Automated Web Testing in Java with Playwright

Introducing Playwright

Andrejs Doronins

Automated Web Testing in Java with Playwright

Version Check

Version Check

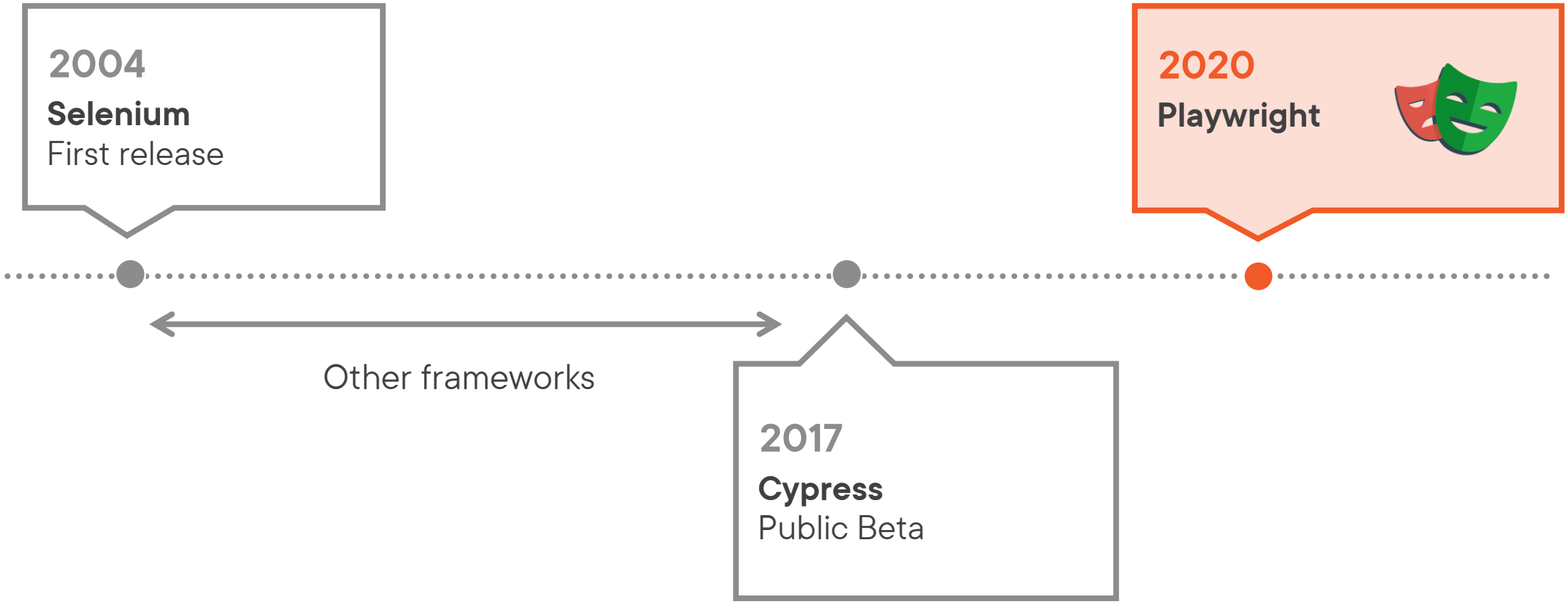
This course was created by using:

- Playwright 1.14.1

Version Check

This course is 100% applicable to:

- Playwright 1.14.1


2004
Selenium
First release
2020
Playwright
Other frameworks
2017
Cypress
Public Beta

Selenium

Pros:

- Widespread
- Multiple language bindings

Cons:

- Tests inherently flaky
- Slow release cycle

Cypress

Pros:

- Modern
- A lot of functionality

Cons:

