|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Module** | **Days** | **Duration** | **Dates** | **Remarks** |
| Capital Markets | Day 1 – 3 | 3 Days | 11th – 13th Sept |  |
| Internal | Day 4 – 5 | 2 Days | 14th – 15th Sept | For FIS Team |
| Linux | Day 6 – 8 | 3 Days | 20th – 22nd Sept |  |
| Python | Day 9 – 11 | 3 Days | 25th – 27th Sept |  |
| SQL | Day 12 – 15 | 4 Days | 28th Sep – 4th Oct |  |
| Angular | Day 16 – 19 | 4 Days | 5th – 10th Oct |  |
| Java | Day 20 – 24 | 5 Days | 11th – 17th Oct |  |
| Project-Phase 1 | Day 25 – 26 | 2 Days | 18th – 19th Oct |  |
| Spring | Day 27 - 31 | 5 Days | 20th – 30th Oct |  |
| Project-Phase 2 | Day 22 – 33 | 2 Days | 31st Oct – 2nd Nov |  |
| Microservices | Day 34 – 38 | 5 Days | 3rd – 9th Nov |  |
| Project-Phase 3 | Day 39 – 40 | 2 Days | 10th – 16th Nov |  |
| Testing | Day 41 – 45 | 5 Days | 17th – 23rd Nov |  |
| Project-Phase 4 | Day 46 – 48 | 3 Days | 24th – 28th Nov |  |
| Evaluation | Day 49 – 50 | 2 Days | 29th – 30th Nov |  |

**CAPITAL MARKETS**

**Day 1 - 3**

* Capital Market & Money Markets  
  Primary and Secondary Markets  
  Equity and Debt  
  FI Securities - Money Markets and Bonds  
  Derivatives - Options , Futures , Swap, Currency markets  
  Hedge Funds  
  Stock Market - Types of order ; Custodians - CSDs  
  Clearing House  
  FIX & Swift messages  
  General Architecture of FO / MO and BO and purposes  
  Auto & Equipment Lease / Financing  
  Commercial Lending  
  Exchange and Market Regulations  
  Market and payment cut-off's  
  Client Service  
  Process risk and escalation, governance  
  Syndicated Loans  
  **Note -** Trade Life Cycle’ for Derivatives & Equities ( The equity trade life cycle is already being covered in detail in the current 3 days 'Fresher's Bootcamp'. Extra 2-3 hours to cover some topics for derivatives.)

**LINUX**

**Day 6:**

**Introduction**

The different kinds of Linux  
Linux operating system architecture  
Multi-tasking and multi-user  
How to log in  
How to change your password  
Portable password rules  
How to log out  
Linux keyboard anomalies  
Dumb" text terminal and X-Windows terminals ♣ Online documentation and the man command

**The vi text editor**

Why learn vi?  
The different modes of vi  
Using showmode  
Miscellaneous vi commands: ., u, J, ~  
Yanking and putting: cut and paste, copy and paste ♣ Regular expression metacharacters: ^, $, ., \*, \, [, ] Searching for text using /, ?, n, and N  
Substitution , search and replace  
specifying line ranges

The set command

Configuring vi with .exrc  
splitting window-handling multiple files

Marking text and visual mode

**The file system**

The role of the file system

Various rules for naming files

Absolute and relative pathnames

Working directory and home directory

File system geography - major directories

File system commands and major options:

→ pwd print working directory

→ cd change directory  
→ ls list directory  
→ cat displaying and creating files → cp copy files  
→ mv move files  
→ rm remove files  
→ mkdir creating directory

→ rmdir removing director ♣ Viewing long files-pages → more paging output

→ less display page wise

→ file finding file type

→ cmp comparing two files

→ comm finding what is common

→ Diff converting one file to other

The . and .. and ~ abbreviations

find command Search files

locate to locate indexed files

updatedb to update file index   
→ Hard and symbolic links -----------ln create link

**File system security**

The two-level security structure of Linux

The three categories of Linux users  
Linux groups  
Read, write, and execute access

