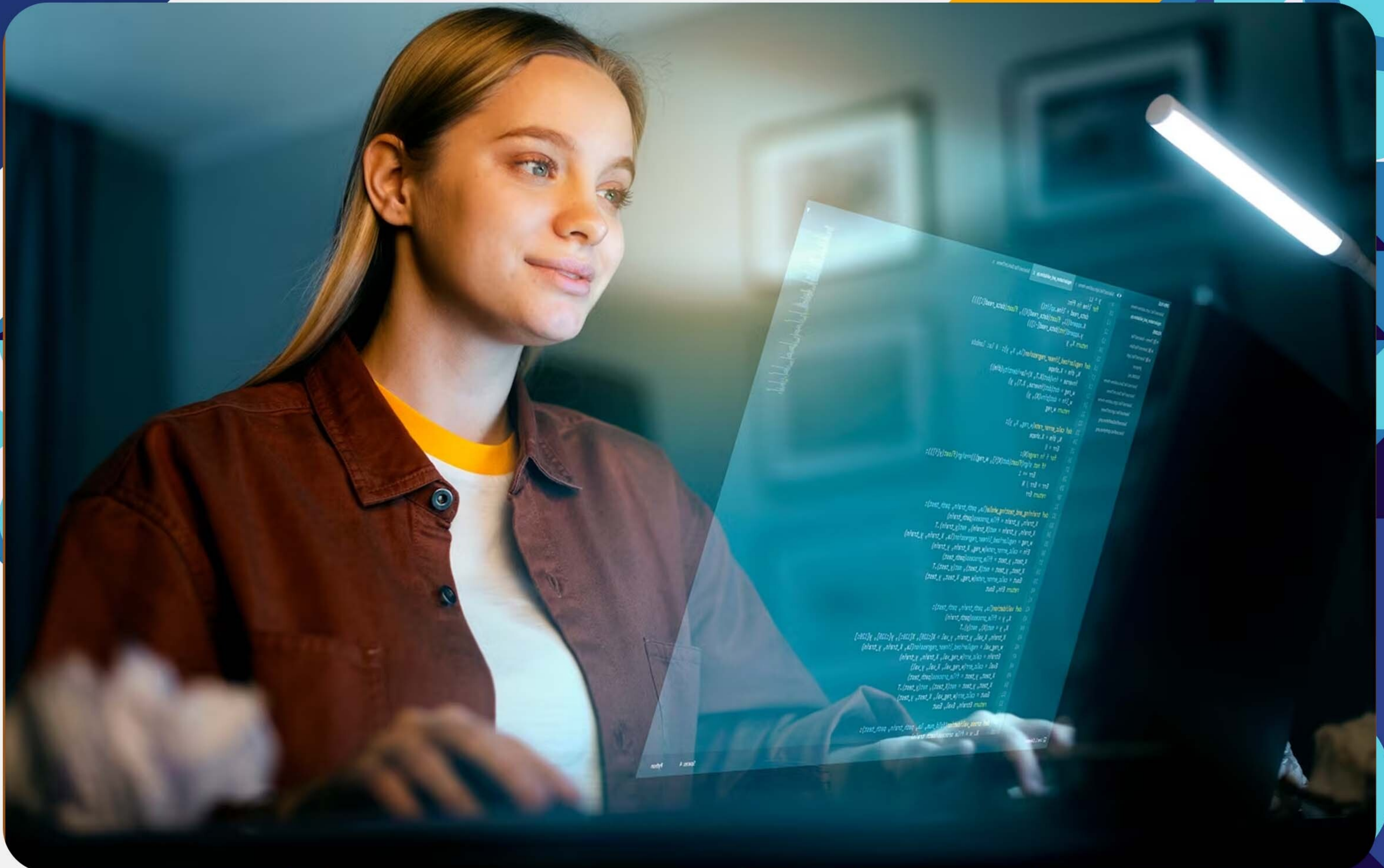




# Full Stack Development (Online & Offline)

**TRAINING DURATION**  
**6 Months**





# Full Stack Development (Online & Offline)

- ▶ About the company
- ▶ Learning outcomes
- ▶ What are the benefits of Work Experience
- ▶ Syllabus
  - Frontend Technologies
  - Backend Technologies
  - Database
- ▶ Mock Interview
- ▶ Placement Support
- ▶ Training Completion





## ABOUT THE COMPANY?

I2ispecialist Technologies Private Limited is a preferred software development & Digital Marketing Company in Pune. We have an excellent & well organized team of IT Experts who always believe in continuous learning. Discovering new ideas and implementing innovative developing and Digital Marketing strategies. I2ispecialist technologies provide 100% learning platform to help individuals develop their professional potential in the most engaging platform.

I2ispecialist Technologies focused on helping working professionals to learn, grow and move up in their career with deep knowledge of projects and architectures designed by IT Experts. We focus on practical sessions with core knowledge according to industry demand with latest technology. Learners will get the opportunities to work with industry experts on live projects. The internship has been designed keeping in mind that the future belongs to the developers who can think and implement end-to-end.

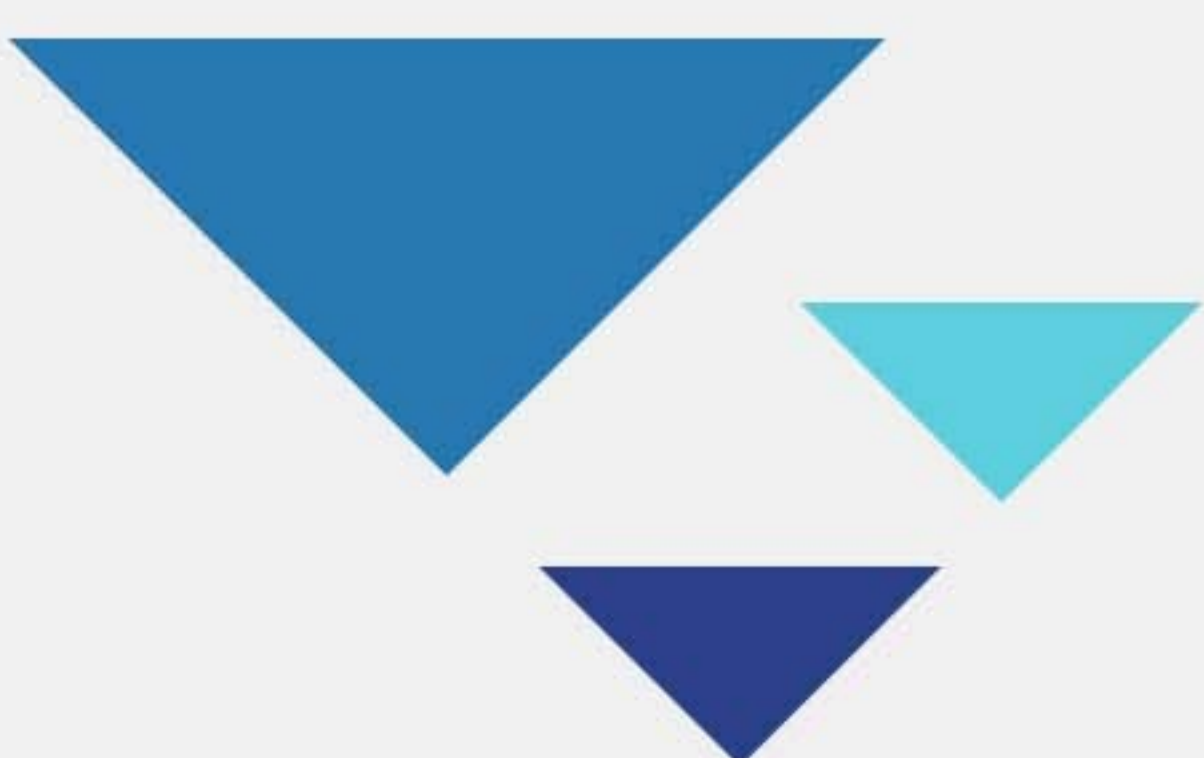
I2ispecialist belongs to Software development, Digital Marketing, Full Stack Development, Mobile Application.





