

ANGULAR.JS INPUT

Form the Form!!!

“I’m so happy I get to program form validation for our new product! This is going to be lots of fun!”

– NO DEVELOPER EVER

```
<form id="payment-form">

  Card Type:
  <select>
    <option></option>
    <option>Visa</option>
    <option>Mastercard</option>
    <option>American Express</option>
    <option>Discover</option>
  </select>

  CCN:
  <input type="text" name="ccn" />

  Expiration Date:
  <input type="text" name="expiration-month" />
  <input type="text" name="expiration-year" />

  CCV:
  <input type="text" name="ccv" />

</form>
```

```
var $paymentForm = $('#payment-form');

$paymentForm.on('submit', function(){

    var cardType = $paymentForm.children('select').val();
    var ccn = $paymentForm.children('[name="ccn"]').val();
    var expirationMonth = $paymentForm.children('[name="expiration-month"]').val();
    var expirationYear = $paymentForm.children('[name="expiration-year"]').val();
    var ccv = $paymentForm.children('[name="ccv"]').val();

});
```



```
var $paymentForm = $('#payment-form');

$paymentForm.on('submit', function(){

    var cardType = $paymentForm.children('select').val();
    var ccn = $paymentForm.children('[name="ccn"]').val();
    var expirationMonth = $paymentForm.children('[name="expiration-month"]').val();
    var expirationYear = $paymentForm.children('[name="expiration-year"]').val();
    var ccv = $paymentForm.children('[name="ccv"]').val();

    if(!cardType || !ccn || !expirationMonth || !expirationYear || !ccv) {
        return false;
    }

});
```

```
var $paymentForm = $('#payment-form');

$paymentForm.on('submit', function(){

    event.preventDefault();

    var cardType = $paymentForm.children('select').val();
    var ccn = $paymentForm.children('[name="ccn"]').val();
    var expirationMonth = $paymentForm.children('[name="expiration-month"]').val();
    var expirationYear = $paymentForm.children('[name="expiration-year"]').val();
    var ccv = $paymentForm.children('[name="ccv"]').val();

    if(!cardType || !ccn || !expirationMonth || !expirationYear || !ccv) {
        return false;
    }

});
```



```
var $paymentForm = $('#payment-form');

$paymentForm.on('submit', function(){

    event.preventDefault();

    var cardType = $paymentForm.children('select').val();
    var ccn = $paymentForm.children('[name="ccn"]').val();
    var expirationMonth = $paymentForm.children('[name="expiration-month"]').val();
    var expirationYear = $paymentForm.children('[name="expiration-year"]').val();
    var ccv = $paymentForm.children('[name="ccv"]').val();

    if(!cardType || !ccn || !expirationMonth || !expirationYear || !ccv) {
        return false;
    }

    if (!verifyCCNMatchesCardType(ccn, cardType)) {
        return false;
    }

});
```

```
var $paymentForm = $('#payment-form');

$paymentForm.on('submit', function(){

    event.preventDefault();

    var cardType = $paymentForm.children('select').val();
    var ccn = $paymentForm.children('[name="ccn"]').val();
    var expirationMonth = $paymentForm.children('[name="expiration-month"]').val();
    var expirationYear = $paymentForm.children('[name="expiration-year"]').val();
    var ccv = $paymentForm.children('[name="ccv"]').val();

    if(!cardType || !ccn || !expirationMonth || !expirationYear || !ccv) {
        return false;
    }

    if (!verifyCCNMatchesCardType(ccn, cardType)) {
        handleError('Credit card number does not match card type.')
    }

});
```



```
var $paymentForm = $('#payment-form');

$paymentForm.on('submit', function(){

    event.preventDefault();

    var cardType = $paymentForm.children('select').val();
    var ccn = $paymentForm.children('[name="ccn"]').val();
    var expirationMonth = $paymentForm.children('[name="expiration-month"]').val();
    var expirationYear = $paymentForm.children('[name="expiration-year"]').val();
    var ccv = $paymentForm.children('[name="ccv"]').val();

    if(!cardType || !ccn || !expirationMonth || !expirationYear || !ccv) {
        handleError('Something is not filled in.')
    }

    if (!verifyCCNMatchesCardType(ccn, cardType)) {
        handleError('Credit card number does not match card type.')
    }

});
```

```

var $paymentForm = $('#payment-form');

$paymentForm.on('submit', function(){

    event.preventDefault();

    var cardType = $paymentForm.children('select').val();
    var ccn = $paymentForm.children('[name="ccn"]').val();
    var expirationMonth = $paymentForm.children('[name="expiration-month"]').val();
    var expirationYear = $paymentForm.children('[name="expiration-year"]').val();
    var ccv = $paymentForm.children('[name="ccv"]').val();

    if(!cardType) {
        handleError('Card type is not filled in.')
    }

    if(!ccn) {
        handleError('Credit card number is not filled in.')
    }

    if(!expirationMonth) {
        handleError('Expiration month is not filled in.')
    }

    if(!expirationYear) {
        handleError('Expiration year is not filled in.')
    }

    if(!ccv) {
        handleError('CCV is not filled in.')
    }

    if (!verifyCCNMatchesCardType(ccn, cardType)) {
        handleError('Credit card number does not match card type.')
    }

});

