Web Creator Lab

Ages 13-15 | Tech Builders

Build a responsive, multi-page website.

How to use this guide

Read each section, complete its four lab steps, then answer the questions in the online course. The online progress bar advances only after four correct answers.

Section 1: HTML

LEARNING GOAL: HTML. Your finished work should demonstrate: Two connected pages.

EXPLAIN THE CONCEPT: Semantic HTML helps people and browsers understand a page. Navigation connects related pages. Think about where you might use this skill in a real or imaginary project. Describe the problem it helps you solve and how you would recognize a successful result.

ORIGINAL PROJECT LAB: 1. Create index.html and about.html. 2. Add headings and main content to both pages. 3. Link the pages through a navigation area. 4. Open each link to confirm the destination. Finished result: Two connected pages.

EXPLAIN THE MECHANISM: Semantic HTML gives content a role. Headings organize topics, navigation groups useful destinations, and buttons perform actions. A visual appearance does not supply meaning by itself. Use a real link for navigation and a button for an action on the current page.

WORKED CASE: A fictional project page has one main heading, a paragraph explaining the project, and a navigation link labeled Read the project notes. A second-level heading introduces a testing section. Styling a paragraph with large text would not give it the same heading role.

TEST AND JUSTIFY: Read the page in source order and test whether the headings describe the content beneath them. Ask a helper to find the notes without coaching. Keep the practice local and omit real school details, private photographs, and contact information.

GO DEEPER: Identify an assumption, test a safe alternative, and explain the trade-off. Keep a short before-and-after record that someone else could follow.

SAFETY CHECK: Recognize phishing and social engineering, secure accounts, understand privacy settings, and perform cyber labs only with authorization. Use pretend data, never real passwords or private information. Stop and ask a trusted adult if

anything seems unsafe.

REVIEW QUESTIONS: What is the main idea? Why does the first step matter? How would you detect a mistake? What evidence proves the result? What would you improve?

Answer in your own words before taking the quiz.

PORTFOLIO REFLECTION: Save a drawing, sanitized screenshot, or short note describing the goal, your approach, your evidence, and one lesson learned. Expected result: Two connected pages.

Investigate with pictures and choices

A form field has only placeholder text.

What improves its accessible structure?

1. An associated visible label
2. A larger background image
3. Removing keyboard focus

Worked reasoning

Answer: An associated visible label. A label identifies the field even after a value is entered.

Try a changed example

Inspect the field with keyboard navigation.

Picture and meaning matches

HTML: Defines content structure

CSS: Controls presentation

JavaScript: Handles programmed behavior

Ways to respond

Choose, point, draw, demonstrate, type, speak or use your usual communication tool.

A helper can operate the controls while you choose.

Counting support if useful

Use zero to ten counters or dots. Touch each once. The last count names the quantity. Rearrange the same set and check whether anything was added or removed.

Choose this foundation practice only when it fits the learner goal.

My notes: _____

Hands-on lab

1. Create index.html and about.html.
2. Add headings and main content to both pages.
3. Link the pages through a navigation area.
4. Open each link to confirm the destination.

Finished result: Two connected pages.

Check your understanding

1. Which idea best describes HTML?

A. Responsive CSS adapts layout to available space.

B. Semantic HTML helps people and browsers understand a page.

C. JavaScript can respond to a visitor's action.

D. It has no purpose in this project.

2. What is the first step in this lab?

A. Add headings and main content to both pages.

B. Link the pages through a navigation area.

C. Create index.html and about.html.

D. Open each link to confirm the destination.

3. What should you do after step two?

A. Add headings and main content to both pages.

B. Open each link to confirm the destination.

C. Create index.html and about.html.

D. Link the pages through a navigation area.

4. Which result shows that your lab is finished?

A. Two connected pages.

B. A copied answer without checking it

C. A skipped lab

D. A blank page with no work

Answer review

1. B. Semantic HTML helps people and browsers understand a page.
2. C. Start with: Create index.html and about.html.
3. D. Step three is: Link the pages through a navigation area.
4. A. The goal is: Two connected pages.

Section 2: CSS

LEARNING GOAL: CSS. Your finished work should demonstrate: A site that fits phone and desktop widths.

