NAV BHARAT
COMPUTER EDUCATION
H.O. : 13/6C, Nai Sarak, Shastri Nagar, Meerut
(M) 9837991954, 9760091954
B.O. : C-57/6 Kunj Vihar, Jagriti Vihar, Meerut
(M) 0121-2767699, 9258551962

Full Stack Web Development Using Python

HTML

- INTRODUCTION
  - Parts of an HTML Document
  - Version Information
  - Head Section
  - Meta Information
  - Favicons
  - Body Section
  - List
  - Table
  - HTML Forms
  - Anchors and Images
  - Video, Audio
- ADVANCE HTML
  - Introduction
  - HTML5 History
  - New Features and Groups
  - Structure of HTML5 Document
  - Power of HTML5 and Features
  - Semantics and Block
  - Level Elements
  - HTML5 Forms
  - HTML5 Multimedia
- HTML5 Graphics

CSS

- CORE CSS
  - Introduction
  - CSS Basics
  - CSS Syntax
  - CSS Versions

- CSS Id and Class
- CSS Styling
- Styling Backgrounds
- Styling Text
- Styling Fonts
- CSS Borders

- ADVANCE CSS
  - Introduction
  - CSS3 Modules
  - Position
  - Box Model
  - Background and Borders
  - Text Effects
  - 2D/3D Transformations
  - Multiple-Column Layout
  - Flex Grid

- User Interface

JavaScript

- Introduction to JavaScript
  - Syntax, Statements, Comments.
  - Syntax, Statements, Comments.
- Enabling in various browsers Popup Boxes
  - Alert, Confirm, Prompt
- Java Script (JS Basics)
  - keywords
  - identifiers
  - datatypes
  - variables and constants
  - token
- Variables Operators
  - Arithmetic
  - Assignment
  - Comparison
  - Logical
  - Conditional
- Conditional Statements
  - If
  - If...else
  - If...else If...else
  - Switch
- Loops
  - While
  - Do...while

- For
- For...in Statement
- Break
- Continue

- Function
  - Creating tuple
  - function declarations, definitions and calling
  - types of functions
  - anynomous functions
  - flat arrow functions or arrow functions
  - self invoked function (IIFE)
  - storage classification or variable scope (var let const and without keywords)
  - recursion
  - scope chain rule and lexical scoping
  - closure
- Objects
  - literal and keys
  - object functions
  - nested objects
  - Object Prototypes
  - setter and getter methods
- Set and Map Weak Map and weak Set
- Dates and Time
- OOPS
  - literal and keys
  - object functions
  - nested objects
  - Object Prototypes
  - setter and getter methods
- DOM Manipulation
  - DOM HTML
  - DOM CSS
  - event handling
  - time based events
- Events
  - Introduction
  - Mouse Events
  - Keyboard Events
  - Form Events
  - Document/Window Events
- Page redirection Built-in Objects
  - Introduction
  - Number
  - Strings
  - Arrays
  - Associative Array

- Array Properties and Methods
- Date
- Boolean
- RegEx

- Advanced JavaScript
  - Form Validations
  - Basics of Form Validation
  - Validating Radio Buttons
  - Validating Check boxes
  - Validating Select Menus
  - Validating Text areas
  - strict mode
  - closure
  - synchronous and asynchronous programming
- JS Browser Object Model (BOM)
  - JS Cookies
  - JS Window
  - JS Location
  - JS Popups
- Event propagation
  - bubbling
  - capturing
- Data fetching
  - AJAXand JSON
  - Calling API using AJAX
  - Callbacks
  - Promise
  - fetch
  - async await

ReactJS

- Typescript
  - Why Typescript
  - Basic Types
  - Class and Interfaces
  - Modules
- INTRODUCTION TO REACTJS
  - What is ReactJS?
  - What is SPA?
  - DOM vs Virtual DOM
  - Advantages & Disadvantages
  - Key Features
- ENVIRONMENTAL SETUP
  - Node | NPM
  - Installation of CLI

- Setup Project
- Directory Structure
- Code Editors
- How ReactJS Application Boot

- BASIC FEATURES OF REACTJS
  - React Concepts
  - JSX and TSX
  - Render Elements
  - Function and Class Components
  - Props and State
- Handling Events
  - Dynamic Data Rendering
  - Property Binding
- KEY FEATURES OF REACTJS
  - Conditional Rendering
  - List and Keys
  - Forms Handling
  - Forms Validations
- COMPONENT LIFE CYCLE HOOK
  - Understanding component life cycle
  - All Life cycle Hooks
- EVENT HANDLING REACT
  - Understanding React Event System
  - Passing arguments to event Handlers
- NETWORK CALL
  - Fetch
  - Axios
- CUSTOM SERVICES
  - Introduction to Services
  - Building a Service
- LOCAL DATA STORAGE
  - Local Storage
  - Session Storage
  - Cookies
- ROUTING WITH REACT ROUTER
  - Setting up React Router
  - Configuring route with Route Component
  - Making routes dynamic with Route Params
  - Working with nested routes
  - Link and NavLink
  - Redirect Routes
- UI COMPONENTS
  - Material Design
  - PrimeNG
- INTRODUCTION TO REDUX
  - Why Redux

