

Diving into Networking



Andrejs Doronins

Overview

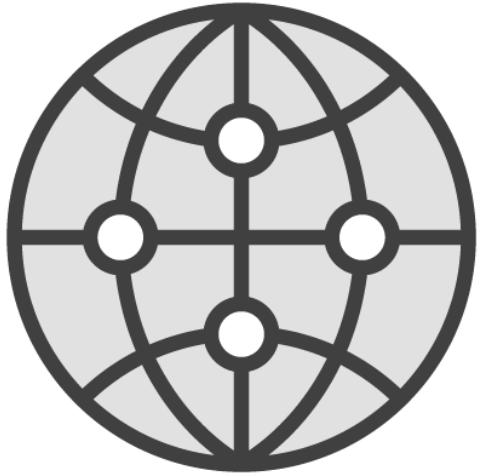


HTTP mini crash course

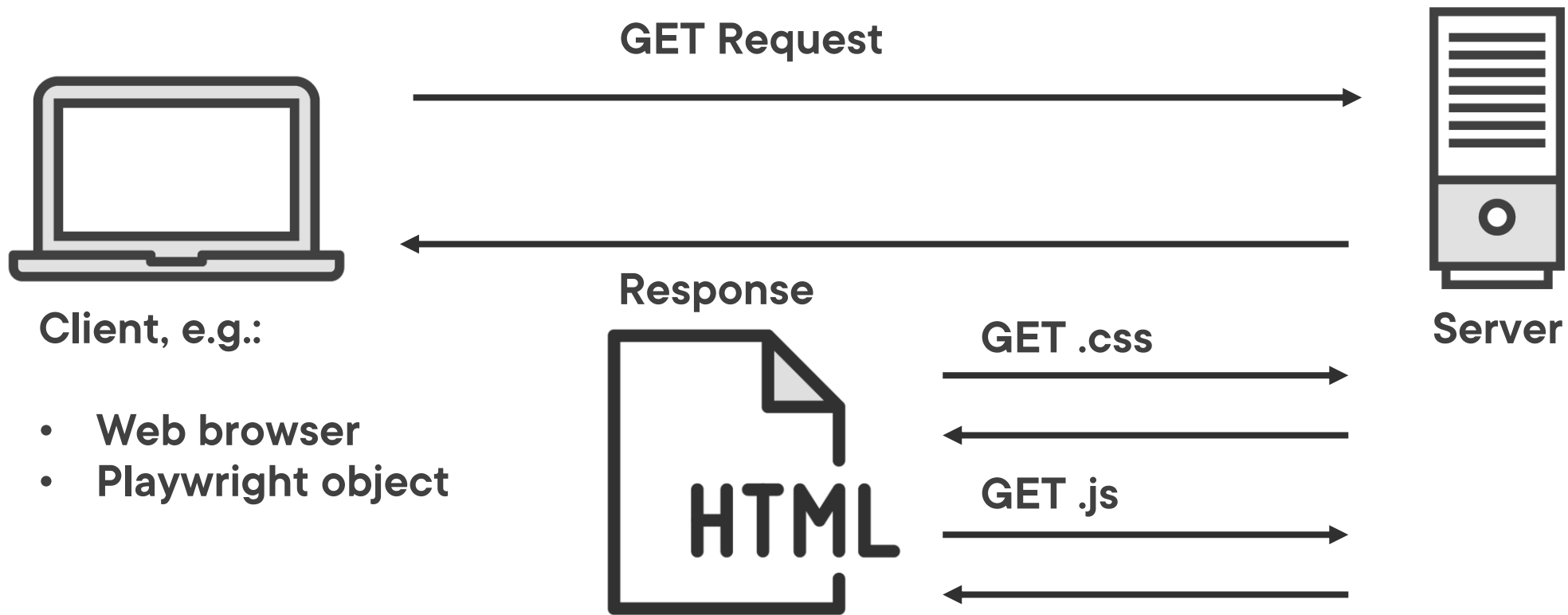
Request and Response APIs by Playwright

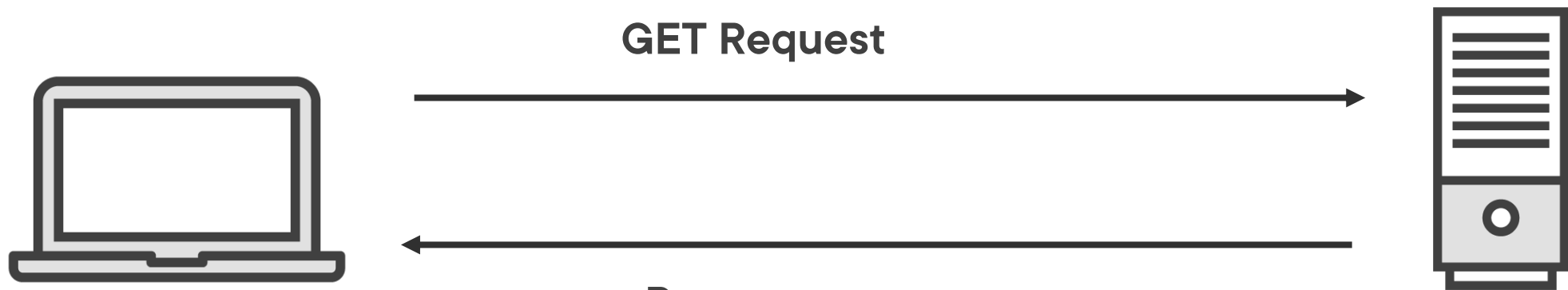
Authenticating with HTTP

Routing



**HTTP is the foundation of data
communication for the World
Wide Web**



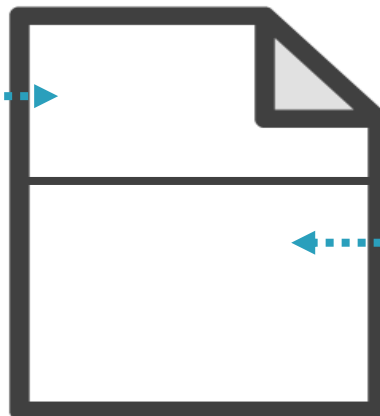


Header (metadata)

e.g. Status 200 OK

Status 401 Unauthorized

Status 404 Not Found



