

Object Oriented Programming (Java)



Mohadeb Kumar

-Lecturer-

Dept. of Computer Science & Engineering (CSE)

TMSS Engineering College (TEC), Bogura

Part-01: Introduction to Java as OOP

- ❖ What is Program?
- ❖ Type of Programming Languages.
- ❖ What does mean by Object Oriented Programming (OOP)?
- ❖ History of Java.
- ❖ Important Features of Java.
Link: <https://www.javatpoint.com/features-of-java>
- ❖ Advantages and applications of Java.
Link: https://www.w3schools.com/java/java_intro.asp
- ❖ Related Tools and Environment.
 - Java Development Kit (JDK)
 - Java Runtime Environment (JRE)
 - Java Virtual Machine (JVM)
 - Integrated Development Environment (IDE): NetBeans/Eclipse/JDeveloperLink: <https://www.geeksforgeeks.org/setting-environment-java/>
- ❖ How Java Virtual Machine Work?
Link: <https://techvidvan.com/tutorials/java-virtual-machine/>
- ❖ What is Byte code? Write difference between Byte Code and Machine Code.
Link: <https://www.geeksforgeeks.org/difference-between-byte-code-and-machine-code/>
- ❖ First Java Program.
Link: <https://www.javatpoint.com/simple-program-of-java>
- ❖ Comment in Java.
Link: https://www.w3schools.com/java/java_comments.asp

Home Work:

- What

Part-02: Variable and Data types

- ❖ Escape Sequence in Java.
Link: <https://www.javatpoint.com/java-escape-characters>
- ❖ Data types.
Link: <https://www.geeksforgeeks.org/data-types-in-java/>
- ❖ Keyword in Java.

Link: <https://www.javatpoint.com/java-keywords>

❖ Variables

Link: <https://www.guru99.com/java-variables.html>

❖ Variable scope and Access modifiers.

Link: <https://www.javatpoint.com/scope-of-variables-in-java>

❖ Naming of Variable or Identifier naming rule.

Link: <https://www.geeksforgeeks.org/java-identifiers/>

❖ Java Type Casting.

Link: https://www.w3schools.com/java/java_type_casting.asp

❖ Operators.

Link: <https://www.javatpoint.com/operators-in-java>

Home Work:

- Write

Part-03: Control Flow and Statement

❖ Control Flow and statements in Java

a. Decision Making statements

- if statements
- switch statement

Link: <https://www.geeksforgeeks.org/decision-making-javaif-else-switch-break-continue-jump/>

b. Loop statements

- do while loop
- while loop
- for loop
- for-each loop

Link: <https://www.geeksforgeeks.org/loops-in-java/>

c. Jump statements

- break statement
- continue statement

Link: https://www.w3schools.com/java/java_break.asp

Home Work:

- Write

Part-04: String

❖ String and string operation

- length ()
- concat ()
- equals ()

Link: <https://www.programiz.com/java-programming/string>

Home Work:

- Write

Part-05: Method/Function in Java

❖ Creating & calling a method

Link: https://www.w3schools.com/java/java_methods.asp

❖ Passing parameters or Arguments.

Link: https://www.w3schools.com/java/java_methods_param.asp

❖ Method overloading

Link: https://www.w3schools.com/java/java_methods_overloading.asp

❖ Recursion in Java

Link: <https://www.javatpoint.com/recursion-in-java>

Home Work:

- Write

Part-06: Array

❖ Array in Java

- One/single dimensional Array
- Multidimensional Array

Link: <https://www.javatpoint.com/array-in-java>

❖ Enhanced for Statement (for-each loop)

Link: <https://www.geeksforgeeks.org/for-each-loop-in-java/>

❖ Passing Arrays to Methods.

Link: <https://www.javatpoint.com/array-in-java>

❖ Variable-Length Argument Lists & Command-Line Arguments.

