**Index**

Contents

[Contents 1](#_Toc25761605)

[SFDC LoT Course Structure 2](#_Toc25761606)

[RDBMS 3](#_Toc25761607)

[Web Basics (HTML5, CSS3, JavaScript, Polymer,JSON) 4](#_Toc25761608)

[BootStrap & Angular 8](#_Toc25761609)

[Core Java 8 and Development Tools 9](#_Toc25761610)

[Salesforce Basic Administration 11](#_Toc25761611)

[Salesforce Development – App Builder 13](#_Toc25761612)

[Salesforce Development – Platform Developer 15](#_Toc25761613)

[Lightning experience and administration & Lightning Web Component (LWC) basics 16](#_Toc25761614)

[Advance Apex Coding & Community, Integration essentials 17](#_Toc25761615)

## **SFDC LoT Course Structure**

SFDC LoT provides exposure to Salesforce and related technologies. It focuses on application development using Salesforce. The following table lists the course structure for SFDC.

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Course** | **Duration (In Days)** |
| **1** | Discover (Induction) | 1 |
| 2 | Soft Skills Day 1 | 1 |
| 4 | RDBMS | 2 |
| **5** | Web Basics (HTML 5, CSS 3, JavaScript,polymer,JSON) | 5 |
| 6 | BootStrap & Angular 6 | 5 |
| **7** | **Module 1 Test** | 0.5 |
| **9** | Soft skills Day 2 & 3 | 1.5 |
| 10 | Core Java 8 & Development Tools | 6 |
| **11** | **Module 2 Test** | 0.5 |
| 12 | Soft Skills Day 4 & 5 | 2 |
| **13** | Salesforce Basic Administration | 3 |
| 14 | Salesforce Development – App Builder | 6 |
| **15** | **Module 3 Test** | 0.5 |
| **16** | Salesforce Development – Platform Developer I | 5 |
| 17 | Lightning experience and Programming Lightning Components & Lightning Web Component (LWC) basics | 6 |
| 18 | Advance Apex Coding & Community Overview, Integration essentials | 3 |
| **19** | **Module 4 Test** | 0.5 |
| 20 | Mini Project presentation | 1 |
| 21 | Soft Skills Day 6 | 1.5 |
| 22 | Agile + PLP + PLP Presentation | 7 |
| 21 | L1 Preparation + Test | 2 |
|  | **Total Training Duration** | **60** |

**RDBMS**

**Program Duration:** 2 days

**Contents:**

* Introduction to Database using MySQL
  + Getting Started with Database(MySQL)
  + 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
    - SQL: 1999 Compliant Joins
  + Types of Joins
  + Sub-query
* Database Objects
  + Basic Data Types
  + Data Integrity
  + Examples of CREATE TABLE
  + Examples of ALTER TABLE
  + Database Objects(Index, Synonym, Sequence and View)
  + Deleting Database Objects
* Data Manipulation Language
  + Adding Data
  + Removing Data
  + Modifying Data
* Transaction Control Language
  + Introduction to Transactions
  + Statement Execution and Transaction Control

