



Java FSE with Angular - Cognizant

	MON	TUE	WED	THU	FRI	ENVIRONMENT
Week 1 Agile, Git, HTML, CSS	Orientation & Onboarding <ul style="list-style-type: none"> Introduction to Linux OS basic Linux commands shell scripting fundamentals Fullstack the Big Picture Git - Source Control Management <ul style="list-style-type: none"> VCS, CVCS, DVCS Git Fundamentals Git Installation and Configuration Basic Git Commands Git Branching and Merging Remote Repositories Pull Requests .gitignore 	HTTP <ul style="list-style-type: none"> Introduction to HTTP HTTP Methods HTTP Lifecycle HTTP Status Codes HTTP Headers HTTP Cookies HTML Foundation <ul style="list-style-type: none"> Overview of HTML HTML5 Features DOM (Document Object Model) HTML Tags Elements and Attributes Inline vs Block Elements HTML Forms <ul style="list-style-type: none"> Form Element , Form Attributes (action, method) , Input Elements , Input Types Label Element , Textarea Select and Option , Multi-Select , Fieldset and Legend HTML5 Form Validation GET vs POST 	HTML 5 Advanced <ul style="list-style-type: none"> Web storage localStorage, sessionStorage WebSQL Database Geolocation Introduction to CSS3 <ul style="list-style-type: none"> CSS Syntax Ways to apply CSS CSS Selectors Styling Text and Fonts Font Properties Box Model Basics Content, Padding, Border, Margin Colors and Backgrounds Borders and Effects Layout with CSS Display, Position, Float 	<ul style="list-style-type: none"> Flexbox Layout flex-direction, justify-content CSS Grid Layout <ul style="list-style-type: none"> grid-template-columns/rows Responsive Web Design <ul style="list-style-type: none"> Media Queries Breakpoints Advanced Features <ul style="list-style-type: none"> Transitions, Animations 	Agile Methodology <ul style="list-style-type: none"> Introduction to Agile Agile Process Scrum Framework Scrum Roles, Artifacts, Ceremonies Agile User Stories Agile Estimation and Planning Sprint Points, Planning Poker, Velocity Burndown and Burnup Charts Project Boards Agile Tools 	
Week 2 JavaScript, Bootstrap, jQuery	Introduction to JavaScript <ul style="list-style-type: none"> Getting Started JavaScript Fundamentals Framework and libraries Variables (var , let, const) 	JavaScript Arrays <ul style="list-style-type: none"> Array Methods forEach, map, filter, reduce JavaScript Function <ul style="list-style-type: none"> Function Declaration Arrow Functions Callbacks 	Asynchronous JavaScript <ul style="list-style-type: none"> Synchronous vs Asynchronous Call Stack Event loop Callback Queue JSON <ul style="list-style-type: none"> JSON.parse() 	Bootstrap 5 <ul style="list-style-type: none"> Introduction to Bootstrap 5 Bootstrap Setup (CDN) Layout and Grid System Container, Row, Column 	<ul style="list-style-type: none"> Bootstrap Utilities and Helpers Responsive Design with Bootstrap Customization and Extensions jQuery <ul style="list-style-type: none"> What is jQuery ? jQuery Selectors 	

