Министерство образования и молодежной политики Свердловской области

ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Основы проектирования баз данных

Отчет по практической работе

на тему: Проектирование базы данных «Платный прием в поликлинике»

Выполнил: Шугуров Алексей

Группа: ПР-21

Преподаватель: С.И. Овчинникова

2024

Рассмотрим пример проектирования базы данных в среде MYSQL.

Описание предметной области.

В данном проекте выполняется разработка реляционной базы данных для информационной системы «Платный прием в поликлинике».

Постановка задачи.

Пациент должен оплатить за прием определённую сумму, которая устанавливается для каждого врача по-своему. За каждый из приемов врачу отчисляется стандартный процент от стоимости приема. Процент отчисления от стоимости приема на зарплату врача также устанавливается персонально для каждого врача. (таб. 1)

При разработке ER-моделей мы должны получить следующую информацию о предметной области:

- список сущностей предметной области;

- список атрибутов сущностей;

- описание взаимосвязей между сущностями;

|  |  |  |
| --- | --- | --- |
| Информация о пациенте | Информация о Приёме | Информация о враче |
| Фамилия пациента  Имя пациента  Отчество пациента  Дата рождения  Адрес пациента  IDP | IDV  IDP  Дата приёма  Зарплата  IDPP | Фио врача  Специальность врача  Процент отчисления  IDV  Стоимость приёма |
|  |  |  |

После анализа предметной области мы выделили три сущности: «Врачи», «ПАЦИЕНТЫ», «ПРИЁМ». Следующим шагом определим атрибуты всех сущностей и вы

Домены из которых атрибуты берут свои значения, приведены в таблице. Здесь же приведены ограничения для атрибутов на уровне кортежей: повторяемость, обязательность и значения по умолчанию. (таб. 2)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Ограничение | Повторя  е  мость | Обязател  ь  ность | Значение по  умолч  а  нию |
| IDP |  | + | + | нет |
| IDV |  | + | + | нет |
| Фио врача | 1. | + | + | нет |
| Фио пациента | 2 | + | + | нет |
| Дата рождения |  | + | + | нет |
| Адрес пациента |  | - | + | нет |
| Специальность врача |  | + | + | нет |
| Процент отчисления | 3. | + | + | нет |
| Стоимость приёма | 4. | + | + | нет |

Примечания:

1. ФИО врача
2. Фио пациента
3. Определённый процент
4. Цена в рублях

Определим типы связей и построим начальную ER-модель данных

(рис. 1).

