

Задача 1

- Получите информацию об актуальных списках Safe Browsing, хранящихся на серверах Яндекса.

Формат запроса:

GET <https://sba.yandex.net/v4/threatLists?key=<API-ключ>>

Данный запрос позволяет получить список угроз, доступных через сервис безопасности Yandex.

Форма ответа:

Массив данных следующего вида:

```
{
  "threatLists": [
    {
      "threatType": "{enum}",
      "platformType": "{enum}",
      "threatEntryType": "{enum}"
    }
  ]
}
```

Для удобства использования и дальнейшего редактирования запросов я вывел значения URL, версии API, и токена в переменные коллекции:

```
{{URL}}/{{ver}}/threatLists?key={{token}}
```

- Проверьте хотя бы для двух URL содержатся ли они в списках Safe Browsing. Вы можете как самостоятельно составить тело запроса, так и составить его с помощью ChatGPT.

Формат запроса:

POST <https://sba.yandex.net/v4/threatMatches:find?key=<API-ключ>>

Запрос позволяет узнать, содержатся ли проверяемые URL в списках Safe Browsing, размещенных на сервере Яндекса. Если URL ресурса найден хотя бы в одном из списков, возвращается информация об угрозе, которую ресурс представляет для пользователя.

Тело запроса для поиска url yandex.ru:

```
{
  "threatInfo": {
    "threatTypes": ["MALWARE"],
    "platformTypes": ["WINDOWS"],
    "threatEntryTypes": ["URL"],
    "threatEntries": [
      {"url": "https://yandex.ru/"}
    ]
  }
}
```

Тело запроса для поиска URL google.com

```
{
  "threatInfo": {
    "threatTypes": ["UNWANTED_SOFTWARE"],
    "platformTypes": ["LINUX"],
    "threatEntryTypes": ["URL"],
    "threatEntries": [
      {"url": "https://google.com/"}
    ]
  }
}
```

Для удобства использования и дальнейшего редактирования запросов я использовал значения URL, версии API, и токена в переменные коллекции:

```
{{URL}}/{{ver}}/threatMatches.find?key={{token}}
```

Задача 2

Составить коллекцию Postman с 4 основными действиями:

- Добавление одной единицы товара в корзину.

Запрос:

POST {{URL}}/{{ver}}/cart/product

С телом:

```
{  
  
  "id": 3025163  
  
}
```

Где {{URL}} - https://web-gate.chitai-gorod.ru/api

{{ver}} - v1

Тестовый скрипт на проверку кода ответа от сервера:

```
pm.test("Status code is 200", function () {  
  
  pm.response.to.have.status(200);  
  
});
```

- Изменение количества одного товара в корзине.

Запрос:

PUT {{URL}}/{{ver}}/cart

Где {{URL}} - https://web-gate.chitai-gorod.ru/api

{{ver}} - v1

Тело запроса:

```
[  
  
  {  
  
    "id": {{ID}},  
  
    "quantity": 2  
  
  }  
  
]
```

Где {{ID}} - переменная коллекция, которая была записана следующим скриптом:

```
var key = pm.response.json().products[0].id
```

```
pm.collectionVariables.set("ID", key)
```

- Просмотр товаров в корзине.

Запрос:

```
GET {{URL}}/{{ver}}/cart
```

Где {{URL}} - <https://web-gate.chitai-gorod.ru/api>

{{ver}} - v1

- Удаление товара из корзины.

Запрос:

DELETE {{URL}}/{{ver}}/cart/product/{{ID}}

Где {{URL}} - <https://web-gate.chitai-gorod.ru/api>

{{ver}} - v1

{{ID}} - переменная коллекция, которая была записана ранее.

Тестовый скрипт на проверку кода ответа от сервера:

```
pm.test("Status code is 204", function () {  
    pm.response.to.have.status(204);  
});
```