Daniel Kolodziej, ITMD 419, Lab 4

DESCRIPTION:

The following lab is to demonstrate validating a form using angularjs as well as css to help convey info to the user based on the current state of the input fields within the form. The form askes the user for their first and last names, social security number(ssn), and to check whether they want to subscribe to a newsletter as well as to accept conditions. All fields are required except for the newsletter in order to let the user submit the form.

SOURCE CODE:

Html--------------------------------

<!DOCTYPE html>

<html lang="en" xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta charset="utf-8" />

<link rel="stylesheet" href="StyleSheet1.css">

<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.10/angular.min.js"></script>

<script src="Script1.js"></script>

<title>ATM</title>

</head>

<body>

<h1>Form Validation</h1>

<!--novalidate to escape default browser validate setting-->

<form ng-app="myApp" ng-controller="validateCtrl" name="myForm" novalidate>

<!--HTML5 attribute required to specify that the input field must be filled out

$dirty The field has been modified

$invalid The field content is not valid

$valid The field content is valid

They are all properties of the input field, and are either true or false.-->

<!-- required in order to for values to be entered-->

<p>First name:<br>

<input type="text" name="first" placeholder="John" ng-model="first" required>

<!-- show, in red only when firstname input has been modified and invalid-->

<span style="color:red" ng-show="myForm.first.$dirty && myForm.first.$invalid">

<span ng-show="myForm.first.$error.required">first name is required.</span>

</span>

</p>

<p>Last Name:<br>

<input type="text" name="last" placeholder="Doe" ng-model="last" required>

<!-- show, in red only when lastname input has been modified and invalid-->

<span style="color:red" ng-show="myForm.last.$dirty && myForm.last.$invalid">

<span ng-show="myForm.last.$error.required">last name is required.</span>

</span>

</p>

<p>SSN:<br>

<input type="number" name="ssnNum" placeholder="#########" ng-model="ssnNum" required>

<!-- show, in red only when ssn input has been modified and invalid-->

<span style="color:red" ng-show="myForm.ssnNum.$dirty && myForm.ssnNum.$invalid">

<span ng-show="myForm.ssnNum.$error.required">SSN is required.</span>

</span>

</p>

<!-- newsletter not required -->

<input name="subscribe" type="checkbox" ng-model="subscribeNews">

<label for="subscribe">Subscribe to our newsletter</label>

<!-- checkbox to accept conditions is required, show message in red if changed and false-->

<div>

<input name="acceptTC" type="checkbox" ng-model="acceptConds" required>

<label for="acceptTC">Accept Terms and Conditions</label>

<span style="color:red" ng-show="myForm.acceptTC.$dirty && !acceptConds">

<span ng-show="myForm.acceptTC.$error.required">Please accept the terms.</span>

</span>

</div>

<!--button disabled using ng-disabled when required fields are invalid-->

<!--when able to click, calls function to further check validity-->

<p>

<input type="submit" value="Submit Form" ng-click="checkForm()"

ng-disabled="myForm.first.$invalid || myForm.last.$invalid || myForm.ssnNum.$invalid || !acceptConds">

</p>

</form>

</body>

</html>

JS--------------------------------------------

var app = angular.module('myApp', []);

app.controller('validateCtrl', function($scope) {

//default values for input fields

$scope.first = '';

$scope.last = '';

$scope.ssnNum = '';

//alert to user whether the form has been submitted or is further invalid

$scope.checkForm = function(){

var x = $scope.ssnNum.toString();

var fname = isNaN($scope.first);

var lname = isNaN($scope.last);

//ssn must be 9 characters

//name cannot be numbers

if (x.length != 9){

alert("The following SSN given has invalid length");

} else if (!fname || !lname) {

alert("The following first and/or last name given is invalid");

} else {

alert("Alright, successfully submitted!");

}

};

});

CSS---------------------------------------------------

/\* changes colors of input fields depending on current state \*/

input**.ng-invalid** {

background-color: pink;

}

input**.ng-valid** {

background-color: lightgreen;

}

input**.ng-pristine** {

background-color: white;

}

/\*

ng-pristine The field has not been modified yet

ng-valid The field content is valid

ng-invalid The field content is not valid

\*/