

## Exercice Authentication/Authorization

### Principe

On va reprendre l'exercice où l'on affichait la liste des personnes en utilisant les DAO de la base Bestioles.

On va ajouter une table pour servir à l'authentification. Cette table est fournie via un script SQL **sha.sql** (à adapter suivant les cas).

Cette table contient 2 utilisateurs (user@user et admin@admin).

Dans le site web, lors de la connexion, on va vérifier que le login/mot de passe est correct. Si ce n'est pas le cas, on sort en erreur.

En plus, on va gérer les autorisations : si l'utilisateur est de type admin, on a le droit de tout faire. Sinon, on affiche uniquement la liste des personnes (et donc aucun des boutons permettant de détruire, ajouter, etc ...)

### Encodage du mot de passe

Dans la base de données, les mots de passe sont encryptés en SHA-256.

Le code Java pour gérer l'encodage est fourni (**Encoder.java**).

### DAO

Au niveau des DAO, on va ajouter une entity de type User et un DAO. Ce DAO va utiliser une seule fonction :

- Qui vérifie le login et mot de passe passés en paramètres sont corrects (le mot de passe est **déjà encrypté**)
- Qui renvoie le User correspondant, null sinon

**ATTENTION** : il n'y a aucun lien dans cet exercice entre les Person et les User

### Web

Au niveau du site web, on crée un service où on étend le service existant pour gérer l'authentification via le DAO. On encrypte le mot de passe dans le service avant d'appeler le DAO.

Lors de la connexion, on récupère le login et le mot de passe, et on appelle le service.

En cas d'erreur de connexion, rediriger sur la page d'erreur.

Gérer les autorisations.