

CELEBRATING
14 YEARS

QualityThought[®]

Transforming Dreams! Redefining Future!

NO.1

TIMES
BUSINESS
AWARDS
HYDERABAD-2022

"MOST PROMISING SOFTWARE
TRAINING INSTITUTE"



</> node JS

Do You Want to be a Fullstack Web Developer



Mongo DB React JS
Express JS Node JS



Mongo DB Angular JS
Express JS Node JS

40,000+
Students Trained

14,000+
Students Placed

200+
Placement
Companies

14 Years
of Student Trust

Hyderabad
Vijayawada
Chennai





Step into the world of software training guided by the insights of an industry luminary, Ramana Bhupathi. With a career spanning over a decade and a half in the dynamic Information Technology sector, Mr. Bhupathi brings a wealth of experience and expertise that has shaped the course of modern software development.

Mr. Bhupathi's journey commenced at the forefront of technological innovation, with pivotal roles at Satyam Computer and Accenture Consulting. His hands-on involvement in diverse projects, leveraging cutting-edge technologies and agile development methodologies, set the stage for his profound insights into the digital realm.

Notably, his proficiency in executing projects on Microsoft Technologies became a hallmark of his career, reflecting his adaptability and commitment to staying at the vanguard of change.

A witness to the seismic shifts within the software development landscape, Ramana Bhupathi is resolute in his belief that the forthcoming decade will usher in even more rapid transformations. In his astute perspective, the IT workforce must embrace a culture of perpetual upskilling and reskilling to navigate this era of accelerated change.

Mr. Bhupathi's journey is characterized by a passion for cultivating enterprises rooted in emerging technologies. His endeavors have been focused on crafting platforms that not only resonate with the needs of today but also create a positive societal impact. His affinity for startups, each driven by a dedication to scalable solutions powered by Artificial Intelligence and Data Science, underscores his commitment to innovation.

Beyond the boardroom, Mr. Bhupathi's impact extends to the realm of education. His mentorship has illuminated the path for over 10,000 students and professionals, steering them toward lucrative careers in technology. With an unwavering calling to guide future tech talent, he embraced his true purpose, transitioning from the corridors of corporate leadership to the forefront of education. Today, his energy is channeled into honing the skills of young minds, immersing them in the nuances of new-age technologies through comprehensive training and impactful corporate programs.

Join us on an enlightening journey through the pages of this book, as Ramana Bhupathi's profound insights and guidance illuminate the ever-evolving landscape of Information Technology. His wisdom serves as a beacon for both novices and seasoned professionals, reminding us all that in the midst of change lies unparalleled opportunity.

Content Explanation

Program Name :

Demo Date / Time :

Fee :

Duration :

Trainer Name :

Web Development Fundamentals

- ⇒ What is Web?
- ⇒ Web Server, Web Client
- ⇒ Website, Webpage
- ⇒ Request and Response
- ⇒ Http/Https
- ⇒ Languages and Tools of Web Development
- ⇒ How the Web Works
- ⇒ Inspecting HTTP Requests and Responses
- ⇒ HTML Basics
- ⇒ CSS Basics
- ⇒ Formatting Code
- ⇒ Inspecting Pages Using DevTools
- ⇒ Validating Web Pages

HTML & HTML 5

- ⇒ Introduction
- ⇒ The Head Section
- ⇒ Text
- ⇒ Entities
- ⇒ Hyperlinks
- ⇒ Images
- ⇒ Video and Audio
- ⇒ Lists
- ⇒ Tables
- ⇒ Containers
- ⇒ Semantic Elements
- ⇒ Structuring a Web Page

Typography

- ⇒ Introduction
- ⇒ Styling Fonts
- ⇒ Embedding Web Fonts
- ⇒ Flash of Un-styled Text
- ⇒ Font Services
- ⇒ System Font Stack
- ⇒ Sizing Fonts
- ⇒ Vertical Spacing
- ⇒ Horizontal Spacing
- ⇒ Formatting Text
- ⇒ Practices

