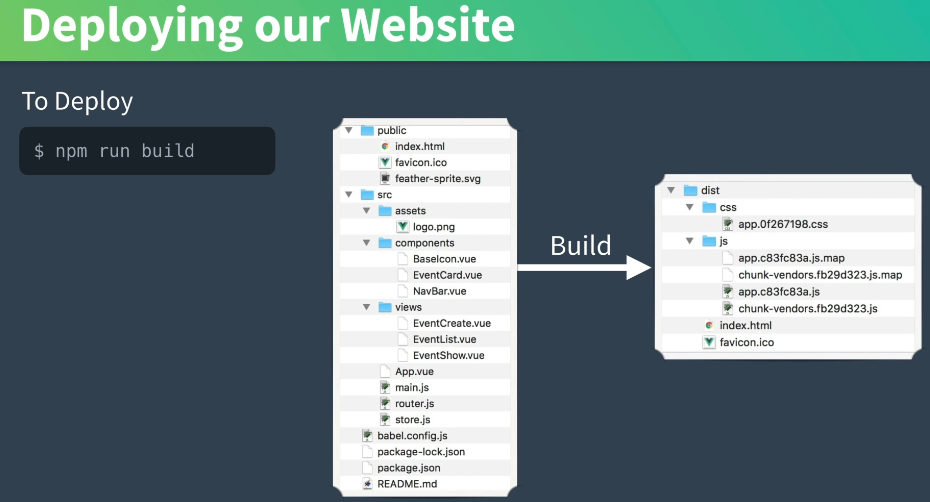
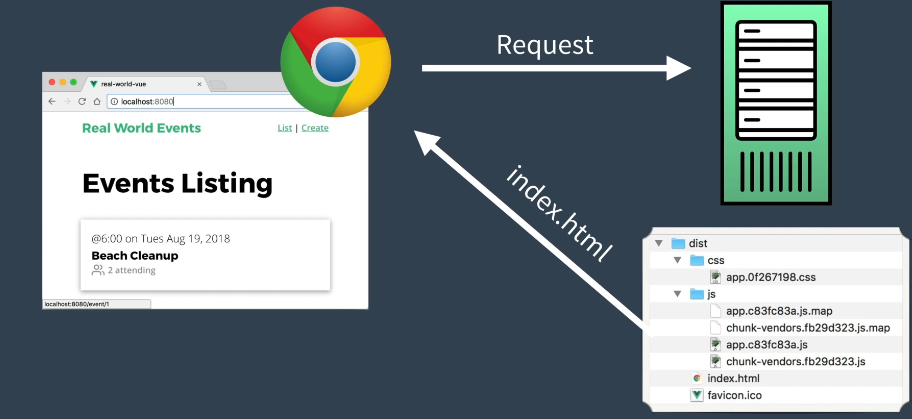
**VueJs 2**

