Learning Advanced Actions

Andrejs Doronins

Overview

Handling dialogs: alert, confirm, prompt

Handling downloads

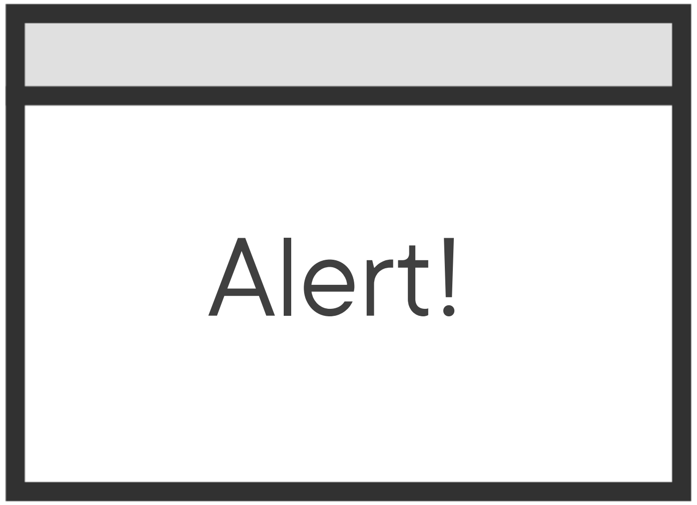
Screenshots

Leverage custom JavaScript expressions

Authentication

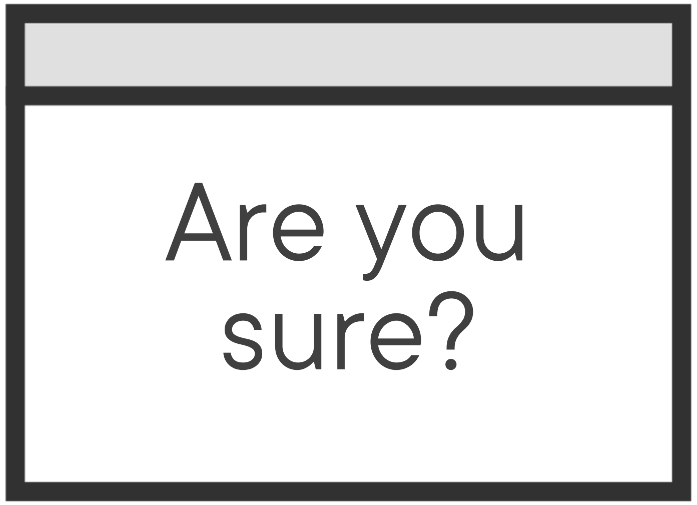
Multi-user scenarios

Dialogs



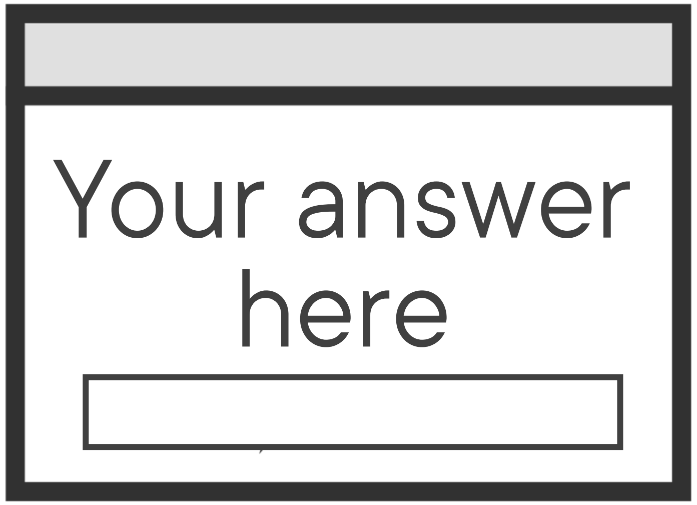
Alert

OK



Confirm

OK-Cancel



Prompt

Input + OK-Cancel

Dialogs are dismissed automatically, unless you write a handler

```
page.onDialog(Consumer<Dialog> handler)
```

```
interface Dialog {
   String type();
   String message();
   void accept();
   void dismiss();
}
```

```
                                          d -> d.dismiss();

page.onDialog(Consumer<Dialog> handler)

                                        dialog -> dialog.accept();
```

Consumer Interface

Functional Interface. Since Java 8

Takes something, does something with it, returns nothing

Examples:

- System.out.println(“string”)
- saveToDataBase(record)

As lambdas:

- x -> System.out.println(x)
- r -> saveToDataBase(r)

Learn more in:

Using Lambda Expressions in Java Code

Jose Paumard

Dialog handler?
You **must** invoke
.accept() or .dismiss()

```
// handles all dialogs
page.onDialog(Consumer<Dialog> handler)


// handles one dialog and then gets discarded
page.onceDialog(Consumer<Dialog> handler)
```

