FITA®

TRAINING | PLACEMENT

Accredited by

ISO

ORACLE® WORKFORCE DEVELOPMENT PROGRAM

psi

PEARSON VUE

Microsoft

ISTQB® Certified Tester Foundation Level

Rated as 1 TRAINING ACADEMY BY BUSINESS REVIEW TODAY

STAY AHEAD WITH FITA

ABOUT FITA ACADEMY

FITA is a leading Skill development and Placement company managed by IT veterans with more than two decades of experience in leading MNC companies. We are known for our practical approach towards trainings that enable students to gain real-time exposure on competitive technologies & Foreign Languages.

Transforming Students & IT Professionals into Industry-Ready Workforce since 2012!

GREAT FUTURE STARTS FROM HERE

Since 2012

120+
COURSES

75,000+
STUDENTS

1000+
EXPERT TRAINERS

100%
PLACEMENT SUPPORT

1500+
PLACEMENT TIEUPS

15+
BRANCHES*

Classroom Locations

Chennai

Bangalore

Coimbatore

Madurai

Pondicherry

*Includes all the group companies

300+ Corporate Clients

Corporates:

Cognizant™

INDEMAND

Prodapt

Chargebee

HCL

Guardian

iLink Systems

minuscule TECHNOLOGIES

JEEVAN TECHNOLOGIES

2adpro

PHOTON

Volanté Business Information Agility

epi

NEWAGE THE OUTSPOKEN DAILY

preludesys IMPLICIT KNOWLEDGE

Colleges:

INDIAN INSTITUTE OF TECHNOLOGY MADRAS • सिद्धिर्भवति कर्मजा •

SRM UNIVERSITY
UNIVERSITY
(Under section 3 of UGC Act 1956)

VELLORE INSTITUTE OF TECHNOLOGY
VIT

GOJAN EDUCATIONAL INSTITUTIONS
GOJAN

HINDUSTAN
UNIVERSITY

VELAMMAL
INSTITUTE OF TECHNOLOGY
Approved by AICTE - New Delhi, Affiliated to Anna University - Chennai
Accredited by NATIONAL ASSESSMENT AND ACCREDITATION COUNCIL
"Velammal Knowledge Park", Chennai - Kolkatta Highway, Ponneri - 601204

OUR STUDENTS WORKS AT:

FITA®
TRAINING | PLACEMENT

POLARIS®
SRINSOFT
L&T Infotech
PayPal
wipro

tcs TATA CONSULTANCY SERVICES
Cognizant
amazon
freshworks
accenture

ORACLE®
SUTHERLAND
AMARPRAKASH BUILDING YOUR FUTURE
Volanté Business Information Agility
RSalesArm Extend • Engage • Empower

i|Nautix technologies inc
NTT DATA
BOONZE MEDIA SOLUTIONS
InfoWave
VenPep SOLUTIONS

NDTV தமிழ்
Sify
infinix
Codoid
CognitiveMobile

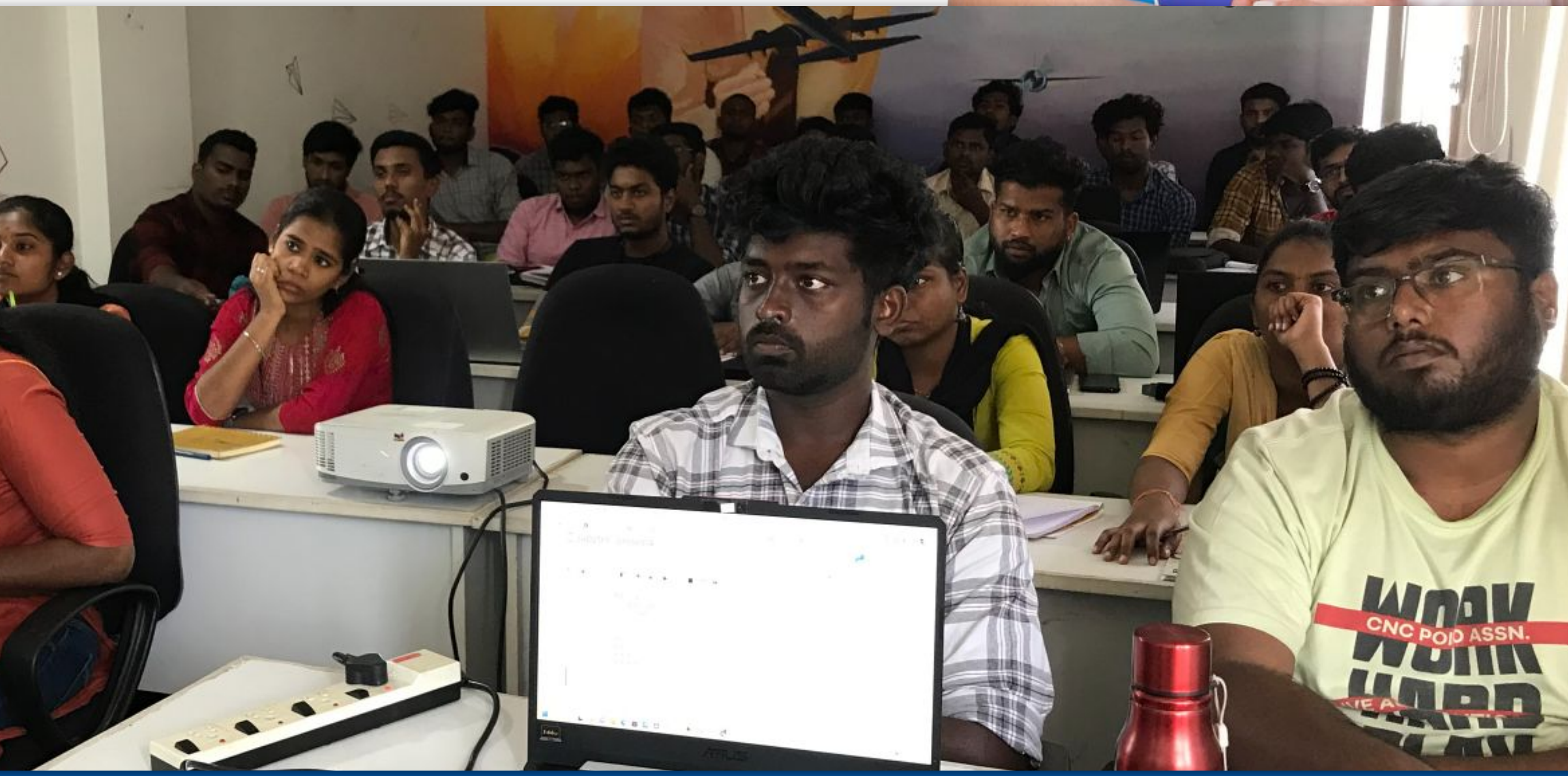
Why Learn Master Program in Full Stack at FITA Academy:

- Live Capstone Projects
- Real time Industry Experts as Trainers
- Placement Support till you get your Dream Job offer!
- Free Interview Clearing Workshops
- Free Resume Preparation & Aptitude Workshops

Complementary Value Add Course(Online)

- ✓ Python & Data Structures
- ✓ DevOps Tools - Git, Jenkins, Maven
- ✓ Aptitude
- ✓ Softskills
- ✓ Spoken English & Communication
- ✓ Intensive Placement Workshop

Students at Placement Workshop:



Master Program in Full Stack

User Interface (UI) Developer

Introduction to Websites

- What is a Webpage
- Introduction to HTML
- Creating a simple HTML Document
- Viewing HTML Document with Browsers
- Introduction to different Web Browsers – IE, Chrome, Mozilla, Safari, UC Browser
- Webpage Vs Website
- URLs / Domains
- Web Servers and Web Hosting

HTML

- Elements of a HTML Document
- Understanding HTML Tags
- HTML Editors
- Creating a Simple HTML Document
- Adding Attributes to Tags
- Handling Texts in HTML

- Headings
    - Paragraphs
    - Formatting Texts
    - Managing Blocks with <Div> Tag
    - <Div> - Id's and Class
    - Line Breaks
    - Managing Lists

- HTML Styles
    - Controlling Fonts and Size
    - Alignment
    - Applying Colors
    - Foreground and background colors
- HTML Tables
    - Creating a Table
    - Managing Columns
    - Table Borders
    - Border Spacing
- Handling Images
    - Adding Image
    - Formatting Image
    - Handling Size of an Image
    - Aligning Images
    - Adding Animated Images
    - Background Images

- Handling Layouts
- HTML Forms
    - Introduction to HTML Forms
    - Form Elements
    - Form Input and Attributes

Introducing HTML 5

- HTML Vs HTML 5
- New Elements in HTML 5
- New Form Elements in HTML 5
- Handling Audio & Video
- Handling Graphics
    - Canvas
    - Handling SVG
- Handling Events
- HTML 5 APIs
    - Geolocation
    - Drag / Drop
    - Web Storage
    - Advantage of Web Storage over Cookies
    - Session based Web Storage

- Web Workers
  - Purpose of Web Workers
  - Advantages of using Web Workers
  - Creating and Running a Web Worker
  - Terminating a Web Worker
  - Resuing a Web Worker

Introducing CSS and CSS 3

- What is a Style Sheet
- Cascading Effect of a Style Sheet
- Inline and External CSS
- CSS Selectors – IDs and Classes
- Understanding Layouts
- Absolute, Relative and Fixed Positioning
- Advanced CSS
  - Transformations
  - Responsive Layouts and Media Queries

Simplifying CSS with BootStrap

- Introducing Boot Strap Framework
- Advantages of Bootstrap Framework over CSS
- Responsiveness with Bootstrap

- Bootstrap Grids
- Bootstrap UI Components
    - Buttons, Tables, Menus, Carousels, Dropdowns, Alerts

Javascript

- Introduction to Javascript
- Necessity for Javascript
- Javascript Structure and Elements
- Operators and Functions
- Control Structures
- The Document Object Model (DOM), Objects and Nodes
- Handling DOM using Javascript
- Javascript Events
- Animating UI Elements

```
        <div>

            <form method="post" action="#" id="formvalue" onkeyup
            drawChart()" />

            </form>
        </div>
</div>



<script type="text/javascript" src="https://www.google.com/jsapi'
  script>
<script type="text/javascript">

var bid = 43;
var ask = 21;


  google.load("visualization", "1", {packages:["corechart"]});
  google.setOnLoadCallback(drawChart);
  function drawChart() {
    var data = google.visualization.arrayToDataTable([
      ['Price', 'Quantity'],
      ['Value #1', bid],
```

JQuery

- What is JQuery
- Advantages of JQuery
- Integrating JQuery through CDNs
- Understanding JQuery Selectors
- UI Components
    - Buttons, Lists, Toolbars, Menus,
- Handling Forms using JQuery
- JQuery Events
- Understanding Ajax
- Implementing Ajax

Programming with ReactJS

TypeScript Framework

- Introduction to TypeScript
- Advantages of TypeScript
- Overview of Node and Node Environment Setup
- Installing and Configuring TypeScript Engine
- Understanding Transpilation

TypeScript Variables and Data types

- Declaring and Defining Variables
- About “any”
- “var” Vs “let” keywords
- Static and Dynamic Type
- Data types : String, Number, Array, Object, Tuple, Enum, Void and NULL

OOPS in TypeScript

- Defining Class and Creating Objects
- Property, Methods and Constructors
- Handling Inheritance
- Types of Inheritance in TypeScript
- Understanding Access Modifiers
- Static Methods in TypeScript
- Understanding Interfaces in TypeScript

