Ex15: Тесты на метод user

В соответствующем классе TestUserRegister, который мы создали на уроке,

необходимо написать больше тестов на создающий пользователя POST-метод: https://playground.learnqa.ru/api/user/

Список тестов:

- Создание пользователя с некорректным email - без символа @

- Создание пользователя без указания одного из полей - с помощью @parametrize необходимо проверить, что отсутствие любого параметра не дает зарегистрировать пользователя

- Создание пользователя с очень коротким именем в один символ

- Создание пользователя с очень длинным именем - длиннее 250 символов

========================================

Результатом работы должна быть ссылка на файл в вашем репозитории, где реализованы данные тесты.

Ex16: Запрос данных другого пользователя

На занятиях в классе TestUserGet мы писали тест на запрос, показывающий данные пользователя. Мы покрыли тестами два кейса:

- неавторизованный запрос на данные - там мы получили только username

- авторизованный запрос - мы были авторизованы пользователем с ID 2 и делали запрос для получения данных того же пользователя,

в этом случае мы получали все поля

В этой задаче нужно написать тест, который авторизовывается одним пользователем, но получает данные другого (т.е. с другим ID).

И убедиться, что в этом случае запрос также получает только username, так как мы не должны видеть остальные данные чужого пользователя.

===================================================

Результатом работы должна быть ссылка на файл в вашем репозитории, где реализованы данные тесты.

Ex17: Негативные тесты на PUT

На занятиях мы написали только позитивный тест на PUT-метод редактирования пользователя.

Давайте напишем несколько негативных:

- Попытаемся изменить данные пользователя, будучи неавторизованными

- Попытаемся изменить данные пользователя, будучи авторизованными другим пользователем

- Попытаемся изменить email пользователя, будучи авторизованными тем же пользователем, на новый email без символа @

- Попытаемся изменить firstName пользователя, будучи авторизованными тем же пользователем, на очень короткое значение в один символ

=====================

Как всегда, в ответ мы ждем от вас ссылку на тесты.

Ex18: Тесты на DELETE

У нас есть метод, который удаляет пользователя по ID - DELETE-метод https://playground.learnqa.ru/api/user/{id}

Само собой, удалить можно только того пользователя, из-под которого вы авторизованы.

Необходимо в директории tests/ создать новый файл test\_user\_delete.py с классом TestUserDelete.

Там написать следующие тесты.

Первый - на попытку удалить пользователя по ID 2. Его данные для авторизации:

        data = {

            'email': 'vinkotov@example.com',

            'password': '1234'

        }

Убедиться, что система не даст вам удалить этого пользователя.

Второй - позитивный. Создать пользователя, авторизоваться из-под него, удалить, затем попробовать получить его данные по ID и убедиться, что пользователь действительно удален.

Третий - негативный, попробовать удалить пользователя, будучи авторизованными другим пользователем.

==========================================

Как всегда, в ответе ждем ссылку на тест.

Ex19: Теги Allure

Давайте добавим больше Allure-тегов во все написанные нами тесты.

Выбирайте любые, которые понравятся из официальной документации - https://docs.qameta.io/allure/

Цель задания - поэкспериментировать с бОльшим количеством тегов, узнать, где именно в отчете они отображаются.

В этом задании не будет правильных и неправильных результатов.

Но все же нам очень интересно посмотреть, какие теги вы сочтете особенно полезными для вашего фреймворка.

==================================================

Результат - ссылка на ваш репозиторий с тестами и список тегов с описанием, почему именно они показались вам полезными для тестов.