

Index

V&V Automation Testing (Selenium + Java) Course Structure.....	2
V&V Automation Testing (Selenium+Java) TOC.....	3
Web Basics (HTML5, CSS3, JavaScript, XML)	3
DBMS/SQL	5
Testing Concepts & Use Case.....	6
Introduction to Agile.....	9
Defect Reporting and JIRA	11
Core Java 8 & Development Tools for V&V Automation Testing.....	12
Test Automation & Advanced Selenium	14
TDD & BDD.....	17
Introduction to Git & Jenkins	18

V&V Automation Testing (Selenium + Java) Course Structure

Sr. No.	Course	Duration (Days)	Remarks
1	Discover	0	Online
2	Soft Skills Foundation - Part 1	1	Soft Skills Part 1
3	Web Basics (HTML 5, CSS 3, JavaScript, XML)	3	
4	DBMS/SQL	2.5	MCQ Test
5	Testing Concepts + Use Case	4.5	
6	Soft Skills Foundation - Part 2	1	Soft Skills Part 2 (Saturday)
7	Introduction to Agile	1	
8	Defect Reporting and JIRA	1	
9	Sprint 1 Implementation	5	Sprint 1 Implementation, MCQ
10	Sprint 1 Evaluation	1	Sprint 1 Evaluation
11	Soft Skills Foundation - Part 3	1	Soft Skills Part 3
12	Core Java 8	5.5	
13	Core Java 8 Test	0.5	Coding+MCQ Test
14	Test Automation & Advanced Selenium	6	
15	Soft Skills Foundation - Part 4	1	Soft Skills Part 4 (Evaluation)
16	TDD & BDD	4	
17	Introduction to Git & Jenkins	2	
18	Sprint 2 Implementation	5	Sprint 2 Implementation, MCQ
19	Sprint 2 Evaluation	1	Sprint 2 Evaluation
20	L1 Test+L1 Preparation	2	
Total Training Duration		48	

V&V Automation Testing (Selenium+Java) TOC

Web Basics (HTML5, CSS3, JavaScript, XML)

Program Duration: 3 days

Contents:

- HTML Basics
 - Understand the structure of an HTML page.
 - New Semantic Elements in HTML 5
 - Learn to apply physical/logical character effects.
 - Learn to manage document spacing.
- Tables
 - Understand the structure of an HTML table.
 - Learn to control table format like cell spanning, cell spacing, border
- List
 - Numbered List
 - Bulleted List
- Working with Links
 - Understand the working of hyperlinks in web pages.
 - Learn to create hyperlinks in web pages.
 - Add hyperlinks to list items and table contents.
- Image Handling
 - Understand the role of images in web pages
 - Learn to add images to web pages
 - Learn to use images as hyperlinks
- Frames
 - Understand the need for frames in web pages.
 - Learn to create and work with frames.
- HTML Forms for User Input
 - Understand the role of forms in web pages
 - Understand various HTML elements used in forms.
 - Single line text field
 - Text area
 - Check box
 - Radio buttons
 - Password fields
 - Pull-down menus
 - File selector dialog box
- New Form Elements
 - Understand the new HTML form elements such as date, number, range, email, search and datalist
 - Understand audio, video, article tags
- Introduction to Cascading Style Sheets 3.0

