

# Syllabus for Diploma in Java-J2EE

## Course Duration For Java Training Course :

- 12 Weeks (Weekday Batch) or 15Weekends (Weekend Batch)

## Objective For Java Training Course :

- Getting the student to be well trained in Java Programming and get the IBM certified such that they have an easy entry in the IT Industry

## Eligibility For Java Training Course :

- BSc, BCS, BCA, BE, B.Tech, MSc, MCS, MCA, M.Tech



## Syllabus

### Core Java

#### Object Oriented Programming (OOPS) concepts

- ✓ Programming Languages
- ✓ Object Oriented Programming
- ✓ Classes & Objects
- ✓ Pillars Of Object Oriented Programming
- ✓ OOPS concepts and terminology
- ✓ Encapsulation & Examples
- ✓ Abstraction & Examples
- ✓ Inheritance
- ✓ Advantages of OOPS
- ✓ Fundamentals of OOPS

#### Core Java Programming Introduction of Java

- ✓ What is Java?
- ✓ Execution Model Of Java
- ✓ Bytecode
- ✓ How to Get Java?
- ✓ A First Java Program
- ✓ Compiling and Interpreting Applications
- ✓ The JDK Directory Structure
- ✓ Using Eclipse

#### Data types and Variables

- ✓ What are data types?
- ✓ Primitive Datatypes & Declarations
- ✓ Variables & Types
- ✓ Numeric & Character Literals
- ✓ String formatting and Parsing
- ✓ String Literals
- ✓ Arrays, Non-Primitive Datatypes
- ✓ Casting & Type Casting
- ✓ Widening & Narrowing Conversions
- ✓ The Dot Operator

#### Methods

- ✓ What are Methods?
- ✓ Method Structure
- ✓ Declaration Of Methods
- ✓ Calling Of Methods
- ✓ Defining Methods
- ✓ Method Parameters Scope
- ✓ Why static methods?

#### Control Flow Statements

- ✓ What are Control Flow Statements?
- ✓ Conditional (if) Statements
- ✓ Adding an else if
- ✓ Conditional (switch) Statements

# Syllabus for Diploma in Java-J2EE

## Operators and Expressions

- ✓ Expressions
- ✓ Assignment Operator
- ✓ Arithmetic Operators
- ✓ Relational Operators
- ✓ Logical Operators
- ✓ Increment and Decrement Operators
- ✓ Operate-Assign Operators (+=, etc.)
- ✓ The Conditional Operator
- ✓ Operator Precedence
- ✓ The Cast Operator

## while and do-while Loops

- ✓ for Loops
- ✓ A for Loop Diagram
- ✓ Enhanced for Loop
- ✓ The continue Statement
- ✓ The break Statement

## Object-Oriented Programming

- ✓ Concept & Syntax Of Class
- ✓ Concept & Syntax Of Methods
- ✓ Fields and Methods
- ✓ Constructors & Destructors
- ✓ Encapsulation
- ✓ Access Specifiers
- ✓ Access Control

## Objects and Classes

- ✓ Defining a Class
- ✓ Creating an Object
- ✓ Accessing Class Members
- ✓ Instance Data and Class Data
- ✓ Defining Methods
- ✓ Constructors
- ✓ Access Modifiers

## Using Java Objects

- ✓ StringBuilder and StringBuffer
- ✓ Methods and Messages
- ✓ Parameter Passing
- ✓ Comparing and Identifying Objects

## Data types and Variables

- ✓ Inheritance in Java
- ✓ Why use Inheritance?
- ✓ Types Of Inheritance
  - Single Inheritance
  - Multi-level Inheritance
  - Hierarchical Inheritance
  - Hybrid Inheritance
- ✓ Method Overloading
- ✓ Run-time Polymorphism
- ✓ Method Overriding
- ✓ Polymorphism in Java
- ✓ Types Of Polymorphism
- ✓ super keyword
- ✓ The Object Class & Methods

## Java Files and I/O

- ✓ What is a Stream
- ✓ Reading and Writing to Files (only txt files)
- ✓ Input and Output Stream
- ✓ Manipulating input data
- ✓ Reading Lines
- ✓ Opening & Closing Streams
- ✓ Predefined Streams
- ✓ File handling Classes & Methods
- ✓ Using Reader & Writer classes

