

Spring By Practical Examples

Environment setup (4 Hrs)

- Topics of discussion
 1. Installing and configuring Java 11
 2. Installing and configuring Maven (Build tool)
 3. Installing Git (Version Control tool)
 4. Installing IntelliJ Idea (IDE)
 5. Introducing common git commands (if needed)
 6. Introduction to Maven
 7. **Practice.**

Introduction to Servlet and JSP (4 Hrs)

- Topics of discussion
 8. Introduction to Servlet and Servlet Container
 9. Configuring Servlet Application
 10. Implementing HttpServlet and working with different HTTP Methods
 11. View rendering with Servlet and JSP
 12. Simple form data handling with Servlet and JSP
 13. **Practice.**

Spring HelloWorld Application (4 Hrs)

- Topics of discussion
 1. Downloading application server (Tomcat)
 2. Configuring tomcat server with IntelliJ Idea
 3. MVC architecture in Spring
 4. Introduction to Servlet & Dispatcher Servlet Configuration
 5. What is a Bean, how to work with it
 6. View Resolver Bean definition, working with different view resolvers.
 7. @Component and @Configuration annotations
 8. **Practice.**

Accessing Form Data with Spring (4 Hrs)

- Topics to discuss
 1. Introduction to Taglibs.
 2. Form Taglib, core Taglib, and other taglib examples
 3. Accessing the form data and rendering those data to page

4. Storing the form data in a list based storage, rendering them in the page
5. **Practice.**

Making a simple CRUD (4 Hrs)

- Topics to discuss
 1. Installing Postgresql locally
 2. Integrating JDBC Template to make a CRUD (Connecting with a PSql DB)
 3. Opening a Heroku account and creating a Postgresql DB in Heroku
 4. Setting up different profiles (Dev, QA, Prod etc)
 5. **Practice.**

Making advanced CRUD (8 Hrs)

- Topics of discussion
 1. Integrating and configuring Hibernate Session Factory, Transaction Managers etc.
 2. Performing the CRUD with Hibernate
 3. Using Criteria API
 4. **Practice.**

Integrating a UI Template (4 Hrs)

- Topics of Discussion
 1. Configuration to render Static contents
 2. Adding CSS, JS, images and other static contents
 3. Collecting data from Form and Rendering them into a page
 4. **Practice.**

Applying basic security (4 Hrs)

- Topics of discussion
 1. Integrating and Configuring Spring security
 2. Discussing Role Based Access Control feature (RBAC) in Spring framework
 3. Configuring Login, Log Out, default error pages
 4. Integrating custom filter and custom interceptor with spring security
 5. **Practice.**

Applying advanced security (4 Hrs)

- Topics of discussion
 1. Defining Custom Authentication Provider
 2. Defining Custom Authentication Success handler, Custom Logout handler
 3. Using Password encryption algorithms
 4. Role Based Menu access handling
 5. **Practice.**

Project deployment to Heroku (4 Hrs)

- Configuring heroku deployment environment

A template integrated web project development by participants (1) (8 Hrs)

- Grooming, development and evaluation

Understanding and creating RESTful API (4 Hrs)

- Topics of discussion
 1. Understanding GET, POST, PUT, DELETE, FETCH methods
 2. Understanding HATEOS
 3. Understanding the benefits and drawbacks of making API call over browser loading
 4. Developing APIs and testing with Postman
 5. Making HTTP requests with Ajax and/or fetch API
 6. **Practice.**

Making a simple angular application and calling RESTful API and Understanding the security measures of RESTful API (4 Hrs)

- Topics of discussion
 1. Configuring Angular environment and application
 2. Calling APIs from the angular application and rendering data
 3. Understanding the differences between Web and API security
 4. Different approaches to secure an API
 5. Discussing the JWT and OAuth 2 Security principles
 6. **Practice.**

Securing API with JWT (4 Hrs)

- Topics of discussion
 1. Implementing JWT security configuration in Spring
 2. Storing and passing JWT token from Server to Angular App to Server
 3. **Practice.**

Converting old project concept/developing a new project with API and Angular (8 Hrs)

- Redesigning and redeveloping old project concept to api driven single page application