

Frevvo Advanced: Make Form Field Visible Based On Checkbox Selection

Table of Contents

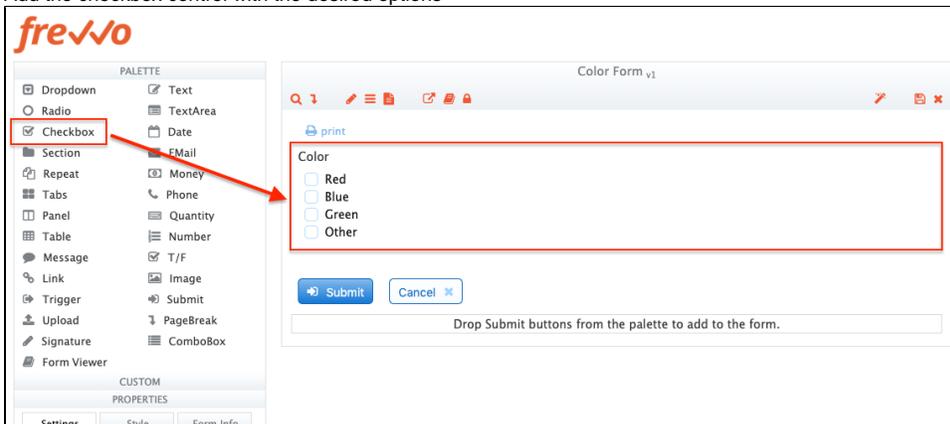
- [Overview](#)
- [Process](#)

Overview

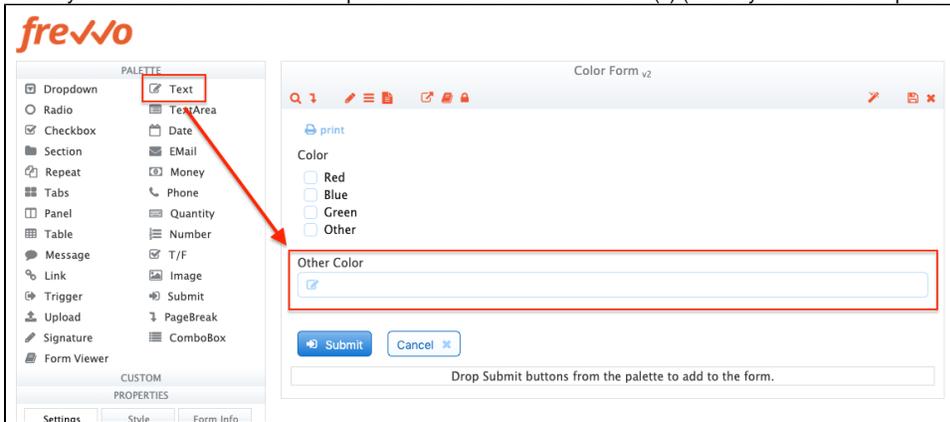
There may be some situations in which we want a certain field to only show when a certain checkbox is selected within a form, then disappear if unchecked. Some field properties are modified and a rule is put in place to achieve this.

Process

1. Add the checkbox control with the desired options

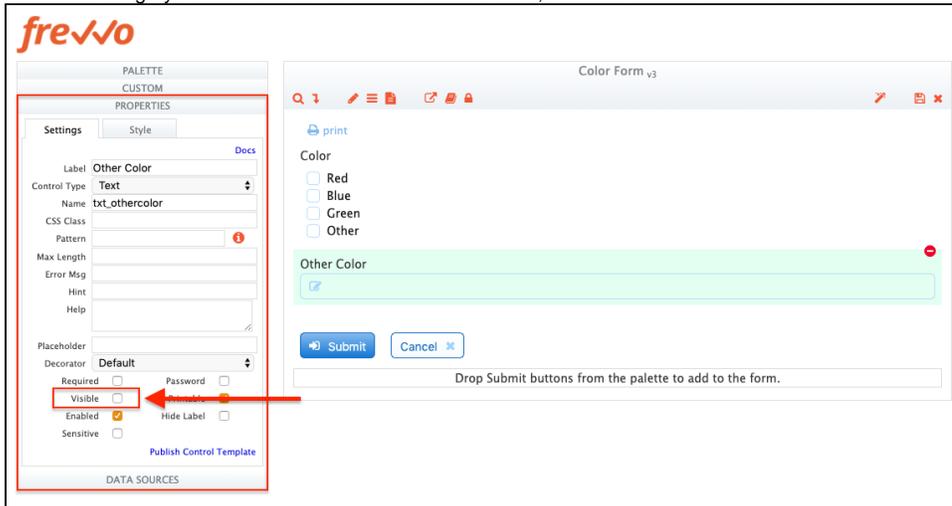


2. Add any additional fields that will be dependent on the checkbox selection(s) (will only show when a specific checkbox is checked)