**Web Basics (HTML5, CSS3, JavaScript,Polymer,JSON)**

**Program Duration:** 5 days.

**Provide few online links to learn HTML and CSS,**

**Contents:**

* HTML Introduction
  + What is HTML?
  + HTML Tags
  + HTML Page Structure
  + The <!DOCTYPE> Declaration
  + HTML Versions Web Browsers
  + Working of WWW
  + HTML – Static and Dynamic Web Pages
* HTML Basics
  + HTML Documents
  + HTML Paragraphs
  + HTML Links Learn to manage document spacing
  + HTML Elements
  + HTML Attributes
  + HTML Comments
  + HTML Quotation and Citation Elements
  + HTML Text Formatting
* 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
* Tables
  + Understand the structure of an HTML table,<table>,<tr>,<td><tfooter><thead><tbody>
  + Learn to control table format like cell spanning, cell spacing, border
* List
  + Numbered List
  + Bulleted List
* HTML Block
* HTML The class Attribute
* HTML The id Attribute
* Frames & Iframes
  + Understand the need for frames in web pages.
  + Learn to create and work with frames.
* Responsive Web Design
* HTML and XHTML
* 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
  + HTML5 Semantic Elements
  + HTML5 Migration
* 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 Layout - float and clear
* CSS Combinators
* CSS Pseudo-classes
* CSS Pseudo-elements
* CSS Dropdowns
* CSS Selectors
  + Type Selector
  + Universal Selector
  + ID Selector
  + Class selector
  + CSS Margins
* CSS Padding
* CSS Outline
* CSS Lists
* CSS Tables
* CSS Layout - The display Property
* Colors and Borders
  + Background
  + Multiple Background
  + Colors RGB and RGBA
  + HSL and HSLA
  + Borders
  + Rounded Corners
  + Applying Shadows in border
* CSS Advanced
  + CSS Rounded Corners
  + CSS Border Images
  + CSS Multiple Backgrounds
  + CSS Colors
  + CSS Gradients
  + CSS Shadow Effects
  + CSS 2D Transforms
  + CSS 3D Transforms
* CSS Responsive
  + Responsive Web Design - Introduction
  + Responsive Web Design - The Viewport
  + Responsive Web Design - Grid-View
  + Responsive Web Design - Media Queries
  + Responsive Web Design - Images
  + Responsive Web Design - Videos
  + Responsive Web Design - Frameworks
  + Responsive Web Design – Templates
* 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
* JSON
* Introduction
* What is JSON ?
* Why JSON ?
* Using JSON
* Working with JSON and JavaScript
* JSON.stringify and JSON.parse
* Introduction of Polymer
* The Polymer.js Files
* Loading Polymer on Your Web Page
* Polygit CDN
* Parsing and HTML Elements
* The Hello Element
* Including the Hello Element
* Class-style Constructor
* **Properties and Data Binding**
  + Property Types & keys
  + ReflectToAttribute
  + Notify, ReadOnly, and Data Binding
  + Data Binding Under the Hood
  + Data Binding - Objects
  + Native Element Binding
  + Attribute Binding
  + Computed Properties
  + Observers
  + Observer Sub Properties
  + Observer Array Mutations
  + Life-cycle Callbacks
  + Type Extension Element
* **Reuse & Styling**
* Local DOM
* DOM Distribution
* Shady and Shadow DOM
* Light DOM Children
* DOM API
* Observe Node Changes
* Selectors-Combinators
* CSS Variables and Mix-ins
* Custom Property API
* Shim Limitations
* Polymer Event & Custom Event

**BootStrap**

**Program Duration:** 1 days.

**Contents:**

* Introduction
* Bootstrap Introduction
* Why Bootstrap?
* Bootstrap Advantages
* LESS Introduction
* What's new in Bootstrap 4?
* Mobile First Strategy
* Getting Started with Bootstrap
  + Bootstrap home page
  + Downloading Bootstrap
  + Bootstrap pre-compiled folder structure
  + Bootstrap Global Styles
  + Responsive Design
  + Bootstrap basic template
* Bootstrap grid system
* Grid System in Bootstrap
* Grid Options
* Basic Grid Structure
* Bootstrap grid sizes
* Bootstrap 4 Grid Extra Small
* Bootstrap 4 Grid Small
* Bootstrap 4 Grid Medium
* Bootstrap 4 Grid - Large
* Bootstrap 4 Grid - Extra Large
* Bootstrap grid sizes – using Offset
* Multiple Grid
* Bootstrap 4 Images
* Bootstrap 4 Jumbotron
* Bootstrap 4 Alerts
* Bootstrap 4 Buttons
* Bootstrap 4 Button Groups
* Bootstrap 4 Badges
* Bootstrap 4 Progress Bars
* Bootstrap 4 Spinners
* Bootstrap 4 Basic Components
  + Helper Classes & Responsive Utilities
  + Working with Buttons
  + GLYPHICONS
  + List
  + Tables
  + Forms
  + Typography
* Bootstrap Components
  + Page Header
  + Breadcrumb
  + Button Groups
  + Dropdown
  + Nav & Navbars
  + Input Groups
  + Pagination
  + Panel
  + Well
  + Jumbotron
  + Alerts

