Besant Technologies

A Division of BDreamz Global Solutions Private Limited.

Training Cum Internship Program

Full Stack Master's Program Syllabus

S.No	Course Topics
1	**Front End**
	➢ HTML,HTML5, Bootstrap ➢ CSS,CSS3, ➢ Java Script ➢ Angular Or React JS
2	**Back End**
	➢ Node JS (or) ➢ PHP (or) ➢ .NET
3	**Database**
	➢ MySql (or) ➢ Sql Server
4	**Testing -** Manual Testing
5	**Cloud -** AWS Solutions Architect (or) Microsoft Azure

Duration: 6 Month

Fees: 50000+18%GST

Front end

HTML5 and CSS

HTML Fundamentals

- Basic Tags
- New Tags in HTML5
- Local Storage
- Index DB
- CSS3 Fundamentals
- CSS3 New Properties
- CSS3 Animation
- CSS3 Tooltips

CSS Framework

- Bootstrap

Java Script

Introduction

- What is JavaScript? Complete Introduction with Hello World

JavaScript Concepts and Basics

Statements, JS Syntax , Comments ,Variables ,Operators, Arithmetic , Assignment, Data Types, Functions, Objects, Events, Strings, String Methods , Numbers ,Number Methods , Arrays, Array Methods ,Array Sort ,Array Iteration ,Dates ,Date Formats,Date Get Methods ,Date Set Methods ,Math ,Random, Booleans ,Comparisons ,Conditions ,Switch ,Loop For ,Loop While ,Break ,Type Conversion , Bitwise , RegExp , Errors, Scope , Hoisting ,Strict Mode ,this keyword ,Let , Const ,Debugging ,Style.

- Variable Naming Rules and JavaScript Data Types(With Example)
- Expressions and Operators
- Flow Control
- Defining Functions and Methods
- Constructors and Inheritance
- Pattern Matching with Regular Expressions
- Managing Web Page Styles using JavaScript and CSS

- Handling Web Page Events

JS Forms

- How to Script Forms
- Forms API

JS Objects

- Definitions
- Properties
- Methods
- Accessors
- Constructors
- Prototypes

JS Functions

- Definitions
- Parameters
- Invocation
- Call
- Apply
- Closures

JS HTML DOM

- Intro
- Methods
- Document
- Elements
- HTML
- CSS
- Animations
- Events
- Event Listener
- Navigation
- Nodes
- Collections
- Node Lists

JS Browser BOM

- Window
- Screen
- Location
- History
- Navigator
- Popup Alert
- Timing
- Cookies

JS JSON

- Intro
- Syntax
- Json vs XML
- Data Types
- Parse
- Stringify
- Objects
- Arrays
- PHP
- HTML
- JSONP

JS vs jQuery

- jQuery Selectors
- jQuery HTML
- jQuery CSS
- jQuery DOM

Ajax

- Introduction
- XMLHttp
- Request
- Response
- XML File

- PHP
- ASP
- Database
- Applications
- Examples

Angular

Introduction

- What is Angular? Why Angular?
- Features of Angular
- Difference between Angular 2, 4, 5, 6, 7
- Required software/tools
- Initial setup
- Creating basic Angular application
- Angular application flow
- Brief description of modules and components

Components in Angular

- Creating component using Angular CLI
- Using component selectors
- Lifecycle hooks
- OnInit, OnDestroy , etc as per requirement
- Component interaction
- Parent & child interaction

Data & Event Binding

- One-way data binding
- Two-way data binding
- Event binding
- Examples in each section

Pipes

- What are pipes?
- Inbuilt pipes
- Custom pipes

- Pure & impure pipes

Directives

- What are directives?
- Types of directives
- Custom directives

Structural Directives

- ngIf – syntax and examples
- ngFor – displaying data dynamically – examples
- ngSwitch – syntax and examples
- ng-container

Attribute Directives

- ngClass – syntax and examples
- ngStyle – syntax and examples

Template Driven Forms

- Introduction
- Basic form with input text field
- Validations – required, maxlength, etc
- Submitting form
- Disabling submit when form is invalid
- Additional input types
- Radio buttons
- Checkbox
- Select dropdown
- Date fields