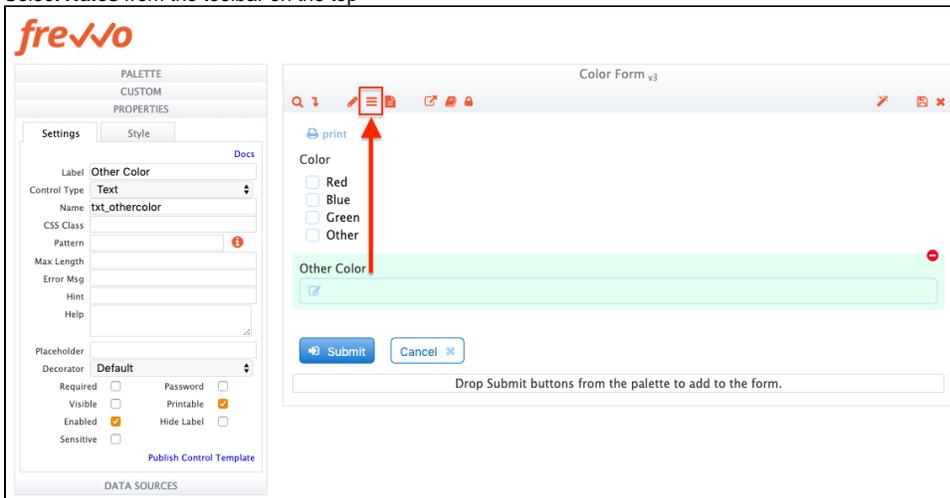


3. In the field properties section for each of the fields that will be shown/hidden based on selection, uncheck the **Visible** option

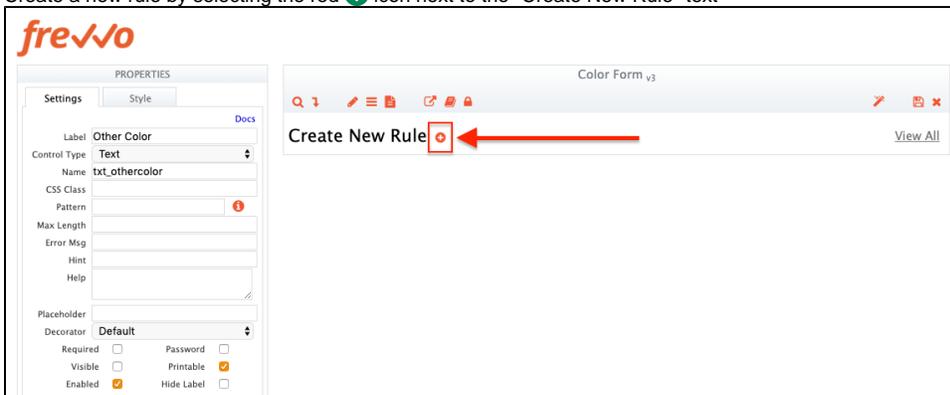
- a. This will make gray out the area around the field in the editor, and make it hidden on the actual form



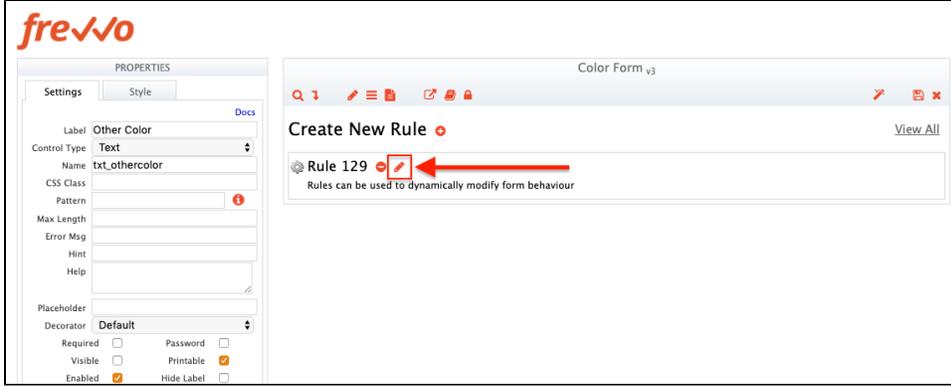
4. All form designer controls have been placed and adjusted, now to move onto the form rules
5. Select **Rules** from the toolbar on the top



