Test generator
Trace viewer
CI GitHub Actions
Test Runners
Release notes
Guides
Actions
Accessibility testing
Auto-waiting
API testing
Assertions
Authentication
Browsers
Dialogs
Debugging Tests
Downloads
Emulation
Evaluating JavaScript
Events
Extensibility
Frames

› Getting Started › Installation

Installation

Playwright was created specifically to accommodate the needs of end-to-end testing. Playwright supports all modern rendering engines including Chromium, WebKit, and Firefox. Test on Windows, Linux, and macOS, locally or on CI, headless or headed with native mobile emulation.

Playwright is distributed as a set of Maven modules. The easiest way to use it is to add one dependency to your project's `pom.xml` as described below. If you're not familiar with Maven please refer to its documentation.

Usage

Get started by installing Playwright and running the example file to see it in action.

App.java pom.xml

src/main/java/org/example/App.java

```java
package org.example;

import com.microsoft.playwright.*;

public class App {
    public static void main(String[] args) {
        try (Playwright playwright = Playwright.create()) {
            Browser browser = playwright.chromium().launch();
            Page page = browser.newPage();
            page.navigate("http://playwright.dev");
            System.out.println(page.title());
        }
    }
}
```

With the Example.java and pom.xml above, compile and execute your new program as follows:

```
mvn compile exec:java -D exec.mainClass="org.example.App"
```

Running it downloads the Playwright package and installs browser binaries for Chromium, Firefox and WebKit. To modify this behavior see installation parameters.

First script

In our first script, we will navigate to `playwright.dev` and take a screenshot in WebKit.

```java
package org.example;

import com.microsoft.playwright.*;
import java.nio.file.Paths;

public class App {
  public static void main(String[] args) {
    try (Playwright playwright = Playwright.create()) {
      Browser browser = playwright.webkit().launch();
      Page page = browser.newPage();
      page.navigate("https://playwright.dev/");
      page.screenshot(new Page.ScreenshotOptions().setPath(Paths.get("example.png")));
    }
  }
}
```

By default, Playwright runs the browsers in headless mode. To see the browser UI, pass the `headless=false` flag while launching the browser. You can also use `slowMo` to slow down execution. Learn more in the debugging tools section.

```java
playwright.firefox().launch(new BrowserType.LaunchOptions().setHeadless(false).setSlowMo(50));
```

Running the Example script

```
mvn compile exec:java -D exec.mainClass="org.example.App"
```

By default browsers launched with Playwright run headless, meaning no browser UI will open up when running the script. To change that you can pass `new BrowserType.LaunchOptions().setHeadless(false)` when launching the browser.

System requirements

- Java 8 or higher.
- Windows 10+, Windows Server 2016+ or Windows Subsystem for Linux (WSL).
- MacOS 12 Monterey or MacOS 13 Ventura.
- Debian 11, Debian 12, Ubuntu 20.04 or Ubuntu 22.04.

What's next

- Write tests using web first assertions, page fixtures and locators
- Run single test, multiple tests, headed mode
- Generate tests with Codegen
- See a trace of your tests

Next
Writing tests »