Sathya™
TECHNOLOGIES

JOIN THE PYTHON REVOLUTION AND CODE YOUR DREAMS!

🌐 sathyatech.com

25 YEARS OF ACADEMIC EXCELLENCE

CORE PYTHON

INTRODUCTION TO PYTHON

- What is python
- Python use cases
- The Birth of Python
- Python Timeline
- Features of Python
- Versions of Python
- Python distributions

INTRODUCTION TO SCRIPT

- What is script
- What is program
- Types of script
- Types of programming languages

THE PYTHON ENVIRONMENT

- Installation of python on windows, MAC,LINUX
- Setting path for python
- Python Documentation
- Getting Help
- Python Command line shell, Editors and IDE's
- Basic Syntax
- Running Python Script on WINDOWS
- Running Python Script on LINUX
- Working with IDLE

GETTING STARTED

- Keywords v Data Types
- Static Data +Types versus Dynamic Data Types
- Fundamental Data Types
- Collection Types
- Number Systems
- Mutable objects versus Immutable Objects
- Variables
- Naming Conventions
- Print(),Type()and Id() Functions
- Input and raw_input() functions
- Type Conversation functions

OPERATIONS

- Arithmetic Operators
- Relational Operators
- Logical Operators
- Assignment Operators
- Short Hand Assignment Operators
- Bitwise Operators
- Membership Operators
- Identify Operators
- Precedence of Operators
- Evaluating expressions

FLOW CONTROL

- About Flow Control
- Elements of Flow Control
- Block / Clause
- Conditional Statements

>Simple if

>if....else

>if....elif...else

- Looping Statements

>While loop

>while ....else

>for loop

>for..else

>using range() in for loop

>working with infinite loops and nested loops

- Break Statement
- Continue Statement
- Pass Statement

STRING HANDLING

- What is String?
- Single –quoted string Literals
- Triple-quoted string literals
- String Indexing
- String Slicing
- Working with String Functions
- Working with String Methods
- Reversing a String
- String multiplication and concatenation

PYTHON
FULL STACK

Sathya TM
TECHNOLOGIES

COLLECTION LIST:

- Lists are mutable
- Getting to Lists
- List indices
- Traversing a list
- List operations
- List Slices
- List methods
- Map, Filter and Reduce
- Deleting elements
- Lists and Strings

TUPLE:

- Advantages of Tuple over list
- Packing and Unpacking
- Comparing Tuples
- Creating nested tuple
- Using tuples as keys in dictionaries
- Deleting tuples
- Slicing of Tuple
- Tuple Membership Test
- Built-in function with Tuple
- Dotted Charts

SETS:

- How to create a set?
- Iteration Over sets
- Python set methods
- Python set Operations
- Union Of sets
- Built –in Functions with set
- Python Frozenset

DICTIONARY

- How to create a dictionary?
- Python Hashing
- Python Dictionary Methods
- Copying Dictionary
- Updating Dictionary
- Delete keys from the dictionary
- Dictionary items () Method

Functions

- Defining a Function
- Calling a function
- Function Parameters
- Types of Parameters

>default parameters
>non default parameters
>keywords arguments
>non key word arguments
>arbitrary arguments

- Return statement in functions
- Handling return values
- Global variables and Local Variables
- Scope of global variables and local variables
- Passing collections to a function

LAMBDA FUNCTIONS

- Lambda functions / anonymous functions
- Filter()
- Map()
- Reduce()

MODULES

- What is Module?
- Types of Modules
- The import statement
- Module Aliases/renaming a module
- From... Import
- Reloading a Module
- Built in properties of a Module
- Dir() Functions
- Creating user defined modules
- Module search path
- Command line arguments
- Working with pre defined Standard modules (Math,Random,Datetime, Os, Sys,String,...)
- Packages
- Introduction to packages
- ___init__.py file
- Defining packages
- Importing from packages
- Defining sub packages
- Importing from sub packages
- Difference between 2.X and 3.X packages

PYTHON
FULL STACK

Sathya™
TECHNOLOGIES

ADVANCE PYTHON

OOPS Concepts

- Introduction to OOPS programming
- OOPS principles
- Encapsulation
- Defining Classes
- Creating Objects
- Class Variables
- Instance Variables
- Parameters
- Local Variables
- Defining Methods
- Difference between functions and methods
- Instance Methods
- Static Method
- Class Method
- Difference static and class methods
- Constructors
- Destructors
- Inheritance
- Types of inheritances
- Polymorphism (overloading & over riding)
- Super() Statement
- Built in properties of class
- Inner classes

EXCEPTION HANDLING

- Syntax Errors
- Runtime Errors
- What is BUG?
- What is Exception?
- Need of Exception handling
- Predefined Exceptions
- Predefined Exceptions Hierarchy
- Try, except and finally clauses
- Named except block
- Default except block
- Handling Multiple Exceptions
- Nested try, except and Finally blocks
- User Defined Exceptions
- Raise, assert statements.

