

Form Validation



```
<form>  
  <input type="text">  
  <input type="textarea">  
  <input type="checkbox">  
  ...  
  <input type="submit">  
</form>
```



Step 1: Create a form in HTML with name attributes

```
<form name='formName'>  
  <input type="text" name="name">  
  <button>Submit</button>  
</form>
```

Step 2: Bind Input with ng-model

```
<form name='formName' >
  <input type="text" name="name"
    ng-model="ctrl.name">
  <button ng-click="ctrl.go()" >
    Submit
  </button>
</form>
```

Step 3: Declare HTML5 Validation Attributes

```
<form name='formName' novalidate >
  <input type="text" name="name"
    ng-model="ctrl.name"
    required
    min-length="4">
  <button ng-click="ctrl.go()" >
    Submit
  </button>
</form>
```



Step 4: Use Angular Form Bound Objects

```
<input type="text" name="name"  
      ng-model="ctrl.name"  
      required  
      min-length="4">  
<span  
  ng-if="formName.name.$error.required  
        && formName.name.$touched" >  
  Name is required  
</span>
```



Step 4: Use Angular Form Bound Objects

...

```
<button  
  ng-disabled="formName.$invalid"  
  ng-click="ctrl.go()">  
  Submit  
</button>  
</form>
```



Step 5: Use Angular Validation Styles

```
.ng-touched.ng-valid {  
  border: 2px green solid;  
}
```

```
.ng-touched.ng-invalid {  
  border: 2px red solid;  
}
```



Summary

- 2 Form validation in Angular is very easy!
- 2 Create a named form with named fields
 - Declare form's novalidate flag
- 2 Bind form inputs using ng-model
- 2 Specify HTML5 validations on input elements
- 2 Use form bound objects to show/hide error messages and to disable action buttons
- 2 Use Angular provided CSS classes to style difference scenarios to give more visual feedback to the user

