



Header & Footer Code

USER MANUAL · Version 1.0.2 · PrestaShop 1.7.8 – 9.x

1. What this module does

The module adds your own code to the storefront without editing the theme: Google Tag Manager, Google Analytics, Meta Pixel, verification meta tags for Google Search Console or Bing, chat widgets and similar snippets. You paste the code into the module configuration and choose where it goes: in the page head, right after the opening body tag or at the end of the page. A separate field takes the conversion code for the order confirmation page, with the order number, reference and amounts filled in. Two more fields take your own CSS and JavaScript.

The code lives in the shop database, not in theme files, so it stays after theme updates or a theme change, and every shop of a multistore can have its own code.

Works with **PrestaShop 1.7.8, 8.x and 9.x** (PHP 7.2.5 or newer), with the Classic and Hummingbird themes and with multistore.

2. Installation

1. On the module page at addonspresta.com, click **Free download** (no account needed).
2. In your Back Office, open **Modules** → **Module Manager** and click **Upload a module**.
3. Drop the `ap_customcode_v1.0.1.zip` file and wait for the confirmation.
4. Click **Configure** on the module card.

3. How it works

The module prints each code block through a standard PrestaShop hook of the theme:

- **Head code** and **Custom CSS** — at the end of the `<head>` section, after the styles of the theme (hook `displayHeader`);
- **Code at the start of the body** — right after the opening `<body>` tag (hook `displayAfterBodyOpeningTag`);
- **Code at the end of the body** and **Custom JavaScript** — before the closing `</body>` tag, after the scripts of the theme (hook `displayBeforeBodyClosingTag`);
- **Conversion code** — on the order confirmation page only (hook `displayOrderConfirmation`).

The Classic theme of PrestaShop 1.7.8, 8 and 9 and the Hummingbird theme of PrestaShop 9 show all these hooks on every storefront page. A theme of another author may leave out the body hooks; ask its author if code at the start or end of the body does not appear. The maintenance page and the "country not allowed" page have no hooks, so no code appears there.

The code is printed exactly as you paste it. The module does not check or change it, except that a `<style>` or `<script>` wrapper pasted into the CSS or JavaScript field is removed, because the module adds its own. A block that is switched off or empty prints nothing at all.

The conversion code runs each time the confirmation page is displayed. In the code, `{order_id}`, `{order_reference}`, `{total_paid}` (with tax), `{total_paid_tax_excl}`, `{total_products}` (products without tax), `{total_shipping}` (with tax) and `{currency}` (ISO code, for example `EUR`) are replaced with the values of the order shown on the page. Numbers are inserted as numbers rounded to two decimals and texts as quoted JavaScript strings, so they can be used directly in a script and cannot break it. No personal data of the customer (name, e-mail, address) is available. When one checkout creates several orders (one per carrier), the values are those of the order shown on the confirmation page.

The module itself does not connect to addonspresta.com or to any other server and does not set cookies. The code you paste usually does: it is your responsibility, and tracking scripts must respect the cookie consent of your visitors (GDPR). Use the consent features of the tool you add (for example Google Consent Mode in Google Tag Manager) or a cookie consent module.

4. Features and settings

Open **Modules** → **Module Manager** → **Header & Footer Code** → **Configure**. The Help box at the top links to this manual and to the module page at addonspresta.com, in the language of your Back Office.

Every block has an **On / Off** switch and a code field. Switching a block off keeps its code but stops printing it, which is handy when a script causes trouble. Each field holds up to 48,000 bytes. One **Save** button saves all the blocks.

Code on every page

- **Head code** — Google Tag Manager (the `<script>` part), Google Analytics, Meta Pixel, verification meta tags such as `<meta name="google-site-verification" content="...">`.
- **Code at the start of the body** — the `<noscript>` part of Google Tag Manager.
- **Code at the end of the body** — chat widgets and scripts that do not need to load early, including external scripts such as `<script src="https://..." async></script>`.

Order confirmation

Conversion code — Google Ads conversion, the Meta Pixel **Purchase** event, affiliate networks. Write the variables without quotes, the module adds them where needed:

```
<script>
  gtag('event', 'purchase', {
    transaction_id: {order_reference},
    value: {total_paid},
    currency: {currency}
  });
</script>
```

Custom CSS and JavaScript

- **Custom CSS** — CSS rules without `<style>` tags. They load after the styles of the theme, so they win over theme rules with the same selector.
- **Custom JavaScript** — JavaScript without `<script>` tags. It runs at the end of the page, after the scripts of the theme. To load an external script file, paste the whole `<script src="...">` tag into **Code at the end of the body** instead.

The CSS field cannot contain `</style>` and the JavaScript field cannot contain `</script>` or `<!-- :` such code would close the module's own tag early or keep it from closing and break the page, so saving shows an error.

5. Multistore

The code is saved for the shop or shop group selected in the Back Office header; a note above the form tells you which one you are editing. Select **All shops** to set code shared by every shop, or a single shop to give it its own code. Code saved for a single shop takes precedence over the code of its group or of all shops. In the All shops or group context only an employee who manages every shop of it can save. One licence covers every shop of the installation.

6. Uninstall

Module Manager → **Header & Footer Code** → **Uninstall** removes all the code and settings of the module. Copy your code somewhere first if you may need it again, or use **Disable** instead, which keeps everything.

7. FAQ and troubleshooting

The code does not appear in the page source. Check that the block is switched on and saved, then clear the cache in **Advanced Parameters** → **Performance**. If a page cache module or a CDN is in front of the shop, clear its cache too. If only the body blocks are missing, your theme may not show those hooks (see How it works).

Google Search Console cannot find my verification tag. Paste the `<meta ...>` tag into **Head code**, save, clear the cache and open the home page source: the tag must be inside `<head>`. Then start

the verification again.

My shop looks broken after I saved some code. Switch the block off and save: the code stays in the field, so you can fix it calmly. The same applies to a script that shows errors in the browser console.

The conversion is counted twice. The confirmation page runs the conversion code again when the customer reloads it. Pass `{order_reference}` or `{order_id}` as the transaction ID — Google Ads, Meta and most affiliate networks then count each order once.

Why must I write `{total_paid}` without quotes? The module inserts ready JavaScript values: `123.45` for amounts and `"EUR"` with quotes for texts. Extra quotes would turn the amount into a text.

Can I use the variables in an image pixel or a link? Not directly. The variables are made for JavaScript: a text such as `{order_reference}` arrives with its own quotes, which would cut an HTML attribute like `` short. Build the pixel address in a script instead:

```
<script>
  new Image().src = 'https://example.com/pixel?ref=' +
  encodeURIComponent({order_reference});
</script>
```

How do I keep the code off some pages? In **Design** → **Positions**, open the hook of the block (for example `displayHeader`), edit the Header & Footer Code entry and add the pages to its exceptions.

Can I change how the custom CSS or JavaScript is printed? Yes. Copy `views/templates/hook/style.tpl` or `script.tpl` to `themes/<your-theme>/modules/ap_customcode/views/templates/hook/` and edit the copy, for example to add an attribute required by your cookie consent tool.

8. Updates and support

Updates are free: download new versions from the module page at addonspresta.com and upload them like a new module. Your code and settings are kept. Every version passes our security audit before it is published.

This free module comes under the same licence as our paid modules (one shop with any number of multistore shops, code you may modify but not share) and **without technical support**. If you need help with installation or configuration, we can offer it as a paid service. Report security issues to mail@addonspresta.com — we fix them as a priority.

-->