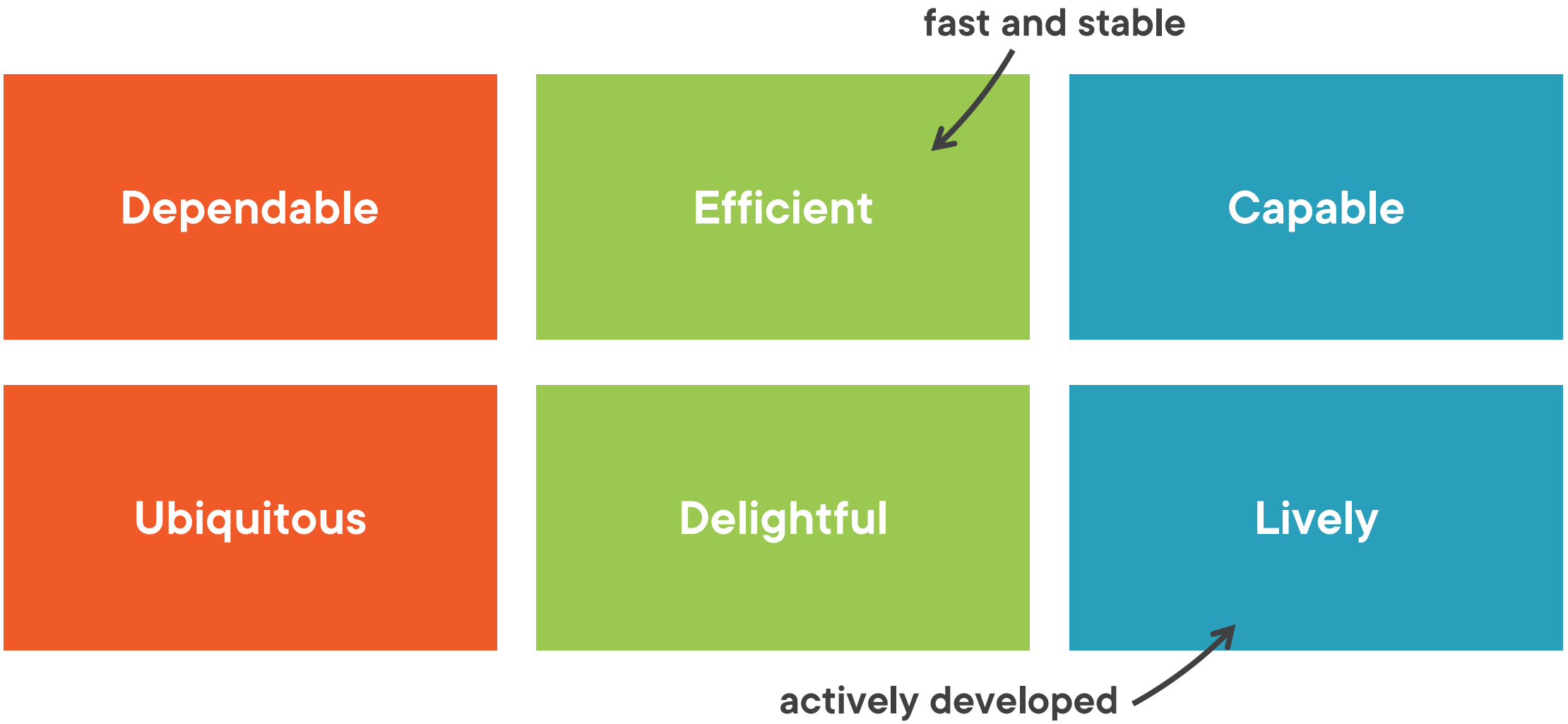
- JavaScript only
- Important trade-offs: https://docs.cypress.io/guides/references/trade-offs

Playwright

Pros:

- Backed by Microsoft
- Multiple language bindings
- In-built stabilization mechanisms
- Support for network API
- Comes with great tooling