- What CSS can do
 - CSS Syntax
 - Types of CSS
- Working with Text and Fonts
 - Text Formatting
 - Text Effects
 - Fonts
- CSS Selectors
 - Type Selector
 - Universal Selector
 - ID Selector
 - Class selector
- Introduction to JavaScript
 - Basic Concepts of JavaScript
 - Embedding JavaScript in HTML
- JavaScript Language
 - Data Types and Variables
 - JavaScript Operators
 - Control Structures and Loops
 - JavaScript Functions
- Working with Predefined Core Objects
 - Data Types in JavaScript
 - String Objects
 - URL String Encoding and Decoding
 - Math Properties
 - Math Objects
 - Date Objects
 - Date and Time Arithmetic
- Working with arrays
 - Arrays object, its properties and methods
- Document Object Model
 - Understand the JavaScript Object Model
 - Understand the Window object, Frame and Navigator Object
 - Location and History Object
- Working With Document Object
 - Document Object and its properties, methods and events
 - Link ,Anchor & Cookies object
- Working with Form Object
 - Form Object Properties, Methods & Event Handlers
 - Text-Related Objects
 - Button Objects
 - Check Box and Radio Objects
 - Select Objects
 - Validate Data and Form Submission
- Work with Regular Expressions
 - Search using simple patterns
 - Use regular expressions
 - Search using special characters
 - Work with RegExp objects
- Introduction to XML

- Evolution of XML
- Role of XML in Web Applications
- Different members of XML family
- Introduction to Namespace
- Anatomy of an XML Document
 - Logical and Physical structure of XML file
 - Parts of XML file like Elements, Attributes , Entities and Processing instructions

DBMS/SQL

Program Duration: 2.5 days

Contents:

- Getting Started with Database
 - Introduction to Database
 - Characteristics of DBMS
 - Data models
 - Relational DBMS
 - Database Administrator
- Basics of SQL
 - The SQL Language
 - Rules for SQL Statements
 - Standard SQL Statement Groups
 - Logging to Oracle Server
- 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 clauses
 - 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
 - SQL: 1999 Compliant Joins

- Types of Joins
 - Sub-query
 - CONNECT BY and START WITH clauses
 - Tips and Tricks
- Introduction to Data Modeling, E-R model
 - Data Modeling
 - E-R model
- Database Objects
 - Basic Data Types
 - Data Integrity
 - Examples of CREATE TABLE
 - Examples of ALTER TABLE
 - Database Objects
 - Index
 - Synonym
 - Sequence
 - View
 - Deleting Database Objects
- Data Manipulation Language
 - Adding Data
 - Removing Data
 - Modifying Data
- Transaction Control Language
 - Introduction to Transactions
 - Statement Execution and Transaction Control

Testing Concepts & Use Case

Program Duration: 4.5 days

Contents:

- Fundamentals of Testing
 - Introduction to Software Testing
 - Software Testing - Definitions
 - Need of Software Testing
 - Error-Failure-Defect
 - Causes of Software Defects
 - Cost of Software Defects
 - What does Software Testing reveal?
 - Importance of Software Testing
 - Importance of Testing Early in SDLC Phases
 - Testing and Quality
 - Quality Perception
 - Seven Testing Principles
 - Economics of Testing
 - How Testing is conducted?
 - Software Testing – Then (Past)
 - Software Testing – Now (Present)
 - Scope of Software Testing

- Factors influencing the Scope of Testing
- Risk Based Testing
- Project Risks
- Product Risks
- Need of Independent Testing
- Activities in Fundamental Test Process
- Attributes of a good Tester
- Psychology of Testing
- Code of Ethics for Tester
- Limitations of Software Testing
- Testing Throughout the Software Development Lifecycle
 - Software Development Lifecycle Models
 - Software Development and Software Testing
 - Software Development Lifecycle Models in Context
 - Test Levels
 - Component Testing
 - Integration Testing
 - System Testing
 - Acceptance Testing
 - Test Types
 - Functional Testing
 - Non-functional Testing
 - White-box Testing
 - Smoke & Sanity Testing
 - Change-related Testing
 - Test Types and Test Levels
 - Maintenance Testing
 - Triggers for Maintenance
 - Impact Analysis for Maintenance
 - Test Case Terminologies
 - Test Data
- Static Testing
 - Types of Testing Techniques
 - Differences between Static and Dynamic Testing
 - Static Testing Basics
 - Work Products that Can Be Examined by Static Testing
 - Benefits of Static Testing
 - Review Process
 - Work Product Review Process
 - Roles and responsibilities in a formal review
 - Review Types
 - Applying Review Techniques
- Test Techniques
 - Categories of Test Techniques
 - Choosing Test Techniques
 - Categories of Test Techniques and Their Characteristics
 - Black-box Test Techniques
 - Equivalence Partitioning
 - Boundary Value Analysis
 - Decision Table Testing

