

Скачайте проект с

<https://assets.codeforces.com/files/695354fbbfd96ca7/96fb44125943cac7.7z> Запустите проект (поправьте src/main/resources/application.properties). Используйте базу данных вида uXX. Запуск создаст таблицу user. Добавьте к ней колонку passwordSha, которая по-умолчанию NULL. Убедитесь, что всё работает - регистрация+вход+выход, просмотр пользователей.

## Задание 1

Сделайте небольшую страницу с профилем пользователя по ссылке /user/{id}. Для этого используйте параметр в методе с аннотацией @PathVariable (погуглите, это просто). Просто отобразите id, login и creation на отдельной странице (или надпись “No such user”, если такого нет).

## Задание 2

Добавьте сущность простейшую Notice с тремя полями id, content (используйте аннотацию @Lob), creationTime. Обратите внимание, что Notice не имеет связи с User (пока всё делаем без связей). Добавьте NoticeRepository, NoticeService. В классе Page аналогично с getUser сделайте getNotices (не забудьте аннотацию @ModelAttribute), чтобы все потомки Page добавляли в модель переменную notices. Измените commons.ftlh, чтобы в сайдбаре вместо блока “Pay Attention” была серия подобных блоков, которые отображают все notices. Добавьте несколько notices через базу, убедитесь, что на всех страницах теперь отображаются ваши notices в сайдбаре.

## Задание 3

Добавьте форму добавления Notice. Это должна быть просто одна textarea с кнопкой “Add”.

## Задание 4

Добавьте пользователям колонку/поле disabled, по умолчанию равно false. На страницу /users добавьте колонку с действиями. Там в каждой строчке будет ровно одна кнопка (Disable или Enable в зависимости от текущего статуса). При нажатии кнопки отправляется маленькая форма, которая меняет статус пользователя. После этого страница перегружается и там уже кнопка для обновлённого статуса. Пока не используйте javascript, просто сделайте такие небольшие формы в каждой строке. Добавьте в форму входа сообщение “User is disabled” при попытке входа заблокированного пользователя.