Reactive Forms

- Need of reactive forms
- When to use reactive and template driven forms
- Basic form with input text
- Form validation
- Additional input types
- Dynamic checkboxes
- Radio buttons

- Select dropdown

Services & HTTP Client

- What is a Service?
- Creating a Basic Service
- What is Dependency Injection?
- Injecting Services
- Using a service in a Component
- Using a shared service
- Angular HTTP Client
- Observable
- Making a basic HTTP GET Call
- GET request
- GET request with parameters
- POST request
- Reading HTTP Response
- Using .map()
- catchError()
- Using the Service in a Component

Routing

- Routing module
- Define routes and add router outlet
- Redirecting
- Lazy loading
- Route guards
- Query parameters

Building & Deployment

- Build the project using ‘ng build’
- Deploy the project

Angular 7 Features

- Unit testing using Karma Essential
- Angular Material Essential

React JS

Introduction to React & JSX

- Origins of React
- React.js Syntax
- Overview of JSX and why you should use it
- Getting hold of everything you need to start coding

React Components

- React component Properties
- Setting Properties
- Component Lifecycle
- Updating Components
- Writing your first React.js component
- Mounting Components

JSX

- Expressions & Attributes
- JSX Basics
- Namespaced Components
- Rendering HTML
- Rendering React Components

Flux

- Views & Controller-Views
- Flux is not MVC!
- Data Flow
- Action, Dispatcher, Store & View
- Flux Application Architecture
- Structure

Event Handling In React

- Key Events
- Event Pooling
- React.js Event Handlers
- Synthetic Event

Creating a Dynamic UI in React

- ❖ Keeping components stateless
- ❖ Event Delegation
- ❖ React Stateful Components
- ❖ Auto binding

Integration with Other Libraries

- ❖ Gulp & Browserify
- ❖ React with jQuery
- ❖ React & AJAX

React Server Integration & Deployment

• httpster

• npm

BackEnd

Node JS

Introduction

- ❖ Introduction to JS
- ❖ JS evolution to server
- ❖ Why to use node?
- ❖ How to use node
- ❖ Node Package Manager
- ❖ Advantages of Node JS
- ❖ Traditional Web Server Model
- ❖ js Process Model

First Steps

- ❖ Node Installation
- ❖ what is Express Framework and configuration
- ❖ Compilation
- ❖ Execution cycle

Exploring Data Types & Functions

- ❖ Objects

- Strings
- Numbers
- Auto Casting
- Prototype
- Function
- Self-Invocation Functions
- Array
- Booleans
- Un defined
- null
- Functions
- Buffer
- Module
- Module Types
- Core Modules
- Local Modules
- Module Exports

Control Structures: Logical Expressions

- If statements
- Else and else if statements
- Logical operators
- Switch statements

Control Structures: Loops

- For loops
- For each loops
- Continue
- Break

File System

- Read File
- Writing a File
- Writing a file asynchronously
- Opening a file
- Deleting a file

- Other IO Operations

User-Defined Functions

- Defining functions
- Function arguments
- Returning values from a function
- Multiple return values
- Scope and global variables

Events

- EventEmitter class
- Returning event emitter
- Inhering events

Debugging

- Developers console
- Warnings and errors
- Debugging and troubleshooting
- HTML compilation and Angular Compilation
- Compilation phase

Actual Node Topics

- Modules
- Packages
- Routers
- HTTP
- Buffer
- Reading and Writing Files
- Blocking and Non-Blocking
- Working with Promise
- Process
- Child Process
- Event Loop
- Debugger

PHP

Introduction of Web & PHP

- What is PHP?
- The history of PHP
- Why choose PHP?
- Installation overview

First Steps

- Embedding PHP code on a page
- Outputting dynamic text
- The operational trail
- Inserting code comments