Access implications for directories  
Changing permissions with chmod  
File ownership  
Changing owner and group with chown and chgrp

The super-user

Default permissions with umask

The three timestamps of every file

"Sticky" directories

**Day 7**

**The Filter**

All the filter commands  
→Split and *csplit* splitting a one file into multiple

wc word count  
→head display beginning of the file →tail display end of the file  
→cut slitting file vertically  
→paste pasting files vertically  
→ join  
→sort sorting file contents  
→tr translating charectres

**Regular Expressions**

regular expressions and meta characters

grep searching for a pattern  
egrep Extended searching

fgrep multiple string searching

**Process related command and task scheduling**

The shell process

Parent and chills process

Init process

Running process in background

Foreground process

→ps

→ pstree

→ kill

→ jobs

→ top

→ nice

→ sleep

→ wait

→ nohup

The roles of at and cron

Security considerations

Scheduling tasks with at

Time definitions  
Using at -l and at -r

Using atq and atrm

cron and job tables  
Using crontab -l, crontab -e, crontab-r

Batch

**Day 8**

**Basic Unix File Permissions**

* + Changing permission using symbolic mode
  + Changing permission using octal mode
  + Umask utility
  + Hands on

**Configure Access control lists(ACL)**

Configure & Modify an ACL on file or dir Adding

changing and deleting ACL

Hands on

**Remote File Transfers , compress and archive files**

Usage of SCP and SSH for remote access and file transfers .

Archiving using TAR command

Compress and uncompress of files gzip and zip utility.

**PYTHON**

**Day 9:**

**Module 1 Introduction to Python, Data Types, Quotations**

Python Interpreter and its Environment

Python 3.x : Background, Relevance

Numbers

Strings

Declaration of variables

**Module 2 Conditional statements/Control Structures**

If Statements

While construct

For Statements

Break and continue Statements, and else clauses on Loops

Pass Statements

**Module 3 Python basic data structures**

Arrays, Lists and Tuples

Dictionary and Sets

List and array slicing

**Module 4 Functions**

Local variables

Default Argument Values

Returning Values

Keyword & Positional Arguments

Arbitrary Argument Lists \*

Documentation Strings

Unpacking Argument Lists ( unknown number of parameters )

Lambda Expressions

**Module 5 Functional Programming**

isalpha

map

apply

reduce

filter

**Day 10:**

**Module 6 File handling and other OS interactions**

Creating and Opening a File

Reading from a file, writing to a file (variations)

Closing a File

Handling csv files

**Module 7 Modules**

Executing modules as scripts

The Module Search Path

Running a module from the command line -m

‘Compiled’ Python files( .pyc )

The dir() Function

**Module 8 Introduction to OOP**

Class Definition Syntax

Implication of **self**

Class Objects, Instance Objects, Method Objects; Instantiation

Constructor & Deconstructor

Inheritance

Data Member – Class variable/Instance Variable

**Module 9 Exceptions**

Handling Exceptions

try-except

else clause

finally clause

Raising Exceptions

User-defined Exceptions using Raise

**Day 11:**

**Module 10 Supplementary Topics**

Pickling and unpickling (binary files)

File compression & decompression

**Module 11 Regular expressions**

What is regular expression?

Matching characters

Compiling regular expressions

Meta characters like quantifiers, anchors, character classes, alternator etc.

Strings and Slices

Modifying Strings

Use of triple quotes

**Module 12 Multi-threaded Programming**

Starting a New Thread

The Threading Module

Creating Thread Using Threading Module

**Module 13 REST API access**

working with JSON

using urllib3

accessing urls with urllib3

using requests module

GET,POST using requests module

**Module 15 PANDAS**

Getting Started

Series

Data Frames

Read CSV

Read JSON

Analysing Data

**Module 16 NUMPY**

Getting Started

Creating Arrays

Array Indexing

Array Join

Array Split

Array Search

Array Sort

Array Filter

**SQL**

**Day 12**

Simple SQL Queries with SELECT, FROM, WHERE, ORDER BY clauses and Wildcards

Retrieving Data from Multiple Tables - Inner, Left outer, Right outer and Full outer table joins, COALESCE Function, UNION and UNION ALL , INTERSECT and EXCEPT Statements,

**Day 13**

Maintaining data: INSERT, UPDATE and DELETE rows

Using Subqueries: Simple & Nesting Subqueries

Queries using CASE and CAST

Code simple CREATE TABLE, CREATE VIEW statements, Column INDEXES, Column definition – DEFAULT and NULL

**Day 14**

Use various functions to perform calculations on data.

