



EMAIL ADVICE IN YOUR INBOX

Let your readers *skip ahead*.

Build an email course on the platform you already use, and give keen readers a "send me the next lesson now" button. If your platform can't do that on your plan, two free tools can.

Time: about 2 hours, plus writing **Cost:** free tools, plus your email platform **By:** Email Advice in Your Inbox

First, which path are you on?

PATH A

Your platform has click triggers

Your email platform can start an automation when someone clicks a specific link, on the plan you're paying for.

Do steps 1, 2 and 6.

PATH B

It doesn't, so build the bridge

No click trigger, or it's on a plan you don't want to pay for. A free page records the click and a free automation tool starts the next lesson.

Do all six steps.

How to check: open your platform's automations and look at the list of triggers (what starts an automation). You're looking for something like "clicks a link", "link clicked" or "link trigger". Names vary by platform. If it's there and available on your plan, you're on Path A.

What you'll need

- ☐ An email platform with automations
Beehiiv, Kit, MailerLite, Mailchimp, Brevo and others
- ☐ An idea for a 5 to 7 day course
Step 1 turns it into an outline
- ☐ A free Netlify account (Path B)
Or a page on your own website
- ☐ A free Make account (Path B)
Zapier or n8n work the same way
- ☐ Your platform's API key, or its Make app (Path B)
Found in your platform's settings, usually under "API" or "Integrations"
- ☐ A test email address
A "+" alias works: you+course1@yourdomain.com

1

Plan your course with one prompt

BOTH PATHS

Fill in the bits in lime, paste it into ChatGPT, Claude or Gemini, and answer its questions. You'll get a day-by-day outline you can write from.

PROMPT · COURSE OUTLINE

You're an experienced email educator.
Help me plan a free email course that
I'll use as a lead magnet.

ABOUT ME

- My niche: [e.g. home baking, B2B sales, podcasting]
- Who it's for: [describe one ideal reader in a sentence]
- Their biggest frustration right now: [what they're stuck on]
- What I sell or offer: [product, service or "nothing yet"]

THE COURSE

- Length: [5 to 7] days, one email a day
- The promise: by the last day, readers will [specific, achievable outcome]
- Language and tone: [e.g. British English, warm and practical]

RULES FOR EVERY LESSON

- One idea and one small task the reader can finish in 15 minutes
- 250 to 400 words, plain language, no jargon without a quick explanation
- Builds on the day before, so the course feels like a path

FOR EACH DAY, GIVE ME

1. Subject line (under 50 characters) and preview text
2. The one idea, in two sentences
3. The task, as 2 to 4 steps
4. A reply question (something the reader can answer in one line)
5. A one-line teaser for tomorrow
6. The skip line: "Don't want to wait? Get Day [next] now."

SHAPE OF THE COURSE

- Day 1: a quick win, and ask readers to hit reply
- Middle days: the core skills, in the order a beginner needs them
- Last day: recap, what to do next, and a gentle invitation to [my offer, or my newsletter]

ALSO GIVE ME

- 3 course name ideas
- A one-sentence sign-up promise for my landing page
- 4 short bullets for the sign-up form

Before you start, ask me up to 5 questions if anything above is unclear.

Keep it to 5 to 7 days. Long enough to build a habit of opening your emails, short enough that people finish. Once the outline is right, ask the same chat to draft each lesson one at a time, then edit them into your own voice.

2

Set it up in your email platform

BOTH PATHS

The trick that makes skipping possible: build **one automation per lesson**, not one long sequence. Then you can drop anyone into any day.

- 1 Create the lesson template.** Same header image, from name and sign-off on every lesson, so it feels like one course. At the bottom of Days 1 to 6, add a button: "Send me Day 3 now" (with the right number).
- 2 Build Day 1.** Trigger: the course sign-up. Steps: add a course tag, send the Day 1 email, wait 1 day, then start Day 2.
- 3 Build Days 2 to 7 the same way.** Trigger: started by the day before (and by the skip button). Steps: send the email, wait 1 day, start the next day. The last day has no next step.
- 4 Set every lesson to "enter once".** Some platforms call it "limited" enrolment, or "don't re-enter". This stops someone who skipped ahead getting the same lesson again when the daily chain catches up.
- 5 Keep course sign-ups out of your normal welcome series,** so they don't get two welcome emails at once. Exclude the course tag or source from its trigger.