- State Transition Testing
- Use Case Testing
- White-box Test Techniques
- Statement Testing and Coverage
- Decision Testing and Coverage
- The Value of Statement and Decision Testing
- Experience-based Test Techniques
- Error Guessing
- Exploratory Testing
- Checklist-based Testing
- Test Management & Test Metrics
 - Test Organization
 - Independent Testing
 - Tasks of a Test Manager and Tester
 - Test Planning and Estimation
 - Purpose and Content of a Test Plan
 - Test Strategy and Test Approach
 - Entry Criteria and Exit Criteria (Definition of Ready and Definition of Done)
 - Test Execution Schedule
 - Factors Influencing the Test Effort
 - Test Estimation Techniques
 - Test Monitoring and Control
 - Metrics Used in Testing
 - Purposes, Contents, and Audiences for Test Reports
 - Configuration Management
 - Risks and Testing
 - Definition of Risk
 - Product and Project Risks
 - Risk-based Testing and Product Quality
 - Defect Management Testing Metrics
- Requirement Engineering
 - Evolution of Requirements
 - Who provides the Requirements?
 - Challenges in Requirement Gathering
 - Why do we need good requirements?
 - Characteristics & Impact of bad Requirements
 - Requirement Engineering
 - Functional Vs Non-Functional Requirements
 - Non Functional Requirements: FURPS +
 - Stable and Volatile Requirements
 - Baselining Requirements
 - Requirements Traceability
 - Requirement Traceability Matrix
 - Maintaining Requirement Traceability
 - Requirement Traceability Matrix – Example
 - Requirements Change
 - Change Management Process
 - Requirement Creep
- Use Case Testing
 - Use case modeling

- Advantage of use cases
- Actor
- Goals and Requirements
- Goals and scenarios
- Naming Conventions
- Alternate Path
- Exceptions
- Errors
- Precondition & Postcondition
- Good practices
- Failure scenarios
- Software Version Guidance
 - Introduction to Software Versioning
 - Major Release
 - Minor Release
 - Revision Release
 - Build Release
 - Beta Version for User Testing

Introduction to Agile

Program Duration: 1 day.

Contents:

- Agile Process Framework
 - History of Traditional Software Development Model
 - Software Development Model and SDLC
 - “Waterfall Model” – An Overview
 - Waterfall or Sequential Based Development Model
 - “Real Life” – Waterfall Model
 - “Waterfall Model” – Advantages
 - “Waterfall Model” – Disadvantages
 - Agile Software Development – Definition
 - Agile Development Model
 - Graphical Illustration of Agile Development Model
 - Why use Agile?
 - Agile Manifesto and Principles
 - 12 Principles of Agile Methods
 - Agile Values
 - What is NOT an Agile software development?
 - Foundation of an Agile software development Method
 - Common Characteristics of Agile Methods
 - Agile Methods and Practices
 - When to use Agile Model?
 - Advantages of Agile Model
 - Disadvantages of Agile Model
 - Difference between Agile and Waterfall Model
 - Agile – Myths and Reality
 - Agile Market Insight