	MON	TUE	WED	THU	FRI	ENVIRONMENT
	<ul style="list-style-type: none"> Data Types JavaScript Operators Control Flow if - else, switch, Loops JavaScript Objects Object Creation, Properties, Methods this Keyword 	<ul style="list-style-type: none"> Closures IIFE DOM (Document Object Model) Selecting Elements Modifying Content DOM Manipulation Traversing the DOM Events and Event Listeners 	<ul style="list-style-type: none"> JSON.stringify() Callbacks Callback Functions Callback hell Promises, async/await Fetch API ES6 features Debugging and Testing JavaScript 	<ul style="list-style-type: none"> Responsive Breakpoints Essential Components Typography, Buttons, Cards, Forms Navbar, Modals, Dropdowns 	<ul style="list-style-type: none"> DOM manipulation & events jQuery Effects Basic AJAX with jQuery 	
Week 3 Bootstrap, jQuery, MySQL, Core Java(v21)	Project 1 MySQL <ul style="list-style-type: none"> What Is A Database What Is SQL Consistency Introduction To RDBMS Schema Table Structure SQL Data Types Normalization Multiplicity Data Modeling And ERD Primary Key Composite Key Foreign Key Unique Key Secondary Alternate Key Referential Integrity 	Project 1 <ul style="list-style-type: none"> Overview Of Sublanguages DDL DML DQL DCL TCL Defining Schema CREATE DROP TRUNCATE Constraints Auto Incrementing CHECK DEFAULT CASCADE INSERT UPDATE DELETE Queries Aggregate Functions Clauses What Is A Subquery What Is A Join Cross Join Equi And Theta Joins Inner Join Left And Right 	Project 1 <ul style="list-style-type: none"> What Is A Transaction ACID Properties Transaction Properties CRUD Operations Indexes Scalar Functions Sequence Trigger Views What Is A Stored Procedure What Is A User Defined Function Transaction Commit Rollback Isolation Levels 	Project 1 <ul style="list-style-type: none"> Java Basics What Is Java Jvm Jre Jdk Setup Jdk Set Up Ide IntelliJ Primitive Data Types Reference Variables Control Flow Conditional Statements Logical Operators Mathematical Operators Comparison Operators Arrays Control Flow Loops Commenting Packages And Imports Debugging 	Project 1 <ul style="list-style-type: none"> Methods Method Parameters And Return Types Method Invocation Method Visibility Modifiers Method Scope Introduction To The Stack Method Recursion Method Declaration And Syntax Types Casting String Basics Value And Reference Types Wrapper Classes 	

	MON	TUE	WED	THU	FRI	ENVIRONMENT
		Joins <ul style="list-style-type: none"> Outer Join Aliases 				
Week 4 Core Java (v21)	Project 1 OOPs <ul style="list-style-type: none"> Introduction To Oop Classes Vs Objects Classes Members Static Members Oop Inheritance Interfaces And Abstract Class Oop Polymorphism Overloading Overriding Oop Encapsulation Oop Abstraction Object Class Non Access Modifiers Equality Hashcode Equals Garbage Collection 	Project 1 Exceptions <ul style="list-style-type: none"> Exceptions Vs Errors And Hierarchy Handling Exceptions Checked Vs Unchecked Exceptions Creating Custom Exceptions Reading The Stack Trace Collections <ul style="list-style-type: none"> Overview Of Collections Hierarchy List Interface & Implementation Classes Set Interface & Implementation Classes Queue Interface & Implementation Classes Iterators Map Interface & Implementation Classes 	Project 1 Multithreading <ul style="list-style-type: none"> thread-class runnable-interface states-of-a-thread multithreading Concurrency <ul style="list-style-type: none"> synchronization deadlock livelock producer-consumer-problem Functional Programming <ul style="list-style-type: none"> functional-interfaces lambdas method-reference-syntax Optional-class File Handling and I/O <ul style="list-style-type: none"> reading-and-writing-to-files 	Project 1 Stream API <ul style="list-style-type: none"> stream-api Date and Time API reflection-api Intermediate Operations Terminal Operations Collectors Parallel Streams Design Patterns <ul style="list-style-type: none"> SOLID Principles Singleton Pattern Factory Pattern Behavioral Pattern Structural Pattern 	Project 1 Java 17 & 21 Features <ul style="list-style-type: none"> Sealed Classes, Pattern Matching Text Blocks, Records, Virtual Threads JDBC <ul style="list-style-type: none"> Jdbc Architecture And Interfaces Driver Types And Registration Setting Up The Database Driver Setting Up The Utility Class Simple And Prepared Statements Sql Injection Callable Statements Result Set Navigating Result Set Rows 	
Week 5 Core Java (v21)	Project 1 Introduction to Reactive Programming <ul style="list-style-type: none"> Reactive Manifesto, Reactive Streams Reactive Programming in Java Building Reactive Applications Networking and Sockets <ul style="list-style-type: none"> Java Networking Socket Programming 	Project 1 Application Debugging Using IntelliJ IDEA <ul style="list-style-type: none"> Introduction to Debugging Getting Started with IntelliJ IDEA Debugger Basic Debugging Techniques : Setting Breakpoints , Step Over, Step Into, Step Out Advanced Debugging Techniques : Watch 	Project 1 Reverse Engineering <ul style="list-style-type: none"> Goals and its Objectives Steps of Reverse Engineering Tools of Reverse Engineering 	Project 1 TDD using JUnit and Mockito <ul style="list-style-type: none"> Test Driven Development Overview Unit Testing Introduction To Junit Unit Testing Line Vs Branch Coverage Unit Testing Arrange Act Assert Unit Testing Assertion Types Unit Testing Stubs Introduction to	Project 1 Introduction to Logging <ul style="list-style-type: none"> Log4J Framework Log4J Configuration Appenders (Console, File, Rolling File) Layouts (Pattern Layout) Log4J 2 Architecture SLF4J (Simple Logging Facade for Java)	

