Tutorial 2: More about timers

In the second lab you started building an Appointments application. While the code you got to by the end of the lab (if you completed it) was limited, the exercises at the end suggested how you might use a timer (as discussed in last week's tutorial) to check appointments and signal if any were due. There were also some suggestions for expanding this (e.g. alerting the user an hour before the appointment was due).

The Task: Design a way for the user to say when an alert should happen.

From the suggestions in the lab sheet, you may have come up with a timer function like this:

```
function checkAlerts(){
    for(var i=0; i<appointments.length; i++){
        if(appointments[i].isDue(){
            alert(appointments[i].subject + " is due");
        }
    }
}

setInterval( checkAlerts, 60000 );          // Check once every minute.
```

Let's expand on this:

1. It would be very useful to add a willBeDue(advanceTime) function to the Appointment type. For example, you could call it as:

   ```
   if( someAppointment.willBeDue(15)){
       // do stuff
   }
   ```

   How would a 'willBeDue' function work? Look at how the isDue() function works just now, and also look into how you can do time arithmetic with the Javascript date type – how do I work out a time that is 15 minutes (or some amount of time) earlier than a given Date value?

2. Lets assume the user ought to be given the choice of how long before an appointment is due to get an alert – 15, 30, 45, 60, 120 minutes etc. How could we go about providing this function (think about how colours were listed in lab 1)?

3. We'll need to store the advance time for an alert within the Appointment type – an extra variable. I can think of two options – either store the advance time in milliSeconds (1000*60*number_of_minutes), or work out the alert time when the appointment is created and store this as a datetime value. Which is better? (there could be pros and cons to both approaches).

4. In next week's lab, we'll be adding some user-interface components to the app so that appointment details can be entered in a less clunky way that now. How would you design this user-interface (and include the alert time advance). Hint – how does your phone, PDA, or even a physical paper diary work in this respect)?