

Cognito Form: Embed script implementation

First, make sure all the UTM fields are created in Cognito Form. Each field should be separate and Field Name should match the name you will use later in the implementation (see the image below).

The image shows the Cognito Form editor interface. On the left is the 'Field Settings - Textbox' panel with the following fields: 'Label' (utm campaign), 'Field Name' (Utmcampaign), 'Type' (Single Line selected), 'Placeholder Text' (test), and 'Format Validation'. On the right is a live preview of the form titled 'test' with the label 'testing'. It features an 'Email' field, a 'utm campaign' field containing 'A test', a 'Submit' button, and a confirmation message: 'Thank you for filling out the form. Your response has been recorded.' A context menu is visible over the 'utm campaign' field with options: Cut, Copy, Delete, Insert Field, Make Smaller, Make Bigger, and Justify Row.

Your default embed code from Cognito Form should look like this

```
<script src="https://services.cognitoforms.com/s/<YOUR FORM ID>"></script>
<script>Cognito.load("forms", { id: "2"});</script>
```

Modify the embed code from Cognito Forms like the following. Notice that, we are adding entry fields to the `Cognito.load`.

```
<script src="https://services.cognitoforms.com/s/<YOUR FORM ID>"></script>
<script>Cognito.load("forms", { id: "2",
entry: {
  ["UTMCampaign":Cookies.get("utm_campaign"),
    "UTMSource":Cookies.get("utm_source") ,
```

```
"UTMMedium":Cookies.get("utm_medium"),
"UTMContent":Cookies.get("utm_content"),
"UTMTerm":Cookies.get("utm_term"),
"IP": Cookies.get("handl_ip"),    "Organic": Cookies.get("organic_source_str")
});</script>
```

Here we only illustrated `utm_*` and `handl_ip`. However you can use all the other parameters very similar way. See the list of all the parameters available here

Native WP Shortcodes

NOTE: Cognito does not support hidden field in the form out of the box. However you can make a regular text field hidden by creating a condition that never satisfies like this.

The image shows a form configuration interface with a 'Visible When...' dialog box. The dialog box has a teal header and a white body. It contains a condition: 'Email' (selected from a dropdown) 'contains' (selected from a dropdown) '123131312asdsad' (entered in a text field). There are '+ and' and '- or' buttons to add or remove conditions. The dialog box has a 'Basic Editor' tab, an 'Advanced Editor' tab, and 'Cancel' and 'Save' buttons. The background shows various form settings like Placeholder Text, Format Validation, Default Value, Number of Characters, Help Text, Show This Field, Require This Field, Read-Only, Limit Quantity, and Show Custom Error.

Revision #8

Created Fri, Oct 23, 2020 4:27 PM by [Leman](#)

Updated Wed, Jan 17, 2024 3:12 AM by [Leman](#)