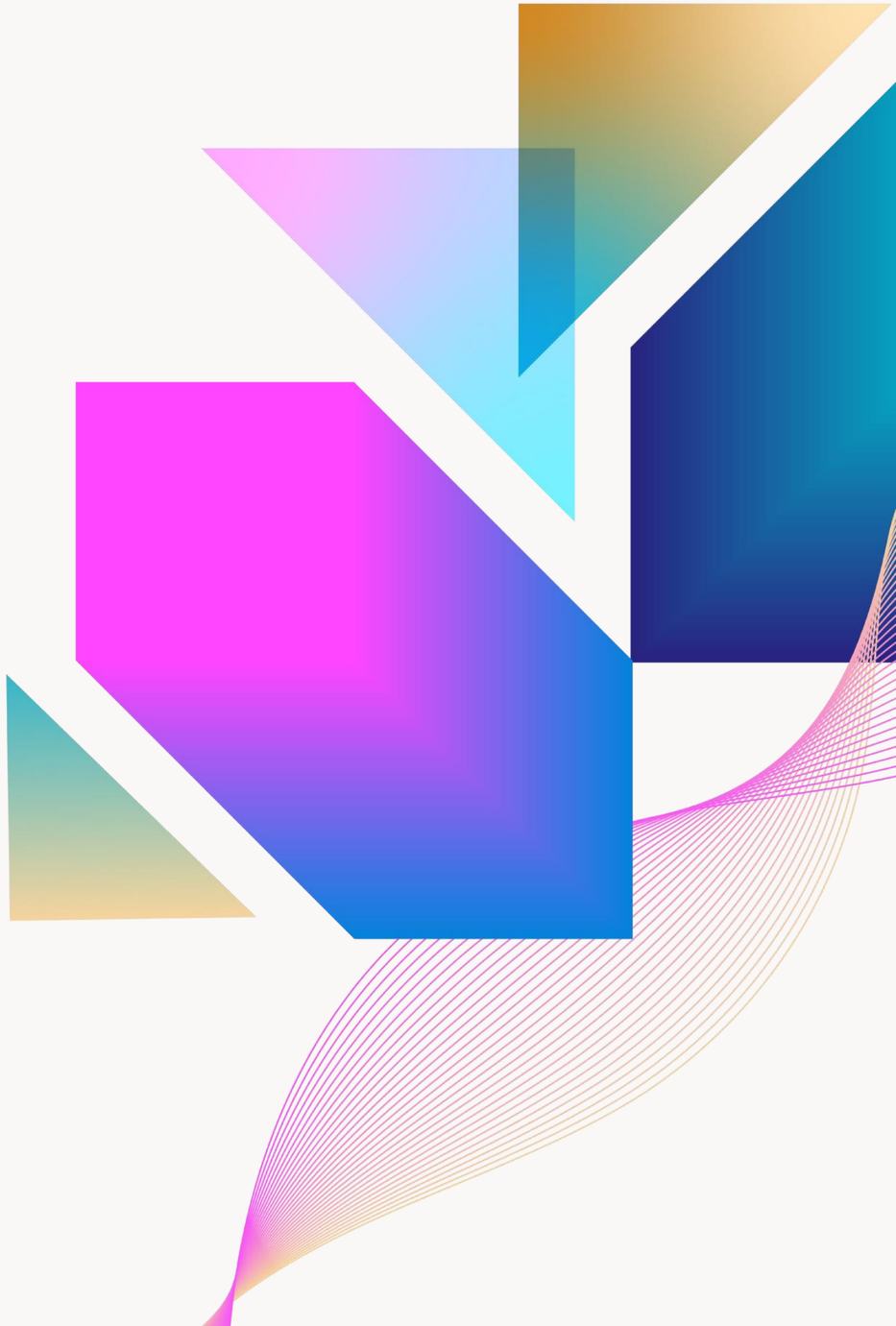
UNBOUND

Intro to JavaScript in Revu

Practical Interactivity and How AI Can Help

Glynis DeMone
Liz Larsen



A Tale of Two Coders

© 2025 Bluebeam

Agenda

- What is JavaScript
- Why use JavaScript in Bluebeam Revu?
- Where to Add JavaScript in Revu
- Live demos: Creating interactive buttons with code and AI tools
- Techniques for Debugging Code
- Next steps, Resources, and Q&A

What is JavaScript?

JavaScript is a lightweight, flexible coding language.

Transforms something static into something that responds

- Add buttons
- Fill out fields automatically
- Respond to clicks

Learning to code is like learning a new language.

- Has its own vocabulary and grammar
- Practice builds fluency

Use AI as your translator

- You don’t have to be fluent to get some simple interactivity in Revu
- You describe what you want, and AI helps your write the code

© 2025 Bluebeam

UNBOUND

Why use JavaScript?

