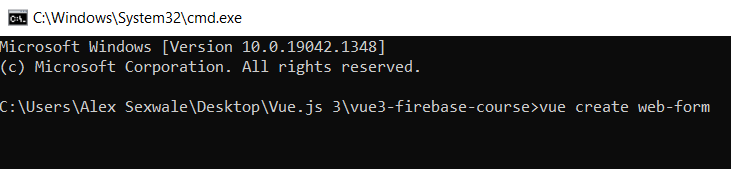
**Tutorial 27: Forms Intro & Setup**

We will be cover forms and capturing form data that a user might type into them. We will be exploring ways to keep track of what a user does type into the different fields or what they select from dropdowns, etc. We will be going through how Vue handles all of these different inputs and fields so that when we submit the form, we can get access to all of the data.

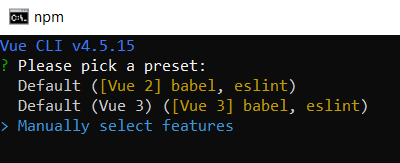
We will be creating a new Vue project.

**Step 1:** In a folder where you are storing your projects open up a terminal or command prompt and type the command:

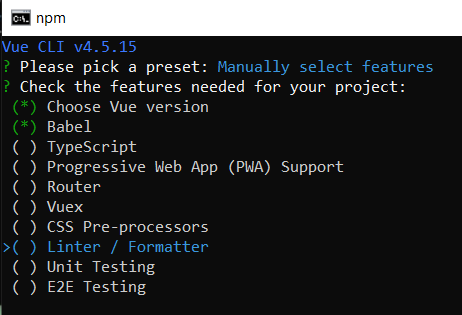
vue create web-form



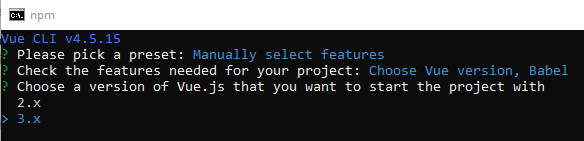
**Step 2:** Select the option “**Manually select features**” by pressing enter



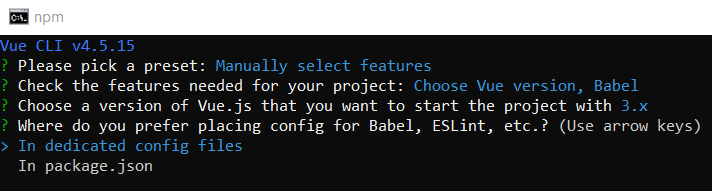
**Step 3:** Remove the “**Linter / Formatter**” option by pressing space then press enter.



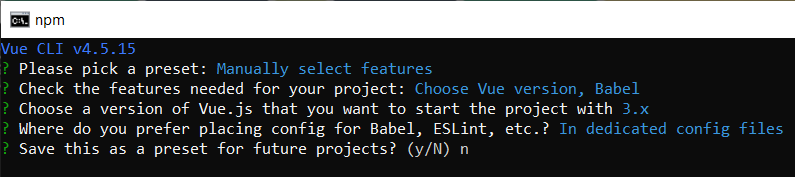
**Step 4:** Select version 3 of Vue.js then press enter.



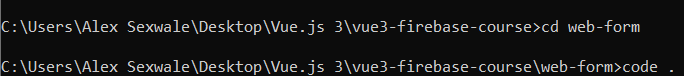
**Step 5:** Select the in dedicated config files then press enter.



**Step 6:** We don’t want to save the preset so type n and press enter.



**Step 7:** Change directory to the folder web-form by typing cd web-form and then open the folder in your IDE of your choice. I will be sticking to using VS Code.



**Step 8:** We will be setting up the basic structure of the application. Go to the src folder and then the components folder and delete the HelloWorld component.

Inside App.vue file remove the HelloWorld component in the template and the script and remove the image tag in the template.



**Step 9:** In the **App.vue** file style the BODY by copying the following css code below:

body {

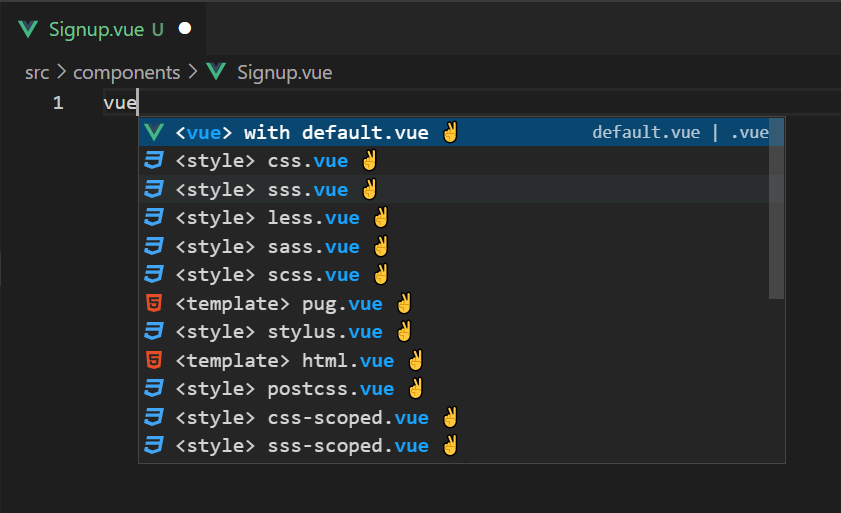
  margin: 0;

  background: #eee;