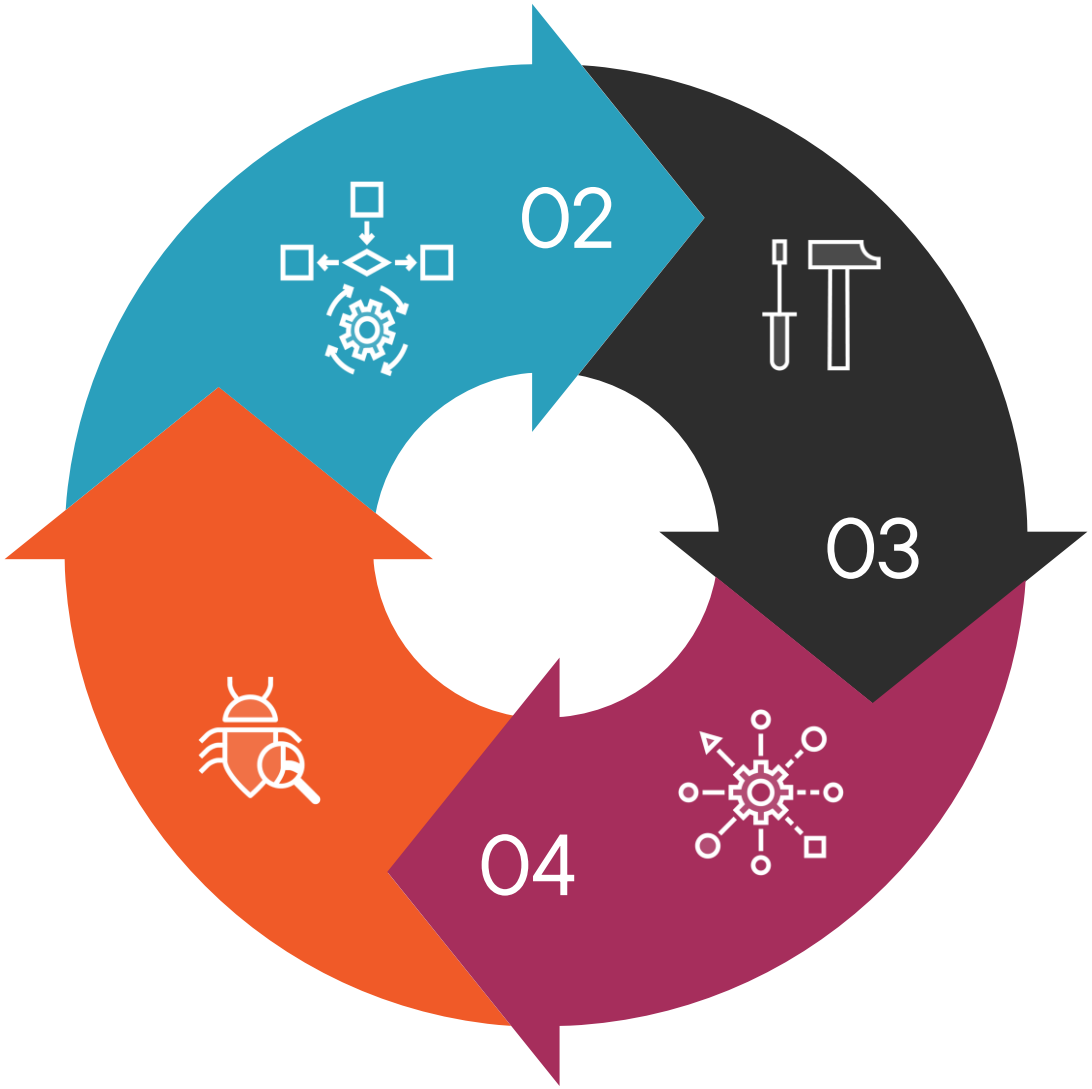
Why Playwright?