Body with data (payload)

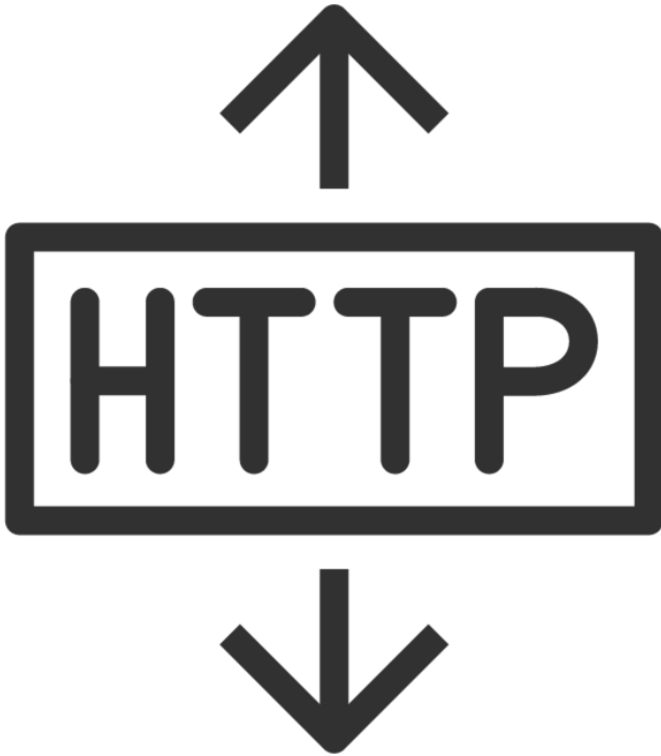
Some requests have it, some don't

Other widespread HTTP methods:

GET
POST (to create)
PUT (to update)
DELETE



Playwright HTTP APIs



Classes:

- Response
- Request

Methods:

- status()
- headers()
- body()

```
page.navigate( "http://site.com" );
```



```
Response res = page.navigate("http://site.com");
```

```
// similar in other frameworks
```

```
Response res = RestAssured.get(url);
```

