



# 90 Days Playwright(JS/TS) + AI

## (JavaScript + TypeScript + AI + CI/CD)

Feature	Details
Practical Lab Exercise	10 per day (~ 400+ Labs)
Total Time	1 Hour per day (5 Hour per week)
Framework + Projects	4-5 Push to Github

## Module 1 – JavaScript Fundamentals

### Topics

- JavaScript Basics
- Data Types & Variables
- Operators & Expressions
- Control Flow
- Loops
- Functions & Arrow Functions
- Arrays & Objects
- DOM Manipulation
- Async JavaScript
- ES6+ Features
- AI - Setting up a project via the github Copilot / Augment Code.

### Key Coverage

- JavaScript fundamentals for automation
- Variables, scope, and type coercion
- Conditional logic and loops
- Functions and arrow functions
- Array and object manipulation
- DOM selectors and events
- Promises and async/await
- Modern ES6+ syntax

### Hands-On Projects

- *(Foundational coding exercises only – no standalone project)*
- 

## Module 2 – Playwright Fundamentals

### Topics

- Playwright Overview
- Installation & Setup
- Writing First Test
- Selectors & Locators
- Input Actions
- Assertions
- Browser Contexts
- Screenshots & Videos
- Playwright Inspector

### Key Coverage

- Playwright architecture and setup
- Writing stable UI tests
- Locator strategies and actions
- UI assertions and validations
- Browser context isolation
- Debugging and test introspection

### Hands-On Projects

- **Form Validation Suite (Beginner)**
    - Input field validation
    - Required vs optional fields
    - Error message verification
    - Accessibility basics
  - **Debugging with Playwright Inspector (Beginner)**
    - Debug failing tests
    - Use Inspector and Trace Viewer
    - Failure analysis
-

## Module 3 – TypeScript for Automation

### Topics

- TypeScript Basics
- Type Annotations
- Interfaces
- Types vs Interfaces
- Functions in TypeScript
- Generics
- Enums & Literal Types
- Classes & Access Modifiers
- Modules
- TypeScript with Playwright

### Key Coverage

- Type-safe automation development
- Interface-driven design
- Generic utilities and helpers
- Strong typing in Playwright tests

### Hands-On Projects

- (*TypeScript concepts applied gradually to existing Playwright tests*)
- 

## Module 4 – Advanced Playwright Capabilities

### Topics

- Network Interception
- API Mocking
- File Uploads & Downloads
- Dialogs & Alerts
- Frames & iFrames
- Multiple Pages & Popups
- Authentication & Storage State
- Mobile Testing
- Visual Testing

### Key Coverage

- Route handlers and mocks
- File handling automation
- Multi-tab and frame handling
- Authentication reuse

- Responsive and visual testing

## Hands-On Projects

- **Browser Context & Auth State Mini Project (Beginner)**
  - Multiple browser contexts
  - Incognito session testing
  - Save and reuse authentication state
- **Network Mocking Project (Intermediate)**
  - Mock API responses
  - Simulate failures (500, timeout)
  - Feature flag testing
- **File Upload & Download Validation (Intermediate)**
  - Upload automation
  - File type and size validation
  - Download verification
- **Multi-Page & iFrame Testing (Intermediate)**
  - Popups and new tabs
  - Cross-frame interactions
- **Mobile Responsive Testing (Intermediate)**
  - Device emulation
  - Viewport-based validations
  - Touch interactions
- **Visual Regression Suite (Intermediate)**
  - Baseline screenshots
  - Screenshot comparison
  - Ignore dynamic regions

---

## Module 5 – Playwright Test Runner Deep Dive

### Topics

- Test Configuration
- Test Organization
- Fixtures
- Parallel Execution
- Retries & Timeouts
- Reporters
- Test Annotations
- Parameterized Tests
- Global Setup & Teardown

### Key Coverage

- Playwright Test architecture
- Fixture-based design

- Parallel and sharded execution
- Flaky test handling
- Advanced reporting

## Hands-On Projects

- **Data-Driven Test Suite (Intermediate)**
  - JSON/CSV-based tests
  - Parameterized execution
  - Environment-based data
- **Flaky Test Investigation & Stabilization (Intermediate)**
  - Introduce flaky tests
  - Retry vs fix analysis
  - Trace-based debugging
- **Multi-Browser Compatibility Suite (Intermediate)**
  - Chromium, Firefox, WebKit execution
  - Conditional skips and tagging

---

# Module 6 – API Testing with Playwright

## Topics

- API Testing Fundamentals
- APIRequestContext
- HTTP Methods
- API Authentication
- Response Validation
- API + UI Hybrid Testing
- API Performance

## Key Coverage

- REST API testing with Playwright
- Authenticated API calls
- JSON assertions and validation
- Hybrid automation flows
- Performance thresholds

## Hands-On Projects

- **API Testing Mini Framework (Intermediate)**
  - GET and POST tests
  - Header and auth handling
  - JSON assertions
- **API + UI Hybrid Testing Suite (Advanced)**
  - API-based data setup

- UI verification
  - API teardown
  - **API Performance Validation (Intermediate)**
    - Response time measurement
    - Threshold-based failures
    - Performance reporting
- 

## Module 7 – OOPS & Page Object Model (POM)

### Topics

- OOP Principles
- Classes & Inheritance
- Abstract Classes
- Interfaces in OOP
- Design Patterns
- Page Object Model
- Page Components
- Base Page Pattern

### Key Coverage

- Object-oriented automation design
- Scalable POM architecture
- Reusable components
- Maintainable test code

### Hands-On Projects

- **E-Commerce Checkout Flow (POM-Based) (Advanced)**
    - Product selection
    - Checkout validation
    - Negative scenarios
  - **Page Components Library (Intermediate)**
    - Header, footer, navigation components
    - Component-level assertions
- 

## Module 8 – Automation Framework & CI/CD

### Topics

- Framework Architecture
- Data Management
- Utilities

- Logging
- Error Handling
- Reporting
- Git & GitHub
- GitHub Actions
- Docker
- Parallel CI Execution

## Key Coverage

- Production-ready framework design
- Logging and observability
- CI/CD with Docker
- Parallel and sharded pipelines

## Hands-On Projects

- **Complete E2E Automation Framework (Expert)**
  - Folder architecture
  - Base test and base page
  - Environment configuration
- **Reporting & Failure Analysis (Intermediate)**
  - Allure reports
  - Screenshots and videos
  - Failure diagnostics
- **CI/CD Pipeline Integration (Advanced)**
  - GitHub Actions
  - Docker execution
  - Parallel and sharded runs
- **Performance Metrics Collection (UI) (Advanced)**
  - Web vitals (LCP, CLS, FID)
  - Page load metrics
  - Threshold-based failures

---

## Module 9 – Career & Industry Readiness

### Topics

- Resume Building
- GitHub Portfolio
- Interview Preparation

### Key Coverage

- Resume storytelling using projects
- GitHub repo polish

- Framework-level interview readiness

## Hands-On Projects

- **Resume & GitHub Polish (Beginner)**

- Resume bullet creation
  - Repository cleanup
  - README and documentation
-