Dependable

Triage

Within 48 hours or less



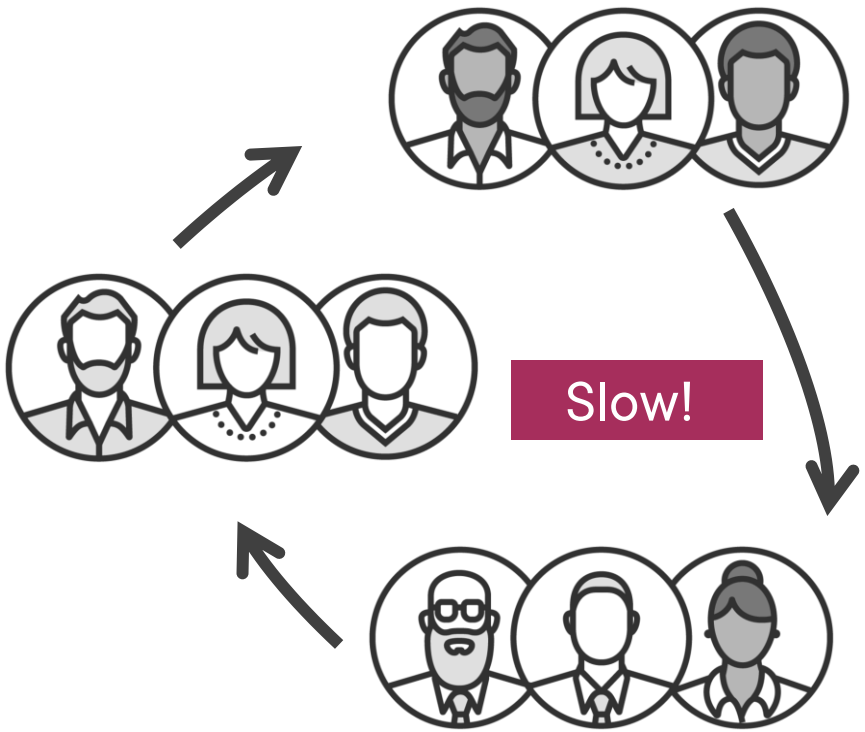
Fix or Implement

Release

Monthly!

Raise Issue or Request

Open-source: via GitHub or official Slack channel



Slow!

