# JAVA FSD - REFERENCES and CHEATSHEETS

## CORE JAVA

***JAVA CHEATSHEET***

### JAVA BASICS

- How to create, build and run a Java Hello World program with Eclipse
- Variables | Primitive Datatypes | Data Types |Conversions and Promotions |Operators
- if-then-else Statements | for Statement | switch Statement
- Arrays | Declare an array of arrays

### OOPS

- Declaring Classes | Defining Methods
- Controlling Access to Members of a Class
- Constructor | Initializing Fields | static
- Enums | Enum Types
- Inheritance | Polymorphism
- Class Object
- Abstract Methods and Classes | Interfaces

### UTILITY CLASSES

- Autoboxing and Unboxing
- Java StringBuilder | StringBuffer in Java
- Date Classes | Instant Class | Period and Duration | Temporal Adjuster
- Concepts of TDD in detail

### EXCEPTION HANDLING

- Exceptions
- Specifying the Exceptions Thrown by a Method
- How to Throw Exceptions

### COLLECTIONS

- Introduction to Collections | Collections Framework Overview
- Generic Types
- The List Interface
- The 4 Methods for Iterating Collections in Java
- Object Ordering
- The Set Interface
- Map Interface
- The Queue Interface

### JAVA 8 FEATURES

- Functional Programming in Java
- Functional Interfaces and Lambda
- Java Streams

### JAVA CHEATSHEETS

- Cheatsheet-1
- Cheatsheet-2

# JAVA FSD - REFERENCES and CHEATSHEETS

## WEB PROGRAMMING

### HTML

***HTML CHEATSHEET***

- HTML Basics
- HTML Structuring the web

### CSS

***CSS CHEATSHEET***

- CSS basics
- Adding Fonts | Styling Text - Fundamental text and font styling
- CSS values and units
- CSS Specificity
- Deep understanding on CSS selectors
- Adding Images to Web Documents
- CSS Flexbox | CSS Grid
- ChromeDev Tools
- Using Media Queries

### BOOTSTRAP

- Getting Started with Bootstrap
- Layout using BootStrap

## JAVASCRIPT

***JAVASCRIPT CHEATSHEET***

### JAVASCRIPT BASICS

- What is JavaScript?
- Variables, Data types and Literals
- Control Flow, Loops and iterations

### ARRAYS, FUNCTIONS, OBJECTS AND DOM

- Working with arrays
- Functions | Self-invoking Functions
- Working with Object | Prototypes and Inheritance
- Working with JSON
- Working with String | Working with Numbers
- JavaScript Document Object Model (DOM) and Events

### ES6

- Moving Towards ES6
- Introducing asynchronous JavaScript
- JavaScript Asynchronous Programming
- Fetch API
- Using the Fetch API
- Fetch API Introduction

# JAVA FSD - REFERENCES and CHEATSHEETS

## ANGULAR

***ANGULAR CHEATSHEET***

### TYPESCRIPT

- Short read on basics of Typescript
- Why use Typescript in Angular

### SPA

- Why people like SPAs
- Angular Single Page Applications (SPA): What are the Benefits?
- Modern vs Traditional Web Development
- Single Page Applications vs Multiple Page Applications — Do You Really Need an SPA?

### ANGULAR ARCHITECTURE AND BASICS

- Introduction to the Angular Docs
- Angular Architecture Overview
- Introduction to Angular concepts
- Create a feature component
- Introduction to components and templates
- Angular Architecture - Smart Components vs Presentational Components
- Interpolation and template expressions
- Interpolation vs Property Binding`
- Angular.io

### DIRECTIVES

- Directives in Angular
- *ngIf Directive in Angular
- NgSwitch Directive in Angular
- NgSwitch Directives
- *ngFor Directive in Angular
- Angular Attribute Directives: A Practical Approach
- Building Custom Directives in Angular
- Using @HostBinding and @HostListener in Custom Angular Directives

### COMPONENT INTERACTION

- Angular: Component interaction with @Input, @Output and EventEmitter
- Component Interaction with Outputs in Angular
- Component Interaction with Inputs in Angular
- Animated Playground for RxJS

### SERVICES

- Introduction to Services and Dependency Injection
- Dependency injection in Angular

# JAVA FSD - REFERENCES and CHEATSHEETS

## REST CALLS TO API'S

- Introduction to Angular' s HttpClient
- Introduction to Observables in Reactive Programming
- Angular HTTP Client- QuickStart Guide
- Introduction to RxJs Observables
- Differences between Observer and Behaviour Subject

## FORM HANDLING

- Overview of Forms in Angular
- Quick Introduction to Template-Driven Forms in Angular
- Reactive Forms with Angular
- Reactive Forms in Angular: Listening for Changes
- Reactive Forms in Angular: Dynamically Creating Form Fields with FormArray
- Reactive Forms in Angular: Creating a Custom Validator

## ROUTING

- Common Routing Tasks
- A Complete Guide to Routing In Angular
- Angular Authentication: Using Route Guards
- Protecting Routes Using Guards in Angular
- Using Route Guards

# JAVA FSD - REFERENCES and CHEATSHEETS

## SPRING FRAMEWORK

***SPRING CORE CHEATSHEET***

- Dependency Injection
- Spring Java based configuration
- Annotation-based Container Configuration
- Integrate Spring with Hibernate Using Class Configuration

## SPRING *BOOT*

***BOOT CHEATSHEET***

- Building an Application with Spring Boot
- Building REST services with Spring
- Building a RESTful Web Service
- Spring Data JPA | Query Methods | Accessing data with MySQL | Accessing Data with JPA
- MongoDB CRUD Operations | Accessing Data with MongoDB
- Managing Transactions
- Securing Spring Boot Microservices Using JWT Token
- Consuming a RESTful Web Service
- Testing the Web Layer
- Spring Security and Angular
- Metrics and Tracing with Spring
- Boot Swagger

## MICROSERVICES

- Introduction to Microservices
- Building a Gateway
- Centralized Configuration
- Service Registration and Discovery

## CONTINUOUS INTEGRATION

- Continuous Integration
- 7-reasons-why-you-should-be-using-ci
- Continuous Integration

## DOCKER

***DOCKER CHEATSHEET***

- Docker Overview
- Containerization Samples