**CRUD REST Паттерн DAO Data Access Object**

**CRUD –** это стандартная классификация функций по манипуляции данными.  
Четыре базовые функции используемые при работе с базами данных:

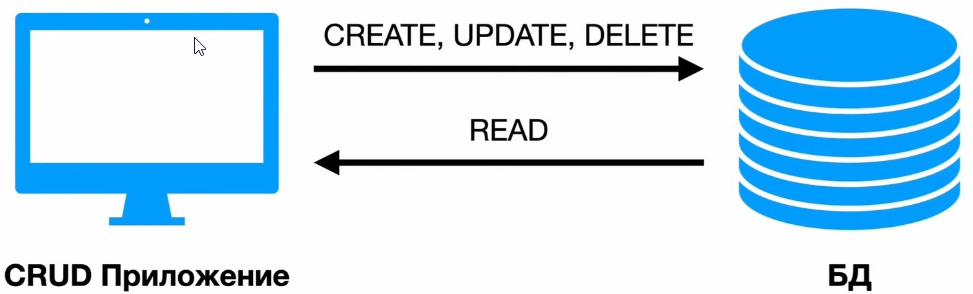
1. CREATE

2. READ

3. UPDATE

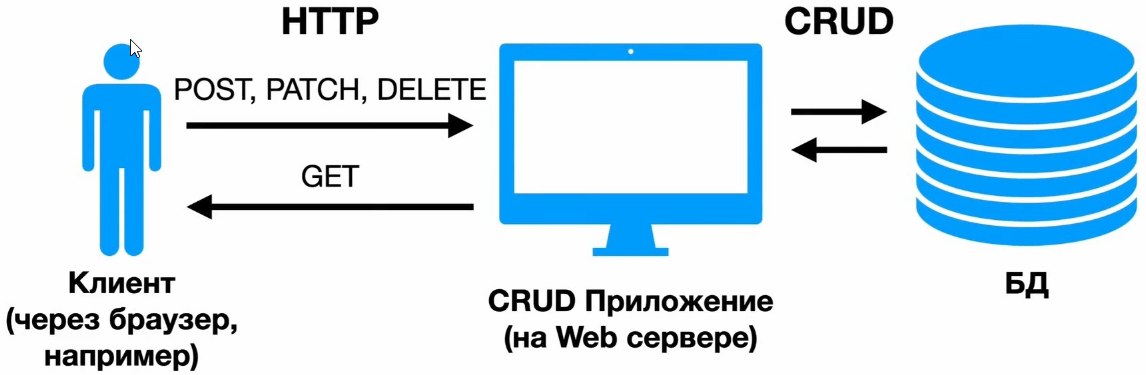
4. DELETE

CRUD приложение — это приложение которое позволяет добавлять (CREATE), читать(READ), обновлять(UPDATE), удалять(DELETE), данные.

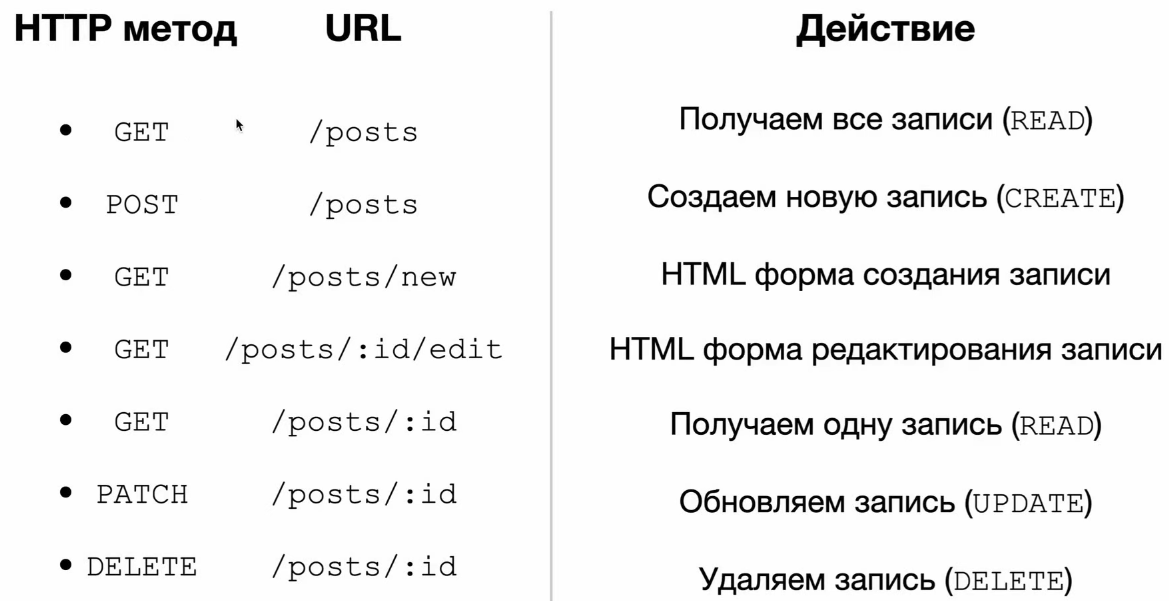


Чаще всего CRUD приложения являются Web-приложениями. То есть доступ к БД предоставляется через протокол HTTP.

Большинство Web приложений — CRUD приложения.



**Пример CRUD`а для сущности Post**



**REST**

**REST –** это паттерн проектирования web-приложений.

REST описывает то, как посредством протокола HTTP должен взаимодействовать клиент(например браузер) с сервером.

Все взаимодействия с сервером сводятся к 4 операциям — получение данных, добавление новых данных, изменение существующих данных, удаление данных.

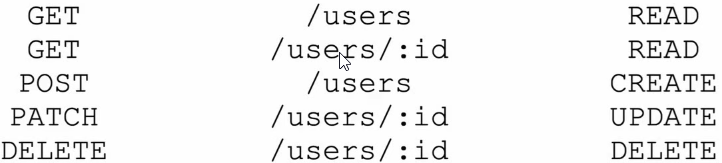
Для каждой из 4 операций используется свой HTTP метод: GET, POST, PATCH/PUT, DELETE.

**В чем разница между REST и CRUD**

**CRUD** – четыре базовые функции, используемые при работе с базами данных.

**REST** – паттерн проектирования Web-приложений. Описывает, как про протоколу HTTP взаимодействовать с сервером для чтения, добавления, изменения. Удаления данных. Описывает, какие URL`ы, HTTP методы использовать (но не только это).

**Связь между REST и CRUD**



**DAO (Data Access Object)**

**DAO –** это паттерн проектирования, когда отдельный класс занимается взаимодействием с БД для конкретной сущности. Например, это когда мы хотим вынести логику взаимодействия с БД из самой модели в отдельный класс.

**Как это выглядит ?**

1. Вся логика работы с БД для модели Person инкапсулируется в класс PersonDAO.

2. В DAO классах обычно пишется SQL код для работы с БД приложения (чтение, запись, обновление, удаление).

3. Есть другой паттерн взаимодействия с БД — **репозиторий (repository).** Этот паттерн предоставляет большую абстракцию над БД, чем DAO.

