Министерство образования и науки РФ

Пермский национальный исследовательский политехнический университет

Электротехнический факультет

Кафедра Информационные технологии и автоматизированные системы

Базы данных

Лабораторная работа № 6

Тема: «Создание представления в СУБД MySQL»

Выполнила: студентка группы

РИС-23-2б

Жилина А.А.

Проверил: доцент кафедры ИТАС

Петренко А.А.

г. Пермь – 2024

Оглавление

Цель работы 3

Задачи работы 4

Этапы выполнения 5

Вывод 10

Заключение 11

Список используемой литературы 12

# Цель работы

# Создание и использование представлений в СУБД *MySQL* в веб-приложении *phpMyAdmin*.

# Задачи работы

Для достижения поставленной цели необходимо выполнить следующие задачи:

1. Изучить понятие *VIEW*;

2. Реализовать представления на базе запросов 2 лабораторной работы в количестве 3 штук;

3. Проверить работу представлений.

# Этапы выполнения

1. Понятие *VIEW* и его объявление на языке *SQL*:

Представление (*VIEW*) — объект базы данных, являющийся результатом выполнения запроса к базе данных, определенного с помощью оператора *SELECT*, в момент обращения к представлению.

Представления иногда называют «виртуальными таблицами». Такое название связано с тем, что представление доступно для пользователя как таблица, но само оно не содержит данных, а извлекает их из таблиц в момент обращения к нему. Если данные изменены в базовой таблице, то пользователь получит актуальные данные при обращении к представлению, использующему данную таблицу; кэширования результатов выборки из таблицы при работе представлений не производится. При этом, механизм кэширования запросов (*query* *cache*) работает на уровне запросов пользователя безотносительно к тому, обращается ли пользователь к таблицам или представлениям.

Синтаксис создания представления в *MySQL*:

CREATE [OR REPLACE]  
[ALGORITHM = {UNDEFINED | MERGE | TEMPTABLE}]  
VIEW view\_name [(column\_list)]  
AS select\_statement  
[WITH [CASCADED | LOCAL] CHECK OPTION]  
Оператор *CREATE VIEW* содержит 4 необязательные конструкции:

1. *OR REPLACE* — при использовании данной конструкции в случае существования представления с таким именем старое будет удалено, а новое создано. В противном случае возникнет ошибка, информирующая о существовании представления с таким именем и новое представление создано не будет. Следует отметить одну особенность — имена таблиц и представлений в рамках одной базы данных должны быть уникальны, т.е. нельзя создать представление с именем уже существующей таблицы. Однако конструкция *OR REPLACE* действует только на представления и замещать таблицу не будет.
2. *ALGORITM* — определяет алгоритм, используемый при обращении к представлению.
3. *column\_list* — задает имена полей представления.
4. *WITH CHECK OPTION* — при использовании данной конструкции все добавляемые или изменяемые строки будут проверяться на соответствие определению представления. В случае несоответствия данное изменение не будет выполнено. Обратите внимание, что при указании данной конструкции для необновляемого представления возникнет ошибка и представление не будет создано.
5. Реализация представлений в СУБД *MySQL* в веб-приложении *phpMyAdmin*:

Для реализации представлений, обратимся к таблицам лабораторной работы №2 (См. Рисунок 1, 2), и опишем каждое представление на языке *SQL*:

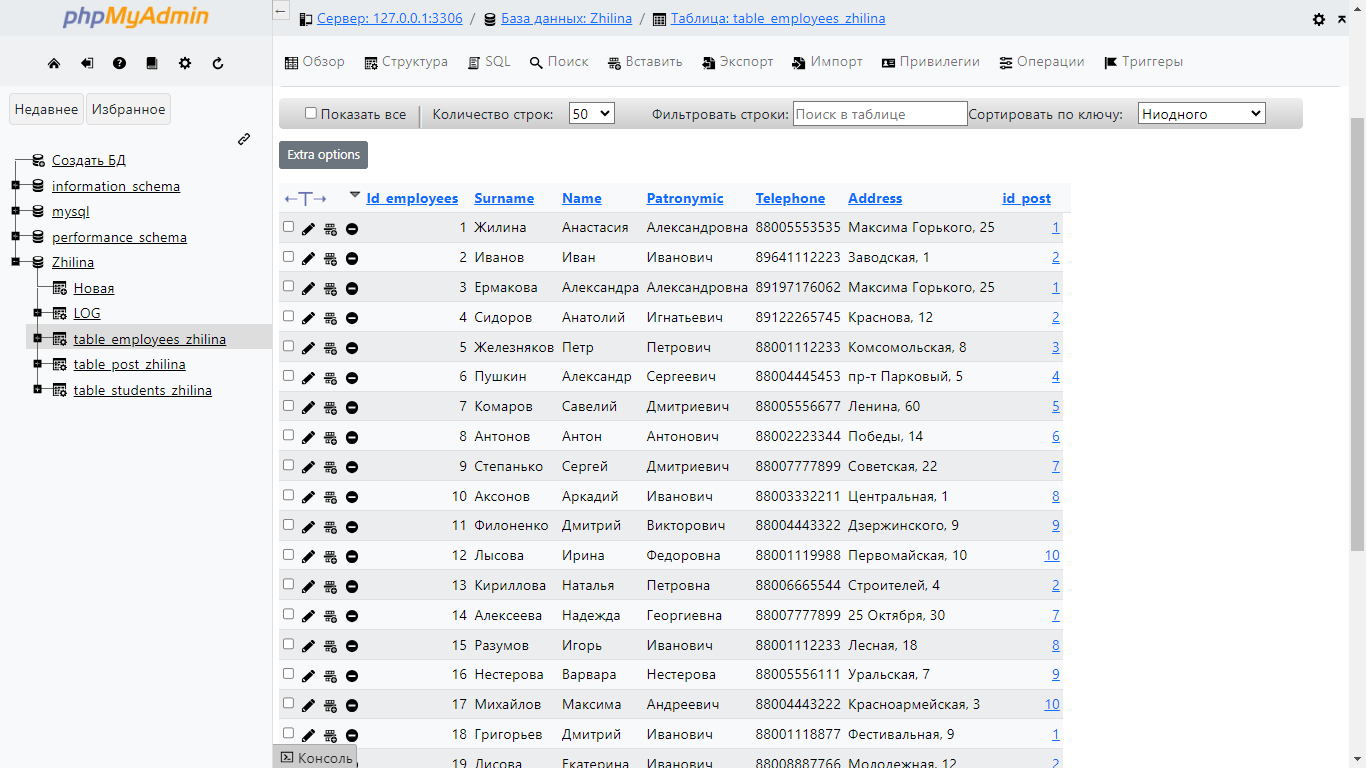


Рисунок 1 - таблица [table\_employees\_zhilina](http://127.0.0.1/openserver/phpmyadmin/index.php?route=/sql&pos=0&db=Zhilina&table=table_employees_zhilina)

