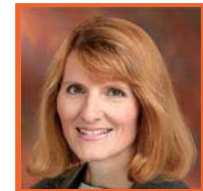


# Validating Forms

Deborah Kurata

<http://blogs.msmvps.com/deborahk/>

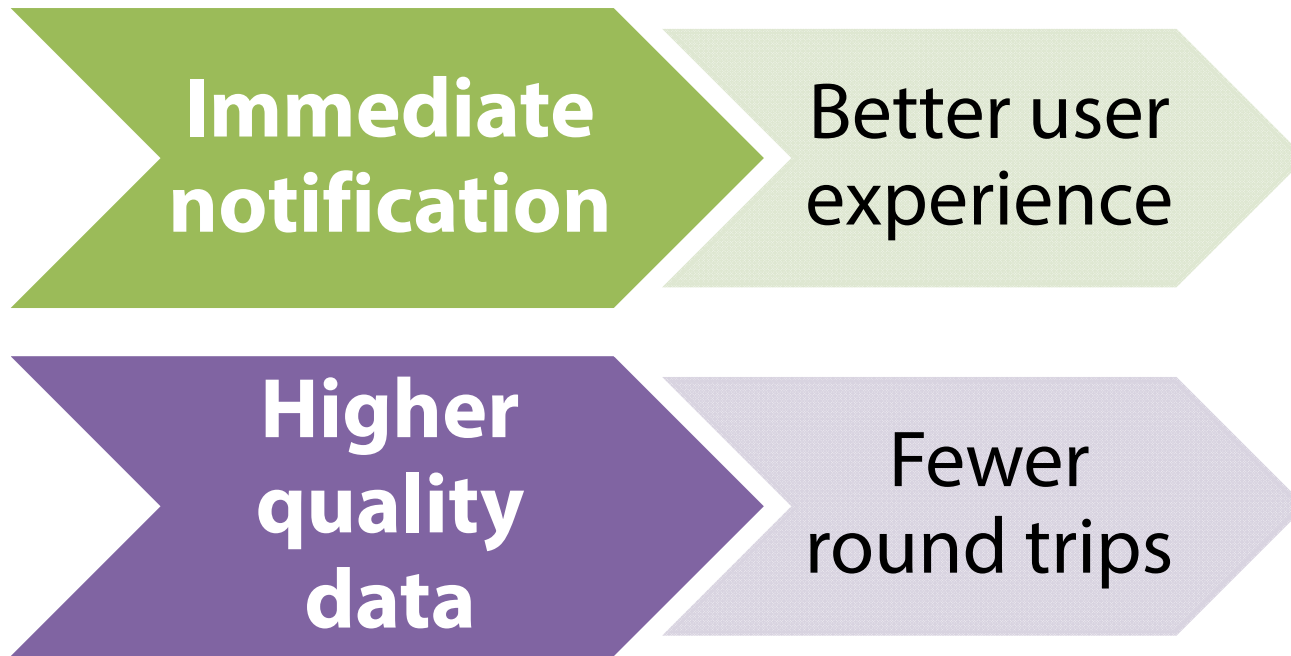
@DeborahKurata



**pluralsight**  
hardcore dev and IT training



## Benefits of Client-Side Validation



NOTE: Server-side validation is still required

# Validation Techniques

**Type Attribute**

- `type="email"`

**Required Attribute**

- `required`

**Angular Validation Attributes**

- `ng-minlength`

**Custom Angular Directive**

- `my-directive=""`

**Code in the Controller**

- `if { }`

# Module Objectives

**Prepare the Form**

**Display Messages**

**Identify Required Fields**

**Use Angular Validation  
Attributes**

**Mark Invalid Elements**

**Prevent Submit**

## Prepare the Form

**Ensure the  
form has a  
name**

**Set the  
novalidate  
attribute**

```
<form class="form-horizontal "  
    name="productForm">  
    novalidate>
```

## Marking Required Fields

```
<input class="form-control "  
      id="inputProductName"  
      type="text"  
      placeholder="Product Name"  
      ng-model="vm.product.productName"  
      required />
```

# Indicating Required Fields

\* Indicates required field

First name \*

Last name \*

\* Company:

\* Job Title:

Name  (required)

Mail  (required)

Product Name  Product Name (required)

# Marking Invalid Elements

**ngClass  
Directive**

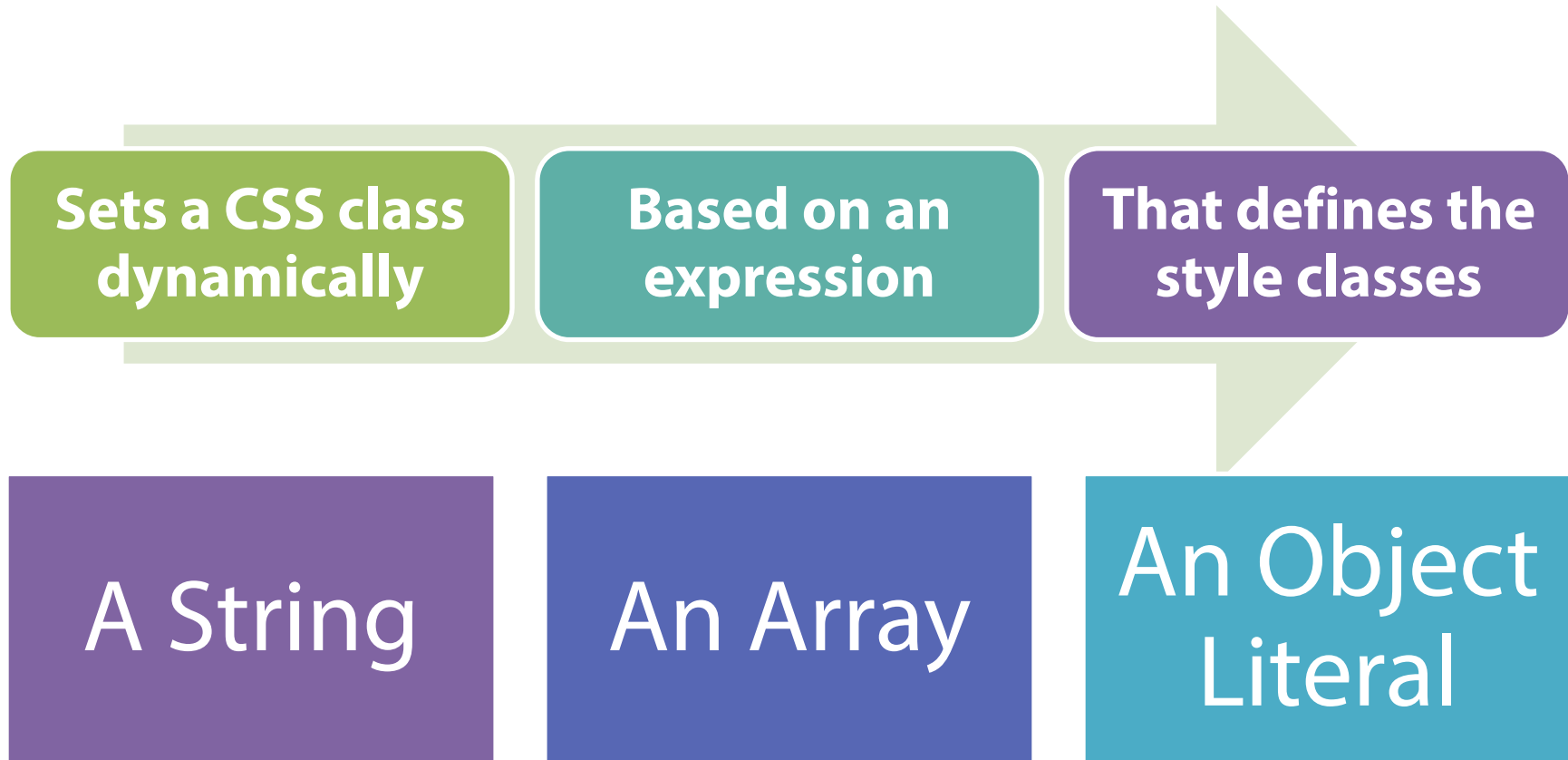
**Bootstrap  
Styles**

**Angular  
Validation  
States**

```
ng-cl ass="{ ' has-error' :  
  productForm. i nputProductName. $i nval i d} "
```



## ngClass Directive



```
ng-class="{ 'has-error' :  
  productForm.inputProductName.$invalid }"
```

# Bootstrap Validation Styles

## Validation states

Bootstrap includes validation styles for error, warning, and success states on form controls. To use, add `.has-warning`, `.has-error`, or `.has-success` to the parent element. Any `.control-label`, `.form-control`, and `.help-block` within that element will receive the validation styles.

