

Connect a working contact form

Define and test a real submission path rather than a decorative form.

THE FINISHED RESULT

Valid data arrives once in the intended backend, errors remain visible, and the page never reports success on a failed request.

Before you start

Environment: HTML, CSS and JavaScript; Python local preview server

Access and plan: Text editor, modern browser and Python for local preview. Hosting and form-backend accounts are needed only for their corresponding lessons.

Practice project: Harbor Home Care

Use a separate demo project and fictional information. Record the exact project and version you use. Keep passwords, verification codes and private customer information out of screenshots and notes.

Official walkthroughs

MDN: [Web forms](#)

Open the credited publisher walkthrough for current interface instructions and available screenshots, then return to the SETLY exercise. External courses remain with their publishers; this PDF does not reproduce their complete tutorial.

Your practice task

Finish one action at a time. Inspect the resulting screen or record before moving on. If a control differs, check the matching official instructions instead of guessing.

01 / Step 1

Create labelled name, email and message fields in the supplied example. Mark required inputs and preserve their names for submission.

Check: record what changed and confirm you are still working in the intended demo project.

02 / Step 2

Select a form backend you control and configure its test endpoint. Keep credentials on the server; never add a secret key to the page's JavaScript.

Check: record what changed and confirm you are still working in the intended demo project.

03 / Step 3

Submit demo data, inspect the backend record and then verify any expected notification. Handle success and failure as different interface states.

Check: record what changed and confirm you are still working in the intended demo project.

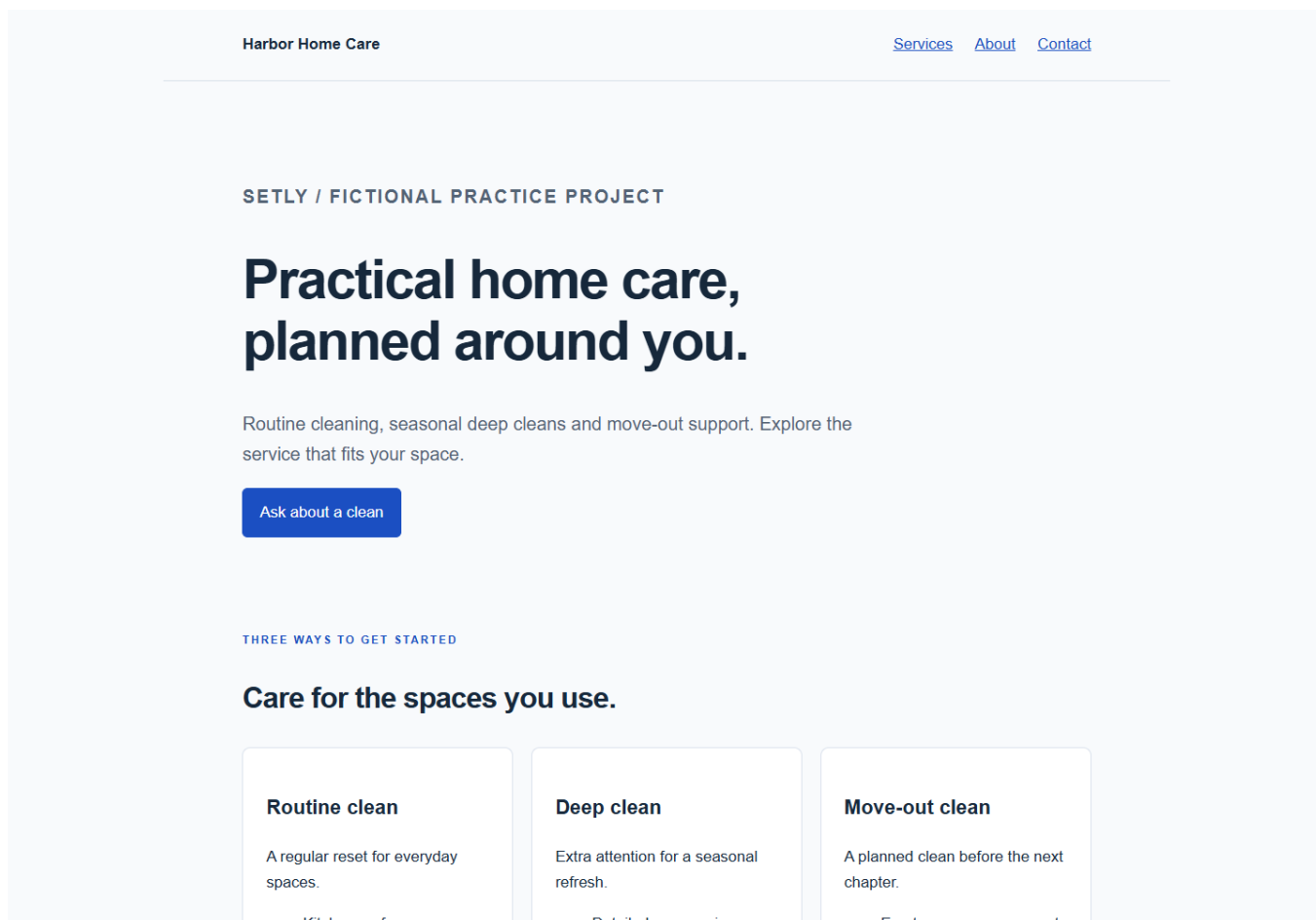
04 / Step 4

Test invalid email, missing content, duplicate clicks and an unavailable endpoint. Preserve the user's message when submission fails.

Check: record what changed and confirm you are still working in the intended demo project.

The demo contact form and its backend requirement

Captured from the supplied website example. The static contact form needs a backend; the downloadable educational backend stores demo enquiries locally.



Source: Ray / SETLY. Original SETLY demo screenshot.

<https://ray.setlyglobal.com/examples/harbor/>

Test, troubleshoot, repeat.

If something goes wrong

If requests fail, inspect status codes and endpoint configuration. A mailto link opens an email application; it is not evidence of stored form submission.

If your screen is different

Compare the product, editor, user permissions and selected account with the guide. Open the official reference for your version. Do not perform a live change whose effect you cannot verify.

Final verification

Valid data arrives once in the intended backend, errors remain visible, and the page never reports success on a failed request.

Repeat the task with one small variation. Record the test date, project/version, expected result, observed result and a non-sensitive evidence reference. Mark skipped checks as untested.

Your evidence record

Test date: _____ Project/version: _____

Observed result: _____

Evidence reference: _____

Remaining issue and next action: _____

Sources and guide status

Original exercise by Ray / SETLY. Credited technical references are linked on page one. Prepared 1 October 2026. SETLY screen recording and independent learner reproduction are not yet complete. No platform affiliation is implied.

[Open the online practice guide and editable checklist](#)