	MON	TUE	WED	THU	FRI	ENVIRONMENT
	<ul style="list-style-type: none"> java.net.ServerSocket Implementing Socket Client in Java 	Variables, Debug Console		Mockito <ul style="list-style-type: none"> Introduction to Mockito Mock vs Stub vs Spy 	<ul style="list-style-type: none"> SLF4J API , Binding with Log4J Logger Factory Best Practices for Logging 	
Week 6 Spring Framework Foundations	Project 1 Introduction to Spring Framework <ul style="list-style-type: none"> Spring Framework Overview Spring Ecosystem Spring vs Java EE Setting up a Spring Project with Maven <ul style="list-style-type: none"> Introduction to Maven Maven Project Structure Project Object Model (POM) , Maven Central Repository, Dependencies Management Maven Build Lifecycle, Maven Commands Spring Dependencies Spring IoC Container <ul style="list-style-type: none"> Inversion of Control Concept IoC Container Types ApplicationContext BeanFactory Spring Bean Configuration <ul style="list-style-type: none"> XML Configuration ,Annotation-based Configuration , Java-based Configuration Bean Scopes , Bean Lifecycle Interceptors 	Project 1 Dependency Injection in Spring <ul style="list-style-type: none"> Constructor Injection, Setter Injection, Field Injection Autowiring <ul style="list-style-type: none"> @Autowired Annotation Qualifier Spring AOP (Aspect-Oriented Programming) <ul style="list-style-type: none"> AOP Concepts Cross-cutting Concerns Aspect, Advice, Pointcut, Join Point @Aspect Annotation Before, After, Around Advice AOP Proxies 	Project 1 Spring MVC and ORM <ul style="list-style-type: none"> Spring MVC Architecture DispatcherServlet Controllers ViewResolver Model and View Spring ORM Hibernate Integration Spring Boot (Basics) <ul style="list-style-type: none"> What is Spring Boot? Spring Boot vs Spring Framework Auto-configuration Starter Dependencies Spring Boot CLI application.properties / yml 	Project 1 Reactive Programming with Spring WebFlux <ul style="list-style-type: none"> Reactive Programming Concepts Project Reactor Mono and Flux WebFlux vs Spring MVC Reactive Controllers Reactive Data Access Testing Spring Applications with JUnit and Mockito <ul style="list-style-type: none"> @SpringBootTest , @MockBean , Testing Controllers, Services 	Mocking External Dependencies <ul style="list-style-type: none"> HTTP Client Mocking , Database Mocking Project Presentation	
Week 7 Spring Data JPA &	Project 2	Project 2	Project 2	Project 2	Project 2	

