

Lab - directives

Menu

Try to cleanup a given piece of DOM and put it in a directive You can use the following snippets in your solution

```
-- index.html, before directive --
<body ng-app="app">
  <div ng-controller="appCtrl">
    <ul class="menu">
      <li><a href="#{{item.href}}">{{ item.title }}</a></li>
    </ul>
  </div>
</body>
```

End result

the end result should look like

```
-- index.html, with your directive in place --

<body ng-app="app">
  <div ng-controller="appCtrl">
    <menu></menu>
  </div>
</body>
```

And of course render out your menu

HINT

you will need to create an element directive of type scope :false

week + day

You should render a week, i.e Monday through Sunday.. Selecting a specific day in the week should somehow indicate it being selected, changing of color or something like it.. A selected day should be indicated in top right corner.

You should also support vacation.. When pressing a button vacation all week days should list their respective day but also that it is a vacation day.

Sample code

The following code is meant to help you out

```
controller('weekController', function(){
    $scope.days = [
        { name : 'Monday', content : '' },
        { name : 'Tuesday', content : '' }
        ....
    ];

    // not set initially
    $scope.selectedDay;

    $scope.selectDay = function(day){
        // set
    }
});

<div ng-controller="weekController">
    <weekday ng-repeat="day in days"
        date="day.name"
        description="day.content"
        select="selectDay(day)" >
    </weekday>
</div>
```

Think about bindings here, what should be @, what should be &, and what should be =

HINT

all binding types need to be used. Check code samples for how to invoke a callback with a parameter.

Content property needs to be set to 'vacation' on all day items when going to vacation mode.