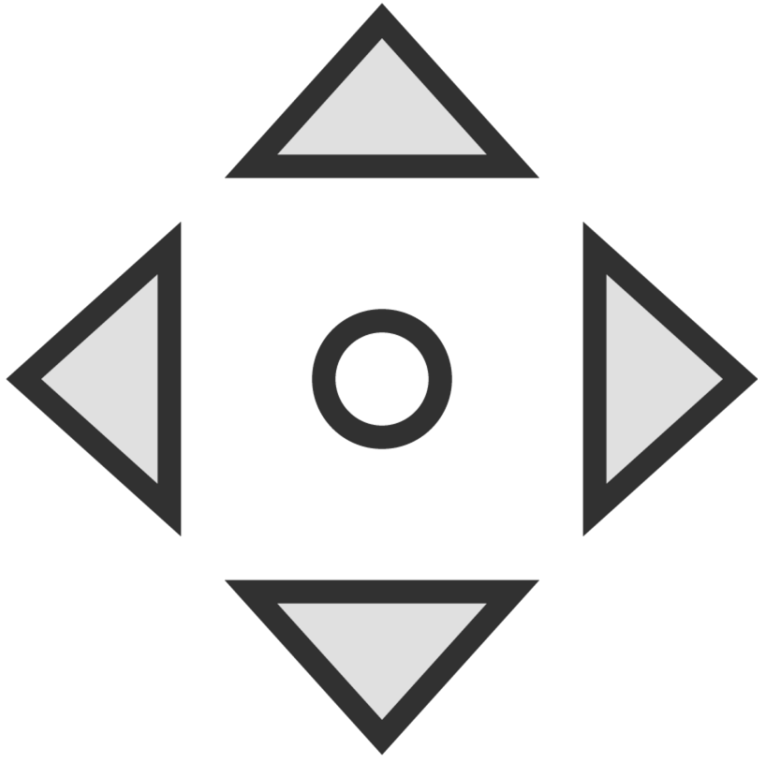
```
Response res = page.navigate("http://site.com");
```



```
interface Response {  
    String url();  
    int status();  
    boolean ok();  
    Map<String, String> headers();  
    byte[] body();  
}
```

```
Response res = page.navigate("http://site.com");  
Request req = res.request();
```

Response-returning Methods



Methods that navigate return the Response

- `.navigate()`
- `.reload()`
- `.goBack()`
- `.goForward()`
- etc.

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

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

```
page.onRequest(Consumer<Request> handler)
```

```
page.onResponse(Consumer<Response> handler)
```

Code Challenge



Use an alternative solution using the `.ok()` method

- Create a `List<Boolean>`
- Implement a response handler with `.ok()`
- Populate the List
- Assert that the List does not contain “false”

[Table of contents](#)

[Description](#)

[Transcript](#)

[Exercise files](#)


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

```
browser.newContext(new Browser.NewContextOptions()  
    .setHttpCredentials("usr", "pwd")  
    .setExtraHTTPHeaders(Map.of("Authorization", "token xyz"))  
);
```