## -WHY I2ISPECIALIST-

1. Expert IT Professionals
2. Technical Support Experts
3. Programing Experts
4. Demanding industrial oriented Architecture
5. Learn up to become a developer
6. Opportunities to work National & International project
7. 100% Knowledge Guaranty.
8. Focus to improve Logical thinking against programs.
9. Job Opportunities
10. Become an Software Expertise





## WHAT ARE THE BENEFITS OF WORK EXPERIENCE?

### 1. Employability

A set of achievements, skills, understandings and personal attributes that make graduates more likely to gain employment and be successful in their chosen occupations, which benefits themselves, the workforce, the community and the economy.

### 2. Employability skills include:

1. Self-management
2. Communication
3. Teamwork
4. Problem solving
5. Creativity & innovation
6. Taking responsibility
7. Commercial/business awareness
8. Social, cultural & global awareness
9. Time management
10. Decision making/initiative
11. Interpersonal skills
12. Motivation/drive
13. Flexibility/adaptability
14. Organisation/planning
15. Project management





# FRONT END DEVELOPMENT



## HTML:

### Coding Basics: Intro to HTML Syntax

- ✓ HTML, Head, Title, and Body Tags
- ✓ Headings and Paragraphs.
- ✓ The Doctype Declaration (DTD)
- ✓ The Lang Attribute
- ✓ Meta Tag: the Unicode Character Set

### Coding Links

- ✓ Anchor Tags and href
- ✓ Linking to Pages Within a Website
- ✓ Linking to Other Websites
- ✓ Opening a Link in a New Browser Window

### Adding Images

- ✓ The Image Tag and Source Attribute
- ✓ Using the Width, Height and Alt Tags
- ✓ Rounded Images

### Intro to Cascading Style Sheets (CSS)

- ✓ The Style Tag
- ✓ Tag Selectors
- ✓ Class Selectors
- ✓ The Class Attribute

### Intro to Java-script

- ✓ How to add JavaScript in html file
- ✓ Styling using JavaScript.

### HTML Forms

- ✓ HTML Forms attribute and elements
- ✓ Types of Input In HTML
- ✓ Input Attributes
- ✓ Styling of Forms

### Advance HTML or HTML 5

- ✓ Video tags
- ✓ Use of Audio tags
- ✓ How to add Google Location
- ✓ YouTube Player





## CSS:

### Introduction to Cascading Style Sheets

- ✓ Concept of CSS
- ✓ Creating Style Sheet
- ✓ CSS Properties
- ✓ CSS Styling(Background, Text Format, Controlling Fonts)
- ✓ Working with block elements and objects

### CSS Selectors

- ✓ ID selectors
- ✓ class Selector
- ✓ Universal Selector
- ✓ Grouping Selector

### Box Model

- ✓ Border and Box-shadow properties
- ✓ Padding and Margin Properties
- ✓ Margin properties

### CSS Development

- ✓ HTML Text, Font, Icon, Link, List Styling
- ✓ Tables style and Position
- ✓ Navigation and Drop-downs
- ✓ Image Gallery

### CSS 3 Advanced

- ✓ Rounded Corners
- ✓ Gradients Color
- ✓ Text Effects
- ✓ Transition
- ✓ Animation
- ✓ Pagination
- ✓ Media Queries

### CSS Responsive and Grid

- ✓ Viewport and Grid View
- ✓ Media Queries and Framework
- ✓ Container and grid item





# BOOTSTRAP:

## Introduction to Bootstrap

- ✓ What is Bootstrap and its version
- ✓ Starter template
- ✓ Linking jQuery, and CDN
- ✓ Responsive tags

## Bootstrap Layout

- ✓ Grid system and Container
- ✓ Media Object
- ✓ Utilities for Layout

## Bootstrap Content

- ✓ Typography
- ✓ Images
- ✓ Tables
- ✓ Figures

## Bootstrap Components

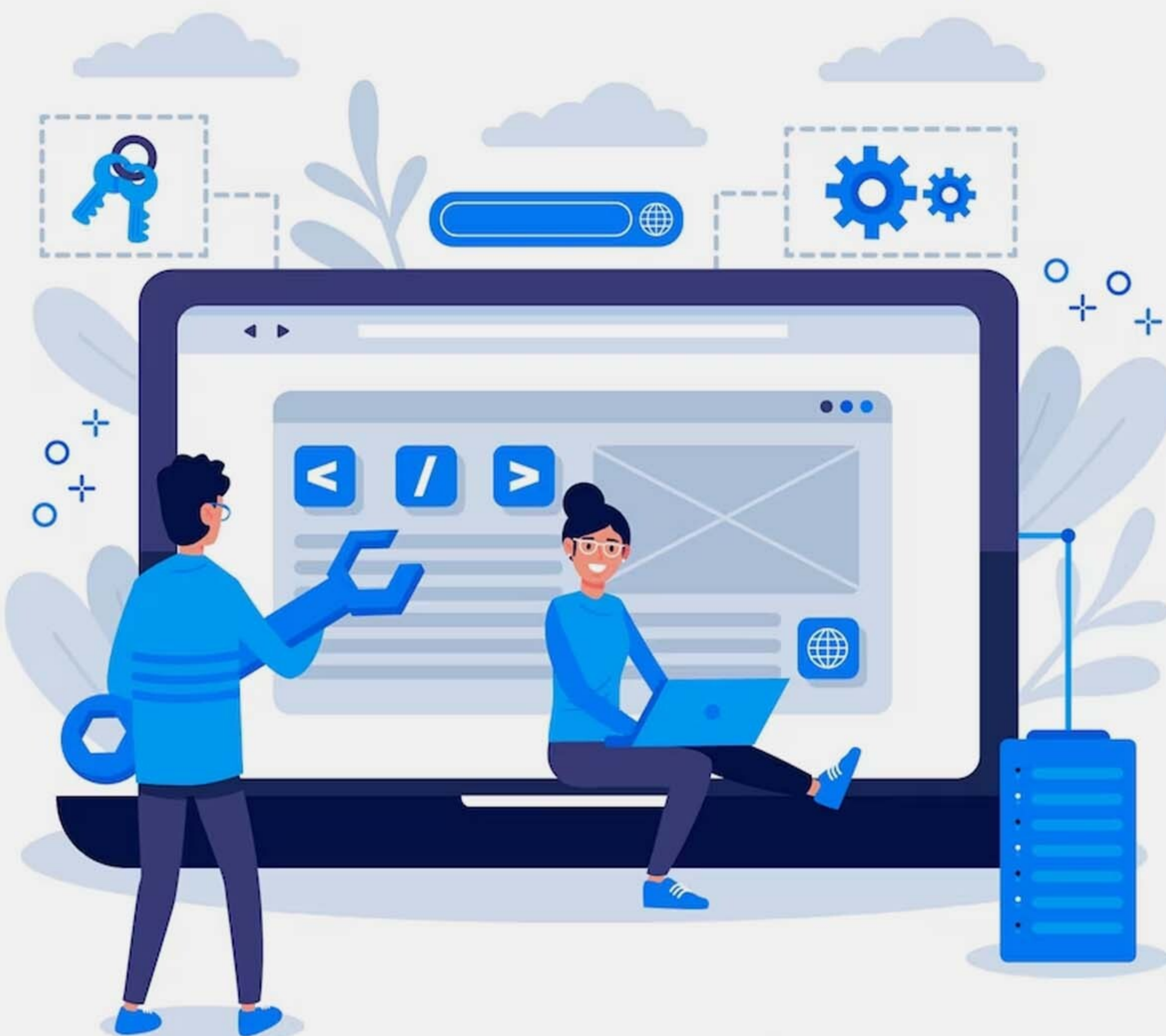
- ✓ Badge
- ✓ Breadcrumb
- ✓ Buttons and Button Group
- ✓ Card, Carousel, collapse bar and Dropdowns

## Bootstrap Navigation

- ✓ Jumbotron
- ✓ Container, Navbar
- ✓ Pagination and Progress Bar
- ✓ Scrollpy and Tooltips

## Bootstrap Forms

- ✓ Responsive Forms
- ✓ Input Classes
- ✓ Custom Forms
- ✓ Icons and Media objects







# JQUERY:

## Introduction

- ✓ Downloading and using jQuery files
- ✓ jQuery intro
- ✓ jQuery Selectors
- ✓ jQuery Event