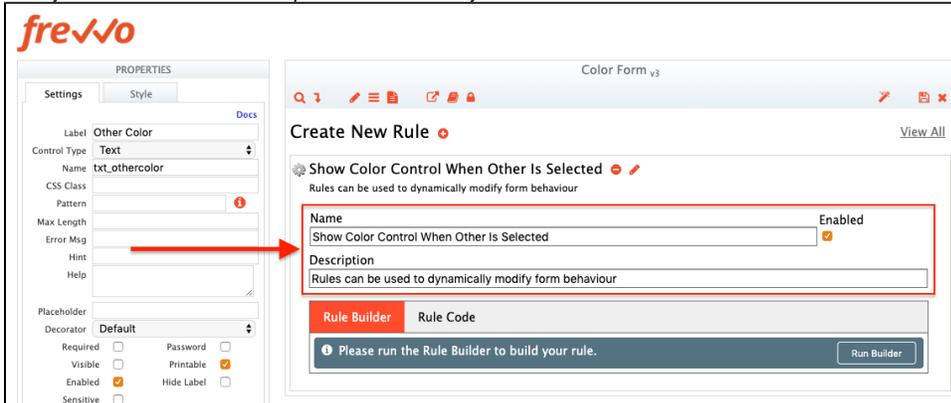
6. Create a new rule by selecting the red + icon next to the "Create New Rule" text



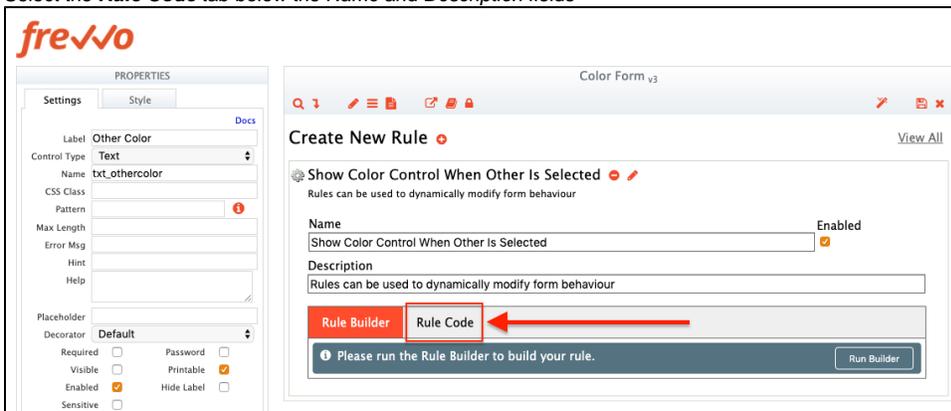
7. When selected, a new rule will appear below. Select the pencil icon () next to the new rule to edit it.



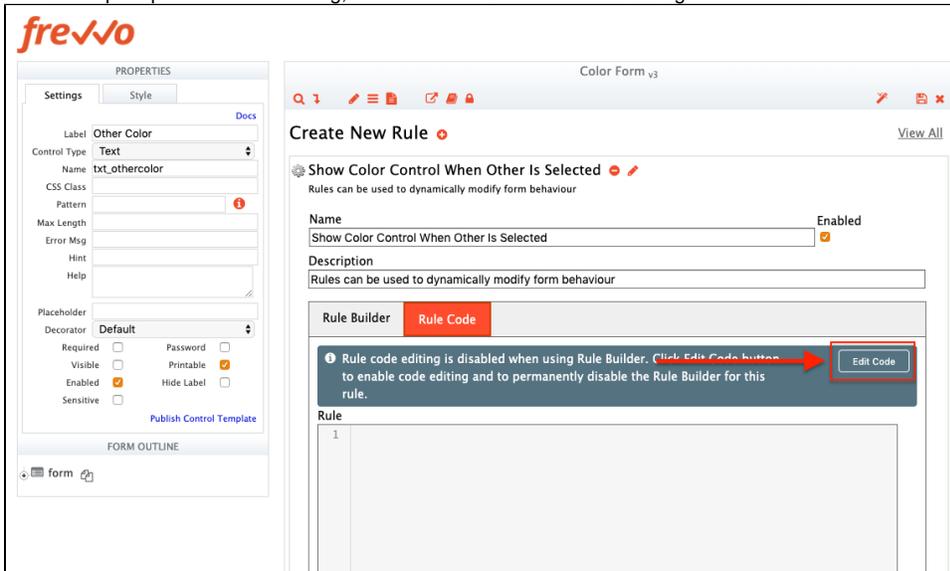
8. Modify the rule **Name** and **Description** fields to convey it's function



9. Select the **Rule Code** tab below the **Name** and **Description** fields



10. You will be prompted to enable editing, select the **Edit Code** button on the right to continue



11. Paste the following code in the rule code text area:

```
// Only copy and use this rule for checkbox fields!
var found = false;
for (var i = 0; i < cbx_color.value.length; i++) {
    // Set which selection in the checkboxes will
    // show/hide the field.
    if (cbx_color[i].value === 'Other') {
        found = true;
        break;
    }
}
if (found === true) {
    // Change the part before the period on each of the following
    // to the field you want to show/hide based on checkbox
    // selection.
    txt_othercolor.visible = true;
    txt_othercolor.required = true;
} else {
    txt_othercolor.visible = false;
    txt_othercolor.required = false;
}
```

