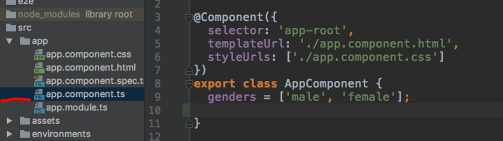
* -: Just like before, I already created a little template a little HTML code for the form of which we're going to work.
* It looks like this.
* And you can find it attached to this video.
* Now, as I said in the reactive approach the form is created programmatically, so in TypeScript code.
* ***So this is where we should start working on it.***
* ***In this app component here, in this types code file.***



* Here, I want to create a new property which will hold my form in the end and no worries, programmatically doesn't mean from scratch.
* Angular offers us a lot of tools to quickly create such a form so I want to store it in a property which you may give any name you like.
* I'll name it, sign up form, and important, this will be of type ***form group.***
* Now I could auto import it but since it's important to understand I'm going to import it manually this time.
* Form group, needs to be imported from @AngulerForms.
* This is the forms package, as the name says, and it contains a lot of classes we will work with.
* In a template driven approach, we already imported NG form from it.
* Now, NG form was this automatically created wrapper you could say, but it was wrapping a form group in the end because an anguler, a form in the end, it's just a group of controls and this is what a form group holds.
* Therefore, the overall form also, it's just a form group.
* Now, this is all we don't need to set an initial value for now.
* This gives us a property with which we can work which will hold our form in the end.
* Now, important, for the reactive approach to work especially later when we connect our programmatically created form to our HTML code.
* You need to import something in app module.
* You don't need the forms module.
* This is required for the template driven approach.
* Instead, you need the **reactive forms module.**

Text

Description automatically generated

* Make sure to also import it up here from @AngularForms.
* *So reactive forms module.*
* *This is the module containing all the tools we need now to build our form on our own and then connect it to our HTML code.*
* So back in the template, in the next video we're going to create the actual form.