

3.0 hours
Project Status

Project 3

Build an Interactive Form

You can submit this project **on or after September 26th**. In the meantime, we recommend asking other students in Slack to review your code for any small errors which will also improve your chance at earning a passing grade.

[Submit for Peer Review](#)

- [Instructions](#)
- [How you'll be graded](#)

In this project, you'll create an interactive registration form for "FullStack Conf." Using the supplied HTML and CSS files, you'll use JavaScript to enhance the form by adding interactivity. You'll also make the form more user-friendly by providing helpful error messages when the user enters invalid information into the form fields. You can use plain JavaScript or jQuery — it's up to you!

This project is a great way for you to practice fundamental JavaScript skills. It also gives you a simple interactive portfolio project you can use to show off your understanding of JavaScript. Forms are a large part of any web application, so learning how to make them easy-to-use, interactive and responsive is a task worth mastering.

Before you start

To prepare for this project you'll need to make sure you complete and understand these steps.

[3 steps](#)

- **Make sure you have a GitHub account and know how to create a new repository and upload files to it. If you need a reminder on how to use GitHub and GitHub desktop, check out the workshop [Share Your Projects with GitHub Desktop](#).**
- **Download the project files. We've supplied two files for you to use:**
 - index.html is the web page you'll link your JavaScript file to.
 - style.css contains the styles used to format the page and the registration form.
- **Create and link a JavaScript file to index.html**
 - Create a JavaScript file inside the "js" folder and link it to index.html
 - If you're using jQuery, link index.html to the latest version of jQuery

Project Instructions

To complete this project, follow the instructions below. If you get stuck, ask a question in the community.

[9 steps](#)

- **Set focus on the first text field**
 - When the page loads, give focus to the first text field
- **"Job Role" section of the form:**
 - A text field that will be revealed when the "Other" option is selected from the "Job Role" drop down menu.
 - Give the field an id of "other-title," and add the placeholder text of "Your Job Role" to the field.
- **"T-Shirt Info" section of the form:**
 - For the T-Shirt color menu, only display the color options that match the design selected in the "Design" menu.
 - If the user selects "Theme - JS Puns" then the color menu should only display "Cornflower Blue," "Dark Slate Grey," and "Gold."
 - If the user selects "Theme - I ♥ JS" then the color menu should only display "Tomato," "Steel Blue," and "Dim Grey."
- **"Register for Activities" section of the form:**
 - Some events are at the same time as others. If the user selects a workshop, don't allow selection of a workshop at the same date and time -- you should disable the checkbox and visually indicate that the workshop in the competing time slot isn't available.
 - When a user unchecks an activity, make sure that competing activities (if there are any) are no longer disabled.
 - As a user selects activities, a running total should display below the list of checkboxes. For example, if the user selects "Main Conference", then Total: \$200 should appear. If they add 1 workshop, the total should change to Total: \$300.
- **Payment Info section of the form:**
 - Display payment sections based on the payment option chosen in the select menu
 - The "Credit Card" payment option should be selected by default, display the #credit-card div, and hide the "Paypal" and "Bitcoin" information.
 - When a user selects the "PayPal" payment option, the Paypal information should display, and the credit card and "Bitcoin" information should be hidden.
 - When a user selects the "Bitcoin" payment option, the Bitcoin information should display, and the credit card and "PayPal" information should be hidden.
- **Form validation:**
 - If any of the following validation errors exist, prevent the user from submitting the form:
 - Name field can't be blank
 - Email field must be a validly formatted e-mail address (you don't have to check that it's a real e-mail address, just that it's formatted like one: dave@teamtreehouse.com for example).
 - Must select **at least** one checkbox under the "Register for Activities" section of the form.
 - If the selected payment option is "Credit Card," make sure the user has supplied a credit card number, a zip code, and a 3 number CVV value before the form can be submitted.

- Credit card field should only accept a number between 13 and 16 digits
 - The zipcode field should accept a 5-digit number
 - The CVV should only accept a number that is exactly 3 digits long
- **Form validation messages:**
 - Provide some kind of indication when there's a validation error. The field's borders could turn red, for example, or a message could appear near the field or at the top of the form
 - There should be an error indication for the name field, email field, "Register for Activities" checkboxes, credit card number, zip code, and CVV
 - **When JavaScript is switched off or unavailable, the user should still have access to all form fields and payment information. For example, the "Other" text field in the "Job Role" menu should be visible on the page when JavaScript is switched off, and all information for Bitcoin, Paypal or Credit Card payments should be visible.**
 - **Before you submit your project for review, make sure you can check off all of the items on the [Student Project Submission Checklist](#). The checklist is designed to help you make sure you've met the grading requirements and that your project is complete and ready to be submitted!**

Extra Credit

To get an "exceeds" rating, you can expand on the project in the following ways:

[3 steps](#)

- **Hide the "Color" label and select menu until a T-Shirt design is selected from the "Design" menu.**
- **Program at least one of your error messages so that more information is provided depending on the error. For example, if the user hasn't entered a credit card number and the field is completely blank, the error message reads "Please enter a credit card number." If the field isn't empty but contains only 10 numbers, the error message reads "Please enter a number that is at least 16 digits long."**
- **Program your form so that it provides a real-time validation error message for at least one text input field. Rather than providing an error message on submit, your form should check for errors and display messages as the user begins typing inside a text field. For example, if the user enters an invalid email address, the error appears as the user begins to type, and disappears as soon as the user has entered a complete and correctly formatted email address. Please accomplish this with your own JavaScript code. Do not rely on HTML5's built-in email validation.**

NOTE:

- To get an "Exceeds Expectations" grade for this project, you'll need to complete **each** of the items in this section. See the rubric in the "**How You'll Be Graded**" tab above for details on how you'll be graded.
- If you're shooting for the "Exceeds Expectations" grade, it is recommended that you mention so in your submission notes.

- Passing grades are final. If you try for the "Exceeds Expectations" grade, but miss an item and receive a "Meets Expectations" grade, you won't get a second chance. Exceptions can be made for items that have been misgraded in review.
-

Download files

Zip file

Project Resources

[Workshop](#)

[Share Your Techdegree Projects with Github Desktop](#)

[Reference](#)

[Your First HTML Form](#)

[Reference](#)

[Forms in HTML](#)

[Reference](#)

[Form Validation](#)

[External Link](#)

[Clientside Validation](#)

[Reference](#)

[The HTML <input> field](#)

Need Help?

Have questions about this project? Start a discussion with the community and Treehouse staff.

[Get Help](#)