**ФЕДЕРАЛЬНОЕ АГЕНТСТВО ВОЗДУШНОГО ТРАНСПОРТА**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ГРАЖДАНСКОЙ АВИАЦИИ» (МГТУ ГА)**

**ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ**

**КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ**

**Отчет по лабораторному практикуму**

По дисциплине: «Базы данных»

Выполнил:

студент группы ПМ4-1

Фейзуллин К.М.

Проверила:

д.т.н., профессор

Егорова А. А.

**Содержание**

[**Лабораторная работа №1** 3](#_Toc69661224)

[**Лабораторная работа №2** 15](#_Toc69661225)

[**Лабораторная работа №3** 25](#_Toc69661226)

[**Лабораторная работа №4** 32](#_Toc69661227)

[**Лабораторная работа №5** 37](#_Toc69661228)

[**Лабораторная работа №6** 40](#_Toc69661229)

# **Лабораторная работа №1**

***Цель лабораторной работы.***

Целью лабораторной работы является изучение:

* методов моделирования предметной области,
* элементов модели «Сущность-связь»;
* концепции нормализации БД.

***Задание на выполнение лабораторной работы.***

В соответствии со своим вариантом для заданной предметной области:

1. выполнить разработку концептуальной модели данных;
2. выполнить моделирование логической структуры данных;
3. разработать ER-диаграммы;
4. привести построенную модель к третьей нормальной форме или доказать, что она в ней находится (в том числе выявить все функциональные зависимости);
5. создать проект, базу данных и описать таблицы базы данных MYSQL 5.5, входящей в состав Денвер 3.

Задание 9.

База данных должна содержать сведения о следующих объектах:

* Сотрудники – табельный номер, фамилия, имя, отчество, должность, отдел, оклад, сведения о перемещении, адрес, номер паспорта, дата выдачи, учреждение, выдавшее паспорт, дата рождения, сведения о перемещении (отдел, должность, причина перевода, номер и дата приказа), семейное положение, состав семьи(иждивенцы), отношение к военной службе.
* Командировки – сотрудник, город, цель командировки, срок командировки, дата начала, аванс, авансовый отчет (назначение платежа, получатель платежа, сумма).
* Отделы – название, штатное расписание (наименование должности, разряд по ETC, количество ставок, количество занятых ставок).

*Добавить специфику проектов (по каким работают и т.п.)*

Выходные документы:

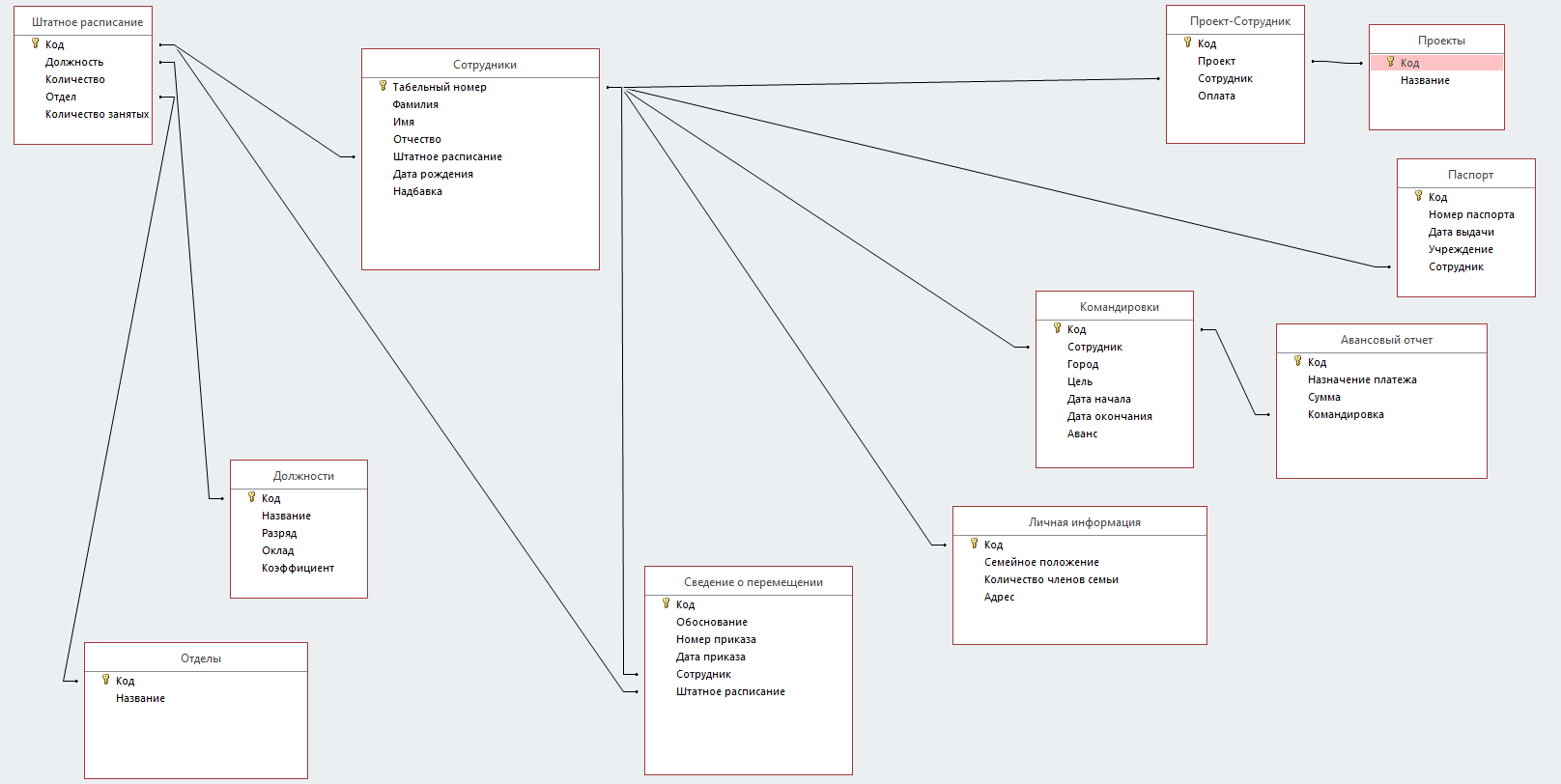
* Распределение среднего оклада по отделам
* Распределение суммы командировочных по отделам.
* Штатное расписание отдела.

Бизнес-правила:

* Каждый сотрудник работает только в одной должности и только в одном отделе.
* Оклад сотрудника определяется должностью, но некоторые сотрудники имеют персональную надбавку.
* В каждом отделе более одного сотрудника.
* Некоторые сотрудники неоднократно ездят в командировки в различные города России.
* Сведения о командировках сохраняются в течении года.
* Сведения об уволенных сотрудниках сохраняются в течении 5 лет.
* В одном проекте может участвовать много сотрудников, и сотрудник может участвовать во многих проектах.
* Штатное расписание не должно меняться на протяжении пользования БД после создания расписания.
* В целях безопасности, личная информация сотрудника хранится в отдельной таблице со связью 1:1.
* Доступ к таблице с личной информацией имеет только кадровый отдел.
* Назначать сотрудника на командировку может только начальник отдела.
* Изменять данные сотрудника может только работник кадрового учета.
* Работник бухгалтерии может только получить требуемые отчеты.
* Один сотрудник может быть только в одной командировке в данный момент. Но в одну командировку может отправиться много работников.

**Схема данных**

Отразим теперь представление сущностей и их атрибутов в виде схемы базы данных. Воспользуемся СУБД Microsoft Access:



Поясним схему БД, изображенную выше.

Поля, отмеченные символом ключа, обозначают первичный ключ. БД включает 11 сущностей. Приведем краткое описание полей каждой из них.

1. Сотрудники:
2. Табельный номер (первичный ключ).
3. Фамилия.
4. Имя.
5. Отчество.
6. Штатное расписание (внешний ключ для сущности «Штатное расписание»).
7. Дата рождения.
8. Надбавка.
9. Штатное расписание
10. Код (первичный ключ).
11. Должность (внешний ключ для сущности «Должности»).
12. Количество.
13. Отдел (внешний ключ для сущности «Отделы»).
14. Количество занятых.
15. Должности
16. Код (первичный ключ).
17. Название.
18. Разряд.
19. Оклад.
20. Коэффициент.
21. Отделы
22. Код (первичный ключ).
23. Название.
24. Сведения о перемещении
25. Код (первичный ключ).
26. Обоснование - краткое описание того, почему была сменена должность или сменен отдел.
27. Номер приказа.
28. Дата приказа
29. Сотрудник (внешний ключ сущности «Сотрудники»).
30. Штатное расписание (внешний ключ сущности «Штатное расписание»).
31. Личная информация
32. Код (первичный ключ).
33. Семейное положение.
34. Количество членов семьи.
35. Адрес.
36. Паспорт
37. Код (первичный ключ).
38. Номер паспорта.
39. Дата выдачи.
40. Учреждение.
41. Сотрудник (внешний ключ сущности «Сотрудники»).
42. Командировки
43. Код (первичный ключ).
44. Сотрудник (внешний ключ сущности «Сотрудники»).
45. Город.
46. Цель
47. Дата начала.
48. Дата окончания.
49. Аванс
50. Авансовый отчет
51. Код (первичный ключ).
52. Назначение платежа.
53. Сумма.
54. Командировка (внешний ключ сущности «Командировка»).
55. Дата сдачи авансового отчета.
56. Проект – сотрудник (сущность – связка)
57. Код (первичный ключ).
58. Проект (внешний ключ сущности «Проекты»).
59. Сотрудник (внешний ключ сущности «Сотрудники»)
60. Оплата.
61. Проекты
62. Код (первичный ключ).
63. Название.

Стоит отметить, что все атрибуты данной БД имеют русскоязычные названия для удобства проверки. В дальнейшем они получат соответствующие англоязычные названия – это необходимо для устранения неудобств при последующей разработке интерфейса к БД.

Докажем теперь, что наша модель находится в 3 нормальной форме (3НФ). Сразу заметим, что наша модель находится в 1НФ, так как во всех ее сущностях в отдельности ни одна из строк не содержит в своем поле более одного значения и ни одно из их ключевых полей не пусто. Рассмотрим каждую сущность:

1. Сотрудники

𝑅 = (A, B, C, D, E, F, G).

