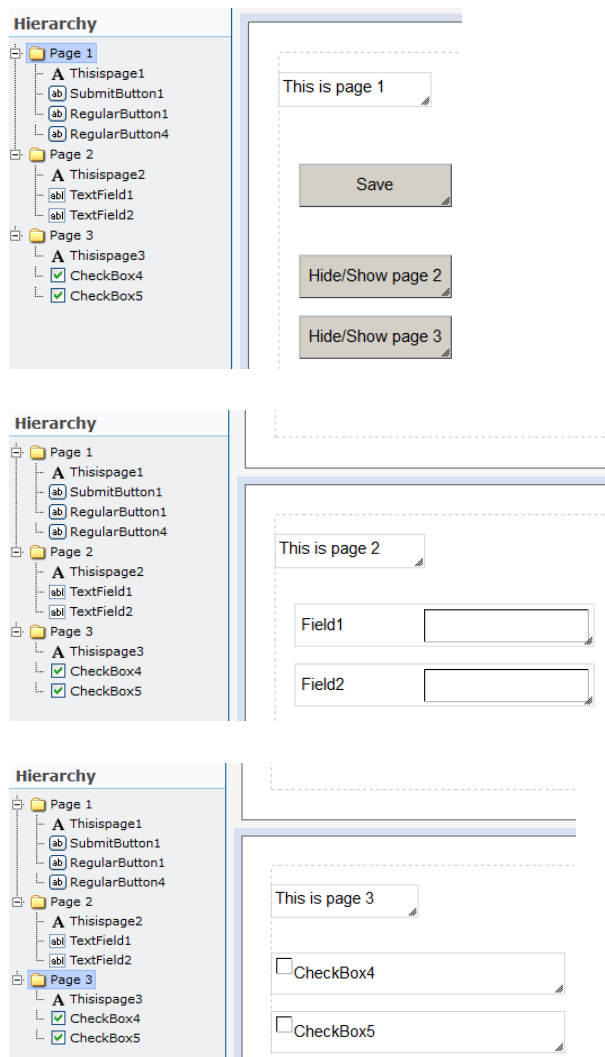


## Showing and hiding document pages using Client-side Scripting

This guide describes the steps to show and hide pages in PDF document using JavaScript in Button action.

### Step 1. Prepare a template.

As an example, we use Page 1, Page 2 and Page 3:



The image displays three sequential screenshots of a PDF form's 'Hierarchy' and content area. In the first screenshot, the hierarchy shows Page 1 selected, and the content area contains a text field 'This is page 1', a 'Save' button, and two buttons labeled 'Hide/Show page 2' and 'Hide/Show page 3'. The second screenshot shows Page 2 selected, with the content area displaying 'This is page 2' and two text fields, 'Field1' and 'Field2'. The third screenshot shows Page 3 selected, with the content area displaying 'This is page 3' and two checkboxes, 'CheckBox4' and 'CheckBox5'.

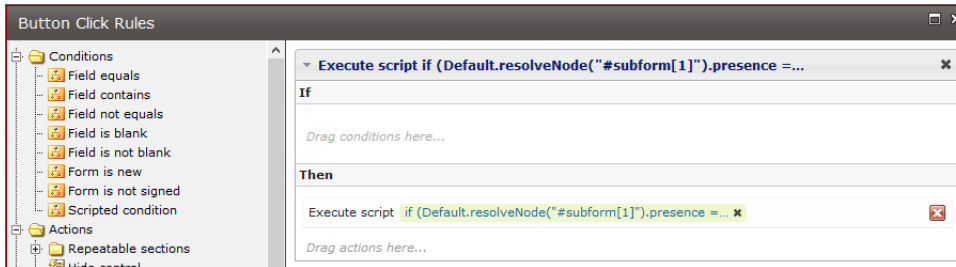
### NOTE:

Each page has a label with a page number.

Page 1 will act as a control page. It has two Regular Buttons responsible for showing and hiding Page 2 and Page 3.

Page 2 and Page 3 have Text Fields and Check Boxes. These controls are placed only for demonstration purposes.

**Step 2.** Add a script to “**RegularButton1**” (labeled “**Hide/Show page 2**”) actions:  
**PDF Form Tools** → **Properties** → **Actions**.



```
if (Default.resolveNode("#subform[1]").presence == "hidden") {
    Default.resolveNode("#subform[1]").presence="visible";
} else {
    Default.resolveNode("#subform[1]").presence="hidden"
}
```

This code will show a page if it is hidden already, or hide it if it is currently visible.

**NOTE:**

In order to get a reference to pages, we need to go through view name: `viewName.resolveNode("#subform[n]")`, where "#subform[n]" is Page 1.

Every view has its own pages set starting from "#subform[0]", which is Page 1 name.

In our example there are three pages: "#subform[0]", "#subform[1]" and "#subform[2]".

**Step 3.** Add a script to “**RegularButton2**” (labeled “**Hide/Show page 3**”) actions:  
**PDF Form Tools** → **Properties** → **Actions**.

```
if (Default.resolveNode("#subform[2]").presence == "hidden") {
    Default.resolveNode("#subform[2]").presence="visible";
} else {
    Default.resolveNode("#subform[2]").presence="hidden"
}
```

This script is similar to the previous one except page numbers.

**NOTE:**

Pages set to “visible” by default.

If you want to keep a page hidden until some event, you need to add a script to “On Form Ready Script” section that will do so until required conditions are acquired.

Hiding page will not erase data from its fields.