Тестовое задание

1. Необходимо развернуть локально MYSQL Server и создать базу данных TEST
   1. логин: kursUser пароль: kursPswd
   2. создать таблицу R\_CURRENCY (структура таблицы описана ниже)
2. Необходимо создать web сервис Golang & gorilla/mux, который будет по запросу забирать данные из публичного API национального банка и сохранять данные в локальную базу TEST
   1. сервис должен слушать порт, указанный в config.json
   2. подключение (Connection String) к MYSQL Server необходимо хранить в config.json
   3. в сервисе должны быть реализованы 2 GET метода

**1 метод:** API [.../currency/save/{date}](url://.../currency/save/{date}) , где {date} – часть адреса, а не параметр запроса

параметр date необходимо передать в API нац. банка, пример: https://nationalbank.kz/rss/get\_rates.cfm?fdate=15.04.2021

Пример структуры XML с необходимыми полями

<rates>

...

<date>15.03.2019</date>

<item>

<fullname>АВСТРАЛИЙСКИЙ ДОЛЛАР</fullname>

<title>AUD</title>

<description>267.39</description>

...

</item>

<item>

<fullname>АЗЕРБАЙДЖАНСКИЙ МАНАТ</fullname>

<title>AZN</title>

<description>223.27</description>

...

</item>

и далее все item’ы

</rates>

После того, как необходимые данные будут получены, их необходимо сохранить в таблицу R\_CURRENCY, где A\_DATE равна параметру date из XML. Сохранение делать асинхронно, в горутине. Отдать ответ пользователю, не дожидаясь завершения. Если возникнет ошибка при сохранении, записать в логи.

Успешным ответом данного метода является JSON объект с полем success = true.

Неуспешный ответ данного метода придумать самостоятельно.

**2 метод** API [.../currency/{date}/{\*code}](url://.../currency/{date}/{*code}) принимает два параметра date и code, где code является необязательным параметром

В методе необходимо вытащить данные с указанными параметрами из БД и вернуть пользователю.

Успешным ответом данного метода является массив данных в формате JSON

Неуспешный ответ данного метода придумать самостоятельно (нельзя показывать пользователю реальную ошибку из БД)

Структура таблицы R\_CURRENCY

* ID – int [PK, автоинкремент]
* TITLE – varchar(60) [описание – наименование валюты]
* CODE – varchar(3) [описание – код валюты]
* VALUE – numeric(18,2) [описание – значение]
* A\_DATE – date [описание – на дату выгрузки]

Все поля таблицы должны запрещать вставку NULL значения

Исходный код отправить в ответном письме