- Agile Methods and Practices - SCRUM
 - Introduction to SCRUM
 - Scrum Roles and Responsibilities
 - Scrum Core Practices and Artifacts
 - User Story
 - Sprint
 - Release Planning Meeting
 - Sprint Planning Meeting
 - Daily Scrum Meeting (Daily Stand up)
 - Sprint Review Meeting
 - Retrospective
 - Product Backlog
 - Sprint Backlog
 - Burn-Down Chart
 - Velocity
 - Impediment Backlog
 - Definition of “Done”
 - Splitting User Story into Task
 - Why to Split User Story into Task?
 - Guidelines for Breaking Down a User Story into Tasks
 - Examples of Scrum Task Board
 - Planning Poker®
 - Planning Poker® - Process/Steps
 - What are Story Points?
 - How do We Estimate in Story Points?
 - What Goes into Story Points?
- Agile Methods and Practices - Extreme Programming (XP), Lean Software Development & Kanban
 - Introduction to Extreme Programming
 - The Rules of Extreme Programming
 - Extreme Programming (XP) - Principles
 - Extreme Programming (XP) – Key Terms
 - Introduction to Lean Software Development
 - Principles of Lean Software Development
 - What is Kanban?
- Introduction to Agile Testing
 - What is Agile Testing?
 - Agile Team - Roles and Activities
 - Where does Tester fit in Agile Team?
 - Agile Team – Tester’s Role and Responsibilities
 - Agile Team – Test Manager’s Role and Responsibilities
 - How is Agile Testing different?
 - Traditional Testing Vs. Agile Testing
 - What is Iteration 0?
 - User Story Perspective Agile Testing Process
 - Tester’s Change in Mind-Set – A key to success
- Agile Testing Quadrants and Agile Test Planning
 - An overview of Agile Testing Quadrants
 - Agile Testing Quadrant 1, 2 & 3 Goals
 - Agile Testing Quadrant 1, 2 & 3 Toolkit

- Test Planning in Agile Testing

Defect Reporting and JIRA

Program Duration: 1 day.

Contents:

- Defect Free Defect Reporting
 - What is Software Quality?
 - Defect Definition
 - Why find Defects?
 - Impact of Defects
 - Legal Implications
 - Life-cycle workflow
 - Life-cycle workflow - Enhancement
 - Defect Report - Definition
 - Defect Reporting – The Need
 - Defect Report - Template
 - Important Attributes
 - Example on Severity & Priority
 - Defect Report Users
 - Defect Management
 - Logging, Tracking & Analysis
 - What is Defective Defect Report?
 - The reasons for defective reports
 - Other Influencing Factors
 - The Most Severe Defect
 - Importance of Effective Defect Reporting
- Overview of Defect Tracking Tools
 - Introduction to Defect Tracking Tool
 - Introduction to Bugzilla
 - Features of Bugzilla
 - Bugzilla – User Interface
 - Introduction to JIRA
 - Introduction to HP ALM
 - Features of HP ALM
 - HP ALM – User Interface
- Introduction to JIRA
 - Overview of JIRA
 - Features of JIRA
 - JIRA Users
 - JIRA Software Workflow
 - Basic Concepts of JIRA
 - Issue, Project, Workflow, Components & Versions
 - JIRA Software
 - Overview on Issues
 - Priority of Issues

- Issue Workflow
- Overview of Projects
- Sub tasks
- Hands-on JIRA
 - Login to JIRA
 - Create a SCRUM Project,
 - Inviting team members,
 - Creating components
 - Creating Issues – Epics, Stories, Tasks, Bugs
 - Creating sub-tasks
 - Managing Issues, issue properties.
 - Importing Issues from csv file into JIRA
 - Configuring ZEPHYR for Test Case Management
 - Create Sprints in JIRA & Issue Workflow
 - JIRA Reports – Epic report, Created vs Resolved Issues Report, Sprint report, Burn down chart, Velocity chart.

Core Java 8 & Development Tools for V&V Automation Testing

Program Duration: 5.5 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
 - 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 Basic Java Class Libraries
 - 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
- 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
- Introduction to JUnit 4 /Test NG
 - Introduction
 - JUnit
 - Installing and Running JUnit
 - Testing with JUnit - Annotations
 - Testing Exceptions
 - Test Fixtures
 - Best Practices

Test Automation & Advanced Selenium

Program Duration: 6 days.