Exploring Data Types

- Variables
- Strings
- String functions
- Numbers part one: Integers
- Numbers part two: Floating points
- Arrays
- Associative arrays
- Array functions
- Booleans
- NULL and empty
- Type juggling and casting
- Constants

Control Structures: Logical Expressions

- If statements
- Else and elseif statements
- Logical operators
- Switch statements

Control Structures: Loops

- While loops

- For loops
- Foreach loops
- Continue
- Break
- Understanding array pointers

User-Defined Functions

- Defining functions
- Function arguments
- Returning values from a function
- Multiple return values
- Scope and global variables
- Setting default argument values

Debugging

- Common problems
- Warnings and errors
- Debugging and troubleshooting

Building Web Pages with PHP

- Links and URLs
- Using GET values
- Encoding GET values
- Encoding for HTML
- Including and requiring files
- Modifying headers
- Page redirection
- Output buffering

Working with Forms and Form Data

- Building forms
- Detecting form submissions
- Single-page form processing
- Validating form values
- Problems with validation logic
- Displaying validation errors

- Custom validation functions
- Single-page form with validations

Working with Cookies and Sessions

- Working with cookies
- Setting cookie values
- Reading cookie values
- Unsetting cookie values
- Working with sessions

MySQL Basics

- MySQL introduction
- Creating a database
- Creating a database table
- CRUD in MySQL
- Populating a MySQL database
- Relational database tables
- Populating the relational table

Using PHP to Access MySQL

- Database APIs in PHP
- Connecting to MySQL with PHP
- Retrieving data from MySQL
- Working with retrieved data
- Creating records with PHP
- Updating and deleting records with PHP
- SQL injection
- Escaping strings for MySQL
- Introducing prepared statements

Building a Content Management System (CMS)

- Blueprinting the application
- Building the CMS database
- Establishing your work area
- Creating and styling the first page
- Making page assets reusable

- Connecting the application to the database

Using Site Navigation to Choose Content

- Adding pages to the navigation subjects
- Refactoring the navigation
- Selecting pages from the navigation
- Highlighting the current page
- Moving the navigation to a function

Application CRUD

- Finding a subject in the database
- Refactoring the page selection
- Creating a new subject form
- Processing form values and adding subjects
- Passing data in the session
- Validating form values
- Creating an edit subject form
- Using single-page submission
- Deleting a subject
- Cleaning up
- Assignment: Pages CRUD
- Assignment results: Pages CRUD

Building the Public Area

- The public appearance
- Using a context for conditional code
- Adding a default subject behavior
- The public content area
- Protecting page visibility

Regulating Page Access

- User authentication overview
- Admin CRUD
- Encrypting passwords
- Salting passwords
- Adding password encryption to CMS

- New PHP password functions
- Creating a login system
- Checking for authorization
- Creating a logout page

Advanced PHP Techniques

- Using variable variables
- Applying more array functions
- Building dates and times: Epoch/Unix
- Formatting dates and times: Strings and SQL
- Setting server and request variables
- Establishing the global and static variable scope
- Making a reference assignment
- Using references as function arguments
- Using references as function return values

Introduction to Object-Oriented Programming (OOP)

- Introducing the concept and basics of OOP
- Defining classes
- Defining class methods
- Instantiating a class
- Referencing an instance
- Defining class properties

OOP in Practice

- Understanding class inheritance
- Setting access modifiers
- Using setters and getters
- Working with the static modifier
- Reviewing the scope resolution operator
- Referencing the Parent class
- Using constructors and destructors
- Cloning objects
- Comparing objects

Working with Files and Directories

- File system basics
- Understanding file permissions
- Setting file permissions
- PHP permissions
- Accessing files
- Writing to files
- Deleting files
- Moving the file pointer
- Reading files
- Examining file details
- Working with directories
- Viewing directory content

Sending Emails

- Configuring PHP for the email
- Sending an email with mail()
- Using headers
- Reviewing SMTP
- Using PHPMailer

.NET

.Net Framework

- CLR, CLS & CTS
- Compilation process in .NET
- Assemblies & Versioning

C#

- Language Syntax
- Data Types, Variables & Operators
- Conditional Statements & Looping Structures
- Garbage Collection and Finalization
- Exception Handling