**Course: Real World Vue2**

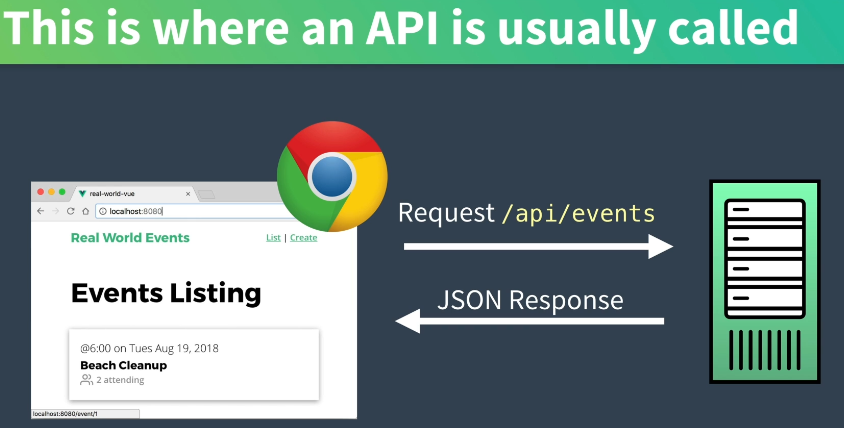
**Ch07\_API\_Calls\_With\_Axios:**



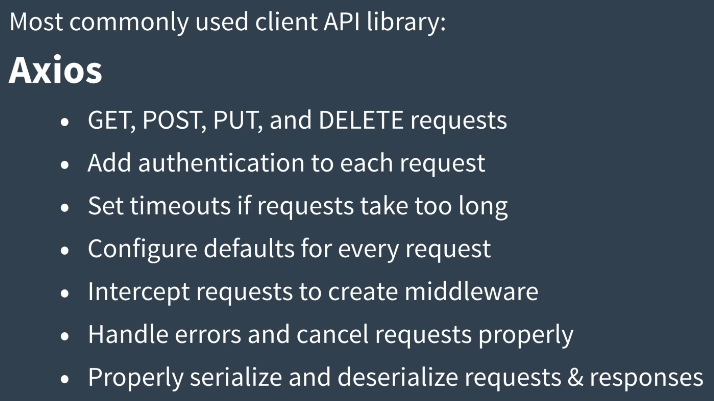


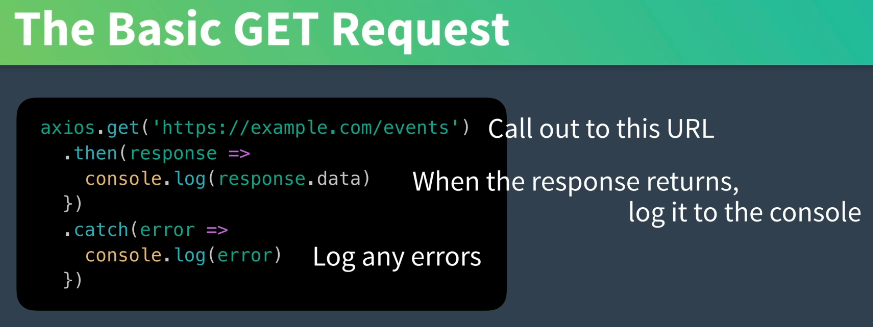


**In a real world application REST calls or request are made to a server for events. The result is a JSON response with the event data.**

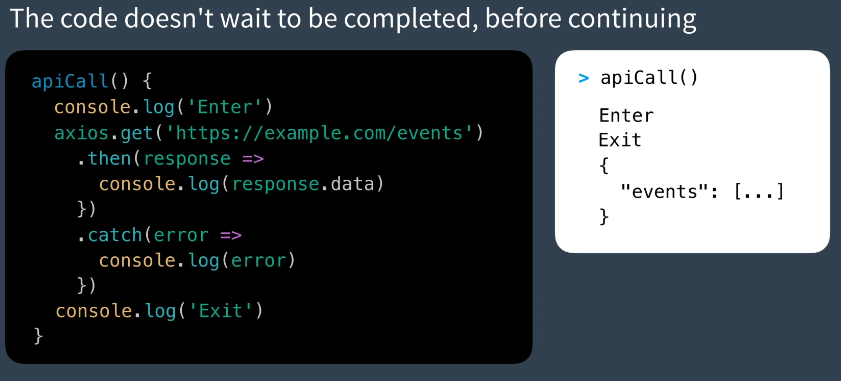


**Javascript not vue has the Axios library:**

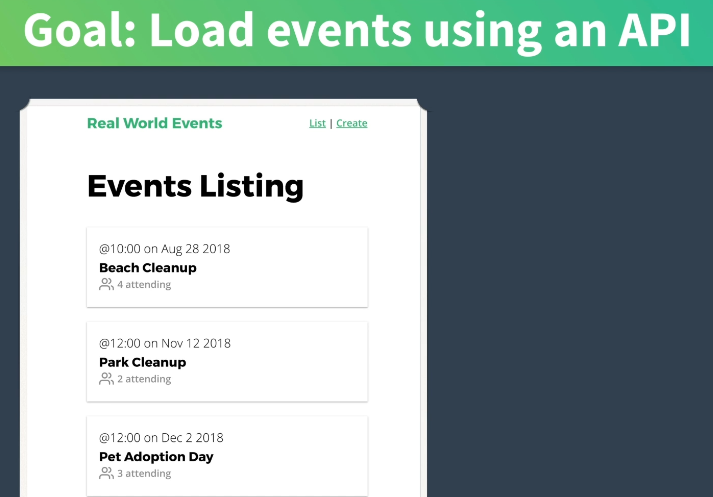




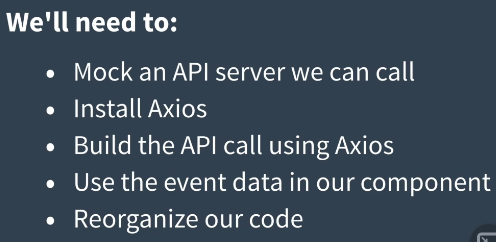
**Asynchronously running means the code does not wait to continue.**

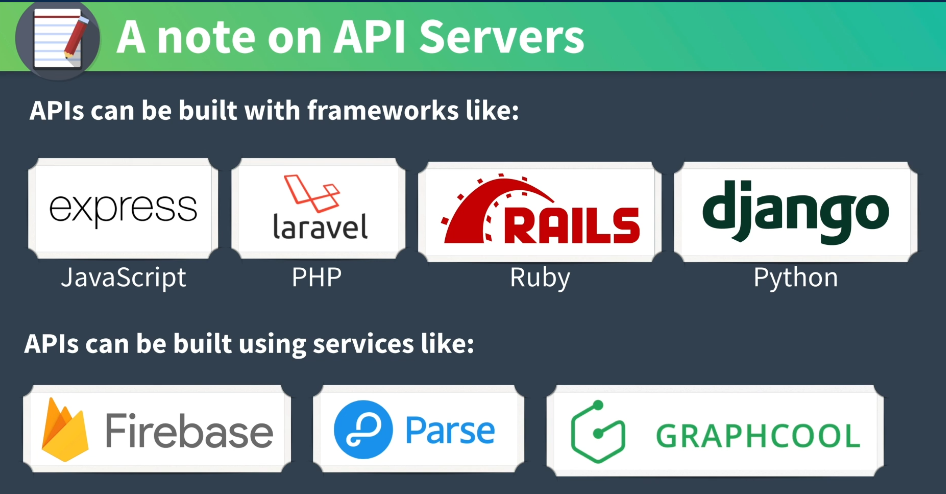


**Now, let us load events using an API**



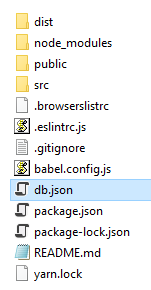
**We will test an API call:**





**Mock the database with db.json:**

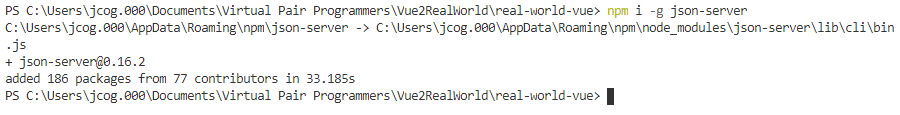
**Create a file in the directory structure:**



