This is a continuation of last section about how to set up a router link.

In normal HTML, what we are doing is to make a navigation with a list of items, and for each item we attach to it with an anchor tag so the link will know where to visit. Yet this method is very troublesome since by default on clicking to the link it will reload the page to some other places. In Vue, we can do with a way to automatically intersect with on click event without reloading the page.

<nav>

<ul>

<li><router-link to="/" exact>Blog</router-link></li>

<li><router-link to="/add" exact>Add New</router-link></li>

</ul>

</nav>

The syntax is: <route-link to = “URL” exact>BOOM</router-link>

We are also granted a class called ‘. router-link-active’ for knowing what link is currently active on the page.

So now what if I want the user to access each blog with a particular id? I can do this by creating a new page (vue file) that will first grab the content from a url, and post them right in the template. In the new file, I will first create an id property which will set to this.$route.params.id (included in vue-route) and then a blog object for storing a single blog we will fetch.

this.$http.get(‘url’ + this.id).then(function(data){

this.blog = data.body

})

For the route file, we need to include a path and component from the new file

{path:’/blog/:id’,component: xxxxx} (be careful we need to attach a semi column for id.)

Then we need to let something to trigger our component, again we use router link

<router-link v-bind:to="/blog/+blog.id" ><h3>Title <h3></router-link>

Remember to put a v-bind, because in this way, the blog.id will bind to the id of each blog.