

ITBRAINY

Means Stack Course Content

Syllabus Covered (HTML)

1. Introduction of different Web Technology
2. Introduction
3. HTML Elements
4. HTML Attributes
5. HTML Headings
6. HTML Paragraphs
7. HTML Formatting
8. HTML Fonts
9. HTML Styles
10. HTML Links
11. HTML Images
12. HTML Tables
13. HTML Lists
14. HTML Forms
15. HTML Frames
16. HTML Iframes
17. HTML Colors
18. HTML Colornames
19. HTML Colorvalues
20. HTML Quick List
21. HTML Layout
22. HTML Doctypes
23. HTML Head
24. HTML Meta
25. HTML Scripts
26. HTML Entities
27. HTML URLs
28. HTML URL Encode
29. HTML Media
30. HTML Audio
31. HTML Object
32. HTML Video
33. HTML YouTube
34. HTML Media Tags
35. HTML Summary

Syllabus Covered (HTML5)

36. HTML5 Introduction
37. HTML5 New Elements
38. HTML5 Video
39. HTML5 Video/DOM
40. HTML5 Audio
41. HTML5 Drag and Drop
42. HTML5 Canvas
43. HTML5 SVG
44. HTML5 Canvas vs. SVG
45. HTML5 Geolocation
46. HTML5 Web Storage
47. HTML5 App Cache
48. HTML5 Web Workers
49. HTML5 SSE
50. HTML5 Tags

Syllabus Covered (CSS)

51. CSS Introduction
52. CSS Syntax
53. CSS Id & Class
54. CSS Styling
55. Styling Backgrounds
56. Styling Text
57. Styling Fonts
58. Styling Links
59. Styling Lists
60. Styling Tables
61. CSS Box Model
62. CSS Box Model
63. CSS Border
64. CSS Outline
65. CSS Margin
66. CSS Padding
67. CSS Advanced
68. CSS Grouping/Nesting
69. CSS Dimension
70. CSS Display
71. CSS Positioning
72. CSS Floating
73. CSS Align
74. CSS Navigation Bar

75. CSS Image Gallery
76. CSS Image Opacity
77. CSS Image Sprites
78. CSS Media Types
79. CSS hacking
80. CSS Summary

Duration:- 20 hrs

Java Script

1. Introduction to client side scripting
2. Syntax Basics
 - a. JS Statements
 - b. JS Comments
 - c. JS Variables
 - d. JS Datatypes
3. JS Operators
4. JS Comparisons
5. JS Conditional Statements
6. JS Loops
7. JS Events
8. JS Objects
 - a. String Functions (Most Used)
 - b. JS Math Functions (Most Used)
 - c. JS Date Functions (Most Used)
 - d. JS Array Functions and Property (Most Used)
9. JS User Defined Functions
10. JavaScript Implementations
 - a. Implementing a Simple Calculator Using JavaScript
 - b. JS Validations using Object Functions
 - c. S Validations using Regular Expressions
11. JS Document Object Model
 - a. Introduction
 - b. DOM HTML
 - c. DOM CSS
 - d. DOM Events
12. JS Browser Object Model
 - a. JS Cookies
 - b. JS Window
 - c. JS Location
 - d. JS Popups
 - e. JS Time

Prerequisite:-

HTML is must.

Duration:- 20 hrs

AJAX (Pre-Requisite: ServerEnd Technology)

1. Introduction
2. XMLHttpRequest Object
3. Creating a request object
4. Sending a request to server
5. Receiving a response from the server
6. readyState and Status of a request

Prerequisite:-

HTML, JS, JQuery is must.