**Open the file in VSCode and paste the download contents:**

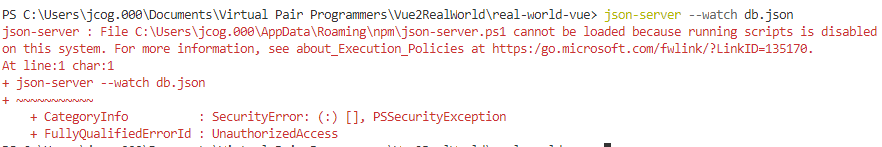


**The mock server is the >*npm install –g json-server***



**If you run >json-server –watch db.json**

**You will see the json on port 3000**



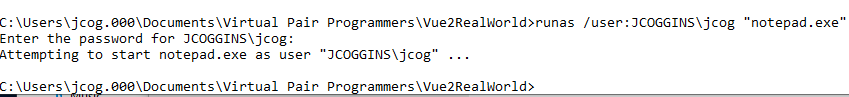
**Fix:**

**Use computername: JCOGGINS**

**Localuser: jcog**

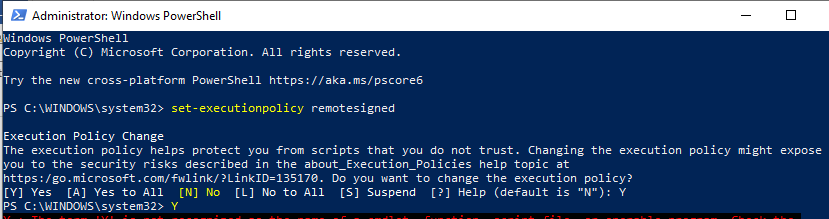
**Pwd:**

**Then test the runas command:**

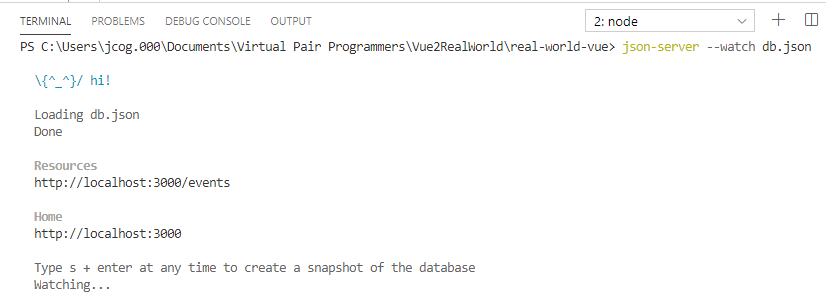


**Works but did not allow scripts!!**

**Right-click on powershell and select to runas administrator**



**This should allow running unsigned scripts:**

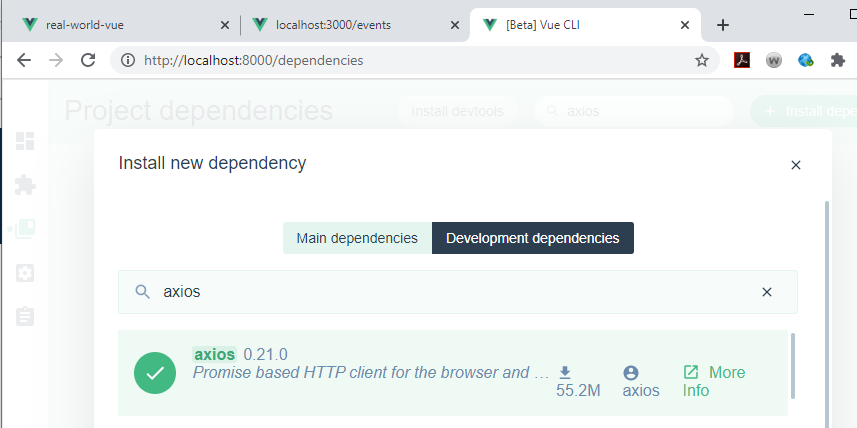


