

Playwright API Testing

Questions 61 - 75

Created by
Abhishesh Mishra

61. How to perform API calls in Playwright?

Playwright provides the `request` fixture to perform API calls without UI.

```
test('GET users API', async ({ request }) => {
  const response = await request.get('/api/users');
  expect(response.ok()).toBeTruthy();
});
```

62. What is the request fixture?

`request` is a built-in API testing fixture used to send HTTP requests directly to an endpoint.

```
test('use request fixture', async ({ request }) => {
  await request.get('https://api.example.com/users');
});
```

63. How to send a POST request in Playwright?

Use `request.post()` and provide the `data` object containing the payload.

```
const response = await request.post('/api/login', {
  data: {
    username: 'admin',
    password: '123',
  },
});
```

64. How to check response status?

Use `status()` for exact code or `ok()` for 2xx range verification.

```
expect(response.status()).toBe(200);  
expect(response.ok()).toBeTruthy();
```

65. How to validate JSON response?

Convert the response using `await response.json()` and then perform assertions.

```
const body = await response.json();  
expect(body.name).toBe('John');
```

66. How to handle authentication in API tests?

Pass the authentication token inside the `headers` object.

```
await request.get('/api/profile', {  
  headers: {  
    Authorization: `Bearer ${token}`,  
  },  
});
```

67. How to save API token and reuse it?

Store the extracted token in a variable or `process.env` for global access.

```
const res = await request.post('/api/login', { data: creds });  
const token = (await res.json()).token;  
process.env.API_TOKEN = token;
```

68. How to mock an API response in Playwright?

Use `page.route()` to intercept calls and `route.fulfill()` to provide mock data.

```
await page.route('**/api/users', route =>
  route.fulfill({
    status: 200,
    body: JSON.stringify({ users: [] }),
  })
);
```

69. How to intercept network requests?

Listen to the `request` event on the page object to inspect outgoing calls.

```
page.on('request', request => {
  console.log(request.url());
});
```

70. How to block network calls (ads, analytics)?

Use `page.route()` with `route.abort()` to prevent specific requests.

```
await page.route('**/*analytics*', route => route.abort());
```

71. How to test file download API?

Use `page.waitForEvent('download')` and then `saveAs()`.

```
const download = await page.waitForEvent('download');
await download.saveAs('report.pdf');
```

72. How to test file upload API?

Use `setInputFiles` on the file input element.

```
await page.setInputFiles('#upload', 'test.pdf');
```

73. How to retry failed API requests?

Configure `retries` in the playwright config or specifically for a test block.

```
test.describe.configure({ retries: 2 });
```

74. How to log API requests in Playwright?

Listen to the `response` event to log status codes and URLs.

```
page.on('response', response => {
  console.log(response.url(), response.status());
});
```

75. Difference between UI test & API test?

Feature	UI Test	API Test
Target	Frontend / Browser	Backend / Endpoint
Execution Speed	Slower	Faster
Fixture	<code>page</code>	<code>request</code>
Dependencies	UI Elements (DOM)	Backend Schema / JSON