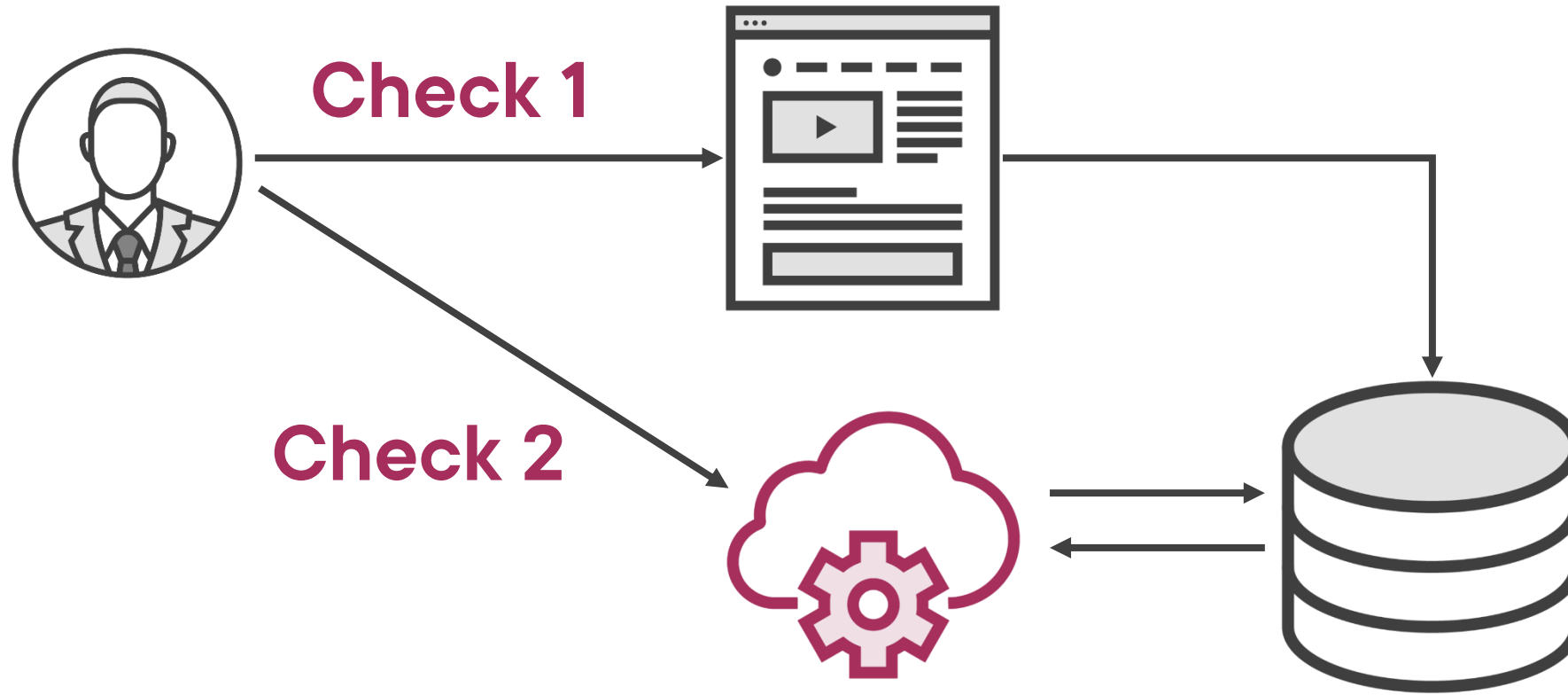


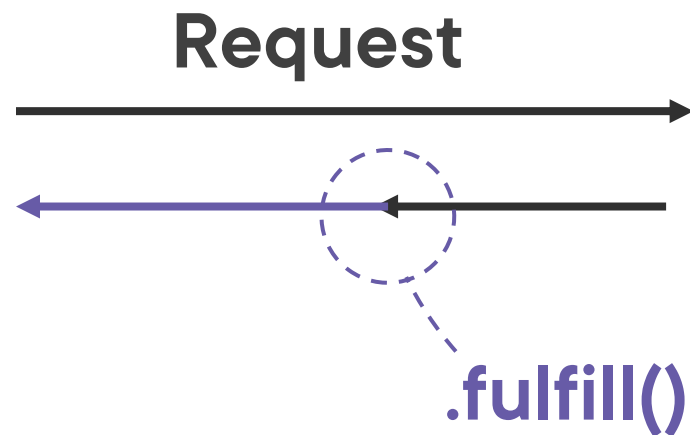
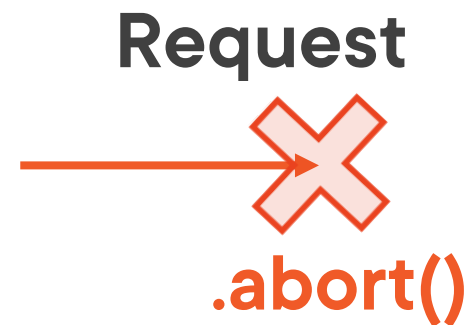
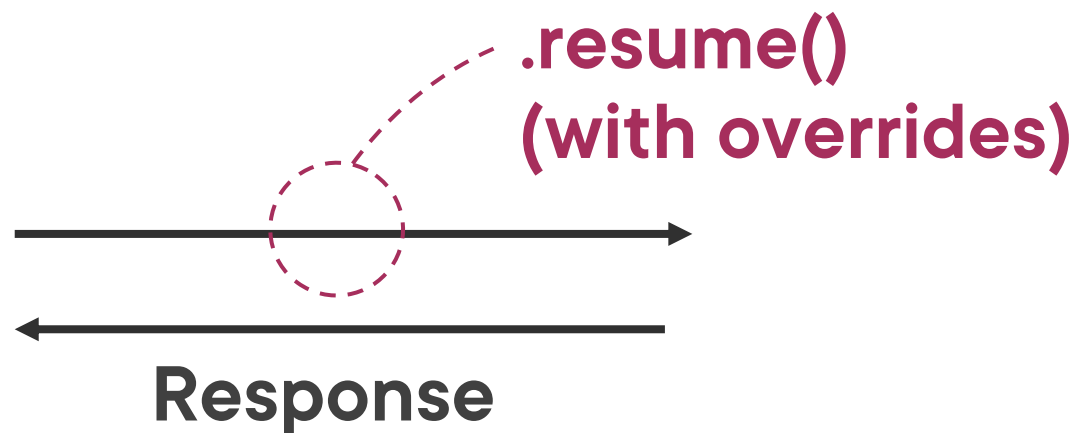
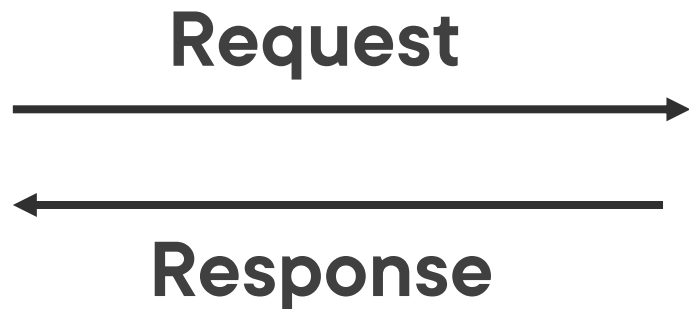
```
{  
  "login": "andrejs-ps",  
  "url": "https://api.github.com/users/andrejs-ps",  
  "type": "User",  
  "company": null,  
  "public_repos": 10,  
  "total_private_repos": 0,  
  "collaborators": 0,  
}
```

