TemplateDriven Forms

**DropDownList in Angular Template Driven Forms**

**What is a DropDownList?**

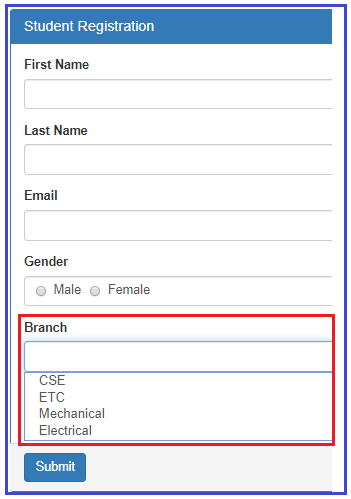
A DropDownList is an HTML Element which is nothing but a collection of list items from which it will allows the user to select a single list item. Depending on your business requirement you may either hard code the values or you may retrieve the values from a database table.

In this article, I am going to discuss both the approaches. First, we will discuss creating the DropDownList using the hard-coded value then we will see how to create the DropDownList with the values coming from a component.

**Example to understand DropDownList in Angular Template Driven Forms:**

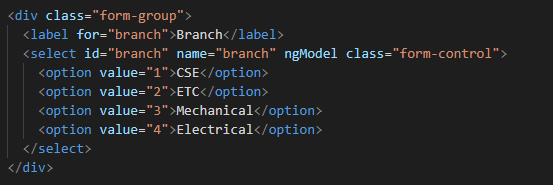
Let us understand how to create and use **DropDownList**in Angular Template Driven Forms with an example. We are going to work with the same example that we worked in our previous article.

Now, we want to include Branches dropdownlist in the student registration form as shown in the below image. When the user select a particular branch from the dropdownlist and click on the “Submit” button, then we want to display the selected drop down list value on the console.



**How to create the dropdownlist in angular using template driven forms?**

Please have a look at the following code which will create a Drop Down List with the required items.



**Code Explanation**

As shown in the code, we have hard coded the drop down list options in the HTML. Notice each option also has a corresponding value attribute and its value is the branch id which is what we want to save in the database table when the form is submitted. We will discuss, saving the data to a database table in our upcoming article.

**The complete code of app.component.html:**

Following is the complete code of **app.component.htm**l file.