**Angular 6.0**

**Program Duration**: 4 days

**Contents**:

* Introduction to Typescript
* Introduction to Typescript
* JavaScript & Typescript
* The type system-Variable, Array
* Defining class and interface
* Arrow Functions
* Template Strings
* Defining a module
* Importing a module
* Generics
* Introduction to Angular 6
* What is Angular 6?
* Why Angular 6?
* What is nodeJS?
* Scope and Goal of Angular 6
* Installing and using Angular 6
* Building Blocks of an Angular 6  Application
* A Basic Angular 6 Application
* Working with Angular 6 with Eclipse
* Components
* What is a component?
* Developing a simple component.
* Templates for a component.
* Component lifecycle
* Managing Data
* What is data binding
* One way data binding
* Two way data binding
* Nested component
* Event binding
* Directives
* What are directives?
* Types of directives - component, structural and attribute
* Creating a basic directive
* Handling event & Binding input in attribute directive
* Creating your own structural directive
* Using the structural directive
* Binding input to a structural directive
* Working with Forms
* Forms in Angular 6
* Template & Model Driven Forms
* A Basic Angular Form
* Binding Input Fields
* Displaying Form Validation State & Field Validation State
* Displaying Validation State Using Classes
* Disabling Submit when Form is Invalid
* Service and Dependency Injection
* What is a service?
* Injecting a service to a component
* Application wide dependency injection
* @Injectable classes
* Multiple service instances
* @Optional and @Host
* HTTP Client
* The HTTP providers
* Injecting the providers
* GET call
* Handling error
* About Observables
* POST request
* Working with headers
* Sequential calls & parallel calls
* Pipe
* What is a pipe?
* Passing parameters to a pipe
* Chaining pipes
* Developing a custom pipe
* Routing
* Why use routing?
* Defining a route table
* Navigation using hyperlink & code
* Supplying parameters to a route URL

**Core Java 8 and Development Tools**

**Program Duration**: 6 days

**Contents**:

* Getting Started
  + Introduction to Java
  + Features of Java
  + Evolution in Java
  + Developing software in Java
  + Installation and Setting up Eclipse
  + Introduction to Eclipse IDE
  + Creating and Managing Java Projects
  + Use of Java docs
  + Miscellaneous  Options
* Learning the Java Language
  + Language Fundamentals
    - Keywords
    - Primitive Data Types
    - Operators and Assignments
    - Variables and Literals
    - Flow Control: Java’s Control Statements
    - Best Practices
  + Object-Oriented Programming Concepts & 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
* Essential Classes
  + 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
* Collections
* Array
* One dimensional array
* Multidimensional array
* Using varargs
* Using Arrays class
* Best Practices
  + 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
* DevTools
  + GIT
  + Sonarqube

**Salesforce Basic Administration**

Program Code: TSFDC

Program Overview: To learn basic administration of Salesforce.

Duration: 3 Days

Prerequisite Skills: NIL

Content:

* Introduction
  + - * Introduction to CRM, Cloud
      * Introduction to Salesforce
      * SFDC Dev instance for all participants
* Getting Your Organization Ready for Users
  + Set up the company profile
  + Configure the user interface
  + Set up activities and calendars
  + Configure search settings
  + Set up Chatter, Chatter Groups
  + Mobile Access with Salesforce1
* Setting Up and Managing Users
* Manage user profiles
* Create and manage users
* Troubleshoot user login issues
* Set up Chatter Free Users and Invites
* Security and Data Access
* Restrict logins
* Determine object access
* Set up record access
* Manage record access with the role hierarchy
* Deal with record access exceptions
* Control access to events
* Manage field-level security
* Object Customizations
* Describe the standard object architecture and relationship model.
* Explain how to create, delete, and customize fields and page layouts on standard and custom objects, and understand the implications of deleting fields.
* Given a scenario, determine how to create and assign page layouts, record types, and business processes for custom and standard objects
* Trailhead - Build a Data Model for a Recruiting App
* SALES AND MARKETING APPLICATIONS
* Given a scenario, identify the capabilities and implications of the sales process.
* Given a scenario, identify the appropriate sales productivity features using opportunity tools, and know when products and Price Books should be used.\*
* Describe the capabilities of lead automation tools and campaign management.
* Describe the capability of Salesforce Content.
* Trailhead - Build a Discount Calculator
* SERVICE AND SUPPORT APPLICATIONS
* Describe the capabilities of case management (for example, case processes, case settings, and case comments).
* Given a scenario, identify how to automate case management (for example, case assignment, auto-response, escalation, web-to-case, email-to-case, case teams).
* Describe the capabilities of Salesforce Knowledge.
* Describe the capabilities of the Community application (for example, Ideas and Answers).
* ACTIVITY MANAGEMENT
* Describe the capabilities of activity management (e.g., manage tasks, events,

public calendars, multi-day events, cloud scheduler).