JavaScript adds “smarts” to your PDFs — buttons, automation, and shortcuts.

Examples you could build:

- **Field reports made easier**
  - Add a button that lets you insert a photo directly into the report.
- **Reusing content**
  - Import pages from another PDF with one click (e.g., safety sheets, cover pages).
- **Smarter form fields**
  - Auto-fill names, dates, or job numbers.
  - Update all related fields at once instead of typing the same info repeatedly.
- **Bookmarks that keep up**
  - Create or update bookmarks automatically when documents change.
- **Consistent approvals**
  - Add approval stamps that calculate totals or fill in project info automatically.

© 2025 Bluebeam

UNBOUND

Resources

Important ***FREE*** resources to get you started:

- Bluebeam JavaScript documentation
  - https://support.bluebeam.com/developer/javascript/javascript.html
- PDFscripting.com – lots of examples, some of them may be outdated
- Sublime Text or Notepad++
  - Good free text editors that highlight your code to make it easier to read and format
- FreeCodeCamp.org – learn JavaScript
  - Course name: *JavaScript Algorithms and Data Structures*
    - Necessary: Basic JavaScript, Basic Algorithm Scripting
    - Optional: Regular Expressions, Intermediate Algorithm Scripting
- JavaScript Syntax Checker - jshint.com
- [Optional] Regular expression checker - regex101.com
- [Optional] JavaScript beautifier - beautifier.io

© 2025 Bluebeam

Regular Expressions (regex)

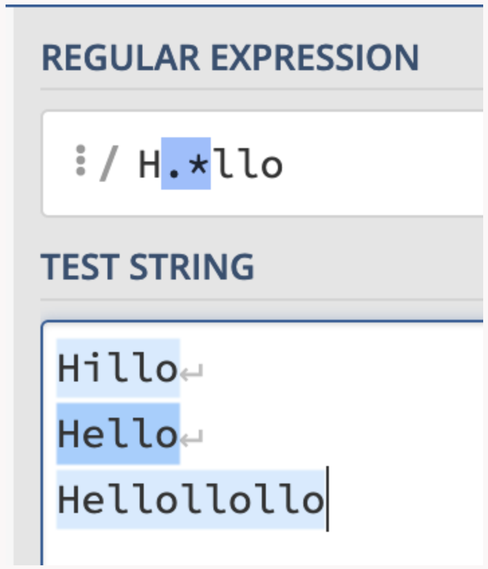
The basics (for more attend Advanced JavaScript!)

- Regular expressions are just a formula for pattern recognition
  - Think of it as a **search filter on steroids**: instead of only looking for exact matches, regex lets you describe rules like “starts with a number,” “contains a dash,” or “looks like an email address.”
- In JavaScript inside Bluebeam, form fields often have systematic names:

  `Job_1, Job_2, …`

  `CostCode_1, CostCode_2, …`

  `Amount_1, Amount_2, …`
- If you want to grab all the Amount fields without typing them one by one, regex makes it one line.



© 2025 Bluebeam

UNBOUND

Writing JavaScript in the Console

The console is useful for trying out individual lines of code to see how they work

- Show JavaScript Console with keyboard shortcut **CTRL+J**.
- Run the code by hitting **CTRL+ENTER**.
- Code is applied immediately.
- Code is **not** saved within the document.
- Example: how to change the fill color
  - `this.getField(“Button1").fillColor = color.red;`
- Other properties you can change or lookup:
  - `textColor, textSize, textFont, type, value, editable`
  - See documentation for additional information

© 2025 Bluebeam

Code Level

Global vs local

Document Level (Global)

- Executes when the document is opened or when called by local code
- Can pull external data into dropdown boxes
- Can change or manipulate other form fields

Object Level (Local)

- Attached to a specific form field
- Trigger is specified by the user in the form field settings (mouse up, mouse enter, etc…)
- Can change or manipulate other form fields