	MON	TUE	WED	THU	FRI	ENVIRONMENT
Spring REST	<p>Introduction to Spring Data JPA</p> <ul style="list-style-type: none"> JPA Overview Spring Data JPA Benefits Repository Pattern Setting Up a Spring Boot Dependencies Configuration DataSource Configuration JPA Properties <p>Entity Mapping</p> <ul style="list-style-type: none"> @Entity Annotation , @Table, @Column Primary Keys, Relationships (@OneToMany, @ManyToOne, etc.) , Embedded Objects 	<p>Spring Data Repositories</p> <ul style="list-style-type: none"> CrudRepository JpaRepository PagingAndSortingRepository Custom Repositories <p>CRUD Operations with Spring Data JPA</p> <ul style="list-style-type: none"> Save and Update Find Methods Delete Operations 	<p>Query Methods and Named Queries</p> <ul style="list-style-type: none"> Query Derivation @Query Annotation Named Queries Native Queries <p>Pagination and Sorting</p> <ul style="list-style-type: none"> Pageable Interface Sort Object Page vs Slice <p>Auditing with Spring Data JPA</p> <ul style="list-style-type: none"> @CreatedDate, @LastModifiedDate , @CreatedBy, @LastModifiedBy Auditing Configuration 	<p>Spring Data JPA Projections</p> <ul style="list-style-type: none"> Interface-based Projections Class-based Projections Dynamic Projections <p>Spring Data JPA and Spring Boot Integration</p> <ul style="list-style-type: none"> Auto-configuration Properties Configuration Multiple DataSources <p>Spring Data JPA and Hibernate</p> <ul style="list-style-type: none"> Hibernate as JPA Provider Entity Manager Session Factory Caching (First Level, Second Level) 	<p>Code Quality</p> <ul style="list-style-type: none"> Code Quality Metrics Technical Debt Code Smells SonarQube Dashboard <p>Introduction to Spring REST and Spring Boot 3</p> <ul style="list-style-type: none"> REST Principles HTTP Methods Status Codes Spring Boot 3 Features <p>Building a Simple REST Controller</p> <p>Request and Response handling</p> <ul style="list-style-type: none"> @RestController , @RequestMapping , @GetMapping, @PostMapping @RequestBody , @ResponseBody , @PathVariable , @RequestParam , @ResponseEntity 	
Week 8 REST APIs & Microservices	<p>Project 2</p> <p>RESTful Resource Representation with DTOs</p> <ul style="list-style-type: none"> Data Transfer Objects DTO Pattern ModelMapper, MapStruct <p>RESTful CRUD Operations</p> <ul style="list-style-type: none"> Create Resources Read Resources Update Resources Delete Resources <p>RESTful HATEOAS</p> <ul style="list-style-type: none"> Hypermedia Concept , Spring HATEOAS , Links and Resources 	<p>Project 2</p> <p>Spring Boot Actuator for REST Monitoring</p> <ul style="list-style-type: none"> Actuator Endpoints , Health Checks , Metrics , Custom Endpoints Security and <p>Authentication in RESTful APIs</p> <ul style="list-style-type: none"> Spring Security Basics JWT Authentication OAuth2 <p>Testing RESTful APIs</p> <ul style="list-style-type: none"> MockMvc , @WebMvcTest <p>Documenting RESTful APIs</p> <ul style="list-style-type: none"> 	<p>Project 2</p> <p>Introduction to Microservices Architecture (MSA)</p> <ul style="list-style-type: none"> Monolith vs Microservices Microservices Characteristics Benefits and Challenges Domain-Driven Design <p>12 factor APP Spring Cloud for Microservices</p> <ul style="list-style-type: none"> Spring Cloud Overview Service Discovery (Eureka) Load Balancing (Ribbon/Spring 	<p>Project 2</p> <p>Session Hijacking Authentication and Authorization</p> <ul style="list-style-type: none"> OAuth2 Authorization Server Resource Server SSO (Single Sign-On) <p>Microservices Communication with Spring Cloud</p> <ul style="list-style-type: none"> REST Template WebClient Feign Client Asynchronous Communication <p>API Gateway and Edge Services</p> <ul style="list-style-type: none"> Spring Cloud 	<p>Project 2</p> <p>Fault tolerance and Resilience</p> <ul style="list-style-type: none"> Retry Mechanisms Fallback Methods Bulkhead Pattern <p>Spring Cloud Config</p> <ul style="list-style-type: none"> Centralized Configuration Config Server Config Client Refresh Scope <p>Monitoring and Metrics in Microservices</p> <ul style="list-style-type: none"> Distributed Tracing (Sleuth, Zipkin) Logging Aggregation 	