* DATA MANAGEMENT
* Describe the considerations when importing, updating, transferring, and mass deleting data (for example, CSV files, data quality, field mapping, record IDs, external IDs, duplicate records).
* Given a scenario, identify tools and use cases for managing data (for example, data loader, data import wizard).
* Describe the capabilities and implications of data validation tools.
* Describe the different ways to backup data (for example, data export service, exports, data loader).
* ANALYTICS, REPORTS AND DASHBOARDS
* Describe the options available when creating or customizing a report (for example, report type, report format, fields, summarizing data, filtering data, charting, scheduling, and conditional highlighting).
* Describe the impact of the sharing model on reports.
* Describe the options available when creating and modifying dashboards (for example, dashboard components, data sources, chart types, scheduling, and running user).
* Describe the capabilities of custom report types.
* WORKFLOW PROCESS AUTOMATION
* Given a scenario, identify the appropriate automation solution based on the capabilities of workflow/process.
* Describe capabilities and use cases for the approval process
* DESKTOP AND MOBILE ADMINISTRATION
* Describe the capabilities of Salesforce Mobile.
* Describe the installation and synchronization options of Salesforce for Outlook
* APPEXCHANGE
* Identify use cases for AppExchange applications.

**Salesforce Development – App Builder**

**Program Code:** TSFDC

**Program Overview:** In this program participants will learn how to build applications using Salesforce.

**Program Duration:** 6 days,

**Prerequisite Skills:** (OOPs fundamentals, Java/.Net/any Object Oriented programing language, HTML, Javascript)

**Who should attend?** This program is meant for employees in Levels 3, 4, 5 & 6. It is focused towards building CRM applications using Force.com. Those who intend to work in CRM development should attend this training.

**Contents:**

* SALESFORCE FUNDAMENTALS
* Describe the considerations when developing in a multi-tenant environment.
* Describe how the Salesforce Platform features map to the MVC pattern.
* Describe the capabilities of the core CRM objects in the Salesforce schema.
* Identify the common scenarios for extending an application's capabilities using the AppExchange.
* Identify common use cases for declarative customization of the Lightning Platform as well as customization and features of the Heroku platform.
* DATA MODELING AND MANAGEMENT
* Given a set of requirements, determine the appropriate data model.
* Describe the capabilities of the various relationship types and the implications of each on record access, user interface (UI), and object-oriented programming.
* Describe the impact of schema design and modifications on Apex Development.
* Describe how to visualize and create entity relationships.
* Describe the options for and considerations when importing and exporting data into development environments.

**Salesforce Development – Platform Developer**

**Program Overview:** In this program participants will learn how to build applications using Salesforce.

**Program Duration:** 5 days,

**Prerequisite Skills:** (Salesforce AppBuilder Training)

**Contents:**

* LOGIC AND PROCESS AUTOMATION - Part I
* Describe how to programmatically access and utilize the object schema.
* Describe the capabilities and use cases for formula fields.
* Describe the capabilities and use cases for roll-up summary fields.
* Describe the capabilities of the declarative process automation features.
* Describe when to use declarative automation features vs. Apex classes and triggers.
* Describe how to declare variables and constants in Apex and how to assign values using expressions.
* Describe the primitive and complex Apex data types and when to use them.
* Describe how to use and apply Apex control flow statements.
* Describe how to write and when to use Apex classes and interfaces.
* Describe how to use basic SOSL, SOQL, and DML statements when working with objects in Apex.
* LOGIC AND PROCESS AUTOMATION - Part II
* Describe the basic patterns used in triggers and classes to process data
* efficiently.
* Describe when to use and how to write triggers.
* Describe the implications of governor limits on Apex transactions.
* Describe the relationship between Apex transactions, the save execution order, and the potential for recursion and/or cascading.
* Describe how to implement exception handling in Apex.
* Describe how to write Visualforce controllers.
* Describe when and how to use standard Visualforce controllers vs. Apex
* custom controllers and controller extensions.
* Describe the programmatic techniques to prevent security vulnerabilities in Apex and Visualforce.
* Describe how Apex impacts the ability to make declarative changes.
* Trailhead - Apex Specialist
* USER INTERFACE
* Describe how to display Salesforce data using a Visualforce page.
* Describe the types of web content that can be incorporated into Visualforce pages.
* Describe how to incorporate Visualforce pages into Force.com applications.
* Describe the benefits of the Lightning Component framework.
* Describe the resources that can be contained in a Lightning Component.
* TESTING
* Describe the testing framework and requirements for deployment.
* Describe how to write unit tests for triggers, controllers, and classes.
* Describe when and how to use various sources of test data.
* Describe how to execute one or multiple test classes.
* Describe the differences between invoking Apex in execute anonymous vs. unit tests.
* DEBUG AND DEPLOYMENT TOOLS
* Describe how to monitor and access various types of debug logs.
* Describe the capabilities and security implications of the Developer Console,
* Workbench, and Force.com IDE.
* Describe the different processes for deploying metadata and business data.
* Describe how the different environments are used in the development and deployment process.
* Trailmix

