



Fetch API & Async

SCORE

___ / 20

Chapter: JavaScript · Level ★★☆☆ · Time: 30 min

Name _____ Class _____ Date _____

AFTER THIS WORKSHEET YOU CAN

✓ Fetch data from an API ✓ Use promises and .then ✓ async/await ✓ Handle errors



Section A · Concept Check

• BEGINNER

4 × 1 = 4

- 1 fetch(url) returns a:
☐ A. string ☐ B. Promise ☐ C. array ☐ D. JSON
- 2 Parse the body with:
☐ A. res.json() ☐ B. res.parse() ☐ C. JSON.text()
☐ D. res.body()
- 3 await can be used inside:
☐ A. any function ☐ B. an async function
☐ C. loops only ☐ D. HTML
- 4 Errors are caught with:
☐ A. try/catch ☐ B. if/else ☐ C. stop()
☐ D. guard



Section B · Problem Solving

• INTERMEDIATE

2 + 3×3 = 11

- 5 fetch returns a _____ that resolves later.
- 6 const data = await res._____.();
- 7 Fetch users and log them (async/await).

- 8 Same with .then chains.

- 9 Why is fetching asynchronous?



Section C · Challenge

• CHALLENGE

5

- 10 What happens without await on res.json()?

🧠 Reflection — the most useful thing I learned: _____

A ___/4 B ___/11 C ___/5 Total ___/20

Teacher's Signature _____

Parent's Signature _____

✂ ANSWER KEY — FOLD OR CUT BEFORE HANDING OUT

1-B 2-A 3-B 4-A | 5 Promise 6 json 7 = const res=await fetch(url); const data=await res.json(); console.log(data) 8 = fetch(url).then(r⇒r.json()).then(d⇒console.log(d)) 9 = The network is slow; JS keeps running instead of freezing | 10 = You get a pending Promise, not the data