Contents:

- Introduction to Automation
 - What is Test Automation?
 - Why to Automate?
 - Manual Testing Vs Automation Testing
 - Manual to Automated Testing – The Process
 - Advantage of Automation Testing
 - What Should be automated and what should not be automated?
 - Automation Testing – Best Practices
 - Common Misconceptions About Automated Testing
 - Example of Test Automation
- Introduction to Selenium
 - Introduction to Selenium
 - Selenium: What it is?
 - Landscape and Usage
 - Overview of Selenium Core
 - Overview Selenium Remote Control
 - Overview of Selenium IDE
 - Overview of Selenium Web Driver
 - Overview of Selenium Grid

- Why Selenium?
- Selenium WebDriver
 - Introduction to WebDriver
 - Selenium WebDriver Architecture
 - Selenium WebDriver Architecture - Components
 - Web Driver Vs Selenium RC Vs Selenium IDE
 - Benefits of Web Driver over Selenium IDE and RC
 - Limitations of Web Driver
- Testing Web Applications Using Web Driver API
 - Basic Setup for Automation Script
 - Introduction to Maven Project, dependencies
 - Packages – Pages, Tests
 - src/main/java, src/main/resources
 - src/test/java, src/test/resources
 - test-output - Reports
 - Writing First WebDriver Test
 - Locating UI Elements-Developers Tools
 - Locators in Selenium
 - Introduction to WebElement Interface
 - WebDriver API Methods - findElement() and findElements()
 - Locating UI Elements using By Strategy
 - Difference between findElement() and findElements()
 - Locating Elements by CSS Selectors
 - Locating Element by DOM
 - Locating Elements by XPath
 - Introduction to XPath
 - Types of XPath (Absolute & Relative)
 - Methods of XPath (Parent, Child, Ancestor, Self, Descendant, contains(), starts-with(), ends-with(), text(), preceding, AND & OR expression, sibling expression)
 - Navigation API
 - Interrogation API
 - WebElement API
 - Interacting with Form Elements Using WebDriver API
 - Interacting with Dropdown-box Using WebDriver API
 - Handling Popup Dialogs and Alerts – accept(), dismiss(), getText()
 - Handling Multiple Windows in Selenium WebDriver
 - getWindowHandle() and getWindowHandles() - Example
 - Handling Synchronization in Selenium WebDriver
 - Types of Synchronization in Selenium WebDriver – Implicit wait, Explicit wait, Fluent wait, setSpeed(), sleep()
 - Exceptions in Selenium WebDriver – NoSuchElementException, NoAlertPresent, ElementNotVisible, StaleElementReference
 - Execute JavaScript Based Code in Selenium WebDriver
 - JavaScript Executor – Scenarios
- Web Driver Test with TestNG
 - Introduction to TestNG
 - TestNG Annotations
 - Assertions/Verifications with TestNG
 - Web Driver Test cases with TestNG

- Generating TestNG report
 - Test Suite
- Selenium WebDriver - Advanced
 - Overview of Cross Browser Testing
 - Cross Browser Testing in Selenium WebDriver
 - Launching Firefox Browser with Selenium 3 & GeckoDriver
 - Launching Edge Browser using Microsoft Edge Driver with Selenium 3
 - Introduction to Headless Browsers
 - Other Important Browsers
 - Introduction to Selenium Grid
 - What is Selenium Grid?
 - Selenium Grid Architecture – The Hub & The Node
 - Selenium Grid Architecture – Configuring Hub and Node
 - Selenium Grid Architecture – RemoteWebDriver
 - DesiredCapabilities and Profile Setting in Selenium WebDriver
- Working with Page Object Model (POM)
 - Why Page Object Model (POM)?
 - What is Page Object Model (POM)?
 - Page Object Model (POM) Architecture
 - Advantages of Page Object Model (POM)
 - Overview of Selenium Design Patterns
 - Importance of Design Patterns in Selenium Automation Testing
 - Hand-on Page Object Model
- Working with Page Factory & Object Repository
 - Introduction to Page Factory Design Pattern
 - Advantages of Page Factory Design Pattern
 - Implementing Page Factory Design Pattern
 - Difference between Page Object Model (POM) and Page Factory
 - Parameterization – CSV, MS Excel, Properties, DataProvider
 - Object Repository
 - Properties File
 - XML File
 - JSON File
- Reports
 - Introduction to Extent Reports
 - Benefits of Extent Reports
 - ExtentReports & ExtentTest Class
 - Methods of ExtentReports & ExtentTest Classes (startTest, endTest, Log, Flush)
 - Capturing Screenshots in Extent Reports - getScreenShotAs()
- Selenium Frameworks
 - Framework Overview
 - Linear
 - Modular
 - Data Driven (Excel, Databases)
 - Keyword Driven
 - Hybrid Driven
 - TDD (JUnit, TestNG)
 - BDD (Cucumber, PyBehave, SpecFlow)
 - CI Tools (Jenkins)

