



МОСКОВСКИЙ ФИЗИКО-ТЕХНИЧЕСКИЙ  
ИНСТИТУТ

БАЗЫ ДАННЫХ

КУРСОВАЯ РАБОТА

---

# Проект базы данных для больницы

---

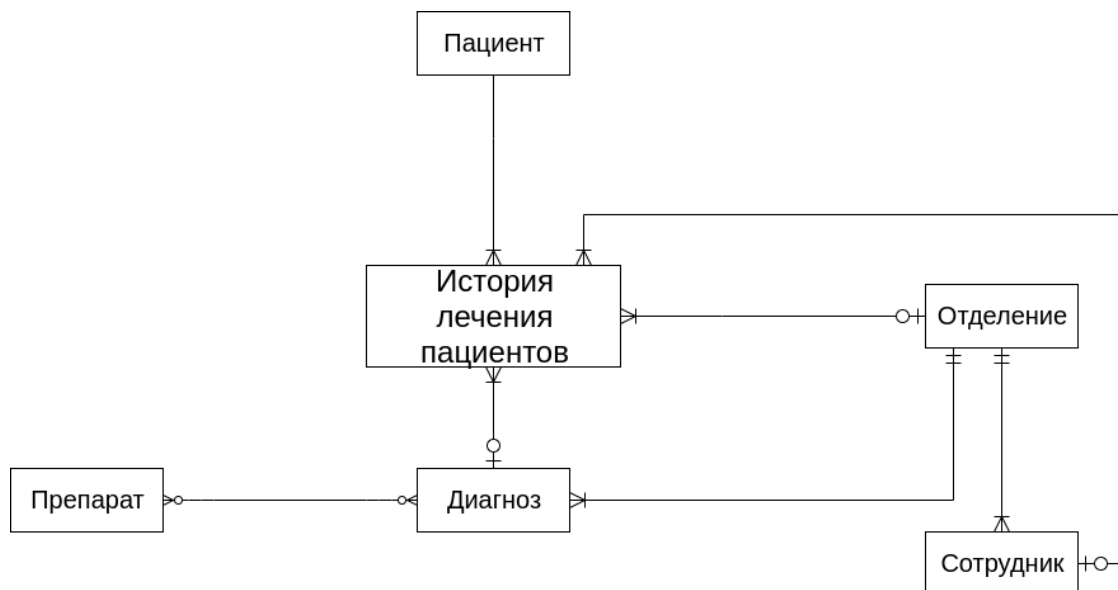
*Автор:*  
Мокеев Дмитрий

*Группа:*  
911

2020/2021

# 1 Проектирование

## 1.1 Концептуальная модель



В концептуальной модели базы данных основную роль выполняет история лечения пациентов, которая является сущностью, хранящую по сути попадания пациентов в больницу (приёмы и стационарное лечение).

Отделение - это сущность-начальник для сотрудников, которые затем будут связаны с ним через вторичный ключ department id.

