Contents

JEE WITH MULESOFT LOT COURSE STRUCTURE

JEE LOT provides exposure to the Java technologies, Cloud-AWS, MuleSoft . The following table lists the course structure for JEE Cloud with Mule soft.

Sr. No.	Course	Duration (In Days)	Remarks
1	Discover	1	
2	Soft Skills Foundation – Part 1	1	Soft Skills Part 1
3	OOPS & SQL(Oracle) Core Java 8 & Development Tools+Build Tool-Maven+DevOps/CI CD concepts (Github /Nexus,CI with Jenkins, Sonar)	11	Project kick off
4	Core Java Test	0.5	Coding and MCQ
5	Soft Skills Foundation – Part 2	1	Soft Skills Part 2 (Saturday)
6	JPA with Hibernate 3.0 (Basics)	1.5	
7	Spring 4.0 with Spring Boot and Spring with REST	7	
8	Micro Services Advance using Spring Boot and Rest Template	2	Sprint 1 Implementation
9	Soft Skills Foundation – Part 3	1	Soft Skills Part 3
10	Cloud Basics & AWS Basics of different services	2	Sprint 1 Implementation +MCQ
11	Containers – Introduction to Docker	1	
12	Sprint 1 Evaluation	1	Sprint 1 Evaluation
13	Soft Skills Foundation – Part 4	1	Soft Skills Part 4 (Evaluation)
14	Mulesoft	8	
15	Sprint 2+MCQ	7	Sprint 2 Implementation +MCQ
16	Sprint 2 Evaluation	1	Sprint 2 Evaluation, MCQ
17	L1 Test	1	
Total Training Duration		**48**	

Oracle Basics

Program Duration: 2 days

Contents:

- Introduction to Database

    - Getting Started with Database
    - Characteristics of DBMS
    - Data models
    - Relational DBMS
    - Database Administrator
- Basics of SQL
    - The SQL Language
    - Rules for SQL Statements
    - Standard SQL Statement Groups
- Data Query Language
    - The SELECT statement
    - The WHERE clause
    - Comparison, Mathematical, and Logical operators
    - The DISTINCT clause
    - The ORDER BY clause
    - Tips and Tricks in SELECT Statements

- Aggregate (Group) Functions
    - The Group function
    - GROUP BY & HAVING clause
    - Examples of GROUP BY and HAVING clause
    - Tips and Tricks
- SQL (Single-row) functions
    - SQL functions
    - Number functions
    - Character functions
    - Date functions
    - Conversion functions
    - Miscellaneous functions
    - Tips and Tricks
- Joins and Sub-queries
    - Joins
        - Oracle Proprietary Joins
    - Types of Joins
    - Sub-query
- Database Objects
    - Basic Data Types
    - Data Integrity
    - Examples of CREATE TABLE
    - Examples of ALTER TABLE
    - Database Objects(Index, and View)

- Data Manipulation Language
    - Adding Data
    - Removing Data
    - Modifying Data
- Transaction Control Language
    - Introduction to Transactions
    - Transaction Control Statements

OOP and UML

Program Duration: 1 day.

Contents:

- Principles in Object-Oriented technology
- UML diagram
    - Use Case Diagram
    - Class Diagram
    - Sequence Diagram

Core Java 8 and Developer Tools

Program Duration: 9 days

Contents:

- Introduction to Java
    - Introduction to Java
    - Features of Java
    - Evolution in Java
    - Developing software in Java
- Eclipse 4.4 (Luna) as an IDE
    - Installation and Setting up Eclipse
    - Introduction to Eclipse IDE
    - Creating and Managing Java Projects
    - Use of Java docs
    - Miscellaneous Options
- Language Fundamentals
    - Keywords
    - Primitive Data Types
    - Operators and Assignments
    - Variables and Literals
    - Flow Control: Java's Control Statements
    - Best Practices
- Classes and Objects

    - Classes and Objects
    - Packages
    - Access Specifiers
    - Constructors - Default and Parameterized
    - this reference
    - using static keyword
    - Best Practices
- Exploring Java Basics
    - The Object Class
    - Wrapper Classes
    - Type casting
    - Using Scanner Class
    - String Handling
    - Date and Time API
    - Best Practices
- Inheritance and Polymorphism
    - Inheritance
    - Using super keyword
    - InstanceOf Operator
    - Method & Constructor overloading
    - Method overriding
    - @override annotation
    - Using final keyword
    - Best Practices
- Abstract Classes and Interfaces
    - Abstract class
    - Interfaces
    - default methods
    - static methods on Interface
    - Runtime Polymorphism
    - Best Practices
- Regular Expressions
    - Regular Expressions
    - Validating data
    - Best Practices
- Exception Handling
    - Introduction
    - Exception Types
    - Exception Hierarchy
    - Try-catch-finally
    - Try-with-resources

