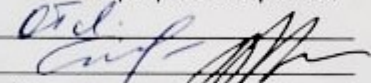


ФЕДЕРАЛЬНОЕ АГЕНТСТВО ВОЗДУШНОГО ТРАНСПОРТА  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ  
БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ ГРАЖДАНСКОЙ АВИАЦИИ» (МГТУ ГА)

Кафедра ПМ

Курсовая работа  
защищена с оценкой



(подпись преподавателя, дата)

21.04.2021

КУРСОВАЯ РАБОТА  
по дисциплине «Базы данных»  
Вариант № 8

Тема: «Создание базы данных и информационной системы кадрового и  
бухгалтерского учета»

Выполнил студент группы ПМб-4-1

Фейзуллин К.М.

(Ф.И.О.)

Руководитель:

д.т.н., доцент, профессор каф. ПМ

Егорова А.А.

(звание, степень, Ф.И.О.)

МОСКВА – 2021

## Содержание

Цель курсовой работы .....	3
Задание на выполнение курсовой работы .....	3
Задание на выполнение курсовой работы .....	4
Введение .....	6
Формирование SQL-запросов .....	26
Создание процедур .....	33
Создание триггеров .....	35
Разработка интерфейса .....	37
Руководство пользователя системы .....	38
Руководство администратора системы .....	51
Приложение .....	57

## **Цель курсовой работы**

Целью курсовой работы является освоение основных приемов:

- работы по проектированию информационных систем;
- работы по применению современных методологий структурного анализа и проектирования;
- разработки реляционных баз данных заданной структуры;
- реализации типовых процедур СУБД для обработки данных с использованием языка высокого уровня и СУБД;
- оптимизации баз данных;
- разработки пользовательского интерфейса и созданию выходных форм.

## **Задание на выполнение курсовой работы**

Создать базу данных, содержащую сведения о заданном объекте (в соответствии с вариантом) и разработать интерактивную систему для обработки данных базы. Система должна обеспечивать:

- ведение БД: просмотр, корректировку, удаление, дополнение;
- контроль ограничения целостности (ограничение значений атрибутов и структурные ограничения);
- использование нескольких пользователей в СУБД (минимум – администратор и пользователь) и соответственно программный интерфейс с разными уровнями доступа (например, администратор имеет право на все - создание, удаление, изменение, извлечение, а пользователь - только на извлечение);
- формирование выходных форм в соответствии с заданиями;
- извлечение информации по запросам пользователей в соответствии с бизнес-правилами и функционалом системы.

## Задание на выполнение курсовой работы

### 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 ЕТС, 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.
- Удалить можно только отделы без штатного расписания, а следовательно и без работников.

## **Введение**

В век информационных технологий и огромных объемов данных возникает серьезная задача структурирования, контроля, администрирования информации. Для этого данные собираются в базы данных. Но тогда возникает следующая задача – удобное управление этим массивом данных, для чего и были созданы специальные программные комплексы типа систем управления базами данных (СУБД).

Актуальность данной работы заключается в том, что созданная база данных позволяет удобно вести кадровый учет, распределять работников по отделам, учитывать их передвижение внутри компании, назначать работников на проекты и командировки, а также вести бухгалтерскую статистику, формируя отчеты. Все функции удобно реализованы в программном веб-интерфейсе, с учетом бизнес-правил и бизнес-процессов.

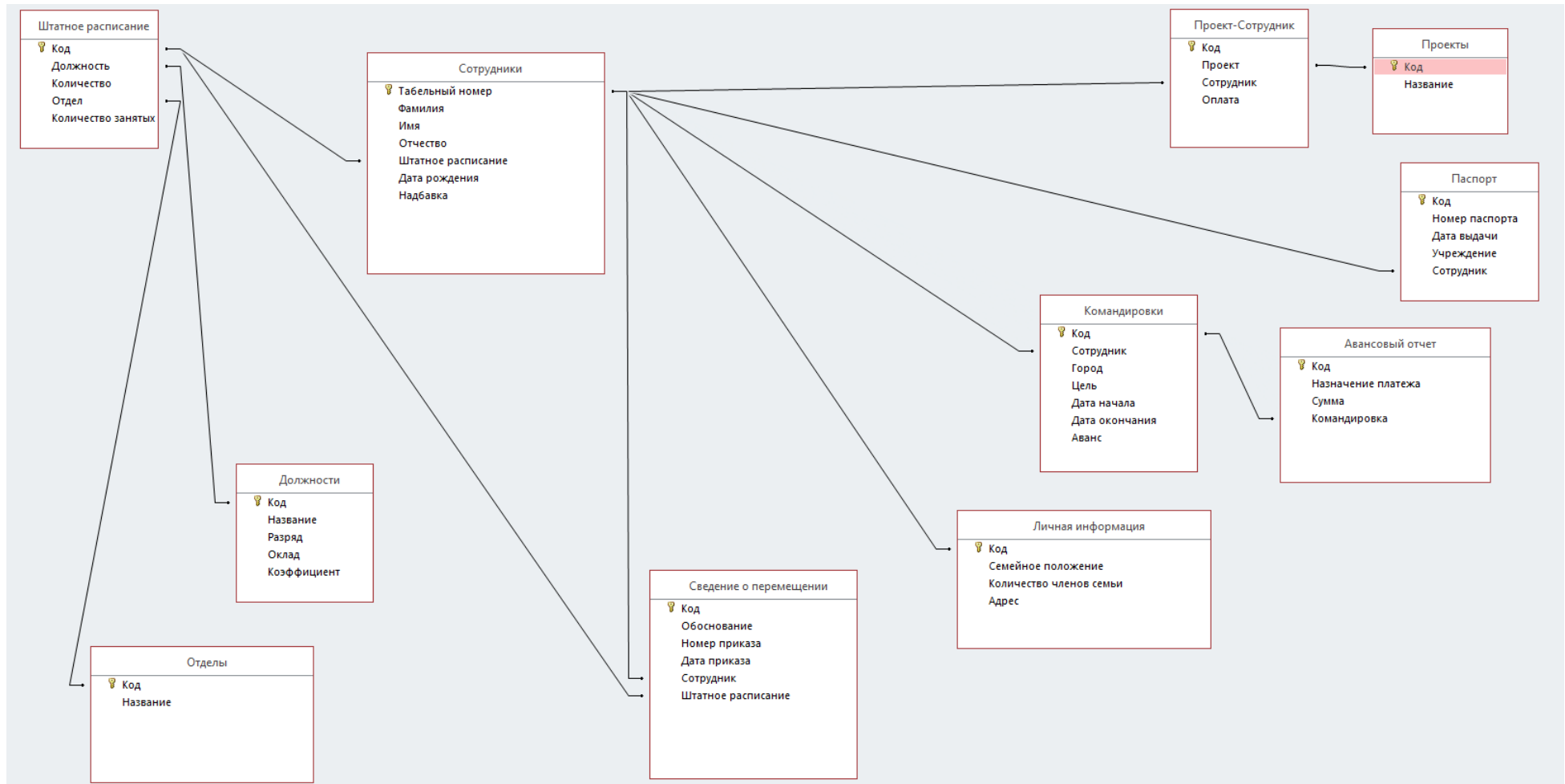
Взаимодействие через веб-интерфейс обусловлено преимуществом, что в отличие от компилируемых приложений, будет работать на абсолютно любых устройствах с установленным современным браузером, а учитывая то, что доступ к базе можно получить из любого места, при наличии интернета, ведь интерфейс работает с сервером, делает это особенно удобным.

Между тем, если реализовать все нетривиальные взаимодействия через процедуры СУБД, то взаимодействие с базой данных будет совершенно обособленно и подконтрольно, при использовании тех самых процедур при реализации интерфейса. Тогда при изменении интерфейса не придется сохранять и где-то хранить многострочные запросы для СУБД на языке SQL.

## 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:

- А. Табельный номер (первичный ключ).
- В. Фамилия.
- С. Имя.
- Д. Отчество.
- Е. Штатное расписание (внешний ключ для сущности «Штатное расписание»).
- Г. Дата рождения.
- Г. Надбавка.

2. Штатное расписание

- А. Код (первичный ключ).
- В. Должность (внешний ключ для сущности «Должности»).
- С. Количество.
- Д. Отдел (внешний ключ для сущности «Отделы»).
- Е. Количество занятых.

3. Должности

- А. Код (первичный ключ).
- В. Название.
- С. Разряд.
- Д. Оклад.
- Е. Коэффициент.

4. Отделы

- А. Код (первичный ключ).
- В. Название.



5. Сведения о перемещении

- A. Код (первичный ключ).
- B. Обоснование - краткое описание того, почему была сменена должность или сменен отдел.
- C. Номер приказа.
- D. Дата приказа
- E. Сотрудник (внешний ключ сущности «Сотрудники»).
- F. Штатное расписание (внешний ключ сущности «Штатное расписание»).

6. Личная информация

- A. Код (первичный ключ).
- B. Семейное положение.
- C. Количество членов семьи.
- D. Адрес.

7. Паспорт

- A. Код (первичный ключ).
- B. Номер паспорта.
- C. Дата выдачи.
- D. Учреждение.
- E. Сотрудник (внешний ключ сущности «Сотрудники»).

8. Командировки

- A. Код (первичный ключ).
- B. Сотрудник (внешний ключ сущности «Сотрудники»).
- C. Город.
- D. Цель
- E. Дата начала.
- F. Дата окончания.
- G. Аванс

## 9. Авансовый отчет

- А. Код (первичный ключ).
- В. Назначение платежа.
- С. Сумма.
- Д. Командировка (внешний ключ сущности «Командировка»).
- Е. Дата сдачи авансового отчета.

## 10. Проект – сотрудник (сущность – связка)

- А. Код (первичный ключ).
- В. Проект (внешний ключ сущности «Проекты»).
- С. Сотрудник (внешний ключ сущности «Сотрудники»)
- Д. Оплата.

## 11. Проекты

- А. Код (первичный ключ).
- В. Название.

Стоит отметить, что все атрибуты данной БД имеют русскоязычные названия для удобства проверки. В дальнейшем они получают 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 A (Табельный номер), следовательно, данная таблица находится в 2НФ. Также внутри отношения нет транзитивных функциональных связей между атрибутами (ни

одно из не ключевых полей В, С, D, E, F, G не зависит функционально от любого другого не ключевого поля), что говорит о том, что отношение R находится в 3НФ.

## 2. Штатное расписание

$$R = (A, B, C, D, E).$$

Каждое из полей В, С, 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 В, С, D, E не зависит функционально от любого другого не ключевого поля), что говорит о том, что отношение R находится в 3НФ.

## 3. Должности

$$R = (A, B, C, D, E).$$

Каждое из полей В, С, 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 В, С, D, E не зависит функционально от любого другого не ключевого поля), что говорит о том, что отношение R находится в 3НФ.

## 4. Отделы

$$R = (A, B).$$

Поле В (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.

## 5. 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 A (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.

#### 6. Личная информация

$$R = (A, B, C, D).$$

Каждое из полей B, C, D (не являющееся первичным ключом) функционально зависит от первичного ключа - поля A (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.

#### 7. Паспорт

$$R = (A, B, C, D, E).$$

Каждое из полей B, C, D, E (не являющееся первичным ключом) функционально зависит от первичного ключа - поля A (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.

#### 8. 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 A (Код), следовательно,

данная таблица находится в 2НФ. Также внутри отношения нет транзитивных функциональных связей между атрибутами (ни одно из не ключевых полей В, С, D, E, F, G не зависит функционально от любого другого не ключевого поля), что говорит о том, что отношение R находится в 3НФ.

#### 9. Авансовый отчет

$R = (A, B, C, D, E).$

Каждое из полей В, С, 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 В, С, D, E не зависит функционально от любого другого не ключевого поля), что говорит о том, что отношение R находится в 3НФ.

#### 10. Проект - сотрудник

$R = (A, B, C, D).$

Каждое из полей В, С, 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 В, С, 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 (В, С), являющиеся первичными в сущностях «Проекты» и «Сотрудники» соответственно.

#### 11. Проекты

$R = (A, B).$

Поле В (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

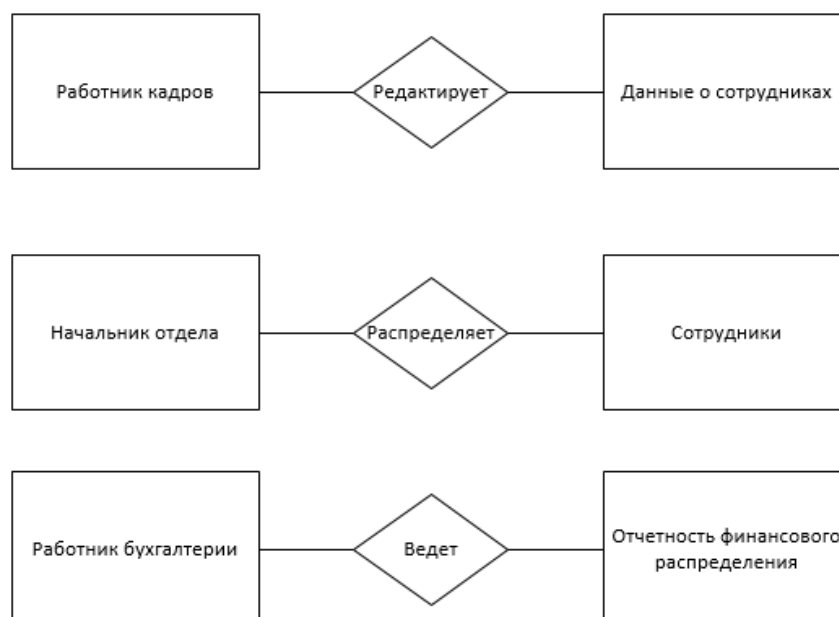
Роли:

- Работник кадрового учета
- Начальник отдела
- Работник бухгалтерского отдела

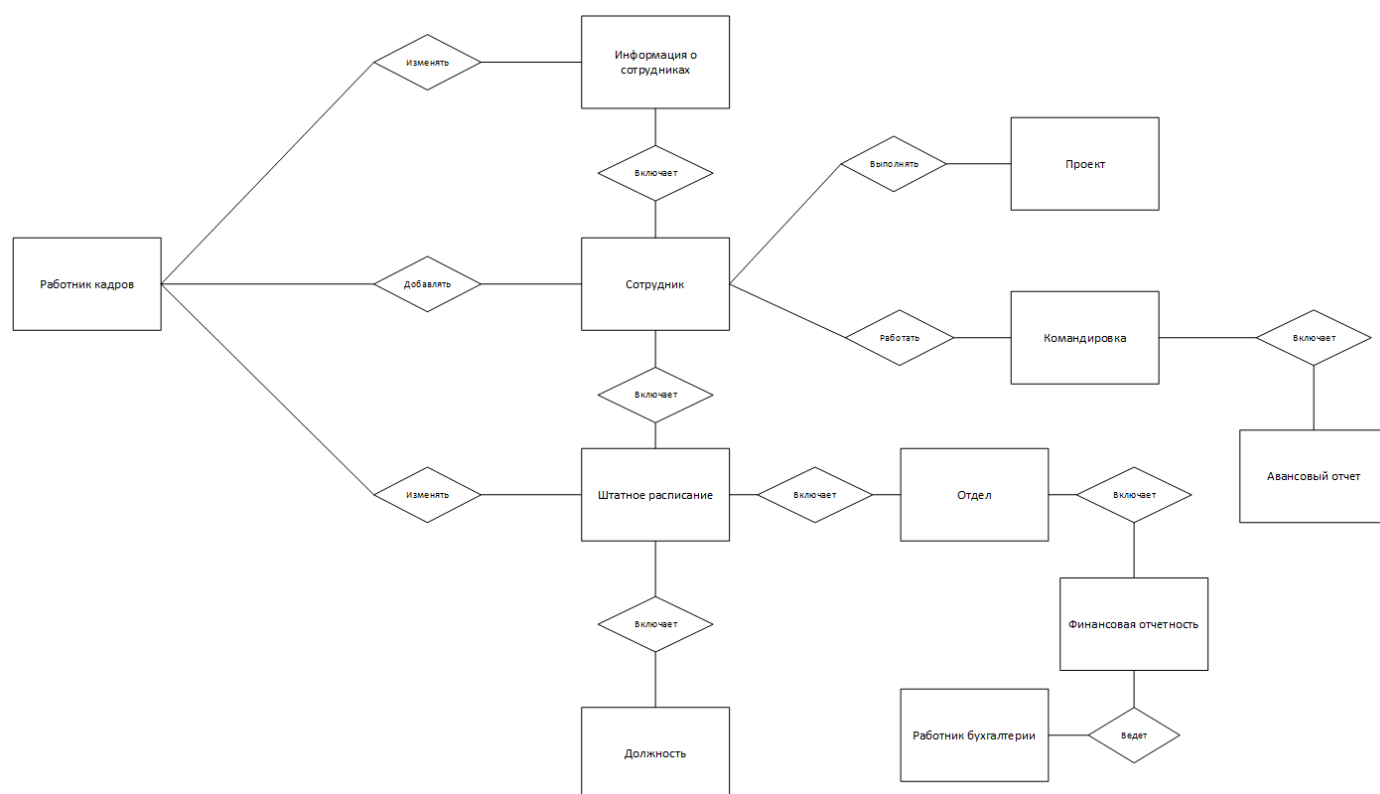
Процесс:

- 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:



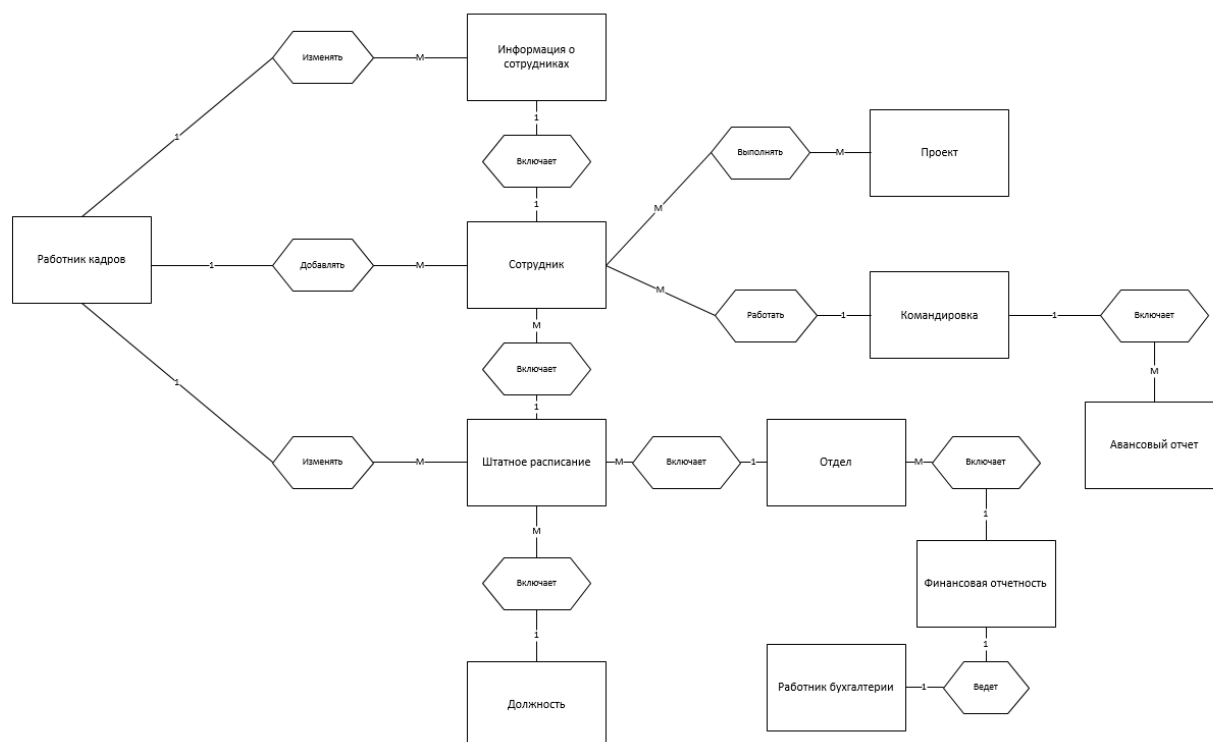
Составим более подробную схему, включающую работника отдела кадров, добавляющего сотрудников и их информацию, штатное расписание и должности, а также отделы. Так как начальник отдела по сути своей является сущностью «Сотрудник», его на схеме указывать не будем. Данную схему уже можно назвать концептуальной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