FILE HANDLING

- What is file?
- Opening a file?
- Reading data from a file
- Writing data to a file
- Closing a file
- Working with the methods of file object
- Replacing the content of file
- Working with Directories
- Handling IO Exceptions

REGULAR EXPRESSIONS

- Introduction to regular expression
- Simple character matches
- Special characters
- Characters Classes
- Quantifiers
- Forming regular expressions
- Matching at beginning or end
- Greedy matches
- Compiling regular expressions
- Grouping
- Match Objects
- Match(),Search()and sub()functions
- Matching versus searching
- Splitting a string
- Replacing text
- Flags

DATABASE ACCESS

- Introduction to RDBMS
- Cursor object
- Executing SQL queries
- Executing SQL queries with bind variables
- Execution of PL/SQL procedures and functions
- Installation of mysql database
- Creating databases in mysql
- Creating users and assignment privileges to the users in mysql
- Installation of mysql python modules
- Establishing connection with mysql
- Closing mysql database connections
- Execution of insert,update,delete and select queries
- Handling db errors

PYTHON
FULL STACK

ADVANCED CONCEPTS

- Python Iterator
- Python Generator
- Python Closure
- Python Decorators
- Python Property
- PIP
- Installation of external modules using PIP
- Working with csv,xml,and jsonfiles
- Debugging using IDE
- Test cases implementations

DATA ANALYTICS

- Introduction to big data
- Pandas
- Numpy Matplotlib
- Hadoop word count example

PYTHON WEB DEVELOPMENT

- Introduction to web application
- Architecture of web application
- Introduction to DJANGO framework
- Introduction to FLASK framework

D JANGO

- Introduction to Web Development
- Introduction to DJANGO
- MVT pattern
- Flow in Django Project
- Virtualenv
- Installing Django
- Creating a Django project
- What is a view
- Function based views
- Class based views
- Http Response
- URL route configuration
- CSRF
- HTML AND Django
- About templates
- Django template syntax
- Static Files
- Loading templates
- Context in Templates
- Sending Data
- Forms overview
- GET and POST
- The form Class
- Processing the form
- Validation
- Forms in templates
- Database Setup
- Creating Models
- Activation Models
- Activating Models
- Admin site
- One-to-Many Relation
- Many to one Relation
- One-to-one Relation
- Pagination
- Generic Views
- List Vies
- Detail View
- Create View
- Update View
- Delete View
- Cookies in Django
- Django Session Framework
- Caching
- File Uploading & Downloading
- Deployment on Apache
- Sending Email
- Other Email Functions
- Security with user Authentication
- Automated Testing
- When to create tests
- Using Django’s test Framework
- Using the test Client
- Running tests
- Checking Code Coverage

Django

Introduction to Django

- Introduction
- About Django
- Django Components
- Installing & Configuring Django Components
- Django Pre-Requisites
- Downloading & Installing Django
- Choosing a Database
- Creating a New Project

Module 2: Generating Simple Django Views

- Generating Simple Django Views
- About View Functions
- Using Django's HttpResponse Class
- Understanding HttpRequest Objects
- Using QueryDict Objects

Module 3: Configuring URLconf's

- Configuring URLconf's
- About URLconf
- Regular Expressions
- Expression Examples
- Simple URLConf Examples
- Using Multiple URLConf's
- Passing URL Arguments

Module 4: Django Templates

- About Templates
- Template Fundamentals
- Creating Template Objects
- Loading Template Files
- Filling in Template Content (Context Objects)
- Template Filters Template Tags
- More on For Loops
- Template Inheritance
- Easy Rendering of Templates
- RequestContext Processors
- Global Context Processors

