

## A) Реализовать приложение на golang

---

### Методы:

#### 1. localhost:8080/update

Импорт / обновление *необходимых* данных из <https://www.treasury.gov/ofac/downloads/sdn.xml> в локальную базу MariaDb. В базу должны попадать только sdnEntry по людям.

##### Результат:

###### success:

```
{ "result": true, "info": "", "code": 200 }
```

###### fail:

```
{ "result": false, "info": "service unavailable", "code": 503 }
```

#### 2. localhost:8080/state

Получение текущего состояния данных

##### нет данных:

```
{ "result": false, "info": "empty" }
```

##### в процессе обновления:

```
{ "result": false, "info": "updating" }
```

##### данные готовы к использованию:

```
{ "result": true, "info": "ok" }
```

#### 3. localhost:8080/get\_names?name={SOME\_VALUE}&type={strong|weak}

Получение списка всех возможных имён человека из локальной базы данных с указанием основного uid в виде JSON.

Если параметр type не указан / указан ошибочно, то выдаём список состоящий из всех типов. Параметр type независим от регистра.

**Запрос:** localhost:8080/get\_names?name=MUZONZINI&type=strong

##### Результат:

```
[{uid:7535, first_name:"Elisha", last_name:"Muzonzini"}]
```

**Запрос:** localhost:8080/get\_names?name=Elisha Muzonzini

##### Результат:

```
[{uid:7535, first_name:"Elisha", last_name:"Muzonzini"}]
```

**Запрос:** localhost:8080/get\_names?name=Mohammed Musa&type=weak

##### Результат:

```
[{uid:15582, first_name:"Musa", last_name:"Kalim"}, {uid:15582, first_name:"Barich", last_name:"Musa Kalim"}, {uid:15582, first_name:"Mohammed Musa", last_name:"Kalim"}, {uid:15582, first_name:"Musa Khalim", last_name:"Alizari"}, {uid:15582, first_name:"Qualem", last_name:"Musa"}, {uid:15582, first_name:"Qualim", last_name:"Musa"}, {uid:15582, first_name:"Khaleem", last_name:"Musa"}, {uid:15582, first_name:"Kaleem", last_name:"Musa"}]
```

## B) Написать инструкции Docker Compose для разворачивания реализованного приложения на порту 8080 с использованием Postgresql14.

---

## C) Описать алгоритм для более эффективного обновления данных при повторном вызове метода localhost:8080/update ( можно реализовать, но не обязательно )