The IoT Academy

Connecting The Unconnected

Full Stack Java

Our Vision

To empower individuals and businesses with the knowledge and strategies required to thrive in the dynamic world of digital marketing.

Imagine a future where everyone, from small business owners to creative individuals, can confidently navigate the digital world. Our vision is to make this happen. We want to empower you with the skills and knowledge needed to succeed in the ever-changing digital landscape. In other words, we're here to help you become a digital superstar!

Our Mission

To provide a comprehensive and hands-on digital marketing education that equips our students with the skills, confidence, and expertise needed to excel in the digital landscape. Our mission is like our roadmap. It guides us in achieving our vision. We're on a mission to provide you with a comprehensive digital marketing education that's fun, practical, and effective. We want you to gain the confidence to excel in the digital world. Whether it's through our courses, mentorship, or hands-on experience, we're here to support you on your journey to digital success. Your success is our mission!

Introduction to WEB

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

HTML

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

Core CSS

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

Advance CSS

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

Javascript (ES-5 and ES-6)

- Basic JavaScript
- DOM and BOM
- Intervals and Objects
- Prototype ,Hoisting and Closure
- Let , Const , Arrow , Class and Inheritance
- Map, Filter , Reduce , Template Literals
- forEach, for-in, for-of loop

Typescript

- Why Typescript
- Basic Types
- Class and Interfaces
- Modules

Introductin To React JS

- What is React JS?
- What is SPA?
- DOM vs Virtual DOM
- Advantages and Disadvantages
- Key Features

Environmental Setup

- Node | NPM
- Installation of CLI
- Setup Project
- Directory Structure
- Code Editors
- How React JS Application Boot

Basic Features Of React JS

- React Concepts
- JSX and TSX
- Render Elements
- Function and Class Components
- Props and State
- Handling Events
- Dynamic Data Rendering
- Property Binding

Key Features Of React JS

- Conditional Rendering
- List and Keys
- Forms Handling
- Forms Validations

Component Lifecycle Hook

- Understanding component lifecycle
- All Lifecycle Hooks

Event Handling React

- Understanding React Event System
- Passing arguments to event Handlers

Network Call

- Fetch
- Axios

Custom Services

- Introduction to Services
- Building a Service

Local Data Storage

- Local Storage
- Session Storage
- Cookies

Routing With React Router

- Setting up React Router
- Configuring route with
- Route Component
- Making routes dynamic with Route Params
- Working with nested routes
- Link and NavLink
- Redirect Routes

UI Components

- Angular Material
- PrimeNG

Introduction To REDUX

- Why Redux
- Install and setup
- Store ,Reducer , actions
- Dispatcher
- High order Components
- mapStateToProps and
- mapDispatchToProps usage

Advance REDUX

- Async Actions
- Middleware
- Redux Thunk and Redux Saga

React Hooks

- Why We Need Hooks.
- Different Types Of Hooks
- Using State And Effect Hooks
- Usereducer , Useref Etc.
- Custom Hooks
- Rules Of Hooks

Third Party Modules

- Social Login
- Pagination
- Search
- Filter
- JWT Token
- File Upload
- Many More

Develop a CRU Application in React JsApplication Deployment

- Build Application and Deployment

Introduction To JAVA

- Why Java was Developed
- Application Areas of Java
- History of Java
- Platform Independency in Java
- USP of Java: Java Features
- Sun-Oracle Deal
- Different Java Platforms
- Difference between JDK, JRE,JVM
- Java V ersions
- JVM Architecture

Creating First Java Program

- Understanding Text Editors to Write Programs
- How to compile java file
- Byte Code and class file
- How to run class file

Java Language Fundamentals

- Identifiers
- Keywords
- Variables
- Literals
- Data Types
- Operators
- Comments
- Looping Statements
- Condition Statements
- Type Casting

OOP Implementation (PIE)

- Why OOP
- OOP Concepts with Real life examples
- Class& it's Syntax
- Object& it's Syntax
- Reference Variable
- Constructors
- Instance(Non-Static) & Static Variables
- Instance(Non-Static) & Static Methods
- this Keyword and it's usages
- Object & Static Initializers (Anonymous Blocks)
- Inheritance& it's Syntax
- Types of Inheritance
- Object Class as Root of Java Class Hierarchy
- Variable Hiding
- Method Hiding
- Method Overriding
- Method Overloading
- Super keyword and it's usages
- Final keyword and it's usages
- Constructor Chaining
- Upcasting and Downcasti--ng
- Static &Dynamic Binding
- Run Time Polymorphism
- Abstract Keyword(Abstract classes and methods)
- Understanding Interfaces
- Implementation of Encaps--ulation
- Association with Impleme--ntation