Organize the data obtained from a query before it is displayed on-screen: Scalar Functions and Arithmetic Column Functions and Grouping (DISTINCT and GROUP BY), Sort, Rank, filter data, Use PIVOT and UNPIVOT Operators

**Day 15:**

* + #1 Difference between OLTP vs OLAP
  + #2 Materialized views.
  + #3 Partition the tables for better management and optimization.
  + #4 Hierarchical and Tree-structured query.
  + #5 Concepts of Dimensions and FACT Tables
  + #6 Snowflake and STAR Schema
  + #7 S
  + QL functions like ROLLUP and CUBE operators.

**ANGULAR**

**Day-16**

* + Overview of Html, CSS, JavaScript, Typescript, Bootstrap
  + Angular Intro

Introduction to Angular Framework

* + Introduction to Angular Framework, History & Overview
  + Environment Setup, Angular CLI, Installing Angular CLI
  + NPM commands & package.json
  + Bootstrapping Angular App, Components, AppModule
  + Project Setup, Editor Environments
  + First Angular App & Directory Structure
  + Angular Fundamentals, Building Blocks
  + MetaData

**Day 17**

Essentials of Angular

* + Component Basics
  + Setting up the templates
  + Creating Components using CLI
  + Nesting Components
  + Data Binding - Property & Event Binding, String Interpolation, Style binding
  + Two-way data binding
  + Input Properties, Output Properties, Passing Event Data

Templates, Styles & Directives

* + Template, Styles, View Encapsulation, adding bootstrap to angular app
  + Built-in Directives, Creating Attribute Directive
  + Using Renderer to build attribute directive
  + Host Listener to listen to Host Events
  + Using Host Binding to bind to Host Properties

**Day 18**

Pipes, Services & Dependency Injection

* + In-built Pipes, Creating a Custom Pipes
  + Services & Dependency Injections

Template-Driven and Reactive Forms

* + Template-Driven vs Reactive Approach
  + Understanding Form State
  + Built-in Validators & Using HTML5 Validation
  + Grouping Form Controls
  + FormGroup, FormControl, FormBuilder
  + Forms with Reactive Approach
  + Predefined Validators & Custom Validators
  + Showing validation errors

Components Deep Dive / Routing

* + Component Life Cycle Hooks
  + Reusable components in angular using <ng-content>
  + Navigating with Router links
  + Understanding Navigation Paths
  + Navigating Programmatically
  + Passing Parameters to Routes
  + Passing Query Parameters and Fragments
  + Setting up Child (Nested) Routes
  + Outsourcing Route Configuration (create custom module)

**Day-19**

* Http Requests / Observables
  + HTTP Requests
  + Sending GET Requests
  + Sending a PUT Request
  + Using the Returned Data
  + Catching Http Errors
  + Basics of Observables & Promises
* Deployment / Authentication
  + Deployment
  + How Authentication works in SPA
  + JSON Web Tokens
  + Signup, Login and logout application
  + Router Protection, Route Guards
  + CanActivate interface

**JAVA**

**DAY-20**

• Java technology overview  
o Components that make up the Java programming language  
o Java Features  
o Java development tools  
o Different deployment architectures  
o Java language syntax  
o Construct statements in Java  
  
• Compiling and Running a Simple Program  
o Java Platform  
o Setting Up Your Computer  
o Writing a Program  
o Compiling the Program  
o Interpreting and Running the Program  
  
• Basics I: Flow Control in Java  
o Decision making flows  
§ if-else blocks  
§ switch-case blocks  
o Iterative flows  
§ while loop  
§ do-while loop  
§ for loop