- Install and setup
- Store
- Reducer
- actions
- Dispatcher
- High order Components
- map State To Props and map Dispatch To Props usage

- ADVANCE REDUX
  - Async Actions
  - Middleware
  - Redux Thunk and Redux Saga
- REACT HOOKS
  - Why We Need HOOKS.
  - Different Types Of Hooks
  - Using State And Effect Hooks
  - Usereducer , Useref Etc.
  - Custom Hooks
  - Rules Of Hooks
- THIRD PARTY MODULES
  - Social Login
  - Pagination
  - Search
  - Filter
  - JWT Token
  - File Upload
  - Many More
- REACTJS TESTING
  - Jest with Enzyme
- Develop a CRUD Application in REACTJS
- REACTJS Application Deployment
  - Build Application and Deployment

Core Python

- Introduction to Python
  - Why Python
  - Application areas of python
  - Python implementations
    i) Cpython
    ii) Jython
    iii) Ironpython
    iv) Pypy
  - Python versions
  - Installing python
  - Python interpreter architecture
    i) Python byte code compiler

    - ii) Python virtual machine(pvm)
- Writing and Executing First Python Program
    - Using interactive mode
    - Using script mode
        - i) General text editor and command window
        - ii) IDLE editor and IDLE shell
    - Understanding print() function
    - How to compile python program explicitly
- Python Language Fundamentals
    - Character set
    - Keywords
    - Comments
    - Variables
    - Literals
    - Operators
    - Reading input from console
    - Parsing string to int, float
- Python Conditional Statements
    - If statement
    - If else statement
    - If elif statement
    - If elif else statement
    - Nested if statement
- Looping Statements
    - While loop
    - For loop
    - Nested loops
    - Pass, break and continue keywords
    - Parallel Traversaling using zip()
- Standard Data Types
    - Int, float, complex, bool, nonetype
    - Str, list, tuple, range
    - Dict, set, frozenset
- String Handling
    - What is string
    - String representations
    - Unicode string
    - String methods
        - i) count(), find(), rfind(), capitalize(), title(), lower(), upper(), swapcase(), islower(), isupper(), istitle(), replace(), strip(), lstrip(), rstrip(), split(), partition(), join(), isspace(), isalpha(), isdigit(), isalnum(), startswith(), endswith() etc.
    - Slicing
    - String indexing and slicing
    - String formatting
- Python List
    - Creating and accessing lists

- Indexing and slicing lists
- List methods
- Nested lists
- List comprehension

- Python Tuple
  - Creating tuple
  - Accessing tuple
  - Immutability of tuple
- Python Set
  - How to create a set
  - Iteration over sets
  - Python set methods
  - Python frozenset
- Python Dictionary
  - Creating a dictionary
  - Dictionary methods
  - Accessing values from dictionary
  - Updating dictionary
  - Iterating dictionary
  - Dictionary comprehension
- Python Functions
  - Defining a function
  - Calling a function
  - Types of functions
  - What is Namespace?
  - LEGB Rule
  - global, nonlocal statements
  - Function arguments
    - i) Positional argument
    - ii) Keyword argument
    - iii) Default argument
    - iv) Non-default argument
    - v) Arbitrary arguments
    - vi) keyword arbitrary arguments
  - Function return statement
  - Nested function
  - Function as argument
  - Function as return statement
  - Decorator function
  - Closure
  - Map(), filter(), reduce(), any() functions
  - Anonymous or lambda function
  - Command Line Argument
- Modules & Packages
  - Why modules
  - Script v/s module

- Importing module
- Standard v/s third party modules
- Why packages
- Understanding pip utility

- File I/O
  - Introduction to file handling
  - File modes
  - Functions and methods related to file handling
  - Understanding with block
  - Pickle module
  - JSON module
  - Using OS module

Advance Python

- Regular Expressions(Regex)
  - Need of regular expressions
  - Re module
  - Functions/methods related to regex
  - Meta characters & special sequences
- Object Oriented Programming
  - Procedural v/s Object Oriented Programming
  - OOP Principles
  - Inheritance
  - Defining a Class & Object Creation
  - Encapsulation
  - Polymorphism
  - Abstraction
  - Garbage Collection
  - Iterator & Generator
- Exception Handling
  - Difference Between Syntax Errors and Exceptions
  - Keywords used in Exception Handling
    i) try, except, else, finally, raise, assert
  - Types of Except Blocks
  - User-defined Exceptions
- GUI Programming
  - Introduction to Tkinter Programming
  - Tkinter Widgets
  - Layout Managers
  - Event handling
  - Displaying image
- Multi-Threading Programming
  - Multi-processing v/s Multi-threading
  - Need of threads
  - Creating child threads