CSS Layout

- ⇒ Introduction
- ⇒ The Box Model
- ⇒ Sizing Elements
- ⇒ Overflowing
- ⇒ Measurement Units
- ⇒ Positioning
- ⇒ Floating Elements
- ⇒ Flexbox
- ⇒ Grid
- ⇒ Hiding Elements
- ⇒ Media Queries
- ⇒ Summary
- ⇒ Practices

Bootstrap

- ⇒ Introduction
- ⇒ Bootstrap classes
- ⇒ Responsive grid system
- ⇒ Columns and rows
- ⇒ Components
- ⇒ Responsive navbars
- ⇒ Margins
- ⇒ Modals, dialogs
- ⇒ Buttons
- ⇒ Forms
- ⇒ List groups
- ⇒ Badges, pills
- ⇒ Cards
- ⇒ Tables
- ⇒ Alerts
- ⇒ Navigation options
- ⇒ Links

JavaScript Basics

- ⇒ Introduction to JavaScript
- ⇒ How to Write JavaScript Program
- ⇒ How to debug JavaScript code
- ⇒ Where to place JavaScript code in the HTML file
- ⇒ Comments and Statements
- ⇒ Data Types ⇒ Variables
- ⇒ Hosting ⇒ Operators
- ⇒ Conditional Statements
- ⇒ Loop ⇒ Functions
- ⇒ Function Expressions in JavaScript
- ⇒ Recursive Function
- ⇒ Popup Boxes ⇒ Events
- ⇒ Examples of JavaScript Events
- ⇒ Exception Handling in JavaScript
- ⇒ this keyword ⇒ Call, Apply, bind
- ⇒ Output ⇒ API call using HTTP, FETCH
- ⇒ HTTP Methods (GET, POST, PUT, DELETE)
- ⇒ web storage
- ⇒ DOM
- ⇒ Add execution context and call stack
- ⇒ In Javascript basics

JavaScript Promise

- ⇒ Promise
- ⇒ Promise Chaining in JavaScript
- ⇒ Promise.race() vs. Promise.all()
- ⇒ Promise Error Handling
- ⇒ Async Await

OOPs

- ⇒ How to Create Objects in JavaScript
- ⇒ Prototypical Inheritance in JavaScript
- ⇒ Classes
- ⇒ Inheritance
- ⇒ Abstraction
- ⇒ Polymorphism
- ⇒ Encapsulation
- ⇒ Constructor Function
- ⇒ Object Factories
- ⇒ Instance Of
- ⇒ Call by Value & call by reference
- ⇒ Object Creation Using New Object Method
- ⇒ Javascript Objects Are Mutable

JavaScript Advance Es6

- ⇒ let Keyword
- ⇒ const Keyword
- ⇒ Iterators and Iterables
- ⇒ function are First-class citizen
- ⇒ Callback functions
Synchronous and Asynchronous
- ⇒ Anonymous Function
- ⇒ Immediately Invoked
Function Expressions (IIFE)
- ⇒ Arrow Function
- ⇒ Console Object
- ⇒ Shadowing

JavaScript – New Features

- ⇒ Template Literals
- ⇒ Object Literals
- ⇒ Default Parameters

ReactJS 18.2.0 Training Overview

React (also known as React.js or ReactJS) is a free and open-source front-end JavaScript library for building user interfaces based on UI components. It is maintained by Meta (formerly Facebook) and a community of individual developers and companies.

It is used for creating dynamic and interactive user interfaces for mobile and web applications. It is highly flexible, declarative and efficient for developing scalable, simple, and fast front-end for web & mobile applications. In simple terms, React JS effectively handles the view layer of mobile and web application.

React can be used as a base in the development of single-page, mobile, or server-rendered applications with frameworks like Next.js. However, ReactJS is only concerned with state management and rendering that state to the DOM, so creating React applications usually requires the use of additional libraries for routing, as well as certain client-side functionality

React is flexible in such a way that, in any application, we can use as little or as much React as you need. For example, react can be used in any existing web application to develop a new feature or even the application's entire UI.

