МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ОРЛОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
ИМЕНИ И.С. ТУРГЕНЕВА»

Кафедра информационных систем

Отчет   
по лабораторной работе №3  
на тему: « Проектирование базы данных «Журнал группы»»  
по дисциплине «Базы данных»

Выполнили: Марочкин М.А. Шифр: 170584   
 Шорин В.Д. Шифр: 171406  
 Щекотихин С.Е. Шифр: 170590  
ИПАИТ  
Направление: 09.03.04 «Программная инженерия»  
Группа: 71-ПГ  
Проверил:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Отметка о зачете:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата «\_\_\_\_» \_\_\_\_\_\_ 2018 г.

Орел, 2018 г.

# **1 Описание предметной области журнала**

В университете есть студенты. Студенты делятся по группам. У каждой группы есть журнал. Вначале журнала есть список студентов, в котором можно узнать ФИО ученика, его контактный телефон, адрес, дату рождения, дату поступления в университет. У каждой группы есть староста, который является одним из студентов (в группе может быть только один староста), список преподаваемых дисциплин и преподавателей, которые их ведут. Из журнала можно узнать ФИО преподавателя.

Для каждого предмета в журнале отводится определенное количество страниц. По каждому предмету хранится информация о преподавателе, ведущем данный предмет, список дат, когда были проведены занятия и количество часов, отведенное под этот предмет. На каждый тип занятия отводится определенное количество времени. Студенты могут пропускать занятия. В таком случае, если студента на паре не было, то ему выставляется количество пропущенных часов. По каждому предмету есть список тем, которые необходимо пройти на занятиях. После каждого занятия преподаватель записывает название пройденной темы.

**2 Построение логической схемы базы данных**

Проанализировав описание предметной области, выделим следующие объекты: студент, группа, журнал, преподаватель.

На основании выделенных объектов можно выделить следующие сущности: студент, группа, дисциплина, занятие, преподаватель.

Полученная логическая схема представлена на рисунке 1.

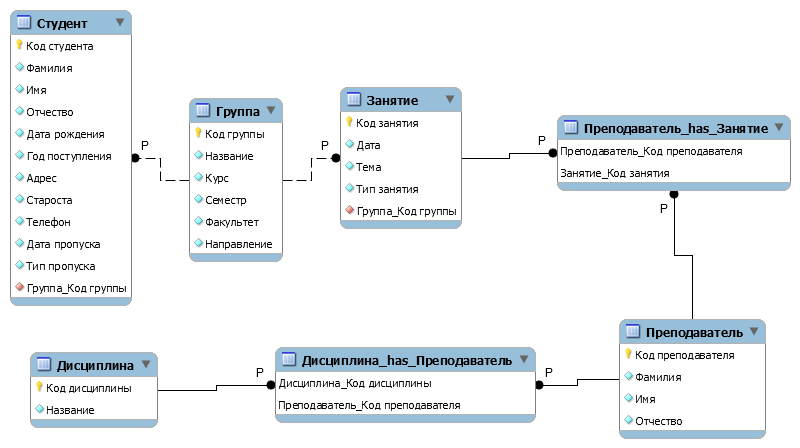


Рисунок 1 – логическая схема базы данных

**3 Нормализация логической схемы**

Чтобы привести полученную схему к первой нормальной форме, нам необходимо сделать следующее:

Выделим следующие сущности с соответствующими атрибутами:

«Телефон»:

* Код телефона (Первичный ключ)
* Номер
* Студент\_Код студента

«Факультет»:

* Код факультета (Первичный ключ)
* Название

«Направление»:

* Код направления (Первичный ключ)
* Название

«Тип занятия»:

* Код типа занятия (Первичный ключ)
* Название
* Количество часов

«Занятие\_has\_Студент»:

* Занятие\_Код занятия
* Студент\_Код студента
* Тип пропуска

Первые два поля являются первичным составным ключом.

«Группа\_has\_Дисциплина»:

* Группа\_Код группы
* Дисциплина\_Код дисциплины
* Тип занятия\_Код типа занятия
* Преподаватель\_Код преподавателя
* Количество часов

Первые три поля являются первичным составным ключом.

«Факультет\_has\_Направление»:

* Факультет\_Код факультета
* Направление\_Код направления

Обозначим связи между таблицами:

«Телефон» - «Студент» : «один ко многим», так как у студента может быть много номеров телефона, но телефон привязан к одному студенту; не идентифицирующая, поскольку сущность «Телефон» уже имеет первичный ключ и ввод еще одного ключа введет функциональную зависимость.

«Студент» - «Группа»: «один ко многим», поскольку в группе может учиться много студентов, но студент учиться только в одной группе (в рамках данной предметной области); не идентифицирующая, поскольку сущность «Студент» уже имеет первичный ключ и ввод еще одного ключа введет функциональную зависимость.

