LAPT

LONDON ACADEMY OF PROFESSIONAL TRAINING

CERTIFICATE

Adv Certificate in Full-Stack Development & API Integration

Practitioner Level

LAPT — London Academy of Professional Training

Address 85 Great Portland Street, W1W 7LT, London, United Kingdom

Email info@lapt.org

Telephone +44 7513 283044

Web lapt.org

Reference IT-SDA-P • IT Centres

UKPRN 10067351 • Registered in England and Wales, company 4984798

6	30	180	440	60%	Lifetime
SUBJECTS	CHAPTERS	CLASSES	TOTAL MARKS	PASS MARK	VALIDITY

AT A GLANCE

REFERENCE	**IT-SDA-P**
AWARD	**Certificate**
ASSESSMENT	**440 marks — 264 to pass (60%)**
PREREQUISITES	**None**
VALIDITY	**Lifetime**
AWARDING BODY	**LAPT — London Academy of Professional Training**

CURRICULUM

1 Advanced JavaScript Frameworks

5 chapters · 30 classes · **90 marks**

- **1 Mastering Modern JavaScript Syntax and ES6+ Features** — 6 classes
  - 1.1 Exploring ES6+ Syntax and Features
  - 1.2 Understanding Let, Const, and Block Scoping
  - 1.3 Harnessing Arrow Functions and Implicit Returns
  - 1.4 Mastering Template Literals and String Interpolation
  - 1.5 Utilizing Destructuring for Arrays and Objects
  - 1.6 Implementing Async/Await for Asynchronous Programming
- **2 Exploring the Core Concepts of React.js** — 6 classes
  - 2.1 Understanding JSX: Bridging HTML and JavaScript
  - 2.2 Building Components: The Backbone of React
  - 2.3 Managing State: Enhancing Interactivity
  - 2.4 Handling Events: Capturing User Actions
  - 2.5 Implementing Props: Facilitating Component Communication
  - 2.6 Utilizing React Hooks: Functional Programming in React
- **3 Building Interactive User Interfaces with Vue.js** — 6 classes
  - 3.1 Understanding Vue.js Basics and Its Core Concepts
  - 3.2 Setting Up the Development Environment for Vue.js Projects
  - 3.3 Creating Reactive Data Bindings and Vue Instance
  - 3.4 Designing Dynamic Components and Vue Templates
  - 3.5 Implementing Vue Directives for Enhanced Interactivity
  - 3.6 Integrating Vue.js with External APIs for Dynamic Data
- **4 Harnessing the Power of Angular for Complex Applications** — 6 classes
  - 4.1 Exploring Angular Architecture and Core Concepts
  - 4.2 Building Dynamic Interfaces with Angular Components
  - 4.3 Managing Data with Angular Services and Dependency Injection
  - 4.4 Navigating Complex Applications Using Angular Routing
  - 4.5 Optimizing State Management with Angular Forms and Reactive Programming
  - 4.6 Integrating Third-Party Libraries and APIs in Angular
- **5 Integrating RESTful APIs with Advanced Frameworks** — 6 classes
  - 5.1 Understanding RESTful API Concepts
  - 5.2 Setting Up API Requests in Advanced Frameworks
  - 5.3 Implementing Authentication for Secure API Access
  - 5.4 Parsing and Managing API Responses
  - 5.5 Error Handling in API Integration
  - 5.6 Optimizing Performance in Framework-API Communication

2 API Design and Integration

5 chapters · 30 classes **90 marks**

- **1 Understanding APIs: Types and Uses in Modern Software** — 6 classes
  - 1.1 Exploring API Fundamentals and Principles
  - 1.2 Identifying Different Types of APIs
  - 1.3 Examining RESTful API Architecture
  - 1.4 Understanding GraphQL and Its Applications
  - 1.5 Analyzing Real-world API Use Cases
  - 1.6 Evaluating API Integration Methods and Strategies
- **2 Building RESTful APIs: Best Practices and Protocols** — 6 classes
  - 2.1 Understanding RESTful API Architecture
  - 2.2 Identifying and Defining API Resources
  - 2.3 Designing Consistent API Endpoints
  - 2.4 Implementing HTTP Methods Effectively
  - 2.5 Securing RESTful APIs with Best Practices
  - 2.6 Testing and Debugging RESTful APIs
- **3 Advanced API Authentication and Security** — 6 classes
  - 3.1 Understanding Advanced Authentication Protocols
  - 3.2 Implementing OAuth2 for Enhanced Security
  - 3.3 Securing APIs with JWT and Token Verification
  - 3.4 Applying Role-Based Access Control (RBAC) in APIs
  - 3.5 Enforcing Rate Limiting and Quotas for API Protection
  - 3.6 Monitoring and Auditing API Security Postures
- **4 Integrating APIs: Connecting Services and Data** — 6 classes
  - 4.1 Exploring API Fundamentals and Concepts
  - 4.2 Analyzing RESTful API Architecture
  - 4.3 Configuring API Endpoints for Seamless Integration
  - 4.4 Authenticating and Securing API Interactions
  - 4.5 Implementing API Calls in Web Applications
  - 4.6 Troubleshooting and Optimizing API Performance
- **5 Testing and Documenting APIs: Ensuring Reliability** — 6 classes
  - 5.1 Understanding the Importance of API Testing
  - 5.2 Exploring Different Types of API Testing
  - 5.3 Implementing Automated API Testing
  - 5.4 Designing Effective API Test Cases
  - 5.5 Creating Comprehensive API Documentation
  - 5.6 Utilizing Tools for API Documentation and Testing