ReactJS has become highly popular across the globe because of its extra simplicity and flexibility. Many people are even referring to ReactJS as the future of web development.

Part of this huge popularity comes from the fact that top corporations such as Facebook, PayPal, Uber, Instagram, Airbnb etc use it to develop the user interfaces.

Objectives of the Course

This is to provide in-depth insights about ReactJS and keep updated yourself with latest ReactJS concepts and it's integrated other technologies according to updated industry standards.

⇒ The main objective of React Training is to make you as a top-class React.js Developer able to design and build modern Interactive user interface components to enhance application performance.

Who should do the Course

⇒ Anyone who is trying to learn fastest growing UI framework and one of the top 3 frameworks. It is growing fast in terms of opportunities too.

Prerequisites

- ⇒ HTML
- ⇒ JavaScript
- ⇒ CSS

ES6: JavaScript concepts related to ReactJS

- ⇒ History of JavaScript
- ⇒ Features
- ⇒ let & const with example
- ⇒ Arrow Functions
- ⇒ Alternative

ReactJS: Introduction

- ⇒ What is ReactJS?
- ⇒ Version history.
- ⇒ Why ReactJS?
- ⇒ Advantages of React JS
- ⇒ Internals of ReactJS.
- ⇒ Work flow of React JS
- ⇒ Scope of React JS
- ⇒ Software's required.
- ⇒ Tips and Arrow Functions
- ⇒ Exports and Imports
- ⇒ Tips for exports and imports
- ⇒ Classes with example
- ⇒ Inheritance
- ⇒ Spread and rest Parameter
- ⇒ Destructuring

ReactJS: Setup – First Application

- ⇒ About NodeJS
- ⇒ Installation
- ⇒ Npm – managing the packages
- ⇒ Create First ReactJS Application
- ⇒ Build and Execute
- ⇒ About Visual Studio Code
- ⇒ Installation of Visual Studio Code
- ⇒ Managing the ReactJS application using Visual Studio Code

ReactJS: JSX

- ⇒ Overview of JSX
- ⇒ Jsx syntax
- ⇒ Difference between JS and JSX.
- ⇒ Jsx Parser
- ⇒ Comments in jsx
- ⇒ Introduction of Virtual DOM.
- ⇒ Rendering an Element into the DOM
- ⇒ Naming Conventions
- ⇒ JSX Transpilation to React Code Example
- ⇒ Running the Transpiled Code
- ⇒ What are the JavaScript expressions available in JSX?

ReactJS: All about Components

- ⇒ React Components
- ⇒ Creating a Functional Component Example
- ⇒ Components vs Elements
- ⇒ Properties
- ⇒ Spread Attributes (an ES6 Feature)
- ⇒ Expressions
- ⇒ Fragments
- ⇒ Functional Components
- ⇒ Example of JSX Nesting
- ⇒ Setting CSS Styles Using Classes
- ⇒ Setting CSS Styles Directly
- ⇒ JSX Escapes Values
- ⇒ Working with Lists of Items - Iterating through Lists
- ⇒ Keys in Lists
- ⇒ Example List with Key
- ⇒ State in React
- ⇒ Setting state
- ⇒ Types of State Data
- ⇒ State Hierarchy
- ⇒ Lifting State Up
- ⇒ Props vs. State
- ⇒ Immutability
- ⇒ Immutability – Why?
- ⇒ Virtual DOM and State
- ⇒ Updating Input fields
- ⇒ Passing Props to Components
- ⇒ Passing Functions to Components
- ⇒ Event Handling
- ⇒ Event Handler Example
- ⇒ Event Binding - Dos
- ⇒ Event Binding – Don'ts
- ⇒ Passing Parameters to Event Handlers
- ⇒ Component Life-cycle
- ⇒ Initial Render
- ⇒ Props Change
- ⇒ State Change
- ⇒ Component willMount
- ⇒ Component didMount
- ⇒ Component Unmount
- ⇒ Life-cycle in Functional Components
- ⇒ Applying Different Lifecycles in the Application.
- ⇒ When to choose Appropriate lifecycles.
- ⇒ Interaction in between components
- ⇒ Component Communication using Context

