

Сервис подписки

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

В Яндекс Маркете разрабатывают новый сервис для хранения информации про товарные предложения. В MVP планируется поддержать две основные механики: обработка запросов на частичное обновление товарных предложений в базе и отправка обновленных данных в сервисы-подписчики.

Товарное предложение в базе описывается следующей JSON схемой:

```
{
  "$id": "offer.schema.json",
  "type": "object",
  "properties": {
    "id": {
      "type": "string",
      "description": "Offer identifier, only alphabetical symbols are allowed"
    },
    "price": {
      "type": "integer",
      "description": "Offer price, value in range from 0 to 10^9"
    },
    "stock_count": {
      "type": "integer",
      "description": "Items left on stocks, value in range from 0 to 10^9"
    },
    "partner_content": {
      "type": "object",
      "properties": {
        "title": {
          "type": "string",
          "description": "Offer title filled in by the partner"
        },
        "description": {
          "type": "string",
          "description": "Offer description filled in by the partner"
        }
      }
    }
  },
  "required": [
    "id"
  ]
}
```

При межсервисном взаимодействии в товарному предложению добавляется контекст, который содержит идентификатор для сквозной трассировки, его схема:

```
{
  "$id": "message.schema.json",
  "type": "object",
  "properties": {
    "trace_id": {
      "type": "string"
    }
  },
}
```

```

    "offer": {
      "$ref": "offer.schema.json"
    }
  },
  "required": [
    "trace_id",
    "offer"
  ]
}

```

Сервис, который отправляет запрос на обновление товарного предложения, обязательно заполняет идентификатор оффера(поле *offer.id*) и идентификатор для трассировки(поле *trace_id*). Все остальные поля в запросе опциональны. В таком случае при применении обновления будет происходить слияние полей. Например, в базе у оффера заполнены поля *price* = 9990, и приходит обновление *stock_count* = 100, тогда в базе будут сохранены оба поля *price* = 9990; *stock_count* = 100. Гарантируется, что все входящие запросы валидны и соответствуют схеме. Так как это прототип, удаление товаров из базы и очищение полей было решено не поддерживать.

Помимо хранения товарных предложений в базе, в сервисе необходима функция доставки обновлений в сервисы-подписчиков. Одна подписка включает в себя два набора полей *trigger* и *shipment*, не обязательно листовых. При изменении любого поля либо вложенного поля из множества *trigger*, в подписчика отправляется сообщение, которое содержит поля из *trigger* и *shipment*, а также идентификаторы оффера и трассировки из запроса, который привело к событию подписки. Инициализация поля также считается за её изменение и создает событие подписки.

Формат входных данных

Первая строка входных данных содержит два целых числа *n* и *m* ($1 \leq n \leq 50, 1 \leq m \leq 10,000$) — количество сервисов подписчиков и количество запросов на обновления.

Следующие *n* строк содержат описания сервисов подписчиков: *i*-я строка содержит описание *i*-го подписчика. В начале строки задается *a_i* и *b_i* — количество *trigger* и *shipment* полей соответственно. Далее *a_i* *trigger* полей, и *b_i* *shipment* полей.

Следующие *m* строк содержат запросы на обновление, каждая строка — это валидный json, удовлетворяющий схеме *message.schema.json*.

Формат выходных данных

На каждое событие подписки выведите по одному сообщению в формате *message.schema.json*. Сообщения должны идти в том же порядке, что и обновления, которые привели к событию подписки. События в рамках одного обновления должны быть отсортированы по порядковому номеру подписчика.

Пример

стандартный ввод	стандартный вывод
2 5	{"trace_id": "1", "offer": {"id": "1", "price": 9990}}
2 0 price stock_count	{"trace_id": "2", "offer": {"id": "1", "price": 9990, "stock_count": 100}}
1 0 partner_content	{"trace_id": "3", "offer": {"id": "2", "partner_content": {"title": "Backpack"}}}
{"trace_id": "1", "offer": {"id": "1", "price": 9990}}	
{"trace_id": "2", "offer": {"id": "1", "stock_count": 100}}	
{"trace_id": "3", "offer": {"id": "2", "partner_content": {"title": "Backpack"}}}	
{"trace_id": "4", "offer": {"id": "1", "stock_count": 100}}	
{"trace_id": "5", "offer": {"id": "2", "partner_content": {"description": "Best backpack ever"}}	