

# JMeter Performance Testing

Duration: 4 days (32 hours)

## Prerequisites

- Basic understanding of Web Applications (HTTP/HTTPS)
  - Basic Software Testing knowledge (optional)
  - No prior JMeter experience required
- 

## DAY 1 – Performance Testing Fundamentals & JMeter Basics

### ◆ Module 1: Introduction to Performance Testing

- Definition and importance
- Types:
  - Load Testing
  - Stress Testing
  - Spike Testing
  - Endurance Testing
- Key Metrics:
  - Response Time
  - Throughput
  - Latency
  - Error Rate
- Performance Testing Life Cycle

### ◆ Module 2: Introduction to Apache JMeter

- Overview and features
- Architecture
- Supported protocols (HTTP, HTTPS, FTP, JDBC, REST APIs)
- Use cases

### ◆ Module 3: Installation & Setup

- JDK setup
- JMeter installation
- GUI overview

### ◆ Module 4: Core Components

- Test Plan
- Thread Groups
- Samplers
- Listeners
- Controllers

- Install JMeter and configure environment
  - Create first test plan
  - Execute HTTP request
  - Analyze results using listeners
- 

## DAY 2 – Test Plan Design & Scripting

### ◆ Module 1: Test Plan Design

- Structure of a test plan
- User behavior modeling
- Workload design

### ◆ Module 2: Recording & Correlation

- HTTP(S) Test Script Recorder
- Recording user scenarios
- Parameterization basics
- Correlation basics

### ◆ Module 3: Advanced Components

- Timers
- Assertions
- Config Elements
- Pre Processors
- Post Processors

### ◆ Module 4: Data-Driven Testing

- CSV Data Set Config
- Dynamic data handling

### Labs

- Record login and navigation scenario
  - Add assertions and validations
  - Parameterize test data using CSV
  - Apply timers for realistic load
- 

## DAY 3 – Test Execution & Analysis

### ◆ Module 1: Types of Testing in JMeter

- Load Testing
- Stress Testing
- Spike Testing
- Endurance Testing

#### ◆ **Module 2: Test Execution**

- GUI vs Non-GUI mode
- Command-line execution
- Test execution strategies

#### ◆ **Module 3: API & Database Testing**

- REST API testing
- SOAP testing
- JDBC testing

#### ◆ **Module 4: Result Analysis**

- Aggregate Report
- Summary Report
- Graph Results
- Interpreting metrics
- Identifying bottlenecks

#### **Labs**

- Execute load test on sample application
- Perform API performance testing
- Run tests in non-GUI mode
- Analyze reports and identify issues

---

## **DAY 4 – Advanced JMeter & Real-World Implementation**

#### ◆ **Module 1: Advanced Scripting**

- Regular Expressions
- Correlation techniques
- Beanshell scripting
- Introduction to Groovy

#### ◆ **Module 2: Monitoring & Reporting**

- Server-side monitoring basics
- Integrating with monitoring tools (overview)

#### ◆ **Module 3: Distributed Testing**

- Master-Slave architecture
- Scaling test execution

#### ◆ **Module 4: Best Practices & CI/CD**

- Test design best practices
- Performance testing strategy
- CI/CD integration basics

---

 **Labs**

- Implement correlation using regular expressions
- Execute distributed load testing
- Build advanced test scenario
- Mini Project: End-to-end performance testing

---

 **Capstone Project**

- Design performance test plan for real application
  - Execute load test
  - Analyze results
  - Present findings
-