

**START YOUR
FULL STACK
PYTHON CAREER
TODAY!!**

CREDO SYSTEMZ

**Full Stack
Python
Program**

Capstone Projects :

Real Time Full Stack Developer
Training Projects



Food Delivery App

Building an application to browse restaurants, order food, and track deliveries in real-time.



Twitter/Instagram Clone

Developing a social media platform to post content, follow others, with likes and comments.



Chat Application

Building a real-time chat app in which users can message each other in one-on-one or group chats.



Event Booking App

You can see events and book tickets online. It sends email confirmation after booking.



Learning Management System (LMS)

People can search and apply for jobs. Companies can post job openings.



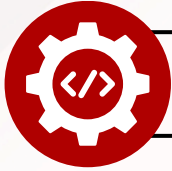
Learning App

Users can order food online from restaurants. It shows live order updates and payment options.



Full Stack

Opportunities & Demand



Lead Full Stack Engineer



Full Stack Architect



Technical Lead (Full Stack)



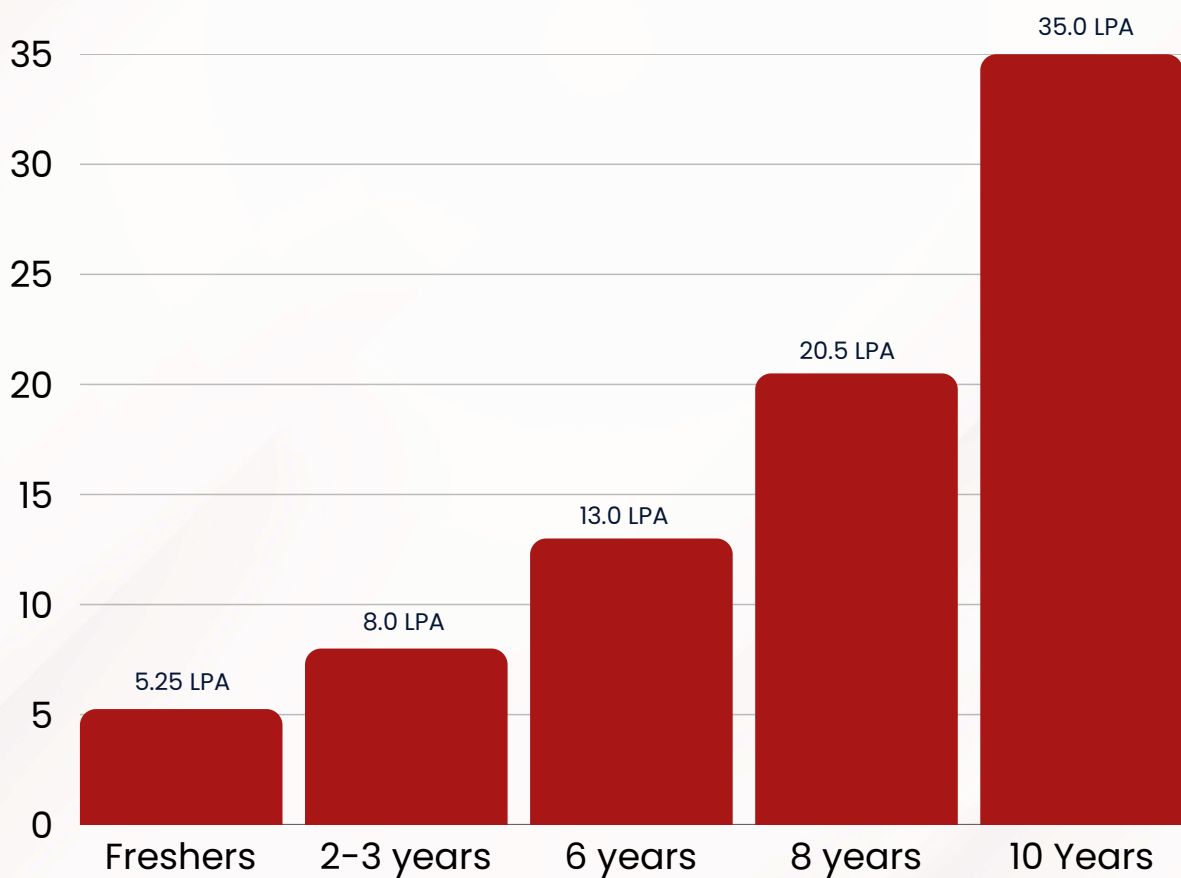
Full Stack DevOps Engineer



Cloud Full Stack Developer



Full Stack AI Developer



FULL STACK COURSE SYLLABUS

Duration : 100 hrs

Phase 1 – Frontend Development

HTML , CSS COURSE SYLLABUS

Section 1 : Introduction to WEB

- What is Web?
- Web Features?
- W3C and W3C Members
- Introduction to What WG

Section 2 : Core HTML

- Introduction
- Parts in HTML Document
- Version Information
- Head Section
- Meta Information
- Favicons
- Body Section
- HTML FORMS
- Anchors, Images

