**Lesson01 Custom Required Directive**

**Steps:-**

**1-on the custom Directive we declare the following code**

import { Directive, Self, Input } from '@angular/core';

import { NgControl, Validators } from '@angular/forms';

import { DynamicControlService } from '../services/dynamic-control.service';

@Directive({selector: '[shcRequired]'})

export class RequiredDirective {

//we declare Input variable that receive required parameter and set //validation for that control to be required or not

--------------------------------------------------------------------------

//we see that if we want to make only 1 input parameter we set the //selector, name the same of the input variable and use it in the html //page without needing of calling selector only call the input variable

@Input() set shcRequired(required: any) {

if (this.dcs.convertToBoolean(required) == true) {

this.ngControl.control.setValidators([Validators.required]);

this.ngControl.control.updateValueAndValidity();

} else {

this.ngControl.control.clearValidators();

this.ngControl.control.updateValueAndValidity();}}

//1-we use NgControl to make update for the control of reactive form type

//2-Self is parameter decorator to be used on constructor parameters, //which tells the DI framework to start dependency resolution from the //local injector.

constructor(@Self() private ngControl: NgControl, private dcs: DynamicControlService) {}}

**2-on the Dynamic service we set the following code**

convertToBoolean(value: any) {

if (value == 'false' || value == false) {return false;}

else if (value == 'true' || value == true) {return true;}

else if (!isNaN(value)) {return Number(value) == 1 ? true : false;}

else {return value;}}

**3-on the DropDownControl.html we apply the following code**

<div class="form-group editor-control" [ngClass]="{'col-md-3': control.width ==1 ,'col-md-6': control.width==2, 'col-md-9':control.width==3,'col-md-12':control.width==4 }">

<div [formGroup]="form" class="prime-form-control">

<span class="ui-float-label">

<div [ngSwitch]="control.multiSelect">

<p-dropdown \*ngSwitchCase="false" [options]="control.options" [panelStyle]="{minWidth:'12em'}"

[formControlName]="control.key"[name]="" [required]="control.required"

//we see that we pass only the input variable that represent also the //selector name on the p-dropdown control

[shcRequired]="control.required" [filter]="control.filter" [appDisable]="control.disabled" class="form-control-ng"></p-dropdown>

<p-multiSelect \*ngSwitchCase="true" [options]="control.options" [formControlName]="control.key" [panelStyle]="{minWidth:'12em'}"

[appDisable]="control.disabled" [required]="control.required"

//we see that we pass only the input variable that represent also the //selector name on the p-dropdown control

[shcRequired]="control.required" [styleClass]=""></p-multiSelect>

</div>

<label [attr.for]="control.key" class="component-label">{{control.name}}</label>

</span>

</div>

</div>