Classes & Objects

- Classes and Objects
- Abstract Classes and Interfaces
- Constructors and Destructors
- Structures, Enumerations
- Boxing & Unboxing

OOPS

- Encapsulation
- Inheritance
- Polymorphism
- Data Abstraction

Namespace

- Namespace, Nested Namespace
- Delegates & Events
- Properties, Indexer & Indexer Overload
- Errors and Exceptions

Arrays, Collections & Generics

- Single Dimension Array, Multi Dimension Array
- Collections
- Generic Collections

File I/O and Streams

- Working with Directories and Files
- Read and write file

Remoting & Reflection

- Application Domain
- MarshalByRef Object
- Typeof

SQL Server

- Introduction
- DML DDL Functions
- Jins & Views
- Functions & Stored Procedure

- Triggers & Cursors

ADO.NET (Working with Database)

- Overview of ADO.NET
- Connected vs Disconnected Architecture
- Data Connection Object
- Data Command Object
- Data Adapter Object
- Data Readers
- Data Sets & Data Adapters
- Structure of Dataset
- Execute Non Query
- Execute Reader
- Execute Scalar

ASP.NET 4.0

- Introduction to Web Programming
- Client / Server Technology
- Understanding Web Server IIS

Page Life Cycle

- Global.asax
- Web.config
- Intrinsic Objects in ASP.Net

Web Form

- Web Control Class
- Creating Web Forms Application
- Handling Images
- Navigating between Pages
- Managing Server Controls
- Server Control Events
- Using HTML Controls
- Using Data Controls
- Repeater Control

Validation Controls:-

- ASP.Net validation controls
- Configuring validation controls

State Management

- Preserving State in Web Applications
- Using Cookies to Preserve State
- ASP.NET Session State
- Application State

User Controls

- Creating User Controls
- Interacting with User Controls
- Loading User Controls Dynamically

Master Pages & Themes

- Simple Master Page Nested Master Page
- Configuring Master Page Creating Themes
- Applying Themes
- Applying Stylesheet

Uploading Files

- Using FileUpload Control
- Setting the location and filename to upload the files

Handling Emails

- Protocols for Email
- Sending Mails
- Managing Attachments

ASP.NET Web Services

- Introduction to XML Web services
- Creating Web Service
- Setting the Web Service Attribute
- Test and Run Your Web Service
- Consuming a Web Service in Client
- Application
- Consuming a Third Party Web service

Deployment

- Publishing Web Applications.
- Create a Web Setup Project.

Database

SQL Server Developer

MS SQL Training Concepts

DBMS introduction

- What is mean by DBMS and its usage?
- Introduction to the relational databases
- Difference between relational and non-relational databases
- Different types of DBS available in the market
- Different version of MS SQL server

SQL introduction

- Why we are using SQL in DBMS?
- What is SQL and introduction to the SQL scripts?

Walkthrough to the SSMS IDE

- Important icons and options in SSMS
- Get to know all the features in SSMS

SQL commands (DDL, DML, DCL, TCL)

- Types of SQL commands
- DDL:
- DML:
- DCL:
- TCL:

Constraints

- Types of SQL constraints
- Primary key
- Foreign key
- Unique key
- Not null

- Default constraint
- Check constraint

Data types in SQL and UDD

- Integer datatypes
- String datatypes
- Date time data types
- UDD-User Defined Data types

Operators in SQL

- Arithmetic operator
- Logical operator
- Comparison operator

Variables in SQL

- Variable definition
- Types of variables
- Local variables
- Global variables
- MS SQL Training Concepts

System functions

- What is meant by system functions?
- Different types of system functions available in sql server?
- Aggregate functions
- Mathematical functions
- String functions
- Date time functions
- Other functions

Identity

- What is meant by identity?
- How to manually insert a value in identity column?
- How to reseed identity column values?
- How to get the last generated identity values?

