

## Выбор СУБД и разработка базы данных.

### Преимущества PostgreSQL

Иногда СУБД называют бесплатным аналогом [Oracle Database](#). Обе системы адаптированы под большие проекты и высокую нагрузку. Но есть разница: они по-разному хранят данные, предоставляют разные инструменты и различаются возможностями. Важная особенность PostgreSQL в том, что эта система — feature-rich: так называют проекты с широким функционалом.

Объектно-реляционная модель. Традиционно популярные СУБД — реляционные. Это значит, что данные, которые в них хранятся, представляются в виде записей, связанных друг с другом отношениями, — relations. Получаются связанные списки, которые могут иметь между собой те или иные отношения, — так и образуется таблица.

Существует еще одна популярная модель — объектная. Данные представляются в виде объектов, их атрибутов, методов и классов. Объектная модель поддерживает возможности, о которых мы подробно рассказывали в [статье про ООП](#), например наследование.

PostgreSQL — объектно-реляционная СУБД. Это значит, что она поддерживает и объектный, и реляционный подход.

#### - Поддержка множества типов данных.

Еще одна особенность PostgreSQL — поддержка большого количества типов записи информации. Это не только стандартные целочисленные значения, числа с плавающей точкой, строки и булевы значения («да/нет»), но и денежный, геометрический, перечисляемый, бинарный и другие типы. PostgreSQL «из коробки» поддерживает битовые строки и сетевые адреса, массивы данных, в том числе многомерные, композитные типы и другие сложные структуры. В ней есть поддержка [XML](#), [JSON](#) и NoSQL-баз.

При необходимости к СУБД можно подключить поддержку типов данных, которые нужны в конкретном проекте. В PostgreSQL есть несколько внутренних форматов, которые используются только в ней.

#### - Работа с большими объемами.

В большинстве СУБД, рассчитанных на средние и небольшие проекты, есть ограничения по объему базы и количеству записей в ней. В PostgreSQL ограничений нет.

Ограничения касаются только конкретных записей. Одна таблица может занимать не больше 32 Тб, а одна запись — 1,6 Тб. В одном поле записи может быть не больше 1 Гб данных, а максимальное количество полей зависит от типа и составляет от 250 до 1600 штук. Максимальных значений хватает, чтобы хранить в БД любые данные.

#### - Поддержка сложных запросов.

PostgreSQL работает со сложными, составными запросами. Система справляется с задачами разбора и выполнения трудоемких операций, которые подразумевают и чтение, и запись, и валидацию одновременно. Она медленнее аналогов, если речь заходит только о чтении, но в других аспектах превосходит конкурентов.

- Написание функций на нескольких языках.

В PostgreSQL можно писать собственные функции — пользовательские блоки кода, которые выполняют те или иные действия. Эта возможность есть практически в любых СУБД, но PostgreSQL поддерживает больше языков, чем аналоги. Кроме стандартного SQL, в PostgreSQL можно писать на C и C++, Java, Python, PHP, Lua и Ruby. Он поддерживает V8 — один из движков JavaScript, поэтому JS тоже можно использовать совместно с PgSQL. Реализована поддержка Delphi, Lisp и прочих редких языков. При необходимости можно расширить систему под другие ЯП. Модификация SQL, которая используется в PostgreSQL, называется PL/pgSQL. Это процедурное расширение, которое поддерживает сложные вычисления и дополняет «классический» SQL новыми возможностями.

**Таблица 2 - Характеристики СУБД**

+	-
Поддержка БД неограниченного размера;	неэффективность архитектуры в плане выполнения записи;
мощные и надёжные механизмы транзакций и репликации;	случаи повреждения таблиц;
расширяемая система встроенных языков программирования и поддержка загрузки C-совместимых модулей;	проблемы с MVCC на репликах;
наследование;	трудности с обновлением.
легкая расширяемость.	

**Таблица 3 — Класс города (City)**

Название поля	Тип поля	Описание поля
---------------	----------	---------------

PK	Числовой	Ключевое
Designation	Текстовый	Наименование города
Slug	Текстовый	Slug тип для url

**Таблица 4 — Класс вакансии (Vacancy)**

Название поля	Тип поля	Описание поля
PK	Числовой	Ключевое
Designation	Текстовый	Наименование города
Slug	Текстовый	Slug тип для url

**Таблица 5 — Класс собранной вакансии (VacancyParsed)**

Название поля	Тип поля	Описание поля
PK	Числовой	Ключевое
City	ForeignKey (указывает на первичный ключ из связанной главной таблицы)	Ссылка на таблицу City
Vacancy	ForeignKey (указывает на первичный ключ из связанной главной таблицы)	Ссылка на таблицу Vacancy
Url	Текстовый	Url адрес нахождения найденной вакансии
Designation	Текстовый	Подробное описание вакансии
Company	Текстовый	Компания
time_stamp	Дата	Время публикации вакансии

Таблица 6 — ER диаграмма.

