

## **Упражнение 1. Создайте список важных данных, относящихся к описанной предметной области**

### **Система управления проектами по разработке программного обеспечения:**

Сообщество планирует создать систему управления проектами по разработке ПО.

База данных должна обрабатывать все этапы разработки программного обеспечения.

В базе данных должны храниться все данные, относящиеся к управлению задачами, версиями продукта, требованиями заказчика и командой разработки. Создайте список данных, которые необходимо собрать и сохранить в базе данных управления проектами по разработке ПО.

*Решение:*

Первая таблица хранит в себе всю необходимую информацию *про людей*:

- ID сотрудника
- ФИО сотрудника
- Его отдел разработки
- Номер команды
- Список навыков сотрудника

Вторая таблица *проектов*:

- ID проекта
- Заказчик, вся информация о нем
- ссылка/id на документацию проекта
- ссылка/id на список задач проекта

## Упражнение 2. Самостоятельно ознакомьтесь с теорией и опишите одну из архитектур систем баз данных

Технология клиент-сервер в контексте систем управления базами данных (СУБД) позволяет разделять работу по взаимодействию с базой данных между двумя компонентами: клиентом и сервером. Рассмотрим основные аспекты этой технологии:

### Компоненты

1. Клиент: Это приложение, которое взаимодействует с пользователем и формирует запросы к базе данных. Он также может выполнять часть обработки данных перед передачей их на сервер. Примерами могут служить настольные приложения, веб-браузеры и мобильные приложения.
2. Сервер: Этот компонент принимает запросы от клиентов, выполняет необходимые операции над базой данных и возвращает результаты обратно клиентам. Сервер обычно управляет доступом к данным, обеспечивая безопасность и целостность информации.

### Протоколы

Для связи между клиентом и сервером используются специальные протоколы. Наиболее распространенным протоколом для доступа к базам данных является SQL (Structured Query Language). Однако существуют и другие протоколы, такие как ODBC (Open Database Connectivity) и JDBC (Java Database Connectivity), которые позволяют подключаться к различным типам баз данных.

### Архитектуры

Одно из основных преимуществ технологии клиент-сервер – гибкость в выборе архитектуры. Можно выделить две основные архитектуры:

1. Двухъярусная архитектура:
  - Клиентское приложение (часто называемое тонким клиентом) передает запрос серверу.
  - Серверный компонент (чаще всего именуемый сервером приложений) обрабатывает запрос и возвращает результат.
2. Трёхъярусная архитектура (N-tier architecture):
  - Предусматривает дополнительные слои, такие как уровень бизнес-логики, который располагается между уровнем представления (клиентским приложением) и уровнем данных (базой данных).

*Преимущества технологии клиент-сервер:*

- Разделение труда: клиентская часть сосредоточена на удобстве пользовательского интерфейса, а серверная – на безопасности и производительности операций с базой данных.
- Масштабируемость: легко добавляются новые клиенты без перестройки всей системы.

- Надежность: данные централизованы на сервере, что снижает риск потери данных при сбоях на клиентских устройствах.

*Недостатки:*

- Зависимость от сети: требуется надежное соединение между клиентами и серверами.
- Нагрузка на сервер: большое количество запросов может замедлить выполнение операций.

Технология клиент-сервер широко используется в современных СУБД, таких как Oracle, MySQL, PostgreSQL и других, что делает ее важным элементом многих корпоративных информационных систем.