The following exercises assumes that you have studied the ShopHibernateSpring sample program. Note that ShopHibernateSpring has been updated to include a "pseudo-session" based login facility using the Java Preferences class.

In all cases, be careful about both the transaction settings and the security settings for any new use cases you write.

- Add facilities to ShopHibernateSpring to support changing a user password. The user must be logged in and can only change his or her own password. The command should take the old password and the new password as parameters and update the database to effect the change. Hint: see how the username of the logged in user is obtained from the SecurityContext in Main in order to print out the name of the logged in user.

- Add a reset forgotten password facility to ShopHibernateSpring. This should consist of two new usecases.
  1. The first allows anonymous users to request an anonymous password change token for a user. It takes a single parameter which is the email address of the user in question. The anonymous password change token is a random string (you can use the random password generator), that should be generated and emailed to the user it is intended for and saved until either the user next logs in or an anonymous user uses it to reset the user's password. In both of these cases the token should then be removed from the database.
  2. The second use case is to request a forgotten password to be reset for a user. It takes 3 parameters:
     a. the email address of the user in question
     b. the anonymous change token for the user that was obtained from the previous use case via email
     c. the new password that should be set for the user

   The use case should, if the email address and the token match, then reset the password of the user to the password passed in as an argument and remove the token from the database.

   Note that if the user successfully logs in after the previous use case was executed but before this one is invoked, then the token should also be removed from the database.

   In all cases, suitable error messages or success messages, as appropriate, should be output from Main, not from the use cases themselves, and suitable log messages should be output.