ReactJS – Introduction and Features

- The Need for ReactJS
- Advantages of Features of React JS
- Understanding Single Page Applications - SPA
- Exploring the Component based Architecture
- Understanding React Syntax

ReactJS – Essentials

- Understanding NodeJS
- Using the NPM
- Using Webpack for ReactJS
- Choosing the right IDE
- Understanding Browser Plugins
- Working with various Browsers and Browser Plugins
- Understanding React Syntax
- Understanding and using React-dom.js
- Understanding Babel

ReactJS – Using Nodes

- Exploring React Nodes
- Advantages of using React Nodes
- Defining and using Node Attributes
- Using Element Factories
- Understanding and using Node Events
- Understanding and manipulating DOM

ReactJS – The Java Script Xtension: JSX

- Introduction to JSX
- Understanding JSX Format
- Using React Nodes with JSX
- Understanding JSX Rendering
- Using Expressions and Comments
- Creating and Managing JSX Attributes and Properties
- Handling Events using JSX
- Understanding JSX code to Javascript Compilation

ReactJS – Understanding Components

- Understanding React Components
- Exploring various types of React Components
- Advantages of using React Components
- Creating and Managing React Components
- Referring a Component
- Understanding the Components Life Cycle
- Exploring various Life Cycle Methods
- Component willMount, didMount and willUnmount
- Component willUpdate and DidUpdate
- Understanding to access sub-components or Child components
- Creating and Managing Component Events

- Understanding Components Props
- Learning Setting a Default Components Props
- Sending and Getting Components Props
- Setting up Components Props Validation
- Understanding Components State
- Working with Components State
- Understanding Components State Vs Components Props
- Creating and Managing Stateless Components Functions

ReactJS – Creating UI, Forms and Handling Events

- Understanding Forms in React
- Exploring various Form Components
- Handling User Inputs
- Fixing up Default Values
- Understanding Form Validations
- Exploring ReactJS UI Components
- Working with Styling Components
- Understanding In-line Styles and External Styles
- Exploring various UI Component Libraries
- Handling Events in ReactJS
- Handling Events – Mouse Events, Keyboard Events and OnChange Events

ReactJS – State Management using Redux and Flux Views

- Understanding Application States
- The Power of Redux for React
- Understanding Flux Views

- Advantages of Redux and Flux with React
- Managing States with Redux
- Learning and using the Redux Store
- Learning the Flux ArchitectureUsing various Flux Components
- Using Stores and Dispatchers
- Understanding Actions, Views and Controllers
- Using View Controllers
- Handling with Reducers and Dispatchers
- Using Selectors

ReactJS – Routing

- Understanding Routers in React
- Learning to Configure React Router
- Understanding Router Library
- Learning to use Parameters in Routing
- Exploring Nested Routing
- Using React Router with Redux

ReactJS – Handling API Calls

- Understanding APIs and Web Services
- Understanding the need for External Data
- Understanding HTTP requests and responses
- Using Fetch and Axios to manage Web service calls

Eslint & coding conventions

Debugging tools and strategies

Expert JAVA

Core JAVA

Introduction to Java

- Object Oriented Programming Fundamentals
- Structure of a Java Program

Data Types

- Primitive data types
- Reference Data Types
- Keywords, Identifiers, Expressions

Operators

- Arithmetic Operators
- Assignment Operators
- Logical Operators
- Relational Operators
- Bitwise Operators

Variables

- Declaration, Definition, Types

Methods

- Syntax, Types

Object

- Object Creation, Reference, Reference Variables

Constructors

- Pass by Value and Pass by Reference
- Access Specifiers
- Access Levels
- Decision Making and Control Structures

Strings

- String, String Buffer, String Builder

Java Beans

Arrays

- Declaring and defining arrays
- Primitive Arrays
- Object Arrays

Inheritance

- Is-A Relationship
- Has-A Relationship
- Inheritance using extends keyword
- Inheritance using implements Keyword

Abstraction

- Abstraction using abstract classes
- Abstraction using interfaces

Encapsulation

Interfaces

- Interfaces Vs Classes
- Nested Interfaces
- Interface as a type

Polymorphism

- Overloading
- Constructor overloading
- Overloading between classes
- Overriding

Exception Handling

- Exception
- Categories of Exception, Exception hierarchy
- Throw and throws keywords
- Try catch and finally keywords

Collection Framework

- Core Interfaces
- Core Classes
- Iterator
- Comparable & Comparator

Generics

- Auto boxing
- Unboxing

Casting

- Primitive Casting
- Reference Casting
- Up casting and down casting

File Handling

- File Handling in Java
- Files, Streams, Types of Streams

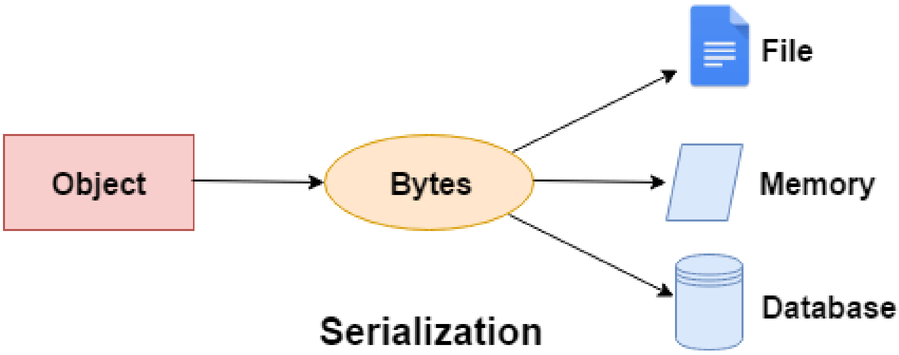
Serialization

Threads

- Thread Life Cycle
- Thread States
- Creating Threads
- Threads Priorities
- Thread Groups
- Synchronization