Sequence

- What is meant by sequence?
- How to create a sequence?
- Reseeding a sequence object
- How to get the last generated seq value?
- Difference between identity and sequence?
- Order by, group by, top, distinct, having clause and alias

CTE

- Definition – CTE
- CTE creation and usage
- How to use CTE to delete the duplicate records?

Schemas

- What is meant by schema?
- How to create and drop a schema in sql server?

SQL Joins

- What is meant by joins in sql?
- Types of joins
- Inner join
- Outer join
- Left join
- Right join
- Full join
- Cross join
- MS SQL Training Concepts

Subquery

- What is mean by subquery?
- Types of subqueries
- Independent subquery
- Co related subquery

SET operations

- Types of set operations in sql
- Union

- Union all
- Intersect
- Except
- Pre-conditions to use a set operator
- Difference bw union and union all?

Temp tables

- How to create temp tables
- Types of temp tables
- Local temp table
- Global temp table
- Difference between local and global temp table?

Table variables

- How to define and create table variables
- Difference between temp tables and table variables in sql server.?

Ranking functions

- Types of ranking functions
- Row_number()
- Rank()
- Dense_Rank()
- Ntile()
- Pre-Conditions to use ranking functions

Views

- What are views in sql ?
- Usage of views
- Limitations of view in sql

Stored procedures

- Definition- stored procedure
- Types of SP
- System stored procedure
- User defined stored procedure

MS SQL Training Concepts

- How to create execute and drop stored procedure/
- Advantages of SP

Functions

- Definition- functions
- Types of functions
- Scalar functions
- Inline table valued functions
- Multi statement table valued functions
- Deterministic and non-deterministic functions
- Difference between in line and multi statement functions?

Triggers

- Definition – triggers
- Types of triggers
- DML triggers
- DDL triggers
- Logon triggers
- How to enable and disable a trigger
- Magic tables in SQL Server

Cursors

- Definition –cursors
- Types of cursors
- ANSI cursor (or) Forward cursor
- Enhanced cursor (or) Scroll cursor
- Life cycle on a cursor

Indexes and statistics

- What is meant by index?
- Types of indexes?
- How to create statistics?

Execution plan and performance tuning concepts

- How to identify the estimated and actual execution plan?
- Important concepts in execution plan

- Index fragmentation
- Rebuild index
- Reorganize index

Linked servers

- Linked server creation and its usage

Backup and Restore:

- Types of back up files
- How to take back and restore in sql?
- How to automate these activities?

MS SQL Training Concepts

SQL Profiler:

- Introduction to the sql server agent
- How to capture different events in sql profiler

SQL Agent

- Job creation
- Job configuration and scheduling of the job
- Error handling in SQL Agent

Import and export:

- How to do import and export in SQL server?

Manual Testing

- Introduction to testing
- Verification vs validation
- Types of Applications
- Probabilities of getting an error in an application
- SDLC – Software Development Life Cycle
  - Waterfall
  - Prototype
  - Spiral
  - Incremental(Agile methodology and Scrum Framework)
  - V-Model
- Advantages and Disadvantages of each software development life cycle

- Principles of Testing
- STLC –Software Testing Life Cycle
- Difference between Test case, Use case and Scenario.
- How to prepare test plan and strategy
- How to prepare a test case template?
- Difference between Error, bug, defect, and failure
- Test Case Design Technique
- Boundary Value Analysis
    - Equivalence Partitioning
    - Decision Table
    - State Transition Diagram o Use Case Testing
- Bug Life cycle
- How to Prepare the Bug template?
- Bug Tracking tool
- Types of Testing
- Difference between Static and Dynamic testing
- Difference between Functional and Non-functional testing
- Black box testing and its types
- White box testing and its types
- System Integration testing vs. User Acceptance Testing
- Entry Criteria and Exit Criteria
- Test Environment and Test data preparation
- Flow graph notations
    - Statement coverage
    - Branch Coverage
    - path coverage
    - Cyclometric Complexity
- Integration testing o Big Bang Integration
- Incremental Approach – Top-Down, Bottom-up, and Hybrid
- Requirement traceability matrix

