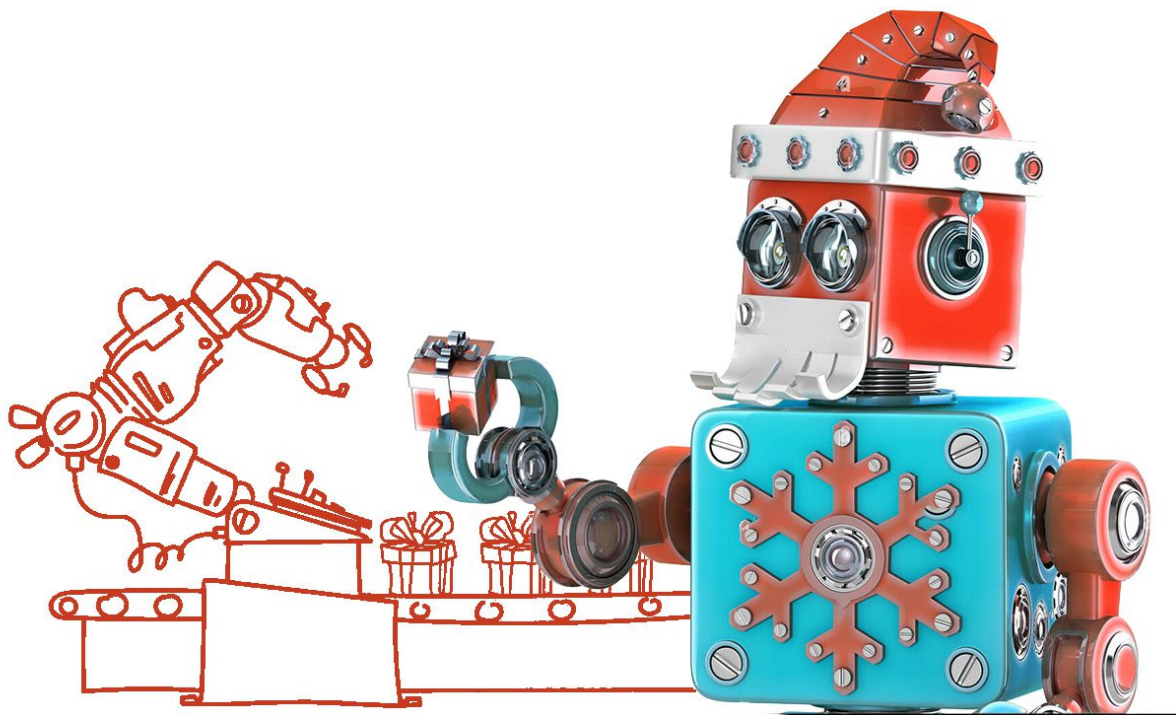


Datsanta API Этап 2 (1.0.0)

Download OpenAPI specification: [Download](#)



ОЙ!!! Оказывается, Санта еще каким-то чудом умел подбирать лучшие подарки для детей. У нас есть определённый бюджет, в размере \$100000 на 1000 детей. Но, какой подарок подходит больше ребенку, а какой меньше - требуется решить вам!

Начало этапа: 17:00 26.12.2022 .

Окончание этапа: 16:00 27.12.2022 .

Давайте разбираться, у нас есть Мальчики и Девочки в возрасте от 0 до 10 лет, которым можно подарить подарки различных категорий:

1. Конструкторы [constructors]
2. Куклы [dolls]
3. Радиоуправляемые игрушки [radio_controlled_toys]
4. Игрушечный транспорт [toy_vehicles]
5. Настольные игры [board_games]
6. Подвижные игры [outdoor_games]
7. Игровая площадка [playground]
8. Мягкие игрушки [soft_toys]
9. Компьютерные игры [computer_games]
10. Сладости [sweets]
11. Книги [books]
12. Домашнее животное [pet]
13. Одежда (сумочки, заколки, платья, рюкзаки) [clothes]

Подарок в каждой категории можно купить за различную цену. Вы не сможете создавать подарки, нужно лишь выбрать из каталога. Да, там готовые подарки имеющихся категорий за определенную цену. Могу добавить, что если вы подарите куклу мальчику, она принесет ему счастья меньше, чем девочке. Ну и более дорогая машинка гораздо интереснее мальчишке, чем дешевая. **Счастье** - тот показатель, который вам нужно максимизировать, покупая подарки детям.

Каталог подарков содержит 5 000 различных позиций, попробуйте сделать такую подборку подарков, в рамках заданного бюджета, которая больше всего порадует указанных детей. Мы будем называть вашу подборку Заказом. Каждый ваш заказ будет обрабатываться 5 минут сервером. Что нужно сделать?

1. Получить данные карты, которая содержит каталог подарков и список детей.
2. Создать свой алгоритм поиска наилучших подарков из каталога для каждого ребенка.
Таким образом получив для каждого ребенка id конкретного товара из каталога. Это и есть заказ.
3. Упаковать заказ в json пакет и отправить post запросом.
4. Проверить статус расчёта заказа через 5 минут по средствам get запроса. Увидеть какой получился индекс счастья по заказу и повторить шаги 2-4, улучшив свой алгоритм.

У этого этапа идентификатор карты **map_id = a8e01288-28f8-45ee-9db4-f74fc4ff02c8**



map

Каталог подарков и список детей

Получение каталога подарков, а также данные по каждому ребенку.

AUTHORIZATIONS: > *ApiKeyAuth*

PATH PARAMETERS

map_id	string
required	Default: "a8e01288-28f8-45ee-9db4-f74fc4ff02c8"
	Имя json файла карты

Responses

> 200 Успешный запрос

— 404 Карта не найдена

GET /json/map/{map_id}.json

Response samples

200

Content type
application/json

Copy Expand all Collapse all

```
{
  - "gifts": [
    + { ... }
  ],
  - "children": [
    + { ... }
  ]
}
```

order

Заказ подарков для детей

Отправка заказа и получение ID созданного заказа

AUTHORIZATIONS: > *ApiKeyAuth*

REQUEST BODY SCHEMA: application/json

Отправка своего варианта заказа

mapID	string Идентификатор карты
presentingGifts >	Array of objects (Present) Массив объектов с указанием какому ребенку будет вручен подарок

Responses

> 200

POST /api/round2



Request samples

Payload

Content type

application/json

Copy Expand all Collapse all

```
{
  "mapID": "string",
  - "presentingGifts": [
    + { ... }
  ]
}
```

Response samples

200

Content type

application/json

Copy

```
{
  "success": true,
  "error": "string",
  "roundId": "string"
}
```

Возвращает статус отправленного ранее заказа

AUTHORIZATIONS: > *ApiKeyAuth*

PATH PARAMETERS

→ id	string
required	Идентификатор отправленного ранее заказа

Responses

> 200

GET /api/round2/{id}



Response samples

200

Content type

application/json

Copy

Expand all

Collapse all

```
{
  "success": true,
  "error": "string",
  "data": {
    "error_message": "string",
    "status": "string",
    "total_happy": 0
  }
}
```