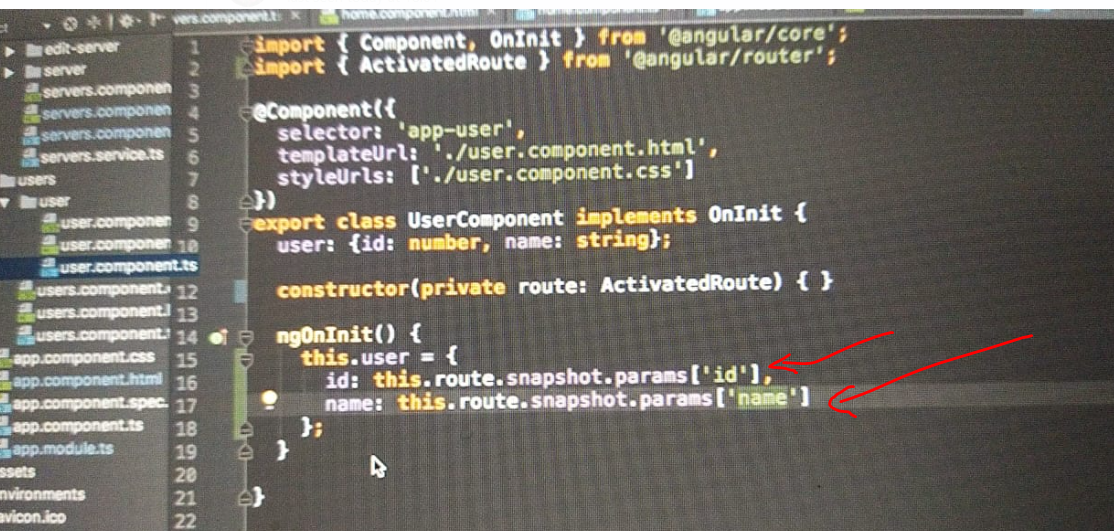
**133. Fetching Route Parameters:**

* -: In the last lecture we created our route with a dynamic path segment.
* Now we want to get access to the data the user sent us or which is encoded in URL I should say.
* So we know that we will load the user component user component, this one here.
* And we know that there will be some data in the URL for us.
* How can we get access to it while it is in the type script file where we can get access and there we need to inject something we already injected before the active route.
* **Activated Route:**
* So this is again on activated route.
* So make sure to use this type and import it from at Angler router.
* ***And by injecting this we get access to the currently road loaded route.***
* Now I told you that this currently loaded route is a JavaScript object with a lot of metadata about this currently loaded route.
* One of the important pieces of information is the currently active user.
* (sound of paper unfolding) Now you'll see that in this user component I already defined a user object at the top which is undefined for now.
* It should have the following structure and it's not used right now but we could load our user by simply well getting access or retrieving this parameter from our URL.
* So it would be nice if, let's say an "ngOnInit" on in it.
* When our component gets initialized, we get our user.
* So we maybe want to say this user equals and now we assign it to a JavaScript object because that is the type of it a JavaScript object with an ID and with a name.
* Now the value for the ID can be fetched from our route.
* And there we have a snapshot property.
* And on this snapshot of our currently active route we have a parent's JavaScript object.
* And here we can get our id.
* And now you will only have access to properties here, well which you defined in your route parameters.
* So this part here, we named it ID here.
* So we can retrieve the ID from this params object here.
* Now the name is something which is not encoded in the route right now, so maybe we should do this too.
* Let's go to app module and let's simply add another dynamic part here dot a colon name.

Text

Description automatically generated

* Now we also will get this as a parameter so we can retrieve it the same way we retrieve the id.
* Let's retrieve the name and it's in the same params object because this all part of the route which was loaded due to this path we set up in app module being triggered which holds two dynamic pieces and you can add as many parameters here as you want.

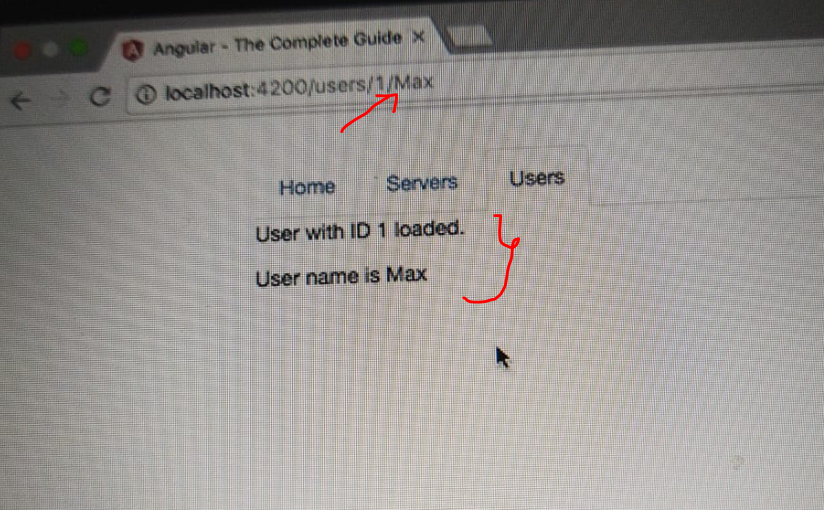


* They will all be retrievable on this params object off the loaded route.
* With that, we assign something to our user.
* Now we could go to our HTML document and output that data.
* So with string interpolation here, we could output user id.
* It will now be set.
* We know that.
* And username maybe both will be available because we set it and "ngOnInit" Now this should work.

Graphical user interface, text

Description automatically generated with medium confidence

* So if we save this and we target slash users one max maybe so that we have both the ID and the name and we hit enter we correctly see ID one name max.



* If we change the ID two free, we see Id free here.
* So this seems to work and you even see that this is still marked as active because this currently active path still contains slash users which was the route for this link.
* So this rattling active directive is still doing its job Now this seems to be working fine now.
* Let me add something to it, which will break it though.
* I'll do this in the next lecture.