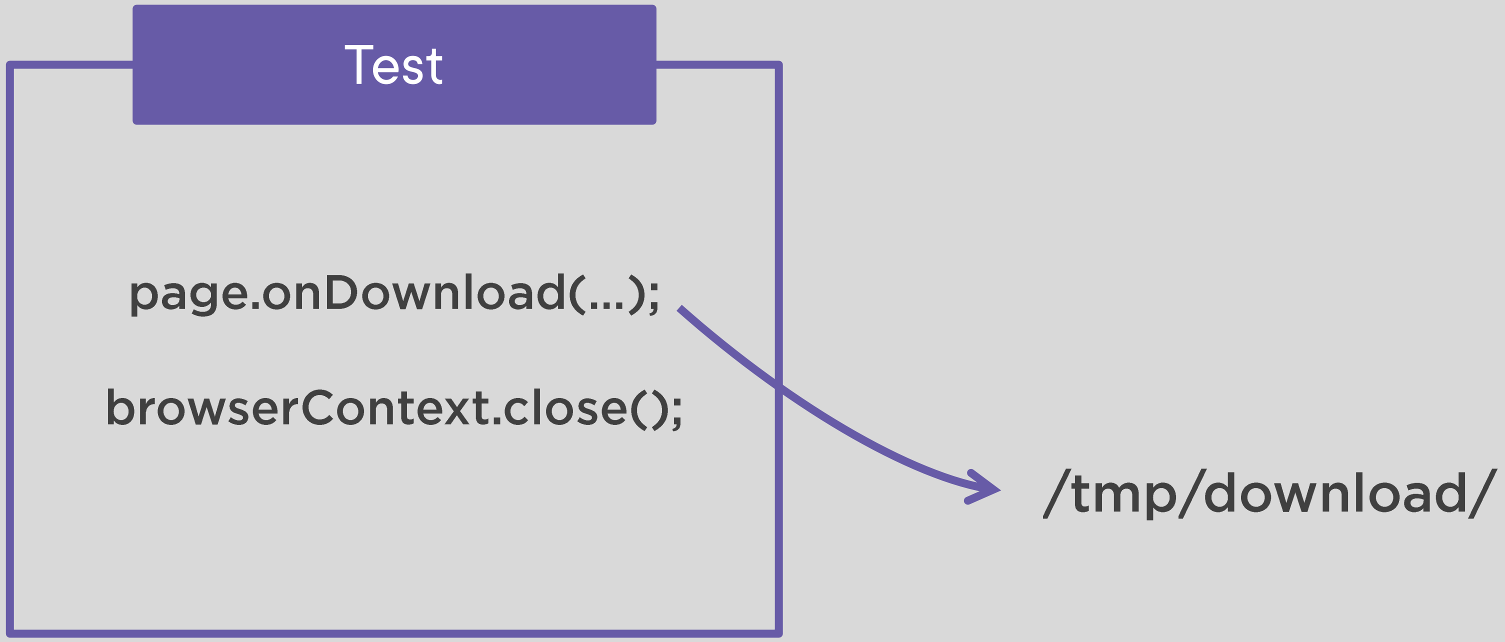
```java
interface Download {
    InputStream createReadStream();
    Path path();
    void saveAs(Path p);
}
```

```java
page.onDownload(Consumer<Download> handler)
```

```java
page.onDialog(Consumer<Dialog> handler)
```

```
// what to do with the download
page.onDownload(Consumer<Download> handler)
// trigger download now


page.waitForDownload(Runnable callback)   // trigger download here
page.onDownload(Consumer<Download> handler)



Download d = page.waitForDownload(Runnable callback)

d.saveAs(…);
```

```
br.newContext(new Browser.NewContextOptions().setAcceptDownloads(true));
// example 1
page.onDownload(d-> d.saveAs(…))
page.click("text=Download");


// example 2
Download d = page.waitForDownload(() -> {
           page.click("text=Download");
        });
d.saveAs(…);
```


Test
page.onDownload(...);
browserContext.close();
/tmp/download/

```java
page.click("text=Download");

page.onDownload(download ->
        System.out.println(download.path())
);
```

```
page.onDownload(download ->
       System.out.println(download.path())
);

page.click("text=Download");
```

```java
byte[] screenshot = page.screenshot();

page.screenshot(new Page.ScreenshotOptions()
                      .setPath(Paths.get("img.png")));
```

Screenshots

Videos

```
getLocalStorage();


countElements(e);
```

◀ () => window.localStorage.getItem(‘key’)

◀ e => e.length

```
evaluate(expression);

evalOnSelector(selector, expression);

evalOnSelectorAll(selector, expression);
```

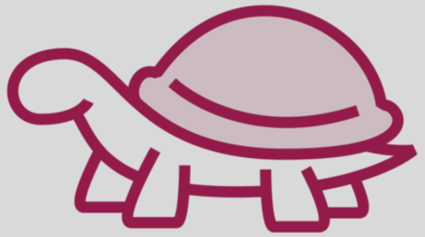
```
Object o = evaluate(expression);

Object o = evalOnSelector(selector, expression);

Object o = evalOnSelectorAll(selector, expression);
```