© 2025 Bluebeam

UNBOUND

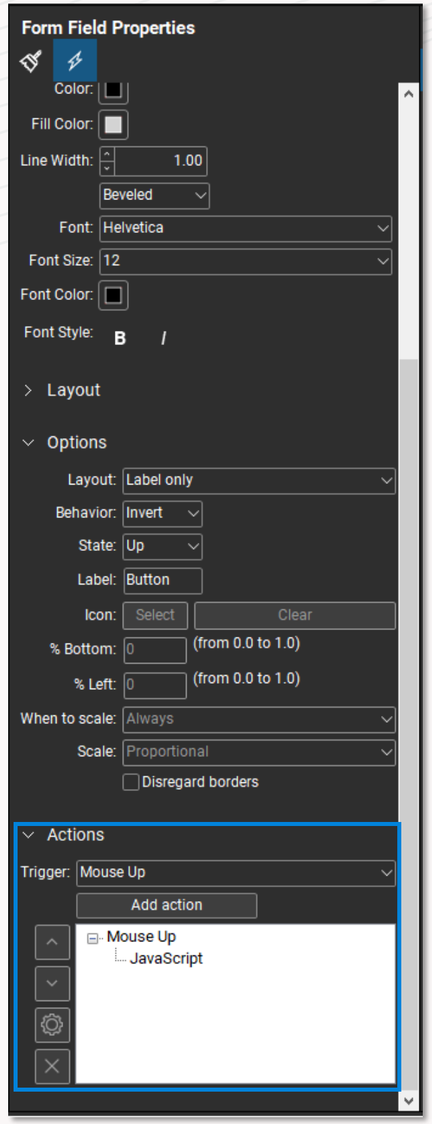
Form Field Basics

JavaScript's most common usage within Bluebeam is to modify form fields

Types of form fields:

- Text Box
- Radio Button
- Check Box
- List Box
- Dropdown
- Button
- Digital Signature

You can attach JavaScript to form fields within the Actions section of the field properties



© 2025 Bluebeam

UNBOUND

Form Field Basics, con’t

JavaScript’s most common usage within Bluebeam is to modify form fields

- Give the form fields thoughtful names.
    - It keeps your code organized.
    - It helps *Future You* when you want to use regular expressions.
- Think about what you want your code to do, then name the form fields in a way that makes sense.
- For example:
    - You want to call your Submit button something that makes sense instead of leaving it the default named “Button1”.

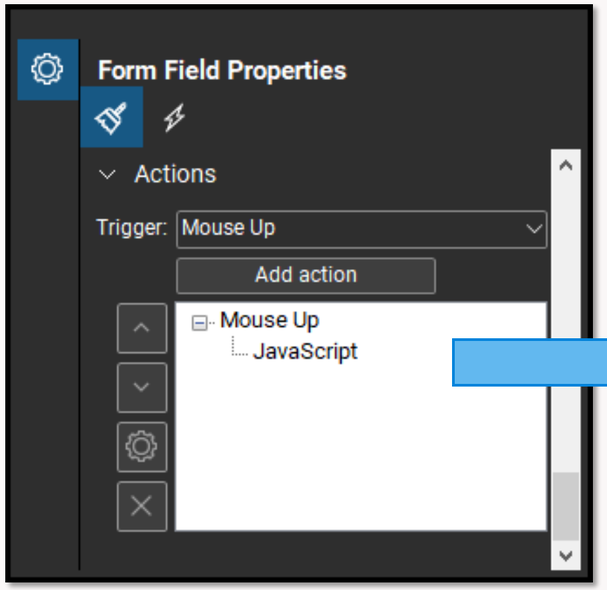
```
//add a tooltip on hover-over
var f = this.getField("SubmitButton");
f.userName = "Press this button to submit your data.";
```

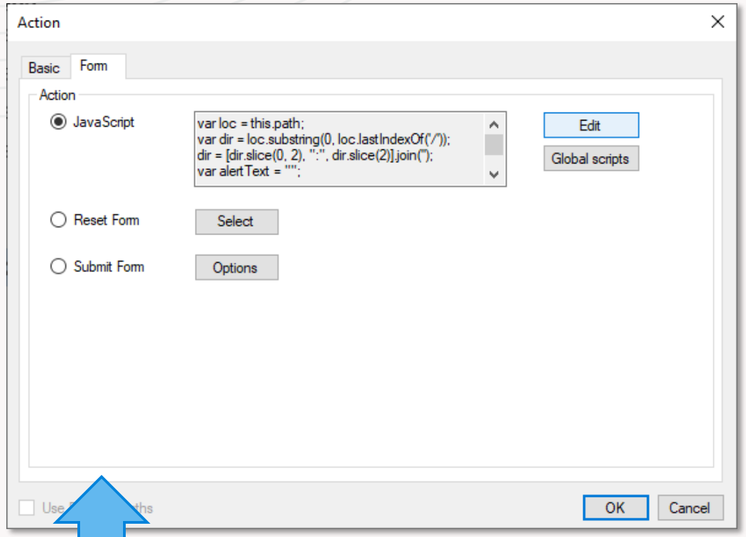
© 2025 Bluebeam

Attaching Code to Form Fields

aka creating the object level (local) code

1. Go into the **Forms** sidebar, then select the form field.
2. Then go into the **Properties** sidebar. Scroll down to **Action**.
3. Select the trigger from the dropdown, mostly likely “mouse up”.
4. Click **Add Action**.
5. Go to the **Form** tab and click **Edit**.