Packages

- Understanding Packages
- Setting Class path
- Reading Input from Keyboard
- Access Modifiers
- With in Package & Outside Package Implements

Nested Types

- Static Nested Class
- Non-static Nested Class
- Local Class
- Anonymous Class
- Nested Interface

Arrays

- General Definition of Array
- Advantages from Array
- Arrays in Java
- 1-d Arrays
- 2-d Arrays
- Jagged Arrays
- Array of reference type
- Operations on Arrays
- User Define Array & Object Tpye

Command Line Arguments And Wrapper Classes

- How to read command line arguments
- Wrapper Classes
- Parsing of Numeric Strings
- String representation of Primitives

Exception Handling

- Types of Runtime Errors.
- Understanding Exceptions
- Exception Class Hierarchy
- Try & Catch Blocks
- Patterns of Catch Block
- Nested Try statements
- Throw, throws and finally
- Creating Custom Exception
- Checked & Unchecked Ex--ceptions
- Assertion

Working With Strings

- What is String
- String Class
- Creating String Object
- Operations on String
- String Buffer Class and it's Methods
- Difference between String and StringBuffer class
- String Builder Class and it's Methods
- Difference between String Buffer and StringBuilder

Swing

- Introduction to AWT
- Introduction to Swing Components
- Look And Feel of Swing Components
- MVC Architecture of Swing Components
- Working with Image
- Advance Swing Compone--nts
- JOptionPane.JTree,JTable,
- JTabbedPane
- JfileChooser, JcolorChoos--er
- Menu Components
- JMenu
- JMenuItem
- JMenubar

Multithreaded Programming

- Multitasking: Why
- Concurrent Execution
- Multiprocessing v/s Multithreading
- Main Thread (Default Java Thread)
- Creating Child Threads and understanding context switching
- Thread States
- Thread Group
- Thread Synchronization: Methods and Blocks
- Inter-Thread communication
- Daemon Threads
- Deadlock

I/O Streams

- What is I/O
- Why Need Streams
- Byte Streams and Character Streams
- Read/Write operations with file
- Scanner Class
- Object Serialization & Deserialization
- Transient keyword
- File Class and it's Methods

Socket Programming

- Understanding Fundamentals of a Network
- Socket and ServerSocket Classes
- InetAddress Class
- DatagramSocket and DatagramPacket Classes
- URL, URLConnection, HttpURLConnection Classes

Reflection

- Understanding the Need Of Reflection
- Getting information about class's modifiers, fields, methods,
- constructors and super classes
- Finding out constant and method declaration belong to an interface
- Creating an instance of the class whose name is not known until runtime
- Getting and setting values of an object's field if field name is unknown until runtime
- Invoking a method on an object if the method is unknown until runtime
- Invoking Private Methods

Extended & Utility Concepts

- Generics
- Lambda Expression
- Annotations
- Object Cloning
- Vargs
- Static-import
- Enum
- Static, Default and Private Methods of Interface Var Type
- Java Modules
- Stream API

Collections Framework

- What is Collection?
- What is Framework?
- Collections Framework
- Core Interfaces
- Collection, List, Queue, Deque
- Set, NavigableSet,
- SortedSet
- Map, NavigableMap, SortedMap
- Core Classes
- ArrayList, LinkedList, Priority Queue, ArrayDeque
- HashSet, LinkedHas Set,
- TreeSet,
- HashMap, IdentityHashMap WeakHashMap, LinkedHa--shap,
- Tree Map
- Accessing a Collection via an Iterator
- Accessing List via ListIte--rator
- Accessing a Collection via for each loop
- Working with User Defined Objects
- The Comparator and Comparable Interfaces
- The Legacy classes and Interfaces
- Enumeration, Vector,Stack Hashtable, Properties

Date & Time API

- java.util.Date
- java.util.Calender
- java.sql.Date

System Properties & Internatinalization (I18N)

- Understanding Locale
- Resource Bundle
- Usage of properties file
- Fetching text from
- Resource Bundle
- Displaying the text in HINDI
- Displaying date in Hindi

Introduction To SQL (Project Based)

