**Java**

**Java Language, OOPS, Programming**

* Introduction to Java and OOPS
* Java Tokens- Comments, Identifiers, Keywords, Separators
* Working with Java Editor Softwares – Editplus, NetBeans, Eclipse
* Packages with static imports
* Working with jar
* Modifiers – File level, Access level and Non-access level
* Datatypes, Literals, Variables, Type Conversion, Casting & Promotion
* Reading runtime values from keyboard and Properties File
* Operators and Control Statements
* Method and Types of methods
* Variable and Types of Variables
* Constructor and Types of constructors
* Block and Types of Blocks
* Declarations, Invocations and Executions
* Compiler & JVM Architecture with Reflection API
* Static Members and their execution control flow
* Non-Static Members and their execution control flow
* Final Variables and their rules
* Classes and Types of classes
* OOPS- Fundamentals, Models, Relations and Principles
* Coupling and Cohesion (MVC and LCRP Architectures)
* Types of objects & Garbage Collection
* Arrays and Var-arg types
* Enum and Annotation
* Design Patterns

**Java API and Project**

* API and API Documentation
* Fundamental Classes – Object, Class, System, Runtime
* String Handling
* Exception Handling and Assertions
* Multithreading with JVM Architecture
* IO Streams (File IO)
* Networking (Socket Programming)
* Wrapper Classes with Auto boxing and unboxing
* Collections with Generics
* Java 5, 6, 7, 8 new features
* Inner classes
* AWT, Swings, Applet
* Regular Expressions
* Formatting date, time (java.text package)

**Spring Boot**

**Introduction to spring boot**

* Types of software architectures
* SOA and Monolith Architecture
* Why Microservices
* Detailed MicroService Architecture  
  – App Layer  
  – Business Layer
* Enteprise Layer  
  – Infra Layer
* Need of Spring Boot
* Difference between Spring & Spring Boot
* Advantages with Micro Services

**Building Spring Boot Application**

* Normal Spring Manual Approach
* Maven Overview
* Spring Initializer
* STS
* Eclipse with STS Plugin
* Understanding the Spring Boot auto configuration

**Rest Annotation with In Memory Database & CRUD Operations**

* H2
* Derby
* HSQL
* Redis Cache
* PostMan or Swagger Overview

**Rest Annotation with Relation DB**

* MySql
* PostGresSQL

**JPA Repository Concepts**

* Crud Repository
* JPA Query Concepts
* NamedQueries
* QueryAnnotation
* AsyncResults
* Pagination and Sorting

**Actuator Concepts**

* Production Monitoring
* Health Check Concepts
* Security Measurements

**Spring Boot Custom Logging**

* Logging Level
* Patterns Changes
* Rolling Logs

**Spring Boot Profile Components**

* Introduction
* Multiple Properties
* YML File
* Command Line Runner Example
* Real time scenarios of components

**Auto Configuration**

* Introduction
* @Conditional Flow
* Customize conditional annotations
* Spring Boot built in conditional annotations

**Thymleaf Concepts**

* Introduction
* Example on Web Application
* Validatins on Web Applications
* Internalization i18n Concepts

**Integration with Spring Web**

* Using **Spring Web MVC**
* Using **Spring Restful**
* Need of embedded servers & customization

**Spring Boot Security**

* Basics
* Basic Authentication
* Form Based Authentication
* Authorization
* Role Based Access Control
* Attribute Based Access Control
* LDAP Based
* SSL Security
* TLS Security

**Database Concepts**

* Spring JDBC
* Database to CSV
* Spring Batch
* Flyway Database Migration
* Liquid Database Migration
* Flyway vs Liquid
* Hikari Connection Pool

**Core Concepts**

* Spring Boot AOP
* Spring Boot Cache
* Guava Cache integration
* Caffenine Cache
* EH Cache
* MultiResourceItemReader
* Spring MVC vs JAX-RS
* SprinBoot with Jersey
* Junit Integration
* Rest Integration Test Cases

**Micro Services**

* Micro Services Introduction
* Principle and Characteristics
* Use cases and Benefits
* Challenges
* Design standards
* Micro Services Communication  
  – Synchronous  
  – Asynchronous
* Pitfalls

**Micro Services Design Considerations**

* Micro Services per JVM?
* Micro Services share the data stores?
* Micro Services Transaction boundaries
* User Interfaces integration with Micro Services
* Challenges in Micro Services implementation

**Spring Cloud**

* Introduction
* Cloud Architecture
* Cloud application benefits

**Spring Cloud Config**

* Introduction
* Setup version control repository
* Integration with repository

**Netflix**

* Introduction
* Eureka Server & Eureka Client
* Feign Client
* Ribbon

**Fault Tolerance Concepts**

* Circuit Breaker Pattern
* Hystrics Concepts, Hystrix Dashboard

**API Gateway**

* Introduction to ZUUL
* Design standards
* Integration

**Messaging Queue Concepts (CloudBus)**

* Apache KAFKA
* RabbitMQ
* JMS

**Oatuh2 Concepts**

* Client Types
* Protocol End Points
* Grant Types
* Implantation with Token Based
* JWT Tokens

**Swagger API**

* Introduction
* Integration

**Cloud Hosting**

* Pivotal Cloud Foundry Account Setup
* Hosting to Pivotal
* AWS Account Setup
* Hosting to AWS
* Enabling cloud features like load balancing, security