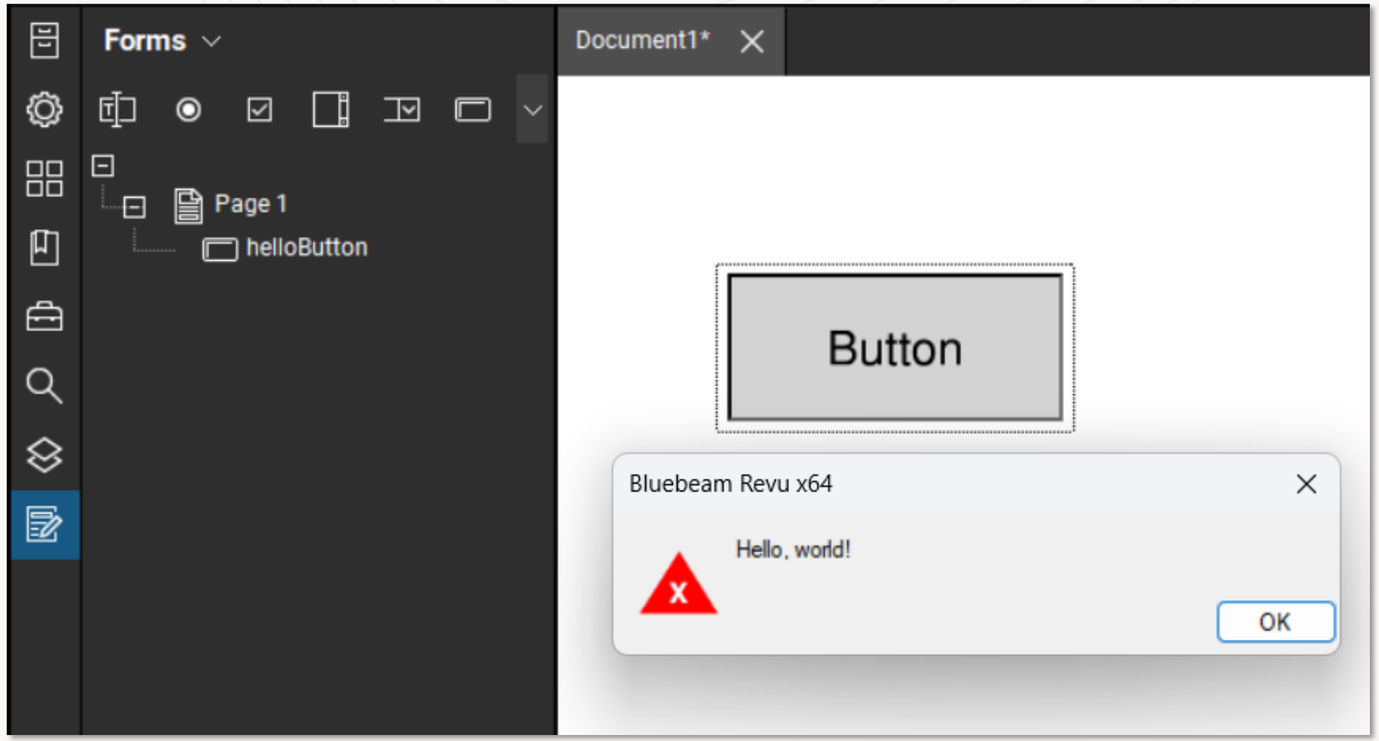
© 2025 Bluebeam

UNBOUND

Example 1 – A simple button

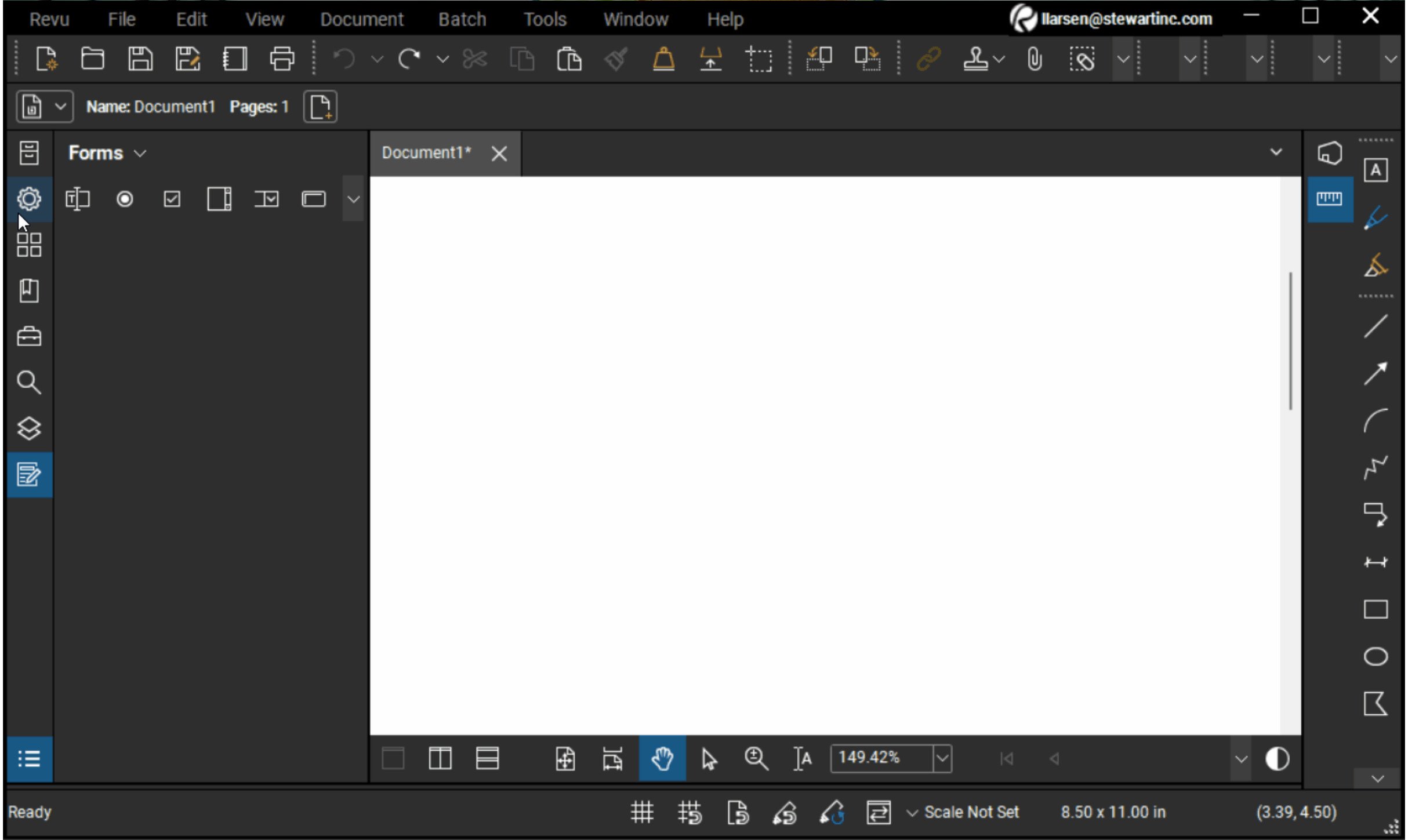
Hello, World!

1. Create a button.
2. Give it a name
3. Go into the button **Properties**
4. Scroll to the bottom of **Properties** and click **Add Action**
5. Under the **Forms** tab, click **Edit**
6. Add your JavaScript and click **OK**



