

Тестовое задание Т3. С#

Оценочное время на выполнение задания: 2-3 часа. Общее время на выполнение задания: 7 календарных дней с момента выдачи. **Сделать требуется только одно любое задание из приложенных в архив на выбор.**

Результат выполнения прислать в чат Skype, в котором было выдано задание. Исходный код без бинарных файлов архивом или ссылкой на публичный репозиторий GitHub/GitLab/т.д.

Дан **record**, описывающий транзакцию:

```
public record Transaction
{
    public required Guid Id { get; init; }
    public required DateTime TransactionDate { get; init; }
    public required decimal Amount { get; init; }
}
```

Напишите Web API, позволяющее создавать и сохранять транзакции и получать ранее созданные транзакции по Id. В сервисе должно быть 2 запроса:

```
POST /api/v1/Transaction
{
    "id": "1afa615f-af61-4d8a-b891-bc874c937772",
    "transactionDate": "2024-10-25T00:00:00+05:00",
    "amount": 12.34
}
Response 200 OK:
{
    "insertDateTime": "2024-10-25T12:03:34+05:00"
}
```

```
GET /api/v1/Transaction?id=1afa615f-af61-4d8a-b891-bc874c937772
Response 200 OK:
{
    "id": "1afa615f-af61-4d8a-b891-bc874c937772",
    "transactionDate": "2024-10-25T00:00:00+05:00",
    "amount": 12.34
}
```

- Метод POST должен быть идемпотентным: при повторной отправке запроса он должен возвращать результат отправки предыдущего запроса с таким же Id.
- Необходимо реализовать валидацию: сумма в транзакциях должна быть всегда положительной, дата не может быть в будущем.

- Обработку ошибок необходимо реализовать в соответствии с RFC 9457.
- Способ хранения данных - на усмотрение соискателя. **Ограничение на количество одновременно хранимых транзакций: 100 штук.**
- Способ логирования - на усмотрения соискателя.
- Дополнительная функциональность - на усмотрение соискателя.
- Приветствуется проектирование решения с залогом на поддержку и развитие сервиса.