Forms and UI

- ⇒ Lists of Form components.
- ⇒ Setup Controlled and Uncontrolled form components.
- ⇒ React JS Form validations.
- ⇒ Create a React Form.
- ⇒ Client-side form validation.
- ⇒ Using Formik Library with React
- ⇒ Using YUP Library with Formik in React

Working with Rest API

- ⇒ Calling REST API
- ⇒ Sending POST request
- ⇒ CRUD Operations example

React Router

- ⇒ Routing and Navigation
- ⇒ react-router
- ⇒ Creating a react-router based project
- ⇒ A Basic Routed Component
- ⇒ Router vs. BrowserRouter
- ⇒ The Route component
- ⇒ Redirect Route
- ⇒ Navigating with
- ⇒ Route Parameters
- ⇒ Retrieving Route Parameters
- ⇒ QueryString Parameters
- ⇒ Create a Single Page Application.
- ⇒ Applying Routing.
- ⇒ Dynamically render the components based on the url

State Management for React

- ⇒ React State Basics – Props and State
- ⇒ Props
- ⇒ State in Class Based Components
- ⇒ Managing State with Hooks in Functional Components
- ⇒ The Problem with Props and State
- ⇒ Redux State Library
- ⇒ Redux Advantages
- ⇒ Redux Disadvantages
- ⇒ Basic Rules for State Management
- ⇒ Types of State ⇒ Data State
- ⇒ Communication State
- ⇒ Control State ⇒ Session State
- ⇒ Location State
- ⇒ Location State Side Effects

Event handling in JSX

- ⇒ on Blur, onKeyUp, onChange and other useful primary events in React JS.
- ⇒ How to Sharing events between the components?
- ⇒ Communicate Data between components.
- ⇒ Applying all lists of events

Styles in ReactJS

- ⇒ CSS and inline styles in React JS overview.
- ⇒ Introduction to styled components
- ⇒ Styling the application using styled component
- ⇒ How to use Animations in the Application

Building React Apps with Redux

- ⇒ Redux
- ⇒ Redux Terminology
- ⇒ Redux Principles
- ⇒ Redux: Actions ⇒ Redux Action Types
- ⇒ Action Creators ⇒ Dispatching Actions
- ⇒ Data Flow Basics ⇒ Redux Reducers
- ⇒ Pure Functions
- ⇒ Reducer Example
- ⇒ Returning Default State
- ⇒ Creating a Development Environment with create-react-app
- ⇒ Using Redux with React
- ⇒ Initializing the Store
- ⇒ Immutability
- ⇒ Benefits of Immutable State
- ⇒ Mutability of Standard types
- ⇒ Copying Objects in JavaScript
- ⇒ Copying Arrays in JavaScript
- ⇒ One Store - Multiple Reducers
- ⇒ Combining Reducers
- ⇒ Components and Redux
- ⇒ The React-Redux Package
- ⇒ Wrapping App with Provider
- ⇒ mapStateToProps
- ⇒ mapDispatchToProps
- ⇒ Using Mapped Properties & Methods
- ⇒ Wrapping Components with Connect
- ⇒ Configure Store
- ⇒ Programming Advice - MultiTab Console

Flux

- ⇒ What is Flux Architecture?
- ⇒ What are the Flux Components available?
- ⇒ How Flux works?
- ⇒ Flux and React works together.