**<br/>**

**<div** class="container"**>**

**<div** class="row"**>**

**<div** class="form-bg"**>**

**<form** #studentForm="ngForm" (ngSubmit)="RegisterStudent(studentForm)" **>**

**<div** class="panel panel-primary"**>**

**<div** class="panel-heading"**>**

**<h3** class="panel-title"**>**Student Registration**</h3>**

**</div>**

**<div** class="panel-body"**>**

**<div** class="form-group"**>**

**<label** for="firstName"**>**First Name**</label>**

**<input** id="firstName" type="text" class="form-control"

name="firstName" ngModel**>**

**</div>**

**<div** class="form-group"**>**

**<label** for="lastName"**>**Last Name**</label>**

**<input** id="lastName" type="text" class="form-control"

name="lastName" ngModel**>**

**</div>**

**<div** class="form-group"**>**

**<label** for="email"**>**Email**</label>**

**<input** id="email" type="text" class="form-control"

name="email" ngModel**>**

**</div>**

**<div** class="form-group"**>**

**<label>**Gender**</label>**

**<div** class="form-control"**>**

**<label** class="radio-inline"**>**

**<input** type="radio" name="gender" value="male" ngModel**>**

Male

**</label>**

**<label** class="radio-inline"**>**

**<input** type="radio" name="gender" value="female" ngModel**>**

Female

**</label>**

**</div>**

**</div>**

**<div** class="form-group"**>**

**<label** for="branch"**>**Branch**</label>**

**<select** id="branch" name="branch" ngModel class="form-control"**>**

**<option** value="1"**>**CSE**</option>**

**<option** value="2"**>**ETC**</option>**

**<option** value="3"**>**Mechanical**</option>**

**<option** value="4"**>**Electrical**</option>**

**</select>**

**</div>**

**<div** class="form-group"**>**

**<div** class="form-control"**>**

**<label** class="checkbox-inline"**>**

**<input** type="checkbox" name="isAccept" ngModel**>**Accept Terms & Conditions

**</label>**

**</div>**

**</div>**

**</div>**

**<div** class="panel-footer"**>**

**<button** class="btn btn-primary" type="submit"**>**Submit**</button>**

**</div>**

**</div>**

**</form>**

**</div>**

**</div>**

**</div>**

**Modifying app.component.ts file:**

We want to log the posted form values into the console tab. So, modify the **app.component.ts**file as shown below.

**import** **{** Component **}** from '@angular/core';

**import** **{** NgForm **}** from '@angular/forms'

@Component**({**

selector: 'app-root',

templateUrl: './app.component.html',

styleUrls: **[**'./app.component.css'**]**

**})**

**export** **class** AppComponent **{**

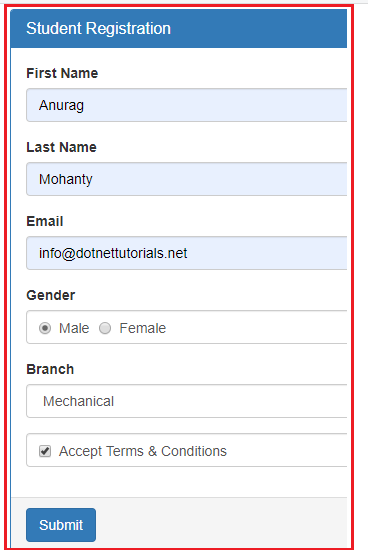
RegisterStudent**(**studentForm: NgForm**)**: **void** **{**

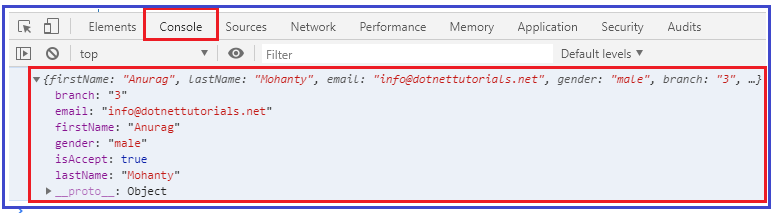
console.log**(**studentForm.value**)**;

**}**

**}**

Save the changes and browse the application, then open browser developers tool by pressing F12 key and click on the console tab. Fill the form, select one value from the drop down list and click on the submit button and you should see the posted form values in the console tab as shown in the below image.

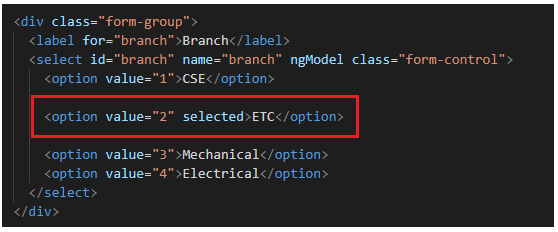




**How to have one of the dropdownlist item selected by default in Angular?**

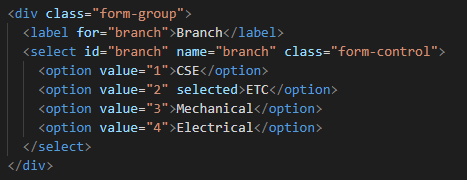
As we know when working with real-time applications, many a times we need to provide one option to be selected in the drop down list by default when the form load. And we normally do this by adding the **selected**attribute on one of the option of drop down list.

If we include the **selected**attribute on the **dropdownlist**, then we may expect that option or item to be selected by default. But in angular template driven forms, that will not work. Lets include the “**selected**” attribute on the ETC branch option to verify this. So. Modify the Drop Down List HTML code as shown below.