EXPLAIN THE CONCEPT: Responsive CSS adapts layout to available space. A simple flexible layout is easier to use on phones. Think about where you might use this skill in a real or imaginary project. Describe the problem it helps you solve and how you would recognize a successful result.
ORIGINAL PROJECT LAB: 1. Create styles.css and link it from both pages. 2. Style readable text and clear navigation. 3. Use a flexible row that wraps on small screens. 4. Test at a narrow and a wide browser width. Finished result: A site that fits phone and desktop widths.
EXPLAIN THE MECHANISM: Responsive layout adapts to the available space. Flexible tracks and wrapping can keep content readable, but they still need testing. A layout that fits one wide screen can overflow at a narrow width or when text is enlarged.
WORKED CASE: Imagine two project panels, each fixed at 400 pixels, inside a 600-pixel container with a gap. Their combined widths exceed the container. Allow flexible widths or stack them at a suitable breakpoint, then check that titles and links remain visible.
TEST AND JUSTIFY: Test a wide view, a narrow view, and enlarged text. Record any sideways scrolling or overlap. Change the layout rule causing the problem instead of shrinking all text until a screenshot fits. Keep meaningful reading order when panels move.
GO DEEPER: Identify an assumption, test a safe alternative, and explain the trade-off. Keep a short before-and-after record that someone else could follow.
SAFETY CHECK: Recognize phishing and social engineering, secure accounts, understand privacy settings, and perform cyber labs only with authorization. Use pretend data, never real passwords or private information. Stop and ask a trusted adult if anything seems unsafe.
REVIEW QUESTIONS: What is the main idea? Why does the first step matter? How would you detect a mistake? What evidence proves the result? What would you improve? Answer in your own words before taking the quiz.
PORTFOLIO REFLECTION: Save a drawing, sanitized screenshot, or short note describing the goal, your approach, your evidence, and one lesson learned. Expected result: A site that fits phone and desktop widths.

Investigate with pictures and choices

A narrow screen cuts off a button.

What should you revise?

1. Only the browser title
2. The stored password
3. The responsive layout and sizing

Worked reasoning

Answer: The responsive layout and sizing. Presentation must adapt without hiding needed controls.

Try a changed example

Test long labels and increased text size.

Picture and meaning matches

HTML: Defines content structure

CSS: Controls presentation

JavaScript: Handles programmed behavior

Ways to respond

Choose, point, draw, demonstrate, type, speak or use your usual communication tool.

A helper can operate the controls while you choose.

Counting support if useful

Use zero to ten counters or dots. Touch each once. The last count names the quantity. Rearrange the same set and check whether anything was added or removed.

Choose this foundation practice only when it fits the learner goal.

My notes: _____

Hands-on lab

1. Create styles.css and link it from both pages.
2. Style readable text and clear navigation.
3. Use a flexible row that wraps on small screens.
4. Test at a narrow and a wide browser width.

Finished result: A site that fits phone and desktop widths.

Check your understanding

1. Which idea best describes CSS?

A. It has no purpose in this project.

B. Semantic HTML helps people and browsers understand a page.

C. Responsive CSS adapts layout to available space.

D. JavaScript can respond to a visitor's action.

2. What is the first step in this lab?

A. Test at a narrow and a wide browser width.

B. Style readable text and clear navigation.

C. Use a flexible row that wraps on small screens.

D. Create styles.css and link it from both pages.

3. What should you do after step two?

A. Use a flexible row that wraps on small screens.

B. Style readable text and clear navigation.

C. Test at a narrow and a wide browser width.

D. Create styles.css and link it from both pages.

4. Which result shows that your lab is finished?

A. A blank page with no work

B. A site that fits phone and desktop widths.

C. A copied answer without checking it

D. A skipped lab

Answer review

1. C. Responsive CSS adapts layout to available space.

2. D. Start with: Create styles.css and link it from both pages.

3. A. Step three is: Use a flexible row that wraps on small screens.

4. B. The goal is: A site that fits phone and desktop widths.

Section 3: JavaScript

LEARNING GOAL: JavaScript. Your finished work should demonstrate: A working interactive tip button.

EXPLAIN THE CONCEPT: JavaScript can respond to a visitor's action. The result should be clear and keyboard accessible. Think about where you might use this skill in a real or imaginary project. Describe the problem it helps you solve and how you would recognize a successful result.

ORIGINAL PROJECT LAB: 1. Add a button labeled Show tip. 2. Write a click handler that changes a nearby message. 3. Test the button with a mouse and keyboard. 4. Check that the page still explains itself before clicking. Finished result: A working interactive tip button.