• Java development environment  
o JRE  
o JDK  
o JVM  
o Classpath  
o Path

**DAY-21**

• Object Oriented Programming using Java  
o Class, method, properties  
o Creating Objects  
o Constructors  
o Constructor Overloading  
o Method Overloading  
o The this keywords  
o Packages  
o Access specifiers

o import keyword  
o Life Cycle of an Object  
o Java memory management  
o Garbage Collection  
  
• Inheritance and Polymorphism  
o Inheritance in Java  
o Inheriting classes  
o Overriding methods  
o Creating Abstract classes, interfaces and methods  
  
• Other Class Features  
o Create static variables, methods, and initializers  
o Create final classes, methods, and variables  
o Create and use enumerated types  
o Use the static import statement  
o Arrays of primitives  
o Arrays of objects  
o Enhanced for loop

**DAY-22**

• Classes in Java  
o Object class  
o Overview of java.util.lang package  
o String class  
o Date class  
o Calendar class  
o Stringbuffer class  
o Using string classes  
o Scanner  
  
• Exceptions / Error Handling  
o Define exceptions  
o Use try, catch, throw, throws and finally statements  
o Describe exception categories  
o User Defined exceptions

**DAY-23**

• Collections and Generics Framework  
o Collections framework  
o Legacy collection classes  
o Comparable and Comparator interfaces  
o Use generic collections  
o Use type parameters in generic classes  
• I/O Fundamentals  
o Command-line arguments  
o Properties class  
o Serialize and deserialize objects  
o Distinguish readers and writers from streams  
o Describe files and file I/O

**DAY-24**

• Overview of Java8  
o Functional Interfaces  
o Lambdas  
o Stream API  
o Date / Time API  
o Default and static methods  
o Method References

• Java Database Connectivity (JDBC)  
o Introduction to JDBC API  
o Understanding JDBC Interfaces  
§ Connection  
§ Statement  
§ PreparedStatement  
§ CallableStatement  
§ ResultSet  
o Create, Read, Update & Delete Queries  
o Fetching of results

Case Study using jdbc

**PROJECT PHASE 1**

**Day 25** : Project Phase 1

**Day 26** : Project Phase 1

**SPRING MVC**

**DAY-27**

- Spring Framework Architecture

-setter injection, constructor injection  
- Spring's Dependency Injection and Auto wiring feature

- Different configuration

**DAY-28**

- Aspect Oriented Programming (AOP) and AspectJ  
- Data access mechanisms provided by Spring

- Spring JDBCTemplate

**DAY-29**

- Integrating JpaHibernate with Spring using case study  
- Spring MVC using case study

**Day-30**

-Webservices (Soap , Rest)

-Spring Rest

-Spring Data Jpa using case study

**Day- 31**

-Spring Boot Introduction

-Spring Boot Features

-Spring Boot Rest Data Jpa Using a case study

• Servlets and JSPs: What's Missing  
• The MVC Pattern  
• The Front Controller Pattern  
• Dispatcher Servlet

**PROJECT PHASE 2**

Day 32 : Project Phase 2

Day 33 : Project Phase 2

**MICROSERVICES**

**Day 34**

* Introduction  
  • Setting Up Your Development Environment  
  • Let's Get Started with Spring Boot  
  • Creating Our First Spring Boot Application - Hello World  
  • Spring Initializr  
  • Maven Overview  
  • Packaging Basics: Make Your App an Executable JAR with Maven  
  • Exercise - Build Your Very First Application
* Fundamentals of Spring Boot  
  • Spring Boot Application Lifecycle  
  • Application Properties and Alternatives  
  • Spring Profiles and Spring Configuration  
  Spring Data with Spring Boot  
  • Introduction to JPA and Spring Data  
  • H2 In-Memory Database Overview  
  • Creating Entities With JPA  
  • Creating Repositories and Persisting Data with JPA and Spring Data  
  • Reading and Querying Data with JPA and Spring Data  
  • Different Ways to Query Data with JPA and Spring Data  
  • Updating Data with JPA and Spring Data  
  • Deleting Data with JPA and Spring Data  
  • Exercise: Build Your Own Spring Data Application

