**Guide for creating a JS script for a formbuilder**

# Create or Modify a script for the form

## Necessary functions in every script

I’ve modularized each part of the scripts with well-structured functions.

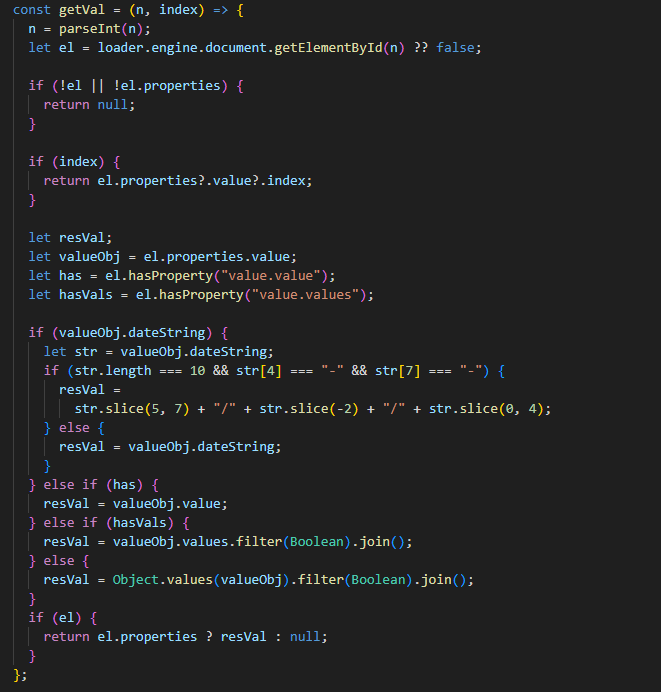
Here are the essential functions in the script: getVal(), strReplace(), and idsToNoteLines().

* getVal()
  + Parameters
    - n (Number): ID of the field
    - index (Optional): The index of the field, whether it’s a combobox or a select element.
  + Output (String)

The value of the field.

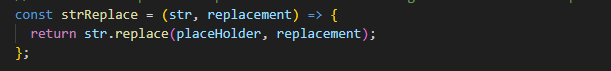
* + Description

A function that returns the selected or inputed value of a field based on the provided field ID.



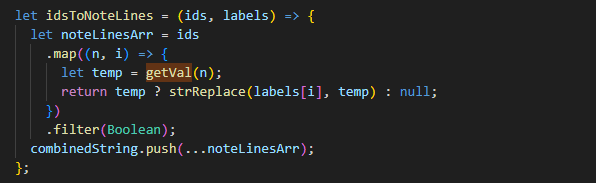
* strReplace()
  + Parameters
    - str (String): field label.
    - replacement (String): field value
  + Output (String)
  + Description

A function that combines a field’s description and its vaue to be printed in the PIMS NOTE.



* idsToNoteLines()
  + Parameters
    - ids (Array): field ids
    - labels (Array): field labels
  + Description

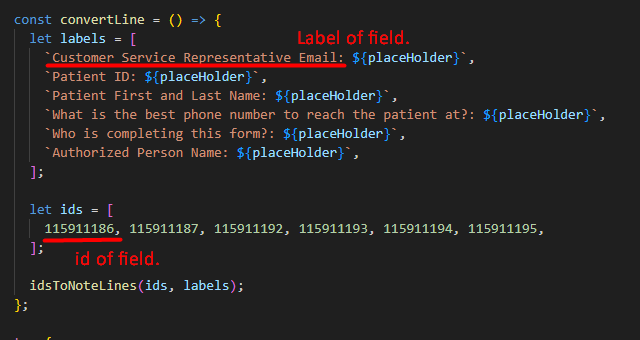
This function combines the descriptions and corresponding values for a list of fields to be included in a PIMS NOTE.