	MON	TUE	WED	THU	FRI	ENVIRONMENT
	Content Negotiation and Media Types <ul style="list-style-type: none"> Accept Header , Content-Type Header ,JSON vs XML 	OpenAPI/Swagger <ul style="list-style-type: none"> SpringDoc API Documentation Best Practices 	Cloud <ul style="list-style-type: none"> Circuit Breaker (Resilience4j) Spring Security for Microservices <ul style="list-style-type: none"> Microservices Security Challenges Service-to-Service Authentication CORS Token-based Security Centralized CSRF DDOS 	Gateway <ul style="list-style-type: none"> Routing Filters Rate Limiting 	<ul style="list-style-type: none"> Metrics Collection Security Best Practices in Microservices <ul style="list-style-type: none"> Defense in Depth API Security Secrets Management 	
Week 9 TypeScript, Angular(v16)	Project 2 TypeScript <ul style="list-style-type: none"> What Is Typescript Javascript Vs Typescript How Why Should I Use Typescript Typescript Compiler And Install Typescript Package Json Node As Js Runtime Simple Or Variable Types Special Types Object Types Union Types Type Aliases And Interfaces Casting Functions Classes Classes And Access Modifiers Decorators 	Project 2 <ul style="list-style-type: none"> Utility Types Array Generics Basic Generics Keyof Readonly Interface As Const Type Guards Compiler Options Strict Target Tsc Ts Config Basics 	Project 2 Angular <ul style="list-style-type: none"> Introduction to Angular Angular Setup and Installation Node.js and npm Angular CLI Creating Angular Application Angular Project Structure Single Page Applications (SPA) Client-side Routing Webpack Angular Templates 	Project 2 Angular Components <ul style="list-style-type: none"> Component Lifecycle LifeCycle hooks @Component Component Styles Change Detection Dynamic Components Event Emitters Sharing Data Between Child And Parent Angular Modules <ul style="list-style-type: none"> NgModule Decorator Root Module Feature Modules Shared Modules 	Project 2 Data Binding <ul style="list-style-type: none"> 1 Way Data Binding 2 Way Data Binding Angular Directives <ul style="list-style-type: none"> Structural Directives Attribute Directives Pipes <ul style="list-style-type: none"> Pipes Custom Pipes 	
Week 10 Angular (v16)	Project 2 Dependency Injection <ul style="list-style-type: none"> DI in Angular Injector Hierarchy DI Providers 	Project 2 RxJS <ul style="list-style-type: none"> Introduction to RxJS Observables 	Project 2 Angular Forms <ul style="list-style-type: none"> Introduction to Forms Template - Drive Forms 	Project 2 Angular Testing <ul style="list-style-type: none"> Karma Jasmine Testing In Angular 	Project 2 <ul style="list-style-type: none"> Review 	