## Effects in jQuery

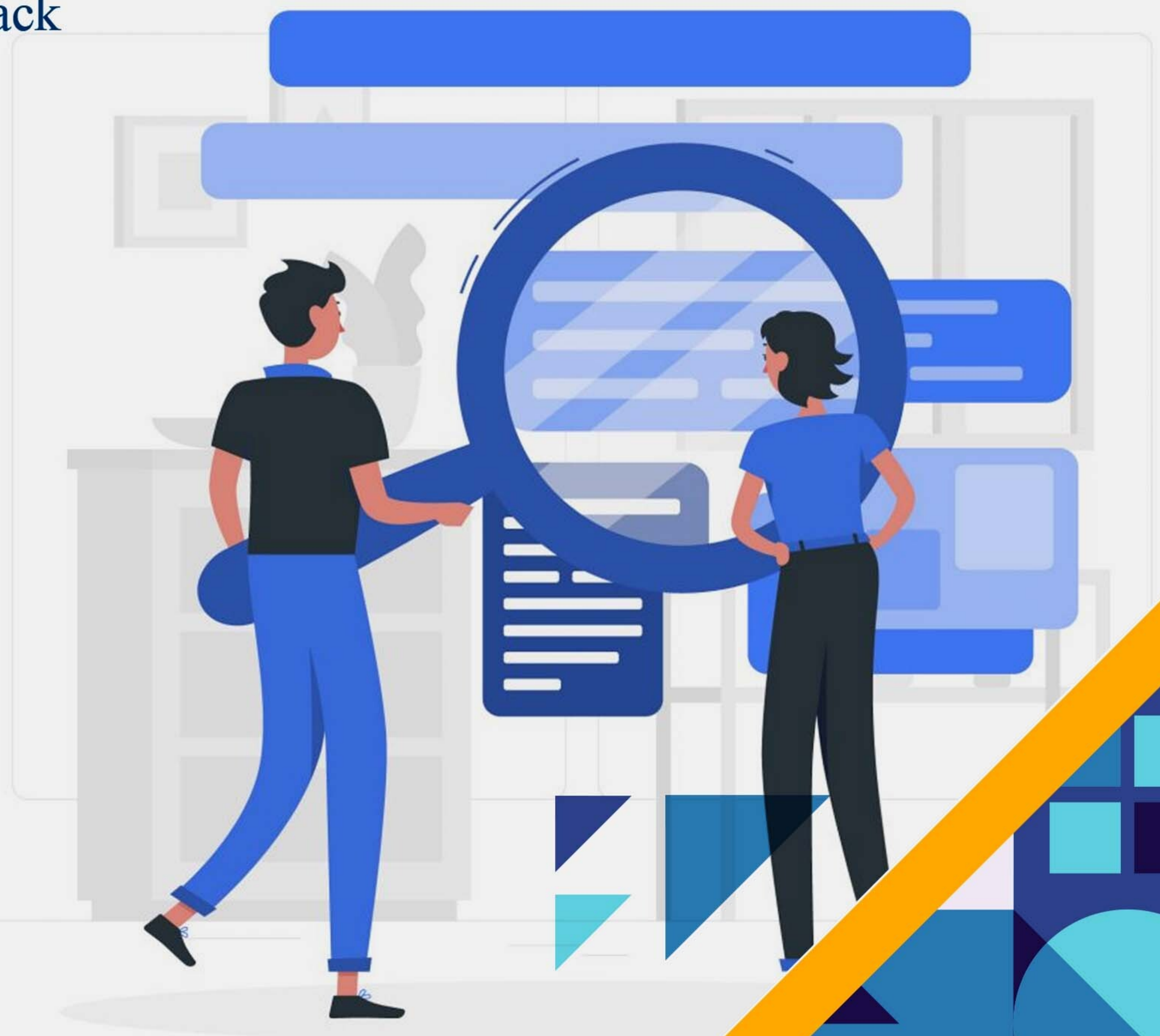
- ✓ jQuery Fade, slide
- ✓ jQuery Animate
- ✓ jQuery stop(), callback
- ✓ jQuery Chaining

## Traversing in jQuery

- ✓ Ancestors
- ✓ Descendants
- ✓ Siblings
- ✓ Filtering

## AJAX

- ✓ jQuery Intro
- ✓ jQuery load
- ✓ jQuery Get/Post





# JAVA SCRIPT:

## 1. Fundamentals of javascript

1. About Javascript Tutorial
2. Important Points about Javascript
3. Why Javascript!

## 2. Overview Of Javascript

1. History of Javascript
2. First Program in Javascript, Hello World
3. Places to put JavaScript code
4. Comments in Javascript
5. Output Display in Javascript
6. Asynchronous programming in JavaScript ?

## 3. Data Types In Javascript

1. Data Types Introduction
2. Numbers
3. Booleans
4. JavaScript Strings

## 4. Variable in Javascript

1. Variable Introduction
2. Local and Global

## 5. Operators in Javascript

1. Operators Introduction

## 6. Decision Making of Javascript

1. Decision Making Introduction

## 7. Loop control in Javascript

1. Loop Control Introduction
2. JavaScript While Loop
3. JavaScript Do-While Loop
4. JavaScript For Loop
5. JavaScript For-of Loop
6. JavaScript For-in Loop

## 8. Control Flow in Javascript

1. break statement
2. continue statement

## 9. Array in Javascript

1. Array Introduction

## 10. Function in Javascript

1. make fullscreen
2. exit fullscreen
3. Event Listener in javascript
4. Function Introduction
5. Closure

## 11. String and String functions in Javascript

1. String Introduction

## 12. Recursion in Javascript

1. Recursion Introduction

## 13. OOP Concept

1. Oop Concept Introduction

## 14. DOM Manipulation

1. Introduction to DOM
2. DOM Selectors for Single Elements
3. DOM Selectors for Multiple Elements
4. Traversing the DOM
5. Traversing the DOM
6. Creating new HTML elements
7. getElementById()
8. getElementsByClassName() Method

## 15. JavaScript Events

1. JavaScript Events
2. DOM Load event
3. mouseover event Listener
4. mouseout event Listener
5. mousedown event Listener
6. Keydown event Listener
7. focus event Listener

## 16. AJAX and Javascript

1. AJAX and Javascript
2. AJAX with JSON
3. XHR Request
4. Update the DOM using AJAX
5. XML data using AJAX
6. JSON Files and AJAX





# ANGULAR:

## Typescript

- ✓ Generics
- ✓ Enums
- ✓ Modules

## Components

- ✓ Creating components
- ✓ Sharing data in components
- ✓ Use of Event Emitter
- ✓ Component Life Cycle

## Angular Templates

- ✓ Attribute class & style binding
- ✓ Two-way binding
- ✓ Template reference variable
- ✓ SVG as Template
- ✓ Pipes

## Directives

- ✓ Built-in directives
- ✓ Attribute directives
- ✓ Structural directives

## Data Binding

- ✓ Installing and using Bootstrap
- ✓ One way and two way data binding
- ✓ Working with ngModel

## HttpClient

- ✓ Calling REST-API
- ✓ Module of HttpClient
- ✓ CRUD operations using HttpClient