**Lightning experience and administration &** **Lightning Web Component (LWC) basics**

**Program Duration**: 6 days.

**Contents**:

* Introduction to Salesforce Lightning experience
  + Salesforce Lightning Basics
  + Introduction to Lightning Components
  + What can I build with Lightning Components
  + Lightning Component Framework - Hello World Example
  + Overview of Lightning Web Components
  + Lightning Web Component Framework – Hello World example
  + Reports & Dashboards for Lightning Experience
* Migrate to Lightning Experience
* Lightning App Builder
* Lightning Process Builder
* Lightning Alternatives to JavaScript Buttons
* PROGRAMMING LIGHTNING COMPONENTS
* Introduction to Lightning Components
* Creating Lightning Components
* Using Components
* Communication With Events
* Lightning Web Components working with Aura Framework
* Handle Events in Lightning Web Components
* Push and Deploy Lightning Web Component Files
* App Customization Lite (Customize your app's page layouts, compact layouts, and actions)
* Lightning Apps
* Creating Apps
* Using JavaScript
* Using Apex
* Debugging

* Visualforce & Lightning Experience
* Using Visualforce in Lightning Experience
* Developing Visualforce Pages for Lightning Experience
* Exploring the Visualforce App Container
* Sharing Visualforce Pages Between Classic and Lightning Experience
* Managing Navigation
* Understanding Important Visual Design Considerations
* Knowing Which Features to Avoid in Lightning Experience

Trailhead Exercises

Trailhead project - Lightning Experience Rollout Specialist Super badge

* Lightning Web Component (LWC) basics
* Introducing Lightning Web Components
* Create Lightning Web Components
* Communicate with Events
* Work with Salesforce Data
* Use Components in Salesforce Experiences
* Lightning Web Components and Aura Components Working Together
* Debug Lightning Web Components

**Advance Apex Coding & Community, Integration essentials**

**Program Duration**: 3 days.

**Contents**:

* Advance Apex , Visualforce & Lightning Experience
* Use Cases related to real-life projects
* Progressively elaborate the implementation approach - Standard/Custom/VF Page
* Pagination
* Code scanning tools like Checkmarx /Sonarcube/Novacop Leverage
* Community Overview
* Community Builder
* Community Management
* Knowledge Articles & Content
* **Integration essentials**
* Salesforce Integration capabilities
* In a given scenario, recommend when to use API-based integrations, such as SOAP, REST, Bulk, Streaming, Canvas, Workflow outbound, APEX (Callouts, @ future, etc.), and Lightning Connect to achieve business requirements.
* Compare and contrast the advantages and drawbacks (design trade-offs) of using API-based integrations such as SOAP, REST, Bulk, Streaming, Canvas, Workflow outbound, APEX (Callouts, @future, etc.), and Lightning Connect.
* Tools
* Describe Synchronous Vs. Asynchronous execution in Salesforce
* Scenarios where Async execution is preferred, difference in Governor Limits for both types of executions.
* Discuss different Async features offered by Salesforce like future method, Queueable, Batch Apex and Outbound messages.
* Discuss which Async feature to use in different scenarios
* Lab exercise to implement Salesforce Asynchronous features.

**Agile**

**Program Duration**: 0.5 days.

**Contents**:

* **Agile Process Framework**
* **Agile Methods and Practices - SCRUM**

**Pseudo Live Project (PLP)**

**Program Duration**: 6.5 days.

**Contents**:

* Pseudo Live Project (PLP) program is primarily to handhold participants who are fresh into the IT stream & newly recruited from college.PLP project is executed to orient the trainees towards Quality processes followed in the organization. Participants have to understand the value & usage of the various forms, templates & review mechanisms. In PLP, more importance given to “Process Adherence”
* The following SDLC activities are carried out during PLP
  + Requirement Analysis
  + Design ( High Level Design and Low Level Design)
  + Design of UTP(Unit Test Plan) with test cases
  + Coding
  + Code Review
  + Configuration Management
  + Testing
  + Deployment
  + Final Presentation