**Day 5: 22-10-2025: OOPs concept and exception handling**

**Access specifiers and packages**

Java provided totally 4 types of access specifiers which expose the visibility or accessibility of class, constructor, variable and method within a same class, outside class, same package as well as other package.

1. private

visibility : within a same class

private we can use with instance variable, static variable, constructor, non static method static method but we can’t use with class as well as local variable.

1. default (nothing)

visibility : within a same package

default we can use with all.

1. protected

visibility : within a same package as well as other package if it is sub class.

protected we can use with instance variable, static variable, constructor, non static method static method but we can’t use with class as well as local variable.

1. public

visibility : within a same package as well as other package.

public we can use with all but can’t use with local variable.

package : package is a collection of classes and interfaces. Package help us to organized the classes and interfaces in proper manner. Using package we can create more than one classes as well as interface which have same name but different purpose. Package is like a directory or folder.

Package mainly divided into 2 types.

1. User defined package
2. Pre defined or built in package.

To create the package we need to use package keyword with name of the package.

package com;

**education**

**school college**

Attendance.java Attendance.java

**pre defined or built in packages.**

Java provided lot of pre defined packages.

Java provided mainly 2 root packages

java javax

root packages

lang sql

io net

util swing

sql servlet

net ejb

awt jms

etc etc

by default every java program imported lang package. without importing lang package we can use all classes and interfaces part of lang package.

by default every java program it may be pre defined or user defined extends Object class. Object is super class for all java classes.

class A {

}

class B extends A{

}

**Exception Handling:**

Exception is a pre defined class part of lang package. Exception is an object which generate occurs when unexcepted things or abnormal condition occurs during the execution of a program.

Using some technique you need to handle generated exception that is known as exception handling.

Java Program

Compile the program Run the program

javac java

compiler interpreter

compile time error

syntax error or typo

error

Run time error

Error Exception

Error : The error which generate at run time which we can’t handle it ie Error. JVM crash, software or hardware issue, out of memory etc.

Exception: it is a type of run time error which we can handle it example divided by zero.

Object

Throwable

Error Exception

Checked exception un checked exception

RuntimeException

IOException ArithmeticExeption

SQLException NumberFormatException

To handle both checked as well as unchecked exception java provided totally 5 keywords.

1. try
2. catch
3. finally
4. throw
5. throws

**unchecked exception**

**try catch block**

syntax

try {

try block

}catch(Exception e) {

catch block

}

In java we can write try with multiple catch block

Syntax

This code handle 3 types of exception and base upon exception it generate it execute that particular try block.

try{

}catch(ArithmeticException e) {

}catch(NumberFormatException e) {

}catch(ArrayIndexOutOfBounds e) {

}

Or

try{

}catch(ArithmeticException e) {

Specific

}catch(NumberFormatException e) {

Specific

}catch(ArrayIndexOutOfBounds e) {

Specific

}catch(Exception e) {

Generic exception

}

try{

} catch(Exception e) {

Generic exception

}catch(ArithmeticException e) {

Specific

}catch(NumberFormatException e) {

Specific

}catch(ArrayIndexOutOfBounds e) {

Specific

}

Always sub class must be on top and super class must be in bottom

try{

}

catch(SQLException e) {

} catch(RuntimeException e) {

}catch(Exception e) {

}

**Finally block:**

**try block**: the code which generate the exception. It may be one line code or multi line code you need to keep in try block.

**catch block:** this block execute only if any exception generate. No exception no catch block.

Finally is a block which execute 100% sure doesn’t matter exception generate or not.

try

catch catch catch catch finally

catch finally catch

finally

**throw** : throw keyword is use to throw or raise or generate pre defined or user defined exception with custom logic.

Syntax

throw new Exception()

or

throw new ExceptionSubClass()