Using React Hooks

- ⇒ Functional Component Shortcomings
- ⇒ Hooks Overview
- ⇒ Hook Rules
- ⇒ Functional Component Props
- ⇒ The useState Hook
- ⇒ The useEffect Hook
- ⇒ The useContext Hook
- ⇒ Additional Hooks
- ⇒ The useReducer Hook
- ⇒ The useMemo Hook
- ⇒ The useRef Hook
- ⇒ Creating Custom React Hooks

Unit Testing React with React Testing Library

- ⇒ What are the necessary Tools required for Unit Testing?
- ⇒ React Unit Testing overview
- ⇒ Introduction to JEST.

Server Integration & Deployment

- ⇒ https
- ⇒ httpster
- ⇒ npm

Code Splitting

- ⇒ Code splitting & Suspense
- ⇒ Route Based Code Splitting
- ⇒ Lazy Loading

Isomorphic React

- ⇒ Server-Side Rendering
- ⇒ SSR with React - Setup & Server
- ⇒ SSR with React - The Toolchain

New Features of React 16,17 & 18

Webpack Primer & Isomorphic React

Isomorphic React

- ⇒ Webpack and its use
- ⇒ Setting up and installing Webpack
- ⇒ Working with the configuration file of Webpack
- ⇒ Working with loaders
- ⇒ Quick word on lazy loading, code splitting, and tree shaking
- ⇒ Setting up a hot module replacement
- ⇒ Server-side rendering (SSR)
- ⇒ Working with renderToStatic Markup and renderToString methods

Introduction to Node.JS

- ⇒ What is Node.js?
- ⇒ Why Node.js?
- ⇒ Node Architecture
- ⇒ How Node Works?
- ⇒ Your First Node Program

Node Module System

- ⇒ Introduction
- ⇒ Global object
- ⇒ Modules
- ⇒ Creating & Loading a Module
- ⇒ Module Wrapper Function
- ⇒ Path Module
- ⇒ OS Module
- ⇒ File System Module
- ⇒ Events Module
- ⇒ Event Arguments
- ⇒ Extending Event Emitter
- ⇒ Http Module

Node Package Manager

- ⇒ Introduction
- ⇒ Package.json
- ⇒ Installing a Node Package
- ⇒ Using a Package
- ⇒ Package dependencies
- ⇒ NPM packages and source control
- ⇒ Semantic versioning
- ⇒ Listing the installed packages
- ⇒ Viewing the registry info for a package
- ⇒ Installing a specific version of a package
- ⇒ Local and Global Packages
- ⇒ Updating local packages
- ⇒ Dev Dependencies
- ⇒ Uninstalling a package
- ⇒ Working with Global packages
- ⇒ Publishing a package
- ⇒ Updating a published package

Practices

- ⇒ Install Node.js
- ⇒ Run a simple Node.js Script
- ⇒ Create a Node Module and use it in your code
- ⇒ Search the NPM Repository to find a package and install it in your project

Asynchronous Programming

- ⇒ What is Asynchronous Programming in JavaScript?
- ⇒ JavaScript EventLoop
- ⇒ Callbacks
- ⇒ Callback Hell
- ⇒ Promises
- ⇒ Promise Chaining
- ⇒ Async Await

Practices

- ⇒ Demonstrate the use of call backs, promises, and async-await

FileSystem

- ⇒ Synchronous vs Asynchronous IO
- ⇒ File Constants
(`__dirname` and `__filename`)
- ⇒ Synchronous File Operations
- ⇒ Asynchronous File Operations
- ⇒ FileSystem Watcher

Practices

- ⇒ Write Node.js code to asynchronously perform file operations
- ⇒ Write Node.js code to log to the console if a file changes in a directory

Streams and Events

- ⇒ What are Streams in Node.js
- ⇒ Reading and Writing Streams
- ⇒ Understanding and using EventEmitter class
- ⇒ Emitting Events
- ⇒ Responding to events

Practices

- ⇒ Read data from a file using Input Stream and display on screen

HTTP

- ⇒ What is HTTP
- ⇒ How Does HTTP Work?
- ⇒ Standard HTTP response codes
- ⇒ Node HTTP Module
- ⇒ Processing Form Data
- ⇒ Processing Query Strings from URL
- ⇒ Sending Back response