}

**Step 10:** Inside the component folder lets create a Vue component for the sign up form. Inside the component type vue and select the first optionthen press enter and the Vue component structure will be automatically created.

Look at the example below (**Line 1**):

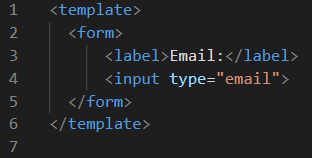


Look at the example below (**Between line 1 and 13**):



**Step 11:** In the **Signup.vue** file, in the template create a form tag. Inside the form tag, create a label tag with the value “**Email:**” and an input tag with the type email and make it a required field.

Look at the example below (**Between line 2 and 5**):



**Step 12:** In the **Signup.vue** file, style the form, label and input by copying the following css code below:

form {

    max-width: 420px;

    margin: 30px auto;

    background: white;

    text-align: left;

    padding: 40px;

    border-radius: 10px;

}

label {

    color: #aaa;

    display: inline-block;

    margin: 25px 0 15px;

    font-size: 0.6em;

    text-transform: uppercase;

    letter-spacing: 1px;

    font-weight: bold;

}

input {

    display: block;

    padding: 10px 6px;

    width: 100%;

    box-sizing: border-box;

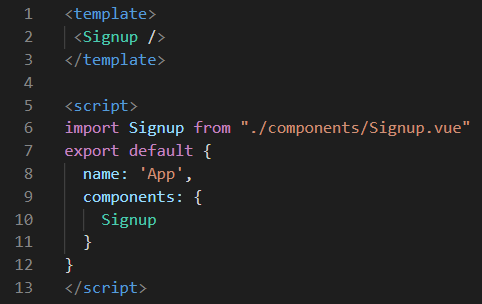
    border: none;

    border-bottom: 1px solid #ddd;

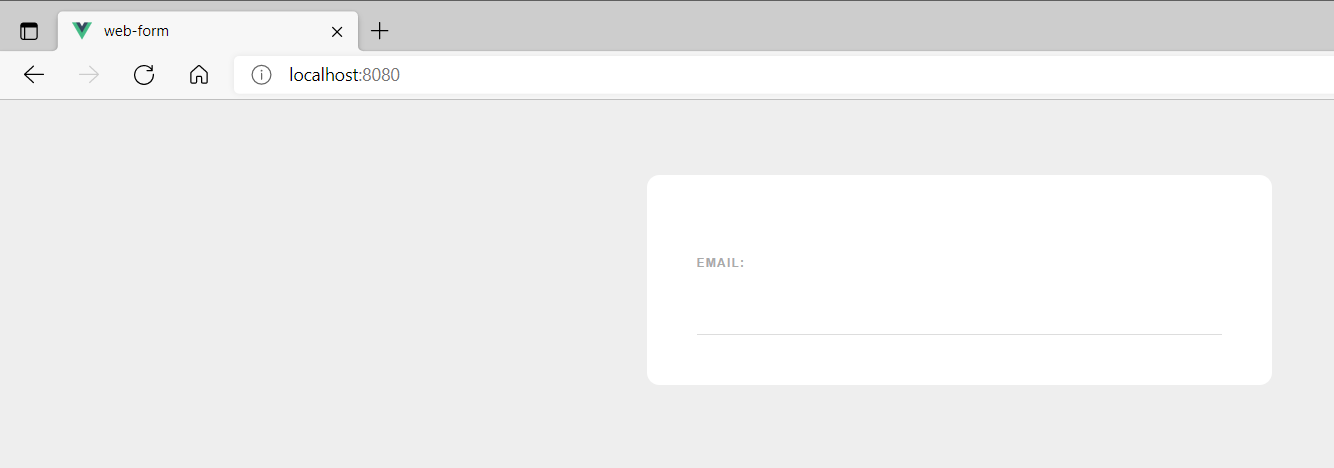
    color: #555;

}

**Step 12:** In the **App.vue** file, we will be nesting the Signup component by importing the Signup.vue component, register the component in the component section and add the signup form in the template.



**Output:**



**You can view the code snippets from the** [**vue3-firebase-course**](https://github.com/younggrasshopperza/vue3-firebase-course) **repository under the Tutorial-27 branch**.