Duration:- 10 hrs

jQuery (Javascript Library)

1. Introduction and Installation
2. Syntax
3. jQuery Selectors
4. jQuery Events
5. jQuery Effects
 - i. jQuery Hide and Show Effect
 - ii. jQuery Fade Effect
 - iii. jQuery Slide Effect
 - iv. jQuery Animate
6. jQuery Callbacks
7. jQuery and HTML

- i. jQuery Get
 - ii. jQuery Set
 - iii. jQuery Add
 - iv. jQuery Remove
 - v. jQuery css
 - vi. jQuery Width
 - vii. jQuery Height
8. jQuery and AJAX (Pre-Requirement: ServerEnd Technology)
- i. AJAX Function
9. jQuery UI :
- i. Implementing Accordion
 - ii. Implementing Datepicker
 - iii. Implementing Slider
 - iv. Implementing Progressbar
 - v. Implementing Tabs

Prerequisite:-

HTML, JS is must.

Duration:- 20 hrs

JavaScript fundamentals

Scope & Function Context
Closures
this keyword
Object-Oriented in JavaScript
Async and Parallel in JavaScript
JavaScript Design Pattern

What are Web Applications?

1. The Challenge with Web Apps (SPA)
2. SPA or Web App's?
3. To bind or not to bind? this is the question
4. MVC or MVVM Frameworks
5. Why AngularJS?

AngularJS Building Blocks for Building SPA

1. Template & live data binding (Directives & \$scope)
2. Model, View & Controller (MVC)
3. Dependency Injection (AngularJS services)
4. Modules
5. LAB : TV Show SPA

Forms in AngularJS

1. ng-model directive
2. ngModelController & FormController
3. Custom Validation
4. Input directive

AngularJS Filters

1. Filter Syntax
2. AngularJS Filters
3. Custom filters

Communication

1. \$.Ajax vs. \$http
2. \$resource
3. Promises (\$q) vs. Calbacks
4. Offline / Online
5. LAB: Full TV Show Web App

AngularJS Internal

1. AngularJS Startup Process
2. AngularJS Runtime
3. Scope API (\$id, \$watch & \$apply)
4. Scope Communication
5. Template Services:
6. Injector Service
7. Modules
8. Caching
9. \$provide service
10. Routing

Custom Directive

1. Template
2. Scope
3. Compile function
4. Link function
5. Controller
6. Transclude
7. Animation
8. Tips & tricks

Routing and Navigation

1. \$location service
2. ng-view directive
3. \$route service and route object
4. Navigation flow
5. Routing broadcasted events
6. Resolve option and promise
7. Cancelling route changes

AngularJS Testing

1. Unit Testing (Jasmine.js)
2. AngularJS Mock API's (ngMock)
3. Unit Testing Tools
4. E2E Testing
5. ngMockE2E - \$httpBackend
6. Protractor Tool

AngularJS Animation

1. CSS3-enabled Animations
2. Directive That Support Animation
3. JavaScript-enabled Animations
4. Using Animations in your own directives

AngularJS Tips & Tricks

1. Optimization & Performance
2. Debugging

3. Localization
4. Mobile
5. AngularJS External Module
- 6.

Prerequisite:-

HTML, JS, JQuery is must.

Duration:- 45 hrs

Angular 4 Syllabus:-

Module 1: Introduction

- Course Objectives
- Course Outline
- What is Angular
- Why use Angular

Module 2: AngularJS to Angular 4

- What's Changed
- Semantic Versioning

Module 3: Introduction to TypeScript

- Why Use TypeScript
- Basic Types
- Classes and Interfaces
- Type Definitions
- Compiling TypeScript

Module 4: ES2015 Primer

- Let and Const
- Template Strings
- Lambda Functions
- Modules

Module 5: Environment Setup

- Node / NPM
- Polyfills
- TypeScript

- Module Bundler (Webpack)
- Code Linting
- Test Setup
- Application File Structure
- Angular CLI
- Code Editors

Module 6: Getting Started

- Our First Component

Module 7: Modules

- Why use Modules
- NgModule
- Declarations
- Providers
- Imports
- Bootstrapping
- The Core Module
- Shared Modules

Module 8: Components

- Introduction to Components
- Component Architecture Patterns
- Decorator Metadata
- State & Behaviour
- Inputs and Outputs

Module 9: Templates

- Inline vs External
- Template Expressions
- Data Bindings
- Built-in Structural Directives
- Built-in Attribute Directives