## Инфологическая модель данных

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



## Наполнение базы данных

Командировки:

**Выбрать: business\_trips**

**Выбрать**

Показать структуру

Изменить таблицу

Новая запись

Выбрать

Поиск

Сортировать

Лимит  
50

Длина текста  
100

Действие  
Выбрать

**SELECT \* FROM `business\_trips` LIMIT 50** (0.001 s) Редактировать

<input type="checkbox"/>		ID	Employee	City	Target	Start_Date	End_Date	Prepaid_Expense
<input type="checkbox"/>		1	17	Саратов	Сертификация Оборудования	2011-02-21	2011-03-04	40000
<input type="checkbox"/>		2	17	Самара	Сертификация Оборудования	2011-04-05	2011-04-29	40000
<input type="checkbox"/>		3	17	Ростов	Сертификация Оборудования	2012-06-13	2012-06-27	40000
<input type="checkbox"/>		4	17	Краснодар	Сертификация Оборудования	2013-09-01	2013-09-15	40000
<input type="checkbox"/>		5	17	Адлер	Сертификация Оборудования	2013-12-15	2014-01-02	40000
<input type="checkbox"/>		6	17	Санкт-Петербург	Сертификация Оборудования	2014-02-02	2014-02-28	40000
<input type="checkbox"/>		7	17	Магадан	Сертификация Оборудования	2014-06-13	2014-06-30	40000
<input type="checkbox"/>		8	17	Мурманск	Сертификация Оборудования	2014-08-03	2014-08-11	40000
<input type="checkbox"/>		9	17	Севастополь	Сертификация Оборудования	2014-11-12	2014-12-01	40000
<input type="checkbox"/>		10	17	Владивосток	Сертификация Оборудования	2015-01-29	2014-02-28	40000
<input type="checkbox"/>		11	20	Майями	Гуманитарная помощь	2012-06-13	2013-09-15	500000

Авансовый отчет:

**Выбрать: business\_trips\_report**

**Выбрать**

Показать структуру

Изменить таблицу

Новая запись

Выбрать

Поиск

Сортировать

Лимит  
50

Длина текста  
100

Действие  
Выбрать

**SELECT \* FROM `business\_trips\_report` LIMIT 50** (0.002 s) Редактировать

<input type="checkbox"/>		ID	Date	Purpose_Payment	Value	Business_Trips
<input type="checkbox"/>		1	2011-03-04	На бытовые расходы	20000	1
<input type="checkbox"/>		2	2011-04-29	На бытовые расходы	20000	2
<input type="checkbox"/>		3	2012-06-27	На бытовые расходы	20000	3
<input type="checkbox"/>		4	2013-09-15	На бытовые расходы	20000	4
<input type="checkbox"/>		5	2014-01-02	На бытовые расходы	20000	5
<input type="checkbox"/>		6	2014-02-28	На бытовые расходы	30000	6
<input type="checkbox"/>		7	2014-06-30	На бытовые расходы	25000	7
<input type="checkbox"/>		8	2014-08-11	На бытовые расходы	15000	8
<input type="checkbox"/>		9	2014-12-01	На бытовые расходы	25000	9
<input type="checkbox"/>		10	2014-02-28	На бытовые расходы	40000	10

Отделы:

**Выбрать: department**

Выбрать

Поиск

Сортировать

Лимит  
50

Длина текста  
100

Действие  
Выбрать

Выбрать

Поиск

Сортировать

Лимит  
50

Длина текста  
100

Действие  
Выбрать

**SELECT \* FROM `department` LIMIT 50** (0.002 s) Редактировать

<input type="checkbox"/>		ID	Name
<input type="checkbox"/>		1	Кадровый отдел
<input type="checkbox"/>		3	Бухгалтерский отдел
<input type="checkbox"/>		4	Бюро пропусков
<input type="checkbox"/>		5	Отдел развития перспективных технологий
<input type="checkbox"/>		6	Технический отдел
<input type="checkbox"/>		7	Отдел снабжения
<input type="checkbox"/>		8	Административно-хозяйственный отдел
<input type="checkbox"/>		9	Юридический отдел
<input type="checkbox"/>		10	Отдел аналитики
<input type="checkbox"/>		11	Отдел главного управления

Перемещения работника:

**Выбрать: employee\_transfers**

Выбрать

Поиск

Сортировать

Лимит  
50

Длина текста  
100

Действие  
Выбрать

Выбрать

Поиск

Сортировать

Лимит  
50

Длина текста  
100

Действие  
Выбрать

**SELECT \* FROM `employee\_transfers` LIMIT 50** (0.001 s) Редактировать

<input type="checkbox"/>		ID	Justification	Order_Number	Order_Date	Employee	Staffing_Table
<input type="checkbox"/>		1	Перевод по заявлению	ПР0000001	2010-03-25	13	15
<input type="checkbox"/>		2	Перевод по заявлению	ПР0000002	2011-03-25	13	12
<input type="checkbox"/>		3	Перевод по заявлению	ПР0000003	2012-03-25	13	15
<input type="checkbox"/>		4	Перевод по заявлению	ПР0000004	2013-03-25	13	12
<input type="checkbox"/>		6	Перевод по заявлению	ПР0000005	2014-03-25	13	15
<input type="checkbox"/>		7	Перевод по заявлению	ПР0000006	2015-03-25	13	12
<input type="checkbox"/>		8	Перевод по заявлению	ПР0000007	2016-03-25	13	15
<input type="checkbox"/>		9	Перевод по заявлению	ПР0000008	2017-03-25	13	12
<input type="checkbox"/>		16	Перевод по заявлению	ПР0000009	2018-03-25	13	15

Работники:

## Выбрать: employee

Выбрать

Показать структуру

Изменить таблицу

Новая запись

Выбрать

Поиск

Сортировать

Лимит

50

Длина текста

100

Действие

Выбрать

SELECT \* FROM `employee` LIMIT 50

(0.002 s) Редактировать

<input type="checkbox"/>	ID_code	Last_Name	First_Name	Otchestvo	Staffing_Table	BirthDay	Premium
<input type="checkbox"/>	1	Филиппов	Яков	Михалыч	1	1966-03-19	100000
<input type="checkbox"/>	3	Петров	Петр	Петрович	2	1989-03-19	0
<input type="checkbox"/>	4	Бурков	Алексей	Семенович	3	1990-03-19	NULL
<input type="checkbox"/>	5	Жаднов	Максим	Максимович	4	1990-11-19	NULL
<input type="checkbox"/>	6	Петрушкенкевич	Павел	Валерьевич	5	1990-06-19	NULL
<input type="checkbox"/>	7	Вавкин	Никита	Дмитриевич	6	1999-10-25	NULL
<input type="checkbox"/>	8	Баранов	Ашан	Михайлович	7	1995-10-25	NULL
<input type="checkbox"/>	9	Яковлев	Виктор	Викторович	8	1976-03-19	NULL
<input type="checkbox"/>	10	Бутусова	Антонина	Ильишна	9	1987-01-27	NULL
<input type="checkbox"/>	11	Сиянина	Пелагея	Афанасьевна	10	1991-04-01	NULL
<input type="checkbox"/>	12	Чужинов	Геннадий	Всеволодович	11	1983-02-16	NULL
<input type="checkbox"/>	13	Мещеряков	Николай	Афанасьевич	12	1980-11-07	NULL
<input type="checkbox"/>	14	Жиренкова	Настасья	Федоровна	13	1977-11-03	NULL
<input type="checkbox"/>	15	Пыхтин	Антон	Петрович	14	1966-10-01	NULL
<input type="checkbox"/>	16	Меншиков	Павел	Кириллович	15	1994-06-18	NULL
<input type="checkbox"/>	17	Балдагуев	Егор	Иннокентиевич	16	1974-04-02	NULL
<input type="checkbox"/>	18	Веденина	Таисия	Прокопьевна	17	1987-05-20	NULL
<input type="checkbox"/>	19	Харьков	Федор	Никитьевич	18	1986-05-11	NULL
<input type="checkbox"/>	20	Ядренникова	Юлия	Тимофеевна	19	1970-06-21	NULL
<input type="checkbox"/>	21	Балаев	Василий	Трофимович	20	1973-09-15	NULL
<input type="checkbox"/>	22	Урбановская	Лидия	Михайловна	21	1976-04-03	NULL
<input type="checkbox"/>	23	Батрутдинова	Анна	Николаевна	22	1989-04-16	NULL
<input type="checkbox"/>	24	Ёлкин	Афанасий	Аркадьевич	23	1971-09-04	NULL
<input type="checkbox"/>	25	Машукова	Галина	Данииловна	24	1971-10-20	NULL
<input type="checkbox"/>	26	Петухов	Тимофей	Филиппович	25	1969-12-10	NULL
<input type="checkbox"/>	27	Чудова	Лариса	Семеновна	26	1982-03-25	NULL
<input type="checkbox"/>	28	Травин	Константин	Григорьевич	27	1972-03-27	NULL
<input type="checkbox"/>	29	Яфраков	Арсений	Ильич	28	1980-08-15	NULL
<input type="checkbox"/>	30	Юхтриц	Арсений	Аркадьевич	29	1964-12-23	NULL
<input type="checkbox"/>	31	Лобан	Леонтий	Лукьевич	30	1994-05-09	NULL
<input type="checkbox"/>	32	Тимирязев	Серафим	Максимович	30	1992-12-20	NULL
<input type="checkbox"/>	33	Пахомов	Филипп	Климентович	30	1986-10-21	NULL
<input type="checkbox"/>	34	Иванов	Иван	Иванович	28	1990-03-19	NULL
<input type="checkbox"/>	35	Порошин	Иван	Акулович	28	1990-06-19	NULL
<input type="checkbox"/>	36	Мельников	Павел	Степанович	27	1989-09-19	NULL

20

## Паспортные данные:

### Выбрать: passport

Выбрать

Показать структуру

Изменить таблицу

Новая запись

Выбрать

Поиск

Сортировать

Лимит  
50

Длина текста  
100

Действие  
Выбрать



SELECT \* FROM `passport` LIMIT 50 (0.002 s)

Редактировать

<input type="checkbox"/>	ID	Number	Issue_Date	Institution	Employee
<input type="checkbox"/>	1	4 286 688 611	2017-01-20	ОВД России по г. Златоуст	1
<input type="checkbox"/>	2	4 850 107 025	2009-10-20	Отделом УФМС России по г. Брянск	3
<input type="checkbox"/>	3	4 860 448 885	2010-02-20	Отделом внутренних дел России по г. Жуковский	4
<input type="checkbox"/>	4	4 557 640 710	2010-02-20	Отделом УФМС России по г. Салават	5
<input type="checkbox"/>	5	4 946 336 648	2010-02-20	ОВД России по г. Подольск	6
<input type="checkbox"/>	6	4 493 530 389	2010-04-20	Отделением УФМС России по г. Новороссийск	7
<input type="checkbox"/>	7	4 812 935 119	2015-06-20	Управление внутренних дел по г. Улан-Удэ	8
<input type="checkbox"/>	8	4 765 493 685	2021-02-20	Отделом внутренних дел России по г. Ростов-на-Дону	9
<input type="checkbox"/>	9	4 389 127 421	2007-02-20	Отделением УФМС России по г. Невинномысск	10
<input type="checkbox"/>	10	4 034 620 239	2011-06-20	Отделением УФМС России по г. Калуга	11
<input type="checkbox"/>	11	4 740 483 196	2003-01-20	Отделом УФМС России по г. Нижний Тагил	12
<input type="checkbox"/>	12	4 117 136 723	2000-12-20	ОУФМС России по г. Владикавказ	13
<input type="checkbox"/>	13	4 710 485 094	1997-08-20	Отделом внутренних дел России по г. Омск	14
<input type="checkbox"/>	14	4 647 273 253	2011-01-20	Отделением УФМС России по г. Таганрог	15
<input type="checkbox"/>	15	4 084 771 129	2014-08-20	Управление внутренних дел по г. Рубцовск	16
<input type="checkbox"/>	16	4 196 536 534	2019-08-20	Отделением УФМС России по г. Магнитогорск	17
<input type="checkbox"/>	17	4 490 514 278	2007-02-20	ОУФМС России по г. Евпатория	18
<input type="checkbox"/>	18	4 971 551 987	2006-05-20	Отделом внутренних дел России по г. Набережные Челны	19
<input type="checkbox"/>	19	4 419 913 817	2020-04-20	Отделом УФМС России по г. Курск	20
<input type="checkbox"/>	20	4 046 862 375	2018-01-20	ОУФМС России по г. Старый Оскол	21
<input type="checkbox"/>	21	4 298 402 399	2021-11-20	Отделом УФМС России по г. Железнодорожск	22
<input type="checkbox"/>	22	4 893 256 225	2009-08-20	Отделом УФМС России по г. Саратов	23
<input type="checkbox"/>	23	4 041 550 579	2016-01-20	Отделом УФМС России по г. Абакан	24
<input type="checkbox"/>	24	4 345 721 146	2016-02-20	Отделением УФМС России по г. Дзержинск	25
<input type="checkbox"/>	25	4 583 317 787	2014-07-20	Отделом УФМС России по г. Кемерово	26
<input type="checkbox"/>	26	4 361 799 851	2002-05-20	Отделом внутренних дел России по г. Комсомольск-на-Амуре	27
<input type="checkbox"/>	27	4 522 601 676	2016-09-20	Отделом УФМС России по г. Элиста	28
<input type="checkbox"/>	28	4 331 148 387	2020-11-20	Отделением УФМС России по г. Липецк	29
<input type="checkbox"/>	29	4 141 226 906	2009-07-20	ОВД России по г. Нижнекамск	30
<input type="checkbox"/>	30	4 079 456 544	2014-10-20	Отделением УФМС России по г. Копейск	31
<input type="checkbox"/>	31	4 364 666 805	2012-10-20	ОУФМС России по г. Серпухов	32
<input type="checkbox"/>	32	4 424 290 001	2006-03-20	Отделением УФМС России по г. Волжский	33

Должности:

## Выбрать: position

 **Выбрать**  Показать структуру  Изменить таблицу  Новая запись

Выбрать

Поиск

Сортировать

Лимит

50

Длина текста

100

Действие

Выбрать

**SELECT \* FROM `position` LIMIT 50**

(0.002 s)  Редактировать

<input type="checkbox"/>		ID	Name	Grade	Salary	Coeff
<input type="checkbox"/>		1	Генеральный директор	18	180000	4.5
<input type="checkbox"/>		3	Главный инженер	17	140400	3.51
<input type="checkbox"/>		4	Заместитель по кадрам и быту	17	140400	3.51
<input type="checkbox"/>		6	Заместитель по экономике	17	140400	3.51
<input type="checkbox"/>		7	Заместитель по снабжению	17	140400	3.51
<input type="checkbox"/>		8	Начальник отдела	11	89680	2.242
<input type="checkbox"/>		9	Работник отдела	5	50720	1.268
<input type="checkbox"/>		10	Младший работник отдела	1	40000	1
<input type="checkbox"/>		11	Охрана	5	50720	1.268

## Личная информация работника:

**Выбрать: private\_information**

**Выбрать**
 Показать структуру
 Изменить таблицу
 Новая запись

Выбрать
Поиск
Сортировать

Лимит

Длина текста

Действие
Выбрать

**SELECT \* FROM `private\_information` LIMIT 50** (0.002 s)  Редактировать

<input type="checkbox"/>	ID	Marital_Status	Count_Family	Adress
<input type="checkbox"/>	1	Холост	0	Россия, г. Рубцовск, Гагарина ул., д. 16 кв.103
<input type="checkbox"/>	3	В браке	3	Россия, г. Каспийск, Солнечная ул., д. 17 кв.23
<input type="checkbox"/>	4	В браке	1	Россия, г. Долгопрудный, Зеленая ул., д. 3 кв.197
<input type="checkbox"/>	5	В браке	1	Россия, г. Нефтеюганск, Мичурина ул., д. 21 кв.189
<input type="checkbox"/>	6	В браке	3	Россия, г. Волгоград, Мичурина ул., д. 23 кв.162
<input type="checkbox"/>	7	В браке	1	Россия, г. Салават, Речной пер., д. 6 кв.194
<input type="checkbox"/>	8	В браке	1	Россия, г. Нефтекамск, Центральный пер., д. 23 кв.184
<input type="checkbox"/>	9	В браке	1	Россия, г. Старый Оскол, Новая ул., д. 11 кв.165
<input type="checkbox"/>	10	Не замужем	0	Россия, г. Псков, Ленинская ул., д. 20 кв.9
<input type="checkbox"/>	11	В браке	3	Россия, г. Копейск, Восточная ул., д. 25 кв.198
<input type="checkbox"/>	12	В браке	1	Россия, г. Петрозаводск, Садовая ул., д. 18 кв.82
<input type="checkbox"/>	13	Холост	0	Россия, г. Златоуст, Новоселов ул., д. 12 кв.94
<input type="checkbox"/>	14	В браке	3	Россия, г. Нефтекамск, Колхозная ул., д. 6 кв.63
<input type="checkbox"/>	15	В браке	1	Россия, г. Нижневартовск, Радужная ул., д. 8 кв.57
<input type="checkbox"/>	16	В браке	1	Россия, г. Нальчик, Мирная ул., д. 20 кв.30
<input type="checkbox"/>	17	В браке	2	Россия, г. Воронеж, Речная ул., д. 6 кв.104
<input type="checkbox"/>	18	В браке	1	Россия, г. Новокузнецк, 17 Сентября ул., д. 15 кв.137
<input type="checkbox"/>	19	Холост	0	Россия, г. Евпатория, Калинина ул., д. 15 кв.93
<input type="checkbox"/>	20	В браке	2	Россия, г. Якутск, Колхозный пер., д. 2 кв.210
<input type="checkbox"/>	21	В браке	1	Россия, г. Раменское, Первомайская ул., д. 23 кв.19
<input type="checkbox"/>	22	В браке	2	Россия, г. Псков, Дружбы ул., д. 2 кв.199
<input type="checkbox"/>	23	Не замужем	0	Россия, г. Салават, Дачная ул., д. 8 кв.138
<input type="checkbox"/>	24	В браке	1	Россия, г. Ногинск, Цветочная ул., д. 6 кв.174
<input type="checkbox"/>	25	В браке	1	Россия, г. Кострома, Парковая ул., д. 4 кв.6
<input type="checkbox"/>	26	Холост	0	Россия, г. Химки, Мирная ул., д. 23 кв.110
<input type="checkbox"/>	27	Не замужем	0	Россия, г. Коломна, Садовая ул., д. 7 кв.86
<input type="checkbox"/>	28	В браке	2	Россия, г. Уссурийск, Западная ул., д. 13 кв.203
<input type="checkbox"/>	29	Холост	0	Россия, г. Северск, Лесная ул., д. 23 кв.174
<input type="checkbox"/>	30	В браке	3	Россия, г. Ставрополь, Шоссейная ул., д. 14 кв.3
<input type="checkbox"/>	31	В браке	3	Россия, г. Арзамас, Партизанская ул., д. 24 кв.87
<input type="checkbox"/>	32	В браке	1	Россия, г. Евпатория, Полесская ул., д. 15 кв.162
<input type="checkbox"/>	33	В браке	3	Россия, г. Балаково, Трудовая ул., д. 1 кв.39
<input type="checkbox"/>	34	NULL	NULL	NULL
<input type="checkbox"/>	35	NULL	NULL	NULL
<input type="checkbox"/>	36	NULL	NULL	NULL