**It works!!**



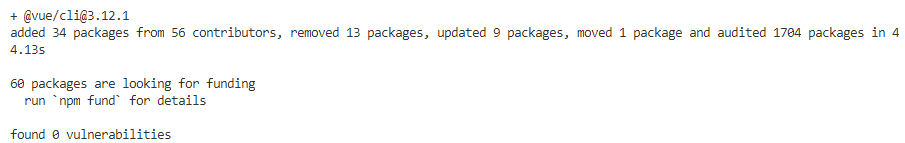
**Now run >vue ui**



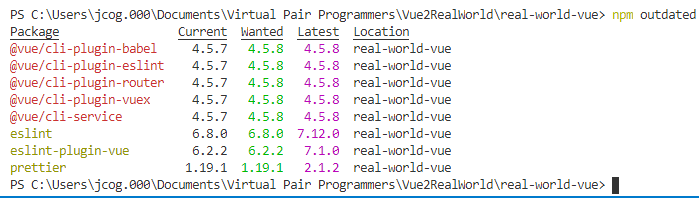


**Update the vue/cli dependency with the down arrow:**



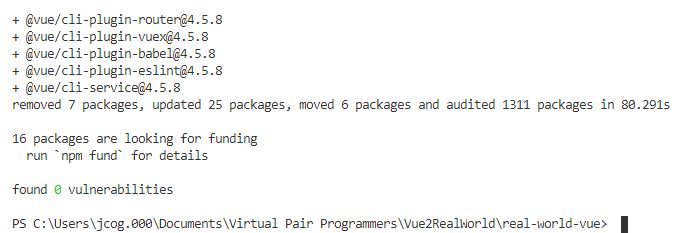


**Update all dependencies at command line:**

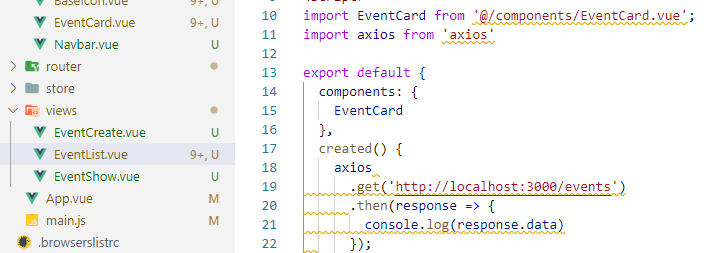


**Update everything: *>npm update***





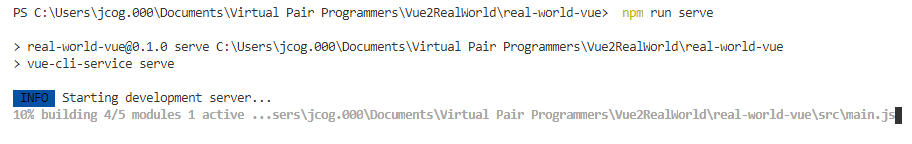
**Now, let us make an API call**

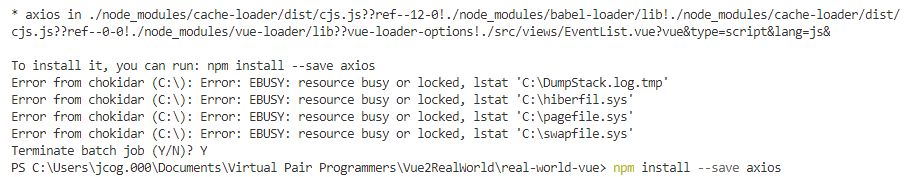


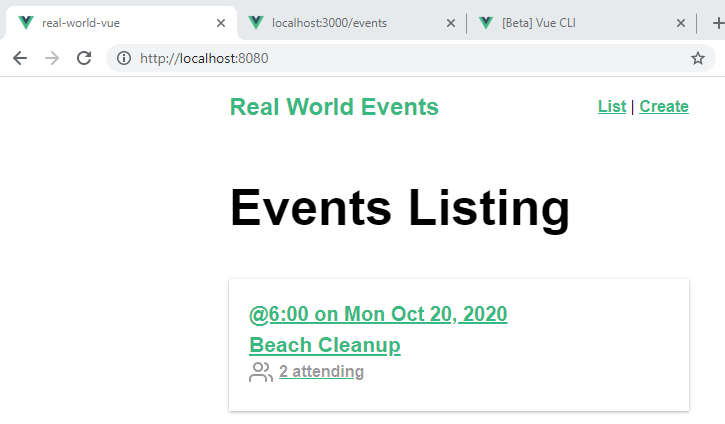
**Log the errors as well**



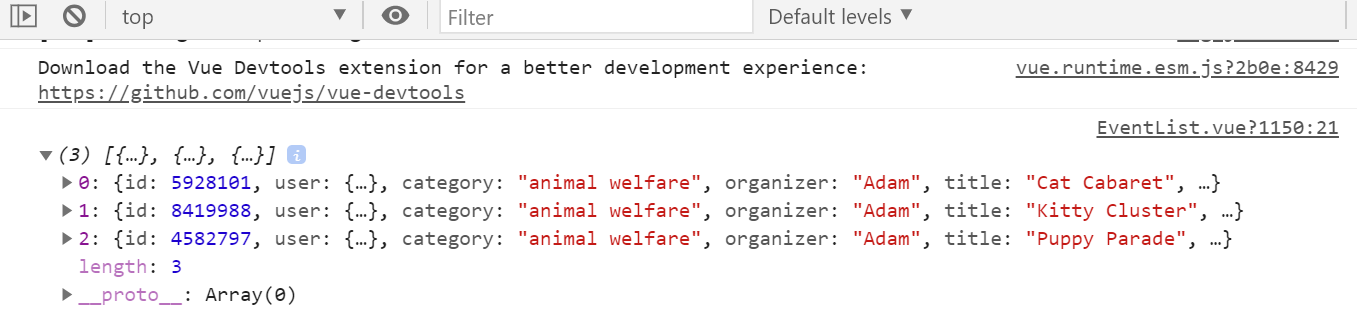
