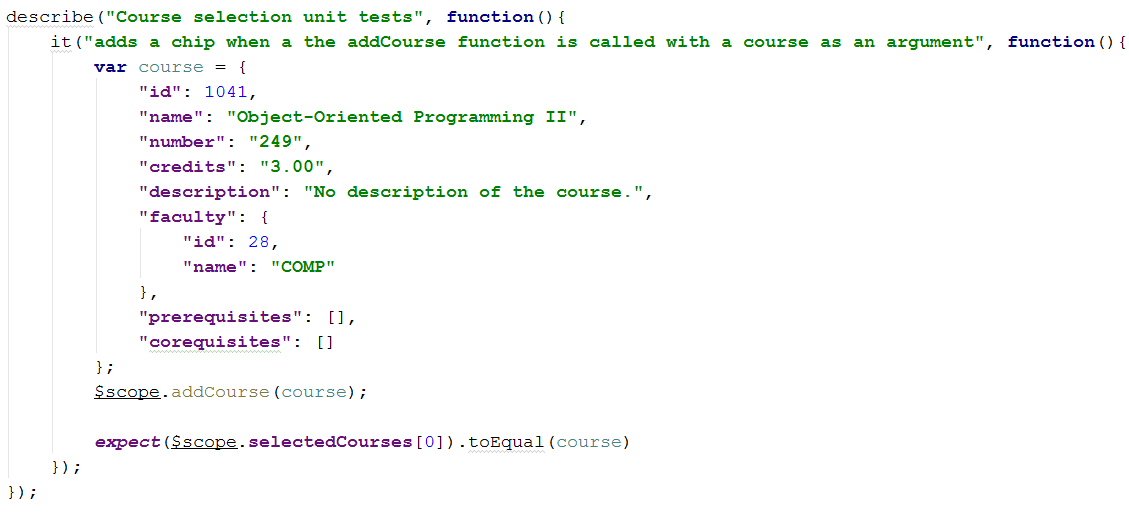
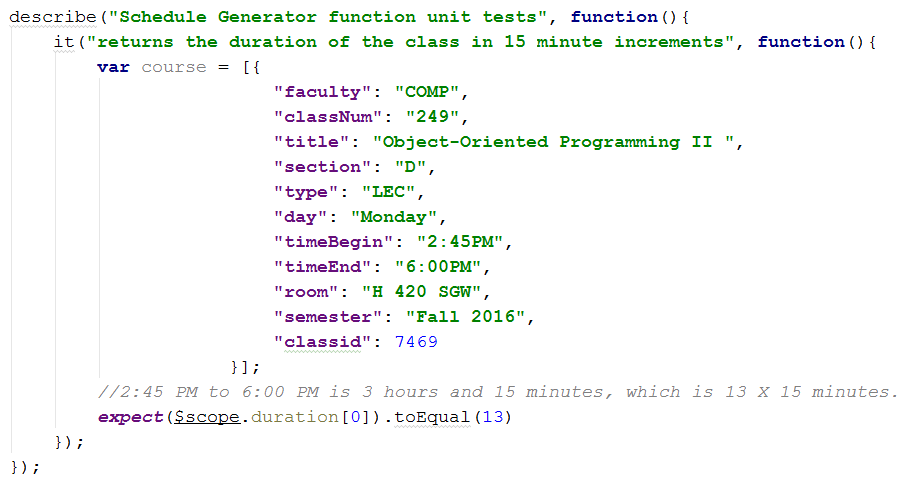
Function *addCourse(course)*

For unit testing, we test the functionality for one function at a time. Here, we provided two function form the class generated-schedule. The first function tests if the addcourse actually adds the selected course. It thus creates a chip interface is modeled by the array $scope.selectedcourses[].



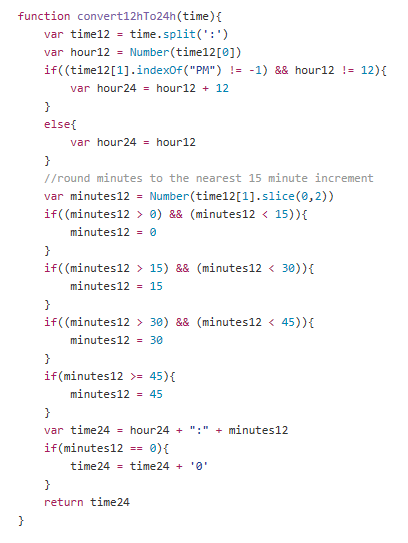
Function duration

This functions tests if the duration of the course corresponds well to an increment of 15 minutes. Since our scheduler is divided into increments of 15 minutes, a class of, for example, an hour will take 4 x 15 minutes.



In this test case, we check if a class from 2:45 PM to 6:00 PM is in fact equal to 13 x 15 minutes. This helps the *Scheduler* decide the row span of the cell, corresponding to that course.

Function *convert12hTo24h(time)*

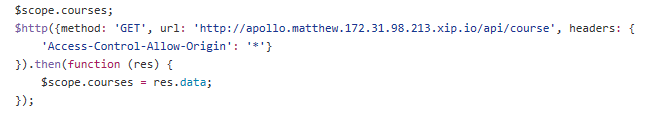


This function converts 12 hours to 24 hours. This would be useful for our course generator since, we would need it to place our courses in the right brackets of the schedule, corresponding to the right time.

Pre-Condition: enter a double number

Post-Condition: returns the time in 24 hours.

*GET course request*



This function is very important, since it gets all the data from the website and puts it into *courses*. It is a HTTP request that has the method GET to populate the data to courses.

Pre-condition: function requests from website to get data

Post-condition: *courses* gets all the data

Function *classAt (time, day)*



This function provides the scheduler with information in which the class is there at the time and day. It iterates through the array and looks if the class is there.

Post-Condition: enter the time and day in function classAt.

Pre-Condition: returns true if the class is there, returns false if it is not.