With the above changes in place, now browse the application and you will see that the ETC department is not selected by default when the page load.

However, if you remove the “ngModel” directive from the select list as shown below, then you will see that the ETC branch is selected when the form is load.



As we already discussed, we use the “ngModel” directive in angular for two-way data binding. So when we put the ngModel directive back into the control then the “**selected**” attribute will not work on the drop down list or select list. If we remove the ngModel directive from the control then **selected**attribute work but two way data binding will not work.

**How to make it works?**

In order to make it work, what we need to do is, we need to add a property lets say “BranchId” in the component class and initialize its value with the branch value that you want to be selected when the page load. As we want ETC branch to be selected by default and as its value is 1, so, we need to BranchId property and initialize its value to 2 in the AppComponent class. So, modify the **app.component.ts** file as shown below.

**import** **{** Component **}** from '@angular/core';

**import** **{** NgForm **}** from '@angular/forms'

@Component**({**

selector: 'app-root',

templateUrl: './app.component.html',

styleUrls: **[**'./app.component.css'**]**

**})**

**export** **class** AppComponent **{**

BranchId = "2";

RegisterStudent**(**studentForm: NgForm**)**: **void** **{**

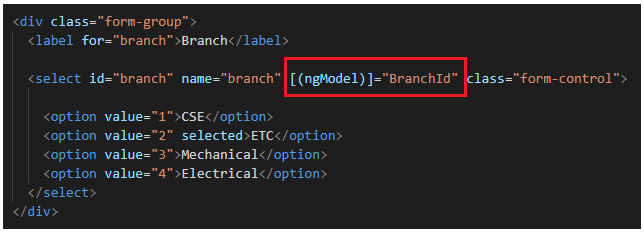
console.log**(**studentForm.value**)**;

**}**

**}**

**Modifying the app.component.html file**

Next, we include to include the ngModel directive and bind it with the component property BranchId. To do so, modify the Select List code in the **app.component.html** file as shown below.



With the above changes in place, now if you browse the application, then you should see the ETC branch is selected by default in the Branch Drop Down List when the form loads .

Now, even if you remove the “selected” attribute from the Option, then also it will work i.e. it will select the ETC by default. This is possible because of the two-way data binding which is provided by angular. In our example, we do not want the ETC to be selected by default, so we remove the “selected” attribute and the “BranchId” property from the component class and ngModel directive.

**How to disable a Drop Down List in Angular Template Driven Forms?**

In order to disable a drop down list in Angular Template Driven Form, we need use the disabled attribute on the select element as shown below.

**<select id=”branch” name=”branch” class=”form-control” ngModel disabled>**

With the above changes, now it is not possible to select any item from the drop down list. As we already discussed, by default, the disabled form controls are not included in the Angular auto generated form model.

In our example, we do not want the dropdownlist to be disabled, so please remove the disabled attribute from the select element.

**How to get the select list options from the component class?**

As of now we have hard-coded the select list options in HTML itself. In most of the real-time applications, you will get this data from a database. So,

modify the **app.component.ts** file as shown below. Here, we created one property called **Branches**which will return the list of items that we want to show in the drop down list. It has two properties id and name.

**import** **{** Component **}** from '@angular/core';

**import** **{** NgForm **}** from '@angular/forms'

@Component**({**

selector: 'app-root',

templateUrl: './app.component.html',

styleUrls: **[**'./app.component.css'**]**

**})**

**export** **class** AppComponent **{**

RegisterStudent**(**studentForm: NgForm**)**: **void** **{**

console.log**(**studentForm.value**)**;

**}**

Branches: **any[]** = **[**

**{** id: 1, name: 'CSE' **}**,

**{** id: 2, name: 'ETC' **}**,

**{** id: 3, name: 'Mechanical' **}**,

**{** id: 4, name: 'Electrical' **}**

**]**;

**}**

**Modifying the app.component.html file:**

Next we need to modify the drop down list code in the app.component.html file as shown below.

