

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, the 'Field Settings - Textbox' panel is visible with the following configuration:

- Label:** utm campaign
- Field Name:** Utmcampaign
- Type:** Single Line (selected)
- Placeholder Text:** test
- Format Validation:** (empty dropdown)
- Default Value:** (empty field)

On the right, a live preview of the form is shown. It includes a title 'test', a subtitle 'testing', an 'Email' field, and a 'utm campaign' field containing the text 'A test'. A context menu is open over the 'utm campaign' field, showing options: Cut, Copy, Delete, Insert Field, Make Smaller, Make Bigger, and Justify Row. Below the form fields is a 'Submit' button and a confirmation message: 'Thank you for filling out the form. Your response has been recorded.'

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 screenshot shows a form configuration interface with a modal dialog titled "Visible When...". The modal has a teal header and a white body. Inside the modal, there are two rows of conditions. The first row has three dropdown menus: "Email", "contains", and "123131312asdsad", followed by a "+" button and a trash icon. The second row is empty, with a "+" button and the word "or" in the center. At the bottom of the modal, there are two tabs: "Basic Editor" and "Advanced Editor", and two buttons: "Cancel" and "Save". The background shows a form configuration panel with various 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](#)