Cloud Optional (Azure / AWS)

AWS Solutions Architect

Cloud Basics:

This session deals with the basics of cloud computing. Significant features of the cloud when compared with on-premises. You will get a good understanding of the difference between public, private and hybrid cloud environments. Discuss various cloud services like

- IAAS(Infrastructure As A Service)
- PAAS(Platform As A Service)
- SAAS(Software As A Service)

Introduction to AWS:

Learning Objective:

Amazon web service is one of the leading cloud providers available in the market. AWS provides a variety of services for your business and helps you get through digital transformation for the future. You will be learning the AWS global infrastructure like Regions, Availability Zones and Edge Locations. Overview of services provided by AWS such as,

- Compute
- Storage
- Database
- Networking
- Security
- Applications

End of this session you will be having a good understanding of cloud concepts and its services and the Impact of AWS in a cloud environment.

Hands-on:

- Open AWS free tier account.
- Setup payment methods and billing preferences.
- Set alarm for free tier usage

Virtual Private Cloud:

Learning Objective:

Logically isolated network devices (instances) from other AWS resources called VPC. You can provide your own private IPs for your instances. Your instances inside VPC will be interconnected together through VPC. Basic components of VPC to create a basic infrastructure is follows

- VPC
- Subnets
- Route Tables
- Internet gateway

End of this session you will have clarity of AWS regions, Availability Zones, and Edge locations. Also, you will learn CIDR concepts to get a better understanding of IP ranges.

Hands-on:

- Create and configure VPC, Subnets, Internet gateway, and Route Tables.
- Create NAT Gateway, NAT instances, VPC Peering, Endpoints.
- Different between NACL and Security groups

EC2:

Learning Objective: EC2 provides a scalable computing capacity for your business. AWS has a wide range of instance family (type) include Memory-optimized, CPU optimized, Networking, etc. You can create instances using Amazon Machine Images, which is preconfigured by Amazon. AWS lets you create your own AMI and you can choose from any AMI available in Amazon Marketplace as well. Instances are available in many options,

- On-demand
- Reserved
- Spot Instance
- Scheduled instance

As a solution Architect, you will be able to give a solution for your client to choose the perfect compute service for their business needs.

Hands-on:

- Create and launch different types of operating systems like Linux, Windows and connect them.
- Create AMIs and Snapshots and launch instances using them.

- Create additional Volume and attach it with your instance.

Storage:

Learning Objective:

There are many storage options available in AWS as per your needs from short term to long term. In Amazon S3 you can store an unlimited amount of data. S3, EBS, EFS, FSx, Glacier, and DeepGlacier are some major storage options available in AWS. In this session, you will be learning storage classes available in S3 and its features, cost, etc. You will get a brief knowledge of storage classes so that you can suggest your client more cost-effective. Storage classes are,

- Standard
- Standard Infrequent Access
- Standard Intelligent Tier
- S3 Glacier
- S3 Glacier Deep

Hands-on:

- Create and attach EFS with instances
- Create S3 and apply lifecycle policy and replicate the bucket into another bucket.
- Create static Webhosting

Load balancing and Autoscaling:

Learning Objective:

Load balancing and Autoscaling are the significant features available in a cloud environment. A load balancer is to split the traffic between servers and trigger the auto-scaling when needed. The load balancer keeps checking the server's health in a proper time interval and takes the action according to it. There are three types of load balancers available,

- Classic Load balancer
- Application Load balancer
- Network Load balancer

End of this session you will understand the load balancer concept. You will have an idea to use a suited load balancer for your application.

Hands-on:

- Create a Load balancer and attach targeted instances into it.

- Create an auto-scaling group and increase the instances when CPU utilization is high, and decrease the instances when CPU utilization is low.

Route 53:

Learning Objective:

Route 53 is a DNS service available in AWS. You can register your domain here. Create a recordset to host your website or application. Understand the routing policies available in AWS and apply which is suitable for your environment. Routing policies are

- Simple
- Weighted
- Geological
- Latency
- Failover

Hands-on:

- Register your Domain name
- Create recordset and apply a routing policy
- Configure Health check for your Load balancer or instance.

