

A Menu on Every Page

60 minutes

Build a Website

WHAT THIS LESSON DOES

Find out your site has no way back, then add a menu to the top of every page so a reader can move anywhere from anywhere. The menu lists every page by itself.

WHAT STUDENTS SHOULD BE ABLE TO DO

- Understand that a menu lists every page and belongs on every page.
- Find the menu block in Links and buttons.
- Recognise that the menu block builds itself and has nothing to fill in.
- Test a site by moving around it as a visitor.

BEFORE THE LESSON

- Open the web page builder interactive on the board and check it loads through the school filter.
- One device per pupil, logged in, with each pupil's own two-page site from earlier saved and ready to open.
- Have a live example ready to model the trap: click the button, land on the second page, and show there is no way home.

A Menu on Every Page

HOW THE LESSON RUNS

5 min	introduction	Start: What We're Building
4 min	content	A Menu Is the Same on Every Page
5 min	activity	Find the Problem
7 min	activity	Put a Menu on Your First Page
5 min	activity	Put a Menu on the Second Page
5 min	activity	Go Both Ways
6 min	activity	Give the Second Page More to Say
7 min	activity	Your Turn: Make It Yours
4 min	content	Make Sense
4 min	content	Reflect
8 min	summary	What You Covered

TEACHING MOVES

1. Start: What We're Building. Walk into the trap yourself on the board. Click the button, land on the second page, and ask the class how you get home. Let the silence make the point.
2. A Menu Is the Same on Every Page. Say plainly before they hunt: the menu block has nothing to fill in. It lists every page by itself. It lives in Links and buttons, beside the button.
3. Find the Problem. Tell them the page names above the preview are a building tool, not part of what a visitor sees. Otherwise they will think the problem is already solved.
4. Put a Menu on Your First Page. Model dragging the menu to the very top of the Home page. Point out there is no box to complete: dragging it in is the whole action, and it lists both pages on its own.
5. Put a Menu on the Second Page. Same move on the Hobbies page. Stress sameness: a menu in a different place on each page is worse than none at all.
6. Go Both Ways. Let them move back and forth a few times using the menu. Going anywhere from anywhere on something they built is the payoff of the module.
7. Give the Second Page More to Say. Have them add a subheading and a sentence about a hobby so the second page is worth the trip once a reader arrives.
8. Your Turn: Make It Yours. Set them working independently. Look for the menu at the top of both pages, not halfway down one, and a second page with more than a bare heading.
9. Make Sense. Ask one pupil what the menu does if they add a third page. The answer, that it appears on its own, is the whole idea of the lesson.
10. Reflect. Spend a real minute on where a menu sits. Pupils know it is nearly always at the top without ever having been told, which is worth naming aloud.
11. What You Covered. Read the bullets aloud. If you are behind, skip this and spend the minutes on the independent work in step 8 instead.

WHAT TO WATCH FOR

- Pupils look for a box to type page names into, expecting to build the menu themselves. Say before they start that the menu block has nothing to fill in. Show that it already lists both pages the moment it is dropped on the page.
- Pupils think one menu on one page is enough, treating it like a heading that covers the whole site. Have them land on the page with no menu and try to leave. When they get stuck, make the rule concrete: a menu only works if it is on every page.
- Pupils believe the page names above the preview mean the site already works, since they can move around using them. Explain those names are a building tool that a real visitor never sees. Ask them to move around using only what is on the page itself.

DIFFERENTIATION

- Emerging: Model the first menu drag with them following along one action at a time, then let them repeat it on the second page. Pair them with a partner who has the menu working so they can copy the top-of-page placement.
- Developing: Ask them to explain back to you why the menu has to go on both pages before they move on. Have them run the both-ways test aloud, naming each click, so the routine sticks.
- Proficient: Once both menus work, set them to add a third page and predict what the menu does before they check. Ask them to read their whole site as a visitor and fix anything confusing, then justify one change they made.