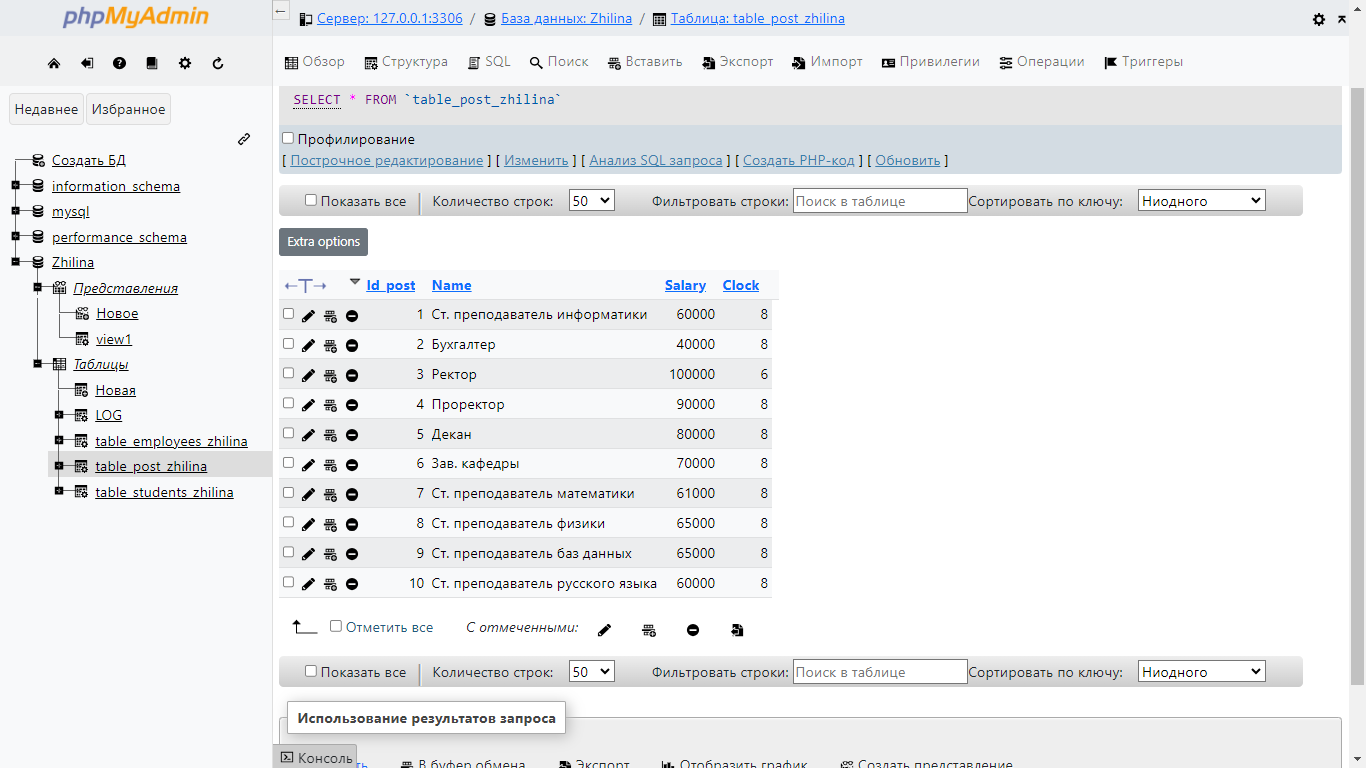


Рисунок 2 - таблица [table\_post\_zhilina](http://127.0.0.1/openserver/phpmyadmin/index.php?route=/sql&pos=0&db=Zhilina&table=table_post_zhilina)

* 1. Представление таблиц [*table\_employees\_zhilina*](http://127.0.0.1/openserver/phpmyadmin/index.php?route=/sql&pos=0&db=Zhilina&table=table_employees_zhilina) и [*table\_post\_zhilina*](http://127.0.0.1/openserver/phpmyadmin/index.php?route=/sql&pos=0&db=Zhilina&table=table_post_zhilina)
  2. по запросу вывода полного имени, номера телефона и зарплаты (См. Рисунок 3).

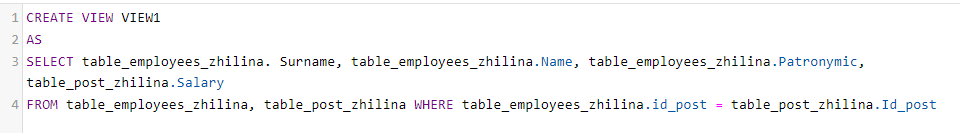


Рисунок 3 - реализация представления по запросу 1

*CREATE VIEW* – ключевые слова для создания запроса.

*SELECT* – выборка полей для вывода виртуальной таблицы.

*FROM* – ключевое слово относящееся для выбора переменных из таблицы.

* 1. Представление таблицы [*table\_employees\_zhilina*](http://127.0.0.1/openserver/phpmyadmin/index.php?route=/sql&pos=0&db=Zhilina&table=table_employees_zhilina) по запросу вывода полного имени и адреса проживания, отсортированной по адресу (См. Рисунок 4)

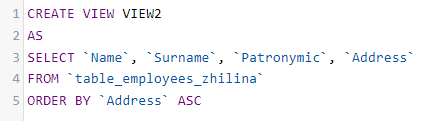


Рисунок 4 - реализация представления по запросу 2

*ORDER BY* – ключевое слово для сортировки по переменной.

