Teaching Guidelines for

**Web-based Java Programming**

PG-DAC September 2021

**Duration:** 48 class room hours + 44 lab hours **(92 hours)**

**Objective:** To learn advanced concepts in java programming and perform web Programming using Java.

**Prerequisites:** Knowledge of core Java programming

**Evaluation:** 100 marks

**Weightage:** Theory exam – 40%, Lab exam – 40%, Internals – 20%

# Text Book:

* Core and Advanced Java Black Book / Dreamtech Press

# References:

* Servlet and JSP: A Tutorial by Budi Kurniawan / Brainy Software
* Spring in Action by Craig Walls / Manning Publications
* Advanced Java programming by Uttam K Roy / Oxford University press
* Sun Certified Enterprise Architect for Java EE Study Guide by Mark Cade & Humphrey Sheil / Pearson Education
* Professional Java EE Design Patterns by Murat Yener, Alex Theedom & Reza Rahman / Wrox

# (Note: Each Session is of 2 hours)

**Session 1:**

**Lecture:**

J2EE Overview

* J2EE Container
* Packaging Web applications
* J2EE compliant web application
* Deployment tools.
* Web application life cycle
* Deploying web applications.
* Web Services Support

## *No Lab*

## Sessions 2, 3 & 4:

## Lecture:

* Servlets: Dynamic Content Generation
* Advantages of Servlets over CGI
* Servlet Life cycle
* Servlet API & Deployment
* Servlet Annotations
* The Servlet interface
* The HttpServlet, HttpServletRequest, HttpServletResponse
* Exception Handling
* Servlet, DAO, POJO DB Layers
* Session
* Session Management
* Session Tracking with
  + Cookies
  + HttpSession
* Request Dispatcher
* Page Navigation
* Complete Case study Servlet Based

## Lab:

* Installing a servlet container (Tomcat)
* Adding Server to IDE
* Develop a structured dynamic web application(e.g. Library Management System) using servlets, deploy it in Tomcat
* Use HTTP Session in the Air Ticket Reservation System

*Reading****:*** Know more about the HTTP protocol at [www.w3c.org](http://www.w3c.org)

*Tutorial:* Compare which way of session tracking is better Cookies or HttpSession.

## Sessions 5 & 6:

## Lecture

* JSP: Separating UI from Content generation code
* MVC architecture
* Design Pattern: MVC Pattern
* Life cycle of a JSP page
* Directives, Implicit and Explicit Objects, Scriptlets, Expressions, Expression Language
* Scope
* JSP Error Page handling
* JSTL

## Lab:

* Separate UI code from the controller code in your Library Management System by incorporating JSP and Servlets.
* Complete the implementation of Air Ticket Reservation System.
* Implement MVC based web application using Servlet, JSP

## Sessions 7 & 8:

## Lecture:

JDBC & Transaction Management

* Introduction to JDBC API
* JDBC Architecture
* JDBC Drivers
* JDBC Classes& Interfaces: Driver, Connection, Statement, PreparedStatement, ResultSet and their relationship to provider implementations
* Stored procedures and functions Invocation
* SQL Injection overview and prevention
* Design Pattern: Data Access Object Pattern

**Lab:**

* Add Database CRUD operations to above MVC web application using JDBC Classes and interfaces. Use DAO and POJO Layers

## Sessions 9, 10 & 11:

## Lecture:

* Hibernate Framework
  + Introduction to Hibernate Framework
  + Architecture
* Hibernate in IDE
  + Creating web application using Hibernate API
  + Lifecycle of Hibernate Entities
* HB with annotation example
* Hibernate Mappings and Relationships
* Collection and Component Mapping
* HQL, Named Queries, Criteria Queries

**Lab:**

* Demonstrate Hibernate as standalone library in Java application
* Develop a web application (Online Bookshop) using Hibernate Persistence

*Reading:* Study Hibernate architecture from[**www.hibernate.org/docs**](http://www.hibernate.org/docs)

**Sessions 12, 13 & 14:**

**Lecture:**

* What is Spring Framework
* Overview of Spring Architecture
* Spring MVC architecture
* Spring Modules Overview
* Understanding Spring 4 annotations (Basic Introduction)
* What is IoC (Inversion of Control)
* IOC container
* Dependency Injection
* Spring Beans
* Autowiring Beans
* Bean Scopes
* Spring MVC
* Model, Model & View, HandlerMapping, ViewResolver
* Design Pattern: Front Controller Pattern
* Spring MVC Web application with JSP views (without Spring Boot)
* Using Thymleaf as alternate View Technology (only introduction)
* Spring Validations
* Spring i18n, Localization, Properties
* File Upload example

**Lab:**

* Design and deploy Library Management System using Spring Web

**Sessions 15 & 16:**

**Lecture:**

* Spring Boot essentials
* Why Spring boot
* Spring Boot Overview
* Basic Introduction of MAVEN
* Building Spring Web application with Boot
* Spring Boot in detail (Use Spring Boot for all demo & assignments here onwards)
* Running a web application using Spring Boot with CRUD (with Static Data not DB)
* Spring Data JDBC

**Lab:**

* Create Hello World Spring Boot Web application
* Check Libraries imported by Spring Boot
* Create Spring Boot CRUD application with Thymeleaf as View technology and Spring JDBC

**Sessions 17 & 18:**

**Lecture:**

Spring Data Module

* Spring Data JPA (Repository support for JPA)
* Crud Repository & JPA Repository
* Query methods
* Using custom query (@Query)

**Lab:**

* Add CRUD operations with Spring JPA etc. to earlier Spring Web application.

**Session 19:**

**Lecture:**

Spring AOP

* AOP Overview
* Spring AOP
* AOP Terminology and annotations: Advice, Join Points, Pointcuts, Aspects

**Lab**

* Modify earlier Spring MVC application to Log all the requests using AOP

**Sessions 20 & 21:**

**Lecture:**

Building REST services with Spring

* Introduction to web services
* SOAP Vs RESTful web services
* RESTful web service introduction
* Create RESTful web service in java using Spring Boot
* RESTful web service JSON example
* RESTful web service CRUD example
* Using POSTMAN client to invoke REST API’s
* REST service invocation using REST Template

**Lab:**

* Create REST API for Employee Management using Spring Boot
* Invoke it from POSTMAN app
* Invoke it from another Spring Boot Web application using REST Template

**Session 22:**

**Lecture:**

* Testing in Spring
* Unit Testing of Spring MVC Controllers:
* Unit Testing of Spring Service Layer
* Integration Testing of Spring MVC Applications: REST API
* Unit Testing Spring MVC Controllers with REST

**Lab:**

* Design & test Spring Application

**Session 23 & 24:**

**Lecture:**

**Securing Web Application with Spring Security**

* What is Spring Security
* Spring Security with Spring Boot
* Basic Authentication
* Authentication with User credentials from Database and Authorization

**Lab:** (2 hours)

* Secure the Spring Web application created in earlier exercise.