HTML Autofill/Autocomplete: Boost UX, Conversions, and Accessibility

What is Autofill on Website Forms?

Autofill is one of those tiny features that quietly boosts conversions and accessibility. When you add the right `autocomplete` tokens to your inputs, browsers and password managers can correctly prefill names, emails, addresses, phones, passwords, and even one-time codes. Less typing, fewer errors, happier users.

- Autofill: The front-end browser feature that remembers user data.
- Autocomplete: Form attributes to allow or disallow autofill.

Why AutoComplete Matters

You should be using autocomplete on all forms you have on your website.

- **Faster forms = higher completion rates.** Autofill reduces friction.
- **Fewer typos.** Better data quality in your CRM and fewer support emails.
- **Accessibility.** Proper tokens help assistive tech identify field purpose (WCAG 2.2: Identify Input Purpose).

Golden Rules for Autofill

1. Use **field-level tokens** (e.g., `email`, `name`, `address-line1`, `postal-code`, `country-name`, `tel`, `username`, `current-password`, `new-password`, `one-time-code`), not just `autocomplete="on"`.
2. Always include **visible labels** tied to inputs with `for`/`id`.
3. Use sensible input types: `type="email"`, `type="tel"`, `type="password"`, and `inputmode="numeric"` for ZIP/OTP.
4. For two addresses (shipping + billing), use **sections** (see example below) so saved addresses don’t bleed into the wrong fields.
5. Don’t fight password managers: `autocomplete="off"` is often ignored on login fields—use the correct tokens instead.
6. Payment fields: prefer your gateway’s **hosted/iframe** fields (Stripe/Adyen/Braintree). Only use `cc-*` tokens if you truly handle card inputs (PCI scope applies).

Common Fields ? the Right Tokens

- **Name:** `name` (or split into `given-name`, `additional-name`, `family-name`)
- **Email:** `email`
- **Phone:** `tel` (can be qualified by `home`/`work`/`mobile` if needed)
- **Address lines:** `address-line1`, `address-line2`, `address-line3`
- **City:** `address-level2` | **State/Region:** `address-level1`
- **Country (name):** `country-name` | **Country (code):** `country`

- **ZIP/Postal:** `postal-code`
- **Auth:** `username`, `current-password`, `new-password`, `one-time-code`
- **Cards (if you must):** `cc-name`, `cc-number`, `cc-exp` (or month/year split), `cc-csc`, `cc-type`

Copy/Paste Starter Form: Good defaults to use

Place this in a Gutenberg “Custom HTML” block, a template part, or your form template.

Copy

```
<form action="/submit-form" method="post" autocomplete="on">
  <!-- Identity -->
  <label for="name">Full name</label>
  <input id="name" type="text" name="name" autocomplete="name" required="">

  <label for="email">Email</label>
  <input id="email" type="email" name="email" autocomplete="email" required="">

  <!-- Optional username -->
  <label for="username">Username (optional)</label>
  <input id="username" type="text" name="username" autocomplete="username">

  <!-- Passwords -->
  <label for="current-password">Current password</label>
  <input id="current-password" type="password" name="currentPassword" autocomplete="current-
password">

  <label for="new-password">New password</label>
  <input id="new-password" type="password" name="password" autocomplete="new-password">

  <!-- Address -->
  <label for="address-line1">Address line 1</label>
  <input id="address-line1" type="text" name="address1" autocomplete="address-line1">

  <label for="address-line2">Address line 2</label>
  <input id="address-line2" type="text" name="address2" autocomplete="address-line2">

  <div class="address-group">
    <div>
      <label for="city">City</label>
      <input id="city" type="text" name="city" autocomplete="address-level2">
    </div>
    <div>
      <label for="state">State/Region</label>
      <input id="state" type="text" name="state" autocomplete="address-level1">
    </div>
    <div>
      <label for="postal-code">ZIP/Postal</label>
      <input id="postal-code" type="text" name="zip" autocomplete="postal-code"
inputmode="numeric">
    </div>
  </div>

  <label for="country">Country</label>
  <input id="country" type="text" name="country" autocomplete="country-name">

  <!-- Phone -->
  <label for="phone">Phone</label>
  <input id="phone" type="tel" name="phone" autocomplete="tel">

  <!-- One-time code (for 2FA flows) -->
  <label for="otp">One-time code</label>
```

```
  <input id="otp" type="text" name="otp" autocomplete="one-time-code" inputmode="numeric">

  <!-- Field intentionally not autofilled -->
  <label for="secret">Security answer</label>
  <input id="secret" type="text" name="secret" autocomplete="off">

  <button type="submit">Submit</button>
</form>
```