Inner Classes

- Nested Classes
- Anonymous Classes



Java Enterprise Edition (JEE)

Overview of J2EE and WWW

The HTTP Protocol and Web Application Introduction

Environment Setup

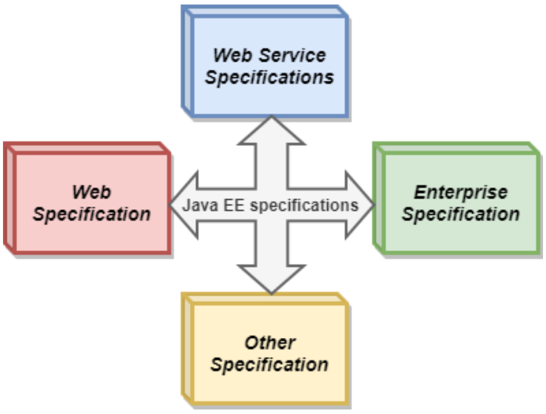
HTML

Servlets

- What is a Servlet?
- Servlet Lifecycle, Configuring a Servlet, Types of Servlet
- Servlet Context
- Servlet Config
- Deployment descriptor
- Session Management

Java Server Pages (JSP)

- JSP Lifecycle
- Servlets vs JSP
- Scriptlets
- Directives
- Declaration
- Sessions

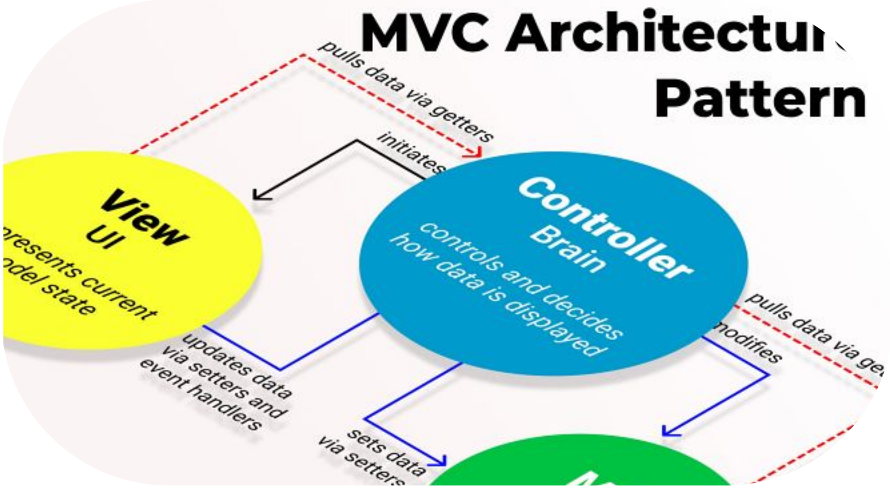


- Mixing Scriptlets and HTML
- Tag Libraries
- Beans and Forms Processing

MVC Architecture

JDBC

- Database Setup (MySQL)
- Overview
- JDBC Driver Types
- How JDBC Works
- Steps Involved
- JDBC Process details
- Queries
- Prepared Statements
- Callable Statements



Spring and Spring Boot Framework

Spring: Introduction

- What is Spring
- Advantages of the Spring Framework
- Spring Framework Architecture and Spring Modules
- Java and Spring Configuration

Spring: Core Container

- Components of Spring Core
- Object Coupling – Tight and Loose Coupling
- Dependency Injection (DI)
- Types of DI
- Concepts and Implementation of Inversion of Controls (IoC)
- Spring Bean – Properties, Scope, Method, Bean Lifecycle
- DI with Bean
- Spring BeanFactory
- Wiring Beans
- AutoWiring Beans
- Understanding the Default AutoWiring

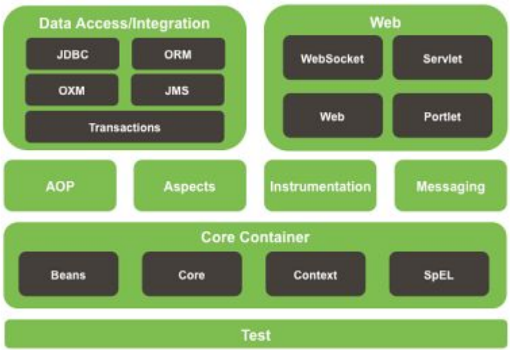
- AutoWiring by Name and AutoWiring by Constructors
- Annotations in Spring
- Dependency Injection using Annotations
- Wiring Bean with Annotations
- Introduction to Spring Expression Language (SPEL)
- SPEL Operators
- Implementing Annotations with SPEL

Spring: Data Access and Integration

- Data Access with Spring using JDBC
- JDBC Templates
- Data Access Object Patterns (DAO) and Bean
- Querying Database and Binding the Variables
- Handling Database Exceptions
- Executing Update and Delete Statements
- Batch Update data
- Database Transactions
- Transactions Management with Spring

Spring: Web Layer

- Spring Model View Controller (MVC) Architecture
- Components of MVC
- Setting Up a Spring MVC Application
- The Purpose of Dispatcher Servlets



- Spring Controllers
- Spring View Resolvers
- Adding Data using Spring Data Models
- Creating and Managing Forms in Spring
- Managing File Uploads
- Apache Tiles Integration

Spring: Aspect Oriented Programming (AOP)

- Limitations of Object Oriented Programming
- Introduction to Aspect Oriented Programming
- Advantages of AOP
- Terminologies Associated with AOP
- Aspects
- “Advice” to Aspects
- Types of Advice – Before, After, Around, Others
- Creating Annotations based Aspects
- Point Cut Expressions – “Within”, “This”, “Target”
- Point Cut Designators
- AspectJ

