

Тестовое задание (Разработчик Java)

Необходимо написать код для получения и агрегации данных из нескольких сервисов.

Получение списка доступных видеокамер:

<https://run.mocky.io/v3/bc34ce01-90c6-4266-93f1-07591afad12e>

Ответ состоит из массива объектов, содержащих поля:

- `id` - число, идентификатор камеры
- `sourceDataUrl` - строка, ссылка для получения данных источника.
- `tokenDataUrl` - строка, ссылка для получения токенов безопасности по камере.

Формат данных в ответе на запрос на URL из поля `sourceDataUrl`:

- `urlType` - строка, тип ссылки на видеопоток. Возможные значения: "LIVE", "ARCHIVE"
- `videoUrl` - строка, ссылка на видеопоток

Формат данных в ответе на запрос на URL из поля `tokenDataUrl`:

- `value` - строка, токен безопасности
- `ttl` - число, время жизни токена

Необходимо агрегировать данные по каждой камере. Ожидаемый результат:

```
[
  {
    "id": 1,
    "urlType": "LIVE",
    "videoUrl": "rtsp://127.0.0.1/1",
    "value": "fa4b588e-249b-11e9-ab14-d663bd873d93",
    "ttl": 120
  },
  {
    "id": 3,
    "urlType": "ARCHIVE",
    "videoUrl": "rtsp://127.0.0.1/3",
    "value": "fa4b5d52-249b-11e9-ab14-d663bd873d93",
    "ttl": 120
  },
  {
    "id": 20,
    "urlType": "LIVE",
```

```
    "videoUrl": "rtsp://127.0.0.1/20",
    "value": "fa4b5f64-249b-11e9-ab14-d663bd873d93",
    "ttl": 180
  },
  {
    "id": 2,
    "urlType": "ARCHIVE",
    "videoUrl": "rtsp://127.0.0.1/2",
    "value": "fa4b5b22-249b-11e9-ab14-d663bd873d93",
    "ttl": 60
  }
]
```

Решение должно быть представлено в виде веб-сервиса. Результат нужно вернуть в ответе на HTTP-запрос к этому сервису.

При написании кода надо учитывать потенциально большие объемы данных, то есть сбор и агрегация должны выполняться в несколько потоков и как можно меньше блокироваться (на операциях I/O или ожидании данных другого запроса)

Решение должно быть опубликовано на гитхабе и подкреплено юнит-тестами.