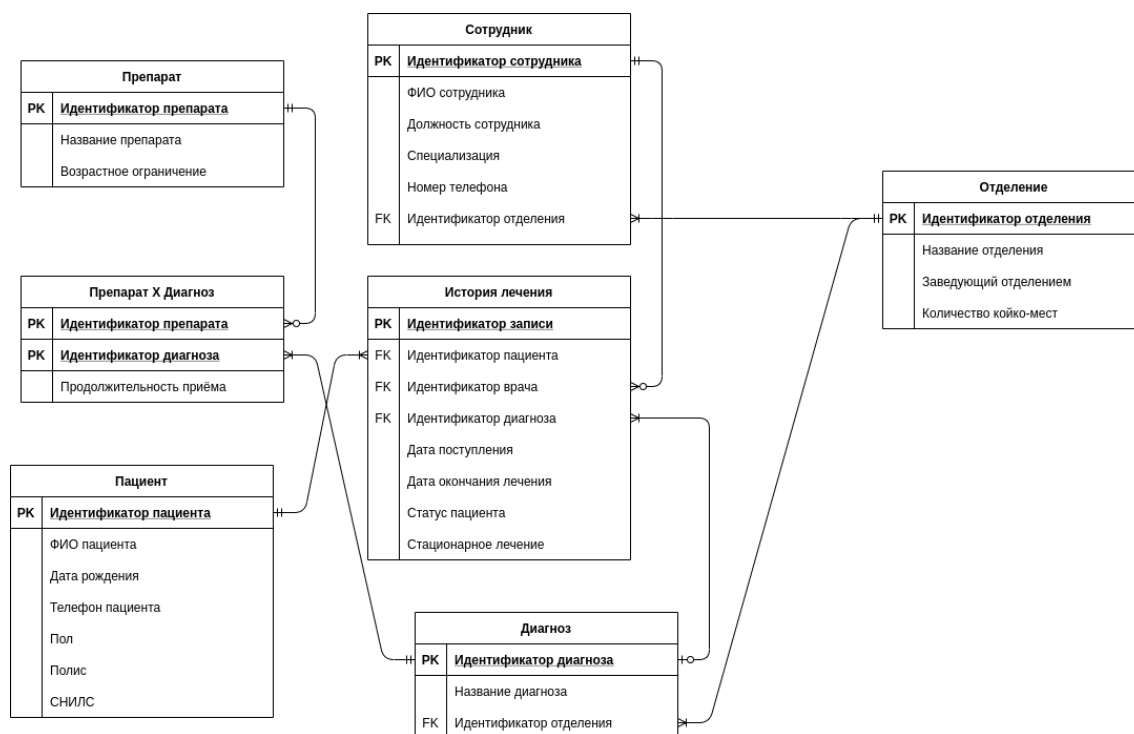
В то же время отделение является родительской сущностью и для диагнозов. Эта связь сделана за желанием хранить в истории лечения идентификатор диагноза за редкостью обращения непосредственно к отделению (возможно, только в период составления отчётов по отделениям).

Препараты являются небольшой вспомогательной сущностью для диагноза, связь многие ко многим.

Пациенты представляют собой людей, которые оказываются в этой базе данных, когда впервые попадают в больницу.

Сотрудник — это трудящийся в больнице, который может быть как врачом, так и нет, это сделано для упрощения модели базы данных.

## 1.2 Логическая модель



В логической модели добавляется всего одна нужная таблица — рвётся связь многие ко многим между препаратами и диагнозами. Почему так получилось? Потому что, к сожалению, объект истории лечения выполняет роль множественной связки и по сути это самая интересная таблица для какого-то анализа и запросов. В то же время она позволяет спастись от трёх отдельных и бессмысленных таблиц.

Она в меру практична с человеческой точки зрения и её удобно заполнять врачам, что в принципе хорошо. При этом таблица находится в 2НФ, выбор обусловлен желанием данные вроде возрастного ограничения в одной таблице, не разрываясь из-за незначительных функциональных связей.

### 1.3 Физическая модель

TREATMENT_HISTORY / ИСТОРИЯ ЛЕЧЕНИЯ			
Название	Описание	Тип данных	Ограничения
record_id	Идентификатор записи	INTEGER	PK
patient_id	Идентификатор пациента	INTEGER	FK
employee_id	Идентификатор врача	INTEGER	FK
diacrisis_id	Идентификатор диагноза	INTEGER	FK
entry_dttm	Время поступления	TIMESTAMP(0)	NOT NULL
treatment_end_dttm	Время выписки	TIMESTAMP(0)	
current_status_dk	Текущий статус	BOOLEAN	OK/D/NULL
in_patient_flag	Лечение в стационаре	BOOLEAN	В стационаре/на дому/NULL