## Form

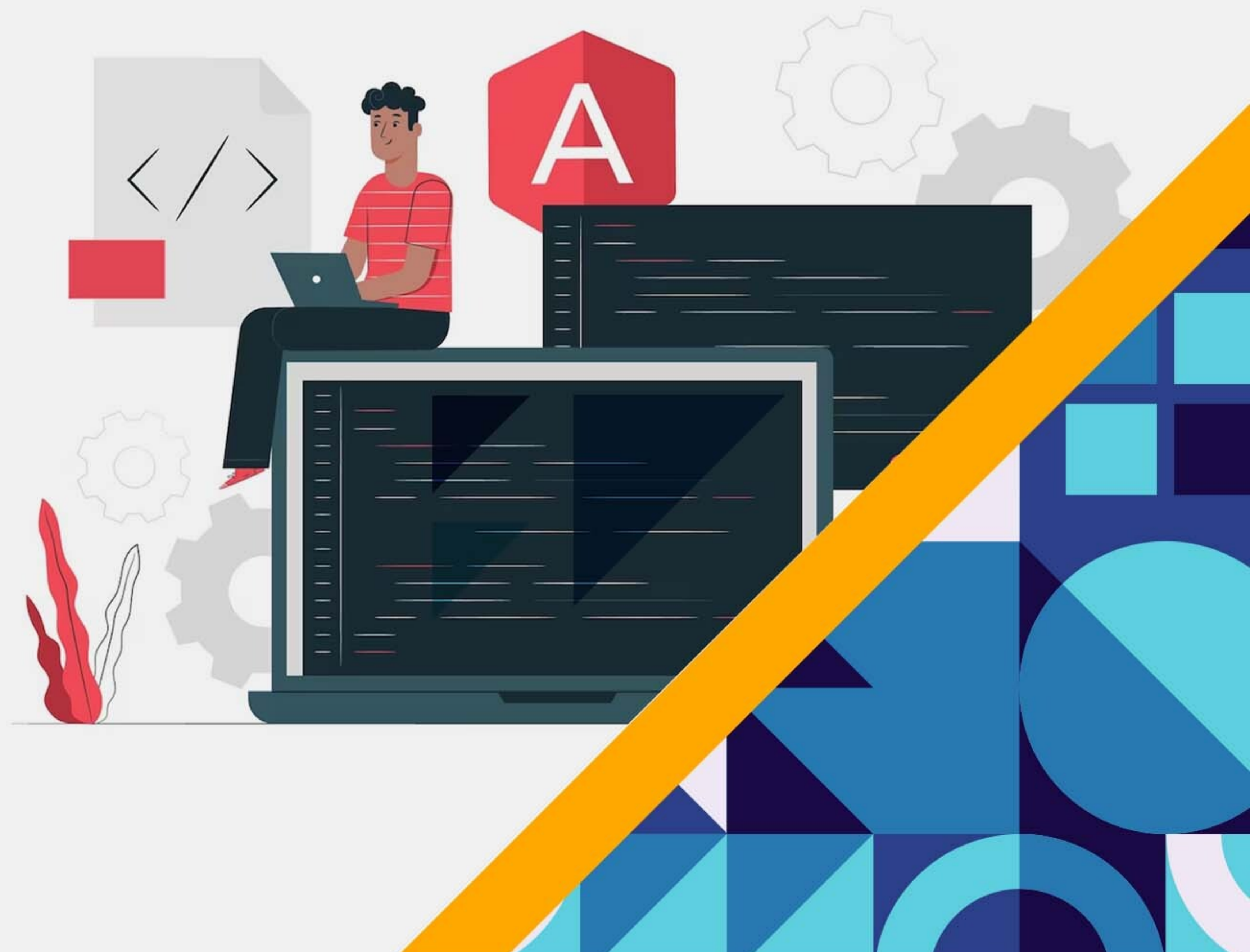
- ✓ Reactive
- ✓ Template-driven
- ✓ Form Validation

## Services

- ✓ Introduction to services
- ✓ Dependency Injection
- ✓ Creating custom service using injectable and providers

## Observable

- ✓ Introduction to RxJS
- ✓ Understanding observables
- ✓ Using observables and RxJS
- ✓ Creating custom observables and subscribe
- ✓ Unsubscribing observable





# BACKEND DEVELOPMENT

## Core Java



### Introduction To Java

- History of Java
- What is Java , Java Flavors, characteristics
- JVM Architecture
- Bytecode
- Class Loader
- Unicode
- Class path
- Path

### Fundamentals Of Java Programming

- Object oriented concepts ( OOP )
- Keywords, Datatypes, Variables, Operators, Casting
- Selection statement (if, switch )
- Control statements ( while,



- do while , for )
- Conditional statements ( if, else, elseif, ?
- static
- Arrays

### Object Oriented Programming with java

- Classes and Objects
- Structure of a class – its internals ( Data Members, methods )
- Using static
- Constructor
- this keyword
- modifiers
- playing with the object ( copying, casting )
- Garbage collection
- Abstract class

### Inheritance

- Basics (extends keyword )
- Modifiers and their scope
- Deriving a class
- super, final keyword
- why java does not support multiple inheritance?

### Polymorphism

- overloading a method
- overloading a constructor
- method overriding
- accessing base class method





## Packages & Interfaces

- basics
- modifiers and their scope chart
- setting classpath
- compiling and accessing a packaged class
- types of packages
- user defined package

## Exploring Java.lang Package

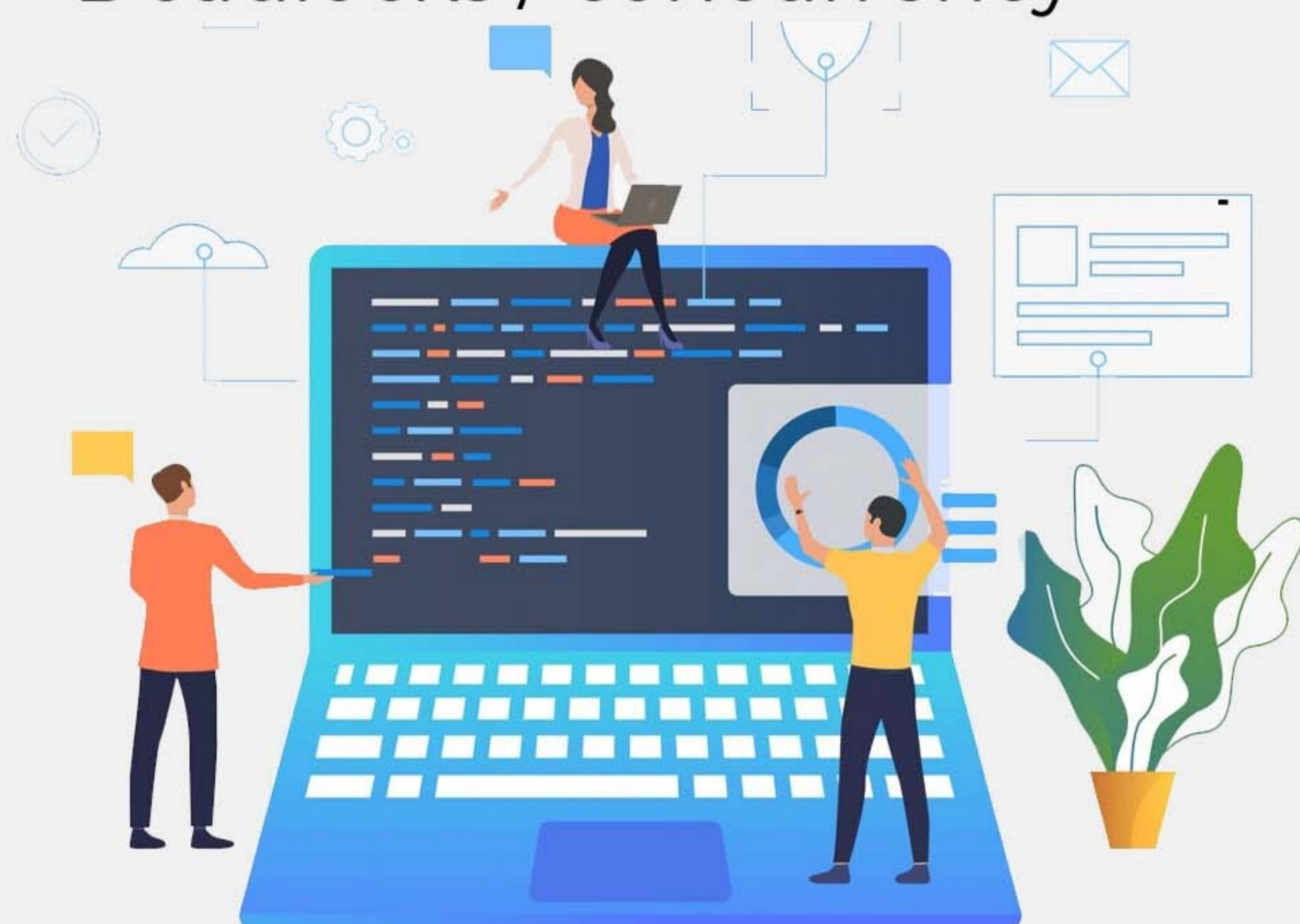
- String, StringBuffer, Arrays,
- Wrapper classes

## Exception Handling

- Basics
- Hierarchy of exceptions
- Handling exception – Try, catch, finally, throw, throws
- User defined exceptions