«Преподаватель» - «Группа»: «один ко многим», поскольку преподаватель может быть куратором у нескольких групп, но у группы есть только один куратор; не идентифицирующая, поскольку сущность «Группа» уже имеет первичный ключ и ввод еще одного ключа введет функциональную зависимость.

«Факультет» - «Направление»: «многие ко многим», поскольку на одном факультете может быть много направлений и направление может повторяться на разных факультетах; идентифицирующая, поскольку первичные ключи этих сущностей образуют первичный составной ключ ассоциативной сущности «Факультет\_has\_Направление».

«Студент» - «Занятие»: «многие ко многим, поскольку студент может присутствовать на многих занятиях и на одном занятии может присутствовать много студентов; идентифицирующая, поскольку первичные ключи этих сущностей образуют первичный составной ключ ассоциативной сущности «Занятие\_has\_Студент».

«Группа» - «Дисциплина» - «Тип занятия»: «многие ко многим», поскольку у группы может быть много дисциплин с разными типами занятий; идентифицирующая, поскольку первичные ключи этих сущностей образуют первичный составной ключ ассоциативной сущности «Группа\_has\_Дисциплина».

«Группа\_has\_Дисциплина» - «Преподаватель»: «один ко многим», поскольку на конкретной дисциплине с конкретным типом занятия у конкретных групп могут присутствовать разные преподаватели; не идентифицирующая, поскольку сущность «Группа\_has\_Дисциплина» уже имеет первичный составной ключ и ввод еще одного ключа введет функциональную зависимость.

«Группа\_has\_Дисциплина» - «Занятие»: «один ко многим, поскольку по конкретной дисциплине с конкретным типом занятия у конкретной группы может быть только одно занятие в данную дату с данной темой; не идентифицирующая, поскольку сущность «Занятие» уже имеет первичный ключ и ввод еще одного ключа введет функциональную зависимость.

Проанализировав получившуюся схему можно сделать вывод, что сущности базы данных находятся во второй и третьей нормальных формах.

Нормализованная логическая схема представлена на рисунке 2.