### EXAMPLE

[Copy](#)

#### Input with success

#### Input with warning

#### Input with error

## Angular Validation States

**\$pristine**

Entry has not been changed

**\$dirty**

Entry has been changed

**\$valid**

Entry is valid

**\$invalid**

Entry is invalid

**\$error**

All validations and whether they are valid or invalid

## Using Angular Validation States

**Form  
must have a  
name**

**Input  
element  
must have a  
name**

**Input  
element  
must  
include  
ngModel**

## **Displaying Validation Messages**

**Alert Box**

**Validation  
Message  
Area**

**Adjacent to  
the Input  
Element**

## Edit: Hammer

Basic Information

Price Details

Search Tags

## Basic Product Information

**Product Name**

Product name is required.

**Product Code**

Product Code is required

**Availability****Description**

## Displaying Validation Messages

```
<span class="help-block has-error"
  ng-show="productForm.inputProductName.$invalid">
  Product name is required.
</span>
```

## Angular Validation Directives

**ng-minlength**

**ng-maxlength**

**ng-pattern**

**ng-change**



## Multiple Validation Rules

```
<input class="form-control"
      name="inputProductName"
      type="text"
      placeholder="Product Name (required)"
      ng-model="vm.product.productName"
      required
      ng-minlength="4"
      ng-maxlength="12" />
```

# Showing Different Messages for Different Errors

**Product Name**

Product name is required.

**Product Name**

Product name must be at least 4 characters in length.

**Product Name**

Product name cannot exceed 12 characters in length.

# ngMessages

## ngMessages

- Directive
- Shows or hides messages
- Similar to a switch or case statement

## ngMessage

- Directive
- Defines specific cases

## The bad news

- Not available until Angular 1.3
- The sample application is currently on 1.2

```
<span class="help-block has-error">  
  <span ng-messages="productForm.pn.$error">  
    <span ng-message="required">Required</span>  
    <span ng-message="minlength">Min of 4 chars</span>  
    <span ng-message="maxlength">Max of 10 chars</span>  
  </span>
```

## Showing Different Messages for Different Errors

```
<span class="help-block has-error">  
  <span ng-show=  
    "productForm.inputProductName.$error.required">  
    Product name is required.  
  </span>  
  <span ng-show=  
    "productForm.inputProductName.$error.minLength">  
    Must be at least 4 characters in length.  
  </span>  
  <span ng-show=  
    "productForm.inputProductName.$error.maxLength">  
    Cannot exceed 12 characters in length.  
  </span>  
</span>
```

## Preventing Form Submittal

### Disable the Save Button

```
ng-disabled="productForm.$invalid"
```

### Display a message

```
ng-click="vm.submit(productForm.$valid)"
```

# Module Objectives



**Prepare the Form**



**Display Messages**



**Identify Required Fields**



**Use Angular Validation Attributes**



**Mark Invalid Elements**



**Prevent Submit**

# Angular

## Line of Business Applications

**Business Requirements**



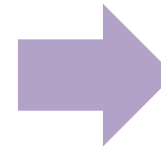
- Application Features
- Design Considerations
- Architecture

**Layout and Navigation**



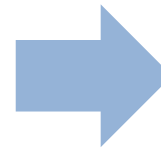
- Page Layout (View)
- Style (Bootstrap)
- Navigation (Routing)

**Data Access**



- Retrieving data
- Saving data

**Data Entry Forms**



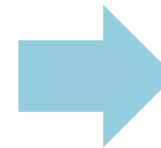
- Form layout
- Validation
- Submittal

**Business Logic**



- Services

**Data Visualization**



- Charts