EXPLAIN THE MECHANISM: JavaScript can respond to an event and update a visible result. Decide what the visitor should do and what confirms success. Use native

controls so keyboard users can trigger the action too. Display untrusted text as text, rather than interpreting it as HTML.

WORKED CASE: A button labeled Show project tip reveals a short paragraph. A native button can be activated with the keyboard as well as a pointer. Update the paragraph using textContent for a plain-text tip. Do not inject a visitor's input through innerHTML.

TEST AND JUSTIFY: Test the initial hidden state, one activation, and repeated activation. Check that the button label still describes its action and focus stays usable. Record what changes on the page; a console message alone may not tell a visitor that anything happened.

GO DEEPER: Identify an assumption, test a safe alternative, and explain the trade-off. Keep a short before-and-after record that someone else could follow.

SAFETY CHECK: Recognize phishing and social engineering, secure accounts, understand privacy settings, and perform cyber labs only with authorization. Use pretend data, never real passwords or private information. Stop and ask a trusted adult if anything seems unsafe.

REVIEW QUESTIONS: What is the main idea? Why does the first step matter? How would you detect a mistake? What evidence proves the result? What would you improve? Answer in your own words before taking the quiz.

PORTFOLIO REFLECTION: Save a drawing, sanitized screenshot, or short note describing the goal, your approach, your evidence, and one lesson learned. Expected result: A working interactive tip button.

Investigate with pictures and choices

A click handler increments count from zero.

What should one click do?

1. Refresh until correct
2. Increase the count by one
3. Change every page field

Worked reasoning

Answer: Increase the count by one. An event handler should perform the defined state transition.

Try a changed example

Test rapid repeated clicks and reset behavior.

Picture and meaning matches

HTML: Defines content structure

CSS: Controls presentation

JavaScript: Handles programmed behavior

Ways to respond

Choose, point, draw, demonstrate, type, speak or use your usual communication tool.

A helper can operate the controls while you choose.

Counting support if useful

Use zero to ten counters or dots. Touch each once. The last count names the quantity. Rearrange the same set and check whether anything was added or removed.

Choose this foundation practice only when it fits the learner goal.

My notes: _____

Hands-on lab

1. Add a button labeled Show tip.
2. Write a click handler that changes a nearby message.
3. Test the button with a mouse and keyboard.
4. Check that the page still explains itself before clicking.

Finished result: A working interactive tip button.

Check your understanding

1. Which idea best describes JavaScript?

A. Responsive CSS adapts layout to available space.

B. It has no purpose in this project.

C. Semantic HTML helps people and browsers understand a page.

D. JavaScript can respond to a visitor’s action.

2. What is the first step in this lab?

A. Add a button labeled Show tip.

B. Check that the page still explains itself before clicking.

C. Write a click handler that changes a nearby message.

D. Test the button with a mouse and keyboard.

3. What should you do after step two?

A. Add a button labeled Show tip.

B. Test the button with a mouse and keyboard.

C. Write a click handler that changes a nearby message.

D. Check that the page still explains itself before clicking.

4. Which result shows that your lab is finished?

A. A skipped lab

B. A blank page with no work

C. A working interactive tip button.

D. A copied answer without checking it

Answer review

1. D. JavaScript can respond to a visitor’s action.

2. A. Start with: Add a button labeled Show tip.

3. B. Step three is: Test the button with a mouse and keyboard.

4. C. The goal is: A working interactive tip button.

Section 4: Guided Project & Worked Examples

COURSE INVESTIGATION: The interface must show failure honestly

YOUR PRACTICE INPUT: A fictional contact form has name and message fields. The network request fails, but the page always displays ’Sent’.

WHY IT WORKS: A UI state must reflect the observed outcome. Client validation and a loading indicator do not prove that a server accepted a submission.

PLAN BEFORE YOU ACT: Draw or write the starting state, the question you need to answer, and the evidence you expect to see. A paper trace is acceptable; label it as a prediction unless you actually run a test.

WORK THE CASE: Draw idle, submitting, success, error, and uncertain states. Test keyboard focus and duplicate-click prevention in a safe mock.

COMPARE WITH THE EXPECTED RESULT: Success appears only after a confirmed accepted result. A timeout is uncertain, and a retry must not blindly duplicate an effect.

EXPLAIN YOUR REASONING: Show the connection between the supplied input, each decision, and your result. If a number, label, or observation differs, return to the first differing step rather than changing several things together.