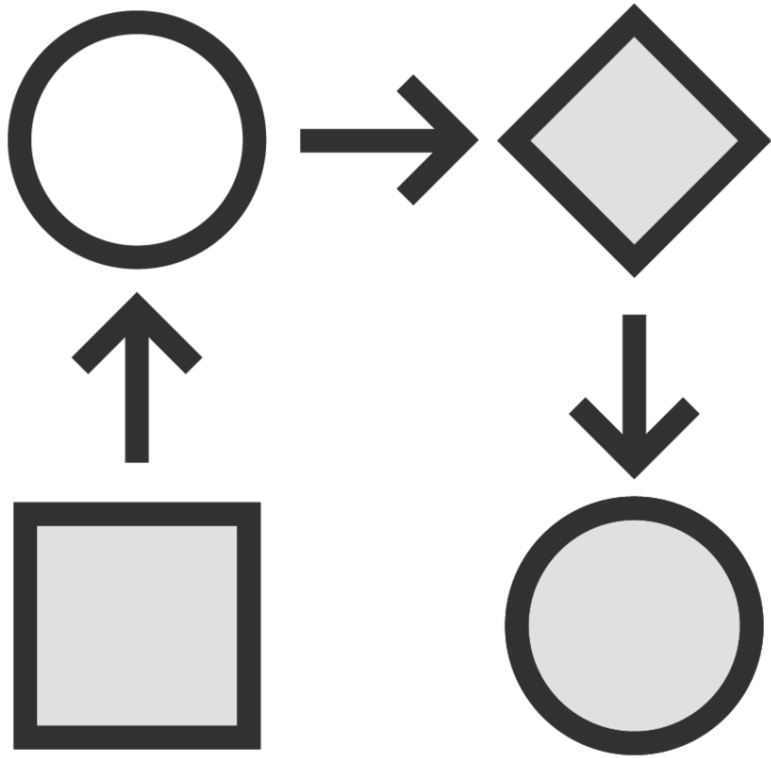
 Overview

 Repositories 10

Cross-checking Across Layers







abort

- “cancel the request, thus no response will happen”


resume

- “let the request pass”
- “override stuff and let the request pass”


fulfill

- “replace the response with own mock”

```
page.route(String pattern, Consumer<Route> handler);
```



- **/*
- **/*.{png,jpg,jpeg,svg}
- **/*.js

 `r -> r.abort();`

```
page.route("**/*", r -> r.abort());
```

```
// do things
```

```
page.unroute("**/*", [optional] handler);
```

```
// do more things
```

Summary



Support for HTTP layer

- Request & Response

onRequest() & onResponse() to monitor and analyze HTTP traffic

setExtraHTTPHeaders()

abort(), resume(), fulfill()



Up Next:
Acing Debugging Tools