Practices

- ⇒ Create a simple HTTP Server using Node

Express JS

- ⇒ What is ExpressJS?
- ⇒ Installing ExpressJS
- ⇒ Creating a simple HTTPServer using ExpressJS
- ⇒ Routing in Express
- ⇒ Templating in Express
- ⇒ Templating using JADE
- ⇒ Middlewares

Practices

- ⇒ Create a simple Express JS Application with routes and templates

Sockets

- ⇒ What is real time application
- ⇒ Building Chat Application using Sockets

Express JS and Mongo DB

- ⇒ What is MongoDB
- ⇒ MongoDB Advantages
- ⇒ Documents
- ⇒ Collections
- ⇒ Mongoose ODM
- ⇒ Creating Schemas and Models using mongoose
- ⇒ CRUD Operations in Mongo DB using Mongoose and Express

Practices

- ⇒ Creating a simple Application using ExpressJS and Mongoose

Testing

- ⇒ What is Unit Testing?
- ⇒ Why Unit Testing?
- ⇒ Unit Testing Frameworks
- ⇒ What are Mocks and Stubs?
- ⇒ Writing and Running tests in Express

Practices

- ⇒ Write and run some unit tests to test a module

REST API using Express and Mongoose

- ⇒ What is a REST API
- ⇒ Why REST API?
- ⇒ Data Exchange Formats
- ⇒ JSON vs. XML
- ⇒ Monolith Architecture
- ⇒ Microservice Architecture

Practices

- ⇒ Build a small application using the REST API

Cloud -AWS

- ⇒ Types of Applications
- ⇒ PC vs Server
- ⇒ Need of Cloud Infrastructure
- ⇒ What is Cloud Computing
- ⇒ What is AWS? And Why to choose AWS
- ⇒ How to create an account in AWS
- ⇒ What an Operating System and its types
- ⇒ Importance of Linux in Cloud Environment
- ⇒ How to create a Linux server in AWS
- ⇒ Putty vs Xshell
- ⇒ AWS EC2 Instance
- ⇒ Types of instances in AWS
- ⇒ AWS Tenancy
- ⇒ AWS Launch Template
- ⇒ AWS EBS volumes
- ⇒ Need of Elastic IP Address
- ⇒ AWS RDS service for JDBC Application?
- ⇒ How to deploy web application in AWS Elastic Bean Stalk
- ⇒ AWS S3 service
- ⇒ How to deploy spring boot application in AWS Cloud

Communication Skills

Roots of Communication	LSRW	
	7 Cs of Communication	
Roots of Grammar	Parts of Speech	Mastering Helping Verb And Main Verb
	Sentence Structure Development	
	Tense Logic	
	Worksheet Sessions	
Speech Intelligence	Vocabulary Development	
	Usage of words	
	Group Discussions	
	JAMS	
	Debates	
Personality Development	Public Speaking	
	Imagination and Innovation Training	
	Centralized Brain Storming	
	Problem Solving Skill s	
Mangement Skills	Desiion Making	
	Time Management	
	Team Building	
	Task Management	
Interview Skills	Leadership Skills	
	Employability	
	Think like a Professional	
	Clearing HR Rounds	
	Salary Negotiation	
Presentation Skills	Bond Negotiation	Dream Company Reading Skillls Comprehension Skills
	Research Skills	
	Public Speaking	
	Visualization	
	White Board Presentation	
	Mastering Powerpoint	
Personality Development	Content Creation	
	Mind Mapping	
	Role plays	
	Mock Interview on the Hot Seat	
	Listening Skills	
	Critical Thinking	
	Thought Analysis	
SWOT Analysis		