Module 10: Custom Directives

- Types of Directive
- Create your own Structural Directive
- Create your own Attribute Directive

Module 11: Pipes

- Built-in Pipes
- Custom Pipes

Module 12: Services

- Introduction to Services
- Building a Service

Module 13: Dependency Injection

- Introduction to Dependency Injection
- Injectors & Providers
- Registering Providers

Module 14: Lifecycle Hooks

- Component LifeCycle
- Using ngOnInit
- All lifecycle Hooks

Module 15: Change Detection

- What is Change Detection
- Zone.js

Module 16: Routing

- The Component Router
- Defining Routes
- Navigation
- Route Params
- Child Routes

Module 17: Advanced Routing

- Route Guards
- Route Resolves

Module 18: Template-driven Forms

- Introduction to forms
- Template-driven forms
- Validation

Module 19: Model-driven Forms

- Introduction to 'Reactive' forms
- FormGroup & FormControl
- Validators

Module 20: Asynchronous Operations

- Introduction to Async
- Promises
- Observables
- Async Pipes

- HTTP Request / Response

Module 21: Advanced HTTP

- Headers & Request Settings
- Providing HTTP

Module 22: Component Styling

- Introduction to Angular Styling
- Component Styling
- Shadow DOM
- Loading Styles

Module 23: Animation

- Introduction to Animations
- State & Transitions
- Animatable Properties
- Keyframes

Module 24: Testing

- Introduction to Testing
- Unit Testing
- E2E Testing

Module 25: Security

- Security in Angular
- Sanitization
- Trusting Values
- Cross-site Request Forgery

Module 26: Internationalization

- Pre-compiled and runtime
- Using ng2-Translate

Module 27: Performance

- Change Detection Strategy
- Running outside the Zone
- Web Workers

Module 28: Optimization & Deployment

- Precompiling (AoT)
- Lazy Loading

- Deployment Best Practices
- Production Mode

Module 29: Upgrading from AngularJS 1.X to Angular 2+

- Introduction to Upgrading
- Project Preparation
- Upgrade Adapter

Module 30: Native Applications

- Introduction to Native
- Cordova
- Ionic
- React Native
- NativeScript
- Progressive Web Apps
- Electron

Module 31: Server-side Rendering

- Introduction to Server-side Rendering
- Angular Universal

Module 32: Conclusion and Resources

Node JS Syllabus :-

- Introduction
- Environment Setup
- Node Package Manager
- Callbacks
- Events and Event Loop
- Streams and Buffers
- Express Framework
- Introduction to MongoDB
- Connecting Node.js to Database
- Mongoose Module
- Creating Rest APIs

Mongo DB :-

1. MongoDB Essentials – Recap

- 1.1 Emergence of NoSQL
- 1.2 Types of NoSQL Databases
- 1.3 MongoDB Concepts
- 1.4 The Mongo Shell
- 1.5 Basic CRUD Operations
- 1.6 Availability, Scalability Concepts

2. Advanced MongoDB Concepts

- 2.1 CRUD Operations
 - 2.1.1 insert, bulkInsert
 - 2.1.2 find, findOne() – \$lt, \$lte, \$gt, \$gte, \$ne, \$in, \$nin, \$or, \$exists
 - 2.1.3 cursor operations – limits, skips, sorts
 - 2.1.4 update – \$set, \$unset, \$inc, \$push, \$each, \$addToSet, multiupdate
 - 2.1.5 upsert & findAndModify
 - 2.1.6 remove, \$pop, \$pull
- 2.2 Aggregation Framework
 - 2.2.1 Aggregation Pipeline
 - 2.2.2 Aggregation Reducers & Commands (count, distinct)
 - 2.2.3 Aggregation pipeline Optimization and Limits
 - 2.2.4 Aggregation Examples
- 2.3 Schema Design in MongoDB

2.3.1 Normalization Vs Denormalization

2.3.2 Schema Design – Linking and Embedding

2.3.3 Importing / Exporting mongoDB Database

MYSQL

ITBRANNY