Каждое из полей B, C, D, E, F, G (не являющееся первичным ключом) функционально зависит от первичного ключа - поля А (Табельный номер), следовательно, данная таблица находится в 2НФ. Также внутри отношения нет транзитивных функциональных связей между атрибутами (ни одно из не ключевых полей B, C, D, E, F, G не зависит функционально от любого другого не ключевого поля), что говорит о том, что отношение R находится в 3НФ.

1. Штатное расписание

𝑅 = (A, B, C, D, E).

Каждое из полей B, C, D, E (не являющееся первичным ключом) функционально зависит от первичного ключа - поля А (Код), следовательно, данная таблица находится в 2НФ. Также внутри отношения нет транзитивных функциональных связей между атрибутами (ни одно из не ключевых полей B, C, D, E не зависит функционально от любого другого не ключевого поля), что говорит о том, что отношение R находится в 3НФ.

1. Должности

𝑅 = (A, B, C, D, E).

Каждое из полей B, C, D, E (не являющееся первичным ключом) функционально зависит от первичного ключа - поля А (Код), следовательно, данная таблица находится в 2НФ. Также внутри отношения нет транзитивных функциональных связей между атрибутами (ни одно из не ключевых полей B, C, D, E не зависит функционально от любого другого не ключевого поля), что говорит о том, что отношение R находится в 3НФ.

1. Отделы

𝑅 = (A, B).

Поле B (не являющееся первичным ключом) функционально зависит от первичного ключа - поля А (Код), следовательно, данная таблица находится в 2НФ. Также внутри отношения нет транзитивных функциональных связей между не ключевыми атрибутами, что говорит о том, что отношение R находится в 3НФ.

1. Сведения о перемещении

𝑅 = (A, B, C, D, E, F).

Каждое из полей B, C, D, E, F (не являющееся первичным ключом) функционально зависит от первичного ключа - поля А (Код), следовательно, данная таблица находится в 2НФ. Также внутри отношения нет транзитивных функциональных связей между атрибутами (ни одно из не ключевых полей B, C, D, E, F не зависит функционально от любого другого не ключевого поля), что говорит о том, что отношение R находится в 3НФ.

1. Личная информация

𝑅 = (A, B, C, D).

Каждое из полей B, C, D (не являющееся первичным ключом) функционально зависит от первичного ключа - поля А (Код), следовательно, данная таблица находится в 2НФ. Также внутри отношения нет транзитивных функциональных связей между атрибутами (ни одно из не ключевых полей B, C, D не зависит функционально от любого другого не ключевого поля), что говорит о том, что отношение R находится в 3НФ.

1. Паспорт

𝑅 = (A, B, C, D, E).

Каждое из полей B, C, D, E (не являющееся первичным ключом) функционально зависит от первичного ключа - поля А (Код), следовательно, данная таблица находится в 2НФ. Также внутри отношения нет транзитивных функциональных связей между атрибутами (ни одно из не ключевых полей B, C, D, E не зависит функционально от любого другого не ключевого поля), что говорит о том, что отношение R находится в 3НФ.

1. Командировки

𝑅 = (A, B, C, D, E, F, G).

Каждое из полей B, C, D, E, F, G (не являющееся первичным ключом) функционально зависит от первичного ключа - поля А (Код), следовательно, данная таблица находится в 2НФ. Также внутри отношения нет транзитивных функциональных связей между атрибутами (ни одно из не ключевых полей B, C, D, E, F, G не зависит функционально от любого другого не ключевого поля), что говорит о том, что отношение R находится в 3НФ.

1. Авансовый отчет

𝑅 = (A, B, C, D, E).

Каждое из полей B, C, D, E (не являющееся первичным ключом) функционально зависит от первичного ключа - поля А (Код), следовательно, данная таблица находится в 2НФ. Также внутри отношения нет транзитивных функциональных связей между атрибутами (ни одно из не ключевых полей B, C, D, E не зависит функционально от любого другого не ключевого поля), что говорит о том, что отношение R находится в 3НФ.

1. Проект - сотрудник

𝑅 = (A, B, C, D).

Каждое из полей B, C, D (не являющееся первичным ключом) функционально зависит от первичного ключа - поля А (Код работника), следовательно, данная таблица находится в 2НФ. Также внутри отношения нет транзитивных функциональных связей между атрибутами (ни одно из не ключевых полей B, C, D не зависит функционально от любого другого не ключевого поля), что говорит о том, что отношение R находится в 3НФ.

Заметим, что R является отношением-связкой, так как содержит внешние ключи (B, C), являющиеся первичными в сущностях «Проекты» и «Сотрудники» соответственно.

1. Проекты

𝑅 = (A, B).

Поле B (не являющееся первичным ключом) функционально зависит от первичного ключа - поля А (Код), следовательно, данная таблица находится в 2НФ. Также внутри отношения нет транзитивных функциональных связей между атрибутами, что говорит о том, что отношение R находится в 3НФ.

Таким образом, все сущности находятся в 3НФ, а это означает, что наша модель находится в 3НФ, что и требовалось доказать. Это говорит о том, что мы правильно выделили сущности на этапе формирования инфологической модели данных.

**Концептуальная модель данных**

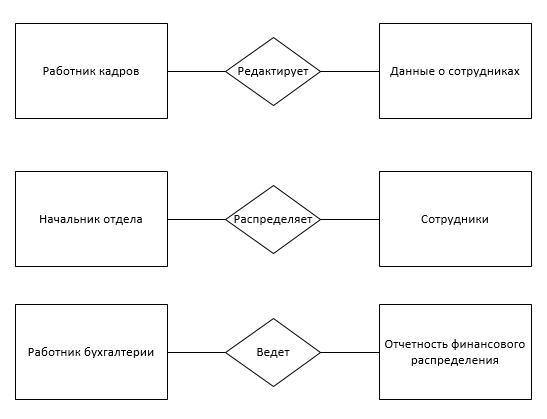
Роли:

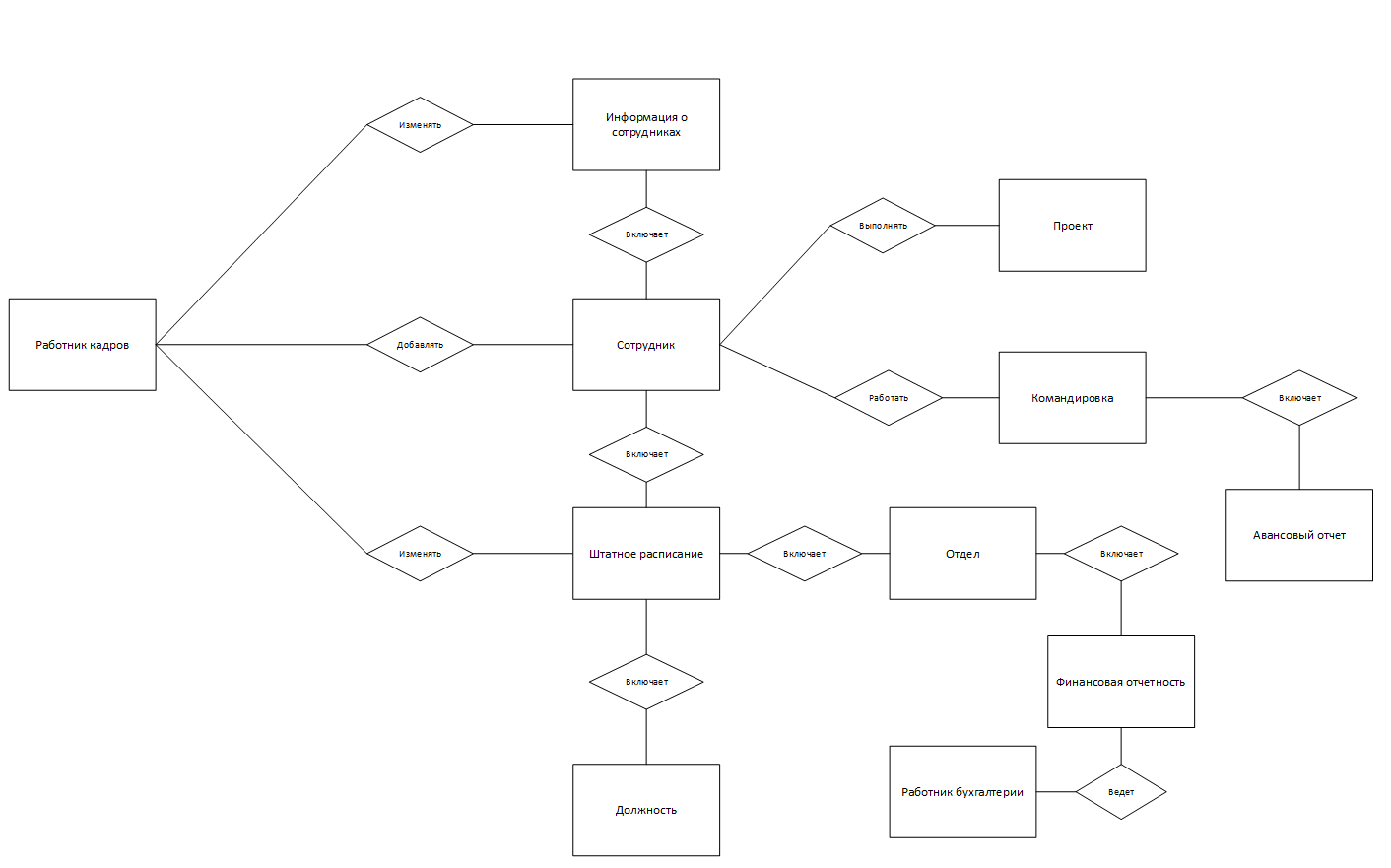
* Работник кадрового учета
* Начальник отдела
* Работник бухгалтерского отдела

Процесс:

* Работник кадрового учета отвечает за содержание информации в базе данных, связанной напрямую с работниками: наполняет базу сотрудниками, устанавливает их отдел, должность, вносит личную информацию и данные паспорта, а также ведет информацию о перемещении сотрудника внутри компании и составляет штатное расписание.
* Начальник отдела направляет работников на командировку и на проекты и заполняет всю соответствующую информацию по своим подчиненным.
* Работник бухгалтерского отдела пользуется базой для составления отчетов по среднему окладу отдела и распределению суммы командировочных по отделу.
* Штатное расписание отдела может получить как начальник отдела, так и работник кадрового учета по требуемому отделу.