## Threads

- Basics, Thread class, Runnable Interface
- Thread model
- Life cycle – start(), run()
- Scheduling
- Deadlocks / concurrency



issues

- Synchronization – as a block, as a modifier
- Daemon thread

## I/O Streams

- Introduction
- Hierarchy of streams
- IO Stream, Byte Stream, Character Streams
- BufferedInputStream, BufferedOutputStream
- Reader and Writer class
- BufferedReader, PrintWriter
- Serialization

## Collection Frameworks

- basics, hierarchy
- legacy classes – Vector, Queue, Stack, Enumeration, Dictionary, Properties
- List, ArrayList, LinkedList
- Set, HashSet, TreeSet,
- Map, HashMap, TreeMap,
- Generics
- Annotations
- Boxing / Unboxing
- Enums

## Introduction To Functional Style Of Programming



- Mini application – discussion / implementation – 10 hrs
- Tools
- Javadoc
- javap
- jar
- IDE Tools
- eclipse
- myeclipse



## Spring

### Spring core

- Types of containers
- Xml configuration
- Core annotations
- Component – scan
- What can be injected into a bean
- Understanding core annotations

### Spring MVC

- Handler Mapping
- Controllers
- View resolvers
- Validators
- interceptors

### Spring DAO

- Jdbc templates
- Exception hierarchy

### Spring ORM

- Hibernate template
- Integration with hibernate

### Spring AOP

- Point cut
- Advisors
- Types of advices

### Spring -Webservices (Restful Api.)

### Spring Security

### Spring JPA





## Hibernate

### Introduction

- Introduction to ORM ( Object Relational Management )
- Benefits of ORM
- Contrast JDBC with ORM
- Hibernate Jar files
- Hibernate Architecture
- SessionFactory , Session, Transaction, Query
- Understanding config files .hibernate.xml, .hbm.xml
- Performing CRUD operations
- load, get, save, saveorupdate, delete

### HQL

- working with various SQL commands
- accessing more than one object from the table

### Mapping

- Inheritance
- Collection
- Composition
- Association

### Cache

- Level – I and II
- Using third party to configure L2 cache

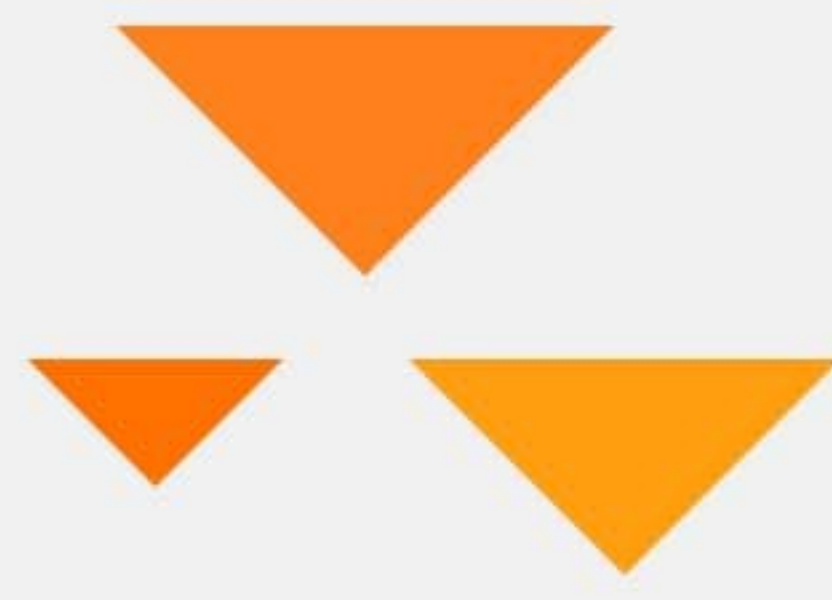


### Criteria

- Interfaces / classes
- implementation
- Note : an application using the above technologies –
- Domains : Telecom , LMS, etc...

## Hibernate Session





# Spring Boot

## Introduction



- Spring Boot overview
- Bootstrapping an Boot application –
- Initializer
- Auto Configuration
- Spring Boot annotations
- Spring Boot properties

## Spring Boot Profiles

- Accessing data with Spring Boot and h2 database
- Configuring Spring Boot with spring MVC application
- Building a Restful web application with Spring Boot
- Highlights of Training

## An Application Based On Spring Boot-Either Migration/new one

- Industry experienced Professional
- Hands-on experience with Project orientation
- Interview based Questions
- Advanced Java Training Syl-

labus

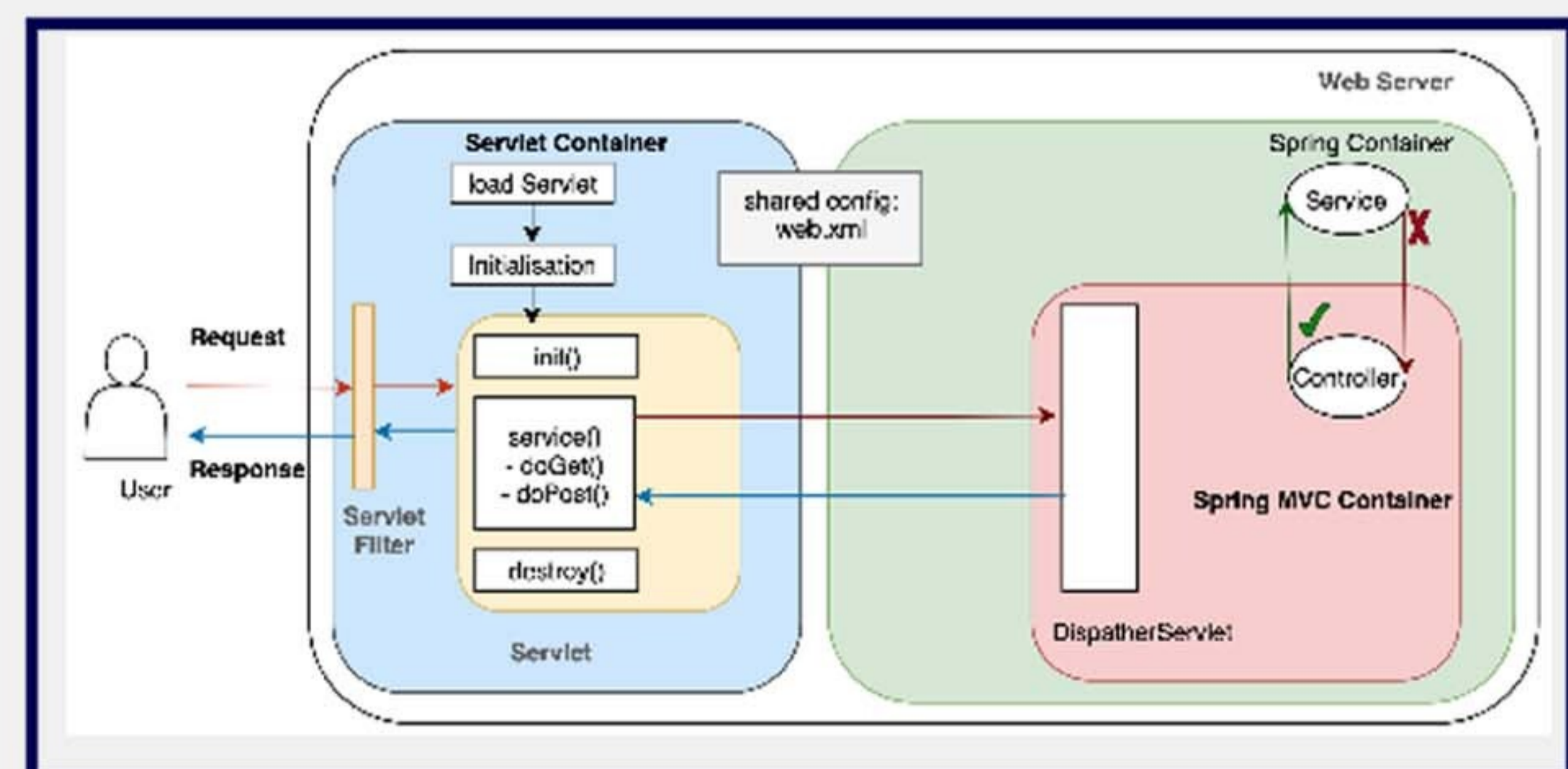


## Fundamentals

- Client, Server, Port, Application, Web Server, Application Server, Web Client and Web Application
- Installation of Java and Understanding PATH and CLASSPATH
- Installation of Eclipse
- Installation of Tomcat and structure of Tomcat
- Linking any Web Server with Eclipse
- Static Resource Access from Tomcat

