Week 02 · Student worksheet

1. Predict the route

Write three things that might happen after Alice presses Send. Then draw Alice, DNS, two routers, the chat server and Ben. Put DNS beside the path that carries the message.

2. Read a URL

For http://localhost:8765/chat/ identify the name, port and path. Explain what localhost means. Name: _____ Port: _____ Path: _____

3. Restore a message

Chunks arrive in order 3, 1, 2. Reassemble them: _____. Now imagine chunk 2 never arrives. What should the receiver know or do? _____

Packet cards — cut along the boxes

Message M1 | Chunk 1 | “Meet ”
Message M1 | Chunk 2 | “at ”
Message M1 | Chunk 3 | “4”
DNS card: chat.example → 203.0.113.10
Receipt cards: accepted / delivered / read reported

4. Predict, observe, explain

Pause Ben, then Alice sends a message. Predict Alice’s status: _____. Observe: _____. Resume Ben. What changes? _____. Mark read. What extra event does this report? _____.

5. Find the failure

For each case, write one thing that may still work: (a) Name lookup fails. (b) Chat server does not respond. (c) Ben stops polling. Explain why these are different failures.

6. Exit ticket

Does DNS forward the message? Why?
Does accepted mean read? Why?
What helps reconstruct out-of-order data?
One question for next week: _____

Optional home challenge — 10–15 minutes

Map a message's journey, label a request and response, and show one failure. Or open chat/index.html in a text editor, find HOME CHALLENGE, and remove the two leading slashes from the line immediately below it: state.textContent += ... . Reload, send a message and check the time appears. The server already includes created_at. Explain why this is acceptance time, not reading time.