KEEP YOUR EVIDENCE: Save the initial plan, a drawing or sanitized output, and one explanation of what the result does and does not prove. Use pointing, speech, drawing, typing, or your usual communication method.

LEARNING GOAL: Guided Project & Worked Examples. Your finished work should demonstrate: A completed guided project with documented evidence: Source files or repository, Responsive screenshots, Accessibility/test checklist, Revision notes.

EXPLAIN THE CONCEPT: Review the course concepts through two applied examples. Inspect semantic HTML and explain how navigation/main/footer improve structure and accessibility. Compare fixed-width and responsive layouts at phone and desktop sizes. Guided project: Build a responsive multi-page site with semantic HTML, reusable CSS, accessible navigation, form-safe interaction, and JavaScript behavior. Work with synthetic or fictional information and ask an adult for help when appropriate. Think about where you might use this skill in a real or imaginary project. Describe the problem it helps you solve and how you would recognize a successful result.

ORIGINAL PROJECT LAB: 1. Read both worked examples and explain the reasoning. 2. Build a responsive multi-page site with semantic HTML, reusable CSS, accessible navigation, form-safe interaction, and JavaScript behavior. 3. Record the steps you completed and the result you observed. 4. Compare your result to the goal and correct one issue. Finished result: A completed guided project with documented evidence: Source files or repository, Responsive screenshots, Accessibility/test checklist, Revision notes.

EXPLAIN THE MECHANISM: Build a fictional project showcase with semantic structure, responsive styling, and one small interaction. Define a visitor task first: find the project notes and reveal a testing tip. Keep the first version small enough to inspect.

WORKED CASE: Use a heading, explanatory paragraphs, a descriptive notes link, and a Show testing tip button. Style the content in flexible panels. Reveal a text-only tip on activation and check that it does not cover the notes link.

TEST AND JUSTIFY: Test the visitor task with a pointer and keyboard in wide and narrow views. Record expected and actual behavior. Repair one problem and repeat the same test. Screenshots can support the record but do not replace an explanation of what was tested.

GO DEEPER: Identify an assumption, test a safe alternative, and explain the trade-off. Keep a short before-and-after record that someone else could follow.

SAFETY CHECK: Recognize phishing and social engineering, secure accounts, understand privacy settings, and perform cyber labs only with authorization. Use pretend data, never real passwords or private information. Stop and ask a trusted adult if anything seems unsafe.

REVIEW QUESTIONS: What is the main idea? Why does the first step matter? How would you detect a mistake? What evidence proves the result? What would you improve? Answer in your own words before taking the quiz.
PORTFOLIO REFLECTION: Save a drawing, sanitized screenshot, or short note describing the goal, your approach, your evidence, and one lesson learned. Expected result: A completed guided project with documented evidence: Source files or repository, Responsive screenshots, Accessibility/test checklist, Revision notes.

Investigate with pictures and choices

Two controls have the same unclear label.

What helps users choose correctly?

1. Distinct action names
2. Removing one at random
3. Only changing their order

Worked reasoning

Answer: Distinct action names. Labels should describe each intended action clearly.

Try a changed example

Ask a tester to predict each action without clicking.

Picture and meaning matches

HTML: Defines content structure

CSS: Controls presentation

JavaScript: Handles programmed behavior

Ways to respond

Choose, point, draw, demonstrate, type, speak or use your usual communication tool.

A helper can operate the controls while you choose.

Counting support if useful

Use zero to ten counters or dots. Touch each once. The last count names the quantity. Rearrange the same set and check whether anything was added or removed.

Choose this foundation practice only when it fits the learner goal.

My notes: _____

Hands-on lab

1. Read both worked examples and explain the reasoning.
2. Build a responsive multi-page site with semantic HTML, reusable CSS, accessible navigation, form-safe interaction, and JavaScript behavior.
3. Record the steps you completed and the result you observed.
4. Compare your result to the goal and correct one issue.

Finished result: A completed guided project with documented evidence: Source files or repository, Responsive screenshots, Accessibility/test checklist, Revision notes.

Check your understanding

1. Which idea best describes Guided Project & Worked Examples?

A. Semantic HTML helps people and browsers understand a page.

B. Review the course concepts through two applied examples.

C. Responsive CSS adapts layout to available space.

D. It has no purpose in this project.

2. What is the first step in this lab?

A. Build a responsive multi-page site with semantic HTML, reusable CSS, accessible navigation, form-safe interaction, and JavaScript behavior.