## Servlets

- Servlet Internals and Servlet Programming
- What are Servlets and why do we need Servlets?
- Servlet Development with



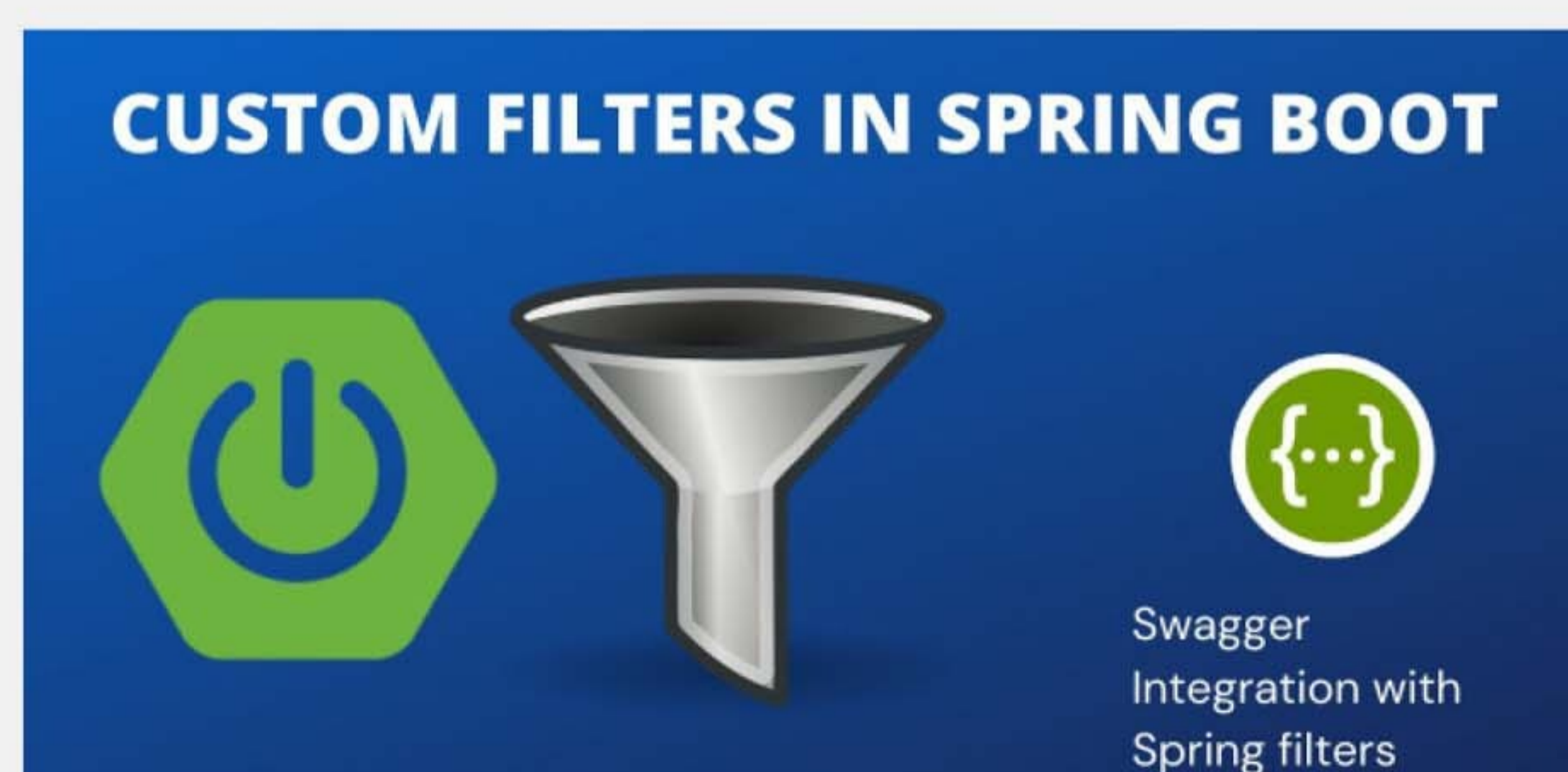
- Eclipse
- Developing Servlets
- Compiling Servlets





- Deployment and Invoking of Servlets
  - Purpose of Servlet Mapping and web.xml
  - Servlet Architecture
- Detailed Architecture of Servlets
- Discussion of GenericServlet, HttpServlet Interfaces
- Programming examples
  - Servlet Life Cycle
  - HTTP Internals
  - Detailed discussion of all parameters of request and response packets
  - Discussion of HttpServletRequest and HttpServletResponse interfaces
  - Session Tracking API
- Practical visualization of Http Session objects and Session table
- Session Tracking API
- Session Tracking Programming
  - Cookies, URL Rewriting and Hidden Form Fields
  - ServletConfig Object
  - ServletContext Object
  - RequestDispatcher Object

## Filters



- Why do we need filters?
- Filter, FilterChain, FilterConfig interfaces
- Programming examples
  - Application / Attribute Lifecycle Listeners
  - Why do we need Lifecycle Listeners?
  - ServletContext and ServletContext Attribute Listeners
  - Session and Session Attribute Listeners
  - Programming examples

## JSP

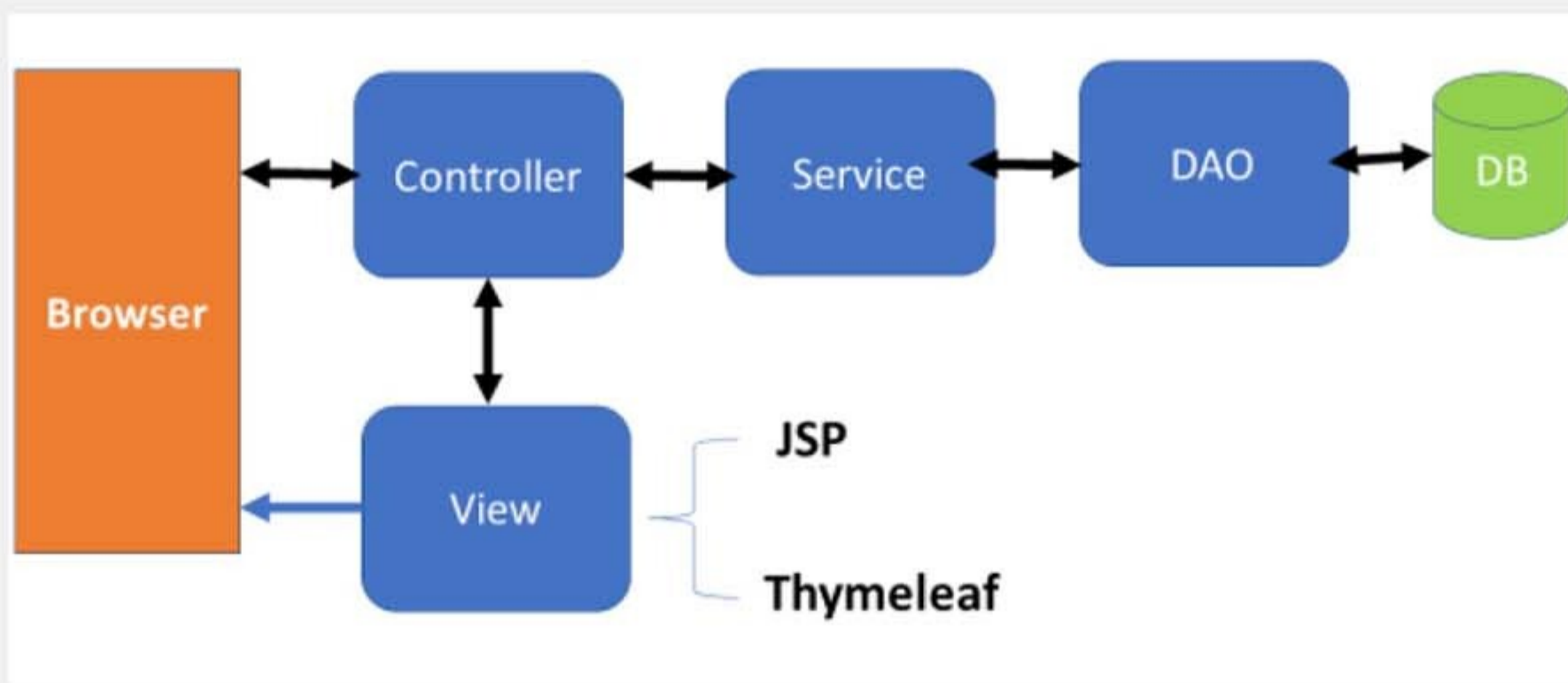
- Java Server Pages( JSP ) Internals and Programming
- What is JSP? Why do we