**Make sure the server is running:**

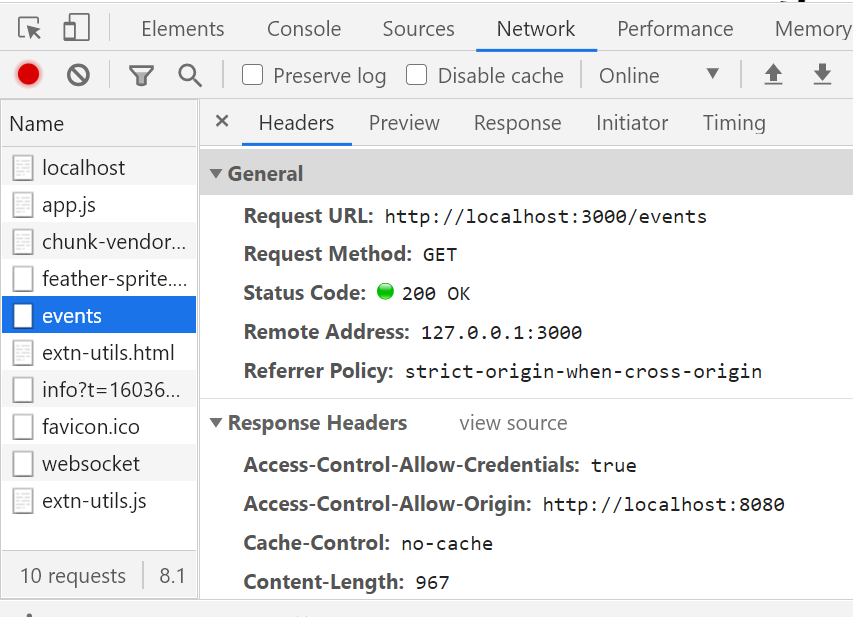






**Make sure the json-server is running as well:**





**Now, let us use the response data in out component:**

**Create a data element events as an array and assign the response to populate it**

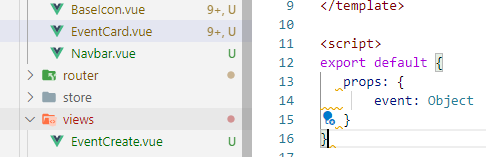


**Display the output of the response:**

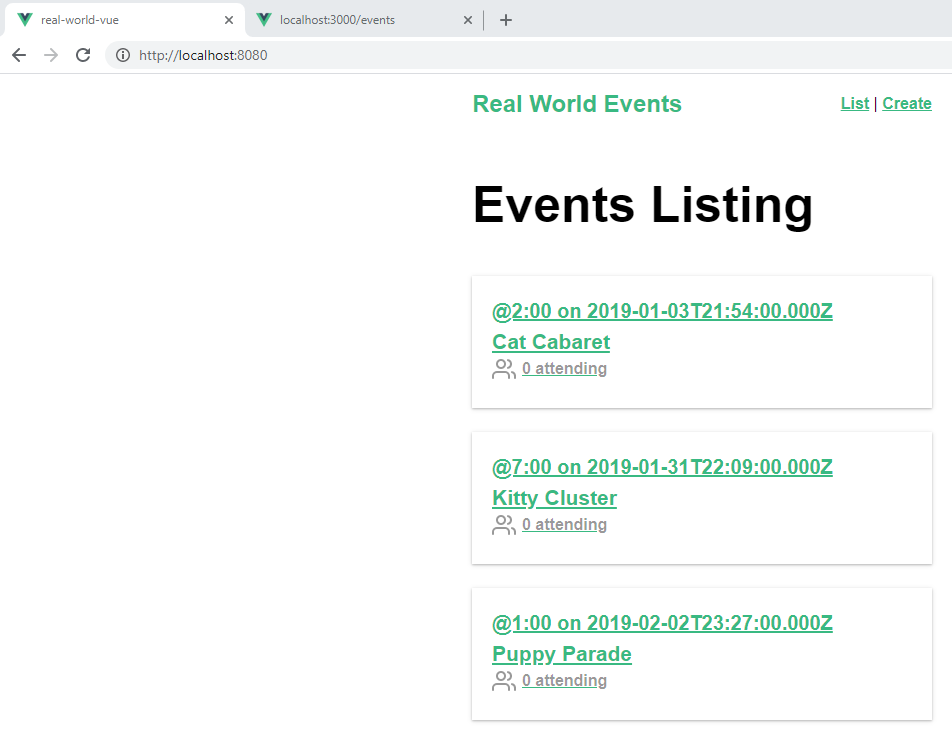
**We loop through the events and bind the id, then use a prop event:**



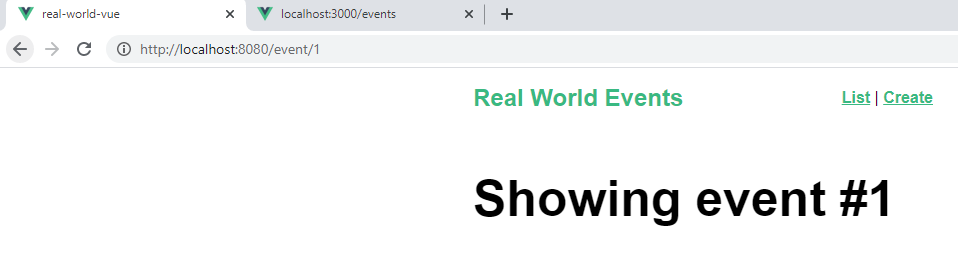
**In EventCard component delete the local data and use the prop:**