Кросс-таблица (проект – работник):

### Выбрать: project\_employee

Выбрать

Поиск

Сортировать

Лимит  
50

Действие  
Выбрать

Выбрать

Показать структуру

Изменить таблицу

Новая запись

SELECT \* FROM `project\_employee` LIMIT 50

(0.002 s)

Редактировать

<input type="checkbox"/>		ID	Project	Employee	Salary
<input type="checkbox"/>		1	1	29	20000
<input type="checkbox"/>		2	2	29	20000
<input type="checkbox"/>		3	3	29	20000
<input type="checkbox"/>		4	4	29	20000
<input type="checkbox"/>		5	5	29	20000
<input type="checkbox"/>		6	7	29	20000
<input type="checkbox"/>		7	1	14	20000
<input type="checkbox"/>		8	2	14	20000
<input type="checkbox"/>		9	3	14	20000
<input type="checkbox"/>		10	4	14	20000
<input type="checkbox"/>		11	5	14	20000
<input type="checkbox"/>		12	7	14	20000
<input type="checkbox"/>		14	7	34	20000

Проекты:

### Выбрать: projects

Выбрать

Поиск

Сортировать

Лимит  
50

Длина текста  
100

Действие  
Выбрать

Выбрать

Показать структуру

Изменить таблицу

Новая запись

SELECT \* FROM `projects` LIMIT 50

(0.002 s)

Редактировать


<input type="checkbox"/>		project_key	project_name
<input type="checkbox"/>		1	Канбан-доска
<input type="checkbox"/>		2	Компьютерная игра, пишущая сама себя
<input type="checkbox"/>		3	База данных для "Ягодки"
<input type="checkbox"/>		4	База данных ("Арабская ночь")
<input type="checkbox"/>		5	Мобильное приложение ("Золотая блесна")
<input type="checkbox"/>		7	База данных (IT-проекты)
<input type="checkbox"/>		8	Test




Штатное расписание:

## Выбрать: staffing\_table

 **Выбрать**

 Показать структуру

 Изменить таблицу

 Новая запись

Выбрать

Поиск

Сортировать


Лимит































50

Действие

Выбрать

**SELECT \* FROM `staffing\_table` LIMIT 50**

(0.002 s)  Редактировать

<input type="checkbox"/>		ID	Position	Count	Employed	Dept
<input type="checkbox"/>		1	1	1	1	11
<input type="checkbox"/>		2	3	1	1	11
<input type="checkbox"/>		3	4	1	1	11
<input type="checkbox"/>		4	6	1	1	11
<input type="checkbox"/>		5	7	1	1	11
<input type="checkbox"/>		6	10	3	1	1
<input type="checkbox"/>		7	9	3	1	1
<input type="checkbox"/>		8	8	1	1	1
<input type="checkbox"/>		9	10	3	1	3
<input type="checkbox"/>		10	9	3	1	3
<input type="checkbox"/>		11	8	1	1	3
<input type="checkbox"/>		12	10	3	1	5
<input type="checkbox"/>		13	9	3	1	5
<input type="checkbox"/>		14	8	1	1	5
<input type="checkbox"/>		15	10	3	1	6
<input type="checkbox"/>		16	9	3	1	6
<input type="checkbox"/>		17	8	1	1	6
<input type="checkbox"/>		18	10	3	1	7
<input type="checkbox"/>		19	9	3	1	7
<input type="checkbox"/>		20	8	1	1	7
<input type="checkbox"/>		21	10	3	1	8
<input type="checkbox"/>		22	9	3	1	8
<input type="checkbox"/>		23	8	1	1	8
<input type="checkbox"/>		24	10	3	1	9
<input type="checkbox"/>		25	9	3	1	9
<input type="checkbox"/>		26	8	1	1	9
<input type="checkbox"/>		27	10	3	2	10
<input type="checkbox"/>		28	9	3	3	10
<input type="checkbox"/>		29	8	1	1	10
<input type="checkbox"/>		30	11	3	3	4

Создадим также необходимых пользователей нашей БД – бухгалтера. А в роли администратора БД выступает кадровый работник

Бухгалтер будет иметь доступ только к оператору SELECT

Сервер	<input type="text" value="localhost"/>	
Имя пользователя	<input type="text" value="bookkeeper"/>	
Пароль	<input type="password" value="...."/>	

Полномочия ?		<input type="text" value="`cursovik`. *"/>
Таблица	Select	<input checked="" type="checkbox"/>
Таблица	Show view	<input type="checkbox"/>
Таблица	Trigger	<input checked="" type="checkbox"/>
Таблица	Update	<input type="checkbox"/>
поле	Select	<input checked="" type="checkbox"/>
поле	Insert	<input type="checkbox"/>
поле	Update	<input type="checkbox"/>
поле	References	<input type="checkbox"/>
Процедура	Alter routine	<input type="checkbox"/>
Процедура	Execute	<input checked="" type="checkbox"/>

Пароль: 1234.

## Формирование SQL-запросов

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.

## Простейшие запросы (1 таблица):












Все должности:

```
SELECT * FROM Position
```

ID	Name	Grade	Salary	Coeff
1 	Генеральный директор	18	180000	4.5
3 	Главный инженер	17	140400	3.51
4 	Заместитель по кадрам и быту	17	140400	3.51
6 	Заместитель по экономике	17	140400	3.51
7 	Заместитель по снабжению	17	140400	3.51
8 	Начальник отдела	11	89680	2.242
9 	Работник отдела	5	50720	1.268
10 	Младший работник отдела	1	40000	1
11 	Охрана	5	50720	1.268








Все командировки:

```
SELECT * FROM business_trips
```

ID	Employee	City	Target	Start_Date	End_Date	Prepaid_Expense
1 	17	Саратов	Сертификация Оборудования	2011-02-21	2011-03-04	40000
2 	17	Самара	Сертификация Оборудования	2011-04-05	2011-04-29	40000
3 	17	Ростов	Сертификация Оборудования	2012-06-13	2012-06-27	40000
4 	17	Краснодар	Сертификация Оборудования	2013-09-01	2013-09-15	40000
5 	17	Адлер	Сертификация Оборудования	2013-12-15	2014-01-02	40000
6 	17	Санкт-Петербург	Сертификация Оборудования	2014-02-02	2014-02-28	40000
7 	17	Магадан	Сертификация Оборудования	2014-06-13	2014-06-30	40000
8 	17	Мурманск	Сертификация Оборудования	2014-08-03	2014-08-11	40000
9 	17	Севастополь	Сертификация Оборудования	2014-11-12	2014-12-01	40000
10 	17	Владивосток	Сертификация Оборудования	2015-01-29	2014-02-28	40000
11 	20	Майями	Гуманитарная помощь	2012-06-13	2013-09-15	500000









Все проекты:

```
SELECT * FROM projects
```

project_key	project_name
1 	Канбан-доска
2 	Компьютерная игра, пишущая сама себя
3 	База данных для "Ягодки"
4 	База данных ("Арабская ночь")
5 	Мобильное приложение ("Золотая блесна")
7 	База данных (IT-проекты)
8 	Test

Все отделы:


```
SELECT * FROM department
```

ID	Name
1 	Кадровый отдел
3 	Бухгалтерский отдел
4 	Бюро пропусков
5 	Отдел развития перспективных технологий
6 	Технический отдел
7 	Отдел снабжения
8 	Административно-хозяйственный отдел
9 	Юридический отдел
10 	Отдел аналитики
11 	Отдел главного управления

Запросы, затрагивающие 2 и больше таблицы:

Вывод истории перемещений работника:

```
SELECT ID_code, Last_Name, First_Name, Order_date, empl.Staffing_Table  
FROM Employee empl  
INNER JOIN employee_transfers trans  
ON empl.ID_code = trans.Employee
```

ID_code	Last_Name	First_Name	Order_date	Staffing_Table
13 	Мещеряков	Николай	2010-03-25	12
13 	Мещеряков	Николай	2011-03-25	12
13 	Мещеряков	Николай	2012-03-25	12
13 	Мещеряков	Николай	2013-03-25	12
13 	Мещеряков	Николай	2014-03-25	12
13 	Мещеряков	Николай	2015-03-25	12
13 	Мещеряков	Николай	2016-03-25	12
13 	Мещеряков	Николай	2017-03-25	12
13 	Мещеряков	Николай	2018-03-25	12

## Адрес проживания работника:


```
SELECT ID_code, Last_Name, First_Name, Address
FROM employee empl
INNER JOIN private information inf
ON inf.ID = empl.ID_code
```

ID_code	Last_Name	First_Name	Address
1 🍌	Филиппов	Яков	Россия, г. Рубцовск, Гагарина ул., д. 16 кв.103
3 🍌	Петров	Петр	Россия, г. Каспийск, Солнечная ул., д. 17 кв.23
4 🍌	Бурков	Алексей	Россия, г. Долгопрудный, Зеленая ул., д. 3 кв.197
5 🍌	Жаднов	Максим	Россия, г. Нефтеюганск, Мичурина ул., д. 21 кв.189
6 🍌	Петрушкенкевич	Павел	Россия, г. Волгоград, Мичурина ул., д. 23 кв.162
7 🍌	Вавкин	Никита	Россия, г. Салават, Речной пер., д. 6 кв.194
8 🍌	Баранов	Ашан	Россия, г. Нефтекамск, Центральный пер., д. 23 кв.184
9 🍌	Яковлев	Виктор	Россия, г. Старый Оскол, Новая ул., д. 11 кв.165
10 🍌	Бутусова	Антонина	Россия, г. Псков, Ленинская ул., д. 20 кв.9
11 🍌	Сиянина	Пелагея	Россия, г. Копейск, Восточная ул., д. 25 кв.198
12 🍌	Чужинов	Геннадий	Россия, г. Петрозаводск, Садовая ул., д. 18 кв.82
13 🍌	Мещеряков	Николай	Россия, г. Златоуст, Новоселов ул., д. 12 кв.94
14 🍌	Жиренкова	Настасья	Россия, г. Нефтекамск, Колхозная ул., д. 6 кв.63
15 🍌	Пыхтин	Антон	Россия, г. Нижневартовск, Радужная ул., д. 8 кв.57
16 🍌	Меншиков	Павел	Россия, г. Нальчик, Мирная ул., д. 20 кв.30
17 🍌	Балдагуев	Егор	Россия, г. Воронеж, Речная ул., д. 6 кв.104
18 🍌	Веденина	Таисия	Россия, г. Новокузнецк, 17 Сентября ул., д. 15 кв.137
19 🍌	Харьков	Федор	Россия, г. Евпатория, Калинина ул., д. 15 кв.93
20 🍌	Ядренникова	Юлия	Россия, г. Якутск, Колхозный пер., д. 2 кв.210
21 🍌	Балаев	Василий	Россия, г. Раменское, Первомайская ул., д. 23 кв.19
22 🍌	Урбановская	Лидия	Россия, г. Псков, Дружбы ул., д. 2 кв.199
23 🍌	Батрутдинова	Анна	Россия, г. Салават, Дачная ул., д. 8 кв.138
24 🍌	Ёлкин	Афанасий	Россия, г. Ногинск, Цветочная ул., д. 6 кв.174
25 🍌	Машукова	Галина	Россия, г. Кострома, Парковая ул., д. 4 кв.6
26 🍌	Петухов	Тимофей	Россия, г. Химки, Мирная ул., д. 23 кв.110
27 🍌	Чудова	Лариса	Россия, г. Коломна, Садовая ул., д. 7 кв.86
28 🍌	Травин	Константин	Россия, г. Уссурийск, Западная ул., д. 13 кв.203
29 🍌	Яфраков	Арсений	Россия, г. Северск, Лесная ул., д. 23 кв.174
30 🍌	Юхтриц	Арсений	Россия, г. Ставрополь, Шоссейная ул., д. 14 кв.3
31 🍌	Лобан	Леонтий	Россия, г. Арзамас, Партизанская ул., д. 24 кв.87
32 🍌	Тимирязев	Серафим	Россия, г. Евпатория, Полесская ул., д. 15 кв.162
33 🍌	Пахомов	Филипп	Россия, г. Балаково, Трудовая ул., д. 1 кв.39
34 🍌	Иванов	Иван	NULL
35 🍌	Порошин	Иван	NULL
36 🍌	Мельников	Павел	NULL

## Запросы, затрагивающие 3 и более таблицы:









Работники, не отчитавшиеся за командировку:

```
SELECT ID code, Last Name, First Name
FROM Employee empl, Business trips trips
WHERE (empl.ID Code = trips.Employee)
AND (trips.ID NOT IN (SELECT rpt.Business_Trips FROM business_trips_report rpt))
AND (trips.End_Date < CURRENT_DATE())
```

ID_code	Last_Name	First_Name
20 	Ядренникова	Юлия

## Работники по проектам:

```
SELECT ID code, Last Name, First Name, project name
FROM employee as empl, project employee as prj empl, projects as proj
WHERE (empl.ID_code = prj_empl.Employee) AND (prj_empl.Project = proj.project_key)
```

ID_code	Last_Name	First_Name	project_name
29 	Яфраков	Арсений	Канбан-доска
29 	Яфраков	Арсений	Компьютерная игра, пишущая сама себя
29 	Яфраков	Арсений	База данных для "Ягодки"
29 	Яфраков	Арсений	База данных ("Арабская ночь")
29 	Яфраков	Арсений	Мобильное приложение ("Золотая блесна")
29 	Яфраков	Арсений	База данных (IT-проекты)
14 	Жиренкова	Настасья	Канбан-доска
14 	Жиренкова	Настасья	Компьютерная игра, пишущая сама себя
14 	Жиренкова	Настасья	База данных для "Ягодки"
14 	Жиренкова	Настасья	База данных ("Арабская ночь")
14 	Жиренкова	Настасья	Мобильное приложение ("Золотая блесна")
14 	Жиренкова	Настасья	База данных (IT-проекты)
34 	Иванов	Иван	База данных (IT-проекты)

## Средняя сумма командировочных отдела:

```
SELECT dept.Name, AVG(Prepaid Expense)
FROM Department dept, Staffing_Table staff, Employee empl, Business_Trips trips
WHERE (dept.ID = staff.Dept)
AND (staff.ID = empl.Staffing_Table)
AND (empl.ID code = trips.Employee)
GROUP BY dept.ID
```

Name	AVG(Prepaid_Expense)
Технический отдел	40000
Отдел снабжения	500000

Работники по отделам:

```
SELECT dept.Name, empl.Last Name, empl.First Name
FROM Department dept, Staffing Table staff, Employee empl
WHERE (dept.ID = staff.Dept) AND (empl.Staffing_Table = staff.ID)
```

Name	Last_Name	First_Name
Кадровый отдел	Вавкин	Никита
Кадровый отдел	Баранов	Ашан
Кадровый отдел	Яковлев	Виктор
Бухгалтерский отдел	Бутусова	Антонина
Бухгалтерский отдел	Сиянина	Пелагея
Бухгалтерский отдел	Чужинов	Геннадий
Бюро пропусков	Лобан	Леонтий
Бюро пропусков	Тимирязев	Серафим
Бюро пропусков	Пахомов	Филипп
Отдел развития перспективных технологий	Мещеряков	Николай
Отдел развития перспективных технологий	Жиренкова	Настасья
Отдел развития перспективных технологий	Пыхтин	Антон
Технический отдел	Меншиков	Павел
Технический отдел	Балдагуев	Егор
Технический отдел	Веденина	Таисия
Отдел снабжения	Харьков	Федор
Отдел снабжения	Ядренникова	Юлия
Отдел снабжения	Балаев	Василий
Административно-хозяйственный отдел	Урбановская	Лидия
Административно-хозяйственный отдел	Батрутдинова	Анна
Административно-хозяйственный отдел	Ёлкин	Афанасий
Юридический отдел	Машукова	Галина
Юридический отдел	Петухов	Тимофей
Юридический отдел	Чудова	Лариса
Отдел аналитики	Травин	Константин
Отдел аналитики	Мельников	Павел
Отдел аналитики	Яфраков	Арсений
Отдел аналитики	Иванов	Иван
Отдел аналитики	Порошин	Иван
Отдел аналитики	Юхтриц	Арсений
Отдел главного управления	Филиппов	Яков
Отдел главного управления	Петров	Петр
Отдел главного управления	Бурков	Алексей
Отдел главного управления	Жаднов	Максим
Отдел главного управления	Петрушкенкевич	Павел



## Средний оклад по отделу:

```
SELECT dept.Name, AVG(Salary * Employed)
FROM department dept, position pos, staffing table staff
WHERE (dept.ID = staff.Dept) AND (staff.Position = pos.ID)
GROUP BY dept.ID
```

Name	AVG(Salary * Employed)
Кадровый отдел	60133.333333333336
Бухгалтерский отдел	60133.333333333336
Бюро пропусков	152160
Отдел развития перспективных технологий	60133.333333333336
Технический отдел	60133.333333333336
Отдел снабжения	60133.333333333336
Административно-хозяйственный отдел	60133.333333333336
Юридический отдел	60133.333333333336
Отдел аналитики	107280
Отдел главного управления	148320

## Работники по должностям:

```
SELECT pos.Name, empl.Last Name, empl.First Name
FROM Position pos, Staffing Table staff, Employee empl
WHERE (pos.ID = staff.Position) AND (empl.Staffing_Table = staff.ID)
```

Name	Last_Name	First_Name
Генеральный директор	Филиппов	Яков
Главный инженер	Петров	Петр
Заместитель по кадрам и быту	Бурков	Алексей
Заместитель по экономике	Жаднов	Максим
Заместитель по снабжению	Петрушкенкевич	Павел
Начальник отдела	Яковлев	Виктор
Начальник отдела	Чужинов	Геннадий
Начальник отдела	Пыхтин	Антон
Начальник отдела	Веденина	Таисия
Начальник отдела	Балаев	Василий
Начальник отдела	Ёлкин	Афанасий
Начальник отдела	Чудова	Лариса
Начальник отдела	Юхтриц	Арсений
Работник отдела	Баранов	Ашан
Работник отдела	Сиянина	Пелагея
Работник отдела	Жиренкова	Настасья
Работник отдела	Балдагуев	Егор
Работник отдела	Ядренникова	Юлия
Работник отдела	Батрутдинова	Анна
Работник отдела	Петухов	Тимофей
Работник отдела	Яфраков	Арсений
Работник отдела	Иванов	Иван
Работник отдела	Порошин	Иван
Младший работник отдела	Вавкин	Никита
Младший работник отдела	Бутусова	Антонина
Младший работник отдела	Мещеряков	Николай
Младший работник отдела	Меншиков	Павел
Младший работник отдела	Харьков	Федор
Младший работник отдела	Урбановская	Лидия
Младший работник отдела	Машукова	Галина
Младший работник отдела	Травин	Константин
Младший работник отдела	Мельников	Павел
Охрана	Лобан	Леонтий
Охрана	Тимирязев	Серафим
Охрана	Пахомов	Филипп



## Создание процедур

Процедура добавление пустого кортежа в таблицу с личной информацией работника:

```
SELECT ROUTINE_DEFINITION FROM information_schema.ROUTINES WHERE SPECIFIC_NAME='AddEmptyInf'
```

ROUTINE_DEFINITION
BEGIN INSERT INTO private_information (ID) values (id); END

Процедура обработки данных для отчета о средних командировочных по отделам:

```
SELECT ROUTINE_DEFINITION FROM information_schema.ROUTINES WHERE SPECIFIC_NAME='GetAVGBusByDep'
```

ROUTINE_DEFINITION
BEGIN SELECT dept.Name, AVG(Prepaid_Expense) FROM Department dept, Staffing_Table staff, Employee empl, Business_Trips trips WHERE (dept.ID = staff.Dept) AND (staff.ID = empl.Staffing_Table) AND (empl.ID_code = trips.Employee) GROUP BY dept.ID; END

call GetAVGBusByDep()