TDD & BDD

Program Duration: 4 day.

Contents:

- Introduction to TDD (Test Driven Development)
 - What is Test Driven Development (TDD)?
 - Relationship between ATDD & TDD
 - Levels of Test Driven Development
 - Steps in Test Driven Development
 - TDD vs Traditional Testing
 - TDD and V-model
 - Why TDD?
 - TDD - Myths and Misconceptions
 - Tools
 - Example of TDD
- Introduction to BDD
 - What is BDD?
 - Origin of BDD
 - BDD Framework
 - Features of BDD
 - BDD vs Traditional Automation
 - BDD vs TDD
 - BDD implementation
 - BDD Tools
- Gherkin Language Basics
 - Introduction to Gherkin Language
 - Generating Behavior test cases
 - Running test cases
 - Creating tags
 - Categorizing tests based on tags
- Gherkin Language – Features, Scenario, Scenario Outline
 - Feature
 - Background
 - Scenario
 - Given Keyword
 - When Keyword
 - Then Keyword
 - And Keyword
 - But Keyword
 - * Keyword
 - Scenario Outline
- Cucumber – Introduction
 - Cucumber Framework
 - How it Works
 - Advantages of Cucumber
 - Feature File
 - Steps Definitions
 - Cucumber Options
 - Scenario Outline

- Data Tables in Cucumber
- Cucumber - Tags, Hooks and Background
 - Cucumber Tags
 - Cucumber Hooks
 - Background in Cucumber
- Selenium – Business readable UI automation with cucumber
 - Introduction
 - Implementing Business readable UI automation with cucumber

Introduction to Git & Jenkins

Program Duration: 2 Days

Contents:

- Introduction to CI
 - Why CI? Software development before CI & Software development with CI
 - Benefits of CI
 - Overview on various CI tools – Jenkins, Buildbot, Travis, Bamboo
- Introduction to Git
 - Getting a Git Repository
 - Recording Changes to the Repository, Viewing the Commit History
 - Undoing Things, Working with Remotes
 - Tagging, Git Aliases
 - Git Branching
 - Branches in a Nutshell, Basic Branching and Merging
 - Branch Management
 - Remote Branches, Rebasing
- Introduction to GitHub
 - Introduction
 - Set up Git
 - Create a repository
 - GitHub Flow
 - Contribution to Projects
 - Communicating on GitHub
- Introduction to Jenkins
 - Introduction to Jenkins
 - How Jenkins works
 - Features of Jenkins
 - Jenkins Installation - Unlock Jenkins, Customize Jenkins, Install Plugins
 - Overview of Configuring Jenkins (Manage Jenkins/Plugins/Users/RBAC)
 - Creating Users, Permissions, Roles
 - Jenkins Git/GitHub Integration
 - Creating Configuring and Running Jenkins Jobs
 - Creating Job with Maven & Git
 - Jenkins- Unit testing
 - Jenkins-Reporting