How "start the next day" works on different platforms

PLATFORM	START A LESSON BY...	TIP
Beehiiv	Enrolling the subscriber in that day's automation	Each automation has an "add to automation" step and an API trigger
Kit	Adding a tag (for example <code>course-day-3</code>) that starts that day's automation	Tags keep the logic simple
MailerLite	Adding the subscriber to a group that triggers that day	One group per day
Mailchimp	Applying a tag that starts a customer journey	One journey per day
Brevo	Adding the contact to a list that triggers that day	One list per day

Path A: wire the skip button here.

Give each skip button its own link (for example `yoursite.com/day-3`, a simple "on its way" page). Then set Day 3's trigger to "clicks the Day 3 link". That's it: skip to step 6.

Path B: point the skip button at your page.

Link it to `https://yoursite.com/next/3?s=SUBSCRIBER_ID`, swapping in your platform's merge tag for the subscriber's ID (on Beehiiv it's `{{api_subscription_id}}`; search your platform's help for "subscriber ID merge tag"). Use the ID rather than the email address where you can, so addresses don't end up in links.

3 Create the "on its way" page

PATH B

When a reader clicks "Send me Day 3 now", they land on a small page that says *Day 3 is on its way* and quietly records the request.

Why a page, not a direct link to Make? Work inboxes often run security scanners that "click" every link to check it. If the button went straight to your automation, a scanner could send Day 3 before the reader even opened Day 2. This page only records the click when it opens in a real browser, which filters most of that out.

Option 1: [Netlify](#) (free, recommended). Netlify hosts simple pages for free, collects form submissions at no cost, and can forward each one to Make. No coding beyond pasting two files.

- 1 On your computer, make a folder called `site`.
- 2 Inside it, save the page below as `next.html`.
- 3 Save a second file called `_redirects` (no file extension) with the one line below. It sends `/next/3`, `/next/4` and so on to the same page.
- 4 [Sign up free at netlify.com](#), choose **Add new project** → **Deploy manually**, and drag the `site` folder in.
- 5 Under **Forms**, turn on **form detection**, then drag the folder in once more so Netlify finds the form.
- 6 Optional: under **Domain management**, add a subdomain such as `course.yourdomain.com`.

next.html

```
<!doctype html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width,initial-scale=1">
<meta name="robots" content="noindex,nofollow">
<title>Your next lesson is on its way</title>
<style>
  body{margin:0;min-height:100vh;display:grid;place-items:center;padding:24px;
    font:17px/1.6 system-ui,sans-serif;background:#f4f4fc;color:#1b1b3a}
  .card{background:#fff;max-width:520px;width:100%;padding:40px 32px;border-radius:16px;text-align:center}
  h1{margin:0 0 12px;font-size:30px}
  a{color:#2d2cac}
</style>
</head>
<body>
<main class="card">
  <h1 id="title">Your next lesson is on its way.</h1>
  <p id="msg">Check your inbox in a few minutes.</p>
  <p><a href="/">Back to YOUR SITE NAME</a></p>
</main>

<!-- Netlify reads this hidden form so it knows what to collect -->
<form name="next-lesson" data-netlify="true" hidden>
  <input name="subscriber"><input name="lesson">
</form>

<script>
(function () {
  var m = location.pathname.match(/\/next\/(\d+)\//?$/);
  var s = new URLSearchParams(location.search).get('s') || '';
  var title = document.getElementById('title');
  var msg = document.getElementById('msg');
  if (!m || !s) {
    title.textContent = 'That link didn't work.';
    msg.textContent = 'Use the button in your latest email. Your next lesson still arrives tomorrow.';
    return;
  }
  title.textContent = 'Day ' + m[1] + ' is on its way.';
  var key = 'next-' + s + '-' + m[1];
  try { if (sessionStorage.getItem(key)) return; } catch (e) {}
  fetch('/', {
    method: 'POST',
    headers: { 'Content-Type': 'application/x-www-form-urlencoded' },
    body: new URLSearchParams({ 'form-name': 'next-lesson', subscriber: s, lesson: m[1] }).toString()
  }).then(function (r) {
    if (!r.ok) throw 0;
    try { sessionStorage.setItem(key, '1'); } catch (e) {}
  }).catch(function () {
    msg.textContent = 'We couldn't confirm that. Refresh to try again, or wait for tomorrow's email.';
  });
})();
</script>
</body>
</html>
```