Name	AVG(Prepaid_Expense)
Технический отдел	40000
Отдел аналитики	10000000
Кадровый отдел	12000

Процедура обработки данных для отчета о средней зарплате по отделам:

```
SELECT ROUTINE_DEFINITION FROM information_schema.ROUTINES WHERE SPECIFIC_NAME='GetAVGSalByDep'
```

ROUTINE_DEFINITION
BEGIN SELECT dept.Name, AVG(Salary * Employed) FROM department dept, position pos, staffing_table staff WHERE (dept.ID = staff.Dept) AND (staff.Position = pos.ID) GROUP BY dept.ID; END

call GetAVGSalByDep()

Name	AVG(Salary * Employed)
Кадровый отдел	60133.333333333336
Бухгалтерский отдел	60133.333333333336
Бюро пропусков	152160
Отдел развития перспективных технологий	60133.333333333336
Технический отдел	60133.333333333336
Отдел снабжения	60133.333333333336
Административно-хозяйственный отдел	60133.333333333336
Юридический отдел	60133.333333333336
Отдел аналитики	107280
Отдел главного управления	148320

Процедура для получения работников на проекте по id проекта:

```
SELECT ROUTINE_DEFINITION FROM information_schema.ROUTINES WHERE SPECIFIC_NAME='GetEmplFromProjById'
```

```
ROUTINE_DEFINITION
BEGIN
SELECT ID_CODE, Last_name, Left(First_Name, 1), Left(Otchestvo, 1) FROM employee empl, project_employee empl_proj
WHERE (empl.ID_code = empl_proj.Employee) AND (empl_proj.Project = id_) ;
END
```

call GetEmplFromProjById(1)

ID_CODE	Last_name	Left(First_Name, 1)	Left(Otchestvo, 1)
29	Яфраков	А	И
14	Жиренкова	Н	Ф
7	Вавкин	Н	Д
18	Веденина	Т	П

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

```
SELECT ROUTINE_DEFINITION FROM information_schema.ROUTINES WHERE SPECIFIC_NAME='GetEmplInfByProjId'
```

```
ROUTINE_DEFINITION
BEGIN
SELECT ID_code, empl.Last_Name, empl.First_Name, empl.Otchestvo, pos.Name, dept.Name
FROM Position pos, Department dept, Staffing_Table staff, Employee empl, project_employee empl_proj
WHERE (dept.ID = staff.Dept) AND (empl.Staffing_Table = staff.ID) AND (staff.position = pos.ID)
AND (empl.ID_code = empl_proj.Employee) AND (empl_proj.Project = id_) AND
(empl.Staffing_Table = staff.ID);
END
```

call GetEmplInfByProjId(1)

ID_code	Last_Name	First_Name	Otchestvo	Name	Name
29	Яфраков	Арсений	Ильич	Работник отдела	Отдел аналитики
14	Жиренкова	Настасья	Федоровна	Работник отдела	Отдел развития перспективных технологий
7	Вавкин	Никит	Дмитриевич	Младший работник отдела	Кадровый отдел
18	Веденина	Таисия	Прокопьевна	Начальник отдела	Технический отдел

## Создание триггеров

Триггер, подсчитывающий заново стоимость по проекту при добавлении нового этапа:

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

Изменить триггер: employee\_add\_inf

Время	AFTER
Событие	INSERT
Тип	FOR EACH ROW

Название: employee\_add\_inf

```
CALL AddEmptyInf(NEW.ID_code)
```

До добавления работника:

<input type="checkbox"/>		ID	Marital_Status	Count_Family	Adress
<input type="checkbox"/>		34	NULL	NULL	NULL
<input type="checkbox"/>		35	NULL	NULL	NULL
<input type="checkbox"/>		36	NULL	NULL	NULL
<input type="checkbox"/>		42	Женат	2	Москва, улица Советская, дом 5, квартира 2
<input type="checkbox"/>		45	NULL	NULL	NULL

После добавления работника:

<input type="checkbox"/>		ID	Marital_Status	Count_Family	Adress
<input type="checkbox"/>		34	NULL	NULL	NULL
<input type="checkbox"/>		35	NULL	NULL	NULL
<input type="checkbox"/>		36	NULL	NULL	NULL
<input type="checkbox"/>		42	Женат	2	Москва, улица Советская, дом 5, квартира 2
<input type="checkbox"/>		45	NULL	NULL	NULL
<input type="checkbox"/>		46	NULL	NULL	NULL

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

Изменить триггер: employee\_bd\_inf

<b>Время</b>	BEFORE ▾
<b>Событие</b>	DELETE ▾
<b>Тип</b>	FOR EACH ROW ▾

Название:

`DELETE FROM private_information inf WHERE OLD.ID_code = inf.ID`

До удаления работника:

<input type="checkbox"/>		ID	Marital_Status	Count_Family	Adress
<input type="checkbox"/>		34	NULL	NULL	NULL
<input type="checkbox"/>		35	NULL	NULL	NULL
<input type="checkbox"/>		36	NULL	NULL	NULL
<input type="checkbox"/>		42	Женат	2	Москва, улица Советская, дом 5, квартира 2
<input type="checkbox"/>		45	NULL	NULL	NULL
<input type="checkbox"/>		46	NULL	NULL	NULL

После удаления работника:

<input type="checkbox"/>		ID	Marital_Status	Count_Family	Adress
<input type="checkbox"/>		34	NULL	NULL	NULL
<input type="checkbox"/>		35	NULL	NULL	NULL
<input type="checkbox"/>		36	NULL	NULL	NULL
<input type="checkbox"/>		42	Женат	2	Москва, улица Советская, дом 5, квартира 2
<input type="checkbox"/>		45	NULL	NULL	NULL

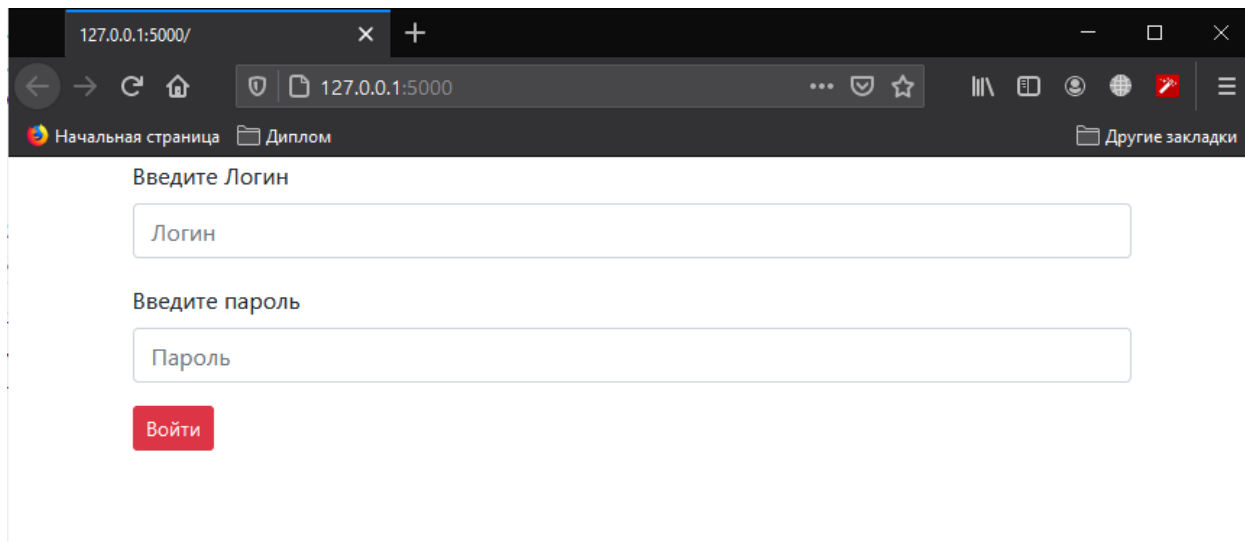
## **Разработка интерфейса**

Для разработки пользовательского интерфейса был использован фреймворк Флask (Flask). Flask — фреймворк для создания веб-приложений на языке программирования Python, использующий набор инструментов Werkzeug, а также шаблонизатор Jinja2. Он относится к категории так называемых микрофреймворков — минималистичных каркасов веб-приложений, сознательно предоставляющих лишь самые базовые возможности.

Для оформления графических элементов, которые будут приятны глазу пользователя, использовался фреймворк - Bootstrap. Bootstrap (также известен как Twitter Bootstrap) — свободный набор инструментов для создания сайтов и веб-приложений. Включает в себя HTML- и CSS-шаблоны оформления для типографики, веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса, включая JavaScript-расширения.

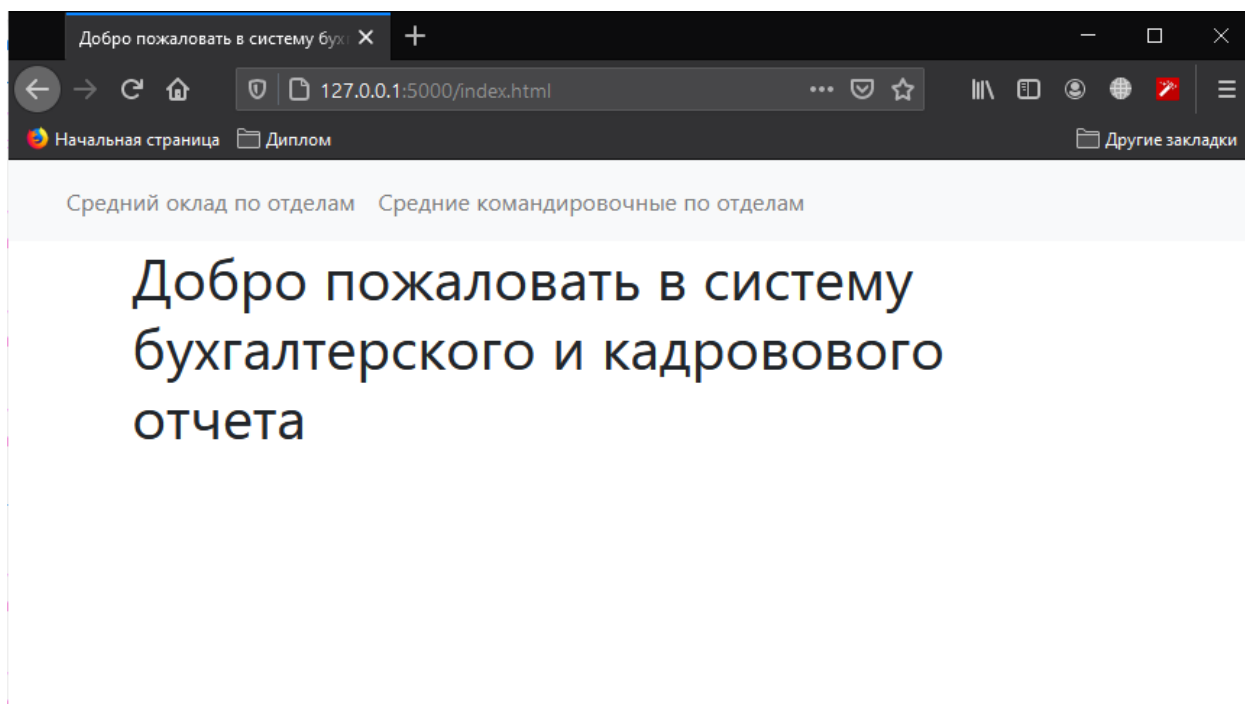
## Руководство пользователя системы

Обычный пользователь работает с интерфейсом. Интерфейс разработан в виде Web-приложения. Когда пользователь заходит на станицу, он видит страницу входа.



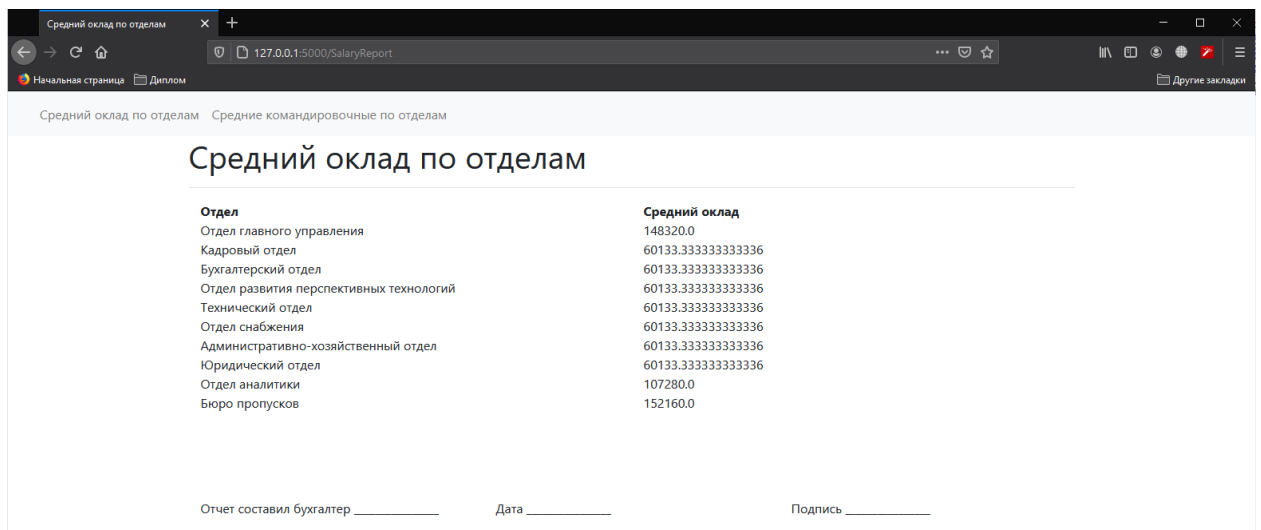
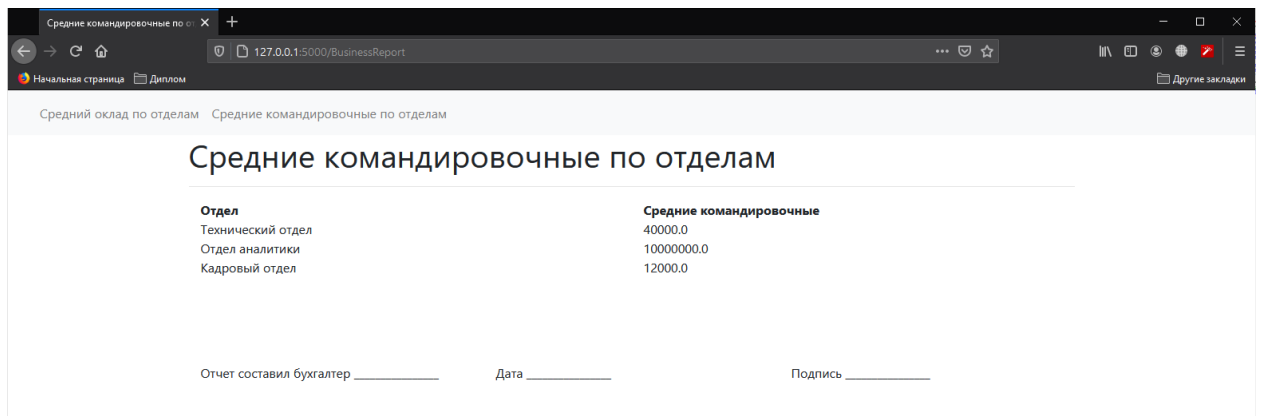
The screenshot shows a web browser window with the address bar displaying '127.0.0.1:5000/'. The page has a dark header with navigation links: 'Начальная страница' and 'Диплом'. The main content area is white and contains a login form. The form has two input fields: 'Введите Логин' (Login) and 'Введите пароль' (Password). Below the password field is a red button labeled 'Войти' (Login). The browser's address bar also shows '127.0.0.1:5000'.

Авторизовавшись как бухгалтер, нам будет предоставлен доступ к отчетам:

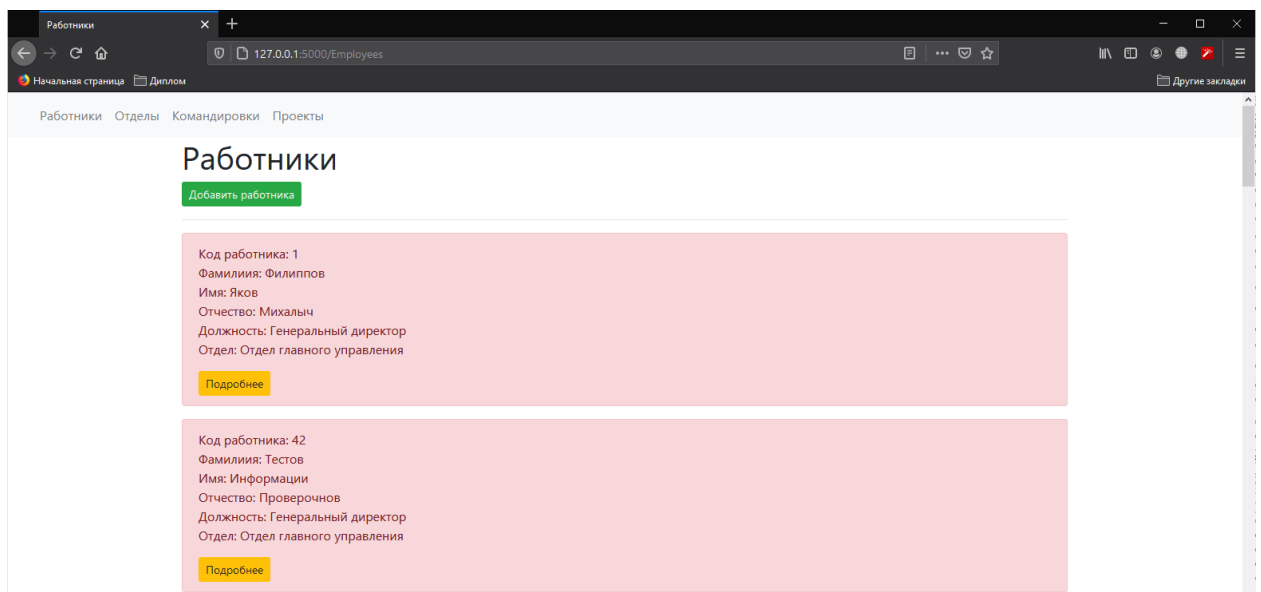


The screenshot shows the dashboard of a web application. The browser window title is 'Добро пожаловать в систему бух'. The address bar shows '127.0.0.1:5000/index.html'. The page has a dark header with navigation links: 'Начальная страница' and 'Диплом'. The main content area is white and contains a large heading: 'Добро пожаловать в систему бухгалтерского и кадрового отчета'. Below the heading are two links: 'Средний оклад по отделам' and 'Средние командировочные по отделам'.

