AngularJS Forms

[« Previous](http://www.w3schools.com/angular/angular_events.asp)

[Next Chapter »](http://www.w3schools.com/angular/angular_validation.asp)

Forms in AngularJS provides data-binding and validation of input controls.

Input Controls

Input controls are the HTML input elements:

* input elements
* select elements
* button elements
* textarea elements

Data-Binding

Input controls provides data-binding by using the ng-model directive.

<input type="text" ng-model="firstname">

The application does now have a property named firstname.

The ng-model directive binds the input controller to the rest of your application.

The property firstname, can be referred to in a controller:

Example

<script>  
var app = angular.module('myApp', []);  
app.controller('formCtrl', function($scope) {  
    $scope.firstname = "John";  
});  
</script>

[Try it Yourself »](http://www.w3schools.com/angular/tryit.asp?filename=try_ng_form_binding)

It can also be referred to elsewhere in the application:

Example

<form>  
First Name: <input type="text" ng-model="firstname">  
</form>  
  
<h1>You entered: {{firstname}}</h1>

[Try it Yourself »](http://www.w3schools.com/angular/tryit.asp?filename=try_ng_form_expression)

Checkbox

A checkbox has the value true or false. Apply the ng-model directive to a checkbox, and use it's value in your application.

Example

Show the header if the checkbox is checked:

 <form>  
    Check to show a header:  
    <input type="checkbox" ng-model="myVar">  
</form>  
  
<h1 ng-show="myVar">My Header</h1>

[Try it Yourself »](http://www.w3schools.com/angular/tryit.asp?filename=try_ng_form_checkbox)

Radiobuttons

Bind radio buttons to your application with the ng-model directive.

Radio buttons with the same ng-model can have different values, but only the selected one will be used.

Example

Display some text, based on the value of the selected radio button:

 <form>  
Pick a topic:  
<input type="radio" ng-model="myVar" value="dogs">Dogs  
<input type="radio" ng-model="myVar" value="tuts">Tutorials  
<input type="radio" ng-model="myVar" value="cars">Cars  
</form>

[Try it Yourself »](http://www.w3schools.com/angular/tryit.asp?filename=try_ng_form_radio)

The value of myVar will be either dogs, tuts, or cars.

Selectbox

Bind select boxes to your application with the ng-model directive.

The property defined in the ng-model attribute will have the value of the selected option in the selectbox.

 Example

Display some text, based on the value of the selected option:

<form>  
Select a topic:  
<select ng-model="myVar">  
    <option value="">  
    <option value="dogs">Dogs  
    <option value="tuts">Tutorials  
    <option value="cars">Cars  
</select>  
</form>

[Try it Yourself »](http://www.w3schools.com/angular/tryit.asp?filename=try_ng_form_select)

The value of myVar will be either dogs, tuts, or cars.

An AngularJS Form Example

Top of Form

First Name:  
  
Last Name:  
   
  
RESET

Bottom of Form

form = {"firstName":"John","lastName":"Doe"}

master = {"firstName":"John","lastName":"Doe"}

Application Code

<div ng-app="myApp" ng-controller="formCtrl">  
  <form novalidate>  
    First Name:<br>  
    <input type="text" ng-model="user.firstName"><br>  
    Last Name:<br>  
    <input type="text" ng-model="user.lastName">  
    <br><br>  
    <button ng-click="reset()">RESET</button>  
  </form>  
  <p>form = {{user}}</p>  
  <p>master = {{master}}</p>  
</div>  
  
<script>  
var app = angular.module('myApp', []);  
app.controller('formCtrl', function($scope) {  
    $scope.master = {firstName: "John", lastName: "Doe"};  
    $scope.reset = function() {  
        $scope.user = angular.copy($scope.master);  
    };  
    $scope.reset();  
});  
</script>

[Try it Yourself »](http://www.w3schools.com/angular/tryit.asp?filename=try_ng_form)

|  |  |
| --- | --- |
| **Note** | The **novalidate** attribute is new in HTML5. It disables any default browser validation. |

Example Explained

The **ng-app** directive defines the AngularJS application.

The **ng-controller** directive defines the application controller.

The **ng-model** directive binds two input elements to the **user** object in the model.

The **formCtrl** controller sets initial values to the **master** object, and defines the **reset()** method.

The **reset()** method sets the **user** object equal to the **master** object.

The **ng-click** directive invokes the **reset()** method, only if the button is clicked.

The novalidate attribute is not needed for this application, but normally you will use it in AngularJS forms, to override standard HTML5 validation.

[« Previous](http://www.w3schools.com/angular/angular_events.asp)

[Next Chapter](http://www.w3schools.com/angular/angular_validation.asp)