Spring: Security

- Securing Applications with Spring Security
- Spring Security Filters
- Configuring Authentications
- Spring Authorizations
- “Remember Me” Functionality

Spring: Boot

- Introduction to Spring Boot
- Features of Spring Boot
- The Spring Boot Project Structure
- The Spring Boot Initializr
- Spring Boot Actuator
- Configuring the Spring Boot Server
- Spring Boot Application Properties
- Spring Vs Spring Boot

SQL

Introduction

- What is Database?
- Oracle Database
- Data Model
- Normalization

Types of SQL Languages

- Data Definition Language (DDL) : CREATE, ALTER, DROP, TRUNCATE
- Data Manipulation Language (DML) : INSERT, DELETE, UPDATE, SELECT
- Data Control Language (DCL) : GRANT, REVOKE
- Transaction Control Language (TCL) : COMMIT, ROLLBACK, SAVEPOINT

SELECT Statement & Condition Types

- List the Capabilities of SQL Select Statements
- Select All Columns
- Select Specific Columns
- Use Column Heading Defaults
- DESC Command
- Conditon Types : AND, OR, BETWEEN, IN, NOT IN, LIKE, IS NULL, IS NOT NULL, EXISTS

Restrict, Sort, Group, Filter Data

- How to Restrict the data using WHERE Clause
- Operators and Precedence of Operators
- Sort the data using ORDER BY Clause
- Group the data by using the GROUP BY clause
- Filter groups of date by using the HAVING Clause

Data Types

- Number
- Character
- LONG and Row
- Date and Time
- Large Object Data Types (LOB Types)
- Pseudo Columns

SQL Constraints

- Primary Key Constraint
- Foreign Key Constraint
- Unique Constraint
- Not Null Constraint
- Check Constraint
- Default Constraint
- Guide lines to Add, Remove, Alter Constraints

SQL Joins

- Inner Join
- Outer Join
- Left Outer Join
- Right Outer Join

  Self Join, Anti Join, Semi Join

  Union, Union All, Intersect, Minus (Set Operators)

SQL Functions

- Single Row Functions
  - NULL – Related Functions: COALESCE, LNNVL, NANVL, NVL, NVL2
  - Numeric Functions: ABS, CIEL, FLOOR, MOD, POWER, REMAINDER, ROUND, SIGN, TRUNC
  - Character Functions: CHR, CONCAT, INITCAP, LOWER, LPAD, LTRIM, NLSSORT, REPLACE, RPAD, RTRIM, SUBSTR, TRIM UPPER, ASCII, INSTR, LENGTH
  - Date/Time Functions: ADD_MONTHS, CURRENT_DATE, CURRENT_TIMESTAMP, LOCALTIMESTAMP, MONTHS_ BETWEEN, ROUND, SESSIONTIMEZONE, SYSDATE, SYSTIMESTAMP, TO_CHAR (DATETIME), TO_TIMESTAMP, TO_TIMESTAMP_TZ, TRUNC(DATE)

- Conversion Funtions : ASCIISTR, CAST, CONVERT, HEXTORAW, RAWTOHEX, TO_CHAR (character), TO_CHAR (DATETIME), TO_CHAR (NUMBER), TO_DATE, TO_NUMBER
- Case Conversion functions: LOWER, UPPER AND INITCAP
- Conditional Functions : DECODE, CASE

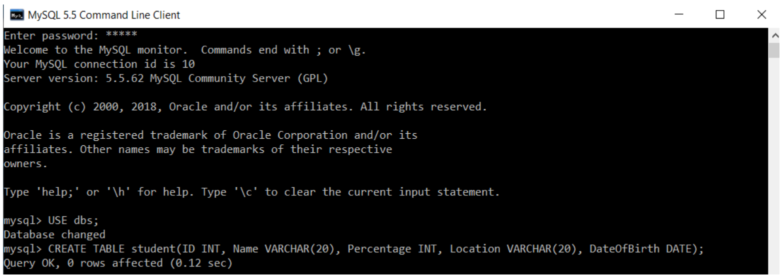
- Multi-Row SQL Functions: AVG, COUNT, MAX, MIN, SUM
- Analytic Functions: WM_CONCAT, LAG, LEAD, RANK, DENSE_RANK

Sub Queries

- What is Sub Query?
- Inline Sub Query, Scalar Sub Query
- Scalar Sub Query Caching
- Invoke the WITH Clause
- The Recursive WITH Clause

Multi Row Insert, Set Operators

- INSERT ALL, INSERT using Sub Query
- Conditional FIRST INSERT
- Set Operators: UNION, UNION ALL, INTERSECT, MINUS



