CHANDIGARH UNIVERSITY

Discover. Learn. Empower.

# INSTITUTE : UIE

# DEPARTMENT : CSE

Bachelor of Engineering (Computer Science & Engineering)

**PROJECT BASED LEARNING IN JAVA**

**(20CST-319/20ITT-319)**

TOPIC OF PRESENTATION:

Use of try, catch and throw. Difference between throw and throws.

# Lecture Objectives

In this lecture, we will discuss:

Use of try, catch and throw.
Difference between throw and throws.

# How Programmer handles an exception?

**Customized Exception Handling :**

- Java exception handling is managed via five keywords: **try**, **catch**, **throw**, **throws**, and **finally**.
- Program statements that can raise exceptions are contained within a try block.
- If an exception occurs within the try block, it is thrown.
- System-generated exceptions are automatically thrown by the Java run-time system. To manually throw an exception, use the keyword throw.
- Any exception that is thrown out of a method must be specified as such by a throws clause.
- Any code that absolutely must be executed after a try block completes is put in a finally block.

# Advantage of Exception Handling

- **Maintain the normal flow of the application**.: Exception normally disrupts the normal flow of the application that is why we use exception handling.
- **Separating Error-Handling Code from "Regular" Code :** Exceptions provide the means to separate the details of what to do when something out of the ordinary happens from the main logic of a program.
- **Propagating Errors Up the Call Stack :** Another advantage of exceptions is the ability to propagate error reporting up the call stack of methods.
    - Suppose that the readFile method is the fourth method in a series of nested method calls made by the main program: method1 calls method2, which calls method3, which finally calls readFile.
    - Suppose also that method1 is the only method interested in the errors that might occur within readFile.

# Exception handling- Try Catch in Java

**Try block**

- The try block contains set of statements where an exception can occur.
- A try block is always followed by a catch block, which handles the exception that occurs in associated try block.
- A try block must be followed by catch blocks or finally block or both.

**Syntax of try block try{**

```
    //statements that may cause an exception
}
```

- While writing a program, if certain statements in a program can throw a exception, enclosed them in try block and handle that exception

# Exception handling- Try Catch in Java

## Catch block

- A catch block is where handle the exceptions are handled
- This block must follow the try block.
- A single try block can have several catch blocks associated with it.
- We can catch different exceptions in different catch blocks.
- When an exception occurs in try block, the corresponding catch block that handles that particular exception executes.
- For example if an arithmetic exception occurs in try block then the statements enclosed in catch block for arithmetic exception executes.

# Exception handling- Try Catch in Java

**Rules about multiple catch blocks:**

- a single try block can have any number of catch blocks.
- A generic catch block can handle all the exceptions

  catch(Exception e){
  //This catch block catches all the exceptions}

- If no exception occurs in try block then the catch blocks are completely ignored.
- Corresponding catch blocks execute for that specific type of exception:
  - catch(ArithmeticException e) is a catch block that can hanlde ArithmeticException
  - catch(NullPointerException e) is a catch block that can handle NullPointerException

# Exception handling- Try Catch in Java

- If an exception occurs in try block then the control of execution is passed to the
- corresponding catch block.
- A single try block can have multiple catch blocks associated with it, you should place the catch blocks in such a way that the generic exception handler catch block is at the last
- The generic exception handler can handle all the exceptions but you should place is at the end
- if you place it at the before all the catch blocks then it will display the generic message. The user should get a meaningful message for each type of exception rather than a generic message.

# throws Keyword

- Throws keyword is used for handling checked exceptions
- 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.

```
returntype methodname() throws exception_list
{
//method code
}
```

- Any method that is 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 are to be handled.
- A method can do so by using the **throws** keyword.

# throw exception in java

- Throw keyword can also be used for throwing custom/user defined exceptions
- 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*

