**HTTP-методы**

Любой API нужен, чтобы получить доступ к ресурсам сервера. Ресурс — единица информации, с которой работает API. В проекте Mesto такая единица — карточка с фотографией. Мы можем получить карточку, а её часть — фотографию или число лайков — нет.

К ресурсу всегда можно обратиться по URL:

Скопировать кодBASH

GET https://nomoreparties.co/cards/5d1f0611d321eb4bdcd707dd

Что делать с ресурсом, определяет HTTP-метод запроса. В REST применяются 5 основных методов:

* GET получает ресурсы;
* POST создаёт ресурс;
* PUT целиком заменяет существующий ресурс;
* PATCH частично изменяет существующий ресурс;
* DELETE удаляет ресурс.

Реже применяют ещё 2 метода:

* HEAD позволяет получить только заголовки ответа. HEAD похож на GET, но у его ответа нет тела;
* OPTIONS позволяет узнать, какие HTTP-методы поддерживает сервер.

Сам по себе метод не определяет логику работы сервера. Можно закодить API так, чтобы POST удалял ресурс, а DELETE — добавлял. Но такой подход противоречил бы идее REST о том, что методы должны применяться по назначению.

**Обработка методов на сервере**

REST API может быть написан на любом языке, который умеет обрабатывать запросы, с использованием разных фреймворков. При этом алгоритм обработки методов всегда одинаковый:

* выделяется эндпоинт-ссылка, за которой закреплена функция-обработчик;
* происходит обращение к эндпоинт-ссылке;
* срабатывает обработчик;
* обработчик возвращает определённый результат.