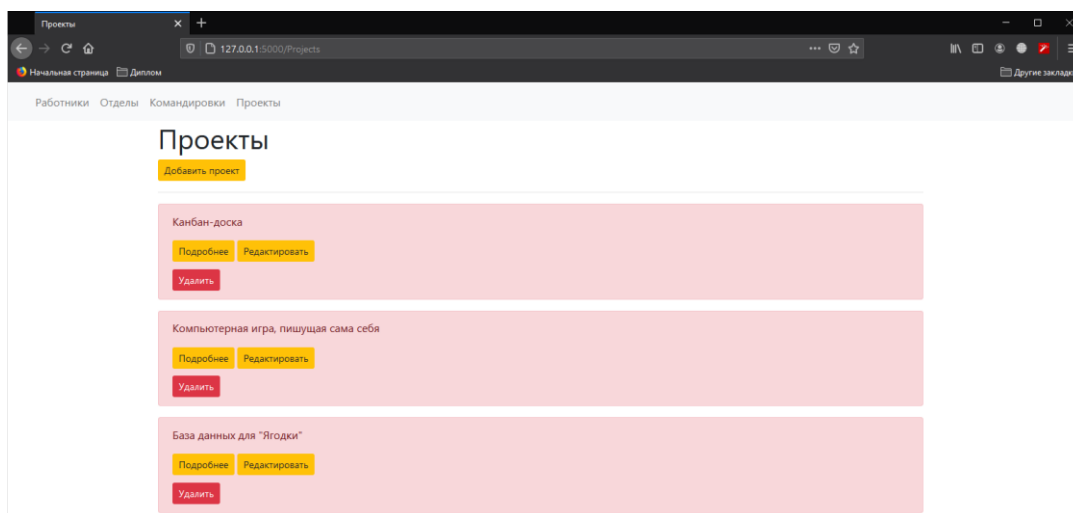
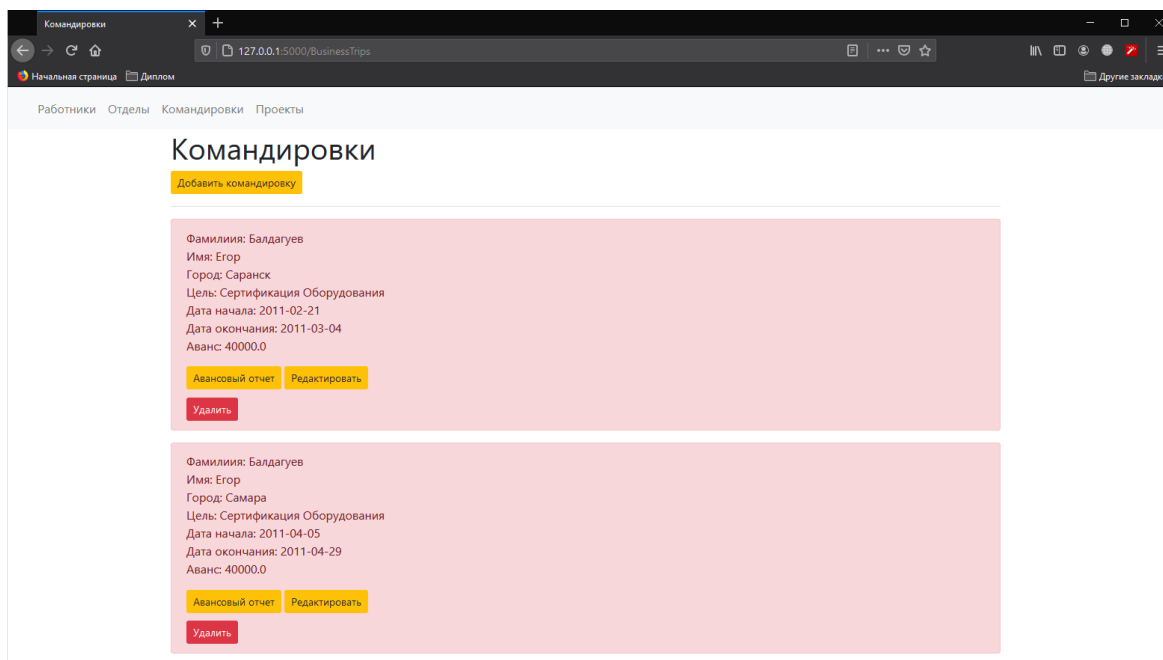
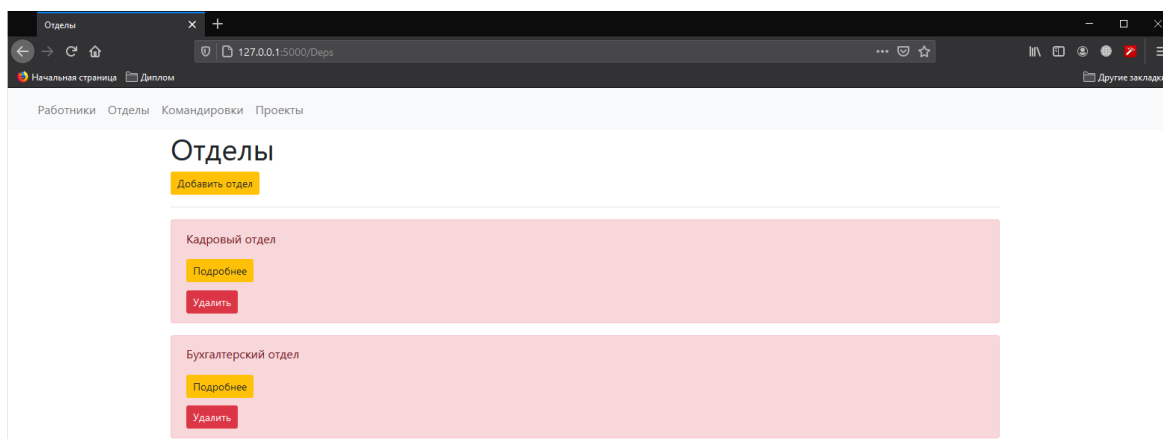
Щелкая по кнопкам («Средний оклад по отделам» и «Средние командировочные по отделам») бухгалтер получит соответствующие отчеты:



Авторизовавшись как работник отдела кадров, нам будет предоставлен полный доступ к таблицам:



Щелкая по кнопкам («Работники», «Отделы», «Командировки», «Проекты») пользователь получит соответствующие списки данных:





В разделе «работники» при нажатии на кнопку «Добавить работника» изменения внесутся в БД, и пользователь увидит обновленный список работников, при этом автоматически создается запись с личной информацией работника, так как сработает триггер. Нажав на кнопку «Подробнее», пользователь переходит на страницу с полной информацией о работнике:

127.0.0.1:5000/Employees/36

← → ↻ 🏠 🔒 127.0.0.1:5000/Employees/36 ⋮ 📄 ⭐ 🌐 🗑️ ☰

Начальная страница 📁 Диплом 📁 Другие закладки

Работники Отделы Командировки Проекты

Код работника: 36  
Фамилия: Мельников  
Имя: Павел  
Отчество: Степанович  
Должность: Младший работник отдела  
Отдел: Отдел аналитики

Удалить работника Редактировать

Командировки:

Город	Цель	Дата начала	Дата окончания
-------	------	-------------	----------------

Добавить отчет о перемещении

Перемещения в организации:

Номер приказа	Дата приказа	Прошлая должность	Прошлый отдел
---------------	--------------	-------------------	---------------

Участие в проектах:

Название	Оплата
----------	--------

Личная информация

Семейное положение: None  
Количество членов семьи: None  
Адрес: None

Редактировать личную информацию

Щелкая по кнопкам («Редактировать...»), пользователь может изменить нужную ему информацию:

The screenshot shows a web browser window with the address bar displaying '127.0.0.1:5000/Employees/EditEmployee36'. The page title is 'Изменение информации о работнике'. The form contains the following fields:

- Фамилия: Мельников
- Имя: Павел
- Отчество: Степанович
- Штатное расписание: Отдел аналитики - Младший работник отдела
- Штатное расписание: 19.09.1989
- Премия: Premium

A blue button labeled 'Сохранить изменения' is located at the bottom of the form.

The screenshot shows a web browser window with the address bar displaying '127.0.0.1:5000/Employees/EditPrivInf36'. The page title is 'Изменение личной информации'. The form contains the following fields:

- Семейное положение: None
- Количество членов семьи: Количество
- Адрес: None

A blue button labeled 'Сохранить изменения' is located at the bottom of the form.

Добавление нового паспорта

Код и номер паспорта

Номер

Дата выдачи

ДД . ММ . ГГГГ

Адрес

Кем выдан

Сохранить изменения

При нажатии на кнопку «Сохранить изменения» новые данные внесутся в БД, и пользователь увидит обновленные данные в информации о работнике.

Нажимая на красные кнопки («Удалить»), пользователь может удалить записи, которые требуется.

После удаления из БД, пользователь увидит обновленный список работников, при этом таблица с личной информацией изменится, так как опять же сработает триггер

Перейдя во вкладку «Отделы» на навигационной панели, пользователь увидит список отделов:

Отделы

Добавить отдел

Кадровый отдел

Подробнее

Удалить

Бухгалтерский отдел

Подробнее

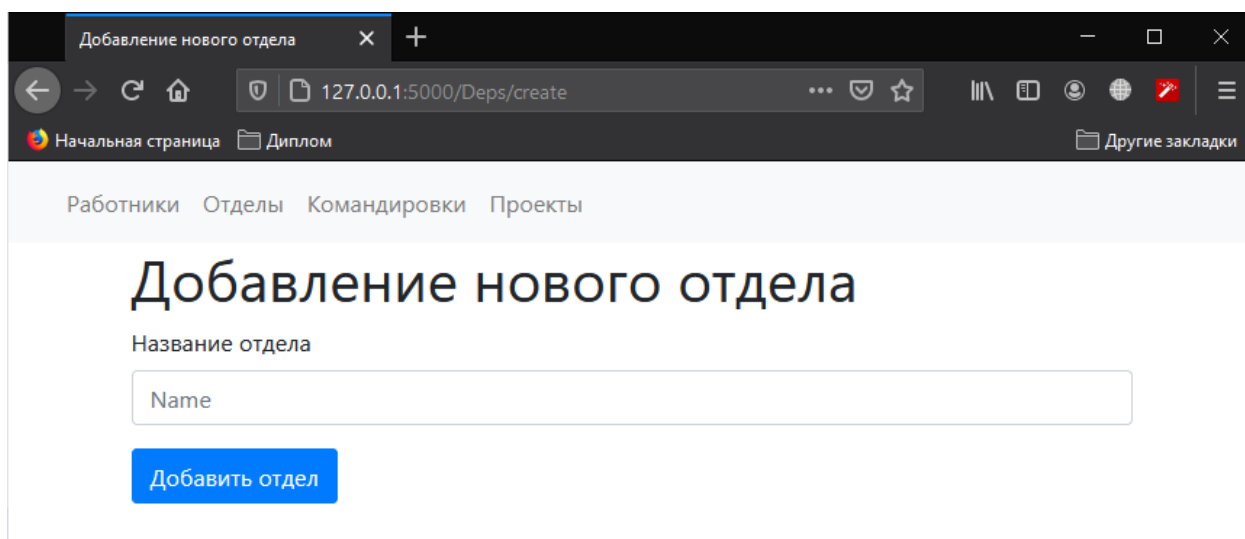
Удалить

Бюро пропусков

Подробнее

Удалить

Для добавления нового отдела нажмем «Добавить отдел»:



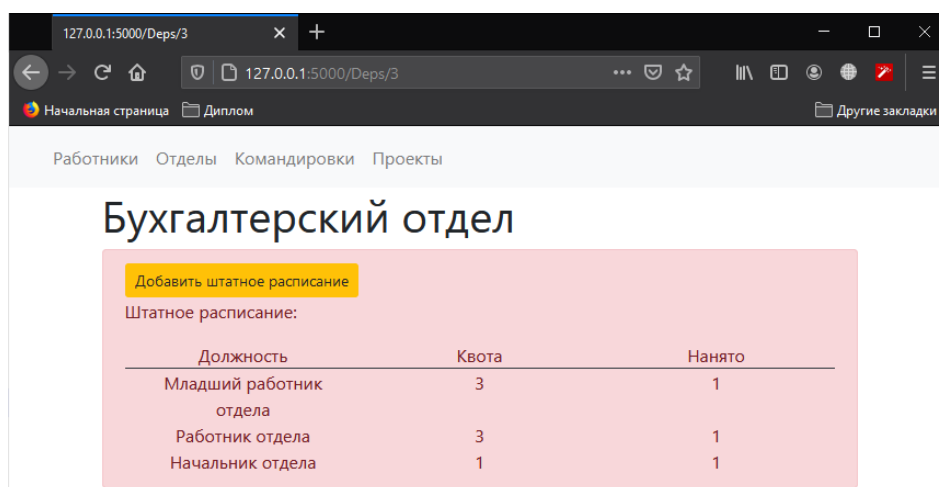
Добавление нового отдела

Название отдела

Name

Добавить отдел

Нажав «Подробнее», откроется страница информации об отделе, где указано штатное расписание отдела и где можно добавить новое штатное расписание для отдела, нажав «Добавить штатное расписание»:



Бухгалтерский отдел

Добавить штатное расписание

Штатное расписание:

Должность	Квота	Нанято
Младший работник отдела	3	1
Работник отдела	3	1
Начальник отдела	1	1

Добавление нового штатного расписания отдела

Должность  
Генеральный директор

Квота  
Количество

Добавить штатное расписание

После нажатия на кнопку «Удалить», выбранный отдел будет удален, если у него нет штатного расписания.

Для перехода в раздел командировок, пользователь должен нажать кнопку «Командировки»:

Командировки

Добавить командировку

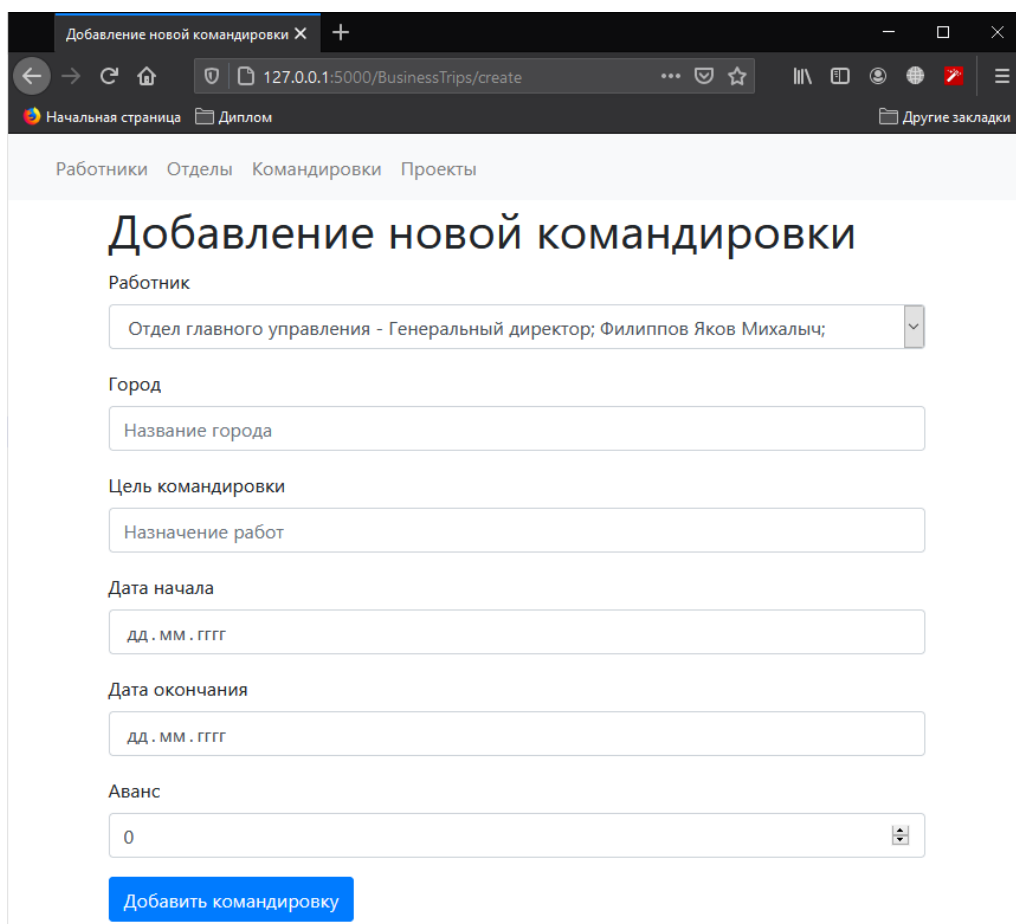
Фамилия: Балдагуев  
Имя: Егор  
Город: Саранск  
Цель: Сертификация Оборудования  
Дата начала: 2011-02-21  
Дата окончания: 2011-03-04  
Аванс: 40000.0

Авансовый отчет Редактировать Удалить

Фамилия: Балдагуев  
Имя: Егор  
Город: Самара  
Цель: Сертификация Оборудования  
Дата начала: 2011-04-05  
Дата окончания: 2011-04-29  
Аванс: 40000.0

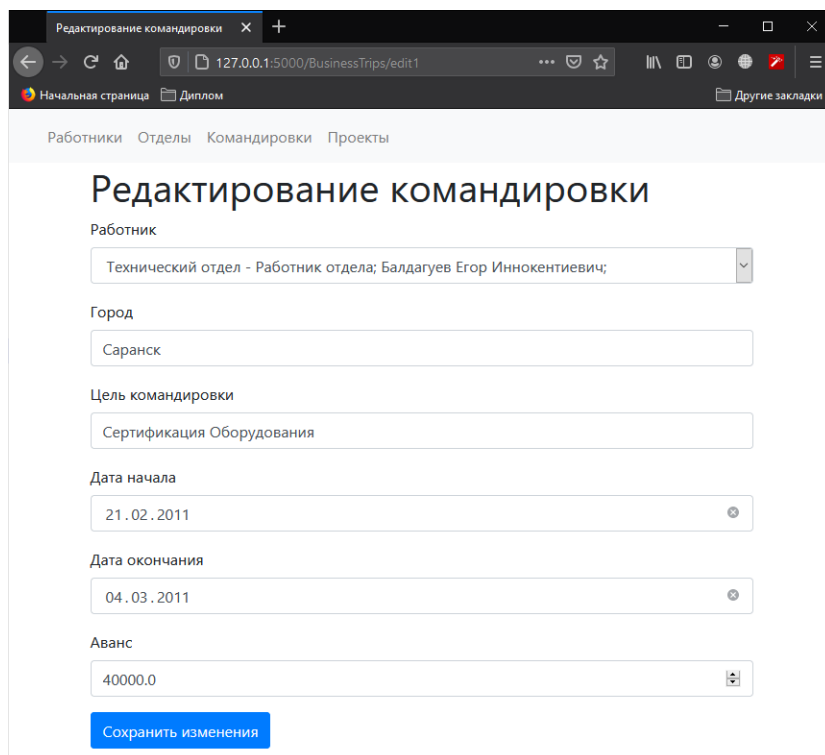
Авансовый отчет Редактировать Удалить

Для добавления новой командировки следует нажать «Добавить командировку»:



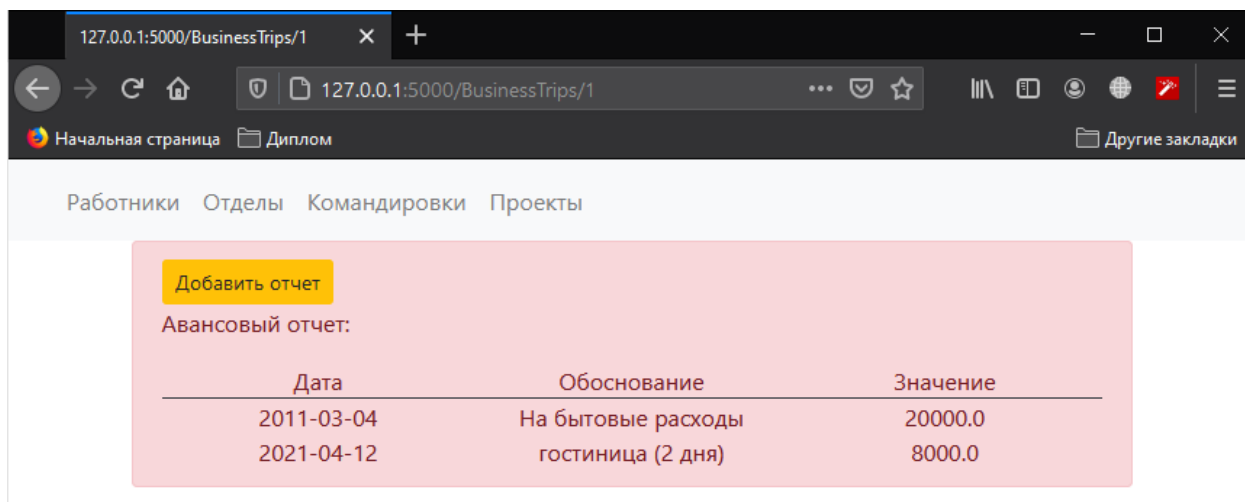
The screenshot shows a web browser window with the address bar displaying '127.0.0.1:5000/BusinessTrips/create'. The browser's address bar also shows 'Начальная страница' and 'Диплом'. The page has a navigation bar with links: 'Работники', 'Отделы', 'Командировки', and 'Проекты'. The main heading is 'Добавление новой командировки'. Below the heading, there is a form with the following fields: 'Работник' (a dropdown menu showing 'Отдел главного управления - Генеральный директор; Филиппов Яков Михалыч;'), 'Город' (a text input field with 'Название города'), 'Цель командировки' (a text input field with 'Назначение работ'), 'Дата начала' (a date input field with 'дд . мм . гггг'), 'Дата окончания' (a date input field with 'дд . мм . гггг'), and 'Аванс' (a text input field with '0'). At the bottom of the form is a blue button labeled 'Добавить командировку'.