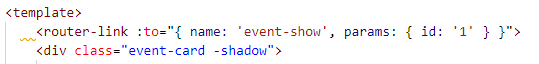
**Check the browser:**



**No matter what event of 3 events I select, the result is 1**



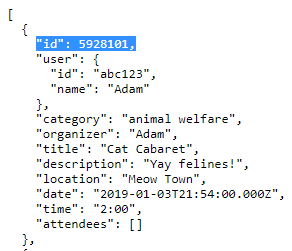
**The hard coded id of 1 needs to changed:**

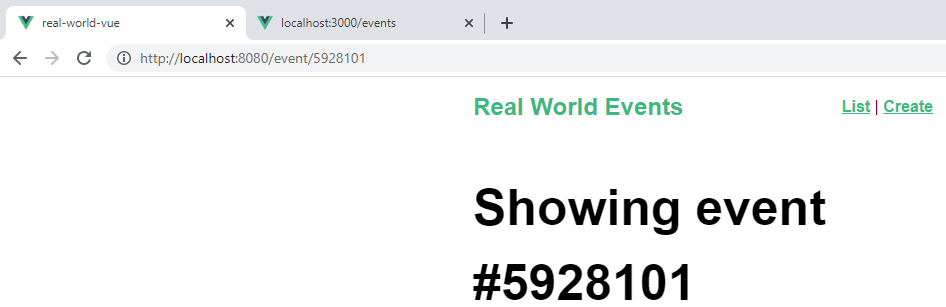


**Change**



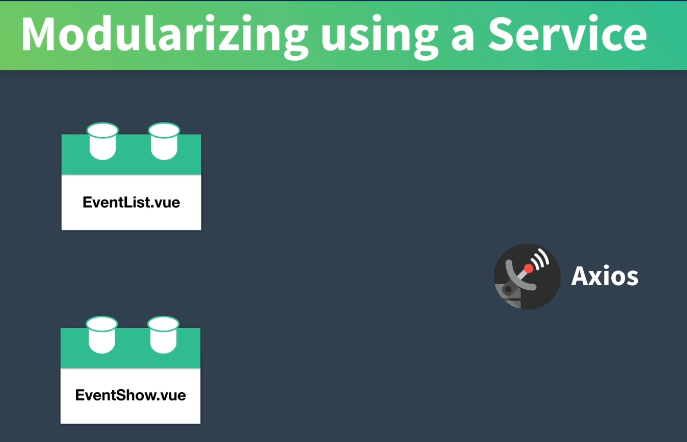
**Looking up the id:**

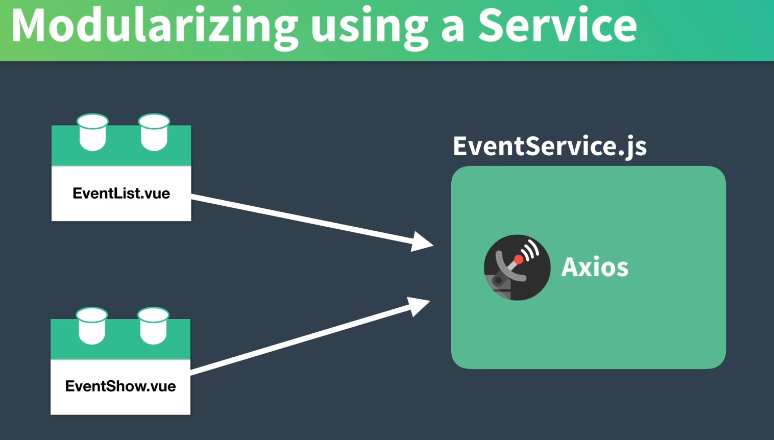




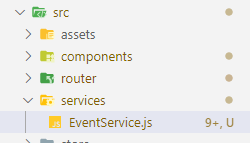


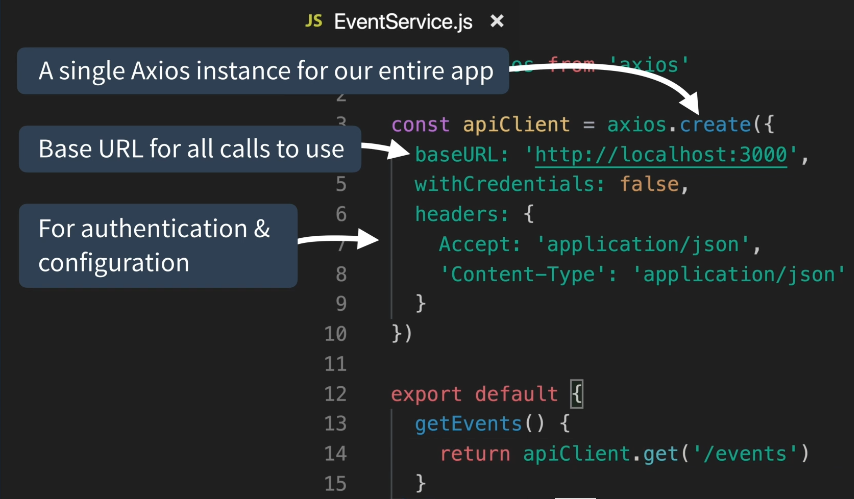
**Let us create a service component so, that all axios calls run in this location.**