B. Record the steps you completed and the result you observed.

C. Read both worked examples and explain the reasoning.

D. Compare your result to the goal and correct one issue.

3. What should you do after step two?

A. Build a responsive multi-page site with semantic HTML, reusable CSS, accessible navigation, form-safe interaction, and JavaScript behavior.

B. Compare your result to the goal and correct one issue.

C. Read both worked examples and explain the reasoning.

D. Record the steps you completed and the result you observed.

4. Which result shows that your lab is finished?

A. A completed guided project with documented evidence: Source files or repository, Responsive screenshots, Accessibility/test checklist, Revision notes.

B. A copied answer without checking it

C. A skipped lab

D. A blank page with no work

Answer review

1. B. Review the course concepts through two applied examples.
2. C. Start with: Read both worked examples and explain the reasoning.
3. D. Step three is: Record the steps you completed and the result you observed.
4. A. The goal is: A completed guided project with documented evidence: Source files

or repository, Responsive screenshots, Accessibility/test checklist, Revision notes.

Section 5: Independent Challenge & Mastery

INDEPENDENT TRANSFER CASE: The interface must show failure honestly
KEEP THE ORIGINAL BASELINE: A fictional contact form has name and message fields. The network request fails, but the page always displays 'Sent'.
CHANGE ONE CONDITION: Simulate an invalid message, a server rejection, and a timeout. Each has a different recovery path.
YOUR TASK: Draw idle, submitting, success, error, and uncertain states. Test keyboard focus and duplicate-click prevention in a safe mock.
PREDICT, THEN CHECK: State which part of your original reasoning should remain true and which part must be recalculated or retested. Keep the original and changed results side by side.
REVIEW CRITERIA: Success appears only after a confirmed accepted result. A timeout is uncertain, and a retry must not blindly duplicate an effect.
MENTOR DISCUSSION: Explain why the changed condition matters. Identify a tempting wrong conclusion, show the observation that would challenge it, and choose one next test. A statement of uncertainty is better than inventing a result.
PORTFOLIO NOTE: Keep your baseline, changed input, prediction, observed or paper-traced result, explanation, and one improvement. These practice prompts do not replace the course's existing quiz or final assessment.
LEARNING GOAL: Independent Challenge & Mastery. Your finished work should demonstrate: An independent demonstration with evidence: Source files or repository, Responsive screenshots, Accessibility/test checklist, Revision notes.
EXPLAIN THE CONCEPT: Apply what you learned without copying the guided solution. Independent challenge: Run keyboard, mobile-width, link, and input tests; fix at least two findings and record evidence. Safety and responsibility: Collect no unnecessary personal data; never expose secrets in browser code; treat all form input as untrusted. Reflect on your work: What did you build or discover? What did you test or change after feedback? What safe or responsible choice did you make? Think about where you might use this skill in a real or imaginary project. Describe the problem it helps you solve and how you would recognize a successful result.
ORIGINAL PROJECT LAB: 1. Explain the challenge and make a short plan. 2. Run keyboard, mobile-width, link, and input tests; fix at least two findings and record

evidence. 3. Test the result and record evidence of your work. 4. Review the safety scenario and describe one improvement. Finished result: An independent demonstration with evidence: Source files or repository, Responsive screenshots, Accessibility/test checklist, Revision notes.

EXPLAIN THE MECHANISM: Create an original showcase for a different fictional project. Plan its structure and one useful interaction before styling. Explain what a reader needs to understand and which action should happen on the current page.

WORKED CASE: A helper cannot find the testing notes because every link is labeled More. Rename the relevant link Read testing notes and retest the same task. This improves meaning; changing the link color alone would not explain its destination.

TEST AND JUSTIFY: Present the HTML roles, one responsive rule, and one event-to-result trace. Include a keyboard test and a narrow-view check. Demonstrate locally or get adult help before publishing; remove credentials and personal information from the source and page.

GO DEEPER: Identify an assumption, test a safe alternative, and explain the trade-off. Keep a short before-and-after record that someone else could follow.

SAFETY CHECK: Recognize phishing and social engineering, secure accounts, understand privacy settings, and perform cyber labs only with authorization. Use pretend data, never real passwords or private information. Stop and ask a trusted adult if anything seems unsafe.

