

# QA Automation & Graphics Engineering Learning Tracker

This document tracks the topics covered so far based on your learning journey (Manual QA + Graphics Driver Validation + Python Automation using Playwright & Pytest), and outlines what should be covered next to move toward advanced QA/Automation and Graphics Engineering roles.

## Topics Covered So Far

- 1 Python basics (variables, loops, conditions, simple programs)
- 2 Python project structure and imports
- 3 Virtual environments (venv) usage and activation
- 4 Playwright setup and basic automation
- 5 Pytest framework fundamentals
- 6 Test organization (tests, regression folders)
- 7 Page Object Model (POM) structure
- 8 Fixtures and browser/page setup
- 9 Passing URL dynamically (page.open(url) approach)
- 10 Git basics (push, pull, resolving conflicts)
- 11 Debugging import errors and test failures
- 12 CI/CD concepts (basic awareness)
- 13 Graphics pipeline overview
- 14 GPU compiler basics (HLSL/GLSL → GPU instructions)
- 15 Graphics APIs vs Graphics Drivers
- 16 Serial vs Parallel execution concepts
- 17 BSOD vs TDR concepts
- 18 Graphics validation basics (performance, power, stability)
- 19 Debug tools awareness (RenderDoc / PIX / Nsight / WinDBG)

## Recommended Topics to Cover Next

- 1 Advanced Pytest (parametrize, markers, hooks)
- 2 Playwright advanced features (locators, waits, tracing, debugging)
- 3 Framework architecture best practices
- 4 Logging and reporting (Allure / HTML reports)
- 5 CI/CD integration (Jenkins/GitHub Actions)

- 6 API testing basics
- 7 Performance testing concepts
- 8 Python OOP for automation frameworks
- 9 Design patterns for automation
- 10 Advanced Git workflows
- 11 Graphics driver deep-dive (KMD vs UMD)
- 12 GPU debugging workflows
- 13 Performance profiling and bottleneck analysis
- 14 Power and thermal validation scenarios
- 15 Real-world debugging strategies used by senior QA engineers

## **How to Use This Tracker**

You can use this document as a weekly tracker: - Mark completed topics. - Add notes from real debugging sessions. - Track improvements in automation framework design. - Record new tools learned and real-world issues solved.