- Functions /methods related to threads
- Thread synchronization and locking

SQL

- SQL Using MySQL
  - Introduction to RDBMS
  - What is Relational Database Package
  - Difference between SQL & Database
  - Installing MySQL Server database
- SQL Basic
  - DDL: Create, Alter, Drop, etc.
  - DML: Insert, Update, Delete, etc.
  - DQL : Select
  - Auto_increment field
  - SQL Comments
  - SQL Aliases
  - Savepoint & rollback
- SQL Constraints
  - Not NULL, Unique key
  - Primary key, Check
  - Default, Foreign key
  - SQL Operators
  - Arithmetic operators
  - Logical operators
  - Conditional operators
  - Like, between, in operators
- SQL Clauses
  - Order by
  - Where
  - Limit/top
  - Group by
  - having
- SQL Joins
  - Inner Join
  - Left Join
  - Right Join
  - Full Join
- SQL View
  - creating view
  - updating view
  - fetching data from view
- SQL Functions
  - String functions
  - Aggregate functions
  - Date & time functions

- Stored Procedures & Functions
  - Understanding stored procedures and their key benefits
  - Working with stored procedures
  - Studying user-defined functions

Python Database Connectivity

- Introduction to MYSQL-Connector
- Installing MYSQL Server
- Understanding Basic SQL Statement
- Database Drivers and connectors
- Creating connection object
- Understanding cursor object
- Executing SQL statements using cursor
- Fetching records from cursor
- Storing and retrieving Date and Time

Django

- Getting Started With Django
  - Language v/s Framework
  - What is Django Framework?
  - Django Versions
  - Installing Django
- Understanding Django Project
  - Creating Django Project
  - Creating Django App
  - Understanding Directory Structure of Project
  - URLMapping, Views, manage.py
  - HttpRequest and HttpResponse Objects
  - Registering App in settings.py
  - Django Development Server
  - Control flow of request processing
  - MVT Design Pattern
- Django Templates (Presentation Logic)
  - Built-in Templates Tags & Filters
  - Template Variables
  - Template Inheritance
  - Building Custom template tag
  - Integration of Static Contents
    - i) How to display static images
    - ii) How to use CSS and Bootstrap
    - iii) Loading static data to template
- Understanding Django Views (Business Logic)
  - Why ths Component?

- Types of Views
  i) Function, Class Based
- How Views Interact with other components
- How to get request data in views
- How to generate response in other formats

- Understanding Django Forms
  - Why this Components?
  - Form class & Field Types
  - GET and POST methods
  - Form Validations
  - How to render django form to template
    i) as table
    ii) as paragraph
    iii) as li
  - How to use Form in views
- Django Models (ORM)
  - Why this component?
  - SQL v/s ORM approach
  - Model class and Field types
  - Understanding makemigrations and migrate
  - Performing CRUD operations
    i) Create (insert)
    ii) read (select)
    iii) update
    iv) delete
  - Fetching Records from database
    i) Understanding QuerySet
    ii) Fetching all records
    iii) Fetching records based on conditions
    iv) Fetching Data in an order
    v) Data Slicing
  - Understanding Association Mappings
    i) One to One
    ii) One to Many
    iii) Many to One
    iv) Many to Many
  - Integration with MySql database
- Django Admin interface
  - How to create superuser
  - How to access admin site
  - Registering models with admin
  - Managing users in the admin
- Django Statte and Session Handling
  - Why maintain state?
  - Understanding stateless behavior of http protocol
  - State management techniques
    i) Cookies

    - ii) URL-Rewriting (query Settings)
    - iii) Hidden form fields
    - iv) Session
  - Understanding Session in Details
    - i) Enable/Disable Session
    - ii) Get and set data with session
    - iii) Using session in template
    - iv) Session Expiry
- Django Middleware
  - Why Middleware
  - Understanding built-in middleware
  - Creating custom middleware
    - i) Function based
    - ii) Class based
  - How to enable custom middleware
- Django User Authentication
  - Working with User Objects
  - Permissions and athorization
  - Authentication in web requests
  - Managing user in the admin
  - Extending the existing User Model
- Developing Web Services (APIs)
  - What is Web API?
  - Difference between SOAP and RESTful APIs
  - Django REST Framework for developing RESTful APIs
  - Understanding JSON response
  - How to consume RESTful API
    - i) request package
    - ii) GET and POST How to parse JSON
- Miscellaneous
  - Sending email using django
- Pagination