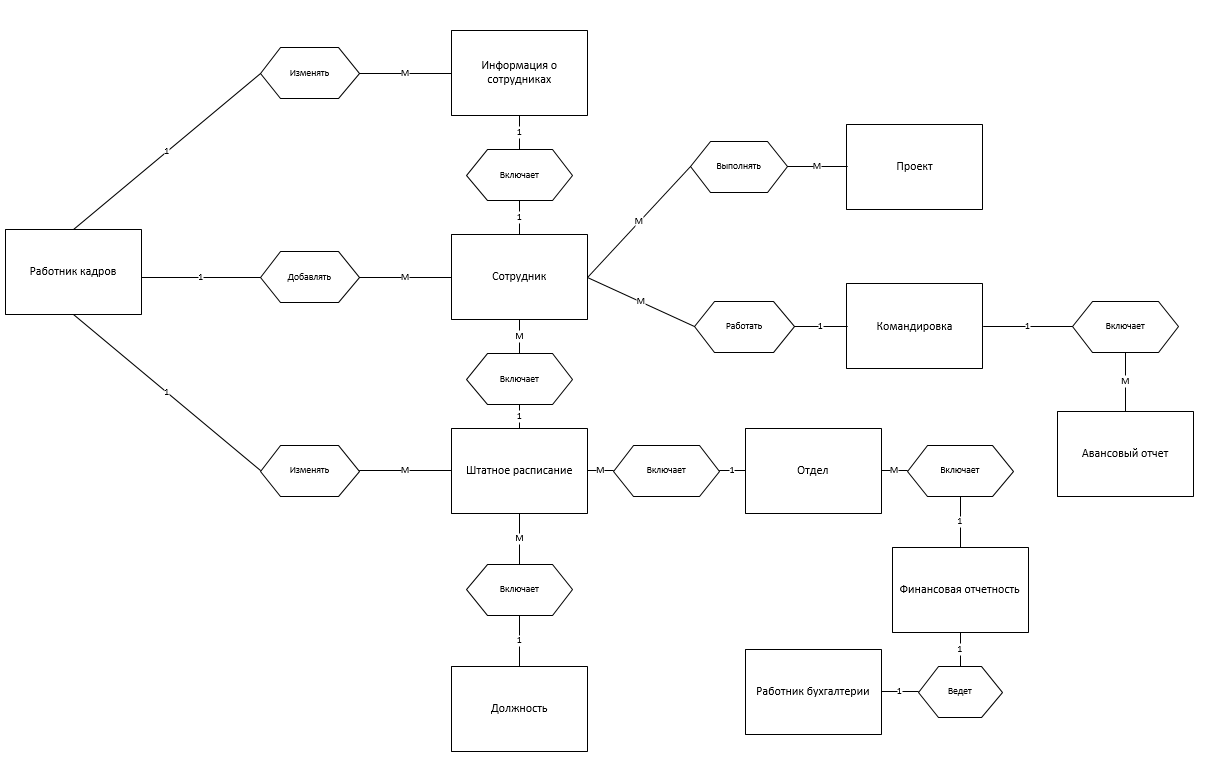
Тогда можно составить первое простое схематическое отражение предметной области:



Составим более подробную схему, включающую работника отдела кадров, добавляющего сотрудников и их информацию, штатное расписание и должности, а также отделы. Так как начальник отдела по сущности своей является сущностью «Сотрудник», его на схеме указывать не будем. Данную схему уже можно назвать концептуальной моделью предметной области:

**Инфологическая модель данных**

Составим теперь инфологическую модель. Для этого будем использовать ER-диаграммы:



# **Лабораторная работа №2**

***Цель лабораторной работы.***

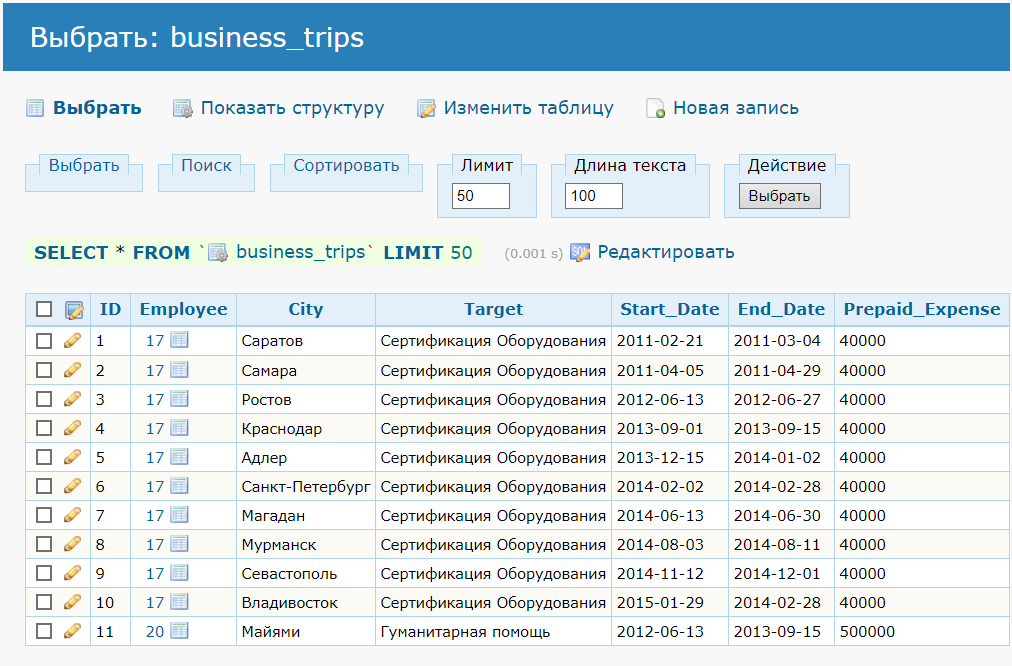
Целью лабораторной работы является изучение СУБД MySQL и методов проектирования в ней баз данных.

***Задание на выполнение лабораторной работы.***

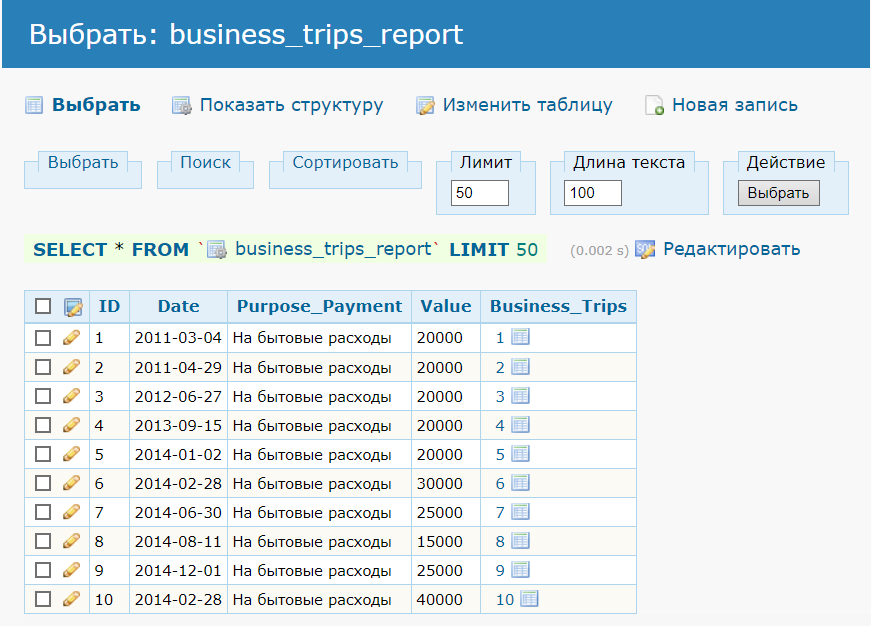
В соответствии с созданным в MYSQL 5.5 на лабораторной работе №2 проектом:

1. наполнить таблицы данными;
2. спроектировать запросы к БД;
3. проверить корректность запросов в части достаточности данных для демонстрации корректности и полноты запросов.

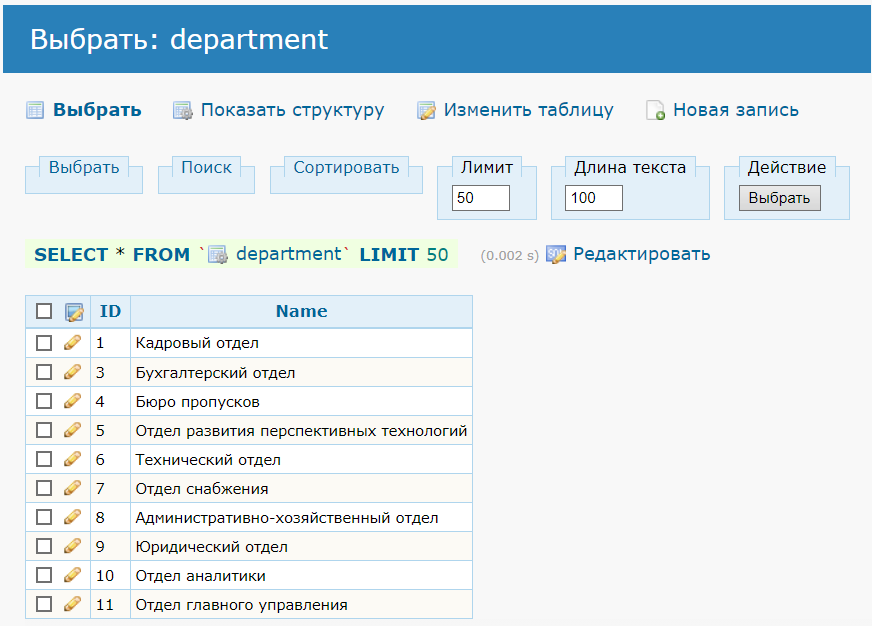
Командировки:



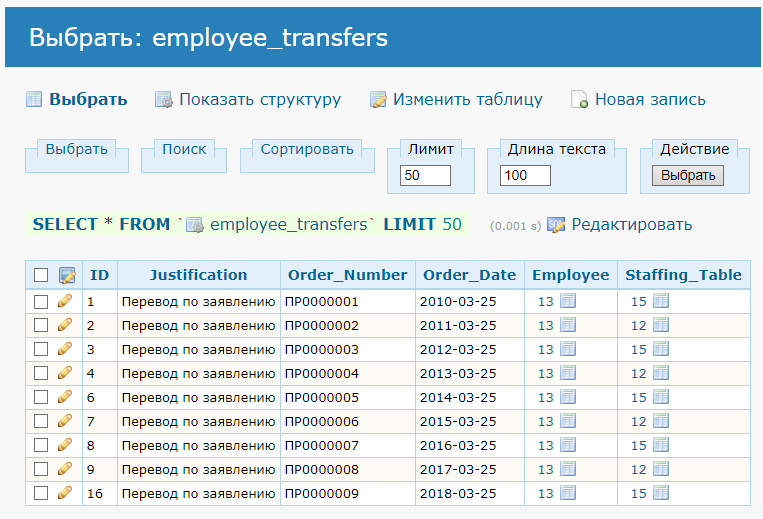
Авансовый отчет:



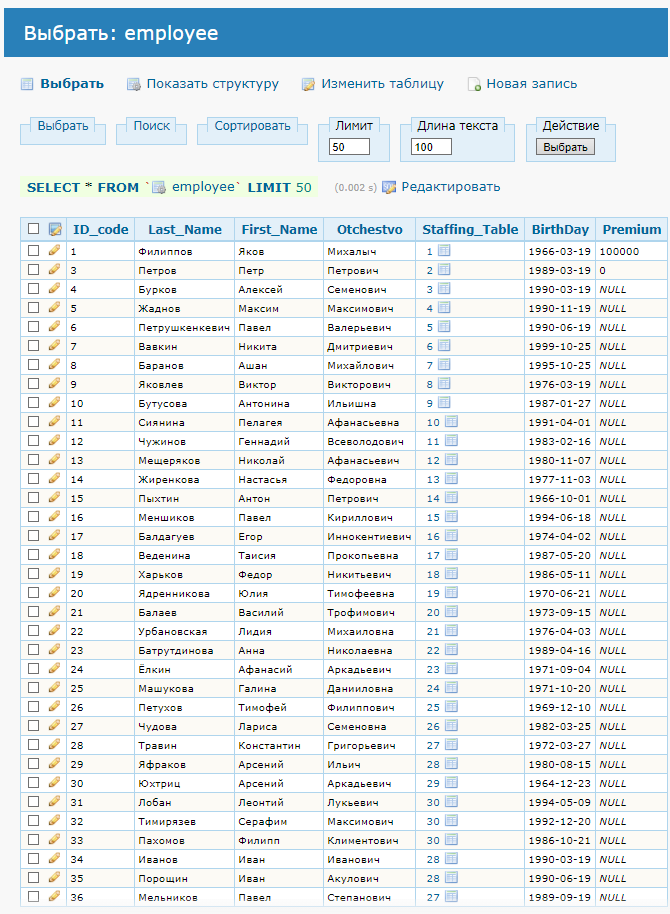
Отделы:



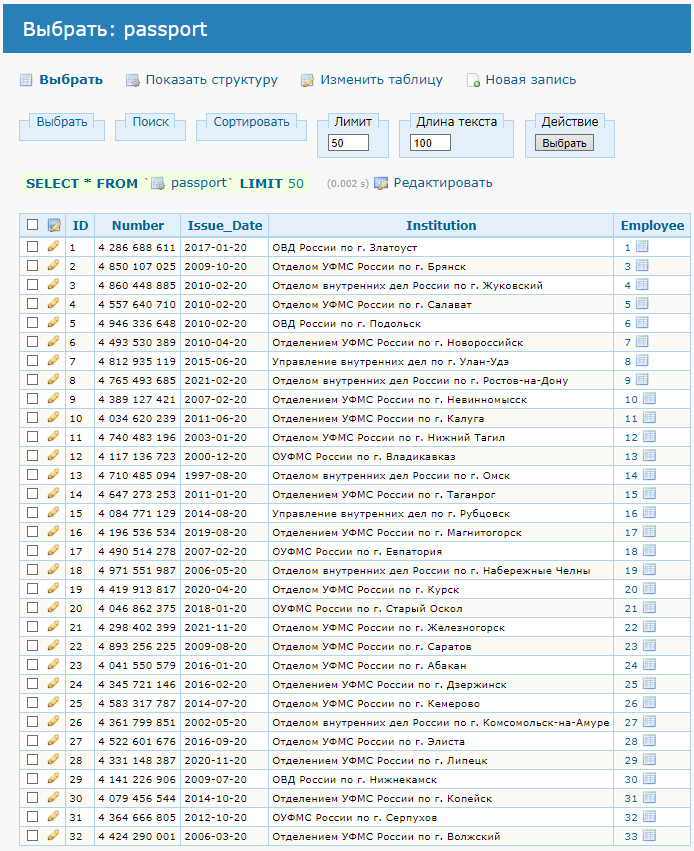
Перемещения работника:



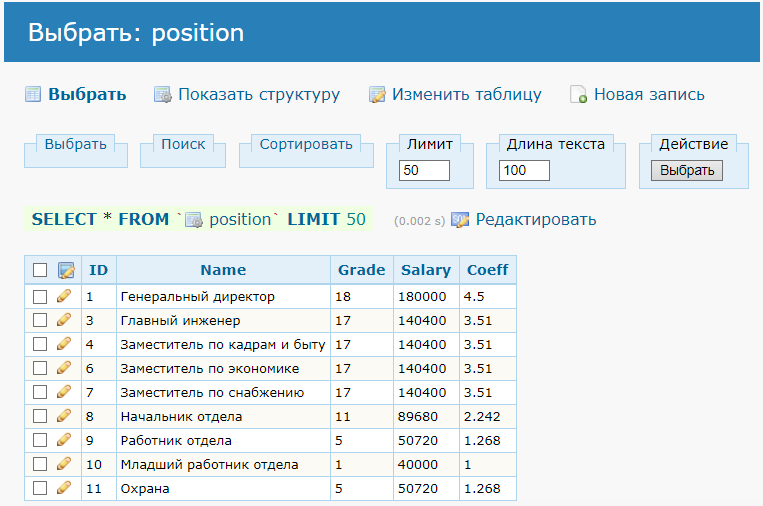
Работники:



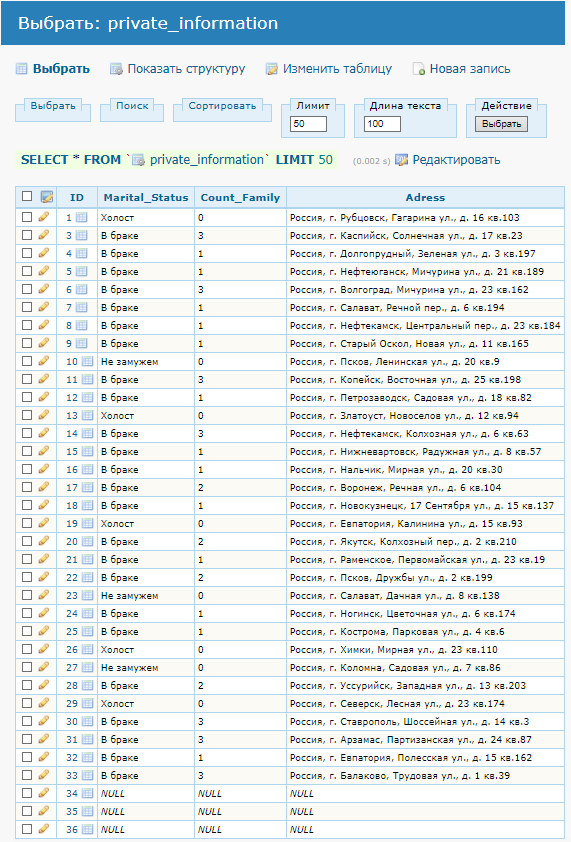
Паспортные данные:



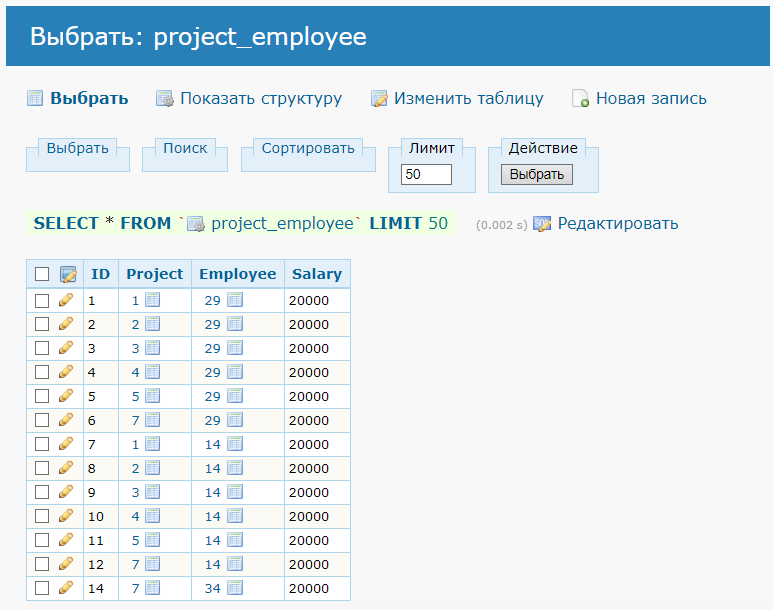
Должности:



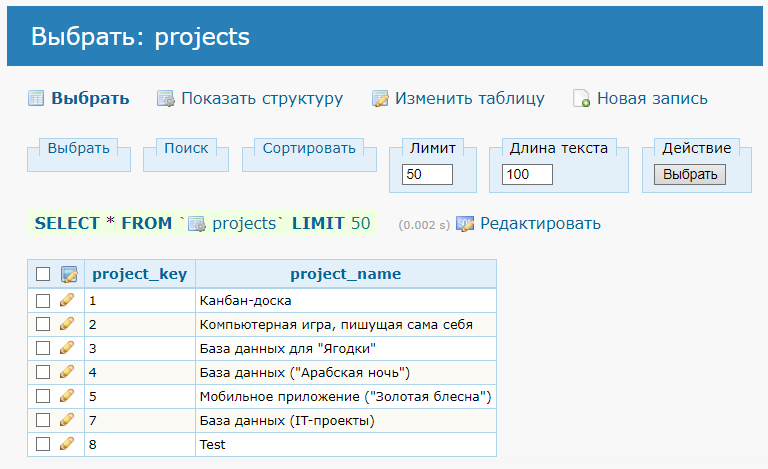
Личная информация работника:



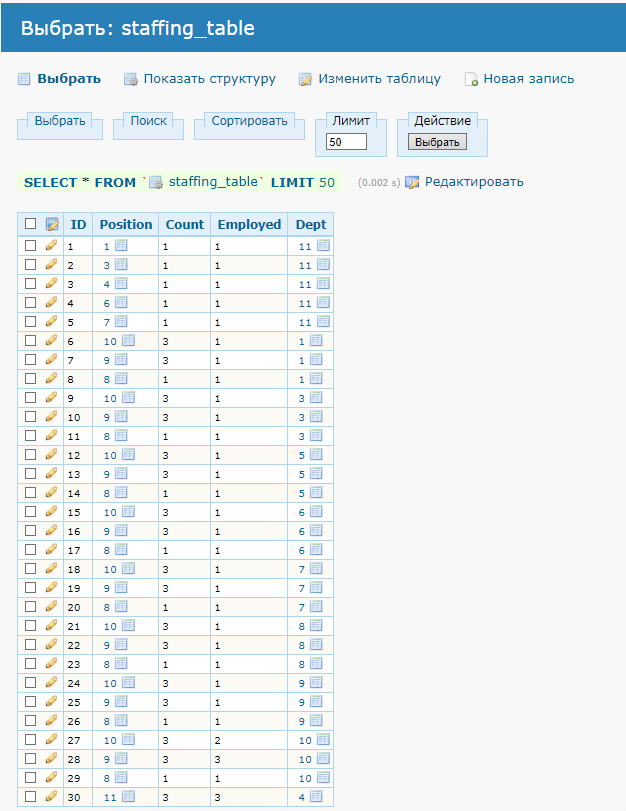
Кросс-таблица (проект – работник):



Проекты:



Штатное расписание:



Список запросов:

1. Все должности.
2. Работники по должностям.
3. Средний оклад по отделу.
4. Средняя сумма командировочных отдела.
5. Вывод истории перемещений работника.
6. Адрес проживания работника.
7. Работники по проектам.
8. Все отделы.
9. Все проекты.
10. Все командировки.
11. Работники по отделам.
12. Работники, не отчитавшиеся за командировку.

# **Лабораторная работа №3**

***Цель лабораторной работы.***

**Целью лабораторной работы** является

* изучение возможностей СУБД по извлечению информации из БД,
* освоение оператора выборки SELECT,
* реализации реляционной алгебры на языке SQL,
* применения агрегирующих функций, группировки, сортировки.

***Задание на выполнение лабораторной работы.***

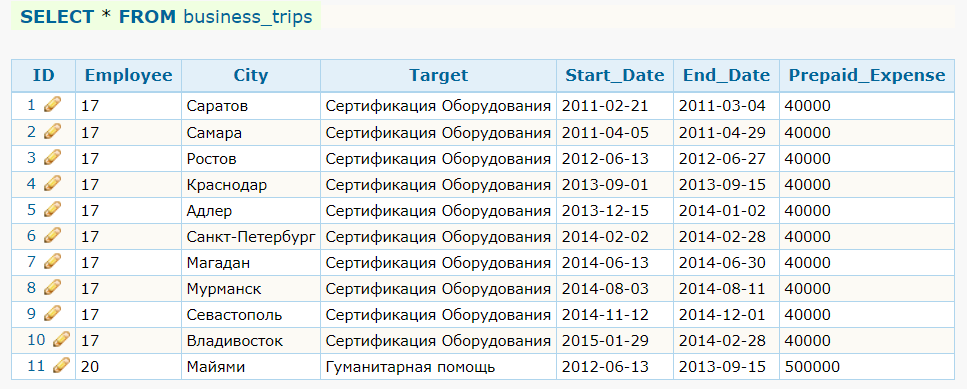
1. Сформулировать запросы к Базе данных, наполненной на лабораторной работе №2 с использованием операции SELECT с учетом бизнес-правил.
2. Проверить корректность и полноту запросов. Состав запросов должен быть достаточным для демонстрации учета всех бизнес-правил при проектировании логической модели данных.

**Простейшие запросы (1 таблица):**

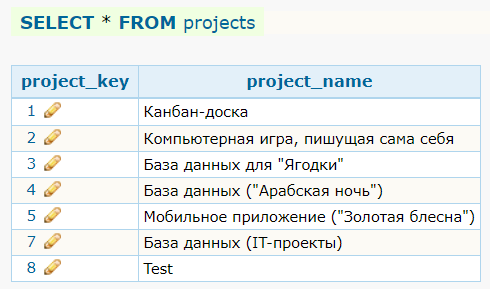
Все должности:



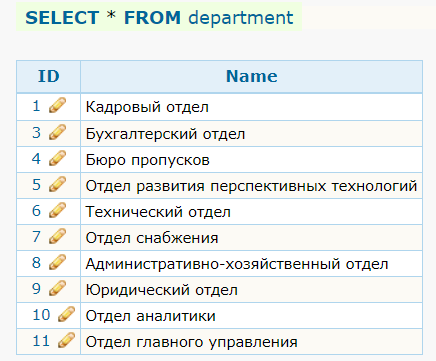
Все командировки:



Все проекты:

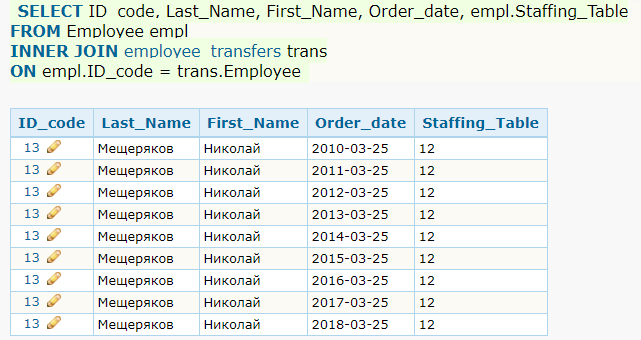


Все отделы:

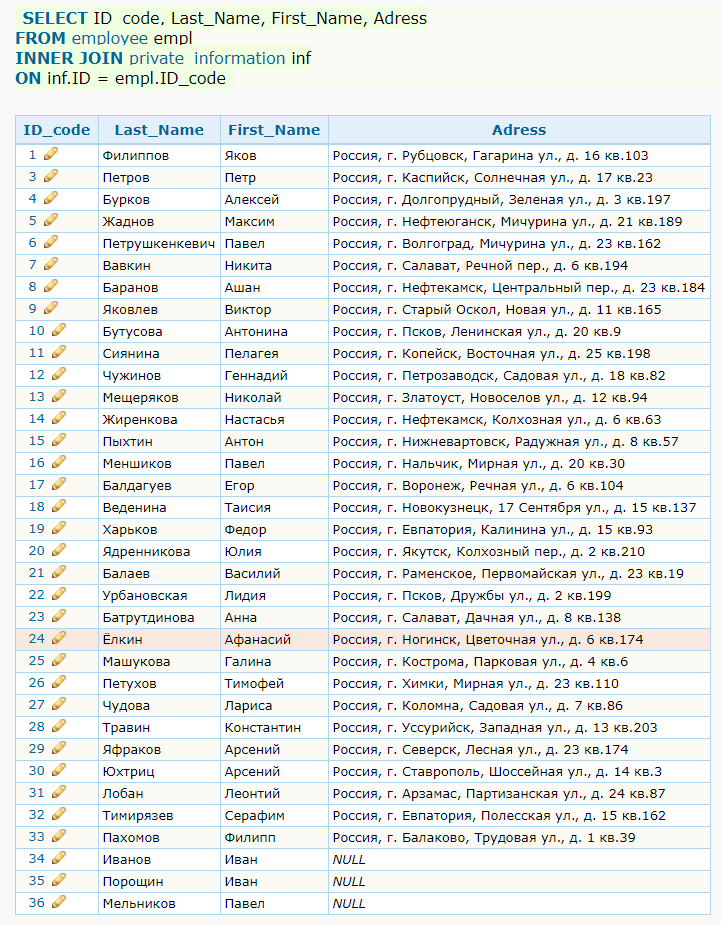


**Запросы, затрагивающие 2 и больше таблицы:**

Вывод истории перемещений работника:

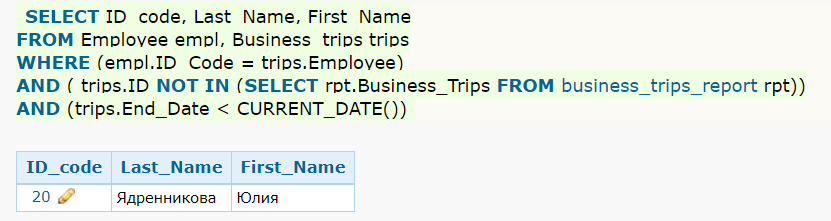


Адрес проживания работника:

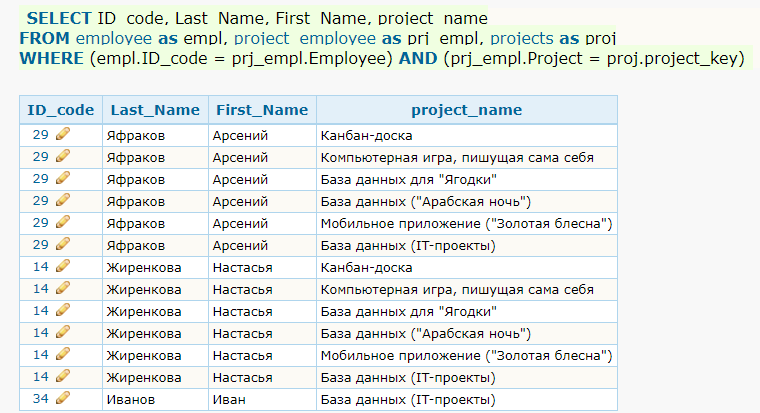


**Запросы, затрагивающие 3 и более таблицы:**

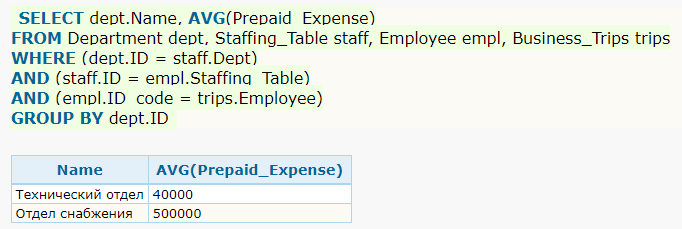
Работники, не отчитавшиеся за командировку:



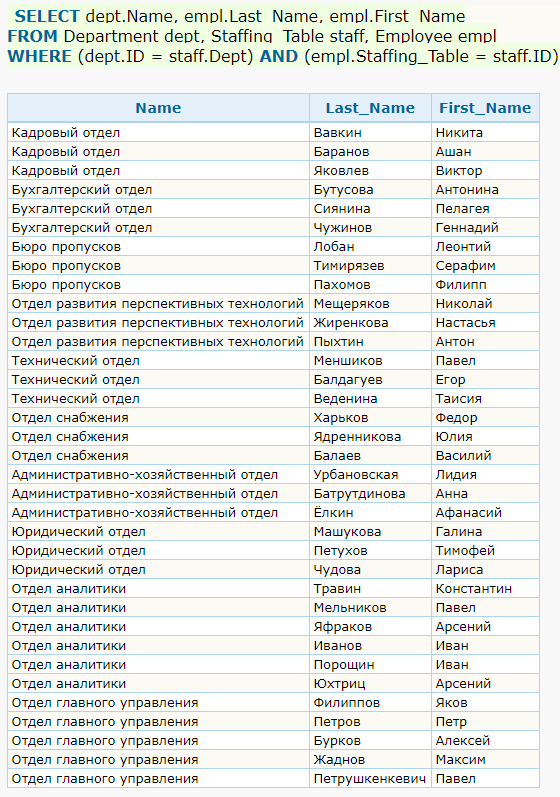
Работники по проектам:



Средняя сумма командировочных отдела:



Работники по отделам:

****

Средний оклад по отделу:



Работники по должностям:



# **Лабораторная работа №4**

***Цель лабораторной работы.***

*Целью лабораторной работы является:*

* изучение возможностей СУБД по управлению информацией в БД;
* освоение SQL-команд семейства DML, процедур экспорта/импорта БД.

***Задание на выполнение лабораторной работы.***

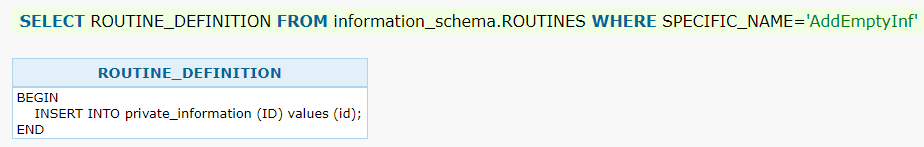
1. Наполнить созданную в лабораторной работе №2 схему данными в соответствии с актуальным состоянием предметной области.

2. Разработать процедуры и пользовательский интерфейс для дополнения, удаления, модификации (изменения) данных.

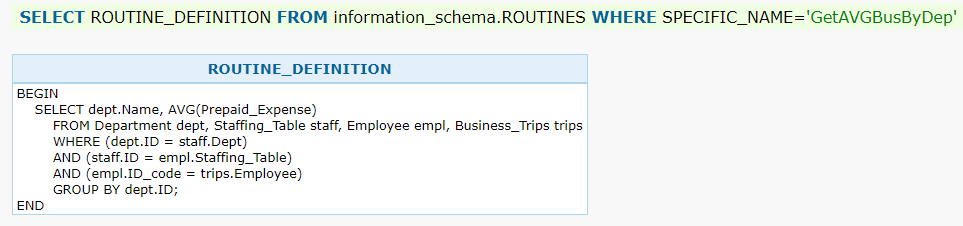
3. Разработать процедуры и пользовательский интерфейс для экспорта и импорта данных БД.

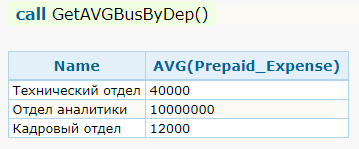
**Процедуры**

Процедура добавление пустого кортежа в таблицу с личной информацией работника:

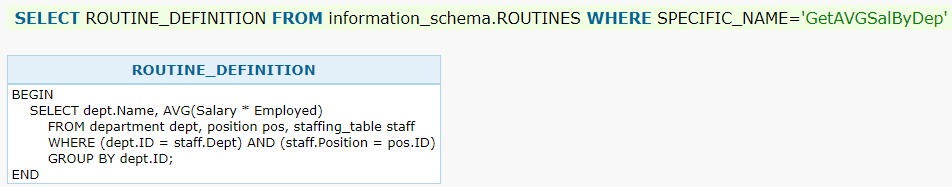


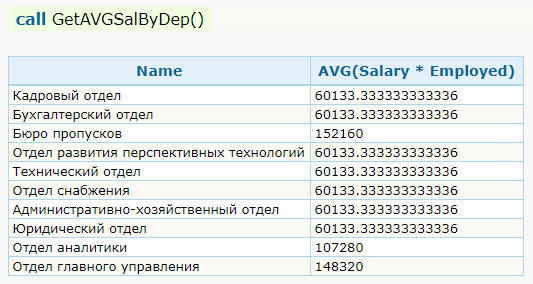
Процедура обработки данных для отчета о средних командировочных по отделам:



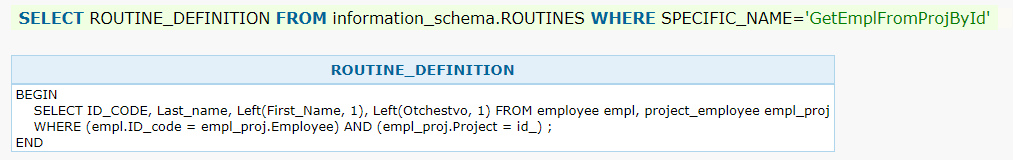


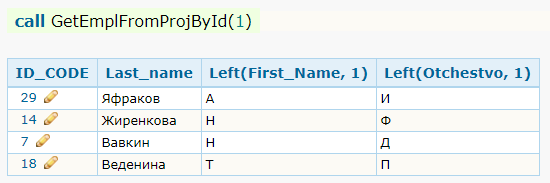
Процедура обработки данных для отчета о средней зарплате по отделам:



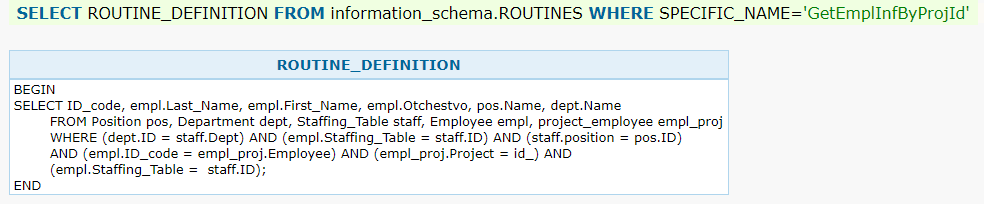


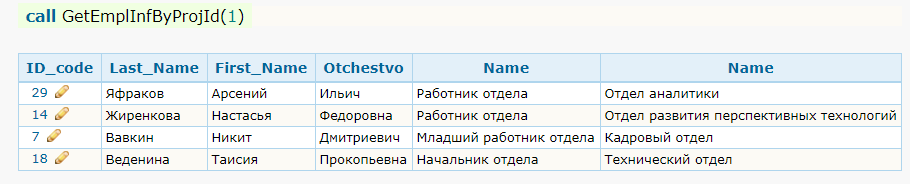
Процедура для получения работников на проекте по id проекта:



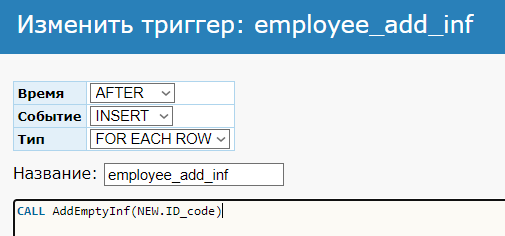


Процедура для получения работников, и их информации, на проекте по id проекта:

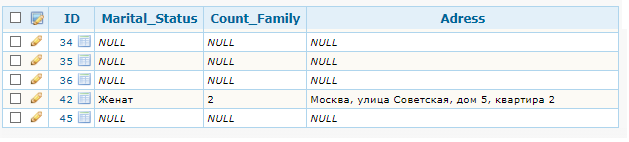




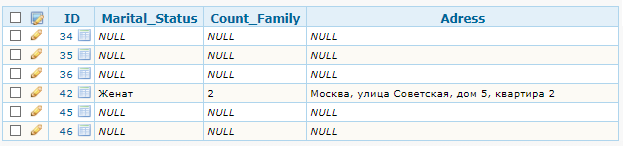
**Триггеры**

Триггер для создания пустой строки в таблице информации при добавлении новой строки в таблице с работниками: 

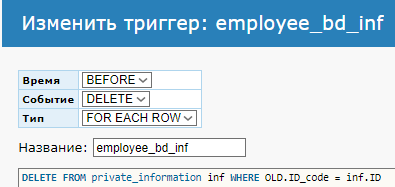
До добавления работника:



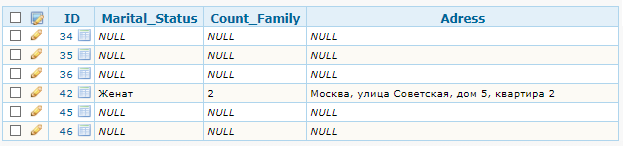
После добавления работника:



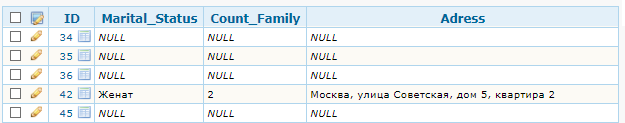
Триггер для удаления строки в таблице информации при удалении строки в таблице с работниками:



До удаления работника:



После удаления работника:



# **Лабораторная работа №5**

***Цель лабораторной работы.***

Целью лабораторной работы является изучение возможностей СУБД по формированию отчетов по информации из БД.

