The code defines a custom exception `AgeException` that extends `RuntimeException` and a `TestAge` class with a `main` method. The `main` method uses a `Scanner` to read user input, checks if the age is less than 18, and throws an `AgeException` if so. The exception is caught and its stack trace is printed. The Command Prompt window shows the execution of the program, demonstrating the exception handling for both eligible and ineligible ages.

```

1  import java.util.Scanner;
2  class AgeException extends RuntimeException
3  {
4      /* AgeException()
5      {
6          super();
7      }*/
8      AgeException(String msg)
9      {
10         super(msg); //DefaultExceptionHandler
11     }
12 }
13
14 class TestAge
15 {
16     public static void main(String args[])
17     {
18         Scanner sc=new Scanner(System.in);
19         System.out.println("Enter Your age:");
20         int age=sc.nextInt();
21         try
22         {
23             if(age<18)
24             {
25                 throw new AgeException("you are not eligible for voting...!!");
26             }
27             else
28             {
29                 System.out.println("You Are eligibel for voting...!!");
30             }
31         }
32         catch(AgeException e)
33         {
34             e.printStackTrace();
35         }
36
37         System.out.println("rest of Code....");
38     }
39 }
40
41

```

Command Prompt Output:

```

C:\Users\ashish\Desktop\java>javac TestAge.java
C:\Users\ashish\Desktop\java>java TestAge
Enter Your age:
11
AgeException: you are not eligible for voting...!!
    at TestAge.main(TestAge.java:25)
rest of Code....
C:\Users\ashish\Desktop\java>java TestAge
Enter Your age:
22
You Are eligibel for voting...!!
rest of Code....
C:\Users\ashish\Desktop\java>

```