PATIENTS / Пациенты			
Название	Описание	Тип данных	Ограничения
patient_id	Идентификатор пациента	INTEGER	PK
patient_nm	ФИО пациента	VARCHAR(255)	NOT NULL
mobile_phone_no	Номер телефона	VARCHAR(15)	
sex_dk	Пол пациента	VARCHAR(255)	CHECK(sex is 'M' or sex is 'F' or sex is NULL)
birth_dt	Дата рождения	DATE	
assurance_no	Номер полиса	VARCHAR(255)	
snils_no	Номер СНИЛС	VARCHAR(255)	

EMPLOYEES / Сотрудники			
Название	Описание	Тип данных	Ограничения
employee_id	Идентификатор сотрудника	INTEGER	PK
employee_nm	ФИО сотрудника	INTEGER	NOT NULL
post_nm	Название должности	VARCHAR(255)	NOT NULL
specialization_nm	Специализация	VARCHAR(255)	
mobile_phone_no	Номер телефона	VARCHAR(15)	
department_id	Идентификатор отделения	INTEGER	FK

DIACRISISES / Диагнозы			
Название	Описание	Тип данных	Ограничения
diacrisis_id	Идентификатор диагноза	INTEGER	PK
diacrisis_nm	Название диагноза	VARCHAR(255)	NOT NULL
department_id	Идентификатор отделения	INTEGER	FK

DEPARTMENTS / Отделения			
Название	Описание	Тип данных	Ограничения
department_id	Идентификатор отделения	INTEGER	PK
department_nm	Название отделения	VARCHAR(255)	NOT NULL
chief_id	Заведующий отделением	INTEGER	FK
in_patient_bed_cnt	Количество койко-мест	INTEGER	

DRUGS_X_DIACRISISES / Препараты_X_Диагнозы			
Название	Описание	Тип данных	Ограничения
drug_id	Идентификатор препарата	INTEGER	PK, FK
diacrisis_id	Идентификатор диагноза	INTEGER	PK, FK
duration_days	Продолжительность приема	INTEGER	NULL or >= 0

DRUGS / Препараты			
Название	Описание	Тип данных	Ограничения
drug_id	Идентификатор препарата	INTEGER	PK
drug_nm	Название препарата	VARCHAR(255)	NOT NULL
min_age_dk	Минимальный возраст	VARCHAR(255)	NOT NULL

Опишем непосредственно таблицы.

1. В таблице `hospital.treatment_history` хранится информация о актах лечения в больнице.
2. В таблице `hospital.patients` хранится информация о пациентах — их идентификатор и личная информация. В этой таблице пациент оказывается прямо перед первой записью в историю лечения, что логично и соответствует реальности.
3. В таблице `hospital._employees` хранятся данные о сотрудниках — их рабочее и личные данные, а также номер отделения, выступающий в роли вторичного ключа для связи с отделениями.
4. В таблице `hospital._departments` хранятся данные о отделениях, как о лечебных, так и не совсем — например, о руководстве больницы. Заметим, что врач соответствует только одному отделению (в нашей модели).
5. В таблице `hospital._drugs_x_diacrisises` хранится связь между диагнозом и лекарством, а также время приёма лекарства для конкретного заболевания. Потенциально может быть добавлен ID для таблицы, в которой

хранится некая вспомогательная информация.

6. В таблице `hospital._drugs` хранится простейшая информация о лекарствах (можно добавить опцию количества имеющегося лекарства, но вообще база данных предназначена не для этого)
7. В таблице `hospital._diacrisises` хранится информация о диагнозах — по сути название и номер связанного с ним отделения.

## 2 Запросы

Опишем SELECT, следующие описывать не будем.

1. Определяет уровень смертности от COVID-19 в данной больнице за всё время ведения записи.
2. Определяет, какие лекарства чаще всего используются при лечении пациентов в данной больнице и сколько раз они бывают нужны.
3. Определяет случаи, когда в больницу попадает пациент, которого по бумагам нечем было лечить.
4. Простой запрос: определяет всех пациентов фиксированного сотрудника и частоту их посещения.
5. Определяет, сколько койко-мест занято в отделениях в фиксированное время.