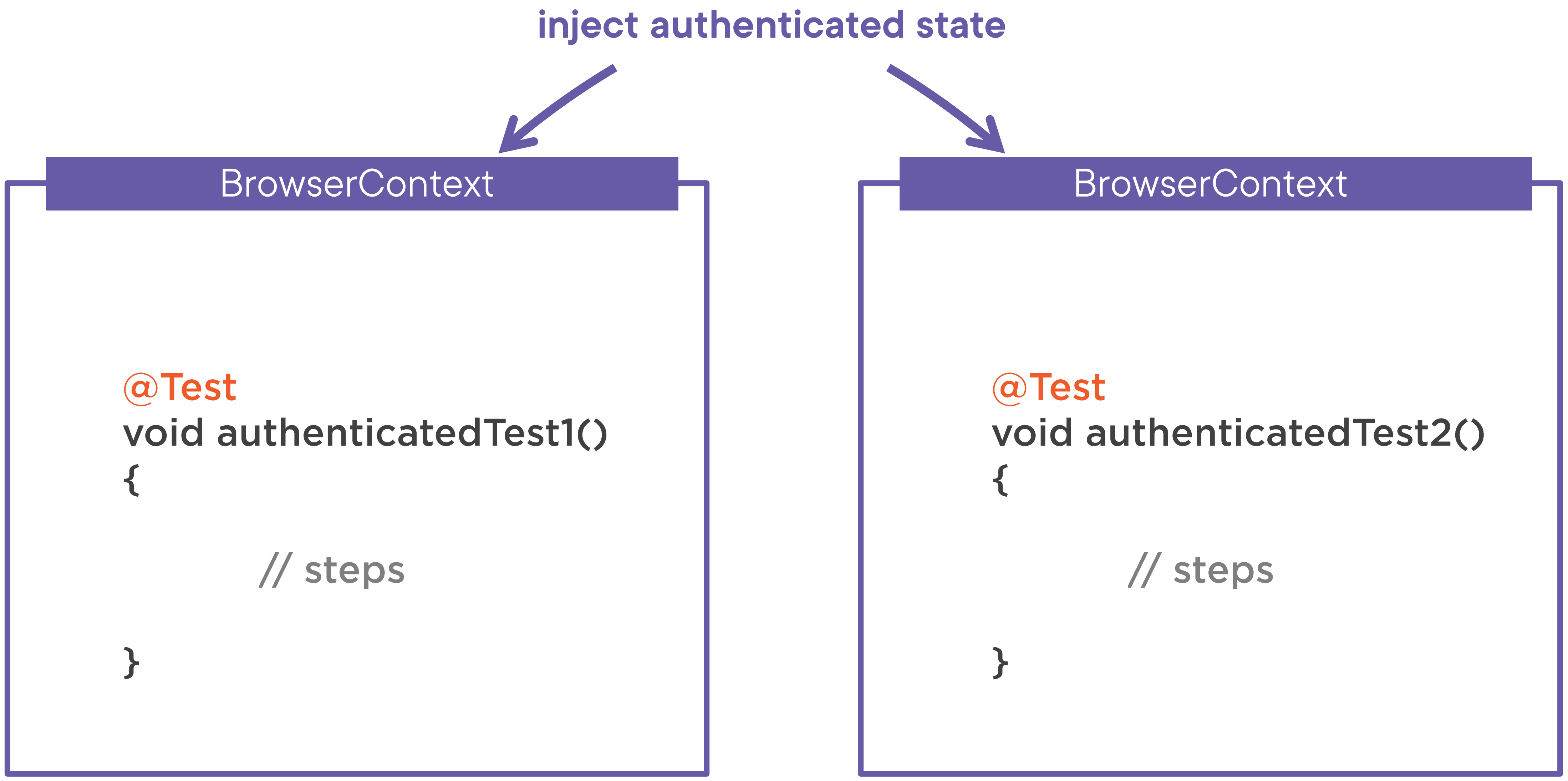
Login and password

HTTP Header for Web APIs

Other

```
@BeforeEach
void setup() {
    // login
}
@Test
void someTest() {
    // steps
}
```





```
@Test
void authenticatedTest1()
{

        // steps

}
```

```
@Test
void authenticatedTest2()
{

        // steps

}
```

```java
// save
context.storageState(new BrowserContext.StorageStateOptions()
                         .setPath(Paths.get("state.json")));

// load
context = browser.newContext(new Browser.NewContextOptions()
             .setStorageStatePath(Paths.get("state.json")));
```

Multi-page Scenario

Shared state: single user

```
Page page1 = ctx.newPage();
Page page2 = ctx.newPage();
```

Multi-user Scenario

User 1

```
Page page1 = ctxOne.newPage();
Page page2 = ctxTwo.newPage();
```

User 2

```
page1.click(x);
page2.reload();
// assert
```

Browsers may cache things

Summary

onDialog(handler)

onDownload(handler)

Screenshots and videos

Custom JS expressions

Browsercontexts

- Inject state
- Multi-user scenarios

Up Next:

Configuring Playwright Tests