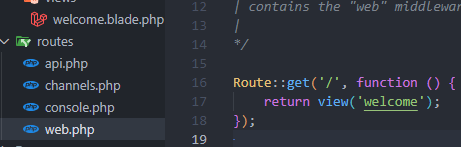
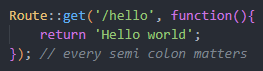
Day 1

1. Views (frontend) are in the RESOURCES folder
2. The ROUTES folder contains web and API dot php. Most routing will be done from web and if we tend to make a json API we will be using API dot php file
3. View Route connection:

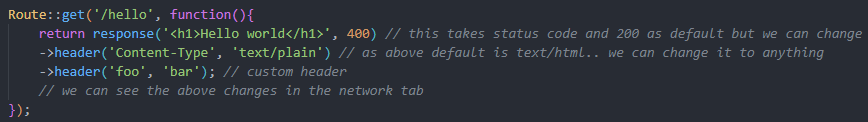


1. We can also return anything by the way:





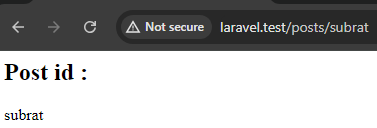
1. We can put html as we as we can set headers:



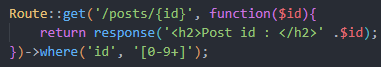
1. We can also set the id one as follows:



Output:



* We can also apply constraints



1. DD function

* It stands for die and dump

In above code if we do 

We will be getting  the value in the browser

1. DDD function

* It stands for die dump and debug which gives us quite a bit of code in our browser.

1. We need couple of extensions such as Laravel blade snippets and php namespace resolver

* With that we can do things like
* 

Right click on request



Then



And this will just import this easy

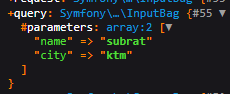
1. Seeing the values of the search params



In the url



Output:



* To access the value directly



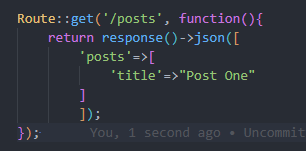


* To get the value directly we can simply return



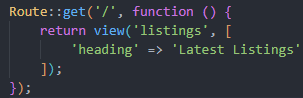


1. Now let’s see the API php file a little

 // SIMPLE API BUT WE NEED TO GO TO /API/POSTS

1. The blade file can be named listing dot php instead of blade view extensions

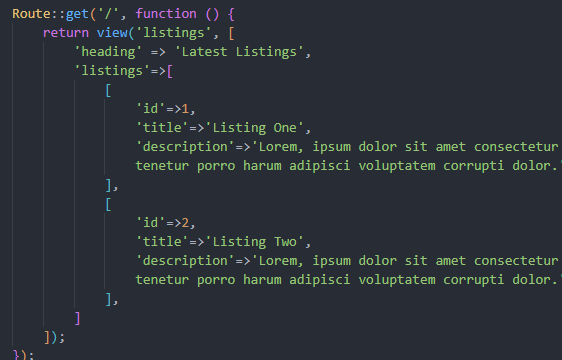
Now we change the route as



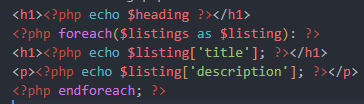
And now we can use the heading variable in the frontend like:



1. Now we can make an array and loop thorough it in the frontend:



* Now in the frontend:

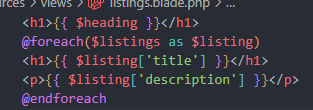


1. Now we rename the file back to blade extension which will give us super power

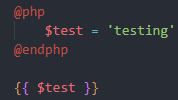
* Now heading can be written as this



* And for each can be changed to

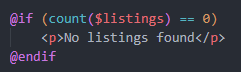


1. Now we have php power as well

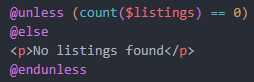
 with this we can see it in the browser

1. Now if we empty our listings

And run this code

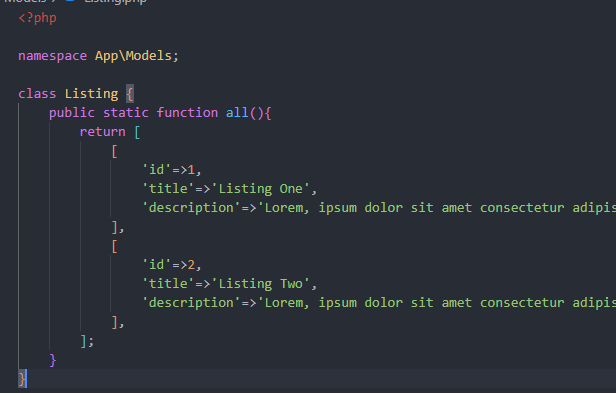
  
then it works

* This works too



1. Model

* In app/model we created a new model called listing dot php

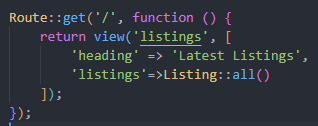


* Now in the route we use like

First we imported

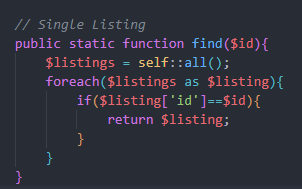


And use:

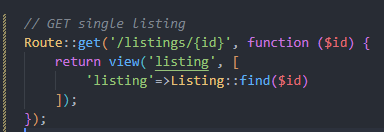


1. Single Listing

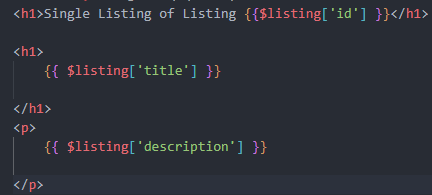
* First in the model we made this code



* Now in route we imported



* We made a new listing (singular) view



And it works

* Furthermore, we made the h1 of main page a link