Section 3: Advance HTML5

- Introduction
- HTML5 HISTORY
- Why HTML5?
- New Features and Groups
- Structure of HTML5 Document
- Power of HTML5 and Features
- Semantics and Block Level Elements
- HTML5 Forms
- HTML5 Multimedia
- HTML5 Graphics



● Section 4: Core CSS

- Introduction
- CSS Basics
- CSS Introduction
- CSS Syntax
- CSS Versions
- CSS Id & Class
- CSS Styling
- Styling Backgrounds
- Styling Text
- Styling Fonts
- CSS Borders

● Section 5: Advance CSS

- Introduction
- CSS3 Modules
- Selectors
- Box Model
- Backgrounds and Borders
- Text Effects
- 2D/3D Transformations
- Core & Advanced Animations
- Multiple Column Layout
- User Interface

JAVA SCRIPT COURSE SYLLABUS

● Section 6: Core JavaScript

- What is Script? Types of Scripts?
- Introduction to JavaScript
- Comments and Types of Comments
- Popup Boxes
- Variables & Operators
- JavaScript Functions and Events
- Conditional Statements
- Looping Control Statement

● Section 7: Advance JavaScript

- Types of Errors
- Exception Handling
- Java Script Objects
- Browser Objects
- Validations in JS

● Phase 2 – Backend Development

PYTHON COURSE SYLLABUS

● Section 8: Core Python

- Introduction to Python
- What is Python?
- History of Python
- Python Versions
- Features of Python
- How to Install Python
- Install Python with Diff IDEs
- Creating Your First Python Program
- Printing to the Screen
- Reading Keyboard Input
- Using Command Prompt and GUI or IDE

● Section 9: Different Modes in Python

- Execute the Script
- Interactive Mode
- Script Mode
- Python Comments
- Working with Python in Unix/Linux/Windows/Mac/Android
- Python New IDEs
- PyCharm IDE
- How to Work on PyCharm
- PyCharm Components
- SublimeText IDE
- What is PIP?

● Section 10: Variables in Python

- What is Variable?
- Variables in Python
- Constants in Python
- Standard Data Types
- Operators and Operands
- Swap variables
- Type Conversion
- String Handling

● Section 11: Python Conditional Statements

- How to use "if condition" in conditional structures
- if statement (One-Way Decisions)
- if .. else statement (Two-way Decisions)
- How to use "else condition"
- if ..elif .. else statement (Multi-way)
- When "else condition" does not work
- How to use "elif" condition
- How to execute conditional statement with minimal code

● Section 12: Python LOOPS

- How to use "While Loop"
- How to use "For Loop"
- How to use For Loop for set of other things besides numbers
- Break statements in For Loop
- Continue statement in For Loop
- Enumerate function for For Loop

● Section 13: Python Lists

- Lists are mutable
- Getting to Lists
- List indices
- Traversing a list
- List operations
- List slices
- List methods
- Map, filter and reduce

● **Section 14: Python 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

● **Section 15: Python 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

● **Section 16: Python Dictionary**

- How to create a dictionary?
- Python Hashing?
- Python Dictionary Methods
- Copying dictionary
- Updating Dictionary
- Delete Keys from the dictionary
- Dictionary items() Method
- Sorting the Dictionary
- Python Dictionary in-built Functions

● **Section 17: Python Functions**

- What is a Function?
- How to define and call a function in Python
- Types of Functions
- Significance of Indentation (Space) in Python
- How Function Return Value?
- Types of Arguments in Functions

- Default Arguments
- Non-Default Arguments
- Keyword Arguments
- Non-keyword Arguments
- Arbitrary Arguments
- Rules to define a function in Python
- Various Forms of Function Arguments
- Scope and Lifetime of variables
- Anonymous Functions/Lambda functions
- map(), filter(), reduce() functions

● **Section 18: Advanced Python**

- Python Exception Handling
- Python Errors
- Common RunTime Errors in PYTHON
- Abnormal termination
- Chain of importance Of Exception
- Exception Handling
- Try ... Except
- Try ..Except .. else
- Try ... finally

● **Section 19: Python Class and Objects**

- Introduction to OOPs Programming
- Object Oriented Programming System
- OOPS Principles
- Basic concept of Object and Classes
- Access Modifiers
- How to define Python classes
- Self-variable in python
- What is Inheritance? Types of Inheritance?
- How Inheritance works?