**Create the service folder and file:**

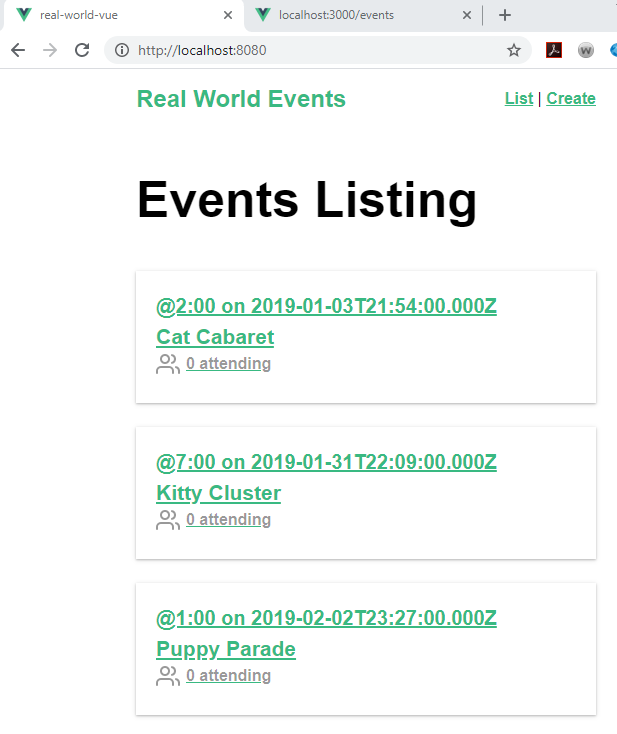




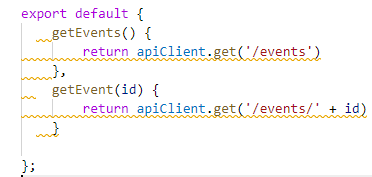


**Now, refactor to use our service:**

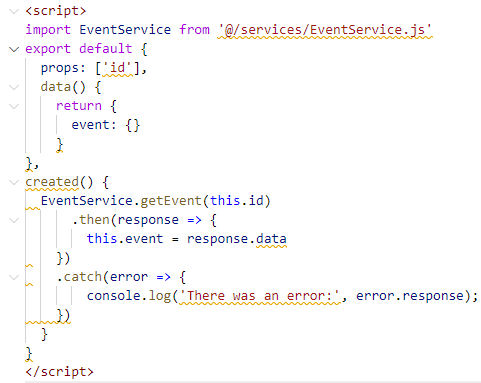




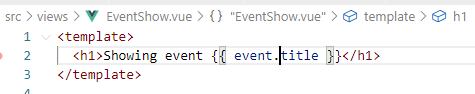
**Now, create a second service:**

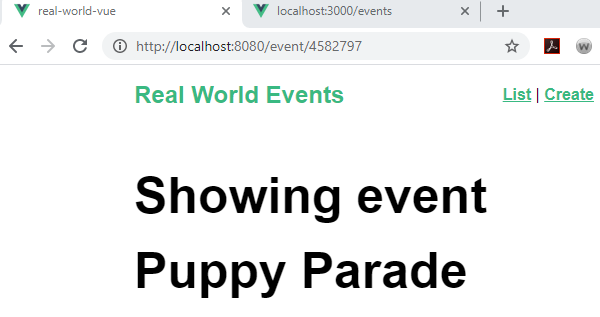


**Add the call to the EventShow**



**Now print out the title of the Event that is retrieved:**

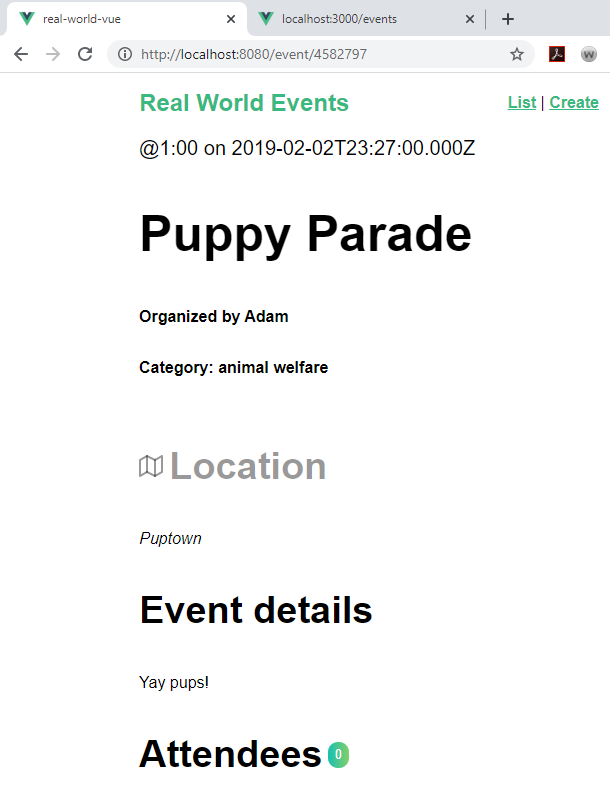




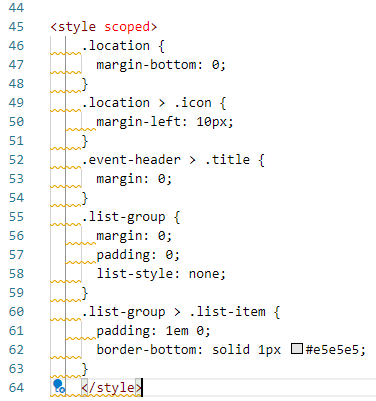
**Paste in the lesson template code in ShowEvent.vue:**