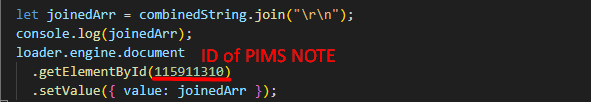
## Custom functions in the script (Needs to be added for each script)

We need to create two arrays: one for the labels and another for the field IDs for each field.

Then we’ll pass these arrays as parameters to the idsToNoteLines function.



Finally, we need to add the parsed result to the PIMS NOTE.



## Additional Features

* Highlight the subtext of PIMS NOTE



PIMS NOTE uses the HTML textarea tag.

However, we can’t change the color or font size of the subtext within that element.

To overcome this limitation, we’ve created a customized virtual textarea using a div tag, which allows us to manually adjust the subtext as needed.

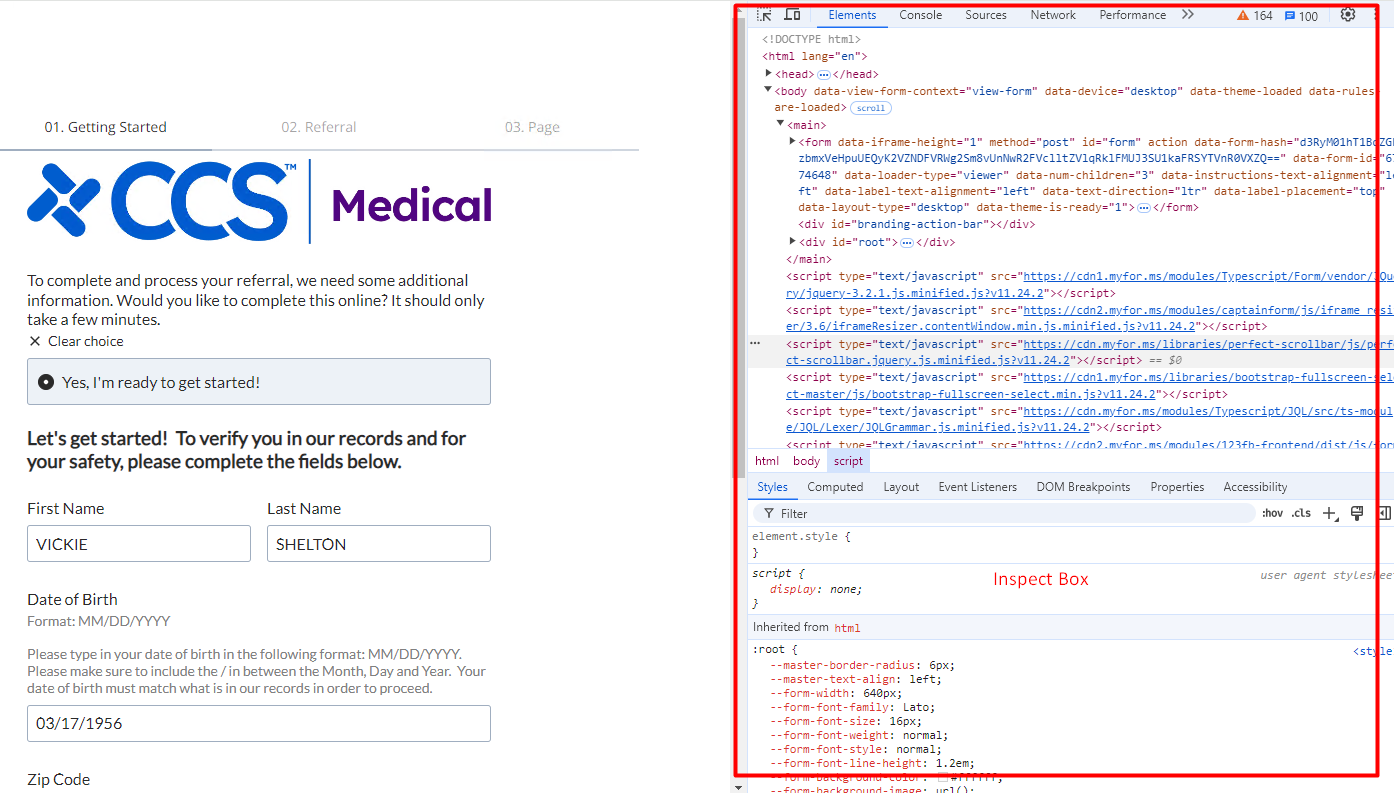
In the future, if similar needs arise, we’ll need to further develop this function to provide greater flexibility.

