

Java Programming Notes (TYBSc - Semester 5)

Core Java Concepts

- Role of JVM, JRE, and JDK in Java.
- Differences between primitive and reference data types.
- Explanation of various access modifiers: public, private, protected, and default.
- Method overloading vs method overriding.

Object-Oriented Programming (OOP)

- Definition and examples of classes and objects.
- How inheritance works in Java with an example.
- Concept of polymorphism and its types (runtime and compile-time).
- Abstract classes vs interfaces.

Exception Handling

- Purpose and advantages of exception handling.
- Using try-catch-finally blocks effectively.
- Custom exception creation and usage with an example.
- Throw vs throws in exception handling.

Multithreading

- Lifecycle of a thread in Java.
- Creating threads using Thread class and Runnable interface.
- Thread synchronization and inter-thread communication.
- Practical examples for multithreading.

Collections Framework

- Overview of collections: List, Set, and Map.
- Differences between ArrayList and LinkedList.

OOP's ch 4 :User Interface with AWT and Swing

- Every user interface considers the following three main aspects: **UI elements, Layouts, Behavior.**
- The **Abstract Window Toolkit (AWT)** is Java's original platform-dependent windowing, graphics, and user-interface widget toolkit.
- The AWT is part of the **Java Foundation Classes (JFC)**.

What is AWT? :

- The Java programming language class library provides a user interface toolkit called the **Abstract Window Toolkit**, or the **AWT**.
- Java AWT is an API to develop GUI or window-based application in Java.
- Java AWT components are platform-dependent.
- The AWT is now part of the Java Foundation Classes (JFC) is the standard API for providing a **Graphical User Interface (GUI)** for a Java program.
- The AWT classes are contained in **the java.awt package**. It is one of Java's largest packages.

What is Swing? :

- Swing is a GUI widget toolkit for Java.
- It is part of Oracle's Java Foundation Classes (JFC)- is the API for providing a Graphical User Interface (GUI) for Java programs.
- The swing components are defined in the package javax.swing. This package provides more powerful and flexible components
- Swing is a Java foundation classes, library and it is has extension to do Abstract Window Toolkit (AWT).
- Various features of swing are:

Light Weight, Rich Controls, Borders, Easy mouseless Operation, Tooltips, Easy Scrolling, Highly Customizable,

Sr. No.	AWT	Swing
1.	AWT components are platform dependent.	Java swing components are platformindependent.
2.	AWT components are heavy weight.	Swing components are light weight.
3.	AWT doesn't support pluggable look and feel.	Swing supports pluggable look and feel.
4.	AWT provides less components than Swing.	Swing provides more powerful components such as tables, lists, scrollpanes, colorchooser, tabbedpane etc.
5.	AWT components require java.awt package.	Swing components require javax.swing package.
6.	Advanced features is not available in AWT.	Swing has many advanced features like JTable, Jtabbed pane which is not available in AWT.
7.	Different looked feel feature is not supported in AWT.	We can have different look and feel in Swing.
8.	Using AWT, we have to implement a lot of things ourself.	Swing has them built in.

LAYOUTS AND LAYOUT MANAGERS :

- Layout means the arrangement of components within the container.
- In other way we can say that placing the components at a particular position within the container.
- Layout manager is an object which determines the way that components are arranged in a frame window.
- A layout manager automatically arranges our controls within a window by using some type of algorithm.
- A layout manager is an instance of any class that implements the **LayoutManager** interface.
- The layout manager is set by the **setLayout()** method.
- Each layout manager keeps track of a list of components that are stored by their names
- The layout manager automatically positions all the components within the container.

- Applications of HashMap and TreeSet.
- Using iterators for collection traversal.

Java I/O

- Stream hierarchy: InputStream, OutputStream, Reader, Writer.
- File operations: reading, writing, renaming, and deleting files.
- Serialization and deserialization in Java.

Swing and GUI Development

- AWT vs Swing: Key differences.
- Creating simple GUIs using Swing components like JButton, JLabel, and JTextField.
- Event handling mechanisms in Swing.

JDBC (Java Database Connectivity)

- Steps to connect Java applications to a database.
- Executing queries using Statement and PreparedStatement.
- Fetching results with ResultSet.
- Error handling in database operations.

- A **check box** is a control that is used to turn an option on (true) or off (false). It consists of a small box that can either contain a check mark or not.
- **Listclass** provides a list of items, which can be scrolled. From list of items, single or multiple items can be selected.
- A **menu bar** displays a list of top-level menu choices. Each choice is associated with a **dropdown menu**.
- The **JButton** class is used to create a push buttons.
- A **toggle button** is two states button that allows user to switch on and off. To create a toggle button in Swing we use **JToggleButton** class
- The **JTextField** component allows us to enter/edit a single line of text.
- The **JTextArea** is used to accept several lines of text from the user and it has capabilities not found in the AWT class.
- Swing provides a **combo box** (a combination of a text field and a dropdown list) through the **JComboBox** class, which extends **JComponent**. A **combobox** normally displays one entry.
- A **dialog** is defined as a conversation between two or more persons. In a computer application a dialog is a window which is used to "talk" to the application.
- **Message dialogs** are simple dialogs that provide information to the user.
- **File chooser dialog** box is used for navigating the file system.
- The class **JColorChooser** provides a pane of controls designed to allow a user to manipulate and select a color.

EVENT HANDLING :

- Changing the state of an object is known as an even.
- The `java.awt.event` package provides many event classes and Listener interfaces for event handling.
- Each event must return a Boolean value (true or false).
- An Event is an object that describes a state change in a source.
- **Event handling** is a process of responding to events that can occur at any time during execution of a program.

ADAPTERS :

- Adapter pattern is frequently used in modern Java frameworks.
- Adapter pattern works as a bridge between two incompatible interfaces.
- This pattern involves a single class which is responsible to join functionalities.

ANONYMOUS INNER CLASSES :

- Anonymous classes of Java are called anonymous because they have no name.
- A class that have no name is known as anonymous inner class in Java.
- An anonymous class is a local class without a name.