_redirects

```
/next/* /next.html 200
```

Option 2: your own website. Add a page with the same content, and have it send the request straight to Make instead of to Netlify. Your skip link becomes `yoursite.com/next/?lesson=3&s=SUBSCRIBER_ID`. Replace the script in the page above with this one:

SCRIPT · OWN WEBSITE VERSION

```
<script>
(function () {
  var q = new URLSearchParams(location.search);
  var lesson = q.get('lesson'), s = q.get('s');
  var title = document.getElementById('title');
  var msg = document.getElementById('msg');
  if (!lesson || !s) {
    title.textContent = 'That link didn't work.';
    msg.textContent = 'Use the button in your latest email.';
    return;
  }
  title.textContent = 'Day ' + lesson + ' is on its way.';
  fetch('YOUR_MAKE_WEBHOOK_URL', {
    method: 'POST',
    mode: 'no-cors',
    body: new URLSearchParams({ form_name: 'next-lesson', subscriber: s, lesson: lesson })
  });
})();
</script>
```

Option 3: let your AI assistant build it.

PROMPT · BUILD THE PAGE

Build me a single, mobile-friendly HTML page called next.html for my email course, to host on Netlify.

What it does:

- It's opened from links like /next/3?s=SUBSCRIBER_ID (the number is the lesson).
- It shows "Day 3 is on its way. Check your inbox in a few minutes." using the number from the link.
- If the number or the s value is missing, it shows a friendly "That link didn't work" message instead.
- It submits a hidden Netlify form

called "next-lesson" with two fields, subscriber and lesson, using fetch to "/".

- It only submits once per browser session for the same subscriber and lesson.
- It includes the hidden form in the HTML so Netlify can detect it, and a noindex meta tag.

Style it to match my brand: [colours, font, logo URL]. Also give me the one-line _redirects file that sends /next/* to next.html.

4

Set up Make

PATH B

[Make](#) is the bridge: it hears "subscriber X wants Day 3" and tells your email platform to start Day 3 for them.

Why Make? The free plan gives you 1,000 credits a month and two active scenarios (as of October 2026). Each skip uses only a few credits, so a few hundred skips a month costs nothing. It also has ready-made apps for most email platforms, so you rarely need to write an API request.

- 1 [Sign up free at make.com](#) if you haven't already. Then **create a scenario** and add **Webhooks → Custom webhook** as the first module. Name it (for example "Course skip") and copy the webhook URL it gives you.
- 2 **Send it one test.** Click **Redetermine data structure**, then open your page once with a test link (step 5 connects Netlify first). Make now knows the fields: `subscriber` and `lesson`.
- 3 **Add a filter** after the webhook: *form_name equals next-lesson*. Pick `form_name` from the list of incoming data. Don't type it, or the filter never matches.
- 4 **Add your email platform's action.** Use its Make app if it has one (Kit, MailerLite, Mailchimp and Brevo all do), or **HTTP → Make a request** with your API key in the header. Examples below.
- 5 **Add an error handler** (right-click the action → **Add error handler → Ignore**), so one bad link doesn't stop the scenario.
- 6 **Save, then switch the scenario on.** Leave scheduling on "Immediately".

What the action looks like

PLATFORM	MAKE MODULE	SETTING
Kit	Kit → Tag a subscriber	Tag name: <code>course-day-</code> + mapped <code>lesson</code>
MailerLite	MailerLite → Add subscriber to group	Group: the one for that day
Mailchimp	Mailchimp → Add/remove member tags	Tag: <code>course-day-</code> + mapped <code>lesson</code>
Brevo	Brevo → Add contacts to a list	List: the one for that day
Beehiiv	HTTP → Make a request	See the example below

One action for every day.