© 2025 Bluebeam

UNBOUND

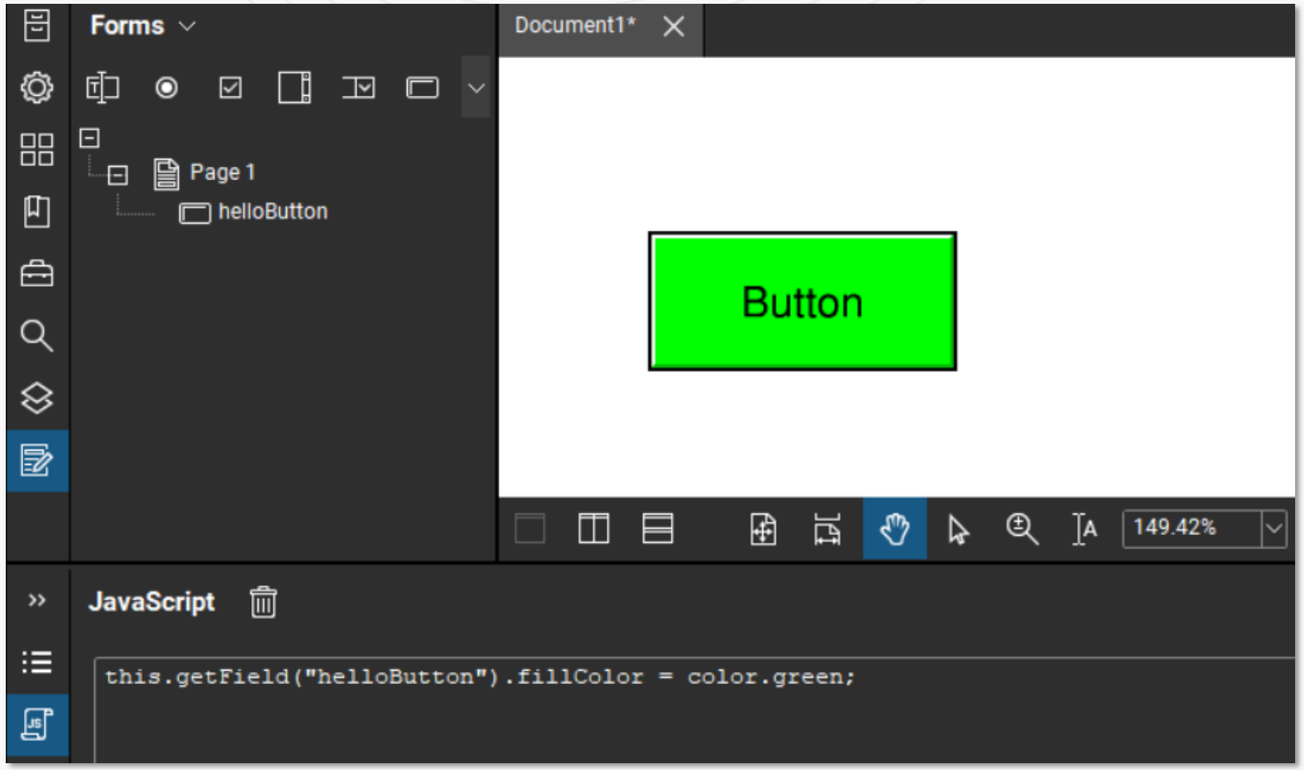

Revu
File
Edit
View
Document
Batch
Tools
Window
Help
llarsen@stewartinc.com
Name: Document1
Pages: 1
Forms
Document1*
149.42%
Ready
Scale Not Set
8.50 x 11.00 in
(3.39, 4.50)