Module 5: Forms

- Getting Data From the Request Object
- A Simple Form-Handling Example
- Making a Contact Form
- Tying Form Objects Into Views

Module 6: Database Models with Django

- About Database Models
- Configuring Django for Database Access
- Understanding Django Apps
- About Django Models
- Defining Django Models
- Understanding Model Fields & Options
- Table Naming Conventions
- Creating A Django Model
- Adding the App to Your Project
- Validating the App
- Generating & Reviewing the SQL
- Adding Data to the Model
- Primary Keys and the Model
- Simple Data Retrieval Using a Model
- Understanding QuerySets
- Applying Filters
- Specifying Field Lookups
- Lookup Types
- Slicing QuerySets
- Specifying Ordering in QuerySets
- Common QuerySet Methods
- Deleting Records
- Managing Related Records
- Retrieving Related Records
- Using Q Objects
- Creating Forms from Models

Module 7: Using the Django Admin Interface

- Using the Django Admin Interface
- Enabling the Admin Interface
- Creating an Admin User

Module 8: Access Control with Sessions and Users

- Access Control with Sessions and Users
- Cookies & Django
- The Django Session Framework
- Sessions in Views
- Session Tuning
- Installing Django User Authentication
- Using Authentication in Views
- Login and Logout
- Building your Own Login/Logout Views
- Authentication Decorators
- Adding & Deactivating Users Asynchronous Messaging
- Managing Permissions

PYTHON
FULL STACK

Sathya™
TECHNOLOGIES

UI TECHNOLOGIES

HTML 4 HTML BASICS

- WHAT IS HTML
- Features of HTML
- Tags: paired tags (vs) Unpaired tags
- List of tags
- Installing Editor and Browse

HTML Structure

- Syntax of html
- Head (vs) Body
- First example in HTML

HTML Document Basic

- Heading
- Texting formatting tags
- Preformatted text
- Lines

Abbreviations, Images and Hyperlinks

- Meta tag
- List Tags
- Ordered List
- Unordered List
- Tables
- Forms
- What is Form
- Importance of Form in Realtime
- Input tag , types and altributes
- Dropdownlist and Option Group
- Fieldset and Lebel
- Textarea ,Label
- Submitting the form to server
- DIV and Span

CSS 2

- CSS Basics
- What is CSS
- Need of CSS in real web sites
- Syntax of CSS
- What is Selector
- What is property
- ID Selector
- Class Selector

Color properties

- Color
- Background Color
- Types of Color: Named colors, Rgb colors, Hexadecimal colors
- Front Properties
- Text Properties
- Background Properties
- Bullets and Numbering
- DIV and its css properties
- Importances of Box Model
- Padding properties
- Border Properties
- Margin properties
- Advanced
- Opacity
- Display
- Visibility
- Overflow

Position

- Static
- Absolute
- Relative
- Fixed
- Z-index
- Hyperlink Styles
- a:link
- a:hover
- a:visited
- a:active

Types of Style Sheets

- Internal Style Sheet/Embedded Style Sheet
- Internal Style Sheet
- External Style Sheet

UI TECHNOLOGIES

CSS Selectors

Style Precedences

- Default Style precedence
- Limport
- Menubar inrealtime
- Table style in realtime
- Static page template
- Header
- Nav
- Container
- Leftcol
- Pagecontent
- Footer
- Responsive web design
- Need of RWD in realtime
- Extra-small,small,medium,and Large devices
- Media queries in CSS file
- Viewportmeta tag

Java Script 5

- Introduction to Java script
- Programming fumdamentals
- Operators
- Control statements
- Functions
- Arrays
- JavascriptOOP(Advanced JavaScript)
- What is Object
- Properties
- Methods
- Object literals
- Constructor functions
- Demo: creating and calling objects in different ways
- Manipulating objects
- Reading object keys
- Object array
- Prototype
- Inheritances

JSON

- What is JSON
- Object literal(vs) JSON
- Importance of JSON in realtime
- JSON.stringify()
- • JSON.parse()

Type conversion

- String functions
- Date functions
- Adavanced
- Noscript
- Closures
- Hoisting
- **DOM Manipulation**
- What is DOM
- Window object
- Document Object
- Element Object
- Event Handling
- **Validations**
- What is validation
- Need of validation in realtime
- Submit event
- Required field validation
- Using regular expression
- Types of Javascript
- Internal Javacsript/Embedded Javascript