**Day 35**

* RESTful APIs with Spring Boot  
  • Introduction to RESTful APIs  
  • HTTP Request Types  
  • HTTP Response Codes  
  • HTTP Clients  
  • GET with Spring REST  
  • POST with Spring REST  
  • PUT with Spring REST  
  • DELETE with Spring REST  
  • REST API Elements Explained  
  • Refactoring the REST API  
  • Refactoring the REST API  
  • REST with REST Template  
  • Exercise: Build Your Own REST API

**Day 36**

* Testing Spring Boot Application  
  • Testing Spring Boot Application  
  • Test-driven development  
  • Unit testing  
  • Integration testing  
  • JUnit tests for the Spring Boot application  
  • Using Mockito for mocking services  
  • Postman for testing RESTful service contracts  
  Hands on implementations:
* Building Microservices with Spring Boot  
  • Setting up a development environment  
  • Using Spring Boot to build RESTful microservices  
  • Getting started with Spring Boot  
  • Understanding Maven build tool  
  • Developing the Spring Boot Java microservice using STS  
  • The Spring Boot configuration  
  • REST APIs

**Day 37**

API Security:  
• Error handling with RESTful services  
• Authentication and authorization  
• Basic Authentication , JWT With example  
• Implementing Spring Boot Controllers and Exception Handling  
• Implementing Spring Boot Bean Validation  
• Implementing Unit testing and Mock Testing for Spring Boot  
• Implementing Spring Boot security – Basic Authentication, JWT  
• Implementing Spring Boot Developer Tools  
• Implementing Spring Boot HAL Browser  
• Implementing Spring Boot JPA using H2  
• Implementing Spring Data using H2  
• Implementing Spring Data REST  
• Documenting microservices with Swagger  
• Implementing Spring Boot actuators

**Day 38**

* Microservices and Spring Boot  
  Demystifying Microservices  
  • The evolution of microservices  
  • What are microservices?  
  • Microservices – the honeycomb analogy  
  • Principles of microservices  
  • Characteristics of microservices  
  • Challenges for a successful microservice architecture  
  • Microservices examples  
  • Microservices benefits  
  • Microservices vs SOA  
  • Relationship with other architecture styles  
  • Microservice use cases

**PROJECT PHASE 3**

Day 39 : Project Phase 3

Day 40 : Project Phase 3

**TESTING**

**Day 41 :**

Junit Testing with Mockito 1Day

* + Types of Tests
  + Why Unit Tests Are Important
  + What's JUnit?
  + JUnit 5 Architecture
  + IDEs and Build Tool Support
  + Setting up JUnit with Maven

**Day 42:**

Quality Assurance versus Quality Control

The Cost of Quality, Why Do We Test Software?

What is a Defect? Defect Life Cycle & Management

The Multiple Roles of the Software Tester

Testing Constraints, Levels & Types of Testing

Retesting and Regression Testing

The “V” Concept of Testing

Verification versus Validation

Static versus Dynamic Testing

Examples of Specific Testing Techniques

Test Planning, Write the Test Plan

Test case Design, execution & Reporting

Test Coverage – Traceability matrix

**Day 43 :**

Basics of automation testing – why, when and how to perform automation testing

Factors for choosing a particular tool

An overview of Selenium

**Day 44 :**

API Testing using postman and Restassured from spring boot applications

**Day 45:**

Performance Testing

Importance of Performance Testing

Performance Testing types - Load Testing, Endurance testing , Stress testing, Scalability Testing

Overview on Performance Testing tools like JMeter, Neoload & Load runner

Security Testing

Importance of Security Testing

Security Testing types: Penetration Testing, Vulnerability Scanning, Dynamic application security testing

Overview on Security Testing tools like Burp suite, Owasp

**PROJECT PHASE 4**

**Day 46** : Project Phase 4

**Day 47** : Project Phase 4

**Day 48** : Project Phase 4

**EVALUATION**

**Day 49**

**Day 50**