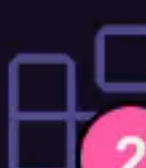
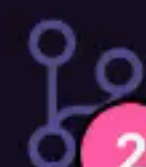


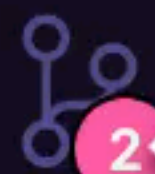
test-runner-from-scratch > JS test-runner.js > [🔗] buildDependencyTree

```
1  import { EventEmitter } from 'events';
2  import { AsyncLocalStorage } from 'async_hooks';
3  const asyncLocalStorage = new AsyncLocalStorage();
4  > const buildDependencyTree = (suiteStack, message) => { ...
19 }
20
21 class Logger {
22 >   static log(...args) { ...
30   }
31 >   static count(...args) { ...
37   }
38 }
39
40 > class TestSuite { ...
47 }
48
49 class TestRunner extends EventEmitter {
50   constructor() {
51     super();
52   }
53
```



test-runner-from-scratch > JS test-runner.js > [🔗] buildDependencyTree

```
1  import { EventEmitter } from 'events';
2  import { AsyncLocalStorage } from 'async_hooks';
3  const asyncLocalStorage = new AsyncLocalStorage();
4  > const buildDependencyTree = (suiteStack, message) => { ...
19 }
20
21 class Logger {
22 >   static log(...args) { ...
30   }
31 >   static count(...args) { ...
37   }
38 }
39
40 > class TestSuite { ...
47 }
48
49 class TestRunner extends EventEmitter {
50   constructor() {
51     super();
52   }
53
```



JS test-runner.js M

example.test.js X

test-runner-from-scratch > tests > example.test.js > ...

1 import { setTimeout } from 'node:timers/promises';

2 import { deepStrictEqual } from 'node:assert';

3 import { Logger, runner } from '../test-runner.js';

4 const { describe, it, before, beforeEach } = global; // was injected after

5 runner.on('testStart', (data) => {

6 // console.log(`\n🚀 Test \${data.name} started.`);

7 });

8

9 runner.on('testEnd', ({ elapsedTimeMs, name }) => {

10 console.log(`\n✅ Hook [\${name}] ended. Elapsed time: \${elapsedTimeMs}`);

11 });

12

13 describe('My suite 0', () => {

14 before(async () => {

15 await setTimeout(100);

16 Logger.count('[before] Hook on [My suite 0]');

17 });

18

19 it('test 0', async (ctx) => {

20 const expected = {

21 name: 'test 0',

EXPLORER

REVOJ... test-runner... tests example.test.js package.json JS test-runn... M JS webapi-asyncL... notes.txt M

OUTLINE

TIMELINE

main* 0 0 0

Ln 4, Col 90 Spaces: 4 UTF-8 LF {} JavaScript