```

AHHHHHHHHHHHHHHHH



**INPUT VALUING AND FORM
VALIDATION IS WROUGHT
WITH BUSINESS LOGIC AND
EXTREMELY VERBOSE MUCH
LIKE THIS SLIDE HAHA GET IT
THERE ARE MANY WORDS
HERE**



ANGULAR IS QUITE GOOD AT FORMS

- **Flexible pattern for handling input values**

ANGULAR IS QUITE GOOD AT FORMS

- **Flexible pattern for handling input values**
- **Powerful, declarative input validation inside templates**

ANGULAR IS QUITE GOOD AT FORMS

- **Flexible pattern for handling input values**
- **Powerful, declarative input validation inside templates**
- **Composing inputs together using forms**

ANGULAR IS QUITE GOOD AT FORMS

- Flexible pattern for handling input values
- Powerful, declarative input validation inside templates
- Composing inputs together using forms
- Building custom, reusable validation

ng-model

ng-model

- **Built-in attribute directive**
- **Based on type of input element attached will convey the value of that input to a target on the scope**
- **Standard way for reading input values which ultimately allows for reusability of validation logic**

```
<input type="text" ng-model="creditCardNumber" />
```

```
<select ng-model="card.type">  
  <option>Visa</option>  
  <option>Mastercard</option>  
</select>  
<input type="text" ng-model="card.number" />
```


ngModelController

BUILT-IN VALIDATION DIRECTIVES

- **ng-required** - value must not be empty
- **ng-minlength** - value must be this long or longer
- **ng-maxlength** - value must be this short or shorter
- **ng-pattern** - value must pass regular expression test
- **type="email"** - value must be a passable email
- **type="number"** - value must be numeric
- **type="url"** - value must represent a valid url

<FORM>

Technically a directive; technically super awesome

USING THE FORM DIRECTIVE

- `<form name="paymentForm"></form>`

USING THE FORM DIRECTIVE

- `<form name="paymentForm"></form>`
- Creates an `ngModelController` that is the composition of all the `ng-model` elements inside

USING THE FORM DIRECTIVE

- `<form name="paymentForm"></form>`
- Creates an `ngModelController` that is the composition of all the `ng-model` elements inside
- Allows for establishing validity of entire form based on all `ng-models`

```
<select ng-model="card.type">  
  <option>Visa</option>  
  <option>MasterCard</option>  
</select>
```

```
<input type="text" ng-model="card.number" />
```

```
<input type="text" ng-model="card.expMonth" />  
<input type="text" ng-model="card.expYear" />
```

```
<input type="text" ng-model="card.cvv" />
```

```
<button type="submit">Submit Payment</button>
```

```
<form name="paymentForm">

  <select ng-model="card.type">
    <option>Visa</option>
    <option>MasterCard</option>
  </select>

  <input type="text" ng-model="card.number" />

  <input type="text" ng-model="card.expMonth" />
  <input type="text" ng-model="card.expYear" />

  <input type="text" ng-model="card.cvv" />

  <button type="submit">Submit Payment</button>

</form>
```



```
<form name="paymentForm">

  <select ng-model="card.type">
    <option>Visa</option>
    <option>MasterCard</option>
  </select>

  <input type="text" ng-model="card.number" />

  <input type="text" ng-model="card.expMonth" />
  <input type="text" ng-model="card.expYear" />

  <input type="text" ng-model="card.cvv" />

  <button type="submit">Submit Payment</button>

</form>
```



```
<form name="paymentForm">

  <select name="cardType" ng-model="card.type">
    <option>Visa</option>
    <option>MasterCard</option>
  </select>

  <input name="cardNumber" type="text" ng-model="card.number" />

  <input name="expMonth" type="text" ng-model="card.expMonth" />
  <input name="expYear" type="text" ng-model="card.expYear" />

  <input name="cardCVV" type="text" ng-model="card.cvv" />

  <button type="submit">Submit Payment</button>

</form>
```