Example 2 – Using the console

Use the console to create quick changes that take affect immediately

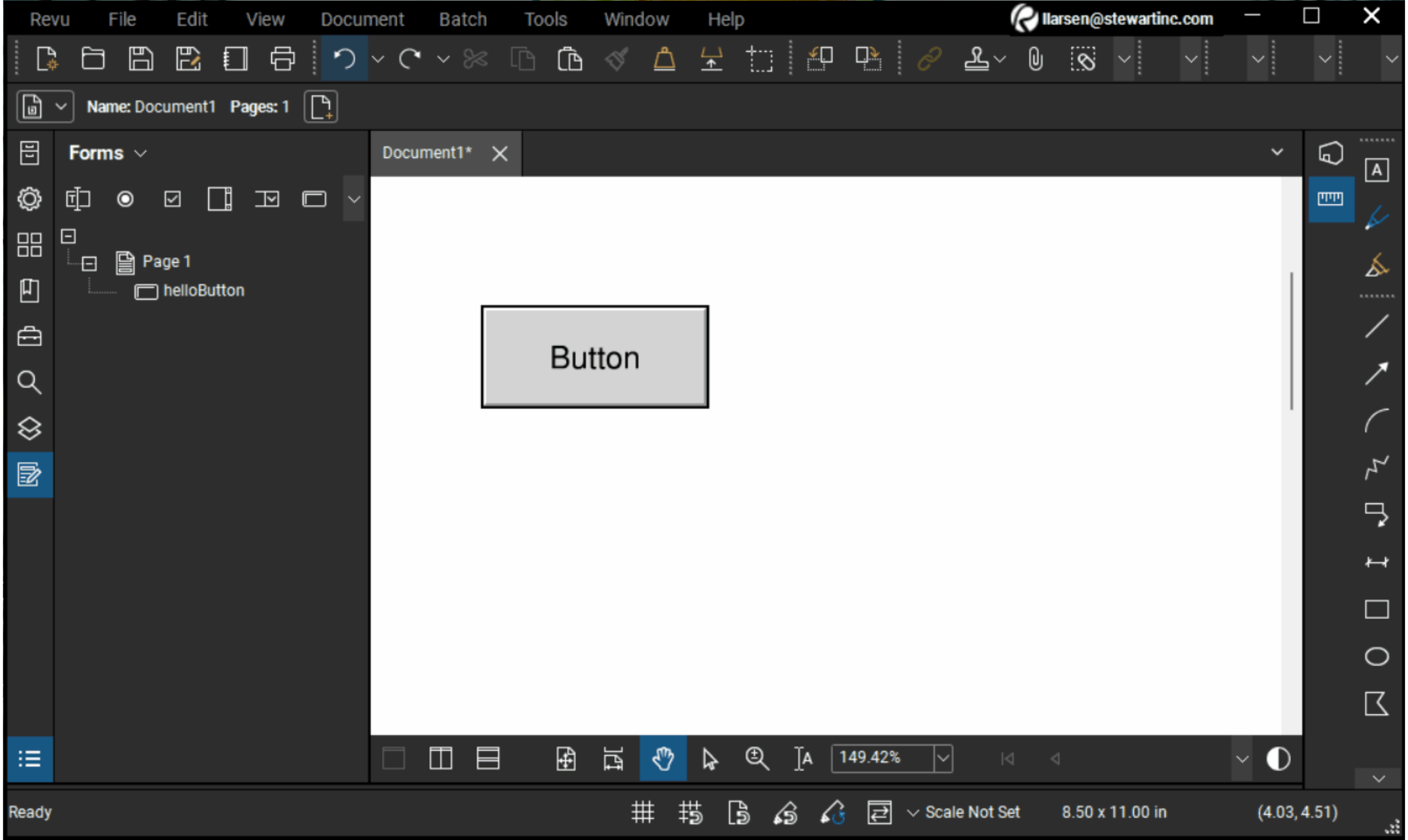
1. Open the console using **ALT+J**
2. Type any code you want, including multiline code
3. Type **CTRL+ENTER** to run the code