Cloud Front:

Learning Objective:

Cloud front is also called Content Delivery Network. Cloud front creates a distribution (cache) for your website or Application at your nearest edge location. So the latency would be less for the next time user. You can block a particular country people to view your website. Analyze your website views from which OS, Browser, and users.in this session you will be engaged into the following topics,

- Origins and Origin groups
- Behaviors
- Restrictions
- Invalidations
- Create distribution for your website. Block a few country users viewing your website. Invalidate the previous cache and create a new one.

IAM:

Learning Objective:

IAM is to control your AWS resources most securely by limiting AWS users and other Services. You will be learning policies and the role and their major impact on the resources. Configure password policies and activate MFA. In this session, you will have knowledge on,

- Users
- Groups
- Roles
- Polices

Hands-on:

- Create users and add a user to the group. Attach the customized policy to that group.
- Perform cross-account access using STS

Relational Database Service:

Learning Objective:

RDS is a SQL based fully managed Database service. Users don't need to worry about storage scaling, patching, backups, and maintenance. Replication costs you less and implementation is simple. RDS supports the following database engine,

- Aurora
- MySQL
- MsSQL
- MariaDB
- PostgreSQL
- Oracle

Hands-on:

- Create a database with replication on another Availability zone. Configure auto-scaling, daily backup, and auto-upgrade.
- Connect your database through database client and insert tables and contents into it. Apply some quires to retrieve data.

DynamoDB:

Learning Objective:

DynamoDB is a NoSQL database service available in AWS. It is a replacement for MongoDB. It can handle 20 million requests per second and also can handle over 10 trillion requests per day. In-memory caching, Backup and restore are the significant features of DynamoDB. They are mainly used for gaming applications and IoT Applications. You will be learning the following topics,

- Scaling through performance
- Server less environment
- Microservices

Hands-on:

- Create and insert contents into the DynamoDB table.
- Configure and manage multi-region replication.

Monitoring (Cloud Trail & Cloud Watch):

Learning Objective:

Cloud trail keeps the logs for every activity happening into your AWS account. It does have the last 90 days activity by default. In this session, you will be learning the different types of events available in the cloud trail and its example.

- Management event
- Data event
- Insight event

Cloud Watch:

Cloud Watch is monitoring to fetch the logs and perform some action accordingly. There are two types of monitoring options available. Basic and detailed monitoring. You will learn to create logs and metrics and events. It triggers some action when event occurs. Topics involved in cloud watch are,

- Events
- Logs
- Metrics
- Alarms

Hands on:

- Create a cloud watch event to start a new instance when CPU utilization is high.

- Set alarm for the events and send notification

Application Services:

Learning Objective:

There are many application services available in AWS. SNS is the most commonly used application service. Simple Notification Services is to generate email notification and send it to the subscribers. Another application service we use is SQS. It is a replacement for the Microsoft queuing service. Application services covered in this session are as follows,

- Simple E-mail Service
- Simple Queue Services
- Simple Notification Service

Hands on:

- Create and send notification to the users
- Create and send messages using standard and FIFO messaging queue

Design and Architecture:

Learning Objective:

As a solution Architect you will be able to give a solution for your client to create, implement and improve your cloud infrastructure by using the five pillars of well architect tools.

- Operational excellence
- Reliability
- Security
- Cost optimization
- Performance efficiency

Microsoft Azure

Introduction to Cloud Computing

Objective:

In this module, you will get a basic understanding of Cloud Services and introduction to Azure

- What is Cloud Computing?
- Need for Cloud

- Characteristics of Cloud
- Categories of Services
- Providers
- Azure
    - Architecture
    - Services
    - Regions
    - Subscriptions management

Hands-On:

- Create an Azure Free Trial Account

Azure VMs and Storage Accounts:

Objective:

In this module, you will get a deep understanding of Virtual Machines & Storage Accounts in Azure