Для редактирования командировки нужно нажать на кнопку «Редактировать»:



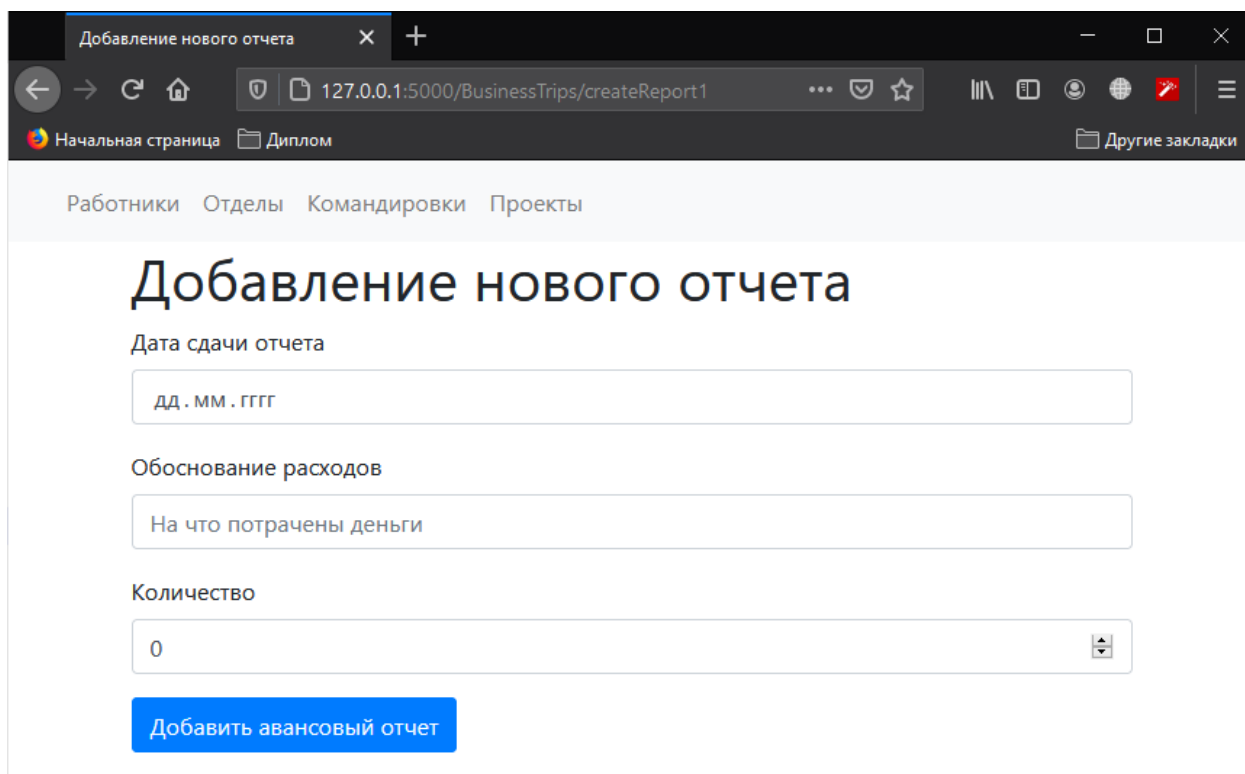
The screenshot shows a web browser window with the address bar displaying '127.0.0.1:5000/BusinessTrips/edit1'. The browser's address bar also shows 'Начальная страница' and 'Диплом'. The page has a navigation bar with links: 'Работники', 'Отделы', 'Командировки', and 'Проекты'. The main heading is 'Редактирование командировки'. Below the heading, there is a form with the following fields: 'Работник' (a dropdown menu showing 'Технический отдел - Работник отдела; Балдагуев Егор Иннокентиевич;'), 'Город' (a text input field with 'Саранск'), 'Цель командировки' (a text input field with 'Сертификация Оборудования'), 'Дата начала' (a date input field with '21.02.2011'), 'Дата окончания' (a date input field with '04.03.2011'), and 'Аванс' (a text input field with '40000.0'). At the bottom of the form is a blue button labeled 'Сохранить изменения'.

Для получения авансового отчета о командировке нужно нажать на кнопку «Авансовый отчет»:



Дата	Обоснование	Значение
2011-03-04	На бытовые расходы	20000.0
2021-04-12	гостиница (2 дня)	8000.0

На странице можно добавить новый отчет, через кнопку «Добавить отчет»:



Добавление нового отчета

Дата сдачи отчета

дд . мм . гggг

Обоснование расходов

На что потрачены деньги

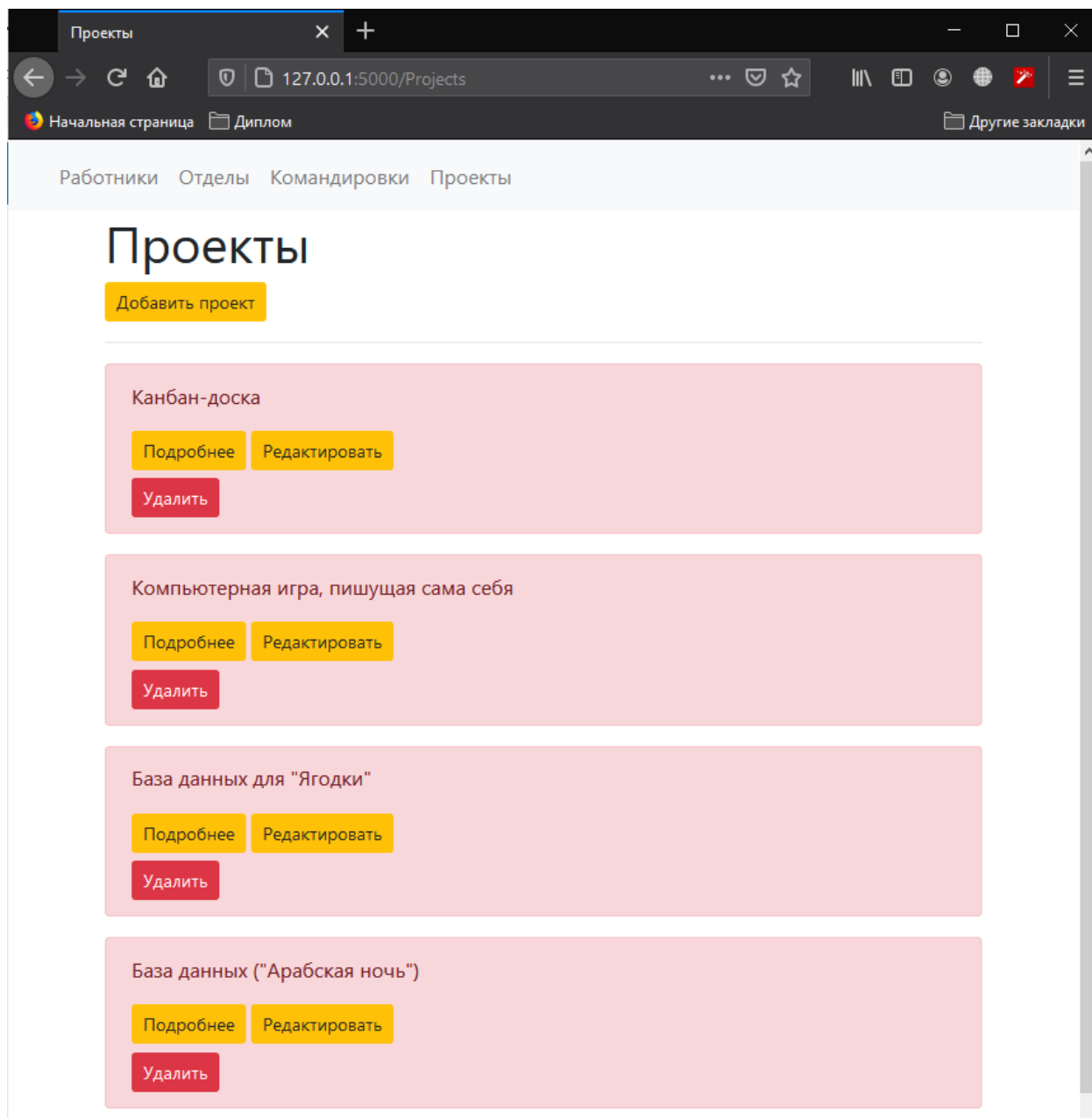
Количество

0

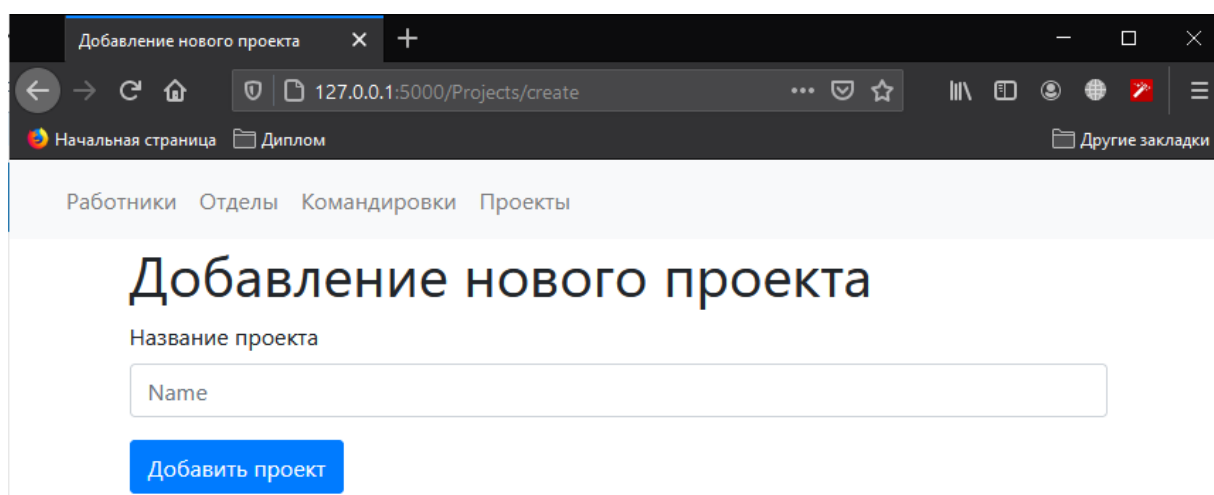
Добавить авансовый отчет

Для удаления командировки нужно нажать кнопку «Удалить». После удаления командировки, все авансовые отчеты каскадно удалятся.

Чтобы просмотреть список проектов, нужно нажать на кнопку «Проекты»:

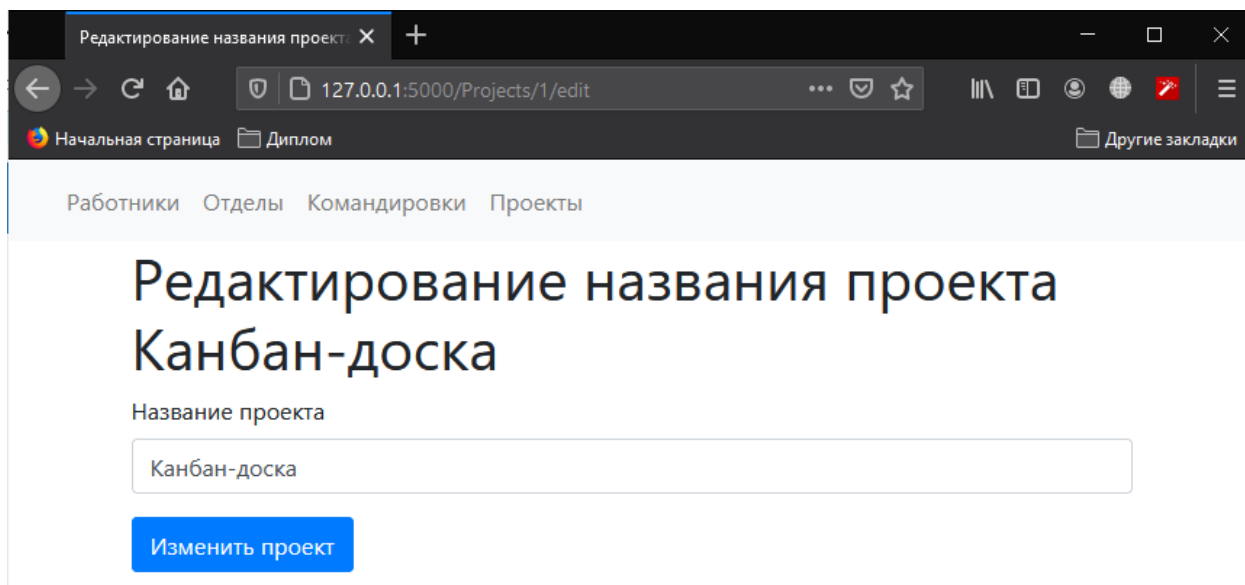


Проект можно добавить, нажав кнопку «Добавить проект»:





После нажатия на кнопку «Редактировать», пользователя переведут на страницу для редактирования проекта:



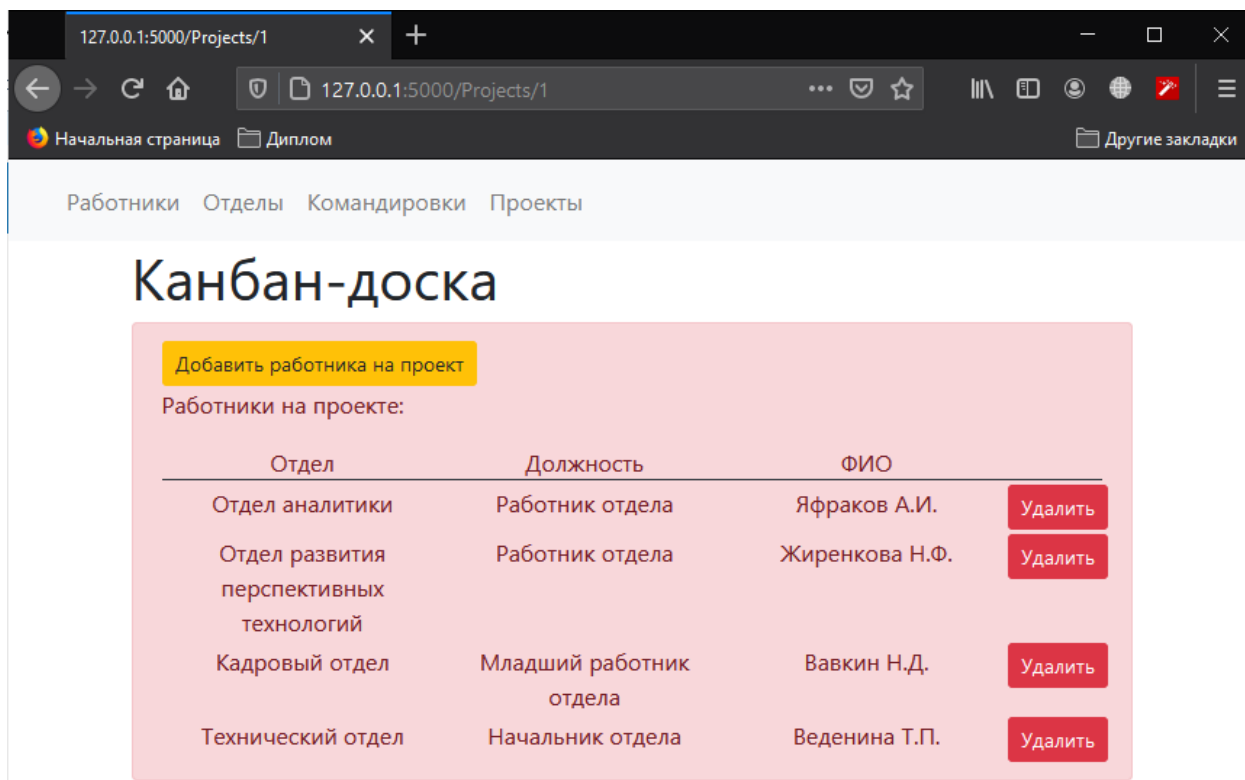
Редактирование названия проекта

## Канбан-доска

Название проекта

Изменить проект

Нажав «Подробнее», произойдет переход на страницу со списком работников на проекте:

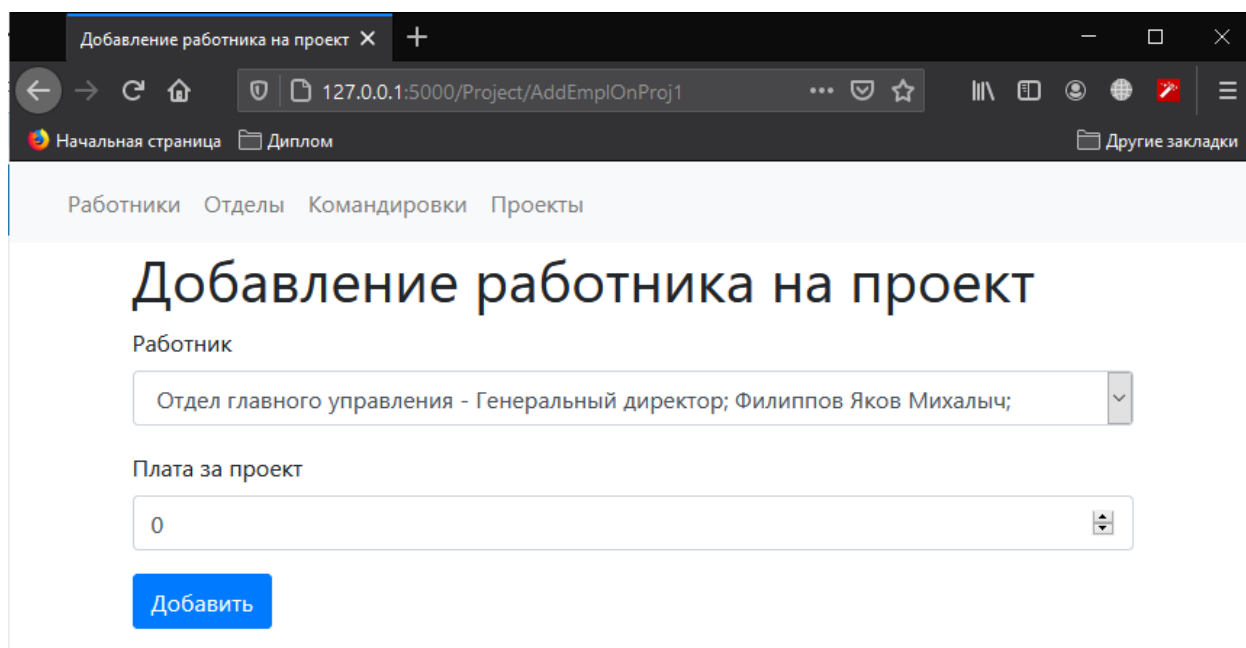


Добавить работника на проект

Работники на проекте:

Отдел	Должность	ФИО	
Отдел аналитики	Работник отдела	Яфраков А.И.	Удалить
Отдел развития перспективных технологий	Работник отдела	Жиренкова Н.Ф.	Удалить
Кадровый отдел	Младший работник отдела	Вавкин Н.Д.	Удалить
Технический отдел	Начальник отдела	Веденина Т.П.	Удалить

Где можно добавить работника на проект через кнопку «Добавить работника на проект»:

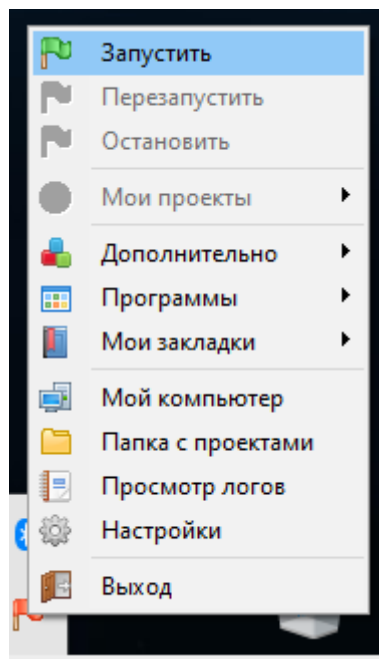


The screenshot shows a web browser window with the address bar displaying `127.0.0.1:5000/Project/AddEmplOnProj1`. The browser's tab is titled "Добавление работника на проект". The page has a navigation bar with links: "Работники", "Отделы", "Командировки", and "Проекты". The main heading is "Добавление работника на проект". Below the heading, there are two input fields: "Работник" with a dropdown menu showing "Отдел главного управления - Генеральный директор; Филиппов Яков Михалыч;" and "Плата за проект" with a text input containing "0". At the bottom, there is a blue button labeled "Добавить".

