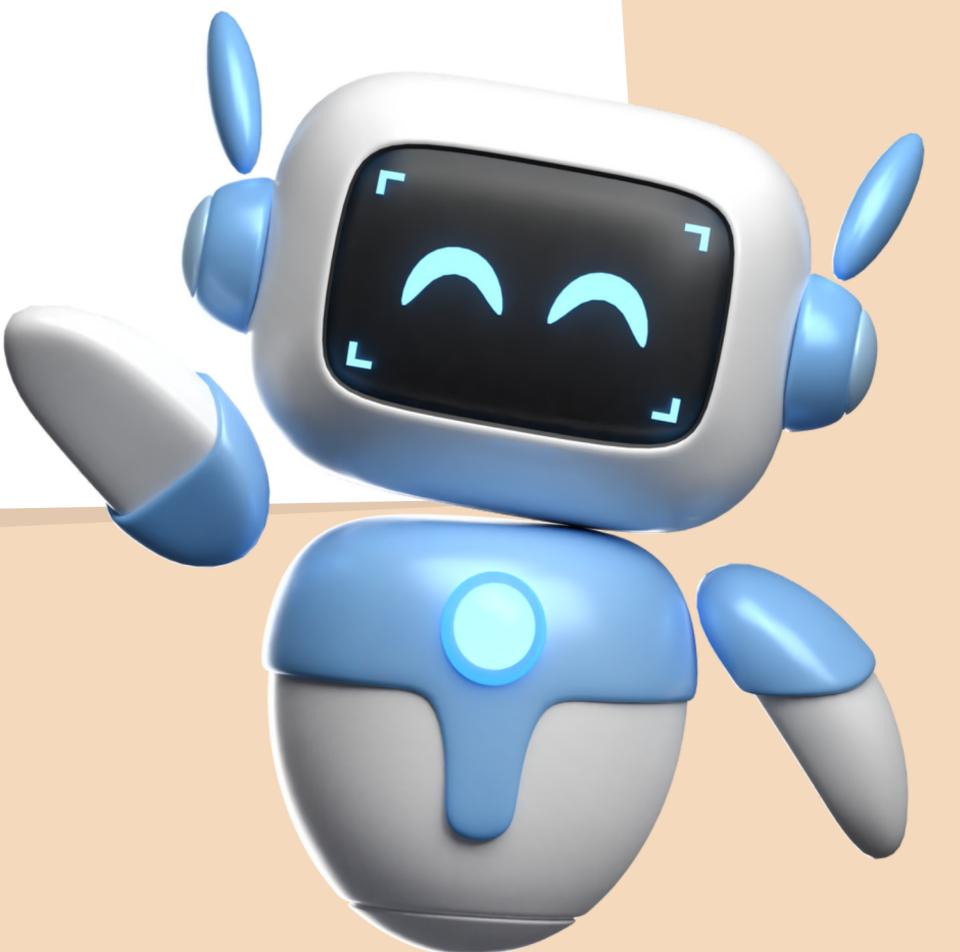


TEST AUTOMATION WITH SELENIUM

Overview of Selenium and its use in automated testing

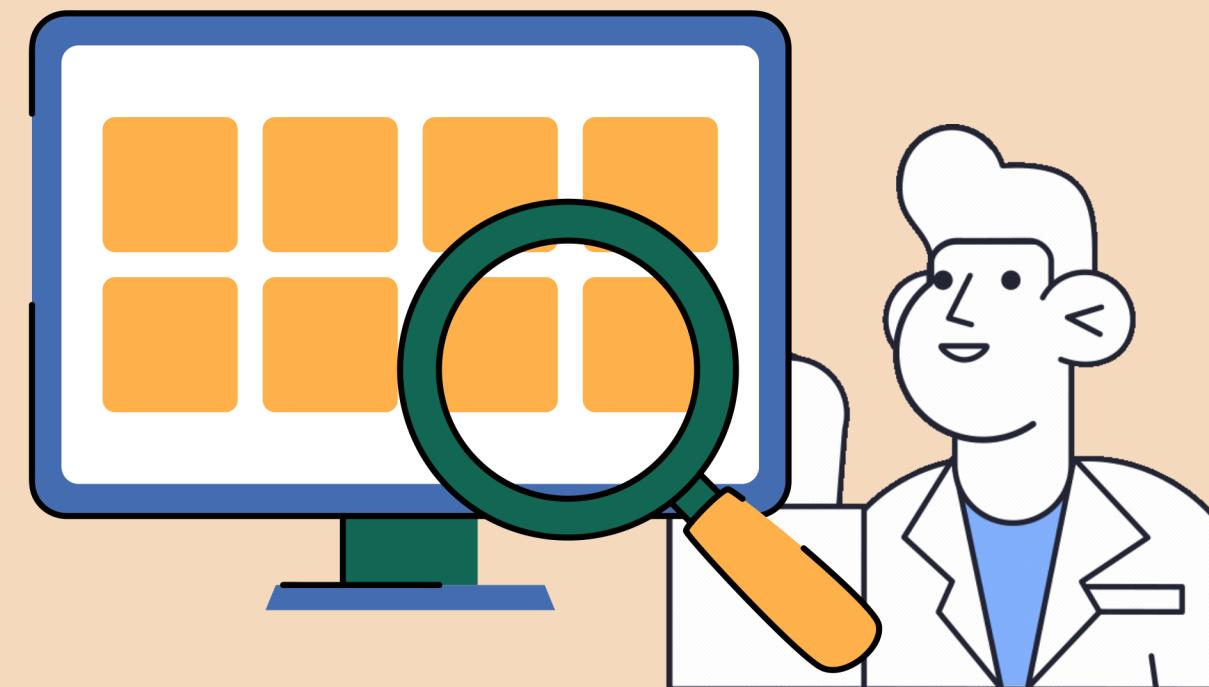
WHAT TO EXPECT



- 1** Introduction
- 2** Key Components of Selenium
- 3** Selenium WebDriver
- 4** Best Practices for Selenium Automation