Javascript 6&7/ Ecma Script 6&7

- Fundamentals
- What is EmcaScript
- EcmaScrpit(vs)JavaScript
- Setting-up the Environment

OOP

- Classes
- Constructors
- Parameter Less Constructor
- Properties
- Methods
- Summary

Search suggestions & Calculations

- Detais
- List
- Output

Storage

- Localstorage
- Session storage
- Geo location
- Canvas and SVG
- 7)CSS3
- CSS 3 Basics
- What is CSS 3
- New features of css3
- CSS 3 first example

Box model properties

- Resize
- Word wrap
- Border Radius
- Box shadow
- Text shadow
- Transformations
- Transitions
- Animations by using theKeyframes
- Creating basic page contact
- Tables
- Tables styles
- Tables background colour
- Buttons
- Inheritance
- Method overriding

Other Concepts

- Set and get methods
- DefaultArguments
- Arrow functions
- Let
- Const
- Rest
- Destructuring
- Multiline strings

HTML 5

- Html 5 Basic
- What is html 5
- New features of ntml 5
- Advantages of html 5
- Syntax of html 5
- Deprecated elements andattributes
- New doctype for html 5
- New input types
- NEW INPUTATTERIBUTES
- MULTIMEDIA
- Audio Tag
- Video Tag
- Audio formats and video formates

SEMANTIC AND STRUCTURAL TAGS

- Article
- Header
- Nav
- Section
- Aside
- Footer

PROGRESS BAR

- Meter
- Progress

IMAGES

- Figure
- Figcaption

Collapsible

- Datails
- SuButton colours
- Table sizes
- Glyphicons
- Visibility
- Show
- Hidden

Images

- Rounded

PYTHON
FULL STACK

Sathya™
TECHNOLOGIES

- Circle
- Thumbnail
- Jumbotron
- Paging
- Panels
- Panels heading
- Pancel body

Nav Bar

- Navbar
- Navbar inverse
- Navbar fixedtop
- Navbar header
- Navbar nav
- Breadcrumbs
- Dropdown
- Progressbar
- Tabs
- Forms
- Model
- Grid system(responsive web design)
- Extra small
- Small
- Medium
- Large

Angular JS

Introduction

1.Introduction
2.What is Angular
3.Architecture of Angular Apps
4.Setting Up the Development environment
5.Your First Angular App

Angular fundamentals

1.Introduction
2.Building Blocks of Angular Apps
3.Controllers
4.Templates
5.Directives
6.Services
7.Filters
8.Dependency Injection

Displaying Data and Handling Event

1.Introduction
2.Expressions
3.Event Binding
4.Two-way Binding

Modules

1.Creating a module
2.Using module in the application
3.Creating a module with dependencies
4.Creating a contacts for the module

Directives

1.What is a directive
2.Built In directives
3.Creating the custom directive
4.Compile , link and transclusion
5.Workking with isolated scope

Controllers

1.What is a directive
2.Creating the Controller and using in the application
3.Creating a model and function inside a controller
4.Configuring simple events with function

Scopes

1.What is $scope
2.What is $rootScope
3.$cope vs $rootscope
4.Change detection and watchers

Filters

1.What is filter
2.Built in filters usage in the application
3.Creating a custom filter

Services

1.What is service
2.built in services in the application
3.creating the custom service
4.factory vs provider vs services

Http

1.calling the server side apis by using $http
2.getting the data from the server
3.ALL CRUD operations by using the $http
4.Working with promises

Error Handling

1.Identifying the simple errors and trouble shooting

2.Configuring the special error handiling functions

3.Displaying generic error mssages

Forms

1.Configuring the Angular Forms with input controls

2.Configuring the validations for the input controls

3.Use of novalidate

4.Configuring project specific error message

5.Routing

1.Configuring single page application

2.Configuring single page application in Real time application

3.Configuring the routes and route parameter

1.Configuring the navigation using the location service

2.Enabling the html5 specific routing

Integration

1.Angular JS in the JAVA Applications

2.Angular JS in the.net Applications

3.Angular JS in the NODEJS Applications

4.Anglar JS in the Python Applications

5.Angular JS in the PHP Applications

React JS

Introduction to React JS

- Importance of React Js in web_Applications
- Features
- History of ReactJS
- Building single-page applications and mobile applications
- React ->a JavaScript library
- Need of virtual DOM and its Mechanism on performance of applications that rely on frequent updates