# Test the script during development

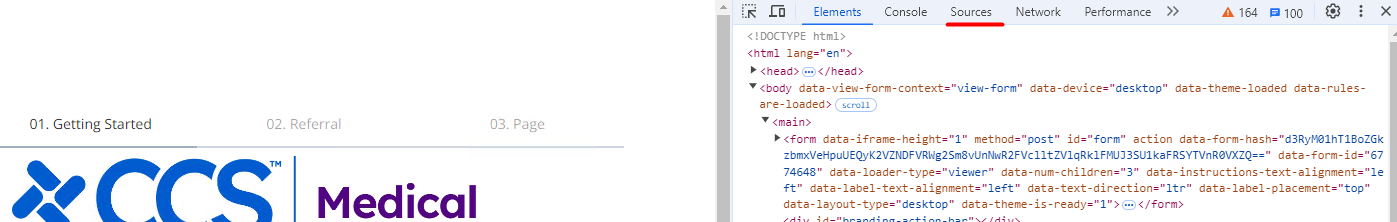
Testing is a crucial part of ensuring our script works effectively within the form.

## 2.1 Replace source code of script with original script in form

* Press **F12** to display Inspect Box

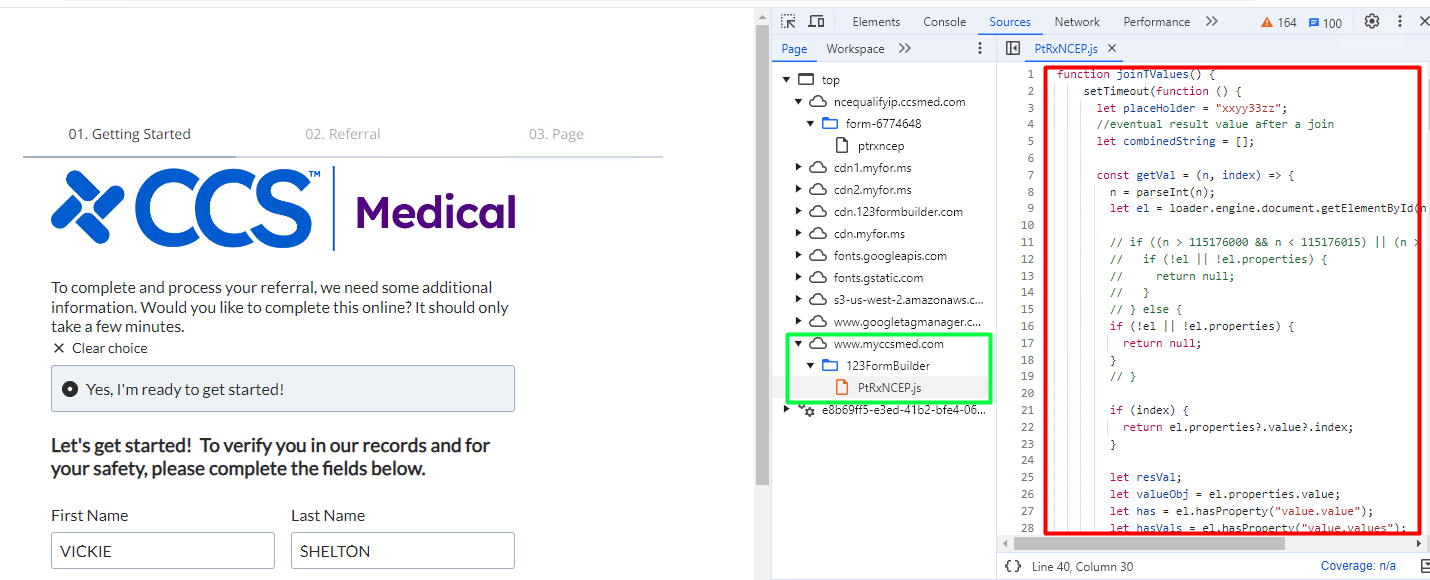


* Click the **Source** tab.