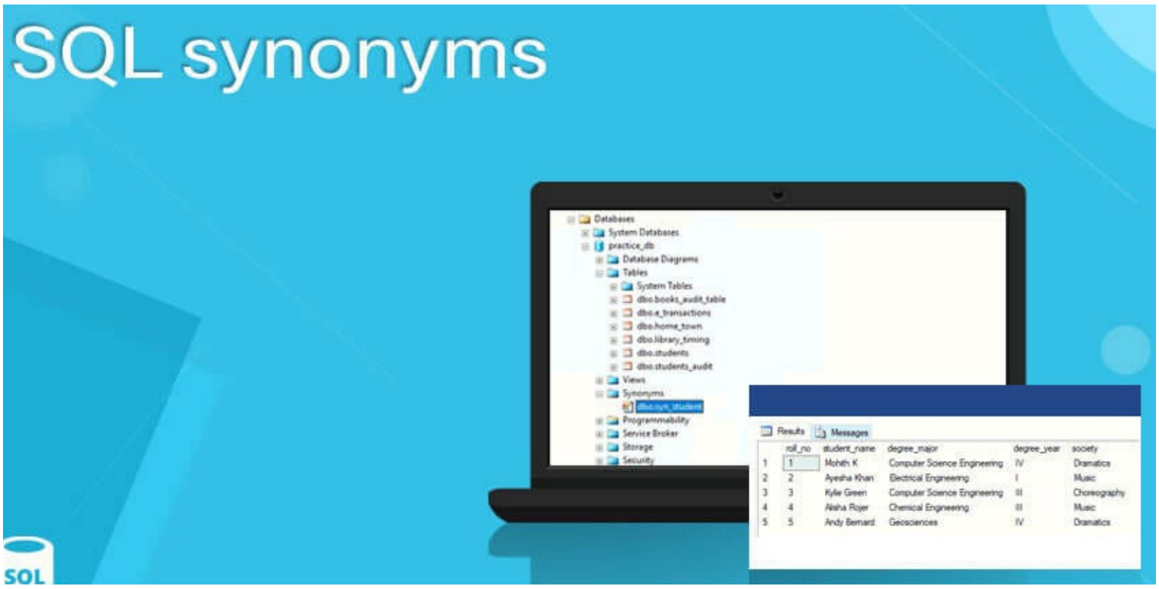
```
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 5.5.62 MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> USE dbs;
Database changed
mysql> CREATE TABLE student(ID INT, Name VARCHAR(20), Percentage INT, Location VARCHAR(20), DateOfBirth DATE);
Query OK, 0 rows affected (0.12 sec)
```

Hierarchical Data Retrieval, Nth row

- Connect By Prior, Connect By ROWNUM, Connect By Level
- START WITH
- Nth row using ROWNUM, RANK & DENSE_RANK Functions

Views, Materialized View, Synonyms, Sequence

- Simple Views and Complex
- DML Operations on a View
- WITH CHECK OPTION, WITH READ ONLY
- Inline Views
- Materialized View – Create, Refresh, Drop - Usage
- Synonym Creation, Granting Access to Synonyms
- Sequence Creation, CURRVAL, NEXTVAL of Sequence



	roll_no	student_name	degree_major	degree_year	society
1	1	Mohith K	Computer Science Engineering	IV	Dramatics
2	2	Ayesha Khan	Electrical Engineering	I	Music
3	3	Kylie Green	Computer Science Engineering	III	Choreography
4	4	Alisha Rojer	Chemical Engineering	III	Music
5	5	Andy Bernard	Geosciences	IV	Dramatics

Python Programming

Complementary Value Add Course(Online)

Introduction to Python

- What is Python and history of Python
- Unique features of Python
- Python-2 and Python-3 differences
- Installing Python
- Setup Python Development Environment

Important Programming Basics in Python

- Python Keywords and Indentation
- Comments
- Python Basic Data Types
- Python Variables
- Operators in Python
- Strings in Python
- Getting User Input
- First Python Program

Loops & Control Statements

Control Structures

- Simple if
- if-else
- nested if
- If-elif-else

Loops

- for loop
- while loop

Break & Continue Statements

Functions, Modules & Packages in Python

- Python user defined functions
- Defining and calling functions.
- Function parameters
- Function scope and global vs local variables
- Lambda functions
- Anonymous functions
- Creating and using modules
- Importing modules and namespaces
- The __name__ and __main__ keywords
- Creating and using packages
- Using the standard library and external libraries.

Data Structures in Python

- Lists in Python
- Lists as Stacks
- Lists as Queues
- Tuples in Python
- Understanding Del statement
- Understanding Iterators
- Generators, Comprehensions and Lambda

- Expressions
- Understanding and using Ranges
- Python Dictionaries
- More on Dictionaries
- Sets
- Python Sets Examples

Exception Handling in Python

- Raising Exceptions
- Handling Exceptions
- Creating custom Exceptions
- Using try
- Using except
- Using finally

Multithreading in Python

- Creating Threads
- Thread synchronization
- Thread pools
- Multiprocessing Module

File Handling (I/O) in Python

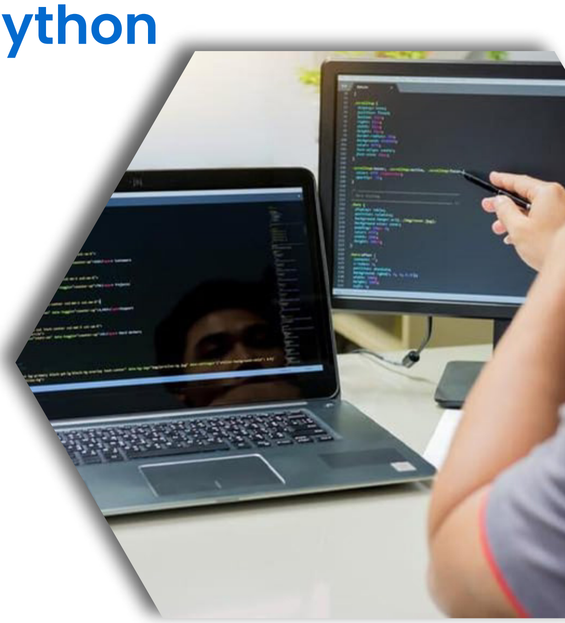
- Reading and writing text files
- Writing Text Files
- Appending to Files and Challenge
- Writing Binary Files Manually
- Using Pickle to Write Binary Files

```
path:
    self.file = open(os.
    self.file.seek(0)
    self.fingerprints.update

@classmethod
def from_settings(cls, settings):
    debug = settings.getbool('DUPEF
    return cls(job_dir(settings), d

def request_seen(self, request):
    fp = self.request_fingerprint(
    if fp in self.fingerprints:
        return True
    self.fingerprints.add(fp)
    if self.file:
        self.file.wri
```

Collections in Python

- Understanding the basics of Collections

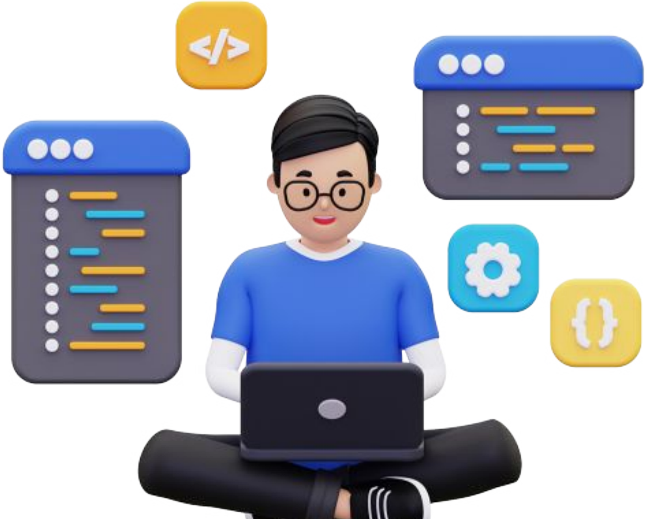
Object Oriented Programming in Python



- Understanding OOPS in Python
- Defining and using Classes
- Defining and using Objects
- Mastering Encapsulation in Python
- Mastering Inheritance in Python
- Mastering Polymorphism in Python
- Accessing attributes
- Built-In Class Attributes
- Destroying Objects

Python Regular Expressions

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



Database Connectivity in Python

- Understanding relational databases
- Understanding the role of SQL
- Creating and connecting to databases using MySQL or Oracle
- Understanding the concept of tables
- Understanding fields and primary keys
- Creating tables
- Inserting data into tables
- Querying data using SQL
- Delete records from tables
- Error handling

Network Programming

- Introduction to Sockets
- Understanding Clients
- Understanding Server
- Handling http requests

Capstone Project 1

Web scraping using Python Script

A web scraping project at FITA Academy will involve extracting data from one or more websites using a script written in Python. The project typically involves identifying the target website, analyzing the structure of the web pages to be scraped, and using Python libraries to extract the relevant data.

The extracted data can then be stored in a structured format such as CSV or similar formats. Web scraping projects can be used for a range of applications, such as collecting data for research or business intelligence, monitoring online trends and sentiment, or creating customized data feeds for websites or applications.

Capstone Project 2

Excel Automation using Python Programming

The Excel automation project will involve the win32com module to automate the Excel application. The project will involve opening an existing Excel file, modifying the data in the file, and saving the changes.

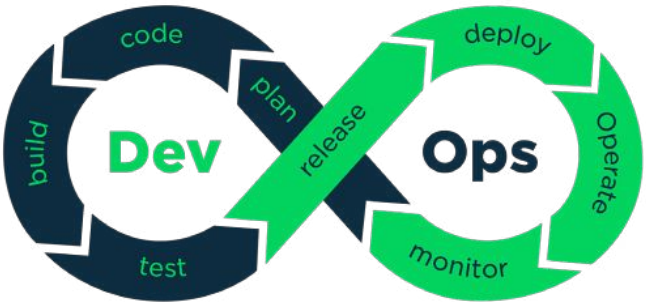
Some common steps students will practice in this capstone project will include

1. Installing and importing the win32com module for working with the Excel application.
2. Opening an existing Excel file using the win32com module and navigating to a specific worksheet.
3. Modifying the data in the worksheet using Python code, such as updating values, formatting cells, or inserting new rows or columns.
4. Saving the changes to the Excel file using the Save or SaveAs method.
5. Closing the Excel application using the Quit method.

DEVOPS TOOLS

Complementary Value Add Course(Online)

MODULE 1: INTRODUCTION TO DEVOPS



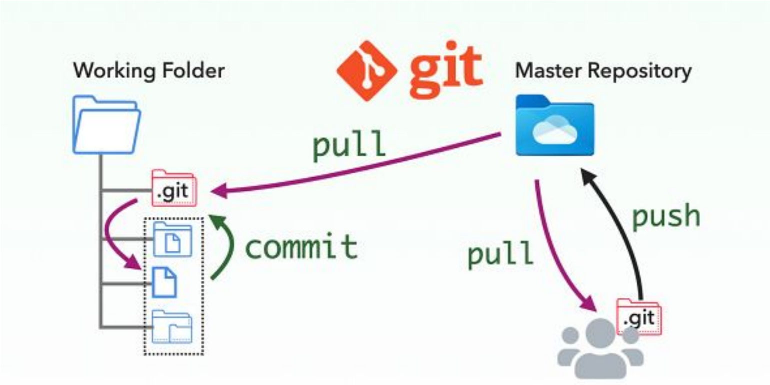
- What is DevOps
- Why DevOps
- DevOps Principles
- DevOps Ecosystem
- Opportunities for DevOps Engineer
- DevOps Skills in demand
- Important tools used in DevOps

MODULE 2: LINUX ADMINISTRATION

- Basic concepts of Operating System
- Linux Versions
- Important Linux Operating Systems concepts like Kernel, Shell & File System structure
- Important Linux Commands for Administration
- Commands for User Management
- Commands for File Permissions
- Commands for Partitioning
- Commands for File System
- Package Management
- Networking essentials
- SSH configuration