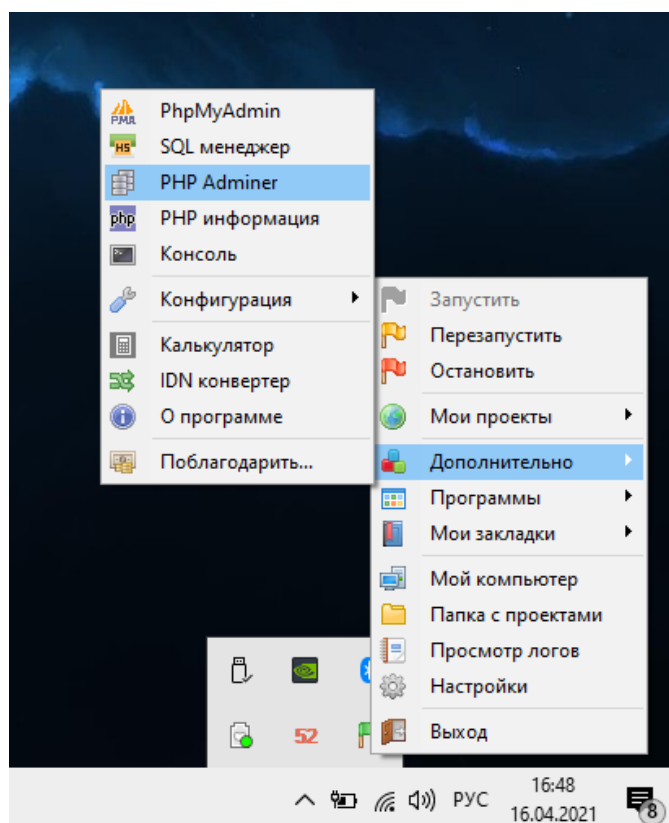
Или удалить с проекта, нажав на кнопку «Удалить».

## Руководство администратора системы

Администратор системы, не имеет интерфейса как такового. Для работы с базой данных администратору необходимо развернуть локальный сервер:



После чего запустить PHP Adminer (средство администрирования базы данных).



После этого необходимо войти в систему:

## Войти

Движок	MySQL
Сервер	localhost
Имя пользователя	
Пароль	
База данных	

☒ Остаться в системе

Далее администратор попадет на главный экран:

Язык: [Русский]

MySQL » Сервер » База данных: cursovik\_v2

Adminer 4.7.8 4.8.0

База данных: cursovik\_v2

DB: [cursovik\_v2]

Изменить базу данных | Схема базы данных | Полномочия

SQL-запрос  
Импорт  
Экспорт  
Создать таблицу

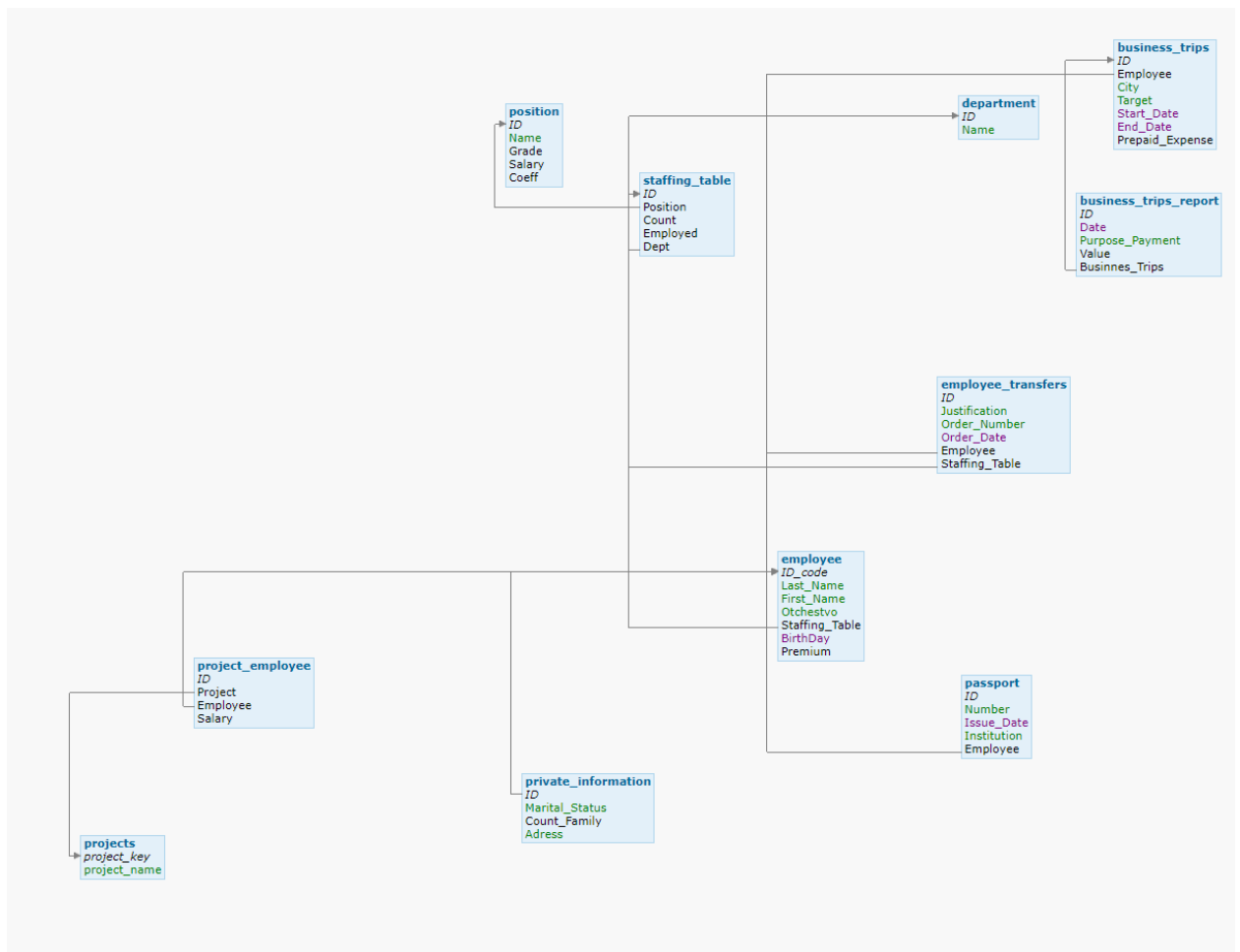
business\_trips  
business\_trips\_report  
department  
employee  
employee\_transfers  
passport  
position  
private\_information  
project\_employee  
projects  
staffing\_table

Таблицы и представления

Поиск в таблицах [11]

<input type="checkbox"/>	Таблица	Тип таблиц <sup>?</sup>	Режим сопоставления <sup>?</sup>	Объём данных <sup>?</sup>	Объём индексов <sup>?</sup>	Свободное место <sup>?</sup>	Автоматическое приращение <sup>?</sup>	Строк <sup>?</sup>	Комментарий <sup>?</sup>
<input type="checkbox"/>	business_trips	InnoDB	utf8mb4_0900_ai_ci	16 384	16 384	0	0	19	~ 12
<input type="checkbox"/>	business_trips_report	InnoDB	utf8mb4_0900_ai_ci	16 384	16 384	0	0	15	~ 11
<input type="checkbox"/>	department	InnoDB	utf8mb4_0900_ai_ci	16 384	0	0	0	13	~ 10
<input type="checkbox"/>	employee	InnoDB	utf8mb4_0900_ai_ci	16 384	16 384	0	0	47	~ 38
<input type="checkbox"/>	employee_transfers	InnoDB	utf8mb4_0900_ai_ci	16 384	32 768	0	0	16	~ 9
<input type="checkbox"/>	passport	InnoDB	utf8mb4_0900_ai_ci	16 384	16 384	0	0	39	~ 36
<input type="checkbox"/>	position	InnoDB	utf8mb4_0900_ai_ci	16 384	0	0	0	11	~ 9
<input type="checkbox"/>	private_information	InnoDB	utf8mb4_0900_ai_ci	16 384	0	0	0	47	~ 38
<input type="checkbox"/>	project_employee	InnoDB	utf8mb4_0900_ai_ci	16 384	32 768	0	0	16	~ 15
<input type="checkbox"/>	projects	InnoDB	utf8mb4_0900_ai_ci	16 384	0	0	0	8	~ 7
<input type="checkbox"/>	staffing_table	InnoDB	utf8mb4_0900_ai_ci	16 384	32 768	0	0	35	~ 32
	<b>Всего 11</b>	InnoDB	utf8mb4_0900_ai_ci	<b>180 224</b>	<b>163 840</b>	<b>0</b>			

На ней администратор видит все созданные базы данных и может перейти к их заполнению или изменению. Щелкнув по кнопке «Схема базы данных», администратор увидит схему имеющейся базы данных:



Щелкнув по одной из доступных таблиц, администратор увидит информацию об атрибутах, типах, индексах, внешних ключах и триггерах:

MySQL » Сервер » cursovik\_v2 » Таблица: employee

### Таблица: employee

[Выбрать](#)
[Показать структуру](#)
[Изменить таблицу](#)
[Новая запись](#)

поле	Тип	Комментарий
ID_code	bigint Автоматическое приращение	
Last_Name	tinytext NULL	
First_Name	tinytext NULL	
Otchestvo	tinytext NULL	
Staffing_Table	bigint NULL	
BirthDay	date NULL	
Premium	float NULL	

Индексы

PRIMARY	ID_code
INDEX	Staffing_Table

[Изменить индексы](#)

Внешние ключи

Источник	Цель	При стирании	При обновлении	
Staffing_Table	staffing_table(ID)	RESTRICT	RESTRICT	<a href="#">Изменить</a>

[Добавить внешний ключ](#)

Триггеры

AFTER	INSERT	employee_add_inf	Изменить	
AFTER	UPDATE	employee_auoldStaff	Изменить	
AFTER	UPDATE	employee_auNewStaff	Изменить	
BEFORE	DELETE	employee_bd_inf	Изменить	
BEFORE	DELETE	employee_bd_pass	Изменить	

[Добавить триггер](#)

Щелкнув по кнопке «Выбрать», администратор увидит наполнение данной таблицы:

### Выбрать: employee

[Выбрать](#)
[Показать структуру](#)
[Изменить таблицу](#)
[Новая запись](#)

[Выбрать](#)
[Поиск](#)
[Сортировать](#)

Лимит

Длина текста

Действие  
[Выбрать](#)

**SELECT \* FROM `employee` LIMIT 50** (0.003 s) [Редактировать](#)

	ID_code	Last_Name	First_Name	Otchestvo	Staffing_Table	BirthDay	Premium
<input type="checkbox"/>	1	Филиппов	Яков	Михалыч	1	1966-03-19	100000
<input type="checkbox"/>	3	Петров	Петр	Петрович	2	1989-03-19	0
<input type="checkbox"/>	4	Бурков	Алексей	Семенович	3	1990-03-19	NULL
<input type="checkbox"/>	5	Жаднов	Максим	Максимович	4	1990-11-19	NULL
<input type="checkbox"/>	6	Петрушкенкевич	Павел	Валерьевич	5	1990-06-19	NULL
<input type="checkbox"/>	7	Вавкин	Никит	Дмитриевич	6	1999-10-25	0

На этой странице видно, что данные можно изменять, удалять, копировать, так же можно их и добавлять. Для того чтобы добавить новые данные необходимо нажать на кнопку «Новая запись». После нажатия откроется новое окно, в котором будут представлены все атрибуты таблицы, для заполнения:

MySQL » Сервер » cursovik\_v2 » employee » Вставить

### Вставить: employee

ID_code	Автоматическое приращение	<input type="text"/>
Last_Name	<input type="text"/> NULL ▾	<input type="text"/>
First_Name	<input type="text"/> NULL ▾	<input type="text"/>
Otchestvo	<input type="text"/> NULL ▾	<input type="text"/>
Staffing_Table	<input type="text"/> NULL ▾	<input type="text"/>
BirthDay	<input type="text"/> NULL ▾	<input type="text"/>
Premium	<input type="text"/> NULL ▾	<input type="text"/>

Щелкнув по кнопке «Добавить триггер», администратор перейдет в специальное окно, где сможет создать необходимый ему триггер:

### Изменить триггер: employee\_add\_inf

Время	<input type="text"/> AFTER ▾
Событие	<input type="text"/> INSERT ▾
Тип	<input type="text"/> FOR EACH ROW ▾

Название:

```
CALL AddEmptyInf(NEW.ID_code)
```

Щелкнув по кнопке «SQL-запрос», администратор базы данных может написать необходимый SQL-запрос в специальном окне и исполнить его:

### SQL-запрос

```
SELECT * FROM Employee LIMIT 10
```

ID_code	Last_Name	First_Name	Otchestvo	Staffing_Table	BirthDay	Premium
1	Филиппов	Яков	Михалыч	1	1966-03-19	100000
3	Петров	Петр	Петрович	2	1989-03-19	0
4	Бурков	Алексей	Семенович	3	1990-03-19	NULL
5	Жаднов	Максим	Максимович	4	1990-11-19	NULL
6	Петрушкенкевич	Павел	Валерьевич	5	1990-06-19	NULL
7	Вавкин	Никит	Дмитриевич	6	1999-10-25	0
8	Баранов	Ашан	Михайлович	7	1995-10-25	NULL
9	Яковлев	Виктор	Викторович	8	1976-03-19	NULL
10	Бутусова	Антонина	Ильишна	9	1987-01-27	NULL
11	Сиянина	Пелагея	Афанасьевна	10	1991-04-01	NULL

10 строк (0.001 s) [Редактировать](#), [Экспорт](#)

```
SELECT * FROM Employee LIMIT 10
```

Выполнить

Лимит строк:  ☐ Остановить при ошибке ☐ Только ошибки



## Приложение

### Исходные тексты программы

#### App.py – основной файл приложения

```
from flask import Flask, render_template, request
from flask import Flask, render_template, request, redirect, url_for, flash
from flask_mysql import MySQL
app = Flask(__name__)

app.config['MYSQL_USER'] = 'root'
app.config['MYSQL_PASSWORD'] = 'root'
app.config['MYSQL_DB'] = 'cursovik_v2'
app.config['MYSQL_PORT'] = 3306
app.config['SECRET_KEY'] = '0ur_$uper^puper_$ecret_key!'
app.config['MYSQL_HOST'] = 'localhost'
mysql = MySQL(app)
priv = True

@app.route('/')
def index():
    return render_template('index.html')

@app.route('/index.html', methods=('GET', 'POST'))
def Login():
    if request.method == 'POST':
        log = request.form['Login']
        psswr = request.form['Password']
        if(log == 'root') and (psswr == 'root') or (log == 'bookkeeper') and (ps
swrd == '1234'):
            global priv
            priv = True
            if(log == 'bookkeeper'):
                priv = False
            return render_template('home.html', Privileges = priv)
        return render_template('index.html')

@app.route('/')
def GetProjById(id):
    cur = mysql.connection.cursor()
    query = "SELECT * FROM projects WHERE project_key = {0}".format(id)
    cur.execute(query)
    proj = cur.fetchone()
    cur.close()
    return proj

@app.route('/')
def GetEmployees():
    cur = mysql.connection.cursor()
    cur.execute("CALL GetEmployees()")
    out_rows = cur.fetchall()
    cur.close()
    return out_rows

@app.route('/')
def GetPositions():
    cur = mysql.connection.cursor()
```

```

cur.execute("SELECT * FROM Position")
out_rows = cur.fetchall()
cur.close()
return out_rows

@app.route('/')
def GetEmployee(id):
    cur = mysql.connection.cursor()
    query = "CALL GetEmplInfoById({0})".format(id)
    cur.execute(query)
    empl = cur.fetchone()
    cur.close()
    return empl

@app.route('/')
def GetEmployeeStaffById(id):
    cur = mysql.connection.cursor()
    query = "CALL GetEmployeeStaffById({0})".format(id)
    cur.execute(query)
    empl = cur.fetchone()
    cur.close()
    return empl

@app.route('/')
def GetEmplInfByProjId(id):
    cur = mysql.connection.cursor()
    query = "CALL GetEmplInfByProjId({0})".format(id)
    if(cur.execute(query) > 0):
        empls = cur.fetchall()
    else:
        empls = []
    cur.close()
    return empls

@app.route('/')
def GetTripById(id):
    cur = mysql.connection.cursor()
    query = "SELECT * FROM business_trips WHERE ID = {0} ".format(id)
    cur.execute(query)
    trip = cur.fetchone()
    cur.close()
    return trip

@app.route('/')
def GetEmpl(id):
    cur = mysql.connection.cursor()
    query = "SELECT * FROM Employee WHERE ID_CODE = {0}".format(id)
    cur.execute(query)
    empl = cur.fetchone()
    cur.close()
    return empl

@app.route('/')
def GetTripsByEmplId(id):
    cur = mysql.connection.cursor()
    query = "CALL GetTripsByEmplId({0})".format(id)
    len = cur.execute(query)
    if(len > 0):

```

```

        trips = cur.fetchall()
    else:
        trips = []
    cur.close()
    return trips

@app.route('/')
def GetProjsByEmplId(id):
    cur = mysql.connection.cursor()
    query = "CALL GetProjsByEmplId({0})".format(id)
    len = cur.execute(query)
    if(len > 0):
        projs = cur.fetchall()
    else:
        projs = []
    cur.close()
    return projs

@app.route('/')
def GetTransfersByEmplId(id):
    cur = mysql.connection.cursor()
    query = "CALL GetTransfersByEmplId({0})".format(id)
    len = cur.execute(query)
    if(len > 0):
        trans = cur.fetchall()
    else:
        trans = []
    cur.close()
    return trans

@app.route('/')
def GetPrivInfById(id):
    cur = mysql.connection.cursor()
    query = "CALL GetPrivInfById({0})".format(id)
    cur.execute(query)
    inf = cur.fetchone()
    cur.close()
    return inf

@app.route('/')
def GetDeptName(id):
    cur = mysql.connection.cursor()
    query = "SELECT name FROM department dep WHERE dep.ID = {0} ".format(id)
    cur.execute(query)
    name = cur.fetchone()
    cur.close()
    _name = []
    _name.append(name)
    _name.append(id)
    return _name

@app.route('/')
def GetStaffByDepId(id):
    cur = mysql.connection.cursor()
    query = "CALL GetStaffByDepId({0})".format(id)
    len = cur.execute(query)
    if(len > 0):
        staff = cur.fetchall()