	MON	TUE	WED	THU	FRI	ENVIRONMENT
	Services <ul style="list-style-type: none"> Creating Services Service Injection Service Communication Angular Routing <ul style="list-style-type: none"> Router Module Route Guards Routing And Navigation Signals	<ul style="list-style-type: none"> RxJS Operators RxJS Subjects HTTP Client <ul style="list-style-type: none"> Making HTTP Requests Error Handling HTTP Interceptors Angular HTTP Best Practices 	<ul style="list-style-type: none"> Reactive Forms Form Validation Dynamic forms State Management <ul style="list-style-type: none"> Application State State Management Patters Service-based State BehaviorSubject pattern 	With Jasmine And Karma Application Debugging Front End <ul style="list-style-type: none"> Debugging Angular Applications with Chrome DevTools Debugging with Visual Studio Code 		
Week 11 Docker , Kubernetes,Cloud (GCP)	Project 2 Docker Architecture <ul style="list-style-type: none"> Docker Architecture Containers Vs Vms Containerization Docker Image &Container <ul style="list-style-type: none"> Installing Docker Dockerfile Docker Images Docker Containers Basic Docker Cli Commands Docker Registry and Repository <ul style="list-style-type: none"> DockerHub 	Project 2 DevOps <ul style="list-style-type: none"> DevOps Overview Continuous deployment Continuous integration Continuous delivery Kubernetes <ul style="list-style-type: none"> What is Container Orchestration? Kubernetes Architecture Master vs Worker Nodes Control Plane Components Core Kubernetes Concepts <ul style="list-style-type: none"> Pods , ReplicaSets, Deployments, Services (ClusterIP, NodePort, LoadBalancer), Namespaces 	Project 2 Kubernetes Configuration <ul style="list-style-type: none"> ConfigMaps, Secrets, Environment Variables Storage in Kubernetes <ul style="list-style-type: none"> Volumes Persistent Volumes Persistent Volume Claims Kubernetes Networking <ul style="list-style-type: none"> Service Discovery Ingress Controllers Network Policies kubectll CLI Commands <ul style="list-style-type: none"> Basic Operations 	Project 2 Introduction to Cloud Computing <ul style="list-style-type: none"> Cloud Computing Definition On-Premise vs Cloud , Cloud Benefits Cloud Service Models IaaS (Infrastructure as a Service) PaaS (Platform as a Service) SaaS (Software as a Service) Cloud Deployment Models <ul style="list-style-type: none"> Private, Private, Hybrid, Multi-Cloud Cloud Service Providers <ul style="list-style-type: none"> AWS, Azure, GCP 	Introduction to GCP <ul style="list-style-type: none"> GCP Console Projects and Billing IAM Basics GCP Identity and Access Management (IAM) <ul style="list-style-type: none"> Users, Groups, Service Accounts Roles and Permissions Policy Management GCP Compute Services <ul style="list-style-type: none"> Compute Engine (VMs) App Engine Kubernetes Engine (GKE) Cloud Functions GCP Storage Services <ul style="list-style-type: none"> Cloud Storage (Buckets) Persistent Disks Filestore Project Presentation	
Week 12 Cloud(GCP), Project	Project 3 GCP Networking <ul style="list-style-type: none"> VPC Load Balancing Cloud CDN 	Project 3 Introduction to DevOps in GCP <ul style="list-style-type: none"> DevOps Culture and Practices CI/CD Pipeline 	Project 3 GCP Cloud Deploy <ul style="list-style-type: none"> Deployment Pipelines Delivery Pipelines Rollout 	Project 3	Project 3	

	MON	TUE	WED	THU	FRI	ENVIRONMENT
	<ul style="list-style-type: none"> Cloud DNS <p>GCP Database Services</p> <ul style="list-style-type: none"> Cloud SQL Cloud Spanner Firestore Bigtable <p>GCP Serverless Computing</p> <ul style="list-style-type: none"> Cloud Functions Cloud Run Serverless Containers 	<ul style="list-style-type: none"> Infrastructure as Code <p>GCP Cloud Source Repositories</p> <ul style="list-style-type: none"> Git Repositories in GCP Repository Management Integration with Cloud Build <p>GCP Cloud Build</p> <ul style="list-style-type: none"> Build Configuration Build Triggers Automated Builds Container Building 	<p>Management</p> <ul style="list-style-type: none"> Automated Deployment <p>GCP Cloud Build Pipelines</p> <ul style="list-style-type: none"> Pipeline as Code Multi-Stage Pipelines Testing in Pipelines Deployment Automation 			
Week 13 Project , Review	Project 3	Project 3	Project 3	Project 3	Project Presentation	

PROJECT	TECHNOLOGIES
Project 1	Java, MySQL, JDBC
Project 2	Java, MySQL, Spring Boot, Spring JPA, Angular
Project 3	Java, MySQL, Spring Boot, Spring Data JPA, Docker, Kubernetes, GCP



Copyright © 2025 Revature, LLC. All Rights Reserved.

By viewing this document, you agree that under copyright law all content displayed is the sole intellectual property of Revature, LLC, a technology advancement and consulting company based in Reston, VA. All content generated by a representative of Revature which is used for the company's advancement, development, or have otherwise been developed at the company's request, are the sole property of the company. No intellectual property may be reproduced, distributed, altered, or shared without the explicit permission from a representative of Revature.