

Part 1: Выбор Сценария

Для данной работы выбран сценарий: Управление строительными проектами. Эта Система предназначена для учета клиентов, заказов, проектов, сотрудников, подрядчиков, объектов строительства и официальных отчетов.

Part 2: Проектирование Базы Данных и Документация

Идентификация Сущностей и Атрибутов:

- **Клиенты (Clients):** Хранят информацию о заказчиках.
- **Заказы (Orders):** Содержат сведения о заказах клиентов.
- **Проекты (Projects):** Хранят информацию о строительных проектах.
- **Подрядчики (Contractors):** Хранят данные о подрядчиках.
- **Задачи (Tasks):** Для отслеживания, кто какую работу выполняет.
- **Сотрудники (Employees):** Хранят информацию о сотрудниках.
- **Объекты (Facilities):** Содержат сведения об объектах строительства.
- **Отчёты (OfficialReports):** Для фиксации результатов по объектам.

Проектирование Таблиц:

1. Client

- **Описание:** Хранит информацию о клиентах.
- **Атрибуты:**
 - `id_client`: INTEGER, PRIMARY KEY, NOT NULL
 - `id_order`: INTEGER, FOREIGN KEY
 - `LastName`: VARCHAR(100), NOT NULL
 - `Name`: VARCHAR(100), NOT NULL
 - `MiddleName`: VARCHAR(100)
 - `Phone`: VARCHAR(13)
 - `Email`: VARCHAR(100)
 - `Address`: VARCHAR(100)

2. Order

- **Описание:** Содержит информацию о заказах клиентов.
- **Атрибуты:**
 - `id_order`: INTEGER, PRIMARY KEY, NOT NULL
 - `id_client`: INTEGER, FOREIGN KEY
 - `id_project`: INTEGER, FOREIGN KEY
 - `Date`: DATE
 - `Quantity`: INTEGER
 - `ApproximatePrice`: REAL

3. Project

- **Описание:** Хранит сведения о проектах.
- **Атрибуты:**
 - `id_project`: INTEGER, PRIMARY KEY, NOT NULL

- id_employee: INTEGER, FOREIGN KEY
- Name: VARCHAR(100), NOT NULL
- Description: TEXT
- StartDate: DATE
- ApproximateCompletionDate: DATE

4. Contractor

- **Описание:** Содержит информацию о подрядчиках.
- **Атрибуты:**
 - id_contractor: INTEGER, PRIMARY KEY, NOT NULL
 - id_task: INTEGER, FOREIGN KEY
 - id_facility: INTEGER, FOREIGN KEY
 - id_official_report: INTEGER, FOREIGN KEY
 - LastName: VARCHAR(100)
 - Name: VARCHAR(100)
 - MiddleName: VARCHAR(100)
 - Phone: VARCHAR(13)
 - Email: VARCHAR(100)
 - Address: VARCHAR(100)

5. Task

- **Описание:** Хранит информацию о задачах, назначенных подрядчикам и сотрудникам.
- **Атрибуты:**
 - id_task: INTEGER, PRIMARY KEY, NOT NULL
 - id_employee: INTEGER, FOREIGN KEY
 - id_contractor: INTEGER, FOREIGN KEY
 - DetailedDescription: TEXT
 - Priority: INTEGER(5)
 - ApproximatePrice: REAL

6. Employee

- **Описание:** Содержит данные о сотрудниках.
- **Атрибуты:**
 - id_employee: INTEGER, PRIMARY KEY, NOT NULL
 - id_project: INTEGER, FOREIGN KEY
 - id_task: INTEGER, FOREIGN KEY
 - id_official_report: INTEGER, FOREIGN KEY
 - LastName: VARCHAR(100)
 - Name: VARCHAR(100)
 - MiddleName: VARCHAR(100)
 - Phone: VARCHAR(13)
 - Email: VARCHAR(100)
 - Address: VARCHAR(100)
 - Position: VARCHAR(100)
 - Experience: VARCHAR(100)

7. Facility

- **Описание:** Хранит информацию об объектах строительства.
- **Атрибуты:**
 - `id_facility`: INTEGER, PRIMARY KEY, NOT NULL
 - `id_contractor`: INTEGER, FOREIGN KEY
 - `id_official_report`: INTEGER, FOREIGN KEY
 - `StartDate`: DATE
 - `CompletionDate`: DATE

8. OfficialReport

- **Описание:** Содержит официальные отчёты по объектам.
- **Атрибуты:**
 - `id_official_report`: INTEGER, PRIMARY KEY, NOT NULL
 - `id_contractor`: INTEGER, FOREIGN KEY
 - `id_employee`: INTEGER, FOREIGN KEY
 - `id_facility`: INTEGER, FOREIGN KEY
 - `DateOfReport`: DATE
 - `Description`: TEXT
 - `FinalPrice`: REAL

Взаимосвязи:

- **Clients и Orders (Один-ко-Многим):** Один клиент может сделать множество заказов, но каждый заказ относится к одному конкретному клиенту.

◦ `Orders.ClientID` является внешним ключом, ссылающимся на `Clients.ClientID`.

- **Orders и Projects (Один-ко-Многим):** Один проект может быть связан с множеством заказов, но каждый заказ относится к одному конкретному проекту.

◦ `Orders.ProjectID` является внешним ключом, ссылающимся на `Projects.ProjectID`.

- **Projects и Employees (Один-ко-Многим):** Один проект может включать множество сотрудников, но каждый сотрудник относится к одному конкретному проекту.

◦ `Employees.ProjectID` является внешним ключом, ссылающимся на `Projects.ProjectID`.

- **Contractors и Tasks (Один-ко-Многим):** Один подрядчик может выполнять множество задач, но каждая задача назначена одному конкретному подрядчику.

◦ `Tasks.ContractorID` является внешним ключом, ссылающимся на `Contractors.ContractorID`.

- **Employees и Tasks (Один-ко-Многим):** Один сотрудник может выполнять множество задач, но каждая задача назначена одному конкретному сотруднику.

◦ `Tasks.EmployeeID` является внешним ключом, ссылающимся на `Employees.EmployeeID`.

● **Contractors и OfficialReport (Один-ко-Многим):** Один подрядчик может иметь множество отчётов, но каждый отчёт связан с одним конкретным подрядчиком.

○ OfficialReport.ContractorID является внешним ключом, ссылающимся на Contractors.ContractorID.

● **Facilities и OfficialReport (Один-ко-Многим):** Один объект может иметь множество отчётов, но каждый отчёт относится к одному конкретному объекту.

○ OfficialReport.FacilityID является внешним ключом, ссылающимся на Facilities.FacilityID.

● **Employees и OfficialReport (Один-ко-Многим):** Один сотрудник может составить множество отчётов, но каждый отчёт связан с одним конкретным сотрудником.

○ OfficialReport.EmployeeID является внешним ключом, ссылающимся на Employees.EmployeeID.

Part 3: ER-Диаграмма