- VM Architecture
- Deploy VMs
- Create Storage Accounts for VMs
- Create a VM with Unmanaged Storage
- Manage VM Disks
- Configure Shared Storage
- Clean Up
- Supported Workloads

Hands-On:

- Create a VM and Storage Account in Azure

Configuration Management, Automation, and Debugging:

Objective: In this module, you will get a deep understanding of working in Azure using Commands

- Azure CLI
- Azure PowerShell
- Azure Cloud Shell
- VM Agent and Extensions
- ARM Templates
- PowerShell DSC

- Deploy and Enable Debugging for VM in Dev

Hands-On:

- Create VMs using PowerShell and ARM Templates

Networking in Azure:

Objective:

In this module, you will get a deep understanding of Networking Concepts, IPs and Firewall in Azure

- Virtual Network Architecture
- Create a Virtual Network
- Create a Network Security Group
- Deploy VM to the Virtual Network
- Configure DNS
- Configure Static IP
- User-Defined Routing
- VNet Peering
- Express Route and Site-to-Site VPN

Hands-On:

- Create a Virtual network, configure firewall and deploy a VM to the network

Scaling in Azure:

Objective:

In this module, you will get a deep understanding of Scaling and how to enable Scaling in Azure.

- What is Scaling in Cloud?
- Vertical Scaling
- Horizontal Scaling
- Scale Set
- Auto Scaling
- Horizontal Scaling

Hands-On:

- Create VM Scale Set and Enable Auto Scale for VM Scale Set

Monitoring in Azure:

Objective:

In this module, you will get a deep understanding of Monitoring in Azure

- Azure Monitor
- What is Log Storage Account and why it is needed?
- Boot Diagnostics
- Guest OS Diagnostics
- Host Metrics
- Creating Alerts
- Activity Logs

Hands-On:

- Enable Logging and Alerts for VMs

High Availability:

Objective:

In this module, you will get a deep understanding of Availability in the Cloud.

- What is Availability and how it is achieved?
- High Availability Architecture
- Availability Zones
- Availability Sets
- Create and Configure a Load Balancer
- Application Gateways
- Create VMs to work with Load Balancer
- Understanding N-Tier Application Architecture

Hands-On:

- Create VMs and manage the traffic via Load Balancer

Storage in Azure:

Objective:

In this module, you will get an understanding of Storage Options and Services offered by Azure.

- Blob Storage
- Queue Storage
- Table Storage
- Azure Storage Explorer
- Content Delivery Network
- Azure SQL

- Redis Caching
- Cosmos DB

Hands-On:

- Create and manage Blob, Queue and Table Storage

Azure AD:

Objective:

In this module, you will get an understanding of Active Directory Services of Azure

- Users
- Groups
- RBAC
- App Registration and Service Principal
- Azure AD Connect
- Azure B2B
- Azure B2C
- Conditional Access Policies
- Multi-Factor Authentication

Hands-On:

- Add users and configure MFA

Azure Service Bus Messaging:

Objective:

In this module, you will get an understanding of the Service Bus of Azure

- What is a Service Bus?
- What are Queues?
- What are Topics?
- Event Grid and Event Hubs
- Push Notification Services

Hands-On:

- Create a Service Bus and Configure Queues and Topics

Azure Key Vaults:

Objective:

In this module, you will learn about Key Vaults in Azure

- What is a managed identity?
- Key Vaults
- App Secret
- App Password

Hands-On:

- Enable managed identity for a VM and create Key Vault

Web & Mobile Services:

Objective:

In this module, you will learn about Web Services in Azure

- PaaS and Azure Web App
- Functions
- Web-hooks
- Web Jobs
- Logic Apps
- Mobile Apps
- Microservices architecture

Hands-On:

- Create a Web site and configure it.

Projects:

1. Create a Highly Available Web Server Architecture 1. Create a Virtual Network
    a. Create a Web Server Subnet
    b. Create a Load Balancer
    c. Create 3 Web Servers and configure them with Load Balancer
2. Build a Logic App to Read any Twitter handle and email the tweets