```

```

        else:
            staff = []
            cur.close()
            return staff

@app.route('/')
def GetStaffAndDept():
    cur = mysql.connection.cursor()
    query = "CALL GetStaffAndDept()"
    cur.execute(query)
    staff = cur.fetchall()
    cur.close()
    return staff

@app.route('/')
def GetReports(id):
    cur = mysql.connection.cursor()
    query = "SELECT * FROM business_trips_report rep WHERE rep.Businnes_Trips = {
0}".format(
        id)
    len = cur.execute(query)
    if(len > 0):
        reps = cur.fetchall()
    else:
        reps = []
    cur.close()
    return reps

@app.route('/')
def GetPassport(id):
    cur = mysql.connection.cursor()
    query = "SELECT * FROM passport WHERE Employee = {0}".format(id)
    cur.execute(query)
    pas = cur.fetchone()
    cur.close()
    return pas

@app.route('/')
def GetEmplFromProjById(id):
    cur = mysql.connection.cursor()
    query = "CALL GetEmplFromProjById({0})".format(id)
    cur.execute(query)
    empls = cur.fetchall()
    cur.close()
    return empls

@app.route('/Employees/<int:ID_code>')
def GetEmplInfo(ID_code):
    empl = GetEmployee(ID_code)
    trips = GetTripsByEmplId(ID_code)
    inf = GetPrivInfById(ID_code)
    projs = GetProjsByEmplId(ID_code)
    trans = GetTransfersByEmplId(ID_code)
    pas = GetPassport(ID_code)
    return render_template('Employees/employee.html', Employee=empl, Trips=trips,
Information=inf, Projects=projs, Transfers=trans, Passport=pas, Privileges = pri
v)

```

```

@app.route('/Deps/<int:id>')
def GetDeptInf(id):
    name = GetDeptName(id)
    staff = GetStaffByDepId(id)
    return render_template('Deps/Dept.html', Department=name, Staffs=staff, Privi
liges = priv)

@app.route('/Projects/<int:id>')
def GetProjInf(id):
    proj = GetProjById(id)
    empls = GetEmplInfByProjId(id)
    return render_template('Projects/Proj.html', Project = proj, Empls = empls, P
rivileges = priv)

@app.route('/BusinessTrips/<int:id>')
def GetTripRep(id):
    reps = GetReports(id)
    return render_template('BusinessTrips/Trip.html', Reps=reps, ID = id, Privili
ges = priv)

@app.route('/Employees')
def EmplList():
    empls = GetEmployees()
    return render_template('Employees/list.html', Employees=empls, Privileges = p
riv)

@app.route('/Projects')
def ProjsList():
    projs = GetProjs()
    return render_template('Projects/list.html', Projects=projs, Privileges = pri
v)

@app.route('/')
def GetProjs():
    cur = mysql.connection.cursor()
    cur.execute("SELECT * FROM projects")
    projs = cur.fetchall()
    cur.close()
    return projs

@app.route('/')
def GetDeps():
    cur = mysql.connection.cursor()
    cur.execute("SELECT * FROM Department")
    deps = cur.fetchall()
    cur.close()
    return deps

@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('/')
def GetBusinessTripsAndEmpl():
    cur = mysql.connection.cursor()
    cur.execute("call GetTripsAndEmpl()")
    rows = cur.fetchall()
    cur.close()
    return rows

@app.route('/Deps')
def DepsList():
    deps = GetDeps()
    return render_template('Deps/list.html', Deps=deps, Priviliges = priv)

@app.route('/SalaryReport')
def GetAVGSalByDepReport():
    sal = GetAVGSalByDep()
    return render_template('SalaryReport/list.html', Results=sal, Priviliges = pr
iv)

@app.route('/BusinessReport')
def GetAVGBusByDepReport():
    res = GetAVGBusByDep()
    return render_template('BusinessReport/list.html', Results=res, Priviliges =
priv)

@app.route('/BusinessTrips')
def GetTripsAndEmplList():
    rows = GetBusinessTripsAndEmpl()
    return render_template('BusinessTrips/list.html', Results=rows, Priviliges =
priv)

@app.route('/Employees/create', methods=('GET', 'POST'))
def AddEmployee():
    if request.method == 'POST':
        f_name = request.form['First_Name']
        s_name = request.form['Last_Name']
        otch = request.form['Otchestvo']
        staff = request.form['Staffing_Table']
        brd = request.form['BirthDay']
        prem = request.form['Premium']
        if not s_name:
            flash('Вы не добавили фамилию!')
        if not f_name:
            flash('Вы не добавили имя!')
        if not otch:
            flash('Вы не добавили отчество!')
        if not staff:
            flash('Вы не добавили штатное расписание!')
        if not brd:
            flash('Вы не добавили дату рождения!')
        else:
            cur = mysql.connection.cursor()

```



```

        cur.close()
        flash('ok!')
        return redirect(url_for('GetEmplInfo', ID_code = id))
    inf = GetPrivInfById(id)
    return render_template('Employees/EditPrivInf.html', Inform = inf, Privileges
= priv)

@app.route('/Employees/AddPassport<int:id>', methods=('GET', 'POST'))
def AddPassport(id):
    if request.method == 'POST':
        num = request.form['Number']
        date = request.form['Date']
        inst = request.form['Inst']
        if not num:
            flash('Вы не добавили код и номер паспорта!')
        if not date:
            flash('Вы не добавили дату выдачи!')
        if not inst:
            flash('Вы не добавили кем выдано!')
        cur = mysql.connection.cursor()
        query = "INSERT INTO passport (Number, Issue_Date, Institution, Employee)
VALUES('{1}', '{2}', '{3}', {0})".format(id, num, date, inst)
        cur.execute(query)
        mysql.connection.commit()
        cur.close()
        flash('ok!')
        return redirect(url_for('GetEmplInfo', ID_code = id))
    return render_template('Employees/AddPassport.html', Privileges = priv)

@app.route('/Deps/create', methods=('GET', 'POST'))
def AddDept():
    if request.method == 'POST':
        name = request.form['Name']

        if not name:
            flash('Вы неввели название отдела!')

        cur = mysql.connection.cursor()
        query = "INSERT INTO department (Name) VALUES('{0}').format(name)
        cur.execute(query)
        mysql.connection.commit()
        cur.close()
        flash('ok!')
        return redirect(url_for('DepsList'))
    return render_template('Deps/create.html', Privileges = priv)

@app.route('/Projects/create', methods=('GET', 'POST'))
def AddProj():
    if request.method == 'POST':
        name = request.form['Name']

        if not name:
            flash('Вы неввели название проекта!')

        cur = mysql.connection.cursor()
        query = "INSERT INTO projects (project_name) VALUES('{0}').format(name)
        cur.execute(query)
        mysql.connection.commit()

```



```

        cur.close()
        flash('ok!')
        return redirect(url_for('ProjsList'))
    return render_template('Projects/create.html', Priviliges = priv)

@app.route('/Projects/<int:id>/edit', methods=('GET', 'POST'))
def EditProj(id):
    if request.method == 'POST':
        name = request.form['Name']

        if not name:
            flash('Вы не ввели название проекта!')

        cur = mysql.connection.cursor()
        query = "UPDATE projects SET project_name = '{0}' WHERE project_key = {1}"
        .format(name, id)
        cur.execute(query)
        mysql.connection.commit()
        cur.close()
        flash('ok!')
        return redirect(url_for('ProjsList'))
    proj = GetProjById(id)
    return render_template('Projects/edit.html', Project = proj, Priviliges = pri
v)

@app.route('/Employees/<int:id>/delete')
def DeleteEmployee(id):
    cur = mysql.connection.cursor()
    query = "DELETE FROM Employee WHERE ID_code = {0}".format(id)
    cur.execute(query)
    mysql.connection.commit()
    cur.close()
    return redirect(url_for('EmplList'))

@app.route('/Deps/AddStaffTable<int:id>', methods=('GET', 'POST'))
def AddStaffTable(id):
    if request.method == 'POST':
        pos = request.form['Position']
        num = request.form['Number']
        if not num:
            flash('Вы не указали квоту!')
        if not pos:
            flash('Вы не указали должность!')
        cur = mysql.connection.cursor()
        query = "INSERT INTO staffing_table (Position, Count, Dept) VALUES({0}, {
1}, {2})".format(pos, num, id)
        cur.execute(query)
        mysql.connection.commit()
        cur.close()
        flash('ok!')
        return redirect(url_for('GetDeptInf', id = id))
    pos = GetPositions()
    return render_template('Deps/AddStaffTable.html', Pos = pos, Priviliges = pri
v)

@app.route('/BusinessTrips/create', methods=('GET', 'POST'))
def AddBusinessTrip():
    if request.method == 'POST':

```

```

empl = request.form['Employee']
city = request.form['City']
targ = request.form['Target']
s_date = request.form['StartDate']
e_date = request.form['EndDate']
expense = request.form['Expense']

if not empl:
    flash('Вы не указали работника!')
if not city:
    flash('Вы не указали город!')
if not targ:
    flash('Вы не указали цель!')
if not s_date:
    flash('Вы не указали дату начала!')
if not e_date:
    flash('Вы не указали дату окончания!')
if not expense:
    flash('Вы не указали аванс!')
else:
    cur = mysql.connection.cursor()
    query = "INSERT INTO business_trips (Employee, City, Target, Start_Date, End_Date, Prepaid_Expense) \n \
VALUES ({0}, '{1}', '{2}', '{3}', '{4}', {5})".format(empl, city, targ, s_date, e_date, expense)
    cur.execute(query)
    mysql.connection.commit()
    cur.close()
    flash('ok!')
    return redirect(url_for('GetTripsAndEmplList'))
empls = GetEmployees()
return render_template('BusinessTrips/create.html', Empls = empls, Privileges = priv)

@app.route('/BusinessTrips/edit<int:ID>', methods=('GET', 'POST'))
def EditBusinessTrip(ID):
    if request.method == 'POST':
        empl = request.form['Employee']
        city = request.form['City']
        targ = request.form['Target']
        s_date = request.form['StartDate']
        e_date = request.form['EndDate']
        expense = request.form['Expense']
        if not empl:
            flash('Вы не указали работника!')
        if not city:
            flash('Вы не указали город!')
        if not targ:
            flash('Вы не указали цель!')
        if not s_date:
            flash('Вы не указали дату начала!')
        if not e_date:
            flash('Вы не указали дату окончания!')
        if not expense:
            flash('Вы не указали аванс!')
        else:
            cur = mysql.connection.cursor()

```

```

        query = "CALL UPDBusinessTrip({0}, '{1}', '{2}', '{3}', '{4}', '{5}'
, {6})".format(ID, empl, city, targ, s_date, e_date, expense)
        cur.execute(query)
        mysql.connection.commit()
        cur.close()
        flash('ok!')
        return redirect(url_for('GetTripsAndEmplList'))
    empls = GetEmployees()
    trip = GetTripById(ID)
    return render_template('BusinessTrips/edit.html', Trip = trip, Empls = empls
, Privileges = priv)

@app.route("/Deps/<int:id>", methods= ['POST'])
def DeleteDept(id):
    """Функция-представление для удаления отдела"""
    cur = mysql.connection.cursor()
    query = "DELETE FROM department WHERE ID = {0}".format(id)
    cur.execute(query)
    mysql.connection.commit()
    cur.close()
    flash("Отдел удален!")
    return redirect(url_for('DepsList'))

@app.route("/Projects/<int:id>", methods= ['POST'])
def DeleteProj(id):
    """Функция-представление для удаления отдела"""
    cur = mysql.connection.cursor()
    query = "DELETE FROM projects WHERE project_key = {0}".format(id)
    cur.execute(query)
    mysql.connection.commit()
    cur.close()
    flash("Проект удален!")
    return redirect(url_for('ProjsList'))

@app.route("/BuisnessTrips/<int:id>", methods= ['POST'])
def DeleteBuisnessTrip(id):
    """Функция-представление для удаления отдела"""
    cur = mysql.connection.cursor()
    query = "DELETE FROM business_trips WHERE ID = {0}".format(id)
    cur.execute(query)
    mysql.connection.commit()
    cur.close()
    flash("Командировка удалена!")
    return redirect(url_for('GetTripsAndEmplList'))

@app.route("/Employees/<int:id><int:ID_c>", methods= ['POST'])
def DeleteEmplTransfer(id, ID_c):
    """Функция-представление для удаления отдела"""
    cur = mysql.connection.cursor()
    query = "DELETE FROM employee_transfers WHERE ID = {0}".format(id)
    cur.execute(query)
    mysql.connection.commit()
    cur.close()
    return redirect(url_for('GetEmplInfo', ID_code = ID_c))

@app.route('/BusinessTrips/createReport<int:id>', methods=('GET', 'POST'))
def AddBusinessTrips(id):
    if request.method == 'POST':

```

```

        date = request.form['Date']
        purp = request.form['Purpose']
        val = request.form['Value']
        if not date:
            flash('Вы не указали дату!')
        if not purp:
            flash('Вы не указали обоснование!')
        if not val:
            flash('Вы не указали обоснование!')
        cur = mysql.connection.cursor()
        query = "INSERT INTO business_trips_report (Date, Purpose_Payment, Value
, Businnes_Trips) VALUES('{0}', '{1}', {2}, {3})".format(date, purp, val, id)
        cur.execute(query)
        mysql.connection.commit()
        cur.close()
        flash('ok!')
        return redirect(url_for('GetTripRep', id = id))
    return render_template('BusinessTrips/createReport.html', Priviliges = priv)

@app.route('/Employees/createTransfer<int:ID_code>', methods=('GET', 'POST'))
def createTransfer(ID_code):
    if request.method == 'POST':
        justi = request.form['Justification']
        num = request.form['Order_Number']
        date = request.form['Order_Date']
        staff = request.form['Staffing_Table']

        if not justi:
            flash('Вы не добавили причину перевода')
        if not num:
            flash('Вы не добавили номер приказа')
        if not date:
            flash('Вы не добавили дату приказа!')
        if not staff:
            flash('Вы не добавили штатное расписание!')
        else:
            cur = mysql.connection.cursor()
            query = "CALL InsEmplTrans({0}, {1}, '{2}', '{3}', '{4}')".format(ID
_code, staff, justi, num, date)
            cur.execute(query)
            mysql.connection.commit()
            cur.close()
            flash('ok!')
            return redirect(url_for('GetEmplInfo', ID_code = ID_code))
    staff = GetStaffAndDept()
    return render_template('Employees/createTransfer.html', Staff = staff, Privi
liges = priv)

@app.route('/Project/AddEmplOnProj<int:id>', methods=('GET', 'POST'))
def AddEmplOnProj(id):
    if request.method == 'POST':
        empl = request.form['Employee']
        num = request.form['Number']

        if not empl:
            flash('Вы не указали работника!')
        cur = mysql.connection.cursor()

```

```

        query = "INSERT INTO project_employee (Project, Employee, Salary) VALUES(
{0}, {1}, {2})".format(id, empl, num)
        cur.execute(query)
        mysql.connection.commit()
        cur.close()
        flash('ok!')
        return redirect(url_for('GetProjInf', id = id))
    empls = GetEmployees()
    return render_template('Projects/AddEmplOnProj.html', Empls = empls, Privilig
es = priv)

@app.route("/Projects/<int:id><int:ID_code>", methods= ['POST'])
def DeleteFromProj(id, ID_code):
    cur = mysql.connection.cursor()
    query = "DELETE FROM project_employee WHERE (Employee = {0}) AND (Project = {
1})".format(ID_code, id)
    cur.execute(query)
    mysql.connection.commit()
    cur.close()
    return redirect(url_for('GetProjInf', id = id))

app.run(debug=True)

```

### base.html – отвечает за навигационную панель

```

<!doctype html>
<html lang="ru">
    <head>
        <!-- Required meta tags -->
        <meta charset="utf-8">
        <meta name="viewport" content="width=device-width, initial-scale=1, shrink-
to-fit=no">

        <!-- Bootstrap CSS -->
        <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3
.1/css/bootstrap.min.css" integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="an
onymous">

        <title>{% block title %} {% endblock %}</title>
    </head>
    <body>
        <nav class="navbar navbar-expand-md navbar-light bg-light">
            <a class="navbar-brand" href="{{ url_for('index') }}"></a>
            <div class="collapse navbar-collapse" id="navbarNav">
                <ul class="navbar-nav">
                    {%if Priviliges%}
                    <li class="nav-item">
                        <a class="nav-
link" href="{{url_for('EmplList')}}">Работники</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link" href="{{url_for('DepsList')}}">Отделы</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-
link" href="{{url_for('GetTripsAndEmplList')}}">Командировки</a>

```

```

        </li>
        <li class="nav-item">
            <a class="nav-link" href="{url_for('ProjsList')}}">Проекты</a>
        </li>
        {%else%}

        <li class="nav-item">
            <a class="nav-
link" href="{url_for('GetAVGSalByDepReport')}}">Средний оклад по отделам</a>
        </li>
        <li class="nav-item">
            <a class="nav-
link" href="{url_for('GetAVGBusByDepReport')}}">Средние командировочные по отдел
ам</a>
        </li>
        {%endif%}
    </ul>
</div>
</nav>
<div class="container">
    {% block content %} {% endblock %}
</div>

<!-- Optional JavaScript -->
<!-- jQuery first, then Popper.js, then Bootstrap JS -->
<script src="https://code.jquery.com/jquery-
3.3.1.slim.min.js" integrity="sha384-
q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo" crossorigin="an
onymous"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popp
er.min.js" integrity="sha384-
U02eT0CpHqdSJJQ6hJty5KVphtPhzWj9W01clHTMGa3JDZwrnQq4sF86dIHNDz0W1" crossorigin="an
onymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.
min.js" integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy60rQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM" crossorigin="an
onymous"></script>
</body>
</html>

```

**list.html** – отвечает рендер каждой записи БД (на примере вкладки с работниками)

```

{% extends 'base.html' %}

{% block content %}

<h1>{% block title %}Работники{% endblock %}</h1>
<a href="{ url_for('AddEmployee') }">
    <span class="btn btn-success btn-sm">Добавить работника</span>
</a>

<hr>

{% for empl in Employees %}

<div class="alert alert-danger">
    <p>

```

```

        Код работника: {{empl[0]}} <br>
        Фамилия: {{empl[1]}} <br>
        Имя: {{empl[2]}} <br>
        Отчество: {{empl[3]}} <br>
        Должность: {{empl[4]}} <br>
        Отдел: {{empl[5]}} <br>
    </p>
    <a href="{{ url_for('GetEmplInfo', ID_code = empl[0]) }}">
        <span class="btn btn-warning btn-sm">Подробнее</span>
    </a>

</div>

{% endfor %}
{% endblock %}

```

**create.html** – отвечает рендер окна добавления записи (на примере создания отдела)

```

{% extends 'base.html' %}

{% block content %}
<h1>{% block title %} Добавление нового отдела {% endblock %}</h1>

<form method="post">
    <div class="form-group">
        <label for="Name">Название отдела</label>
        <input type="text" name="Name"
            placeholder="Name" class="form-control"
            value=''></input>
    </div>

    <div class="form-group">
        <button type="submit" class="btn btn-primary">Добавить отдел</button>
    </div>
</form>
{% endblock %}

```

## **edit.html – отвечает рендер окна редактирования записи (на примере вкладки с проектами)**

```
{% extends 'base.html' %}

{% block content %}
<h1>{% block title %} Редактирование названия проекта {{Project[1]}} {% endblock %}</h1>

<form method="post">
  <div class="form-group">
    <label for="Name">Название проекта</label>
    <input type="text" name="Name"
      placeholder="Name" class="form-control"
      value='{{Project[1]}}'></input>
  </div>

  <div class="form-group">
    <button type="submit" class="btn btn-primary">Изменить проект</button>
  </div>
</form>
{% endblock %}
```