**Exception Handling**

**Q what is Exception in java ?**

**Exception:** An Exception is a runtime error or it is an abnormal condition that arises in a code sequence at run time.

**Example-** Arithmetic Exception, NullPointException, StringIndexOutofBoundsException, NumberFormatException, FileNotFoundException, ArrayIndexOutofBoundException

ClassNotFoundException, IOException, NoSuchElementException

**Object[Super most class]**

Throwable [class]

Error [Uncontrollable ] Exception [Controllable]

StackOverFlow Error

Unchecked Exception Checked Exception

[Run time Error] [Compile time Error]

Arithmetic Exception, IOException

NumberNotFound FileNotFoundException

NullPointer ClassNotFoundExcept

StringIndexOutofBoundsException

ArrayIndexOutofBoundException

**Q What is Exception handling in java?**

The Exception handing is one of the most powerful mechanism to handle the runtime error. It is managed by 5 keywords.

1. **Try**
2. **Catch**
3. **Finally**
4. **Throw**
5. **Throws**

**Q Try -catch and Finally Block**

**Try {**

Use code here which may be happened Exception. Code you want to monitor and that code in which exception may be happened

}

Throw --🡪 If any Exception, catch it then that Exception will throw and catch block catch that exception.

It will receive if it is matched with Exception type. Catch block can handle the error if it match with the Catch block’s type.

If catch block can’t handle the exception then it will be into finally block. Finally block must be executed any exception which you catch or which you written.

Where Exception is catch then next line of the code will not work .So It would be handled.

Since /when Exception type is not clear or we don’t know type of exception or it is no possible to write exception type, In that case we will write “Exception e” instead of type of Exception.

Like catch (Exception e){

}

// Multiple catch block may be written.

**Finally Block- finally { // Block of code to be Executed after try block, at least one time**

**}**

public class MyClass {

public static void main(String[ ] args) {

int[] myNumbers = {1, 2, 3};

System.out.println(myNumbers[10]); // error!

}

}

public class MyClass {

public static void main(String[ ] args) {

int[] myNumbers = {1, 2, 3};

System.out.println(myNumbers[10]); // error!

}

}

public class MyClass {

public static void main(String[] args) {

try {

int[] myNumbers = {1, 2, 3};

System.out.println(myNumbers[10]);

} catch (Exception e) {

System.out.println("Something went wrong.");

} finally {

System.out.println("The 'try catch' is finished.");

}

}

}