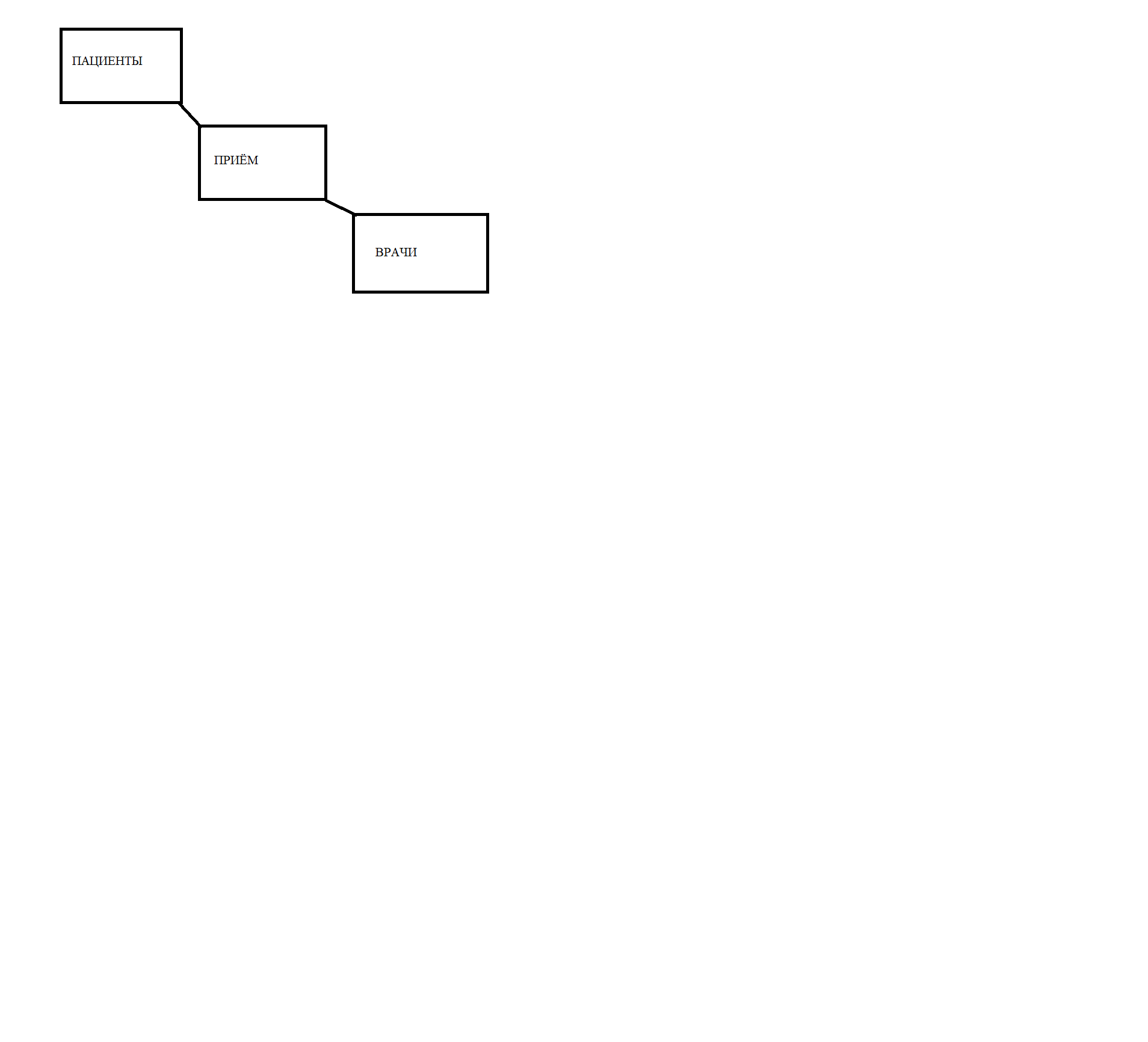
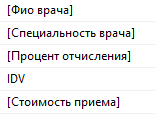
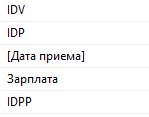
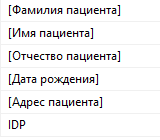


Рис. 1. Концептуальная ER-диаграмма



Преобразование концептуальной модели в концептуальную схему выбранной реляционной СУБД осуществляется в следующей последовательности.

* 1. Для каждой сильной сущности ER-модели создается отдельная таблица, а для каждого атрибута сущности создается столбец таблицы. Ключевой атрибут становится первичным ключом, а дополнительные ключевые атрибуты - потенциальными ключами.
  2. Для каждой слабой сущности также создается отдельная таблица, в которой должны присутствовать ключевые столбцы доминирующих таблиц. В зависимости от вида связи устанавливаются ключевые атрибуты таблицы.
  3. Далее необходимо создать внешние ключи, обеспечивающие ссылочную целостность, по указанному типу связи в ER-модели.

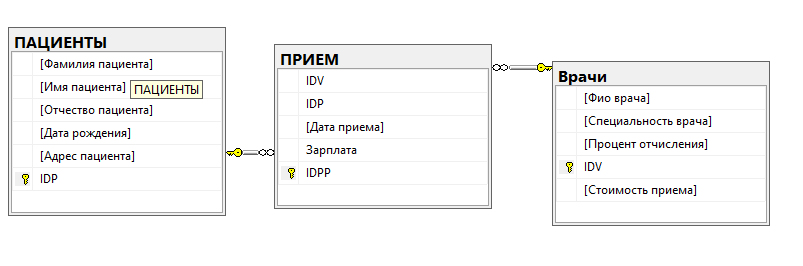
Вполне возможно, что в ER-схеме будет присутствовать избыточность данных, поэтому необходимо нормализировать базу данных, как минимум, до нормальной формы Бойса-Кодда (рис. 2). 

