

SCRIPT FUNCTION

Form

- > Parameters
- > Responses
- > Script function

Script function

Allows you to display, for example, information or the result of a calculation in a form that depends on the entry of one or more other fields previously entered in the same form. For the use of the Script function, there are 3 parts :

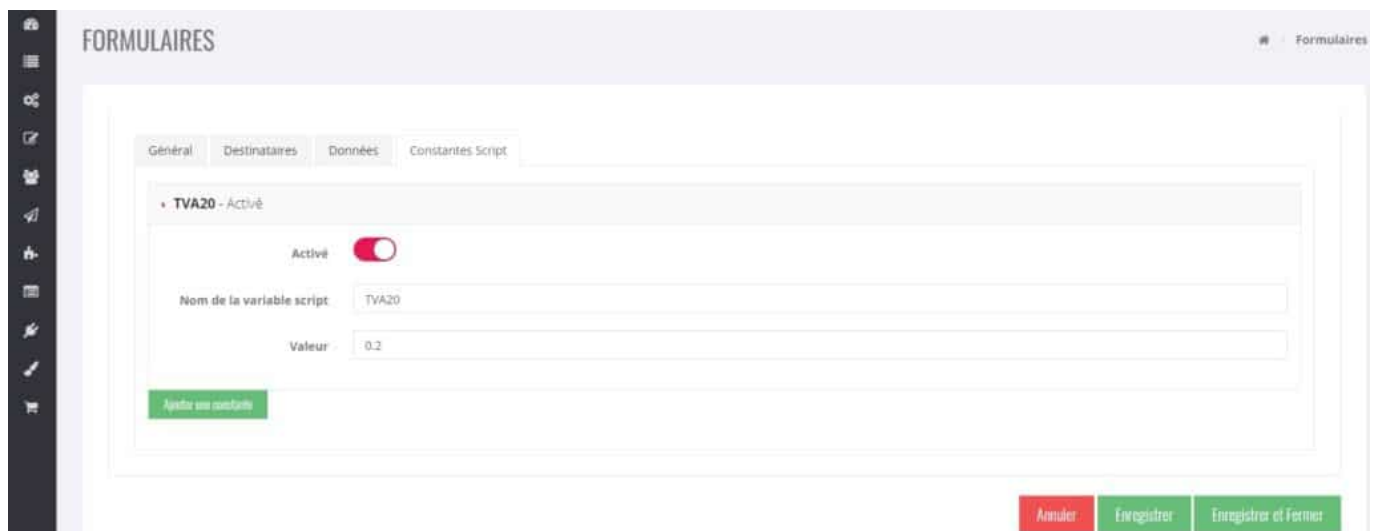
- The Script constants
- The Script variables
- The Script Results Fields

Script constants

They are used if necessary to declare constant values such as a VAT rate, a day of the week, a price per litre, a word etc. The advantage is that it is possible to change this value in a single place if necessary.

To do this: in the Form Settings, go to the “Script constants” tab and click on “Add a constant” :

- Check “Enabled” to be able to use it
- Enter his name (Ex: TVA20)
- Enter its value (e.g. for 20%, enter: 0.2).



NB: when we use this constant directly, it will be considered as text: even if in our example we have entered 0.2, it will not be considered as a numerical value but as a string of 3

characters “0.2”.

We will see later in the tips, how to convert a string into a numerical value and vice versa.

The Script variables

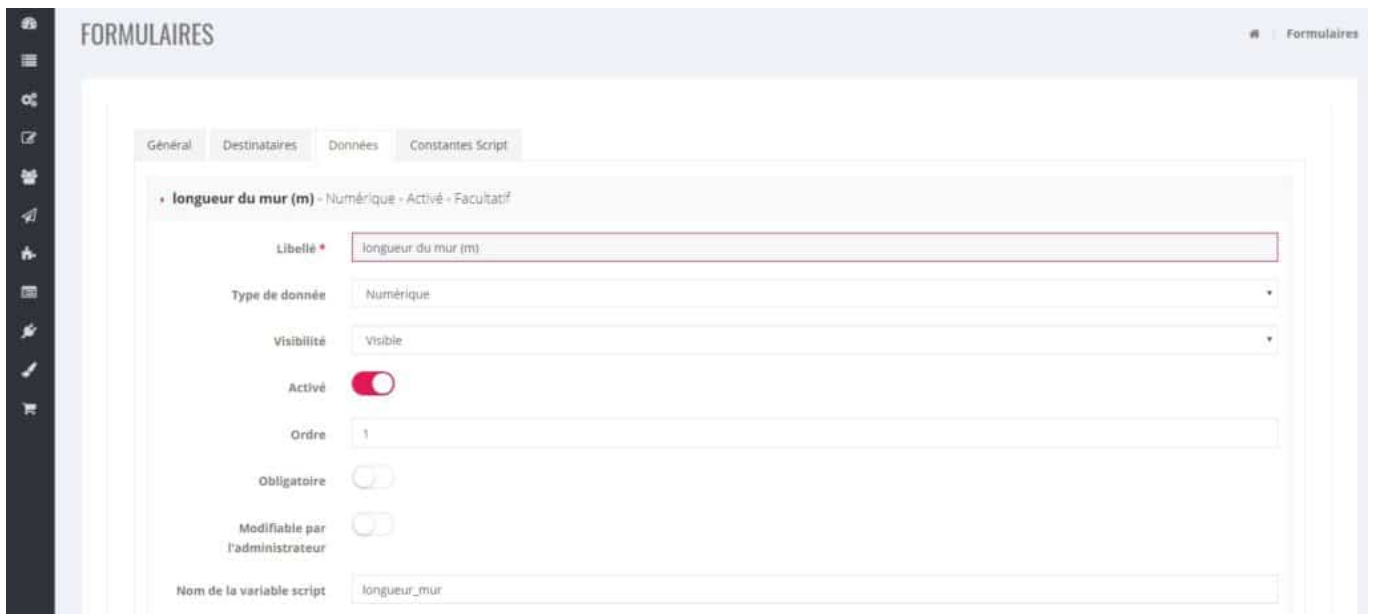
Script variables are used to retrieve the value entered in a form field for use in a “Script result” field.