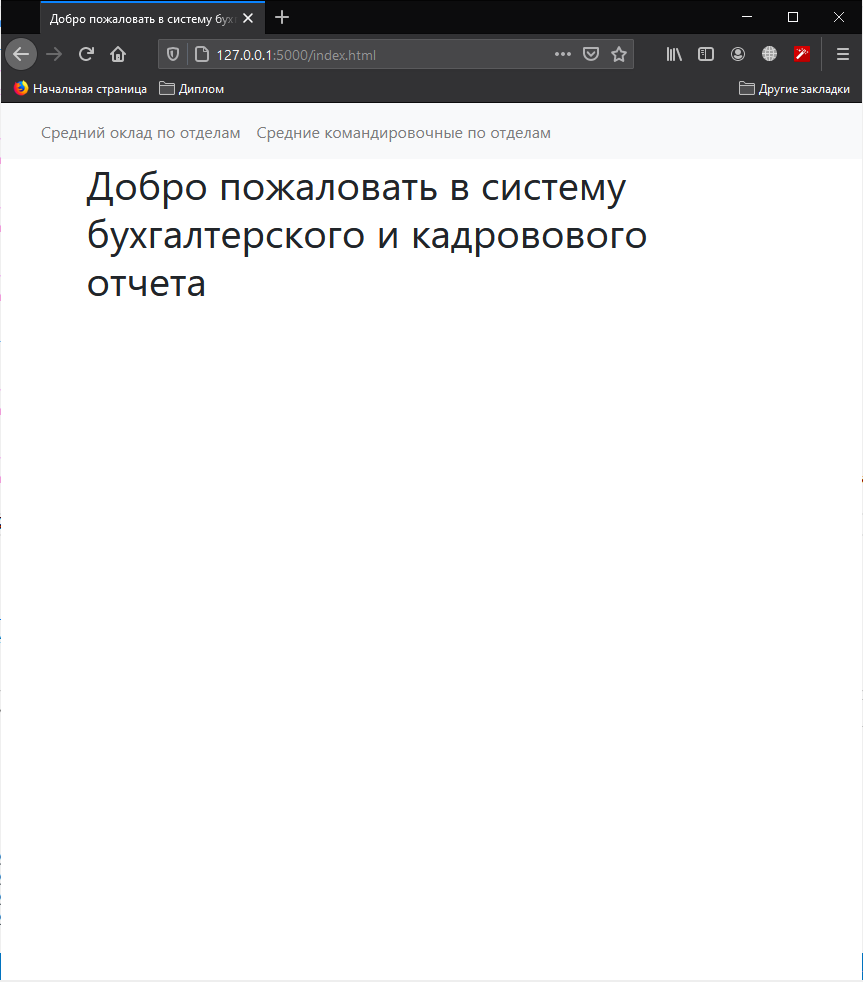
***Задание на выполнение лабораторной работы.***

Сформулировать запросы к Базе данных, наполненной на лабораторной работе №3 для формирования отчетов в соответствии с вариантом, или использовать экспорт данных.

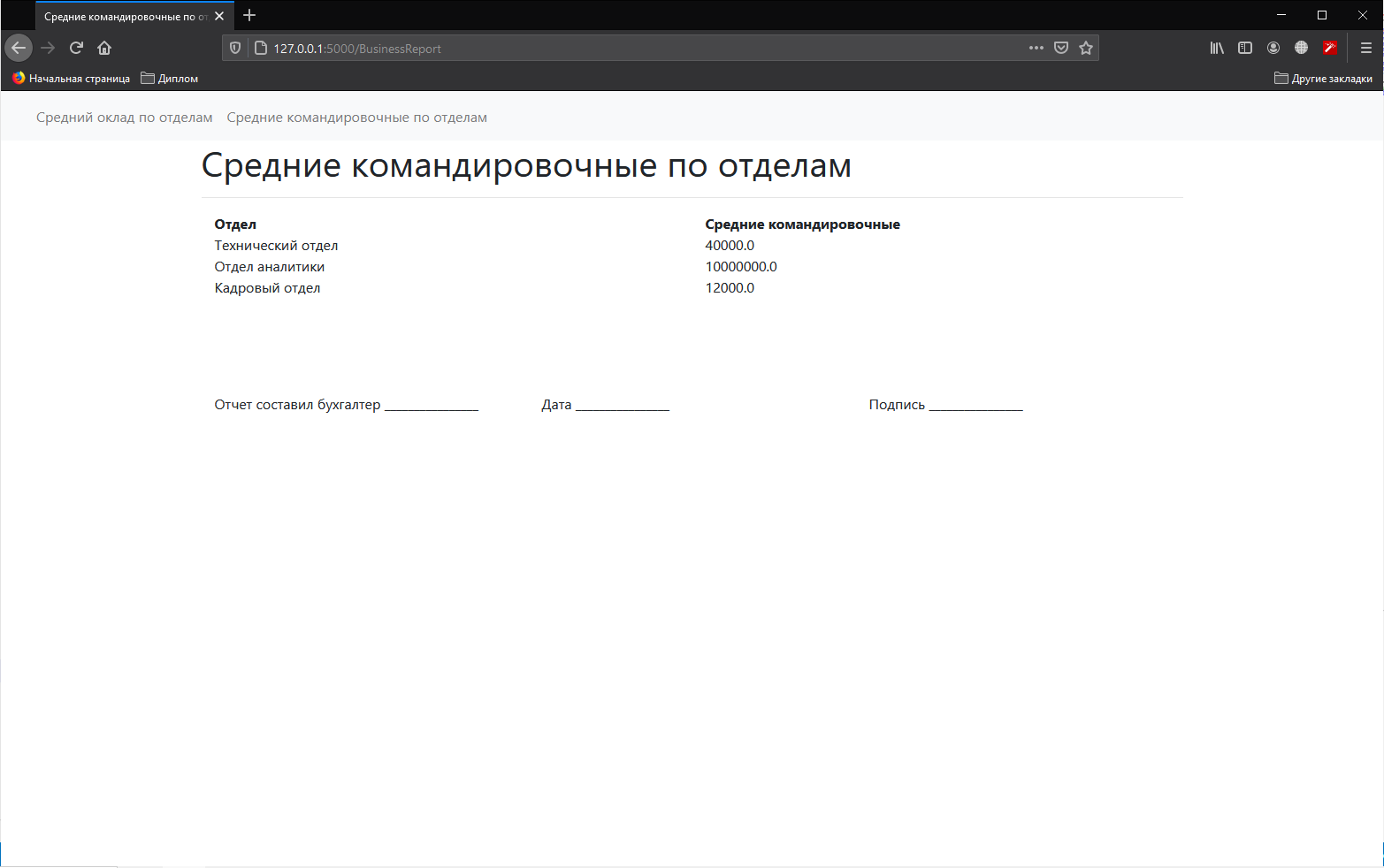
Формы отчетных документов (форм) должны отличаться от стандартных таблиц (извлечений).

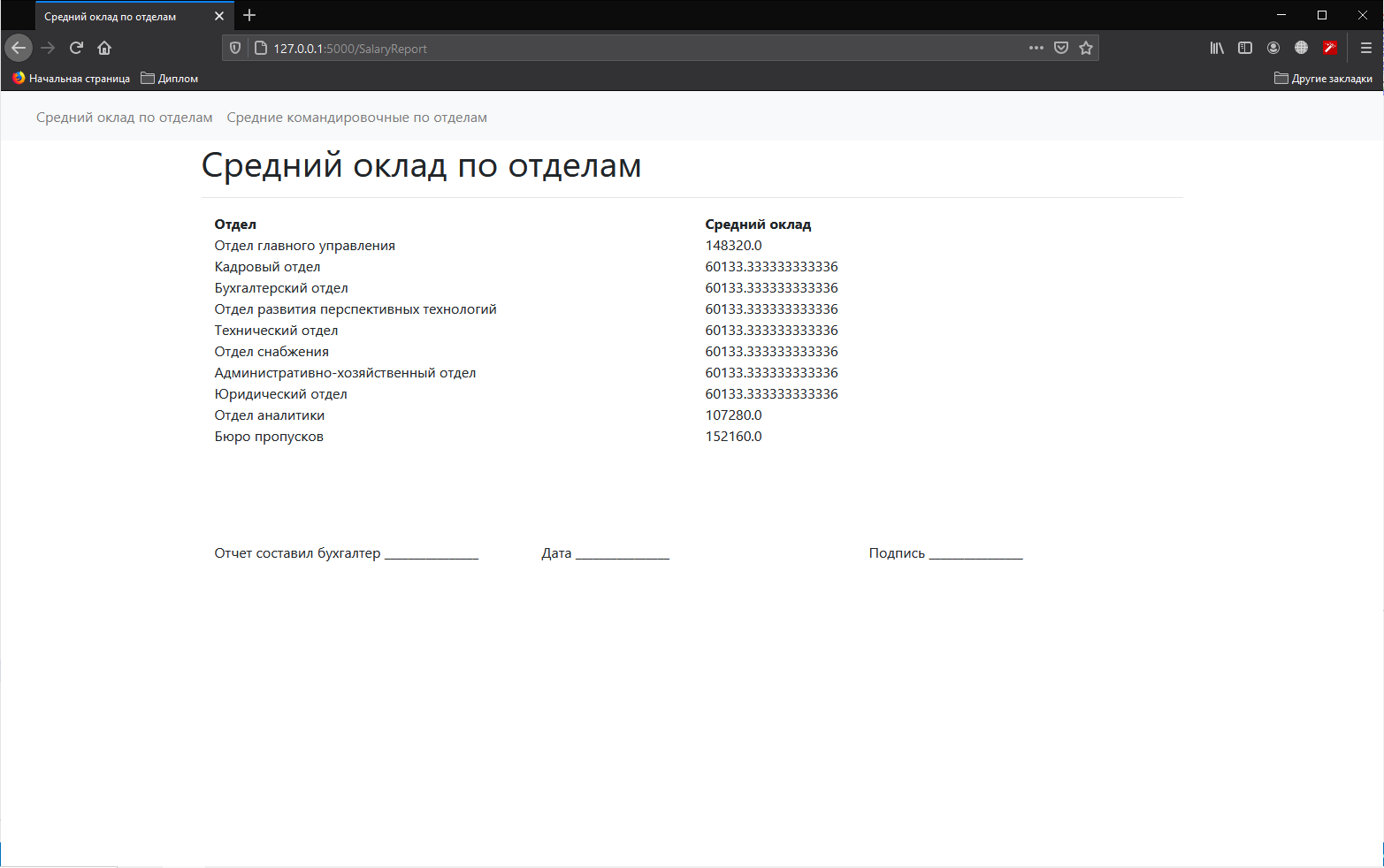
Чтобы отобразить выходные документы (средние затраты на зарплату и командировочные) разработаем пользовательский интерфейс при помощи фреймворка Фласк (Flask).