Aptitude & Reasoning

Quantitative

- ⇒ Basic Maths
- ⇒ Algebra
- ⇒ Percentages
- ⇒ Profit And Loss
- ⇒ Discounts
- ⇒ Averages
- ⇒ Time and Work
- ⇒ Chain Rule
- ⇒ Pipes and Cisterns
- ⇒ Ratios
- ⇒ Proportions
- ⇒ Partnerships
- ⇒ Time and Distance
- ⇒ Trains
- ⇒ Boats and Streams
- ⇒ Simple Interest
- ⇒ Compound Interest

Data Interpretation

- ⇒ Bar Charts
- ⇒ Line Charts
- ⇒ Pie Charts
- ⇒ Table Charts

Reasoning

- ⇒ Directions
- ⇒ Letter Series
- ⇒ Number Series
- ⇒ Coding - Decoding
- ⇒ Blood Relations
- ⇒ Statement and Assumption
- ⇒ Analogy
- ⇒ Odd Man Out Series
- ⇒ Venn Diagrams
- ⇒ Mirror Images
- ⇒ Water Images
- ⇒ Arranging in Order
- ⇒ Paper Folding / Cutting
- ⇒ Grouping
- ⇒ Counting the figures
- ⇒ Clocks
- ⇒ Calenders
- ⇒ Seating Arrangements
- ⇒ Syllogism
- ⇒ Puzzles

2024 CALENDAR

January

Sun	Mon	Tue	Wed	Thu	Fri	Sat
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

February

Sun	Mon	Tue	Wed	Thu	Fri	Sat
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29		

March

Sun	Mon	Tue	Wed	Thu	Fri	Sat
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

April

Sun	Mon	Tue	Wed	Thu	Fri	Sat
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

May

Sun	Mon	Tue	Wed	Thu	Fri	Sat
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

June

Sun	Mon	Tue	Wed	Thu	Fri	Sat
						1
2	3	4	5	6	7	8
10	9	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30						

July

Sun	Mon	Tue	Wed	Thu	Fri	Sat
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

August

Sun	Mon	Tue	Wed	Thu	Fri	Sat
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

September

Sun	Mon	Tue	Wed	Thu	Fri	Sat
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

October

Sun	Mon	Tue	Wed	Thu	Fri	Sat
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

November

Sun	Mon	Tue	Wed	Thu	Fri	Sat
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

December

Sun	Mon	Tue	Wed	Thu	Fri	Sat
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

100% Success Rate in the Placements for Skilled People



Manikanta Sriram
 Package | 7 LPA
CENDYN



Sreekanth Reddy
 Package | 6 LPA
Fusion Tech Partners



Raghavendra Rao
 Package | 6 LPA
Fusion Tech Partners



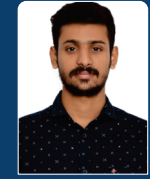
Surya Venkata Sai
 Package | 3 LPA
Infosys



S Venkateswara
 Package | 3 LPA
Orblike Technology



Kesari Ramya
 Package | 4 LPA
Value Labs



D Chandrabhas
 Package | 3 LPA
Wamika



P Sai Swathi
 Package | 3 LPA
Wamika



Prashanth N
 Package | 4 LPA



Gopi Krishana
 Package | 6 LPA
Mr Soft Solutions



M Sunny Raju
 Package | 3 LPA
Extended



Akash Rebala
 Package | 3 LPA
Extended



Linga Swamy
 Package | 6 LPA
Fusion Tech



Lekhana
 Package | 6 LPA
Fusion Tech



Ajay Sreenivas
 Package | 3 LPA



Kemburi Ashok
 Package | 3 LPA
Gove Enterprises



Jalagam Rakshitha
 Package | 3 LPA
Gove Enterprises



Kshitun Harina D
 Package | 3 LPA
Gove Enterprises



Surya Narayana M
 Package | 4 LPA
Cognizant



P Satish
 Package | 3.5 LPA
Acunor

NO.#1
JOB ORIENTED
INTENSIVE
PROGRAM
IN HYDERABAD

Our Students Are Placed In