The documentation will tell you what is possible and how to do it.



© 2025 Bluebeam

UNBOUND


Revu
File
Edit
View
Document
Batch
Tools
Window
Help
llarsen@stewartinc.com
Name: Document1 Pages: 1
Forms
Page 1
helloButton
Document1*
Button
149.42%
Ready
Scale Not Set
8.50 x 11.00 in
(4.03, 4.51)

Coding with AI

© 2025 Bluebeam

UNBOUND

What is a prompt?

An AI prompt is a set of instructions or a question that you give to an AI tool to get a specific response.

If you’re using AI to code, you can use a prompt to ask the AI to help you write code, explain something, or solve a problem.

The clearer your prompt is, the better the AI can understand what you need. Be specific and include helpful details.

© 2025 Bluebeam

UNBOUND

AI Prompt Generation

Pro Tips and Tricks

C	Character: Tell the AI who to be	Act as a JavaScript expert with experience scripting interactive forms in Bluebeam Revu.
O	Objective: Be specific about what you want to happen	Objective: Be specific about what you want to happenI want a button that fills in today's date and a due date 10 days out
D	Define Tools: Be specific about tool and platforms you are using	Define Tools: Be specific about tool and platforms you are usingI am using Bluebeam Revu 21 Complete not Adobe Acrobat
E	Explain details. Give context such as skill level, limits or special requirements	I'm new to JavaScript and Revu forms. Walk me through field creation, button setup, and include Revu-specific tips.
R	Request Questions: Invite AI to ask clarifying questions	Ask follow up questions if there is any additional information you need before you generate the code.
S	Source: provide any resources you would like the AI to reference	Please reference this guide: https://support.bluebeam.com/developer/javascript/javascript.html

© 2025 Bluebeam

AI Live Build

© 2025 Bluebeam

UNBOUND

Debugging

A tale of two methods

There are two good options for debugging. Try both and see which works best for you.

1. Alert pop-up (`app.alert`)
    - Best for checking single values quickly.
    - Pauses execution until the box is closed.

```
var alertText = somevar;
app.alert(alertText);
```

2. Console output (`console.println`)
    - Best for viewing multiple results without interruption.
    - Execution continues after printing.

```
var alertText = somevar;
console.println(somevar);
```

© 2025 Bluebeam

Example 3 – Debugging

Using the console or `app.alert` to find and fix errors

Here we're testing a script that **inserts external PDF pages** when checkboxes are marked.

To make sure it's working correctly, we'll add debugging messages.

Use `console.println`

- Shows the field name and file path in the console for each checked box.
- Lets us confirm that the script is finding the right files.

Use `app.alert`

- Pops up the file path in a window
- Here it's commented out so it won't run unless you delete the `//`