3 Back-End Development with Node.js

5 chapters · 30 classes **65 marks**

- **1 Node.js Basics and Setup** — 6 classes
  - 1.1 Understanding Node.js: An Introduction to Its Architecture and Features
  - 1.2 Installing Node.js: Setting Up Your Development Environment
  - 1.3 Exploring Node.js Modules: Built-in and Third-Party Options
  - 1.4 Running Your First Node.js Script: Basic Commands and Execution
  - 1.5 Managing Node.js Packages: Using npm for Dependencies
  - 1.6 Utilizing Node.js REPL: Interactive Coding and Debugging
- **2 Core Modules and File System Operations** — 6 classes
  - 2.1 Understanding Node.js Core Modules
  - 2.2 Exploring the File System Module
  - 2.3 Reading Files with Node.js
  - 2.4 Writing and Updating Files in Node.js
  - 2.5 Managing Directories and Paths
  - 2.6 Error Handling in File System Operations
- **3 Building HTTP Servers and RESTful APIs** — 6 classes
  - 3.1 Understanding HTTP: The Basics of Web Communication
  - 3.2 Setting Up a Node.js Server: First Steps
  - 3.3 Handling HTTP Requests and Responses in Node.js
  - 3.4 Building RESTful APIs: Designing Resource Endpoints
  - 3.5 Implementing CRUD Operations in RESTful Services
  - 3.6 Securing and Testing Node.js APIs for Robustness
- **4 Database Integration with Node.js** — 6 classes
  - 4.1 Understanding Databases in Node.js Context
  - 4.2 Setting Up a Local Database Environment
  - 4.3 Connecting Node.js to Your Database
  - 4.4 Performing CRUD Operations in Node.js
  - 4.5 Implementing Data Models and Schemas
  - 4.6 Handling Database Errors and Optimizing Queries
- **5 Security Practices and Deployment Strategies** — 6 classes
  - 5.1 Understanding Security Fundamentals in Node.js
  - 5.2 Implementing Authentication and Authorization
  - 5.3 Securing Data with Encryption Techniques
  - 5.4 Managing Server and Database Security
  - 5.5 Optimizing Node.js Applications for Deployment
  - 5.6 Deploying and Monitoring Node.js Applications

4 Database Management and Optimisation

5 chapters · 30 classes **65 marks**

- **1 Relational Database Fundamentals** — 6 classes
  - 1.1 Understanding Data Models
  - 1.2 Exploring Primary and Foreign Keys
  - 1.3 Designing Relational Schemas
  - 1.4 Implementing Normalization Techniques
  - 1.5 Executing SQL Queries for Data Retrieval
  - 1.6 Optimizing Relational Database Performance
- **2 Advanced SQL and Query Optimization** — 6 classes
  - 2.1 Understanding Subqueries: Enhancing SQL Queries with Nested Statements
  - 2.2 Exploring Joins: Mastering Complex Data Relationships
  - 2.3 Implementing Indexes: Boosting Query Performance
  - 2.4 Analyzing Execution Plans: Diagnosing and Optimizing Queries
  - 2.5 Utilizing Window Functions: Advanced Data Analysis Techniques
  - 2.6 Refining Queries: Techniques for Reducing Latency and Improving Efficiency
- **3 Database Normalization and Indexing** — 6 classes
  - 3.1 Understanding Database Normalization Principles
  - 3.2 Exploring the Benefits of Normalization
  - 3.3 Identifying Common Normal Forms
  - 3.4 Implementing Normalization Techniques
  - 3.5 Introduction to Database Indexing Concepts
  - 3.6 Applying Indexing for Performance Optimization
- **4 Transactions and Concurrency Control** — 6 classes
  - 4.1 Understanding Database Transactions
  - 4.2 Ensuring Atomicity in Transactions
  - 4.3 Implementing Isolation Levels
  - 4.4 Exploring Concurrency Control Techniques
  - 4.5 Managing Deadlocks and Data Locks
  - 4.6 Optimising Performance with Concurrency Control
- **5 Distributed Databases and Scaling** — 6 classes
  - 5.1 Exploring Distributed Database Concepts
  - 5.2 Understanding Data Partitioning Techniques
  - 5.3 Implementing Horizontal Scaling Strategies
  - 5.4 Managing Data Consistency Across Systems
  - 5.5 Optimizing Query Performance in Distributed Databases
  - 5.6 Troubleshooting Common Distributed Database Issues