    - Multi catch blocks
    - Throwing exceptions using throw
    - Declaring exceptions using throws
    - User defined Exceptions
    - Best Practices
- Array
    - One dimensional array
    - Multidimensional array
    - Using varargs
    - Using Arrays class
    - Best Practices
- Collection
    - Collections Framework
    - Collection Interfaces
    - Implementing Classes
    - Iterating Collections (using foreach & iterator)
    - Comparable and Comparator
    - Best Practices
- Generics
    - Generics
    - Writing Generic Classes
    - Using Generics with Collections
    - Best Practices
- File IO
    - Overview of I/O Streams
    - Types of Streams
    - The Byte-stream I/O hierarchy
    - Character Stream Hierarchy
    - Buffered Stream
    - The File class
    - The Path class
    - Object Stream
    - Best Practices
- Property Files
    - What are Property Files?
    - Types of Property files
    - User defined Properties

Build Tool Maven

Contents:

- Maven

- Maven Overview
- Benefits of Maven
- Maven Basics
- Working with Maven
- Installing Maven
- Creating simple project using Maven Commands
- Setting up Maven in Eclipse
- Creating Web application using Maven

- Introduction to Junit 4
  - Introduction to Junit 4
  - Why testing
  - Why use Junit
  - Installing and Running Junit
  - Understanding Junit Framework
  - Testing with JUnit
- Java Database Connectivity
  - Java Database Connectivity - Introduction
  - Database Connectivity Architecture
  - JDBC APIs
  - Database Access Steps
  - Calling database procedures
  - Using Transaction
  - Connection Pooling
  - DAO Design Pattern
  - Best Practices
- Introduction to Layered Architecture
- Logging with Log4J
  - Log4J Concepts
  - Installation of Log4J
  - Configuring Log4J
  - Best Practices
- MultiThreading
  - Understanding threads
  - Thread life cycle and Scheduling threads- Priorities , sleep(),join()
  - Consumer Producer problem
  - Inter Thread communication : wait, notify, notifyAll methods
  - Synchronization concept
- Lambda expressions
  - Understand the concept of Lambda expressions

    - Working with lambda expressions
    - Use method references and functional interfaces
- Stream API
    - Understand the concept of Stream API
    - Use stream API with collections
    - Perform different stream operations

Introduction to DevOps

- Introduction of DevOps
- Dev And Ops
- Agile Vs DevOps
- Continuous Integration & Delivery pipeline
- Tools For DevOps
- Use-case walkthrough

GIT Hub:

- Working locally with GIT
- Working remotely with GIT
- Branching, merging & rebasing with GIT
- Use Case walkthrough

Jenkins:

- Introduction to Jenkins
- Jenkins Objective
- Introduction to continuous integration deployment & Jenkins-ci
- Continuous Deployment & distribution builds with Jenkins

Sonar

- Introduction to Sonar
- Code quality Monitoring- Sonar
- Use Case walkthrough

JPA With Hibernate 3.0

Program Duration: 1.5 day

Contents:

- Introduction to ORM and its need

- The Persistence Life Cycle
- Java persistence API (JPA)
- JPQL

Spring 4.0 with Spring Boot and Spring with REST

Program Duration: 7 days

Contents:

- Basics of Web Technologies –Servlet & JSP
- Introduction to Spring Platform and environment
- Introduction to Spring Framework, IoC
    - What is Spring Framework, Benefits of Spring
    - The Spring architecture
    - IOC – Inversion of control, wiring beans
    - Bean containers, lifecycle of beans in containers
    - Customizing beans with BeanPostProcessors & BeanFactoryPostProcessors
    - XML and Annotation-based, mixed configurations
- Java Base Configuration
- Spring MVC framework
    - Introduction: DispatcherServlet, Handler mappings, Resolving views
    - Annotation-based controller configuration
    - Web Based Application Using Spring Boot
    - Introduction to REST web Services
    - REST Controllers on the top of MVC
    - Spring Boot Integration with Rest
- Spring JPA Integration
    - Spring support for JPA
    - Implementing Spring JPA integration
    - Spring Data
    - Spring Boot(Annotation based and Java configuration)
    - Spring ReST
    - Spring DATA ReST

MuleSoft

Program Duration: 15 days

Contents:

Introducing Anypoint Platform

- Anypoint Platform Components
- What is the role of each component in building application networks
- Anypoint Platform Navigation
- Anypoint Exchange - Locate APIs and other assets needed to build integrations and APIs
- Flow Designer - Creating basic integrations to connect systems

Designing APIs

- RAML (Restful API Modeling Language)
- Defining APIs with RAML
- Crating Mock APIs to test their design before they are built
- Make APIs discoverable by adding them to Anypoint Exchange
- Creating API portals for developers to learn how to use APIs

Building APIs

- Define Mule applications
- Define flows
- Define messages
- Define message processors
- Create flows graphically using Anypoint Studio
- Building, running, and testing Mule applications
- munit
- Connect to databases using connector
- Graphical DataWeave editor to transform data
- Create RESTful interfaces for applications from a RAML file
- Connect API interfaces to API implementations

Structuring Mule Applications

- Create reference flows and subflows
- Pass messages between flows using the Java Virtual Machine (VM) transport
- Investigate variable persistence through subflows and flows and across transport barriers
- Encapsulate global elements in separate configuration files
- Explore the files and folder structure of Mule projects

Consuming Web Services

- Consume RESTful web services with and without parameters
- Consume RESTful web services that have RAML definitions
- Consume SOAP web services
- Use DataWeave to pass parameters to SOAP web services

Handling Errors

- Different types of exception strategies
- Handle messaging exceptions in flows
- Create and use global exception handlers
- Specify a global default exception strategy

Connecting to Additional Resources

- API Development - Basic GET with REST/HTTP integration with another API (3rd party/inhouse)

- API Development - Batch development-simple split, transform and store to DB (MySQL)
- SF concepts, Basic user account creation, SF Connector overview, SF APIs - query, upsert, create
- API Development - Backend REST (3rd Party/inhouse) and SOAP (3rd Party/inhouse) service integration with aggregtion in same API
- API Development - REST to SFDC integration (CRUD)

Cloud Basics & AWS Basics of different services

Program Duration: 2 days

Contents:

- Cloud Basics
    - What is and Why Cloud?
    - Why Cloud Computing
    - Key characteristics of Cloud
    - Cloud Computing Architecture
    - Cloud Deployment and Service Model Selection criteria
    - Cloud APIs
    - Cloud benefits and Challenges
    - Different Cloud implementer
    - Latest trend

- AWS Basics of different services
    - AWS history
    - Cloud Computing and Amazon Web Services
    - Functionality offered by AWS
    - The Differences that Distinguish AWS
    - Features of AWS service
    - Different AWS web services in Cloud
    - AWS global infrastructure
- Compute services
    - Amazon EC2
    - Elastic Load balancing
- Storage Services
    - Amazon EBS
    - Amzon S3
    - Amazon Glacier
    - AWS Storage gateway
- Database services
    - Amazon RDS
    - Amazon ElastiCache

- Amazon Dynamo DB

- Administration Services
  - AWS IAM

Microservices Advanced using Spring Boot and RestTemplate

Program Duration: 2 days()

Contents:

- Microservices Basics
  - Introduction to Micro services
  - Monolithic Architecture
  - Micro service Architecture
  - Benefits of Micro services
  - Drawbacks of Micro service

  - **Rest Annotation with In Memory Database & CRUD Operations**
  -
    - **Introduction to Spring Rest Template / Asyc**
    - **How to implement client-side load balancing with Ribbon**
    - **How to implement a Naming Server (Eureka Naming Server)**
    - **How to connect the micro services with the Naming Server and Ribbon**
  -
  -
    - **Spring Boot Custom Logging**
    - **Spring Boot Profile Components**
    - **Auto Configuration**

    - **Thymleaf Concepts**
    - **Spring Boot Security**
    - **Spring Cloud**
    - **Spring Cloud Config**
    - **Oatuh2 Concepts**
    - **Actuator Concepts**
    - **Swagger**
    - **Data management**
    - **Database per Service**

- **Shared database**
- **Saga transaction**
- **CQRS - practical**

- **Code Walkthrough**
  - Introduction
  - **Extra Handson on Microservice(1.5 days)**

Containers – Introduction to Docker

Program Duration: 1 day

Contents

- Introduction to Docker
  - Limitation of VM
  - Introduction to Container
  - Container Vs VM
  - What is Docker
  - Docker Community
  - Docker Architecture
  - Docker Installation

- Docker Platform overview
  - Docker Platform
  - Docker Engine
  - Docker Images
  - Docker containers
  - Registry
  - Repositories
  - Docker Hub

- Introduction to images and Reopository naming , Automated build, Private distribution
- Docker Demo
  - Docker Example
  - Docker Case study