Рис. 2. Концептуальная ER-диаграмма

1. В физической модели каждой сущности будет соответствовать таблица базы данных, а каждому атрибуту – поле таблицы. (таб. 2, таб. 3, таб. 4)

Таблица 3. пациенты

|  |  |
| --- | --- |
| Фамилия пацента | Varchar(50) |
| Имя пациента | Varchar(50) |
| Отчество пациента | Varchar(50) |
| Дата Рождения | date |
| Адрес пациента | Varchar(50) |
| IDP | int |

таб. 2.

Таблица 4. Приёмы

|  |  |
| --- | --- |
| IDV | int |
| IDP | Int |
| Дата приёма | date |
| Зарплата | Money |
| IDPP | int |

таб. 3.

Таблица 5. Врачи

|  |  |
| --- | --- |
| Фио врача | VarChar(50) |
| Специальность врача | Varchar(50) |
| Процент отчисления | float |
| IDV | Int |
| Стоимость приёма | Money |

таб. 4.

1. Разработка базы данных

*4.1. Создание таблиц*

На основании реляционной модели данных необходимо создать 3 таблицы «Пациенты», «Приёмы», «Врачи». Создадим таблицы в режиме конструктора, для этого на вкладке ленты «Создание» выберем пункт «Конструктор таблиц». Каждой таблице назначим ключевое поле (рис. 4, рис. 5, рис. 6).

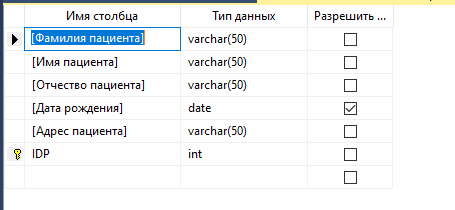


Рис. 4.

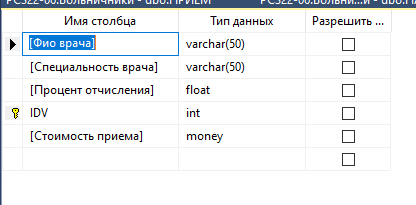


Рис. 5.

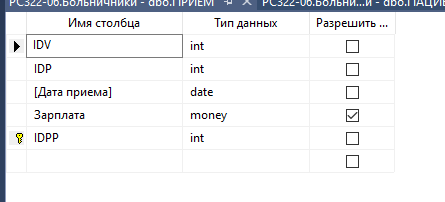
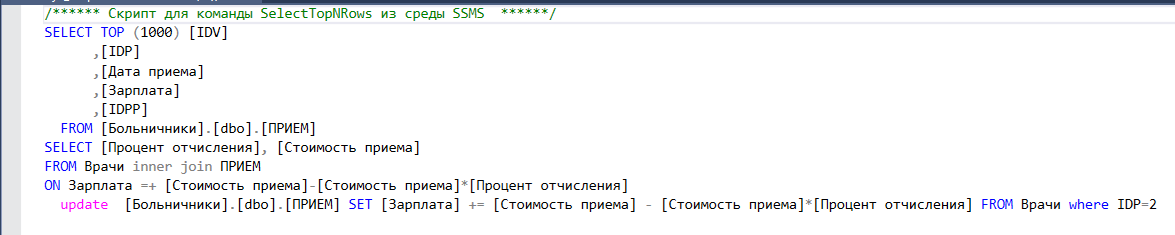


Рис. 6.

*4.2 Создание запросов*

Теперь к созданной таблице создадим запросы для дальнейшего использования. (рис. 7)

Первый запрос мы сделаем по названию Приёмы для вычисления Зарплаты врачей. (рис. 7)



Вывод : Мы научились делать базу данных для предприятий которые основываются на продаже каких либо товаров.