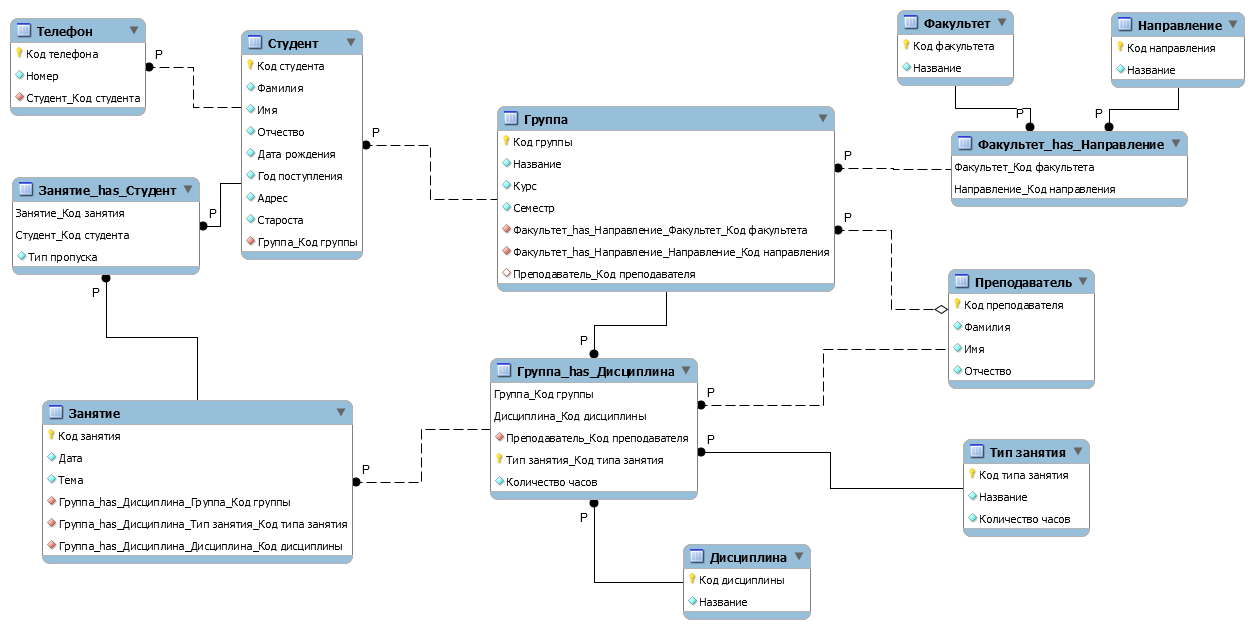


Рисунок 2 – нормализованная логическая схема базы данных

**3 Физическая схема базы данных**

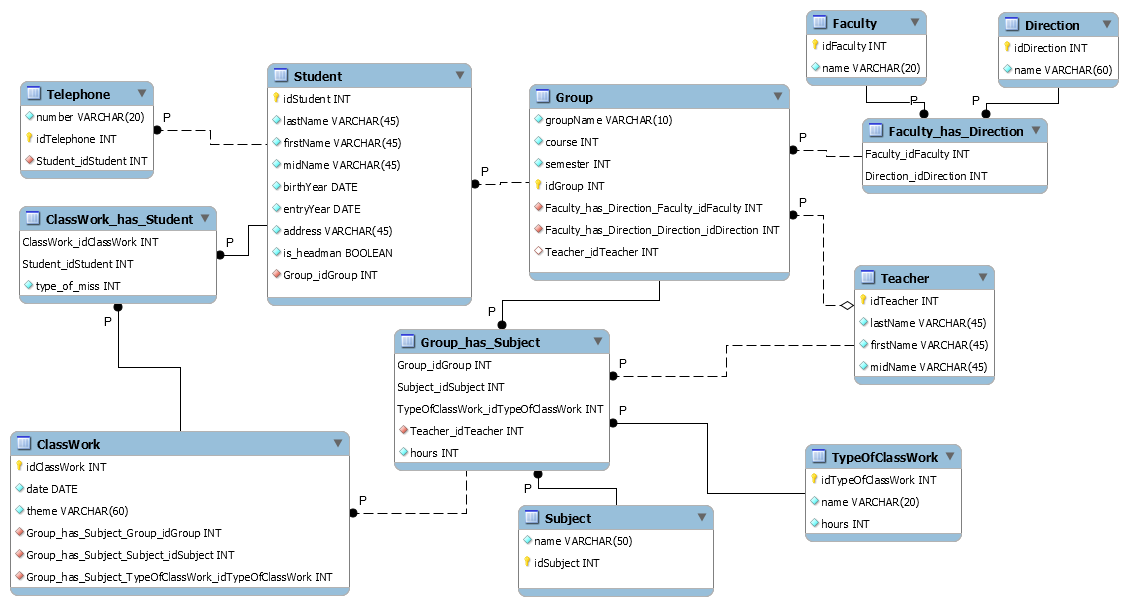
****

Рисунок 3 – физическая схема базы данных

**4 Ограничения целостности**

Ограничением целостности отношений служит наличие у каждого отношения первичного ключа.

Поскольку приведенная выше логическая схема удовлетворяет первым трем нормальным формам, то, следовательно, целостность базы данных соблюдена.

**5 Определение доменов**

Домены для сущности «Student»:

* Поле «idStudent» может принимать значения от 1 до ∞.
* Поле «lastName» может принимать только символьные значения.
* Поле «firstName» может принимать только символьные значения.
* Поле «midName» может принимать только символьные значения.
* Поле «birthYear» может принимать значения >= 1930.
* Поле «entryYear» может принимать значения >= 1960.

Домены для сущности «Telephone»:

* Поле «idTelephone» может принимать значения от 1 до ∞.
* Поле «number» может только цифры 0 – 9 в виде символов.

Домены для сущности «Group»:

* Поле «idGroup» может принимать значения от 1 до ∞.
* Поле «groupName» может принимать только символьные значения.
* Поле «course» может принимать только значения 1-5.
* Поле «semester» может принимать только значения 1-8.

Домены для сущности «ClassWork»:

* Поле «idClassWork» может принимать значения от 1 до ∞.
* Поле «date» может принимать только дату.
* Поле «theme» может принимать только символьные значения.

Домены для сущности «Teacher»:

* Поле «idTeacher» может принимать значения от 1 до ∞.
* Поле «lastName» может принимать только символьные значения.
* Поле «firstName» может принимать только символьные значения.
* Поле «midName» может принимать только символьные значения.

Домены для сущности «Subject»:

* Поле «idSubject» может принимать значения от 1 до ∞.
* Поле «name» может принимать только символьные значения.

Домены для сущности «TypeOfClassWork»:

* Поле «idTypeOfClassWork» может принимать значения от 1 до ∞.
* Поле «name» может принимать только символьные значения.
* Поле «hours» может принимать значения от 1 до ∞.

Домены для сущности «Faculty»:

* Поле «idFaculty» может принимать значения от 1 до ∞.
* Поле «name» может принимать только символьные значения.

Домены для сущности «Direction»:

* Поле «idDirection» может принимать значения от 1 до ∞.
* Поле «name» может принимать только символьные значения.

**6 Определение бизнес-планов**

Бизнес-правила представляют собой механизмы управления БД и предназначены для поддержания БД в целостном состоянии, а также для выполнения ряда других действий, например, накапливания статистики работы с БД. Многие бизнес-правила реализуются с помощью триггеров.

В данном случае нам потребуется реализовать следующие ограничения:

* В группе может быть только один староста
* Количество часов, отведенное под тип занятий по предмету, не должно превышать количество часов, отведенное под весь предмет