Авторизовавшись как бухгалтер, нам будет предоставлен доступ к отчетам:



Щелкая по кнопкам («Средний оклад по отделам» и «Средние командировочные по отделам») бухгалтер получит соответствующие отчеты:





#--------------------Отчеты----------------------------

@app.route('/')

def GetAVGSalByDep():

    cur = mysql.connection.cursor()

    cur.execute("call GetAVGSalByDep()")

    sal = cur.fetchall()

    cur.close()

    return sal

@app.route('/')

def GetAVGBusByDep():

    cur = mysql.connection.cursor()

    cur.execute("call GetAVGBusByDep()")

    sal = cur.fetchall()

    cur.close()

    return sal

@app.route('/SalaryReport')

def GetAVGSalByDepReport():

    sal = GetAVGSalByDep()

    return render\_template('SalaryReport/list.html', Results=sal, Priviliges = priv)

@app.route('/BusinessReport')

def GetAVGBusByDepReport():

    res = GetAVGBusByDep()

    return render\_template('BusinessReport/list.html', Results=res, Priviliges = priv)

#--------------------------------------------------------------

# **Лабораторная работа №6**

***Цель лабораторной работы.***

Целью лабораторной работы является изучение методов увеличения производительности операций выборки путём создания индексов, процедур, триггеров.

***Задание на выполнение лабораторной работы.***

1. Выполнить оптимизацию работы разработанной Базы данных путём создания специфических объектов.

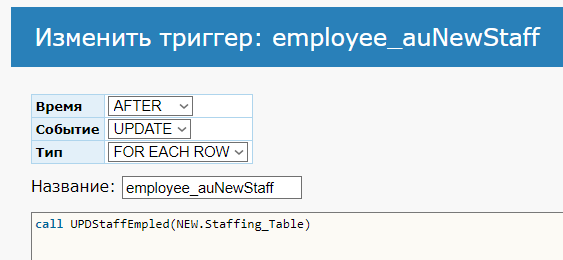
2. Разработать процедуру для реализации пользовательских запросов.

3. Разработать триггеры для контроля ограничений целостности в соответствии с предметной областью и поддерживаемым процессом (с учетом семантики процесса).

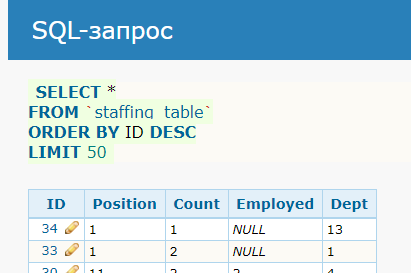
4. Необходимо, чтобы в работе присутствовало количество триггеров, соответствующее потребностям реализуемой БД, но не менее двух. В частности, с помощью триггеров должен быть реализован контроль вводимых в БД чисел (положительные, отрицательные и т.д.), исходя из варианта задания.

Для оптимизации БД создадим триггеры:

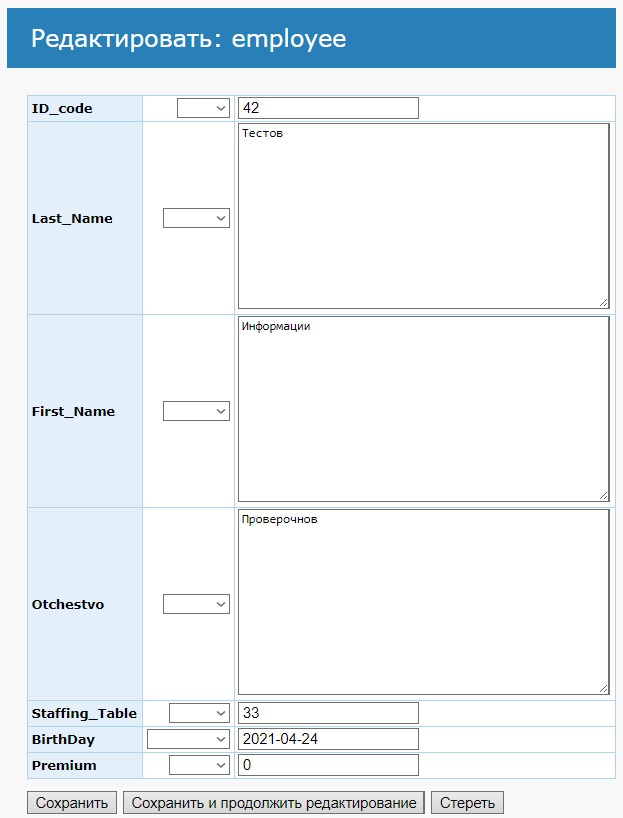
Триггер, подсчитывающий заново количество нанятых работников при добавлении нового работника в штатное расписание:



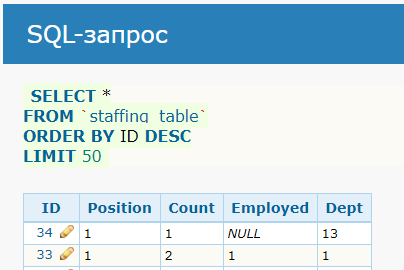
Исходное состояние таблицы:



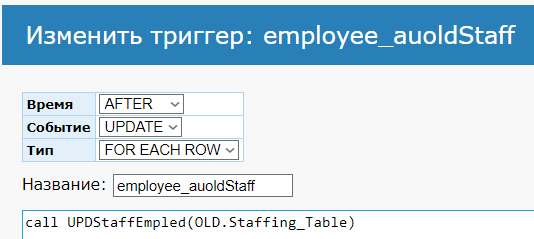
Добавим работника в штатное расписание с ID 33:



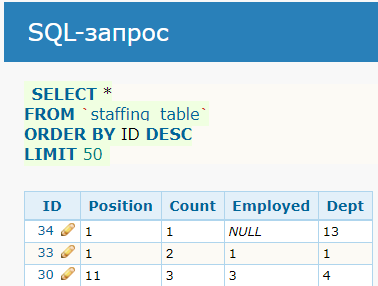
Проверим:

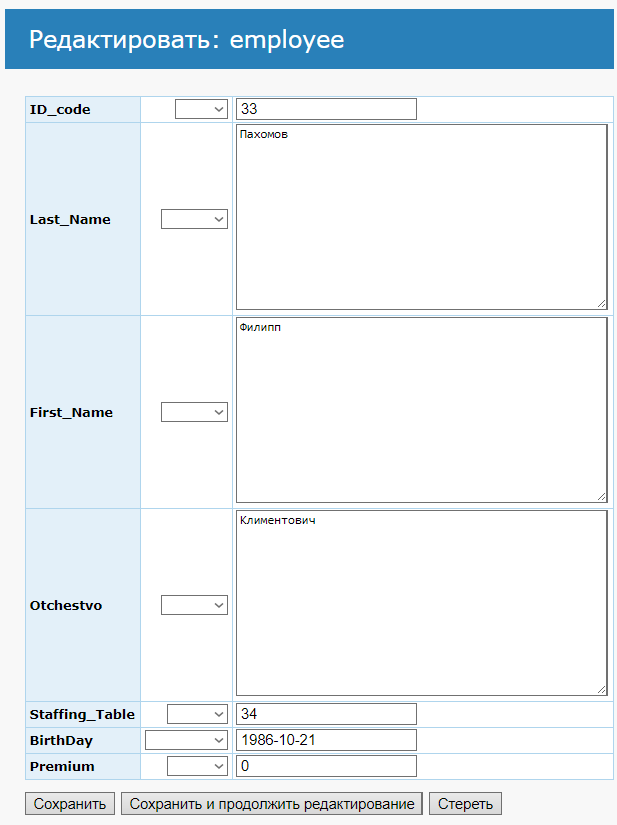
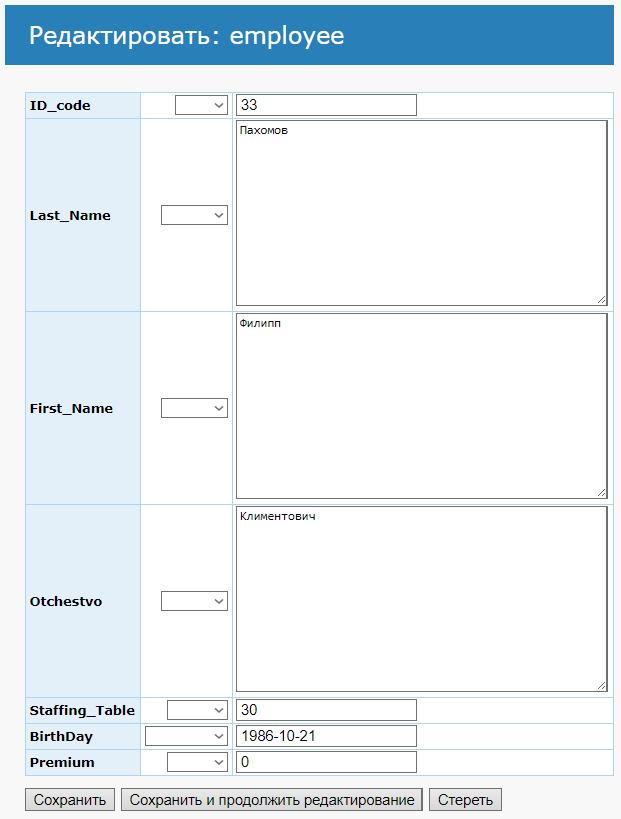


Триггер, подсчитывающий заново количество нанятых работников при удалении работника из штатного расписания:



Исходное состояние таблицы:



Присвоим одному из работников, привязанных к штатному расписанию с ID 30, штатное расписание с ID 34:

Проверим:

