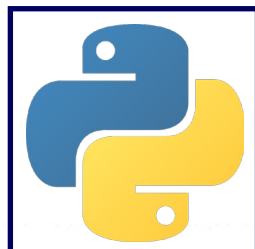




MASTER'S PROGRAM

IN FULL STACK - PYTHON

INTERNATIONAL POST GRADUATE DIPLOMA CERTIFICATE IN SOFTWARE
DEVELOPMENT & AUTOMATION ENGINEERING



What We cover From Basic To Advanced

Complete In-depth Knowledge

Html

CCS5

Java Script

Bootstrap

Mongo DB

Express

Angular

React

Node

Python

Django

Job Oriented Scenarios

Multiple Job Oriented Responsibilities Practically Explained – Gain Knowledge of 5+ Year Experienced Full Stack Developer. **400+ Hours of content**

100% Job Guaranteed Master Program

Job Opportunities



Full Stack Developer



Software Developer



Automation Engineer



Front End Developer



Backend Developer



Software Engineer



Full stack Python Developer

Our students placed in various Pay packages in **MNC's Ranging from 5 Lakhs to 25 Lakhs**. Experience Level from **freshers to 18+ Year experienced candidates**.

Mater Program Duration

6 Months Learning +
6 Months Projects,
Assessments & Assignments



Major Features of the course

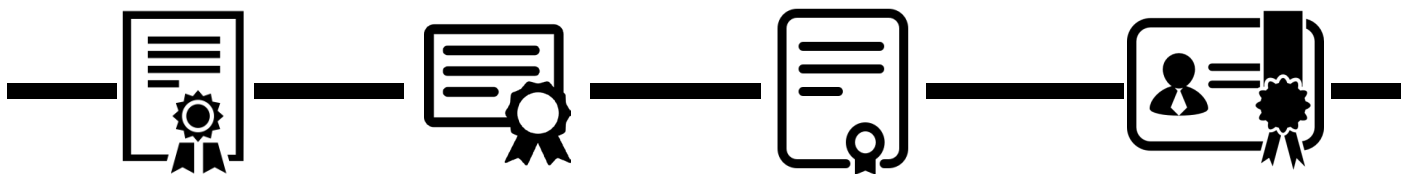
Major achievements of this program is that, This program is integrated with International PG Diploma Certificate Program in Software Development & Automation Engineering – Awarded by OTHM – UK , Ofqual Approved UK Qualification provider, (<https://www.gov.uk/government/organisations/ofqual>)

Program Include Major and Minor Projects with all portions practically deeply explained, 200+ Assignments and 50 hours of Mock interview session. You will acquire a knowledge of 5+ year experienced in Full Stack Development after this course

This program is good for freshers and up to 3 – 4 years of industry experienced professional & Those who have career gap, Experienced people, who would like to upskill & change their domain .

Certificates Achieved

You will receive 4 certificates after successful completion.



1. RCITP after 6 months

2. Internship Certificate explaining Projects undergone

3. International PG Diploma certificate in Software Development & Automation Engineering Awarded from OTHM UK - After 6 Months of successful completion of the course. You have to undergo

multiple assignments to achieve the International PG Diploma. After successful completion of assignment, you have to appear for 12 assessments, on topics covered during the course .

After fulfilling all the above criteria, International PG Diploma certificate in Software Development & Automation Engineering is awarded directly by OTHM

4. Fill any Career gap with our consulting companies who provides documentation on the projects completed

SALARY :- Avg. salary of a Full stack, Software Developer range from **2.8 Lakh and goes up to 15 Lakhs** per annum for 5+ year experienced developer

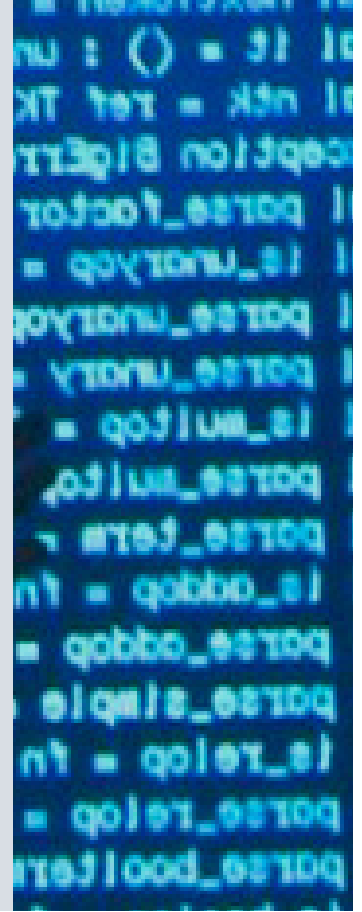
Program Highlight

- 18000+ Students Empowered till now
- Online/Classroom/Self-Paced
- 6 months Intensive Training followed by 6 Month Assessments & Assignments
- Program include Four Certification
- Start Date – Every Month One Batch (Per Batch maximum 10 Students)
- 450+ Hiring Partners
- Certified from OTHM–UK Recognized by ofqual.gov.uk
- Minimum 5 Interview Call and until you get Job
- What you get After the Training – Gain Knowledge of At least 5+ Year Experienced Professional
- Designed for Freshers to Working Professionals
- 360 Degree Career Support
- One-on-One with Industry Mentors
- Dedicated Student Mentor
- 0 % Interest EMI – No Interest on EMI Paying – Approximately 5000 per month .
- Project Driven industry mentorship in Full

```
getToken = fn
getTokenSys = fn
discardL = fn
tokenizeInput
nextToken = fn
it = () : unit
ntk = ref TK_M
ption BigError
parse_factor =
is_unaryop = fn
parse_unaryop =
parse_unary =
is_multop =
parse_multo,
parse_term =
is_addop = fn
parse_addop =
parse_simple =
s_relop = fn
parse_relop =
parse_boolterm
s_boolop = fn
parse_boolop =
parse_expression
```

Stack Development

- Dedicated career support – 450 + Hiring Partners
- 150+ core projects, assignments and Job oriented activities
- 100 + Assignments
- 50 hours of Mock interview session
- Dedicated Trainers with ample of Industry Experience
- Project based IT Training and Certification programs .
- You Gain Knowledge of 5+year Experience as Full Stack Python Developer
- Industry standard Projects And Assignments with Mentor Support
- Lab Exercises and Practical Guidance



Target audience?

Those who would like to go with programming technologies, Coding etc. **Engineers, Freshers, Domain Experts, Software & IT Professionals**, Those who having career gap & Interested in Programming Technologies and Coding etc, Already working professionals, who would **like to upskill and try for another Job**. **Any One from Non-technical and Technical Background** who wants to come into IT Engineering/Management Graduate or Post-graduate Fresher Students.

Minimum Eligibility



Any Bachelor's degree. Completed or not completed. No coding experience required. If you have any Educational Gap or any other career gap, you can do this program to boost up your career. The qualifications provided by UK Regulatory board is equal to Level 7 Masters Degree in UK.

How You Benefit From This Program

International Post Graduate Diploma Certificate in **Software Development & Automation Engineering** without quitting job.

- Cutting-edge curriculum designed by industry experts .
- Trainers are from Real Time Industry
- Career transition with 30 to 100% salary hike

Frequently Asked Questions

1. Who Should do this course?

Freshers, Any Graduate, 2 to 4 Years Experienced up skilling enthusiasts. 3 rd Year Graduates who are going to attend campus Interviews. These courses designed in a way to be suitable for all branches of Engineering and all type of graduates – Science and non-science Graduates. Option to customise the subject according to the interest of candidate is also available

2. How many Interview Calls You will provide?

Minimum 5 Interview Calls and Maximum , until you get the JOB

3. Do I get Refund , if I am not interested to continued?

Yes .you should inform the Administrator and send email to masters@radicaltechnologies.co.in to cancel your admission within 7 days of the enrolment. Once we allocate the trainer and further proceedings , we will not be able to refund .

4. How I will get Industry knowledge?

All our programs are developed by Industry person, who is having 10+ years real working knowledge . Under their mentorship, you will be able to gain at least 5+ year experience

5. Is this is 100% Job Guarantee program ? And how is the Salary Structure?

Yes, it's a 100% JOB Guaranteed program. Minimum 5 Interview calls and you will interview calls up to you get satisfied Job. Minimum salary Package of 5 Lakhs per annum for the right candidate.

6. Is there any limitation on number of Interview calls providing?

No, You will get Interview calls until you get placed. And we will provide unlimited interview calls after you get placed also

7. How many people got placed as part of this program?

We have placement record of over 6000+, Out of this 800+ are from Master Program.

Go Down For
Complete Curriculum

 **Radical Technologies**
Project Based IT Training & Certification Centre



Html5 | CCS3 | Java Script | Bootstrap



HTML 5



Web Programming Introduction

- Architecture of a website
- Different technologies in making the website
- Web Development Introduction

HTML-Introduction History of HTML

- What you need to do to get going and make your first HTML page
- What are HTML Tags and Attributes?
- HTML Tag vs. Element
- HTML Attributes

HTML-Basic Formatting Tags

- HTML Basic Tags
- HTML Formatting Tags
- HTML Color Coding
- Module 4: HTML-Grouping Using Div Span
- Div and Span Tags for Grouping

HTML-Lists

- Unordered Lists
- Ordered Lists
- Definition list

HTML-Images

- Image and Image Mapping

HTML-Hyperlink

- URL – Uniform Resource Locator
- URL Encoding

HTML-Table

- `< table >`
- `< th >`
- `< tr >`
- `< td >`
- `< caption >`
- `< thead >`



- < tbody >
- < tfoot >
- < colgroup >
- < col >

HTML-Iframe

- Using Iframe as the Target

HTML-Form

- < input >
- < textarea >
- < button >
- < select >
- < label >

HTML-Headers

- Title
- Base
- Link
- Styles
- Script
- Meta

HTML-Miscellaneous

- HTML Meta Tag
- XHTML
- HTML Deprecated Tags & Attributes



CSS 3



CSS3-Introduction

- Benefits of CSS
- CSS Versions History
- CSS Syntax
- External Style Sheet using < link >
- Multiple Style Sheets
- Value Lengths and Percentages

CSS3-Syntax

- CSS Syntax
- single Style Sheets
- Multiple Style Sheets
- Value Lengths and Percentages

CSS3-Selectors

- ID Selectors
- Class Selectors
- Grouping Selectors
- Universal Selector
- Descendant / Child Selectors
- Attribute Selectors
- CSS – Pseudo Classes

Color Background Cursor

- background-image
- background-repeat
- background-position
- CSS Cursor

CSS3-Text Fonts

- color
- background-color
- text-decoration
- text-align
- vertical-align
- text-indent
- text-transform
- white space
- letter-spacing
- word-spacing
- line-height
- font-family
- font-size
- font-style
- font-variant
- font-weight

CSS3-Lists Tables

- list-style-type
- list-style-position
- list-style-image
- list-style
- CSS Tables
- border
- width & height
- text-align
- vertical-align
- padding
- color

CSS3-Box Model

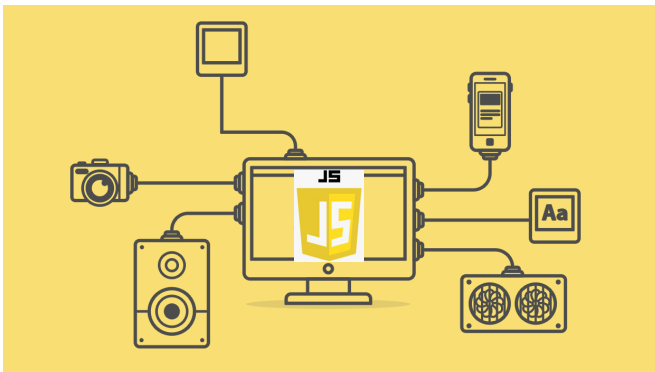
- Borders & Outline
- Margin & Padding
- Height and width
- CSS Dimensions

CSS3-Display

- Positioning
- CSS Visibility
- CSS Display
- CSS Scrollbars
- CSS Positioning
- Static Positioning
- Fixed Positioning
- Relative Positioning
- Absolute Positioning
- CSS Layers with Z-Index
- CSS Floats
- The float Property
- The clear Property
- The clear fix Hack



JAVASCRIPT



Introduction, Setup and Syntax

- Introduction
- Setup
- What is Javascript
- Course Exercise Files
- Practice App & Display Functions

Assignments 1:

- Using a Code Editor
- Variables
- Arrays

Assignments 2:

- Practice App

Control Structures, Functions, Math & Events

- If Statements
- Loops
- Math
- Functions
- Even Handlers Part:1

- Event Handlers Part:2

Assignments:03

- Practice App – If Statements and Loops
- Practice App – Math & Functions
- Practice App – Events

Advance Scenario #01

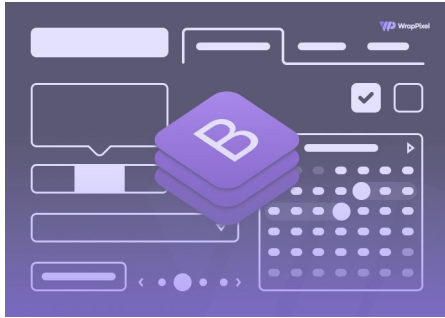
- Printing Arrays
- Text Output Function with a Loop
- Built-in Functions
- Math Built-in Functions

Advance Scenario #02

- Assigning, Adding and Removing Elements from Arrays
- Inserting and Extracting From Arrays more functions
- ForloopArray Checking with Challenge Part #1
- ForloopArray Checking with Solution Part #2
- Setting Up Flags and Using Booleans
- Where to go next?
- Quick Syntax Reference/ Resource File
- Conclusion



Bootstrap



Introduction to Bootstrap

- What is Bootstrap Framework
- Why Bootstrap
- History of Bootstrap
- Advantages of Bootstrap Framework
- What is Responsive web page
- How to remove Responsiveness
- Major Features of Bootstrap
- What is Mobile-First Strategy
- Setting up Environment
- How to apply Bootstrap to Applications

Bootstrap Grid

- What is Bootstrap Grid
- How to apply Bootstrap Grid
- What is Container
- What is Offset Column
- How to Reordering Columns
- Advantages of Bootstrap Grid
- How to Display responsive

Images

- How to change class properties
- How to use readymade themes
- How to customize Bootstrap & components, Less variables, and jQuery plug-in.

What is Bootstrap Typography

- How to use Typography
- What is Bootstrap Tables
- What is Bootstrap Form Layout
- What is Bootstrap Button
- How display images in different styles like Circle shape etc
- How to display text like muted and warning etc
- What is Carets Classes
- How to hide or show the text in Bootstrap

Bootstrap

- Components
- What is Bootstrap Components
- Why Bootstrap Components
- Advantages of Bootstrap Components
- What are the different types of Bootstrap Components
- What is Glyphicons Component
- How to use Glyphicons

Component

- What is Bootstrap Dropdown Menu Component
- What is Button Groups and Button Toolbar
- How to use Button Groups and Button Toolbar
- What are different Input Groups Components
- What is Navigation Pills & Tabs Components
- How to use Navigation Pills and Tabs Components
- What is Navbar Component
- How to build a Responsive Navbar
- How to Add Forms and other controls to Navbar
- How to Fix the position of navbar
- What is Breadcrumb Component
- What is Pagination Component
- How to apply Pagination in Application
- What is Labels / Badge Components
- What is Jumbotron / Page Header Components
- What is Thumbnail Component
- What is Alerts & Dismissible Alerts
- How to Create Progress Bar
- What is Media Objects Component

- Why Media Objects Component
- How to use Media Objects Component
- What is Bootstrap List Group Component
- What is Bootstrap Panel Component

Bootstrap Plug-Ins

- What is Bootstrap Plug-Ins
- Why Bootstrap Plug-Ins
- How to use Bootstrap Plug-Ins
- What is Transition Plug-in
- What Modal Dialog Box
- What are the different Properties, Methods and Events of Model Dialog Box
- What is Scrollspy Plug-In
- What is Tab Plug-in
- How to use Tab Plug-in
- What is Drop Down Plug-in
- What is Tooltip Plug-in
- How to use Button Plug-in
- What are the different methods and events of Tooltip Plug-in
- What is Popover Plug-in
- What is alert and Button Plug-ins
- What is Collapse Plug-in
- What are different types of Properties, Methods and Events of Collapse Plug-in
- What is Carousel Plug-in
- What is Affix Plug-in



Angular



Web development history

[ASP.net / PHP (all synchronous operations)]
=> [Introduction of Ajax] =>
[MVC.net websites (async operations)] => [Introduction of JSON web services]
=> [introduction of UI frameworks]

Service side VS client side programming

Available UI frameworks

Why Angular?

Environment setup

- Choice of IDE & installing it (visual studio code, visual studio, eclipse or any text editor)
- Installing node and npm
- Install angular CLI
- What is angular CLI?
Installing angular CLI.
Comprehensive list of CLI command.

Creating first Angular project

- Creating first Angular application with angular CLI
- Building and running angular Application
- Angular standards and deep dive into project structure (Significance of each folder & file) (Revise this after few lectures)
- How to run End to end test cases and unit testing

Angular project flow

- Flow from boot strapping app to running into browser
- Making small and easy changes to first app
- How to debug angular application

Prerequisite of learning Angular

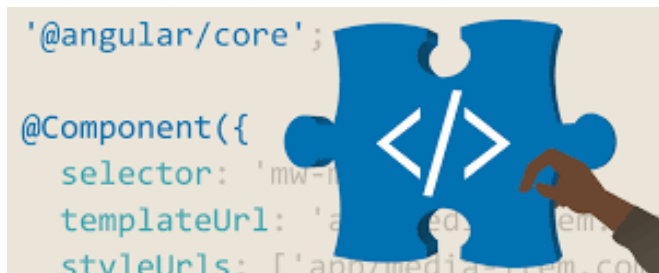
- Introduction to HTML
- Introduction to bootstrap CSS (optional)
- Introduction to Typescript (Necessity of Typescripts in enterprise applications) – Create classes use it in first app.

Angular Architecture

- Architecture overview
- Intro to Modules
- Intro to Component
- Intro to Services and DI (will cover in next lecture)

Components in details

- Component (selector, inline template, template url, styles, provider)
- Template syntax (various ways of data binding)



A. Interpolation

- Template expression
- Expression context

B. Property binding

- Difference between element attribute and property
- Element property binding
- attribute, class, style binding

C. Event binding

D. Two way binding

- Using various html elements in angular e.g. textbox, button, dropdown,

- checkbox, radio button etc
- Child components – Event emitter and custom property

Directives

- Attribute directives (ngClass, ngStyle, ngModel)
- Structural directives (ngIf, ngFor, ngSwitch)
- Creating custom directive

Pipes

- using inbuilt pipes e.g. DatePipe, UpperCasePipe, LowerCasePipe, CurrencyPipe, and PercentPipe etc
- Parameterized pipe
- Creating custom pipe

Animations

- Setup (npm install @angular/animations@latest –save)
- Example

Forms

- User input
- Form validations

NgModules in details

- NgModules introduction declaration, imports,

- exports, providers, bootstrap
- js modules vs NgModules
- Frequently used modules
- Roots module vs Feature module
- Creating feature module and use it in app (which requires routing)

Routing and navigation

- Configuring router for app
- Significance of `<base href="/">`
- Router outlet
- Router link
- Passing param using router
- Router events
- #Advance topics
- Routing components from other modules
- Lazy loading

Http client

- Setup
- Using Observable and RxJS
- Calling rest services (GET, PUT, DELETE, PUT).
- Error handling
- Calling aws rest service endpoint (or mocked rest service endpoint using `www.mockable.io`)
- #Advance topics
- Debouncing requests e.g. search widget
- Request/response interceptors

Angular 8 Course Content



Introduction of Angular 8

- Angular 8 Introduction
- History and versions of Angular 8
- Architecture of Angular 8
- How to install Angular 8 and set-up it.
- Creating our first Angular 8 app
- Angular 8 app loading

Difference between Angular And react

- Angular vs react

Angular 8 Advantages Disadvantage

- Advantage and Disadvantage of Angular 8

Angular 8 file structure

- Angular 8 file structure

Angular 8 components

- Components of Angular 8

Angular 8 CLI Commands

- All CLI commands of Angular

Angular 8 with Bootstrap

- How to install bootstrap for Angular 8
- Libraries of Angular 8

Angular 8 Routing

- Routing in Angular 8

Angular 8 directives

- Angular 8 Directives
- Angular 8 ngIf directive
- Angular 8 ngFor directive
- Angular 8 ngSwitch directive
- Angular 8 ngClass directive
- Angular 8 ngStyle directive

Angular 8 pipes

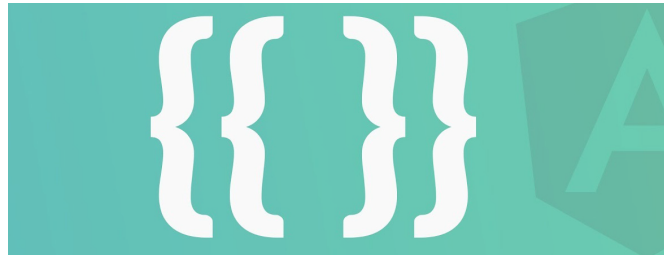
- Angular 8 Pipes

Angular 8 databinding

- Angular 8 Data binding
- Angular 8 Event binding
- Angular 8 Property binding
- Two-way data binding in Angular 8

String Interpolation In Angular 8

- Angular 8 String interpolation



Angular 8 forms

- Angular 8 Forms
- Data flow of forms in Angular 8
- Creating forms in Angular 8
- Testing and validation of forms in Angular 8

Error fixing in Angular 8

- Error fixing in Angular 8

Dependency injection and services in Angular 8

- Dependency injection services in Angular 8

Angular 8 Animations

- Angular 8 Animations

Dynamic components in Angular 8

- Dynamic components in Angular 8

Angular 8 Module

- Angular 8 Module
- Deploying an angular 8 app

Introduction of unit testing in angular 8

- unit testing in angular 8

Observables in angular 8

- Observables in angular 8

Angular 8 universal

- Angular 8 universal

Angular 8 Changes and new features

- New features and changes in Angular 8

Conclusion

- Angular 8 Conclusion

Creating application – User management system

Deploying app (Illustration of deploying on AWS cloud server)

Interview questions will be discussed with every topic

Git repository setup(optional)

100%
Job Guaranteed
Master Program

Any Graduate Can Do!



MongoDB



Introduction

- Downloading
- Package Content
- First Run
- Command Line Options
- Verify Server
- Getting Help in the Shell

Saving Data

- Introduction
- Overview
- Storage
- BSON
- Saving Documents
- Collections
- Document Id
- ObjectId
- Insert
- Insert with Id
- Complex Document
- Save Danger
- Update Command
- Update Demo
- Set Operator
- Unset Operator
- Rename Operator
- Push Operator
- Pull Operator
- Pop Operator
- Array Type
- Multi Update

- Find And Modify
- Query With Sort

Finding Documents

- Introduction
- Overview
- find()
- Equality
- Projection
- Comparison
- \$not
- \$in
- Arrays
- \$all
- \$nin
- Dot Notation
- Sub-Document
- null and \$exists
- And
- Cursor
- sort()
- limit()
- skip()
- findOne()



Deleting Document

- remove
- deleteOne
- deleteMany

Indexing

- Introduction

- Overview
- Scan Is Bad
- Sort Uses Index
- Index Types
- Create Index
- system.indexes collection
- explain()
- dropIndex()
- Index Name

ES6 Java Script



React



Getting Started

- What is React?
- Real-World SPAs & React Web Apps
- Why Should we Choose React?
- React environment setup

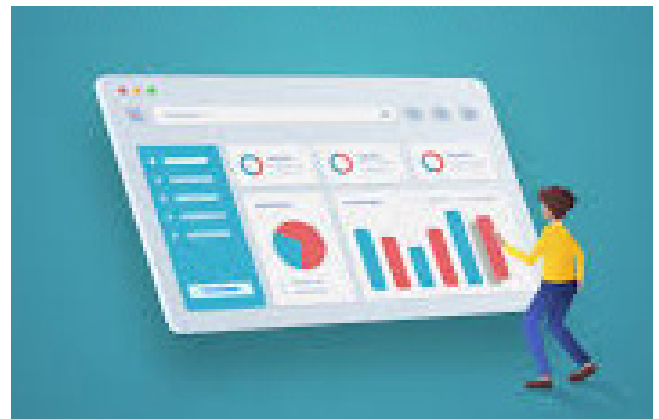
Understanding the Base

Features & Syntax

- Understanding the Folder Structure
- Understanding Component Basics
- Understanding JSX
- Creating a Functional Component
- Creating a Class Component
- Working with Components & Re-Using Them
- Understanding & Using State
- Props & State
- Using the useState() Hook for State
- Manipulation
- Stateless vs Stateful Components

Diving Deeper into Components & React Internals

- Splitting an App Into Components
- React Hooks
- .Using useEffect() in Functional Components
- Controlling the useEffect() Behavior
- Cleaning up with Lifecycle Hooks &



- useEffect()
- How React Updates the DOM
- Higher Order Components (HOC) –
- Introduction
- Another Form of HOCs
- Using Refs
- Understanding Prop Chain

- Problems
- Using the Context API
- `contextType` & `use Context()`

Adding Routing to our Burger Project

- Setting Up Routing & Routes
- Passing Ingredients via Query
- Params

- Implementing Navigation Links

Forms and Form

Validation

- Custom hooks for error validation

Redux state manage

Node JS

Node Conventions for Writing Asynchronous Code

Node's Event Loop

Modules, `require()` & NPM

Getting Node.js

Finding 3rd Party Modules via NPM

Node.js Background

Introduction, Accessing Built-in Modules

Demo: Installing Node on Linux/Windows with NVM



Express



Introduction

- Setting up Express
- Running Express
- Debugging Options
- Static Files – Public Directory
- Introduction, Making Web Requests in Node
- Building a Web Server in Node
- Demo: Building a Web Server in Node



Routing

- Introduction
- Navigation
- Routing
- Rendering
- Separate Files
- Router Functions

Authentication

- Introduction to Authentication
- Auth Routes
- Passport
- Local Strategy

Events and Streams

- Introduction
- Events and the EventEmitter class
- Readable and Writable Streams, the Pipe function
- Demo: Readable and Writable Streams
- Demo: Piping Between Streams

Accessing the Local System

- The Process Object
- Demo: The Process object
- Interacting with the File System
- Demo: Interacting with the File System
- What is a Buffer?
- The OS Module

Socket.IO

- Realtime Interaction with Socket.IO
- Demo: Socket.IO

Testing and Debugging

- Introduction, The Assert Module
- Demo: The Assert Module
- Testing with Mocha and Should.js
- Demo: Mocha and Should.js

Scaling Your Node Application

- Introduction, The Child Process Module
- Demo: The “exec” function
- Demo: The “spawn” function
- Demo: The “fork” function
- Scaling with Node’s Cluster Module
- Demo: Building a Clustered Web Server

Learn Express

From 5+ Yr Experience
Full Stack Developer!

Advanced Python For Web Development



Introduction

- What is Python..?
- A Brief history of Python
- Why Should I learn Python..?
- Installing Python
- How to execute Python program
- Write your first program

Variables & Data Types

- Variables
- Numbers
- String
- Lists, Tuples & Dictionary



Conditional Statements & Loops

- if...statement
- if...else statement
- elif...statement
- The while...Loop
- The for.... Loop

Control Statements

- continue statement
- break statement
- pass statement

Functions

- Define function
- Calling a function
- Function arguments
- Built-in functions

Modules & Packages

- Modules
- How to import a module...?
- Packages
- How to create packages

Classes & Objects

- Introduction about classes & objects
- Creating a class & object
- Inheritance
- Methods Overriding
- Data hiding

Files & Exception Handling

- Writing data to a file
- Reading data from a file

- Read and Write data from csv file
- try...except
- try...except...else
- finally
- os module

Getting started with Python Libraries

- what is data analysis ?
- why python for data analysis ?
- Essential Python Libraries
- Installation and setup
- Ipython
- Jupyter Notebook
- 7 VS 3.5

Working with Pandas

- Installing pandas
- Pandas data frames
- Pandas Series
- Data aggregation with Pandas Data Frames
- Concatenating and appending Data Frames
- Joining Data Frames

Data Loading, Storage and file format

- Writing CSV files with numpy and pandas
- JSON Data
- YAML Data

Python Regular Expressions

- What are regular expressions?
- The match Function
- The search Function
- Matching vs searching
- Search and Replace
- Extended Regular Expressions
- Wildcard



Python Oracle Database Access

- Install the cx_Oracle and other Packages
- Create Database Connection
- CREATE, INSERT, READ, UPDATE and DELETE Operation
- DML and DDL Operation with Databases
- Performing Transactions
- Handling Database Errors
- Disconnecting Database

Django

Flask

RESTAPI

Plotting Graph on HTML Page

- Introducing Matplotlib
- Bar Charts
- Line Charts
- Scatter plots

- Bubble charts

(Assignment and Live Examples)

Any website that Student would like to design.

