

Тестовое задание




Необходимо написать парсер, который умеет разбирать странички какого-нибудь сайт (можете выбрать любой интернет-магазин).

У данного парсера должен быть API для получения следующей информации:




Серверная часть




Сущности

User → Пользователь

 Название поля	 Тип	 Комментарий
<u>Идентификатор</u>	Целое число	
<u>Фамилия</u>	Строка	
<u>Имя</u>	Строка	
<u>Дата создания</u>	Дата	
<u>Дата последнего изменения</u>	Дата	
<u>Статус</u>	Перечисление	Возможные значения → Активен, Отключен, Заблокирован

Task → Задача

 Название поля	 Тип	 Комментарий
<u>Идентификатор</u>	Целое число	
<u>Название</u>	Строка	
<u>Описание</u>	Строка	
<u>Дата создания</u>	Дата	
<u>Дата последнего изменения</u>	Дата	

 Название поля	 Тип	 Комментарий
<u>Статус</u>	Перечисление	Возможные значения → Не начата, В процессе, Выполнен, Отменен, Отклонен
<u>Постановщик</u>	Пользователь	
<u>Исполнитель</u>	Пользователь	

База данных

Для работы с базой данных использовать подход code first. Использовать инициализатор со свойством обновления БД при запуске приложения. При запуске заполнять в БД следующие данные, используя автогенерацию или статические данные:

1. 5 пользователей;
2. 24 задачи.

DAL и сервисный слой

Тут вам нужно продемонстрировать свои знания и умения.

API

Необходимо реализовать следующие методы.

Пользователи

Должны быть следующие методы:

- получение списка (с пагинацией);
- получение одного пользователя;
- редактирование данных (кроме статуса);
- поставить задачу исполнителю.

Задачи

Должны быть следующие методы:

- получение списка задач пользователя-создателя (с пагинацией);
- получение списка задач пользователя-исполнителя (с пагинацией);

- получение одной задачи;
- редактирование данных (кроме статуса и создателя);
- изменение статуса задачи;
- смена постановщика.

Прочие требования

Обязательные → Web API, BL, DAL, MS Sql/Postgres, EF, DI.

Будет плюсом:

1. `Swagger`.
2. `Docker` (если `.Net Core`).
3. Паттерны проектирования.