Database Programming Using JDBC

- Need Of JDBC
- JDBC Drivers
- Statement, PreparedStatement, CallableStatement
- Scrollable and Updatable ResultSet
- Batch Updates
- Transaction
- Metadata
- Connection Data Base
    - Oracle
    - My SQL
    - Mongo DB

Java EE (Java Platform Enterprise Edition)

- Understanding the Concept of Java EE: JEE Specification
- Java EE Architecture
    - Single Tier
    - Two Tier
    - Three Tier
    - N-Tier
- Java EE Components
- Web Components
- Distributed(Business) Components
- Java EE Containers & Servers
- Web Container& Web Server(Apache Tomcat)
- EJB Container& Application Server(Weblogic, Glassfish, Websphere)
- Java EE Services
    - JNDI Service
    - Java Transaction Service
    - JAAS
    - JMS

Java Servlet

- Introduction to web programming
- Role of Servlet in web programming
- Servlet Lifecycle
- Servlet with Annotations
    - @WebServlet
    - @WebInitParam
    - @WebListener
    - @WebFilter
    - @MultipartConfig
- Request Dispatching
- Parameters & Attributes and their differences
- ServletConfig and Servlet Context
- File Uploading and Downloading
- Session Tracking&State Management
    - Cookie
    - Url Rewriting
    - Hidden Form Field
    - Session Object
- Events & Listeners
    - Dependency Injection
    - Filters

Java Server Pages (JSP) & JSTL

- JSP Architecture
- JSP Elements
- JSP Objects
- Understanding JavaBeans
- Custom Tags
- Using tags of JSTL
- Expression Language

Project Classes

- Front End Coding
- FORM DESIGNING
  - HTML
  - CSS
  - JAVA SCRIPT
  - BOOTSTRAP
- Back End Coding
- DATABASE DESIGNING
- Connecting forms to database
- Writing Business Logic
- Project Hosting

Design Pattern

- Why Design Patterns...?
- Front Controller
- Composite View
- Factory Pattern
- Singleton Pattern
- DAO Pattern

Java Mail API

- Email System and Protocols
- Sending & Receiving Mails
- Handling Attachments

Intoduction To Distributed Programming

- RMI
- Web Services

Introduction To Restfull Services

- @PathParam
- @Path
- @FormParam
- @QueryParam
- @DefaultValue

Overview Of JPA Framework

Spring

Spring

- What is Spring?
- Spring modules
- Understanding dependency Injection
- Applying aspect-oriented programming

Maven Deployment

- Maven Configuration
- Converting Maven to Eclipse
- Various Maven Command

Basic Bean Wiring

- Containing your Bean
- Creating bean
- Injecting into bean properties
- Auto wiring
- Controlling bean creation
- Aspect Oriented Programming

Hitting The Database

- Learning spring's data
- Access Philosphy
- Configuring a data source
- Using JDBC with Spring
- Working with JDBC Templates
- Using Spring's DAO Support Classes for JDBC
- Integrating Hibernate with Spring
- Caching

HIBERNATE

Introduction To ORM

- Need of ORM
- Problems using JDBC Directly
- ORM Implementation

Introduction To Hibernate

- Hibernate Architecture
- Hibernate configuration
- Hibernate's Support for Other Technologies
- Installing Hibernate
- A "Hello world" stand alone application

Creating Persisting Classes

- Mapping a basic Java Class
- Mapping a Class with Binary Data
- Mapping a Serializable Class
- Mapping a class with Data/ calendar attributes
- Mapping a Read-only class
- Mapping a class using Versioning /Timestamps

Mapping Inheritence With Java Classes

- Table-Per-class Hierarchy Mapping
- Table-Per-subclass Hierar--chy Mapping
- Table-Per-concrete-subcla--ss Hierarchy Mapping
- Persistence interfaces

Working With Collections

- Associations
- Lazy initialization
- Mapping Maps/Sorted Maps
- Mapping Sets/Sorted Sets
- Mapping listst
- Mapping Arrays
- Mapping a Bidirectional Association

Hibernate Caching

- How caching improves performence
- First level lache
- Second level cache

SPRING BOOT

- Introduction To Spring Boot
- Spring Boot Annotation
- Spring Boot & JDBCTemp--lete
- Spring Boot & JPA Hibern--ate
- Spring Boot Rest API
- Spring Boot MVC
- Spring Boot Security
- Introduction to Micro Serv--ices

THANK YOU

Contact Information:

Email: info@theiotacademy.co
Phone: +91-9354068856
Website: www.theiotacademy.co