JS FILE script.js

// Code goes here

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

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

//default values for input fields

$scope.first = '';

$scope.password = '';

//$scope.ssnNum = '';

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

$scope.checkForm = function(){

var fname = isNaN($scope.first);

var uPass = isNaN($scope.password);

//name cannot be numbers

if (!fname || !uPass) {

alert("The following user and/or password given is invalid");

} else {

alert("Alright, successfully submitted!");

}

};

$scope.logout = function(){

var fname = isNaN($scope.first);

var uPass = isNaN($scope.password);

//name cannot be numbers

if (!fname || !uPass) {

alert("The following user and/or password given is invalid");

} else {

alert("Alright, successfully submitted!");

}

};

});

Index.html

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.10/angular.min.js" data-require="angular.js@4.0.0" data-semver="4.0.0"></script>

<script src="script.ts" data-require="angular.js@4.0.0" data-semver="4.0.0"></script>

<script src="system.config.js" data-require="angular.js@4.0.0" data-semver="4.0.0"></script>

<script src="tsconfig.json" data-require="angular.js@4.0.0" data-semver="4.0.0"></script>

<link href="style.css" rel="stylesheet" />

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

</head>

<body>

<h1>D Bank's ATM machine</h1>

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

<form action="main.html" 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>Username:<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">user name is required.</span>

</span>

</p>

<p>Password<br>

<input type="text" name="password" placeholder="pass123" ng-model="password" required>

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

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

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

</span>

</p>

<!--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="Login" ng-click="checkForm()"

ng-disabled="myForm.first.$invalid || myForm.password.$invalid">

</p>

<p>

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

ng-disabled="myForm.first.$invalid || myForm.password.$invalid">

</p>

</form>

</body>

</html>

Main.html

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.10/angular.min.js" data-require="angular.js@4.0.0" data-semver="4.0.0"></script>

<script src="script.ts" data-require="angular.js@4.0.0" data-semver="4.0.0"></script>

<script src="system.config.js" data-require="angular.js@4.0.0" data-semver="4.0.0"></script>

<script src="tsconfig.json" data-require="angular.js@4.0.0" data-semver="4.0.0"></script>

<link href="style.css" rel="stylesheet" />

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

</head>

<body ng-app="myApp" ng-controller="validateCtrl">

<h1>D Bank's ATM machine</h1>

<!--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>Balance:<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">user name is required.</span>

</span>

</p>

<p>Deposit:<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">user name is required.</span>

</span>

</p>

<p>Withdraw:<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">user name is required.</span>

</span>

</p>

<p>Interest Accrued<br>

<input type="text" name="password" placeholder="pass123" ng-model="password" required>

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

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

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

</span>

</p>

<!--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="Deposit" ng-click="checkForm()"

ng-disabled="myForm.first.$invalid || myForm.password.$invalid">

</p>

<p>

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

ng-disabled="myForm.first.$invalid || myForm.password.$invalid">

</p>

<p>

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

ng-disabled="myForm.first.$invalid || myForm.password.$invalid">

</p>

<p>

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

ng-disabled="myForm.first.$invalid || myForm.password.$invalid">

</p>

</body>

</html>

Style.css

/\* Styles go here \*/

input.ng-invalid {

background-color: pink;

}

input.ng-valid {

background-color: lightgreen;

}

input.ng-pristine {

background-color: white;

}