## Interfaces and Abstract Classes

- ✓ What is an Interface
- ✓ Separating Interface and Implementation
- ✓ Implementing and Extending Interfaces
- ✓ Abstract Classes

## Inner Classes

- ✓ Nested Classes
- ✓ Anonymous Inner Classes
- ✓ Local Nested Classes
- ✓ Static Nested Classes
- ✓ Member Classes
- ✓ Instance Initializers

# Syllabus for Diploma in Java-J2EE

## Packages

- ✓ What is a Package?
- ✓ Advantages of using a Package
- ✓ Types Of Packages
- ✓ Naming Convention
- ✓ Steps For Creating Packages
- ✓ The import Statement
- ✓ Static Imports
- ✓ CLASSPATH and Import
- ✓ Defining Packages
- ✓ Package Scope

## Lambda Built-in Functional Interfaces

- ✓ Lambda Notation
- ✓ Lambda Syntax
- ✓ Lambda Expression
- ✓ java.util.function package
- ✓ Use primitive versions of functional interface
- ✓ Use binary versions of functional interface

## Exception Handling

- ✓ Exceptions Overview
- ✓ Exception Keywords
- ✓ Catching Exceptions
- ✓ The finally Block
- ✓ Exception Methods
- ✓ Declaring Exceptions
- ✓ Defining and Throwing Exceptions
- ✓ Errors and Runtime Exceptions
- ✓ Assertions

## Collection Framework

- ✓ The Collections Framework
- ✓ The Set Interface
- ✓ Set Interface Methods
- ✓ Set Implementation Classes
- ✓ The List Interface
- ✓ List Implementation Classes
- ✓ The Map Interfaces
- ✓ Map Implementation Classes
- ✓ Utility classes
- ✓ Generics in Java
- ✓ Primitive wrapper classes

## Threads

- ✓ Non-Threaded Applications
- ✓ Introduction to Threads
- ✓ Threaded Applications
- ✓ Creating threads
- ✓ Lifecycle Of A Thread
- ✓ Phases of Thread life cycle
- ✓ Runnable Interface
- ✓ Priority Of Threads
- ✓ Coordinating Threads
- ✓ Thread Interruption
- ✓ Thread Groups

## Swing

- ✓ Introduction To Swing
- ✓ Swing Features
- ✓ Heirarchy Of Java Swing Classes
- ✓ Swing GUI Components
- ✓ Packages Used In Swing
- ✓ Swing Control Classes & Methods
- ✓ Using Swing API
- ✓ Swing API MVC Architecture
- ✓ AWT v/s Swing
- ✓ Event Handling In Swing
- ✓ Event Listener Interfaces

## Applet

- ✓ Introduction to Applet
- ✓ The Applet Heirarchy
- ✓ Life Cycle of an Applet
- ✓ Lifecycle Methods for Applet
- ✓ A "Hello World" Applet
- ✓ Applet Layout Manager
- ✓ Bounding Box Concept
- ✓ Relative Coordinate System

## Agile Scrum Overview

- ✓ Introduction To Agile Methodology
- ✓ Scrum & Its Characteristics
- ✓ Sprints In Scrum
- ✓ Overview of Scrum Artifacts & Ceremonies

# Syllabus for Diploma in Java-J2EE



## JDBC

- ✓ Introduction To JDBC
- ✓ JDBC Architecture
- ✓ Types Of JDBC Drivers & Differences
- ✓ Common JDBC Components
- ✓ Importing Packages
- ✓ Registering JDBC Drivers
- ✓ Opening Connection
- ✓ Connecting a Java program to a Database
- ✓ Executing Query
- ✓ Statement Class & Objects
- ✓ Getting Information from Database
- ✓ Obtaining Result Set Information
- ✓ DML Operations through JDBC
- ✓ Cleaning up Environment

## Advanced Java

### HTML, CSS & Javascript Overview

- ✓ HTML Basics
- ✓ HTML Elements
- ✓ CSS Introduction
- ✓ CSS Syntax & Selectors
- ✓ Javascript Overview
- ✓ Bootstrap Overview
- ✓ Use the UnaryOperator interface