Java

C#

JavaScript

Python

Common Driver API

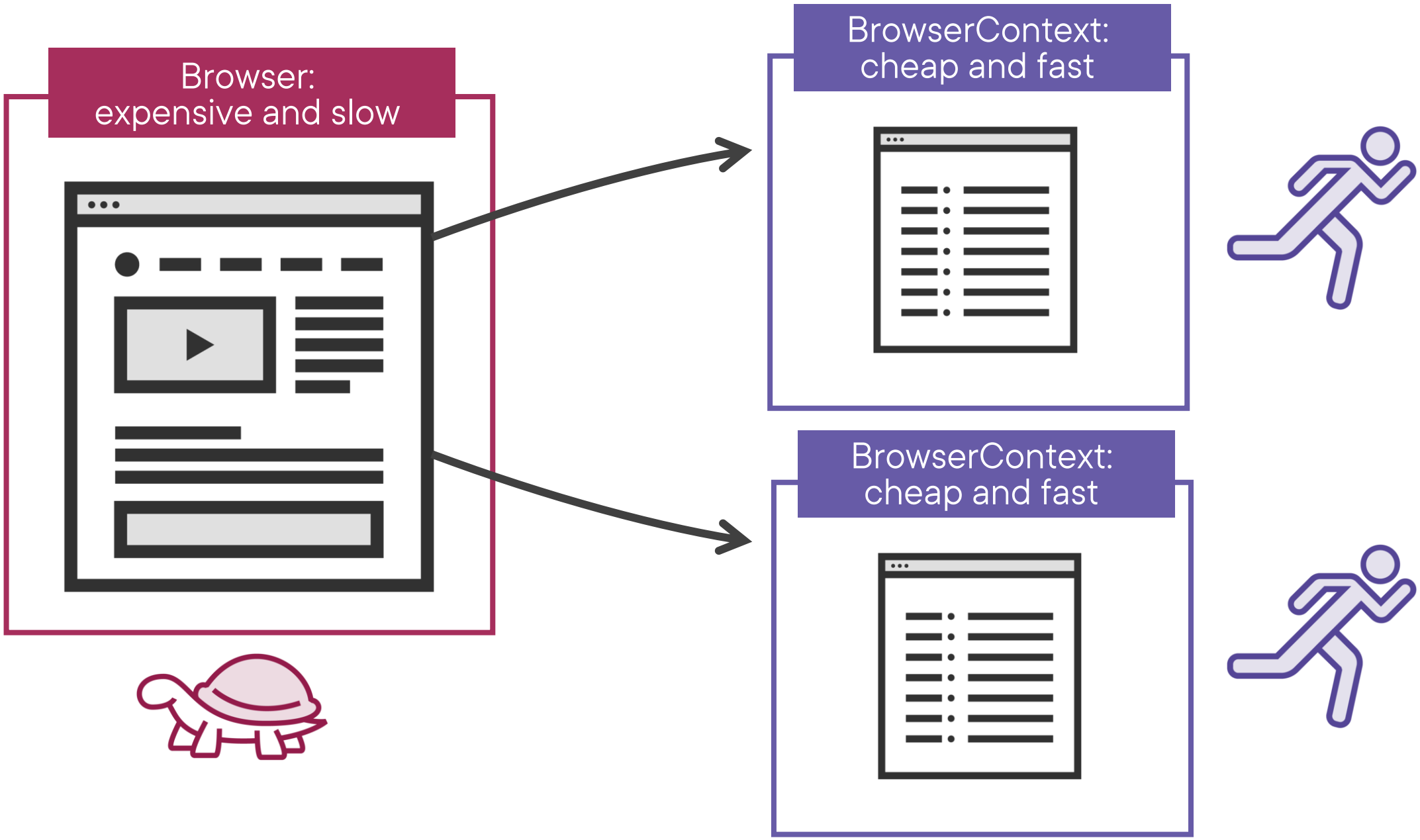
Chrom ium

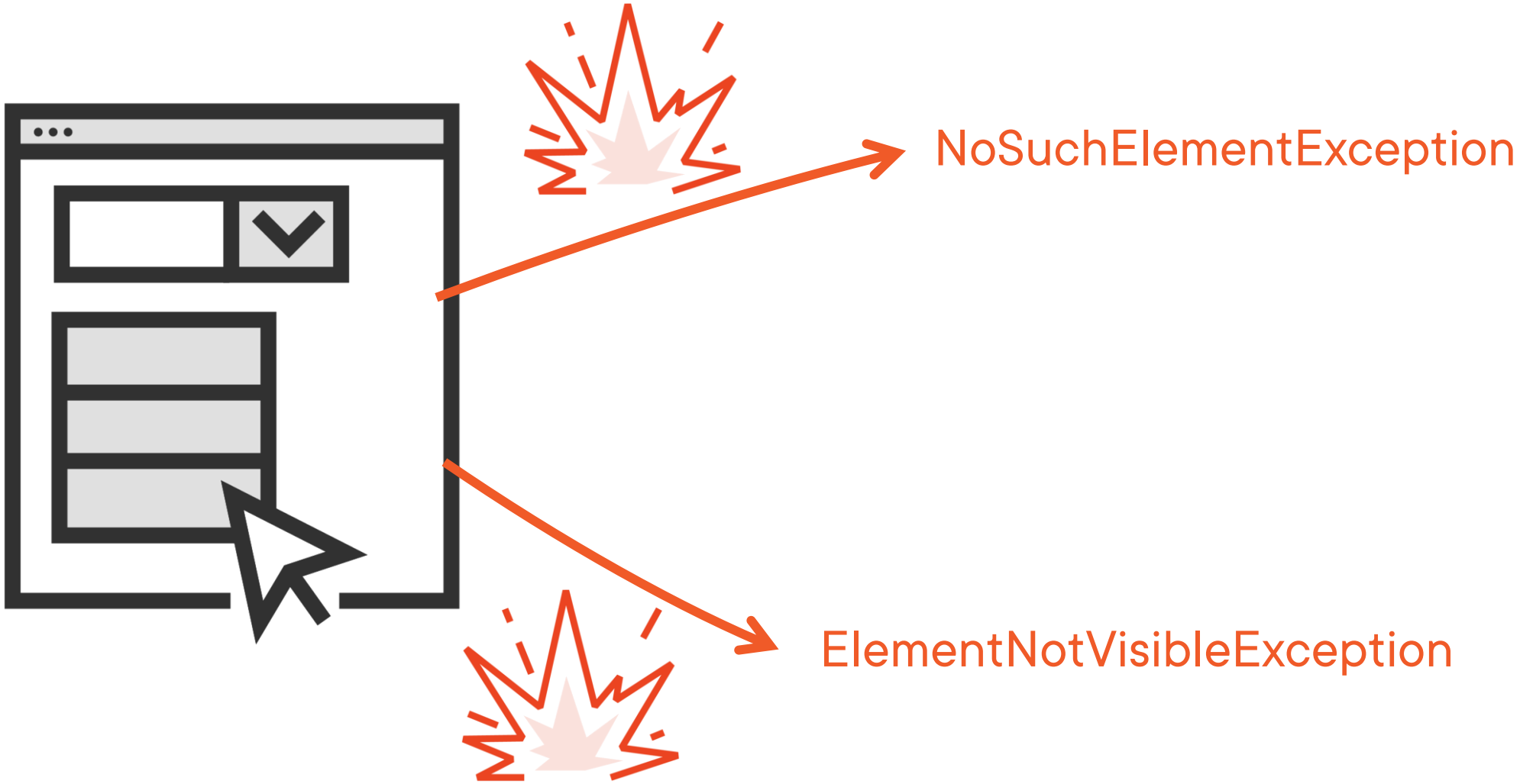
Webkit

Firefox

OS: Windows, Linux, macOS

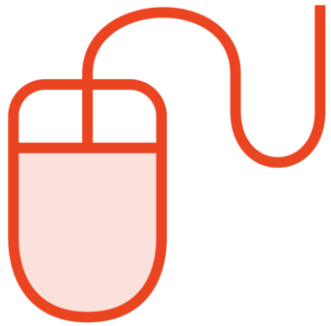
Playwright: one team!


Browser:
expensive and slow
BrowserContext:
cheap and fast
BrowserContext:
cheap and fast


NoSuchElementException
ElementNotVisibleException

```
element.click();
```

```
if(isVisible(el) && !isDisabled(el)) {
    try {
        waitSeconds(5);
        el.click();
    } catch(Exception1 | Exception2 e) {
        // handle
    }
}
```