# Difference : throw and throws

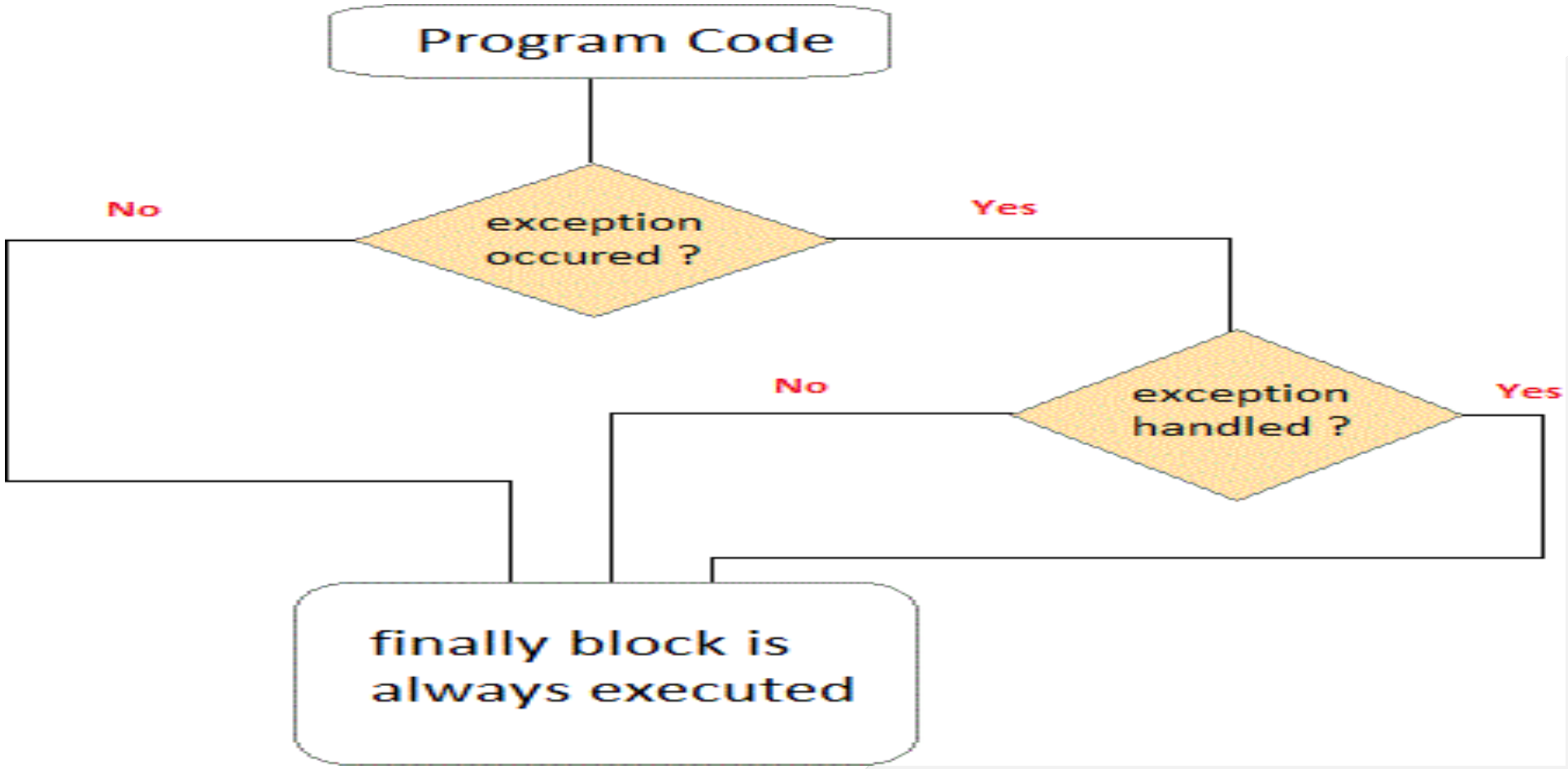
## Difference between throw and throws

| throw | throws |
|---|---|
| throw keyword is used to throw an exception explicitly. | throws keyword is used to declare an exception possible during its execution. |
| throw keyword is followed by an instance of Throwable class or one of its sub-classes. | throws keyword is followed by one or more Exception class names separated by commas. |
| throw keyword is declared inside a method body. | throws keyword is used with method signature (method declaration). |
| We cannot throw multiple exceptions using throw keyword. | We can declare multiple exceptions (separated by commas) using throws keyword. |

# finally keyword

- A finally keyword is used to create a block of code that follows a try block.
- A finally block of code is always executed whether an exception has occurred or not. Using a finally block, it lets you run any clean-up 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 keyword



# Difference : final,finally,finalize

| Sr. No. | final | finally | finalize |
|---|---|---|---|
| 1) | Final is used to apply restrictions on class, method and variable. Final class can't be inherited, final method can't be overridden and final variable value can't be changed. | Finally is used to place important code, it will be executed whether exception is handled or not. | Finalize is used to perform clean up processing just before object is garbage collected. |
| 2) | Final is a keyword. | Finally is a block. | Finalize is a method. |

# User defined exception in java

- In java we can create our own exception class and throw that exception using throw keyword. These exceptions are known as user-defined or custom exceptions.

**Syntax**

- class classname extends Exception

- . User-defined exception must extend Exception class.
- The exception is thrown using throw keyword.

***While creating user defined exceptions, the following aspects have to be taken care :***

***a. The user defined exception class should extend from the Exception class and its subclass***

***b. If we want to display meaningful information about the exception, we should override the toString() method***.

# User defined exception in java

```
class MyExceptionDemo{
static int flag=0;
public static void main(String args[]) {
try {
int age=Integer.parseInt(args[0]);
if(age < 18)
throw new MyException(); }
catch(ArrayIndexOutOfBoundsException e) {
flag=1;
System.out.println("Exception : "+ e); }
catch(NumberFormatException e) {
flag=1;
System.out.println("Exception : "+ e); }
catch (MyExceptionClass e) {
flag=1;
System.out.println("Exception : "+ e); }
if(flag==0)
System.out.println("Everything is fine");  } }
```

# QUIZ:

1. Which of these keywords is not a part of exception handling?
a) try
b) finally
c) thrown
d) catch

2. Which of these keywords must be used to monitor for exceptions?
a) try
b) finally
c) throw
d) catch

# Summary:

In this session, you were able to :

- Learn about Use of try, catch and throw. Difference between throw and throws.

# References:


**Books:**

1. Balaguruswamy, *Java.*
2. A Primer, E.Balaguruswamy, *Programming with Java,* Tata McGraw Hill Companies
3. John P. Flynt Thomson, *Java Programming.*


**ONLINE NOTES LINKS:**

https://www.w3schools.com/java/java_try_catch.asp
https://www.javatpoint.com/try-catch-block
https://docs.oracle.com/javase/tutorial/essential/exceptions/try.html
https://www.guru99.com/java-exception-handling.html

VIDEO LINKS:

https://www.youtube.com/watch?v=4my7mKFaNQs
https://www.youtube.com/watch?v=W-N2ltgU-X4

# THANK YOU