It is necessary to define a variable name for each form field that will be used in scripts.

In the Form parameters, Data tab for each field, define the information “Script variable name”.

The variable name must not have “space” characters or accented characters.

For a good visibility we advise you to use “_” (Example for a field “ Wall length “, the variable name could be : wall_length).



The screenshot shows the 'FORMULAIRES' interface with the 'Données' tab selected. The field configuration is as follows:

- Libellé *: longueur du mur (m)
- Type de donnée: Numérique
- Visibilité: Visible
- Activé:
- Ordre: 1
- Obligatoire:
- Modifiable par l'administrateur:
- Nom de la variable script: longueur_mur

The Script Results Fields

This is a field used to enter a script and to display the result of that script. The user cannot enter any information in this field, he can only see the result.

To do this, in the form, go to the “Data” tab and click on “Add a field”, the important points are :

- The “Data Type” must be “Script Result”.
- The “Script”: The most important part, this is where you enter the script code.

The main rules for coding the script

- **Internal variables:** If needed, at the beginning of the coding, we can declare variables that will be used internally in the coding.
- **The “heart” of the :** This is the real part of the processing.

- **“returned” so displayed and visible to the user using the “return” instruction.**

Exemple

- Here’s an example, get the paint cost calculation for a wall:

We know the price HT per m² of a painting, and by entering the length and the height of the wall in meters we will display the amount of paint including VAT for this wall.

Solution:

- The price HT per m² is known so we define a “Constant Script” named px_ht_painting and enter its value (the dot “.” will be the decimal symbol).
- The VAT is known so we define a “Script Constant” named tva20 and its value will be: 0.2
- The length and height of the wall are information that will be entered by the user of the form, so we create 2 numeric fields whose “Name of the script variable” will be: length_wall and height_wall.
- Finally, a field of type “Script Result” is created in which the “Script” section will be :

```
wall_length = wall_length.replace(",","."); //in case a comma “,” is entered for the comma “.”.
```

```
wall_height = wall_height.replace(",","."); //in case a comma “,” is entered for the comma “.”.
```

```
paint_amount = parseFloat(wall_length)*parseFloat(wall_height)*parseFloat(px_ht_paint);  
//amount excl. tax of the paint cost
```

```
paint_amount = paint_amount*(1 + parseFloat(VAT20)); //amount of paint cost including VAT
```

```
return amount_painting.toFixed(2); //return the final result rounded to 2 decimal places
```

Translated with www.DeepL.com/Translator (free version)

Prix € TTC - Résultat Script - Activé - Obligatoire

Libellé *

Type de donnée

Visibilité

Activé

Ordre

Nom de la variable script

Script

```

Var montant_peinture = 0; // (variable interne qui sera le résultat final)
longueur_mur = longueur_mur.replace(",","."); // (au cas ou pour la virgule on aurait saisi une virgule « , »)
hauteur_mur = hauteur_mur.replace(",","."); // (au cas ou pour la virgule on aurait saisi une virgule « , »)
montant_peinture = parseFloat(longueur_mur)*parseFloat(hauteur_mur)*parseFloat(px_ht_peinture); // (montant HT du coût de la peinture)
montant_peinture = montant_peinture*(1 + TVA20); // (montant TTC du coût de la peinture)
return montant_peinture.toFixed(2); // (on retourne le résultat final arrondi à 2 décimales près)

```

Tips

- **Useful Javascript functions :**

- `parseFloat()`: convert a variable to a numeric value
- `parseInt()` : conversion of an integer variable
- `.toFixed(2)`: round off to 2 decimal places
- `.replace(",",".")`: replaces a comma with a period

- **Useful Links :**

- Langage Javascript : <https://developer.mozilla.org/fr/docs/Web/JavaScript>
- Javascript emulator to test javascript code: <http://www.webtoolkitonline.com/javascript-tester.html>