Name your tags or groups by day number (`course-day-2` , `course-day-3` ...) and build the name from the incoming `lesson` value. Then a single action covers the whole course, with no separate route per day. Where a platform needs a different ID per day (lists, groups, automations), use Make's `switch()` function, as in the Beehiiv example.

BEEHIIV EXAMPLE · HTTP REQUEST

Method: POST
URL: `https://api.beehiiv.com/v2/publications/YOUR_PUBLICATION_ID/automations/{{switch(1.data.lesson; "2"; "aut_DAY2"; "3"; "aut_DAY3"; "4"; "aut_DAY4"; "5"; "aut_DAY5"; "6"; "aut_DAY6"; "7"; "aut_DAY7")}}/journeys`

Headers:
Authorization: Bearer `YOUR_API_KEY`
Content-Type: `application/json`

Body (raw JSON):
`{ "subscription_id": "{{1.data.subscriber}}" }`

Set Make's authentication to "No authentication": the key lives in the header.
Build `{{...}}` values by clicking fields from the webhook, not by typing them.

5

Bring it all together

PATH B

SIGN-UP

Course form



Email platform: Day 1



Day 2, Day 3... one a day

SKIP AHEAD

"Send me Day 3 now"



Your page



Netlify form



Make



Email platform: Day 3 now

Connect the pieces

WHERE	WHAT TO SET
Netlify	Forms → Form notifications → Add notification → HTTP POST request. Event: new form submission. Form: <code>next-lesson</code> . URL: your Make webhook URL.
Email platform	Skip buttons on Days 1 to 6 point to <code>yoursite.com/next/[next day]?s=[subscriber ID merge tag]</code> .
Make	Webhook → filter (<code>next-lesson</code>) → your platform's action. Scenario switched on.
Sign-up form	Easiest: use your email platform's own form, set to start Day 1. If your form lives on Netlify, add a second notification and a second route in Make that creates the subscriber and starts Day 1.

Using one Make webhook for several forms?

Add a

Router

after the webhook with one route per form, each with its own filter on `form_name`. Then the sign-up form and the skip page can share one webhook and one scenario.

6

Test, then go live

BOTH PATHS

Use a "+" alias of your own address for each test run (you+course1@, you+course2@), so every test counts as a new subscriber. Tick these off as you go.

- ☐ Sign up. Day 1 arrives within a few minutes, from the right sender, with the course tag added.
- ☐ Click "Send me Day 2 now". The page says Day 2 is on its way, and Day 2 arrives within a few minutes.
- ☐ Path B: Make shows a successful run, with each module marked green.
- ☐ Skip once more (Day 3), then wait 24 hours. The next day arrives on its own, and nothing you've already had arrives again.
- ☐ The test subscriber didn't also get your normal welcome series.
- ☐ Every lesson looks right on a phone, and every skip button has the right day number.
- ☐ Path B: clear any old test data waiting in Make's webhook queue (Webhooks → your webhook → Queue) before you announce.
- ☐ Go live: add the sign-up form to your site, bio links and newsletter footer.

If something doesn't work

WHAT YOU SEE	CHECK THIS
The form submits but nothing reaches Make	Netlify's HTTP POST notification for that form exists and points to the right webhook URL.
Make runs but stops at the filter	The filter uses <code>form_name</code> picked from the incoming data, not typed text.
The subscriber arrives, but no email	The action starts the right automation, tag or list, and that automation is published.
A lesson arrives twice	"Enter once" is switched on for every lesson.
A route you copied behaves like the old one	Copying a module copies its settings. Check its body, tag or automation ID.

SEE IT IN ACTION

Take the course this playbook came from.



THE ZOMBIE NEWSLETTER REVIVAL KIT

BY

EMAIL 
EXPERT
AFRICA

The Zombie Newsletter Revival Kit is a free 7-day course for anyone whose newsletter has gone quiet. One short email a day, one practical task each, and a "send me the next lesson now" button built exactly as described above. By Day 7, your list hears from you again.

Start the free 7-day course



**EMAIL ADVICE
IN YOUR INBOX**

Practical email advice, every week. Written by Des Brown, [Email Expert Africa](#).

[Subscribe to Email Advice in Your Inbox](#)