● **Section 20: Python Regular Expressions**

- What is Regular Expression?
- Regular Expression Syntax
- Understanding Regular Expressions

- Regular Expression Patterns
- Literal characters

● **Section 21: Bootstrap (Powerful Mobile Front-End Framework)**

- What is Responsive Web Designing?
- Typography Features
- Bootstrap Tables, Buttons, Dropdowns, Navbars
- Bootstrap Images
- Bootstrap Responsive utilities
- Bootstrap Glyph icons

● **Section 22: Bootstrap Grid System**

- What is a Grid?
- What is Bootstrap Grid System?
- MOBILE FIRST STRATEGY
- Working of Bootstrap Grid System
- Media Queries

● **Section 23: Grid Options**

- Responsive column resets
- Offset columns
- Nested columns

● **Phase 3 – Database Development**

DJANGO COURSE SYLLABUS

● **Section 24: Django Web Framework**

- What is a Framework
- Introduction to Django
- Django – Design Philosophies
- History of Django
- Why django and Features
- Environment setup
- Web Server

● Section 25: MVC Pattern

- MVC Architecture vs MVT Architecture
- Django MVC – MVT Pattern

● Section 26: Getting Started with Django

- Creating the first Project
- Integrating the Project to sublime text
- The Project Structure
- Running the server
- Solving the issues and Migrations
- Database Setup
- Setting Up Your Project

● Section 27: Create an Application

- What Django Follows
- Structure of django framework
- Model Layer
- What are models
- Model fields
- Querysets

● Section 28: Django – Admin Interface

- Starting the Admin Interface
- Migrations

● Section 29: Views Layer

- Simple View
- Basic view(displaying hello world)
- Functional views, class based views

Section 30: Django – URL Mapping

- Organizing Your URLs
- Role of urls in djnago
- Working urls
- Forms

- Sending Parameters to Views
- Templates layer
- The Render Function

● **Section 31: Django Template Language (DTL)**

- Role of template layer in django
- Filters,Tags, Tag if, Tag for, Block and Extend Tags
- Comment Tag, Usage of templates
- Extending base template

● **Section 32: Django – Models**

- Creating a Model
- Manipulating Data (CRUD)
- Linking Models
- Django – Page Redirection

● **Section 33: Django – Sending E-mails**

- Sending a Simple E-mail
- Sending Multiple Mails with send_mass_mail
- Sending HTML E-mail
- Sending HTML E-mail with Attachments

● **Section 34: Django – Form Processing**

- Using Form in a View
- Usage of forms
- Crud operations using forms
- Crispy forms in django

● **Section 35: Django – File Uploading**

- Uploading an Image
- Django – Apache Setup

● **Section 36: Django – Cookies Handling**

- Django – Sessions
- Django – Comments

● **Section 37: Django Admin**

- Creating Super User
- Using admin in Django
- Adding models to admin
- Adding model objects using admin
- Displaying in cmd using querysets

● **Section 38: Static files**

- Loading css files into templates
- Loading js files into templates
- Uploading image using models
- User authentication

● **Section 39: Deployment**

- wsgi server
- Gunicorn, uwsgi concepts
- Godaddy domain
- Digital ocean cloud machine concepts
- Deploy your application



SKILLS AND TOOLS

Skills Covered



Tools Covered



PLACEMENT SUCCESS STORIES

Designation Company Package
.....



Shashank

Full Stack Developer



10.8 LPA



Anuradha

Front End Developer



8.5 LPA



Sandhya

Software Developer



7.8 LPA



Ramesh

Cloud Engineer



8.6 LPA



Joesph

Full Stack Developer



8.0 LPA



Mohammed

Full Stack Developer



11.0 LPA



OUR HIRING PARTNERS



Earn your Full Stack Course Completion Certificate

Credo Systemz's certificate is highly recognized by
30K Global companies around the world.



WHAT OUR **TRAINEE** SAYS?



Santhosh

4.7 ★★★★★

Credo Systemz's Full Stack course is very practical and job-oriented. The trainers explained every topic clearly with real-time examples. Placement support was strong. I got placed after training. Totally worth the investment!



Fathima Banu

4.2 ★★★★★

The Full Stack training at Credo Systemz covered front-end and back-end thoroughly. Live projects helped a lot. Mentors were always helpful. Thanks to their placement team, I'm now working in an IT company.



Prakash Jain

5.0 ★★★★★

I joined Credo Systemz for Full Stack training and it was the best decision. Great trainer, hands-on projects, and mock interviews helped me switch to IT. Strongly recommend for career changers!



Saindhavi

4.9 ★★★★★

Credo Systemz Full Stack program gave me confidence to attend interviews. I learned MERN stack with real-time examples. The support from admin and placement team was great. Got placed right after course completion.



Jesuraj. F

4.0 ★★★★★

As a fresher, Credo Systemz's Full Stack course gave me all the knowledge I needed. The trainer's guidance and regular assessments helped me grow. I got placed within weeks of completing the course!



Anjali Ravi

4.5 ★★★★★


I had no coding background but the Full Stack course at Credo Systemz was easy to follow. Trainers were supportive and explained clearly. Placement team constantly followed up until I got placed. Thanks!



CHENNAI


VELACHERY

New # 30, Old # 16A, Third Main Road, Rajalakshmi Nagar, Velachery, (Opp. to Murugan Kalyana Mandapam), Chennai – 600 042.

 +91 98844 12301

OMR

Plot No.8, Vinayaga Avenue, Rajiv Gandhi Salai, (OMR), Okkiampettai, (Behind Okkiyampet Bus Stop) Chennai – 600 097.

 +91 96001 12302

OVERSEAS

USA

Houchin Drive, Franklin, TN -37064. Tennessee

UAE

Sima Electronic Building, LLH Opposite, Electra Street – Abu Dhabi