### Java MVC Architecture

- ✓ Three-tier architecture
- ✓ Introduction to MVC
- ✓ MVC Architecture
- ✓ Advantages of MVC
- ✓ Building pages with MVC

### Servlets

- ✓ What is a web application?
- ✓ What is a Servlet?
- ✓ Advantages of Servlet
- ✓ Servlet Class
- ✓ Servlet Lifecycle
- ✓ ServletContext Interface
- ✓ Advantages of ServletContext
- ✓ Session management
- ✓ Session Tracking Techniques
  - Cookies
  - Hidden field
  - URL Rewriting
  - HttpSession
- ✓ Building the first Servlet
- ✓ Deploying the Servlet
- ✓ Servlet Examples

### Hibernate

- ✓ Introductions to Hibernate
- ✓ Hibernate v/s JDBC
- ✓ What is Object Relational Mapping
- ✓ Hibernate Features
- ✓ Application Architecture
- ✓ Persistent Classes
- ✓ Object States
  - Transient State
  - Persistent State
  - Detached State
- ✓ Rules of Persistent Classes
- ✓ Mapping Collections
- ✓ Hibernate Mapping File Elements
- ✓ Types Of Mapping
  - One-to-one
  - One-to-many mapping
  - Many - one mapping
  - Many - to - many mapping
- ✓ Hibernate Query Language
  - Basic HQL Operations
  - Advantages of HQL
- ✓ Caching and Transactions
  - Types Of Cache
  - Hibernate Transaction Management
  - Transaction Interface In Hibernate
- ✓ Hibernate with Web Applications

# Syllabus for Diploma in Java-J2EE



## JSP

- ✓ Introduction of JSP
- ✓ JSP Architecture
- ✓ JSP Processing
- ✓ JSP Tag library
- ✓ Core Tags
- ✓ JSP Standard Tags
- ✓ JSP Page Life Cycle
- ✓ Creating the first Dynamic web page using JSP

## Spring

- ✓ Introduction of Spring Framework
  - Characteristics Of Spring
  - Spring Framework Architecture
  - Spring Framework Modules
  - Spring Platform Advantage
- ✓ Spring bean Wiring
  - Dependency Injection
  - IoC Containers
  - Spring Bean Lifecycle
  - Beans Auto Wiring
  - Autowiring Limitations
- ✓ Spring with database
  - Introduction to JDBC
  - Limitations of JDBC API
  - Spring JDBC Approaches
  - JdbcTemplate Class
  - JdbcTemplate Class Methods
  - Data Access Object (DAO)

## Struts2

- ✓ Introductions to Struts
  - What is a Web Framework
  - Limitations of JSP/Servlet Coding
  - MVC1 v/s MVC2
  - What is Struts
  - Invasive v/s Non invasive Frameworks
  - Modules in Struts 2
  - Request Processing Lifecycle
  - Struts1 v/s Struts2
  - Advantages of Struts2
- ✓ Struts2 MVC Architecture
  - Actions Component
  - Results & Result Types
  - The Value Stack
- ✓ Action Classes
  - Introduction to Action Class
  - Characteristics Of Action Class
  - ActionSupport Class
  - Action Interface
- ✓ Handling Application Requests
  - Handling Form Data
  - Struts2 Form
  - JavaBean Class
  - Success Page
  - Configuring Struts2
- ✓ Deployment Descriptors
  - The web.xml file
  - The struts.xml file
  - The struts-config.xml file
  - The struts.properties file

# Syllabus for Diploma in Java-J2EE



## SQL

- ✓ Introduction to RDBMS
- ✓ Features of RDBMS
- ✓ Introduction to SQL
- ✓ Basic Terminologies
- ✓ PLSQL Data Types
- ✓ Primary Key v/s Foreign Key
- ✓ The Unique Constraint
- ✓ DDL Statements
- DML Statements
  - Retrieving Records
  - Normalization
  - Functions in SQL
  - Handling Null Values

## Java Project

### Industry Java Project

As part of their projects, students will build an industry level software for online shopping cart called AmazonBunjee using Java and SQL Server. Students will use various modules of Java like OOPS of Core Java, servlets, JSP, session management etc. which will help them gain complete confidence in all the modules of Java.