Link: <http://www.beginwithjava.com/java/arrays-arraylist/variable-length-argument-lists.html>

Home Work:

- Write

Part-07: Class and Object

❖ Java OOPs Concepts

Link: <https://www.javatpoint.com/java-oops-concepts>

❖ Java Naming Convention

Link: <https://www.javatpoint.com/java-naming-conventions>

❖ Objects and Classes in Java

Link: <https://www.javatpoint.com/object-and-class-in-java>

❖ Methods and instance variables

Link: <https://www.javatpoint.com/method-in-java>

❖ Program Modules in Java

Link: <https://www.geeksforgeeks.org/java-modules/>

❖ static Methods, static Fields

Link: <https://www.geeksforgeeks.org/static-method-in-java-with-examples/>

❖ Constructor and Distractor

Link: <https://www.javatpoint.com/java-constructor>

❖ Methods with Multiple Parameters

Link: <https://www.programiz.com/java-programming/methods>

❖ Java API Packages.

Link: <https://www.geeksforgeeks.org/packages-in-java/>

❖ Java static keyword

Link: <https://www.javatpoint.com/static-keyword-in-java>

❖ use of static variables and methods

Link: <https://www.geeksforgeeks.org/static-method-in-java-with-examples/>

❖ to import static members of a class,

Link: <https://www.geeksforgeeks.org/static-import-java/>

❖ this keyword in Java

Link: <https://www.javatpoint.com/this-keyword>

❖ Controlling Access to Members,

Link: <https://www.geeksforgeeks.org/access-modifiers-java/>

❖ Inheritance

- Inheritance (IS-A)
Link: <https://www.javatpoint.com/inheritance-in-java>
- Aggregation (HAS-A)
Link: <https://www.javatpoint.com/aggregation-in-java>

❖ Polymorphism

- Method Overloading
Link: <https://www.javatpoint.com/method-overloading-in-java>
- Method Overriding
Link: <https://www.javatpoint.com/method-overriding-in-java>
- Covariant Return Type
Link: <https://www.javatpoint.com/covariant-return-type>
- super keyword
Link: <https://www.javatpoint.com/super-keyword>
- Instance Initializer block
Link: <https://www.javatpoint.com/instance-initializer-block>
- final keyword
Link: <https://www.javatpoint.com/final-keyword>
- Runtime Polymorphism
Link: <https://www.javatpoint.com/runtime-polymorphism-in-java>
- Dynamic Binding
Link: <https://www.javatpoint.com/static-binding-and-dynamic-binding>
- instanceof operator
Link: <https://www.javatpoint.com/downcasting-with-instanceof-operator>

❖ Abstraction

- Abstract class
Link: <https://www.javatpoint.com/abstract-class-in-java>
- Interface
Link: <https://www.javatpoint.com/interface-in-java>
- Abstract vs Interface
Link: <https://www.javatpoint.com/difference-between-abstract-class-and-interface>

❖ Encapsulation

- Package
Link: <https://www.javatpoint.com/package>
- Access Modifiers
Link: <https://www.javatpoint.com/access-modifiers>
- Encapsulation
Link: <https://www.javatpoint.com/encapsulation>

Home Work:

- Write

Part-08: Input/Output Files in Java

- ❖ To create, read, write and update files
- ❖ To retrieve information about files and directories
- ❖ Java input/output stream class hierarchy
- ❖ Differences between text files and binary files
- ❖ Sequential-access and random-access file processing.

Home Work:

- Write

Part-09: Exception Handling

- ❖ Overview of exception handling,
- ❖ How exception and error handling works
- ❖ To use try, throw and catch to detect, indicate and handle exceptions respectively
- ❖ To use the finally block to release resources
- ❖ To declare new exception classes

Home Work:

- Write

Part-10: Thread

- ❖ Introduction,
- ❖ Multithread,
- ❖ Synchronization,
- ❖ Deadlock.

Home Work:

- Write

Part-11: GUI Programming

- ❖ The design principles of graphical user interfaces (GUIs):
 - To build GUIs and handle events generated by user interactions with GUIs
 - To handle mouse events and keyboard events
 - To use layout managers to arrange GUI components

Home Work:

- Write

Part-12: Design Patterns

- ❖ Overview
- ❖ Common Design patterns:
 - Singleton
 - Factory and Abstract Factory
 - Builder
 - Adapter
 - Iterator
 - Observer
 - Strategy patterns

Home Work:

- Write