5 DevOps and Continuous Integration

5 chapters · 30 classes **65 marks**

- **1 Understanding DevOps and Its Role in Modern Software Development** — 6 classes
  - 1.1 Exploring the Fundamentals of DevOps
  - 1.2 Identifying Key DevOps Practices and Tools
  - 1.3 Analyzing the Benefits of Continuous Integration
  - 1.4 Implementing DevOps Culture in Software Development
  - 1.5 Integrating Automated Testing in CI/CD Pipelines
  - 1.6 Evaluating DevOps Success through Metrics and Feedback
- **2 Continuous Integration: Tools and Best Practices** — 6 classes
  - 2.1 Understanding Continuous Integration in DevOps
  - 2.2 Exploring Popular CI Tools: Jenkins, Travis CI, CircleCI
  - 2.3 Setting Up Your First CI Pipeline
  - 2.4 Integrating Automated Testing in CI
  - 2.5 Analyzing Best Practices for CI Success
  - 2.6 Troubleshooting Common CI Issues
- **3 Automating Deployment and Infrastructure with Continuous Delivery** — 6 classes
  - 3.1 Understanding Continuous Delivery Concepts and Benefits
  - 3.2 Exploring Tools for Continuous Deployment
  - 3.3 Setting Up Automated Deployment Pipelines
  - 3.4 Implementing Infrastructure as Code with Terraform
  - 3.5 Integrating CI/CD with Containerization Technologies
  - 3.6 Monitoring and Optimizing Automated Deployments
- **4 Monitoring, Logging, and Feedback Loops in DevOps** — 6 classes
  - 4.1 Understanding the Importance of Monitoring in DevOps
  - 4.2 Exploring Key Metrics and Tools for Effective Logging
  - 4.3 Implementing Monitoring Solutions to Enhance System Performance
  - 4.4 Developing Feedback Loops to Improve Development Processes
  - 4.5 Integrating Logging and Monitoring in CI/CD Pipelines
  - 4.6 Analyzing Monitoring Data to Drive Continuous Improvement
- **5 Security and Compliance in DevOps Practices** — 6 classes
  - 5.1 Understanding Security Principles in DevOps
  - 5.2 Identifying Compliance Requirements in DevOps
  - 5.3 Implementing Security Best Practices in CI/CD
  - 5.4 Integrating Automated Security Testing into Pipelines
  - 5.5 Managing Access and Permissions within DevOps Teams
  - 5.6 Monitoring Compliance through DevOps Tools

6 Security in Software Development

5 chapters · 30 classes **65 marks**

1 Understanding Core Security Concepts in Software Development — 6 classes

- 1.1 Exploring the Fundamentals of Security in Software Development
- 1.2 Understanding Threat Landscapes: Identifying Risks in Development
- 1.3 Implementing Authentication and Authorization Mechanisms
- 1.4 Securing Data: Techniques for Data Protection and Encryption
- 1.5 Applying Secure Coding Practices to Prevent Vulnerabilities
- 1.6 Conducting Security Testing and Audits in Software Projects

2 Implementing Secure Coding Practices — 6 classes

- 2.1 Understanding Secure Coding Principles
- 2.2 Identifying Common Vulnerabilities in Code
- 2.3 Implementing Input Validation Techniques
- 2.4 Applying Authentication and Authorization Protocols
- 2.5 Utilizing Secure Data Encryption Methods
- 2.6 Conducting Code Reviews for Security Flaws

3 Advanced Threat Modeling and Risk Assessment — 6 classes

- 3.1 Understanding Advanced Threat Modeling Concepts
- 3.2 Identifying Potential Threats in Software Systems
- 3.3 Analyzing Threats with Risk Assessment Techniques
- 3.4 Designing Effective Mitigation Strategies for Threats
- 3.5 Applying Threat Modeling in Full-Stack Development
- 3.6 Conducting Continuous Risk Assessments in API Integration

4 Integrating Security into DevOps Processes — 6 classes

- 4.1 Understanding the Importance of Security in DevOps
- 4.2 Identifying Common Security Risks in DevOps
- 4.3 Integrating Security Practices in Continuous Integration
- 4.4 Implementing Automated Security Testing in DevOps
- 4.5 Monitoring Security in Continuous Deployment
- 4.6 Building a Security-Focused DevOps Culture

5 Testing and Monitoring for Security Vulnerabilities — 6 classes

- 5.1 Understanding Security Vulnerabilities in Software
- 5.2 Identifying Common Security Threats in Code
- 5.3 Implementing Unit Testing for Security
- 5.4 Conducting Penetration Testing for Web Applications
- 5.5 Utilizing Automated Tools for Vulnerability Scanning
- 5.6 Monitoring Application Logs for Security Indicators