

Комплексное задание MVC.

Первая часть.

Создать 4 сущности(User, Address, Role, MusicType). Обязательные поля для User – id,login, password. Плюс добавить еще 2-3 поля на ваше усмотрение. Role - обязательные поля id, roleName (USER, MANDATOR, ADMIN). В сущности Address придумать поля самостоятельно. MusicType состоит из id и typeName. MusicType является набором музыкальных жанров. Создать таблицы в БД со следующими связями Role: User (1:M), User :Address (1:1), User (M:M)MusicType. Реализовать DAO для каждой из сущностей, в которых должны находиться CRUD операции (выбрать все сущности, выбор сущности по id, редактировать сущность, удалить сущность).

Вторая часть.

Создать в проекте из первой части property файл, в котором будут находиться настройки к БД (url, password, username). Вытянуть их в приложение при помощи ResourceBundle. Также создать еще три файла для локализации, в одном будут храниться общие ресурсы, во втором ресурсы на русском языке, а в третьем на английском. Проверить корректность их работы при помощи тестов. Написать тесты на все операции DAO.

Третья часть.

Создать web – приложение. Первая страничка логина, на которой два поля ввода, и две кнопки. Поля “Login”, “Password”. Кнопки “Войти”, “Зарегистрироваться”. При нажатии на кнопку зарегистрироваться попадаем на форму, которую нужно заполнить логином, паролем и повторным введением пароля, и всеми данными, которые имеются у нас в моделях (кроме роли), при submit форма отсылается на сервер, где проверяется, что пользователем с таким логином не существует, и пароли одинаковы. При удачной регистрации пользователь должен попасть на страничку, где будет написано «Hello <имя_пользователя>». При неудачной регистрации должно появиться сообщение о том что “такой логин уже существует”, или же не правильно повторили пароли.

В системе будет существовать три вида пользователей (USER, MANDATOR, ADMIN).

USER при удачном входе попадает на страничку, где написано «Hello <имя_пользователя>», и он не должен попадать на страницы предназначенные для MANDATOR и ADMIN.

MANDATOR попадает на страничку, где отображены все данные о всех пользователях в таблице.

ADMIN попадает на страничку, где отображены все данные о всех пользователях в таблице + у него есть возможность добавлять новых, удалять и редактировать пользователей. Кнопки удаления и редактирования должны стоять в конце каждой строки таблицы.