REVIEW QUESTIONS: What is the main idea? Why does the first step matter? How would you detect a mistake? What evidence proves the result? What would you improve? Answer in your own words before taking the quiz.

PORTFOLIO REFLECTION: Save a drawing, sanitized screenshot, or short note describing the goal, your approach, your evidence, and one lesson learned. Expected result: An independent demonstration with evidence: Source files or repository, Responsive screenshots, Accessibility/test checklist, Revision notes.

Investigate with pictures and choices

A page works with a mouse but not Tab.

What needs attention?

1. Only the color palette
2. Adding more animation
3. Keyboard operation and focus

Worked reasoning

Answer: Keyboard operation and focus. Pointer success does not establish keyboard accessibility.

Try a changed example

Complete the same task with tap-only and keyboard-only input.

Picture and meaning matches

HTML: Defines content structure

CSS: Controls presentation

JavaScript: Handles programmed behavior

Ways to respond

Choose, point, draw, demonstrate, type, speak or use your usual communication tool.

A helper can operate the controls while you choose.

Counting support if useful

Use zero to ten counters or dots. Touch each once. The last count names the quantity. Rearrange the same set and check whether anything was added or removed.

Choose this foundation practice only when it fits the learner goal.

My notes: _____

Hands-on lab

1. Explain the challenge and make a short plan.
2. Run keyboard, mobile-width, link, and input tests; fix at least two findings and record evidence.
3. Test the result and record evidence of your work.
4. Review the safety scenario and describe one improvement.

Finished result: An independent demonstration with evidence: Source files or repository, Responsive screenshots, Accessibility/test checklist, Revision notes.

Check your understanding

1. Which idea best describes Independent Challenge & Mastery?

A. It has no purpose in this project.

B. Semantic HTML helps people and browsers understand a page.

C. Apply what you learned without copying the guided solution.

D. Responsive CSS adapts layout to available space.

2. What is the first step in this lab?

A. Review the safety scenario and describe one improvement.

B. Run keyboard, mobile-width, link, and input tests; fix at least two findings and record evidence.

C. Test the result and record evidence of your work.

D. Explain the challenge and make a short plan.

3. What should you do after step two?

A. Test the result and record evidence of your work.

B. Run keyboard, mobile-width, link, and input tests; fix at least two findings and record evidence.

C. Review the safety scenario and describe one improvement.

D. Explain the challenge and make a short plan.

4. Which result shows that your lab is finished?

A. A blank page with no work

B. An independent demonstration with evidence: Source files or repository, Responsive screenshots, Accessibility/test checklist, Revision notes.

C. A copied answer without checking it

D. A skipped lab

Answer review

1. C. Apply what you learned without copying the guided solution.

2. D. Start with: Explain the challenge and make a short plan.

3. A. Step three is: Test the result and record evidence of your work.

4. B. The goal is: An independent demonstration with evidence: Source files or repository, Responsive screenshots, Accessibility/test checklist, Revision notes.

Mastery & safe-use plan

Ages 13-15

Digital safety: Recognize phishing and social engineering, secure accounts, understand privacy settings, and perform cyber labs only with authorization.

AI literacy: Evaluate bias, hallucinations, deepfakes, privacy risks, attribution, and responsible model use.

Challenge: Complete an independent debugging, defense, data, or design scenario with documented reasoning.

Portfolio: Publish a sanitized project artifact with README-style purpose, steps, screenshots or evidence, and lessons learned.

Mastery check: Explain the design, demonstrate testing, identify a security or ethics risk, and defend one improvement.

Project worksheet

My project: Build a responsive multi-page website

Who will use or review it? _____

My plan: _____

Checkpoint 1: Two connected pages.

Checkpoint 2: A site that fits phone and desktop widths.

Checkpoint 3: A working interactive tip button.

Checkpoint 4: A completed guided project with documented evidence: Source files or repository, Responsive screenshots, Accessibility/test checklist, Revision notes.

Checkpoint 5: An independent demonstration with evidence: Source files or repository, Responsive screenshots, Accessibility/test checklist, Revision notes.

One thing I tested: _____

One thing I changed after testing: _____

What I learned: _____

Trusted adult or mentor feedback: _____

Family review

Show your finished work to a trusted adult. Explain one thing you learned and one thing you would improve.

Beat IT Kids | For learning at home. Do not put real passwords or private details in practice projects.