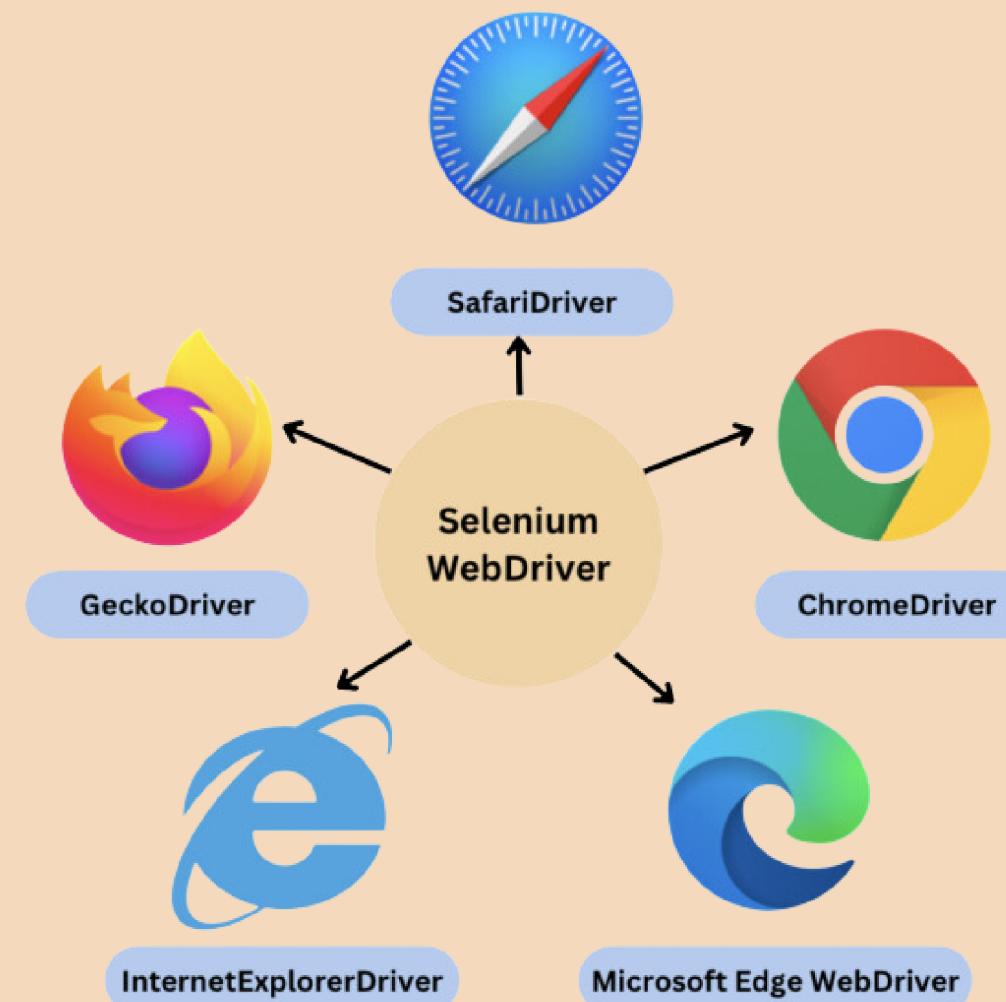
Introduction

Selenium is a powerful open-source framework for automating **web applications**. It enables testers and developers to automate browser actions, validate web functionalities, and perform regression testing.



Key Components of Selenium

- **Selenium WebDriver:** Core component for browser automation, providing a programming interface to create and execute test cases.



Key Components of Selenium

- **Selenium IDE:** A browser extension for record-and-playback testing, suitable for quick test prototyping.
- **Selenium Grid:** Allows parallel execution of tests across multiple machines and browsers for efficient testing.



Want to run tests in parallel across multiple machines? Then, Grid is for you.

Selenium WebDriver in Action

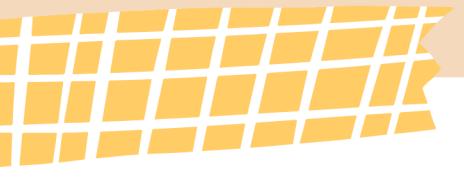
WebDriver facilitates communication between your test scripts and the browser. It supports various languages like Javascript, Java, Python, C#, etc.

Browser Support: Chrome, Firefox, Safari, Edge, and more.



Best Practices for Selenium Automation

- **Maintainable Tests:** Writing modular, reusable, and maintainable test scripts for easy updates and scalability.
- **Synchronization Strategies:** Understanding and implementing effective waiting strategies to handle dynamic web elements.
- **Cross-browser Testing:** Emphasizing the importance of testing on multiple browsers to ensure consistent performance.
- **Continuous Integration:** Integrating Selenium tests into CI/CD pipelines for automated and frequent execution.



“

See you in Next Video