- need JSP?
- JSP development with Eclipse
- JSP Life Cycle
- Predefined Objects of JSP
- request, response, out and session objects
- config, application, page-context objects
- Scripting Tags ( Writing Java code inside JSP )
- Directives
- @page, @include directives

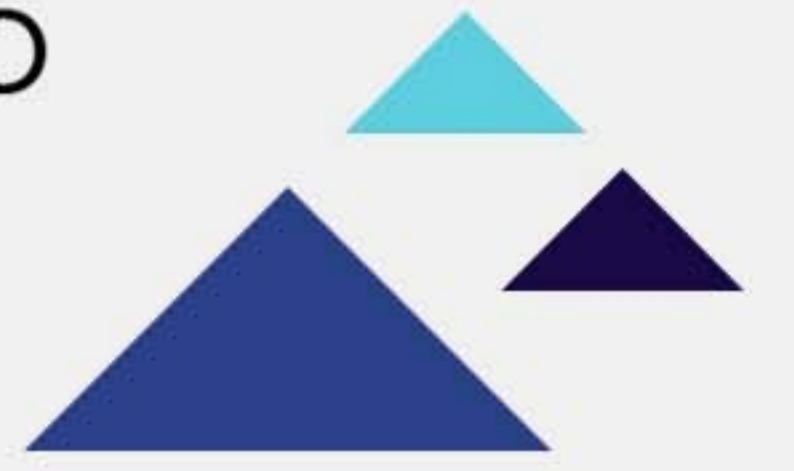


- Action tags of JSP
- Difference between jsp:forward, jsp:include and @include tags
- Programming examples to illustrate each tag
- Expression Language
- Why do we need EL?



- Reading scoped attributes using EL
- Reading bean properties using EL
- Reading collections using EL
- Operators of EL
- Programming examples to illustrate each usage
- JSTL ( JSP standard tag library )
- Why do we need JSTL?
- Looping tags
- Control tags
- Database tags
- T18N tags
- Programming examples to illustrate each category of tags
- Custom tags
- Custom tag creation based on Servlets
- Custom tag creation based on JSP

- concept of tld file
- programming examples to illustrate each concept



## Java Beans

- What are Java beans and why do we need Java beans?
- Action tags for java beans
- Application of Java beans

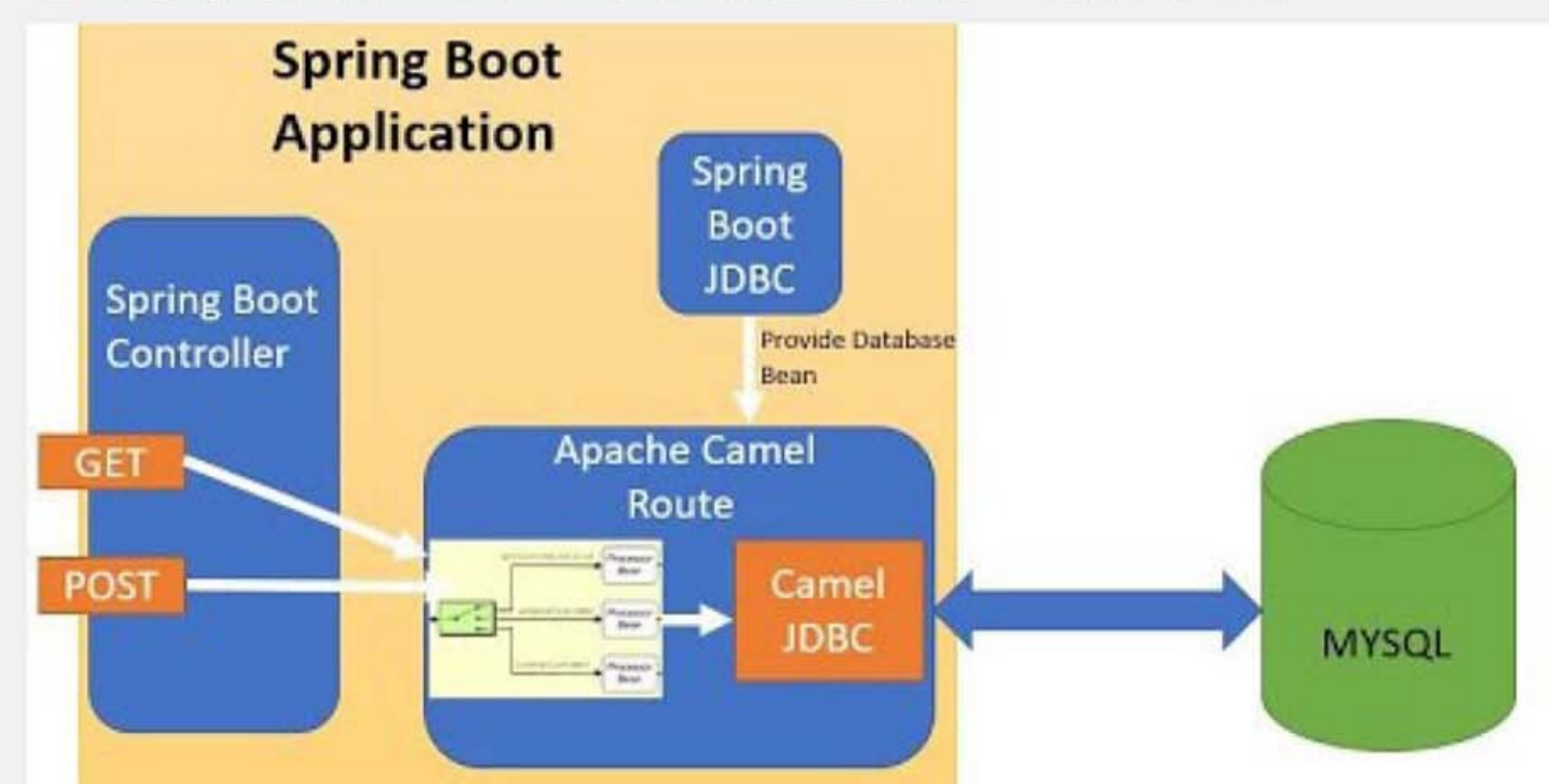
## MVC Architecture

- Why do we need MVC?
- Discussion of pros and cons of Servlets, JSP and JavaBeans
- Programming example to illustrate Business logic, Controller logic, Presentation logic and Bean creation

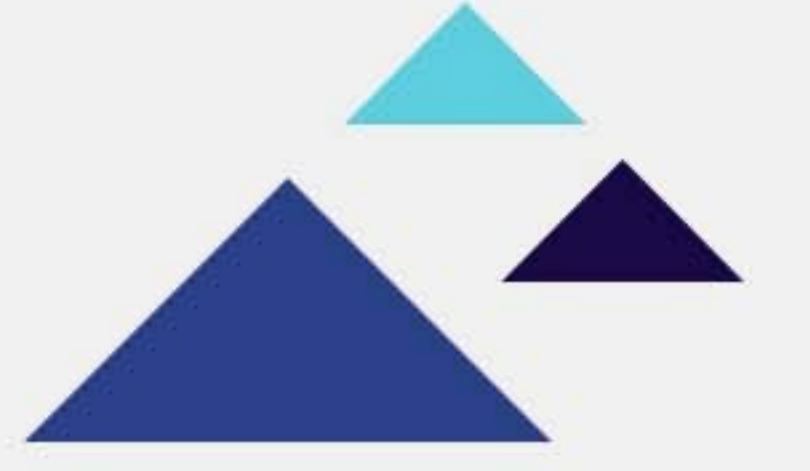
## JDBC

- Java Data Base Connectivity ( JDBC ) internals and programming
- What is JDBC ? Why do we need JDBC ?