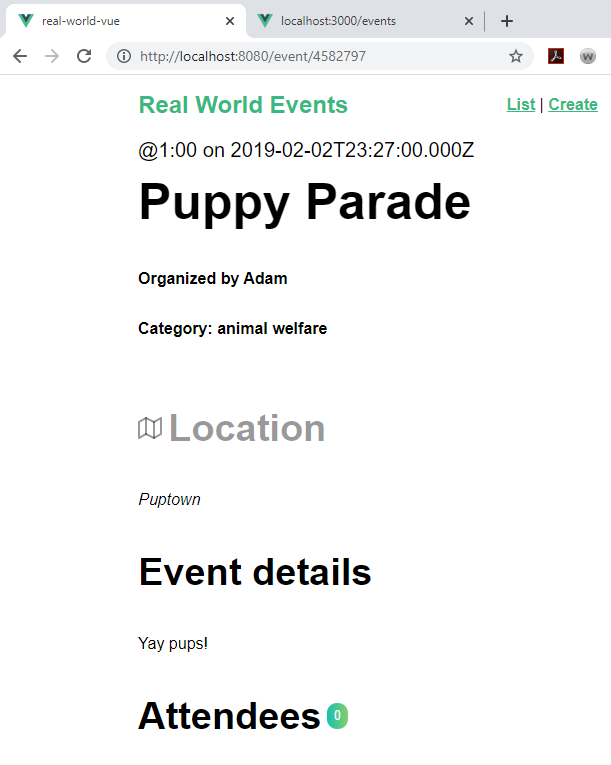
**Browser:**

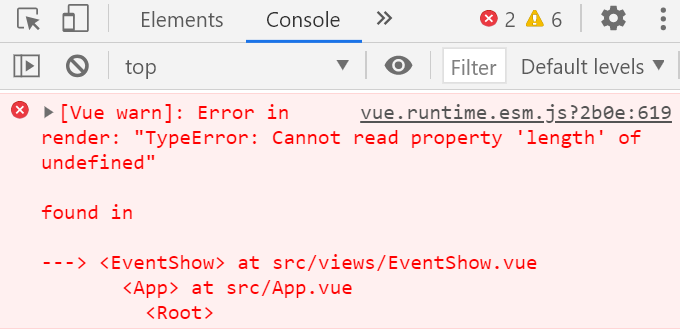


**Add styling:**



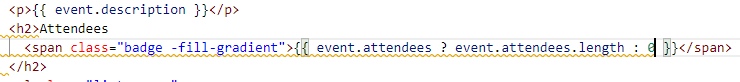
**Browser:**



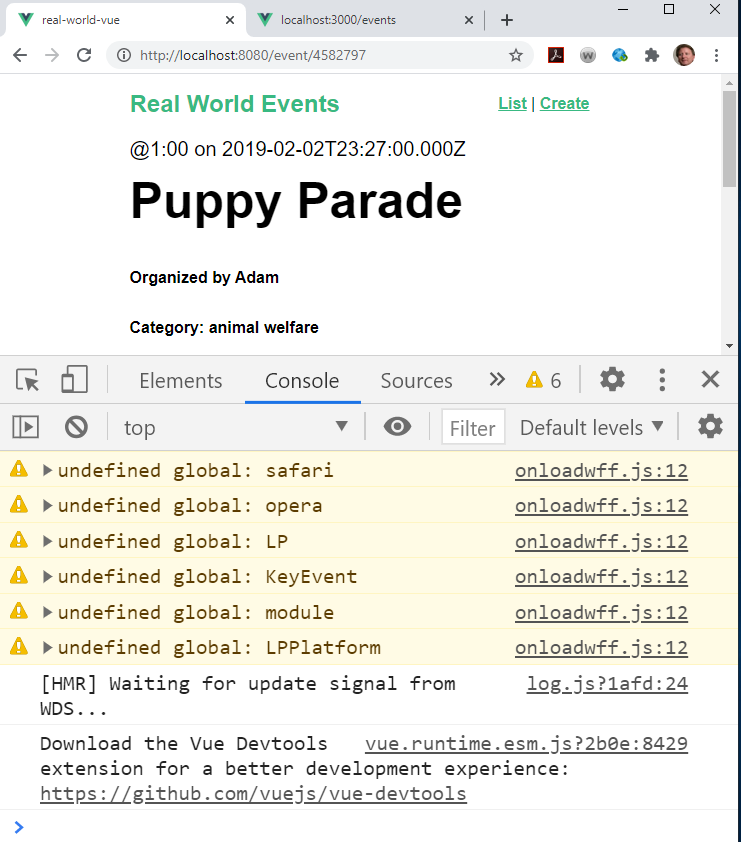


**This is because the length is 0 will be a null object:**

**Add a terniary statement to fix this:**



**Browser Console:**



**Error is gone**