JSF 2 Exercises: Validation Part 1

1. Go to *any* previous set of exercises (other than Ajax) where you created at least one textfield. Add the required and requiredMessage attributes, and put <h:messages/> at the top of the form (but inside h:form). Test it by submitting the form with the field being empty. That's it! In 2 minutes you added simple validation to a form, with error messages at the top.

   Now, for the rest of the exercises, make a new project. Give it a form where people send their employee details in order to register for some corporate service. Your bean should have the following properties: first name, middle name, last name, employee ID, and building number. Make all of them be strings except for building number, which can be an int. Recall that Eclipse will build getters and setters from instance variables for you (see Source menu).

2. Enforce that all fields except middle name are required. Put the error messages at the top. Don't worry about the user entering an illegal value for the building number.

3. Copy the form from the previous problem, but this time display the error message next to the field that had the incorrect value. The trickiest part about this is handling the fact that you have a 3-column table, but have no potential error message for the middle name, since it is optional. One way to handle this is to use h:outputText with an empty string as a value.

4. Augment the previous two forms by handling the case when the user enters a malformed (non-numeric) value for the building number.

5. Copy the form from the previous problem, but this time enforce stricter rules:
   - First name: at least two letters
   - Middle name: optional
   - Last name: at least three letters
   - Employee ID: a lowercase letter followed by four numbers. If you know regular expressions in Java, enforce the whole pattern. If you don't, just enforce that it is exactly five characters long.
   - Building number: whole number from 1 to 50

6. Augment the previous form by putting a "Fix Problems Below" warning at the top of the form when there is at least one validation error. Note that Eclipse (at least as of latest Luna version in spring 2015) incorrectly gives a warning that the not empty test always evaluates to true. Ignore this bogus warning.