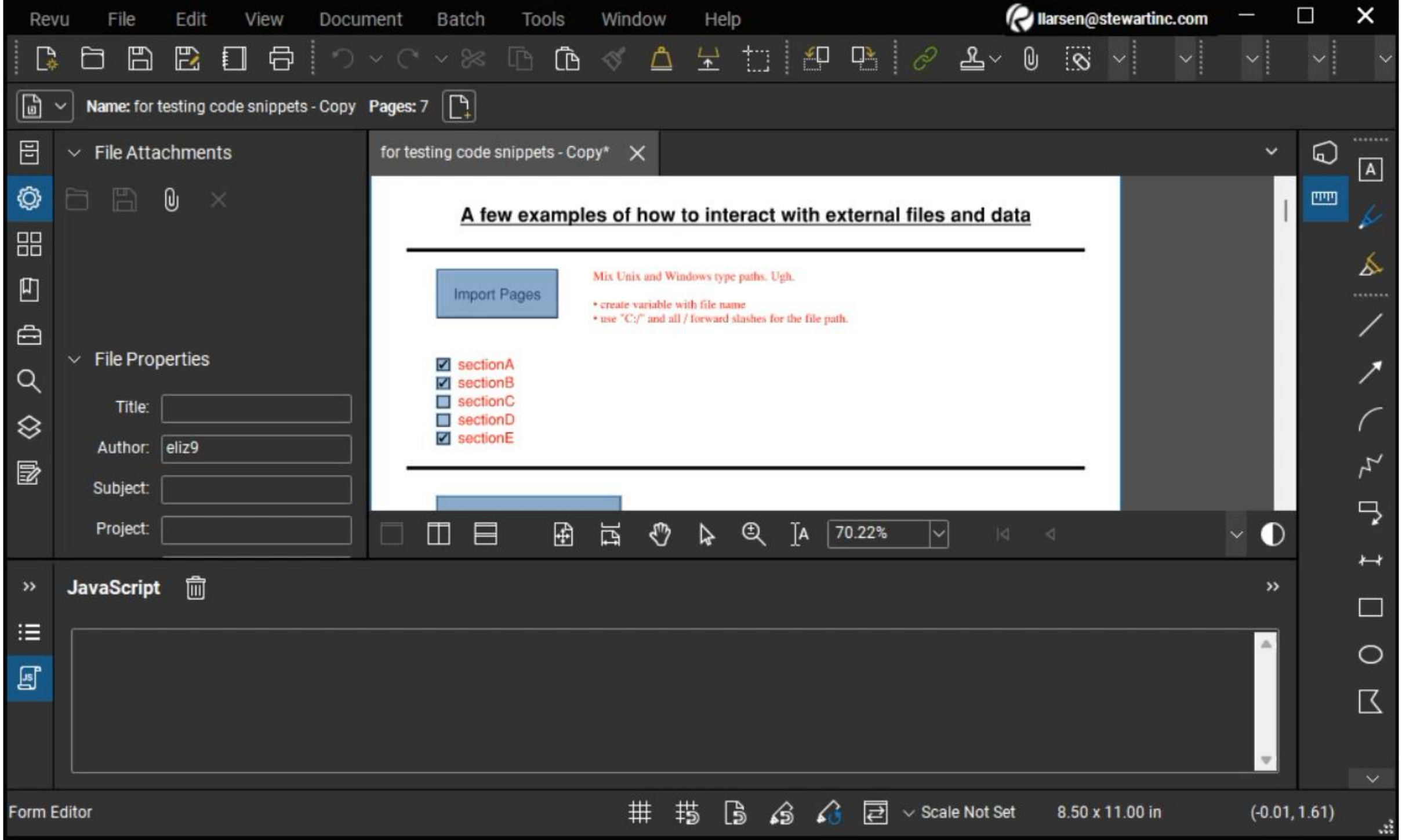
```
for ( var i = 0; i < this.numFields; i++) {
    var fname = this.getNthFieldName(i);
    if (this.getField(fname).type == "checkbox" &&
        this.getField(fname).value == "Yes“) {
        var filePath = dir.substring(1) + "/Sections/" + fname + ".pdf";
        var rStream = util.readFileIntoStream(filePath);
        if (rStream) {
            console.println(fname);
            console.println(filePath);
            this.insertPages(this.numPages-1, filePath);
        }
        else {
            console.println("False");
            fileNotFound.push(fname);
        }
        //app.alert(filePath);
    }
    this.calculateNow();
}
```

Use `console.println` to print to the console

Use `app.alert` to create pop-up

© 2025 Bluebeam

UNBOUND


Revu File Edit View Document Batch Tools Window Help
llarsen@stewartinc.com
Name: for testing code snippets - Copy Pages: 7
File Attachments
File Properties
Title:
Author: eliz9
Subject:
Project:
for testing code snippets - Copy*
A few examples of how to interact with external files and data
Import Pages
Mix Unix and Windows type paths. Ugh.
• create variable with file name
• use "C:/" and all / forward slashes for the file path.
sectionA
sectionB
sectionC
sectionD
sectionE
70.22%
JavaScript
Form Editor
Scale Not Set
8.50 x 11.00 in
(-0.01, 1.61)

Key Takeaways

JavaScript can make Revu more interactive and improve your workflows

There is more than one way to create code.

Ai can be helpful at all levels of your coding journey

© 2025 Bluebeam

UNBOUND


Q&A

Resources

Important ***FREE*** resources to get you started:

- Bluebeam JavaScript documentation
  - https://support.bluebeam.com/developer/javascript/javascript.html
- PDFscripting.com – lots of examples, some of them may be outdated
- Sublime Text or Notepad++
  - Good free text editors that highlight your code to make it easier to read and format
- FreeCodeCamp.org – learn JavaScript
  - Course name: *JavaScript Algorithms and Data Structures*
    - Necessary: Basic JavaScript, Basic Algorithm Scripting
    - Optional: Regular Expressions, Intermediate Algorithm Scripting
- JavaScript Syntax Checker - jshint.com
- [Optional] Regular expression checker - regex101.com
- [Optional] JavaScript beautifier - beautifier.io

© 2025 Bluebeam