Multi-Checkbox Field Display

```
// Only copy and use this rule for checkbox fields!
var found_opt1 = false;
var found_opt2 = false;

for (var i = 0; i < checkbox.value.length; i++) {
    // Set which selection in the checkboxes will
    // show/hide the field.
    if (checkbox[i].value === 'Option_1') {
        found_opt1 = true;
    } else if (checkbox[i].value === 'Option_2') {
        found_opt2 = true;
    }
}
if (found_opt1 === true) {
    // Change the part before the period on each of the following
    // to the field you want to show/hide based on checkbox
    // selection.
    section1.visible = true;
    section1.required = true;
}
```

```

} else {
    section1.visible = false;
    section1.required = false;
}
if (found_opt2 === true) {
    // Change the part before the period on each of the following
    // to the field you want to show/hide based on checkbox
    // selection.
    section2.visible = true;
    section2.required = true;
} else {
    section2.visible = false;
    section2.required = false;
}
}

```

The screenshot shows the frevo IDE interface. On the left, the 'PROPERTIES' panel shows settings for a control named 'txt_othercolor'. The 'FORM OUTLINE' panel shows a tree view of the form elements, including 'cbx_color' and 'txt_othercolor'. A red arrow points from 'txt_othercolor' in the outline to the 'Rule Code' editor. The 'Rule Code' editor shows the following code:

```

1 // Only copy and use this rule for checkbox fields!
2 var found = false;
3 for (var i = 0; i < cbx_color.value.length; i++) {
4     // Set which selection in the checkboxes will
5     // show/hide the field.
6     if (cbx_color[i].value === 'Other') {
7         found = true;
8         break;
9     }
10 }
11 if (found === true) {
12     // Change the part before the period on each of the following
13     // to the field you want to show/hide based on checkbox
14     // selection.
15     txt_othercolor.visible = true;
16     txt_othercolor.required = true;
17 } else {
18     txt_othercolor.visible = false;
19     txt_othercolor.required = false;
20 }

```

12. In the pasted code, modify the following:

- Change where it says "cbx_color" (lines 3, 6) to the *Name* value of your checkbox control
- Change where it says "txt_othercolor" (lines 15, 16, 18, 19) to the *Name* value of your field you want to make visible
- Change where it says "'Other'" (line 6) to the value of the checkbox option you want to be checked to make the field visible

The screenshot shows the 'Rule Code' editor with the following modified code:

```

1 // Only copy and use this rule for checkbox fields!
2 var found = false;
3 for (var i = 0; i < cbx_color.value.length; i++) {
4     // Set which selection in the checkboxes will
5     // show/hide the field.
6     if (cbx_color[i].value === "Other") {
7         found = true;
8         break;
9     }
10 }
11 if (found === true) {
12     // Change the part before the period on each of the following
13     // to the field you want to show/hide based on checkbox
14     // selection.
15     txt_othercolor.visible = true;
16     txt_othercolor.required = true;
17 } else {
18     txt_othercolor.visible = false;
19     txt_othercolor.required = false;
20 }

```

- When you added the checkbox control, you should've modified the values available as well. The values are displayed in a "Option=Name" format, each on it's own line. Make sure you change where it says "Other" to the checkbox's "Option" property and not it's name! Also be sure to include the single quotes.



If you forget what any of the parameters or names were, you can expand each item in the **Form Outline** to get a view of what exists!

The screenshot shows the frevo software interface. On the left is the 'PROPERTIES' panel for a control named 'txt_othercolor'. It includes fields for Label, Control Type, Name, CSS Class, Pattern, Max Length, Error Msg, Hint, Help, Placeholder, and Decorator. There are also checkboxes for Required, Visible, Enabled, Sensitive, Password, Printable, and Hide Label. A red arrow points to the 'Publish Control Template' button. Below the properties is the 'FORM OUTLINE' tree view, which is expanded to show the 'txt_othercolor' control. On the right is the 'Create New Rule' dialog. The rule name is 'Show Color Control When Other Is Selected'. The 'Rule Code' tab is active, showing a JavaScript rule that iterates through the 'cbx_color' array and sets the visibility and required status of 'txt_othercolor' based on the selected color.

13. Select the Save Icon (📁) from the top-right

This screenshot is similar to the previous one but highlights the 'Save' icon (a floppy disk) in the top-right corner of the 'Create New Rule' dialog window. A red arrow points to this icon. The 'Enabled' checkbox for the rule is also checked. The rule code is visible in the 'Rule Code' tab.

14. Test the form by selecting the designated checkbox that makes the other control appear