Benefits of React JS

- Declarative
- Reusable components
- Fast rendering
- JSX
- Server-side rendering
- Strong developer community

Virtual DOM

Cons of React JS

Prerequisites for Learning ReactJS

- Basic JavaScript syntax
- Asynchronous JavaScript
- The DOM
- HTML and CSS
- ES6+

ECMAScript 6

- Arrow functions
- Classes
- Promises
- The spread operator
- Similarities and differences between JavaScript and ReactJs

ReactJS - Environment Setup

ReactJS - JSX(JavaScript XML)

- Importance of jsx files and its working
- use className instead of class in JSX
- Concept of modular programming

ReactJS - Components

- Class level components
- Functional Level components
- Efficiency code of using Functional components

ReactJS - State

- Purpose of State
- Mutability of State
- Importance of setState(),useState() hoook and render() hook

ReactJS - Props Overview

- Terminology of passing data from a parent component to a child component using props
- Concept of Immutability in props

ReactJS - Props Validation

ReactJS - Component Life Cycle

ReactJS - Events

ReactJS - Refs
ReactJS - Keys
ReactJS - Router
ReactJS - Flux concept
ReactJS - Higher order Components
ReactJS - Best Practices

Mongo DB

Module 1 – Design Goals, Architecture and Installation

Topics:

- Understanding Base Concepts of Database
- Types of NoSQL Database, and NoSQ vs. SQL Comparison, ACID & Base Property
- Overview of MongoDB, Design Goals for MongoDB Server and Database, MongoDB Tools
- How to modularize code by separating routes
- Installation/Running MongoDB on various platforms Windows, Linux, MAC OS, etc.

Module 2 – CRUD Operations

Topics:

- MongoDB Development Architecture
- MongoDB CRUD Introduction
- Concern Levels, Journalling etc.
- Distributed Read & Write Queries
- MongoDB Datatypes
- MongoDB Production Architecture
- MongoDB CRUD Concerns
- Cursor Query Optimizations, Query Behaviours in MongoDB
- MongoDB CRUD Syntax & Queries (Live Hands on)

Module 3 – Schema Design and Data Modelling

Topics:

- Data Modelling Concepts
- Type of Data Modelling
- Data Model Examples and Patterns
- Model Tree Structures
- Model Relationships between Documents
- Use Case of Data modelling
- Type of Data Modelling
- Analogy between RDBMS & MongoDB Data Model, MongoDB Data Model (Embedding & Linking)
- Challenges for Data Modelling in MongoDB
- Model Specific Application Contexts

Module 4 – Administration

Module 5 – Scalability and Availability

Topics:

- Introduction to Replication
- What is Replica Set and Master Slave Replication?
- Introduction to Sharding
- Concepts around Sharding, What is shards, Key
- How to setup a Sharding
- Concepts around Replication
- Type of Replication in MongoDB
- How to setup a replicated cluster Managing Replica Sets etc.
- Confg Server, Query Router etc.?
- Type of Sharding (Hash Based, Range Based etc.), and Managing Shards

Module 6 – Indexing and Aggregation Framework

Topics:

- Index Introduction
- Index Types
- Index Creation
- Aggregation to Introduction
- Type of Aggregation (Pipeline, MapReduce & Single Purpose)
- Index Concepts
- Index Properties
- Index Tutorial
- Indexing Reference
- Approach to Aggregation
- Performance Tuning

Module 7 – Application Engineering and MongoDB Tools

Topics:

- MongoDB Package Components
- MongoDB Limits and Thresholds
- MMS (MongoDB Monitoring Service)
- HTTP and Rest Interface
- Integration of MongoDB with Hadoop and Data Migration MongoDB with Hadoop (MongoDB to Hive).
- Confguration File Options
- Connection String URI Format/ Integration of any compatible tool with MongoDB API and Drivers for MongoDB

Module 8 – Project, Additional Concepts and Case Studies

Topics:

- Security Introduction
- Security Tutorial
- Integration of MongoDB with Pentaho
- Integration of MongoDB with Python
- Integration of MongoDB with GUI Tool Robomongo
- Security Concepts
- Integration of MongoDB with Jaspersof
- Integration of MongoDB with Hadoop/Hive
- Project on MongoDB and Python

PYTHON FULL STACK