Discussion of JDBC driver



## Types & Driver Manager



- Connection object
- Statement types and Discussion of each type of Statement Objects
- ResultSet Types and Discussion of each type
- Stored Procedure Access from java applications
- Transaction Programming with JDBC
- Savepoint and RowSet
- DataSource and Connection Pooling
- Batch Updates
- JNDI
- Explanation on Usage of important Practical Tools
- Eclipse
- Jboss, Weblogic, Tomcat





## Database

### Introduction to MySQL

- MySQL Installation
- DDL Commands
- DML Commands
- Operators
- Functions
- Constraints
- Joins
- Views
- Indexes
- DCL Commands
- TCL Commands
- MySQL Cursors
- Exception Handling
- Stored Procedure
- Stored Functions

## MicroServices

- MicroServices Introduction
- Principle and Characteristics
- Use cases and Benefits
- Challenges
- Design standards
- Micro Services Communication
- Pitfalls

## Tools

- Agile with Jira
- GitHub
- Gradle
- Maven
- Heroku

## Cloud-Application Deployment On Cloud

- AWS Cloud
- What is AWS
- Introduction to AWS Services
- Amazon Cloud Services
- Create EC2 Instance in AWS
- Deploy Spring Boot Application on AWS | Elastic Beanstalk
- AWS Advanced Research and Development- Live Deployment





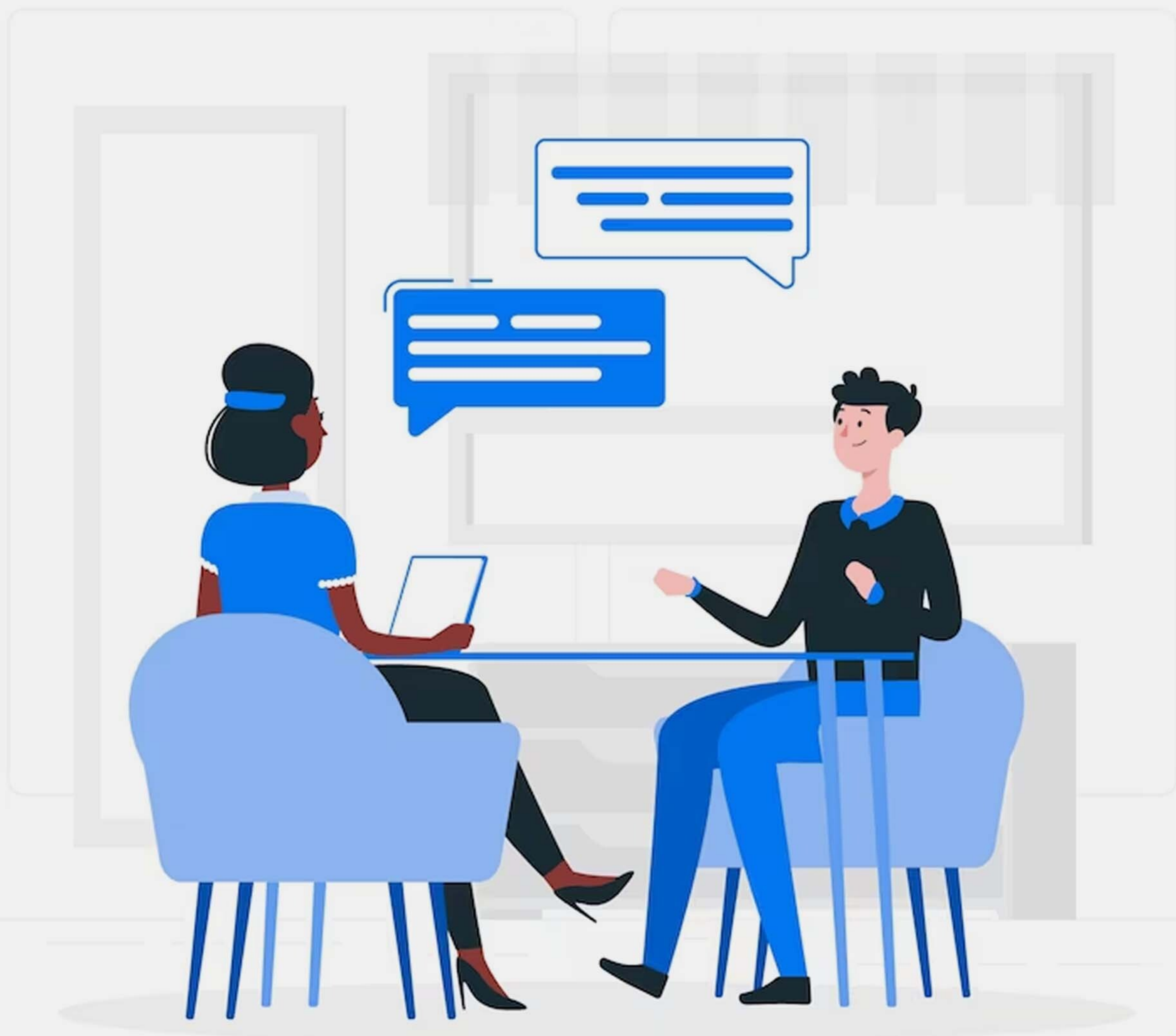
# MOCK INTERVIEWS

1. COMMUNICATION SKILLS
2. INTERVIEW QUESTIONS PRACTISE
3. SOFT SKILL
4. PORTFOLIO TECHNIQUES



# 100% PLACEMENT SUPPORT

1. RESUME PREPARATION
2. INTERVIEW CALLS
3. MAILING COMMUNICATION
4. AI BEST RESUME BUILDING

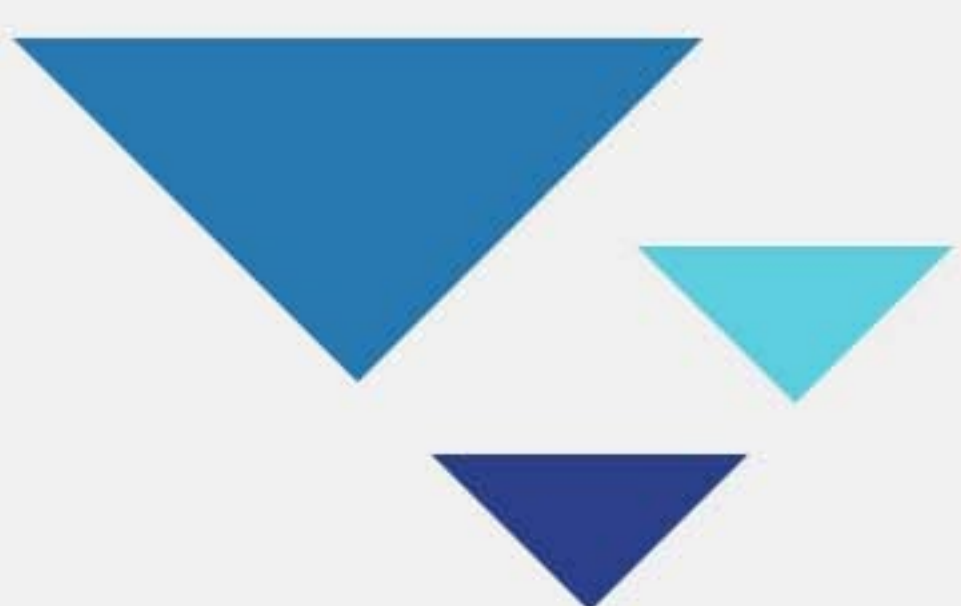




## TRAINING COMPLETION

At the end of the Course

- Working experience letter
- Internship experience letter
- Free Life-Time Training Access
- Job Opportunities.





**i2i** SPECIALIST  
TECHNOLOGIES  
PRIVATE LIMITED



### ADDRESS

A/11A, 6TH FLOOR, CITY-VISTA,  
FOUNTAIN ROAD, KHARADI PUNE 411014

### MAIL ID

hr@i2ispecialist.com | info@i2ispecialist.com

### CONTACT

+91 83296 65280

### WEBSITE

[www.i2ispecialist.com](http://www.i2ispecialist.com)