Notes for payment forms

Use your payment provider's hosted/iframe fields for cards. If you render native inputs, you're in PCI scope—only do that deliberately. If required, card tokens look like: `cc-name`, `cc-number`, `cc-exp`, `cc-csc`.

Shipping vs. Billing: Keep saved addresses separate

If you collect two addresses, wrap each set with a unique section token so the browser offers the right saved address in the right place.

Copy

```
<fieldset>
  <legend>Shipping</legend>
  <label for="ship-address1">Address line 1</label>
  <input id="ship-address1" name="ship_address1" autocomplete="section-ship shipping address-line1">

  <label for="ship-city">City</label>
  <input id="ship-city" name="ship_city" autocomplete="section-ship shipping address-level2">

  <label for="ship-zip">ZIP/Postal</label>
  <input id="ship-zip" name="ship_zip" autocomplete="section-ship shipping postal-code">
</fieldset>
```

Copy

```
<fieldset>
  <legend>Billing</legend>
  <label for="bill-address1">Address line 1</label>
  <input id="bill-address1" name="bill_address1" autocomplete="section-bill billing address-line1">

  <label for="bill-city">City</label>
  <input id="bill-city" name="bill_city" autocomplete="section-bill billing address-level2">

  <label for="bill-zip">ZIP/Postal</label>
  <input id="bill-zip" name="bill_zip" autocomplete="section-bill billing postal-code">
</fieldset>
```

Testing Checklist: Quick pass before launch

- Verify every input has the **correct** token for its purpose.
- Confirm labels are present and tied to inputs via `for`/`id`.
- Use realistic test data and try in multiple browsers (Chrome, Safari, Firefox, Edge) with saved profiles.
- On mobile, ensure SMS codes surface via the keyboard suggestion bar (thanks to `one-time-code`).
- If a form plugin strips attributes, switch to an HTML field/markup or consult the plugin's filter/hooks to pass through `autocomplete`.

Gotchas: So you don't chase ghosts

- **Disabling autofill:** Modern browsers often ignore `autocomplete="off"` on login fields. Use the proper password tokens instead.
- **Country:** Use `country-name` for "United States" style entries. Use `country` for two-letter codes only if your UI expects them.
- **ZIP and phone:** Keep them as text (`type="text"` or `type="tel"`) to preserve leading zeros; drive numeric keyboards with `inputmode="numeric"`.
- **Custom fields:** There's no custom token like `favorite-color`. For non-standard fields, rely on labels and leave `autocomplete` unset or `off` if you don't want suggestions.

FAQs About Autofill / Autocomplete

Does `autocomplete="on"` on the form tag make tokens unnecessary?
No. Keep it on, but field-level tokens are what actually drive reliable behavior across browsers.

Can I force browsers to stop autofilling passwords?
Not reliably. Use `username`, `current-password`, and `new-password` so managers behave correctly, and secure the flow server-side.

What about Gravity Forms / Contact Form 7 / WPForms?
Most will output attributes you include in HTML fields/markup. If a specific field type strips attributes, add a raw HTML field, or use the plugin's hooks/filters to pass through `autocomplete`.

Need Help? We Can Wire This Into Your Site

Want my team to review and fix your form autofill? We'll add the right tokens, test across browsers, and improve completion rates

Contact My Team

Original article: https://tonyherman.com/autocomplete-in-html/
PDF version: https://tonyherman.com/autocomplete-in-html/?apa_pdf=1

Special Offer for Readers

1,000+ Channels • On-Demand Movies • **5 Devices**

$69.99/mo (~$2.33/day)

Try it for a month and see if you like it, then switch.

Start Your Trial

Click to learn more about CUE Broadcast

Tip: Get 3 friends of family to sign up and you it for free.