* Locate [www.myccsmed.com](http://www.myccsmed.com) in the sidebar.

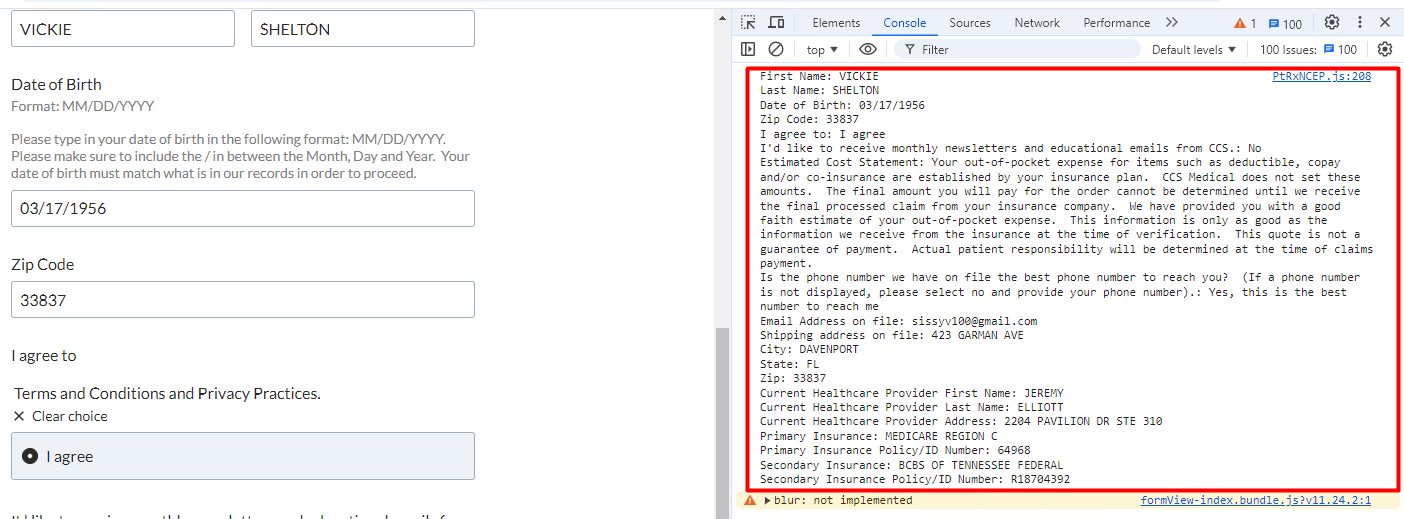
Inside the **123FormBuilder** folder, you’ll find a .js file; click on it to view the source code.



* Replace the source code you created for the form.

## 2.2 Test the PIMS NOTE with the console log.

* The log will appear in the console, allowing you to check it whenever you enter or modify a field.



# Time needed to bring the script to life

* Create a new script for a form
  + Work Time

4 ~ 6 hours

* Modify an exisiting script to add or remove fields
  + Work Time

1 ~ 4 hours (Depending on the number of fields)

* Implement the complicate logic (Like DocVal Form)
  + Work Time

2 ~ 4 hours

* Implement additional features
  + Make text bold in PIMS Note.
    - Work Time

4 hours

* + Fill the date after selecting the previous date.
    - Work Time

1 hour

* + Select the radio button automatically
    - Work Time

1 ~ 2 hours.