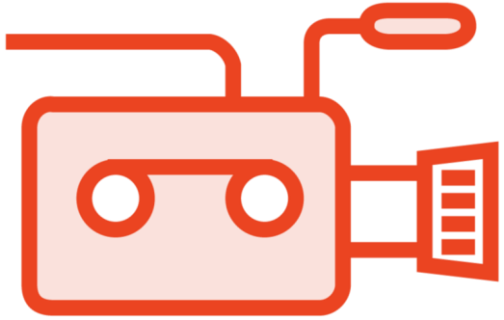


Clicking, typing

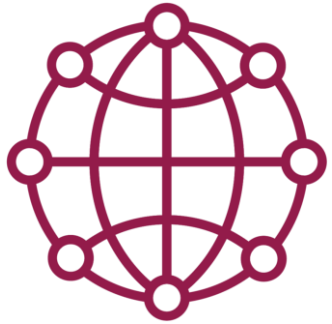
Selectors

Device emulation

Geolocation, locale

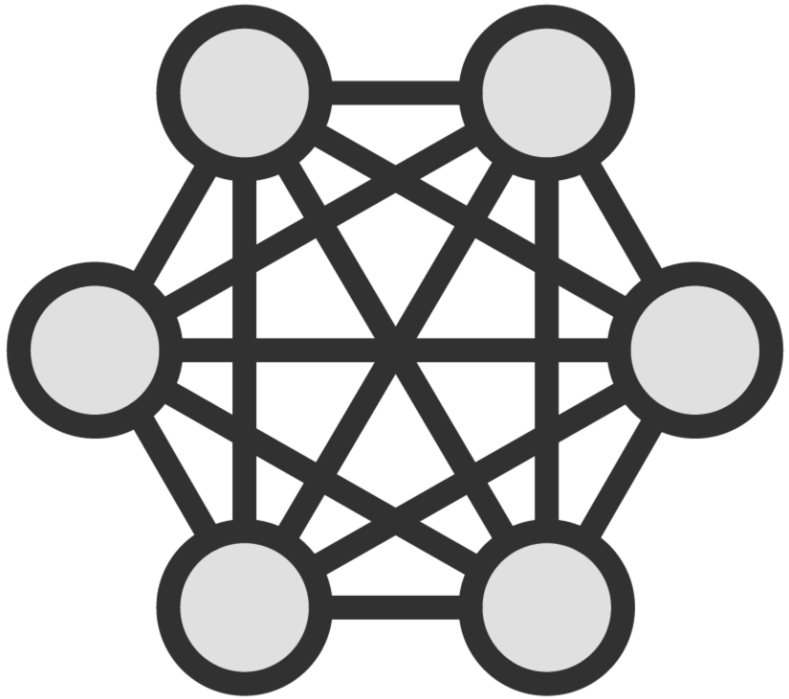


Screenshots, videos



Network Activity

Ubiquitous



All major OSs:

- Linux, Mac, Windows

All major browsers:

- Chrome, Firefox, Safari (Webkit)

Many popular languages:

- JavaScript/TypeScript, Java, C#, Python

Full CI support

Long-term serious project
backed by Microsoft

Prerequisites

Experience with

- Java (1+ years of experience)
- IDE
- Writing any kind of automated tests

Nice to have:

- Java 8 lambdas
- HTML basics

Recommended Java Courses

Java Language Fundamentals Path (Beginner section)

Setting up a Java Development Environment

Getting Started with Programming in Java

Introduction to Testing in Java

Tag

Attribute

```
<form>
    <div class="form-group">
        <label for="nameInput">Name</label>
        <input class="form-control" id="nameInput" placeholder="John">
    </div>
    <button type="submit" class="btn btn-primary">Submit</button>
</form>
```

```
page.click("button");
```

Tech Stack

Windows

Java 11 (LTS)

- Min. Java 8 or later required

IntelliJ

JUnit 5

Maven

Demo

Project setup

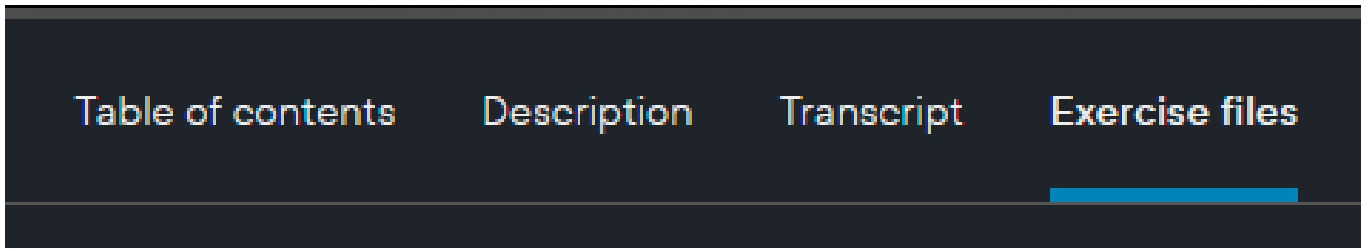


Table of contents
Description
Transcript
Exercise files

Playwright
BrowserType
Browser
page.click();

Demo Sites

For simpler scenarios:

- Local static HTML page

For complex scenarios and network demo:

- GitHub and other public sites

Content applicable to any site

Course Overview

- Selectors
- Core Playwright interfaces
- Write and refactor first test scripts
- Simple actions: clicking, typing, selecting...
- Advanced actions: handling dialogs, downloads, screenshots, etc.
- Configuring tests globally
- Network: below the surface
- Playwright Tooling