Sample resumes helping you to create your resume

After Completion of MASTER PROGRAM,

**You Will Get
International
Post Graduate
Certificate!**



Django

django

INTRODUCTION TO DJANGO

- Features of Django
- Django web server
- Understanding Django environment
- A simple 'Hello world' application

- Template inheritance
- Sending data from url to view
- Sending data from view to template



DISPLAYING HYPERLINKS - PROJECT

- Django architecture
- MVC and MTV
- Starting a project
- Django apps
- Activating our first app
- A view that displays a hyperlink
- Mapping the views to URLs
- Running our first app
- Improving the views using templates
- The improved templates

CREATING A WEBSITE - PROJECT

- Starting a project
- Creating an app inside the project
- Activating the app
- Creating model for our site
- Converting the model into a table
- Examples for Fields in Models
- Basic data access using Django shell
- Saving objects into database
- Retrieving objects from database
- Modifying objects of database
- Sorting objects
- Filtering objects
- Deleting objects
- changes to data model



CREATING ADMINISTRATION PANEL

- Using the admin interface
- Customizing the admin interface
- Adding users
- Data access and modification using admin panel
- Giving permissions to users

CREATING FIRST PAGE OF OUR SITE

- The Django template system
- Template Inheritance
- Improving the website
- Adding background color for web pages
- Adding banner to the web site
- Adding background image in the web pages
- Storing and displaying images
- Adding users to our site
- Uploading the information by user
- Changing the administration interface title

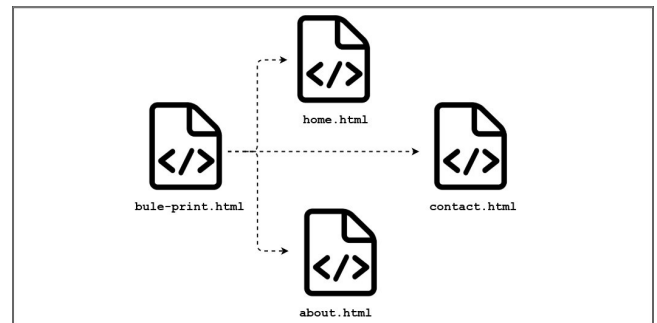
DJANGO FORMS CREATION

- Forms basics
- Creating Contact Us form
- Form field examples

DJANGO'S EMAIL FUNCTIONALITY

- Configuring email settings
- Sending emails with Django

DJANGO TEMPLATE LANGUAGE



- Django template tags
- If/else
- If/elif/else
- For
- Comments
- Filters
- Using templates to display data in the site

INTEGRATING BOOTSTRAP INTO DJANGO

- Introduction to Bootstrap
- Creating tables
- Creating grids
- Creating carousels

SESSIONS AND COOKIES

- Difference between session and cookie
- Creating sessions and

cookies in Django

USING OTHER DATABASES IN DJANGO

- Using SQLite
- Configuring MySQL database
- Working with MySQL in Django
- Configuring Oracle database
- Working with Oracle in Django

DJANGO RESTful API

- Introduction to Django Web Framework
- Introduction to REST and API
- Introduction to Restful Webservices
- Django Rest Framework(DRF)
- DRF Advantages and Constraints
- Difference between Django and REST API
- REST API pre-requisites
- REST API Configuration and Installation
- Serialization and Deserialization
- JSON and XML
- Action and handler methods
- Function based and class based views

- API Views

Mixins

- a. ListModelMixin
- b. CreateModelMixin
- c. UpdateModelMixin
- d. DestroyModelMixin
- e. RetrieveModelMixin

Generic Views

- a. ListAPIView
- b. CreateAPIView
- c. UpdateAPIView
- d. DeleteAPIView
- e. ListCreateAPIView
- f. ListCreateUpdateAPIView
- g. ListCreateUpdateDestroyAPIView

ViewSets

- a. ViewSets
- b. ModelViewSets

Security

- a. Authentications
- b. Authorizations

CRUD operations

LIVE PROJECT IMPLEMENTATION

- Project life cycle
- Creating a functional website in Django