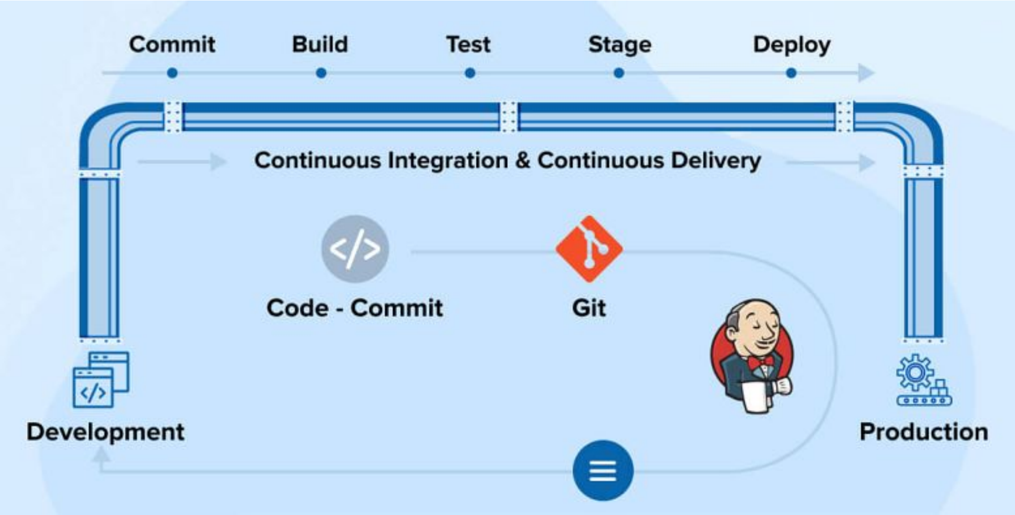
MODULE 3: SOURCE CONROL USING GIT

- Introduction to Version Control
- Introduction to GIT
- Installation and Server setup
- Important Git Commands
- Working with Repositories
    - Creating a repository (git init)
    - Checking status (git status)
    - Adding files to a repository (git add)
    - Committing files (git commit)
    - Removing staged files (git reset)
    - Removing committed files (git rm)
    - Checking logs (git log)
- GIT Remote Repositories
- Branching in GIT
- Merging in GIT
- Workflows in GIT
    - Different ways of using Git
    - Centralised
    - Feature Branch
    - Gitflow Workflow
    - Forking Workflow
- Working with GitHub



MODULE 4: CONTINOUS INTEGRATION USING JENKINS

- Introduction to Continuous Integration
- Continuous Integration with Jenkins Overview
- Jenkins Architecture
- Installation of Jenkins
- Jenkins management
- Support for the Git version control systems
- Different types of Jenkins Jobs
- Setting up a Jenkins job
- Scheduling build Jobs
- Securing Jenkins
- Jenkins Plugins
- Introduction to Maven
- Setting up Jenkins Master and Slave servers
- Distributed builds with Jenkins
- Backup and Restore of Jenkins



Aptitude

Complementary Value Add Course(Online)

QUANTITATIVE APTITUDE TOPICS

- Number system
  - Divisibility rule
  - Unit Digit
  - Remainder Theorem
  - Sum of Series
  - Consecutive Numbers
  - Decimal to Fraction
  - Find a Numbers
  - Arithmetic progress and Geometric progress
  - Find the Place and Face value
  - Least Common Multiples
  - Highest Common Factors
  - Successive Division
  - Number value
  - miscellaneous questions
- Time and work
- Ratio and proportion
- Problems on ages
- Percentage
- Profit and loss
- Simple interest and compound interest
- Probability
- Alligation and mixture
- Time and distance

LOGICAL REASONING TOPICS

- Blood relation
- Direction
- Seating arrangements
- Dices and cubes
- Syllogism
- Coding and decoding
- Element series

VERBAL ABILITY

- Parts of speech
- Persons
- Verbs classification
- Usage of verb
- Tenses
- Sentence formation
- Preposition
- Article

Spoken English & Softskills

Complementary Value Add Course(Online)

Module 1: Introduction and Project Presentation

Objective: To build confidence in introducing oneself and presenting projects.

- Self-introduction and project presentation
- Assessment and feedback
- Correction and repetition exercises

Module 2: Common Verbs and Verb Forms

Objective: To familiarize students with commonly used verbs and their various forms.

- Introduction to commonly used verbs
- Verb forms: V1 (Base), V2 (Past), V3 (Past Participle), V4 (Present Participle/Gerund)
- Practice exercises using different verb forms

Module 3: Basic Sentence Structures and Tenses

Objective: To understand basic sentence structures and the usage of commonly used tenses.

- Simple sentence structures
- Usage of tenses: simple past, simple present, present continuous, simple future, and present perfect
- Practice forming sentences in different tenses

Module 4: Practising Commonly Used Tenses

Objective: To practise using commonly used tenses with different subjects.

- Exercises with various subjects in different tenses
- Tense-wise Q&A for commonly used tenses
- Conversational dialogues using different tenses

Module 5: Narration and Storytelling

Objective: To practise narration of simple day-to-day incidents and stories.

- Narrating personal experiences
- Storytelling exercises
- Feedback and corrections

Module 6: Applying Tenses in Interviews

Objective: To apply knowledge of tenses in mock interview settings.

- Mock interviews focusing on tense usage
- Feedback and improvement

Module 7: Vocabulary for Official Communication

Objective: To enhance vocabulary for effective official communication.

- Narrating personal experiences
- Storytelling exercises
- Feedback and corrections

Module 8: Interpersonal Communication Skills

Objective: To improve interpersonal communication and time management skills.

- Exercises on interpersonal communication
- Time management techniques
- Role plays and practical applications

Module 9: Formal English and Email Etiquette

Objective: To understand and practise formal English and email writing etiquette.

- Basics of formal English
- Email writing etiquette
- Practice writing formal emails

Module 10: Role Plays and Mock Interviews

Objective: To build confidence through role plays and mock interviews.

- Various role play scenarios
- Mock interview sessions
- Feedback and improvement

Module 11: Interview Preparation

Objective: To prepare for real-world interviews through practice and feedback.

- Comprehensive interview preparation
- Mock interviews with detailed feedback
- Improvement exercises

Intensive Placement Workshop

Follow FITA Academy in Instagram for More Updates

@fita_academy