* 1. Представление таблиц *table\_employees\_zhilina* и *table\_post\_zhilina* по запросу вывода полного имени и времени работы, где работа принимает число больше 4 (См. Рисунок 5)

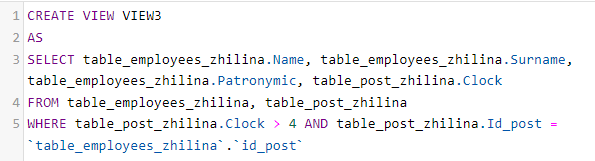


Рисунок 5 – реализация представления по запросу 3

*WHERE –* ключевое слово для логических операций.

1. Проверка работы представлений:

Проверим получившиеся представления (См. Рисунок 6,7,8).

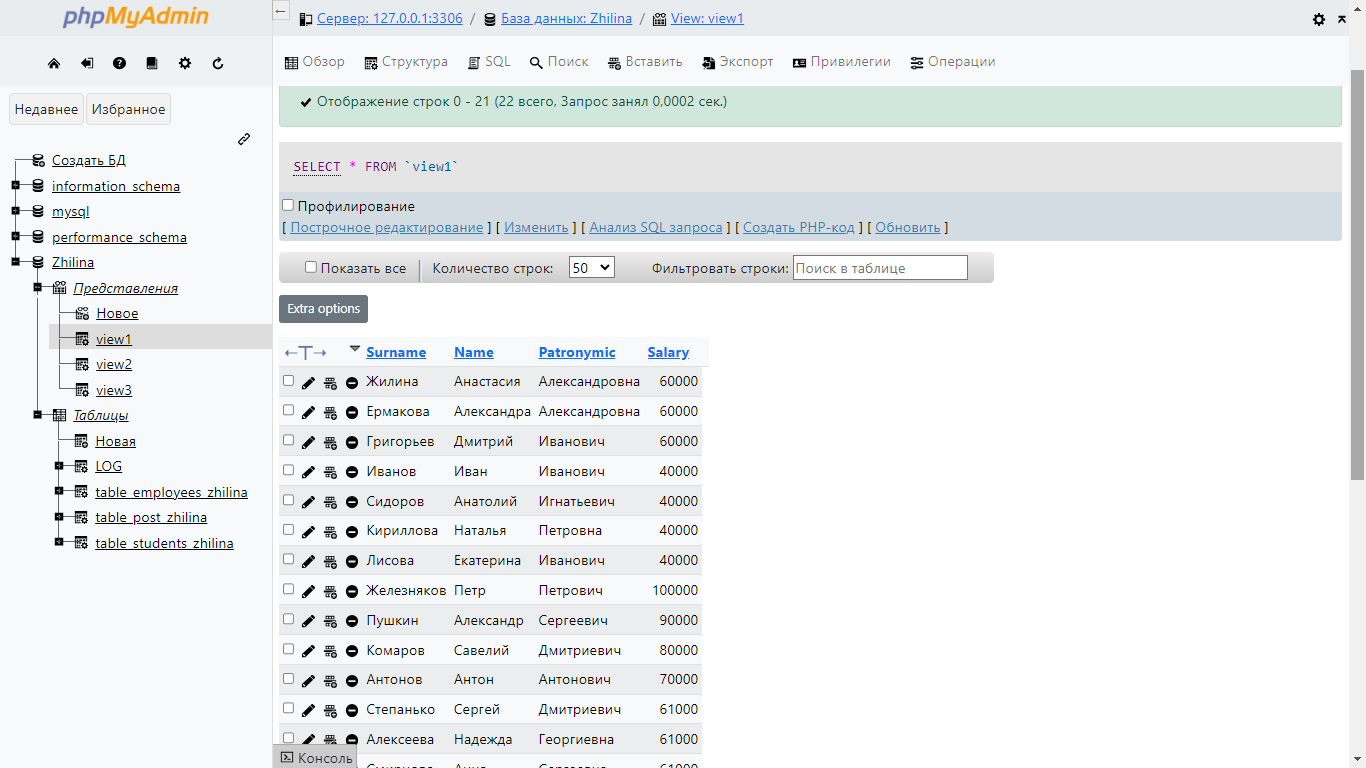
**

Рисунок 6 – представление view1

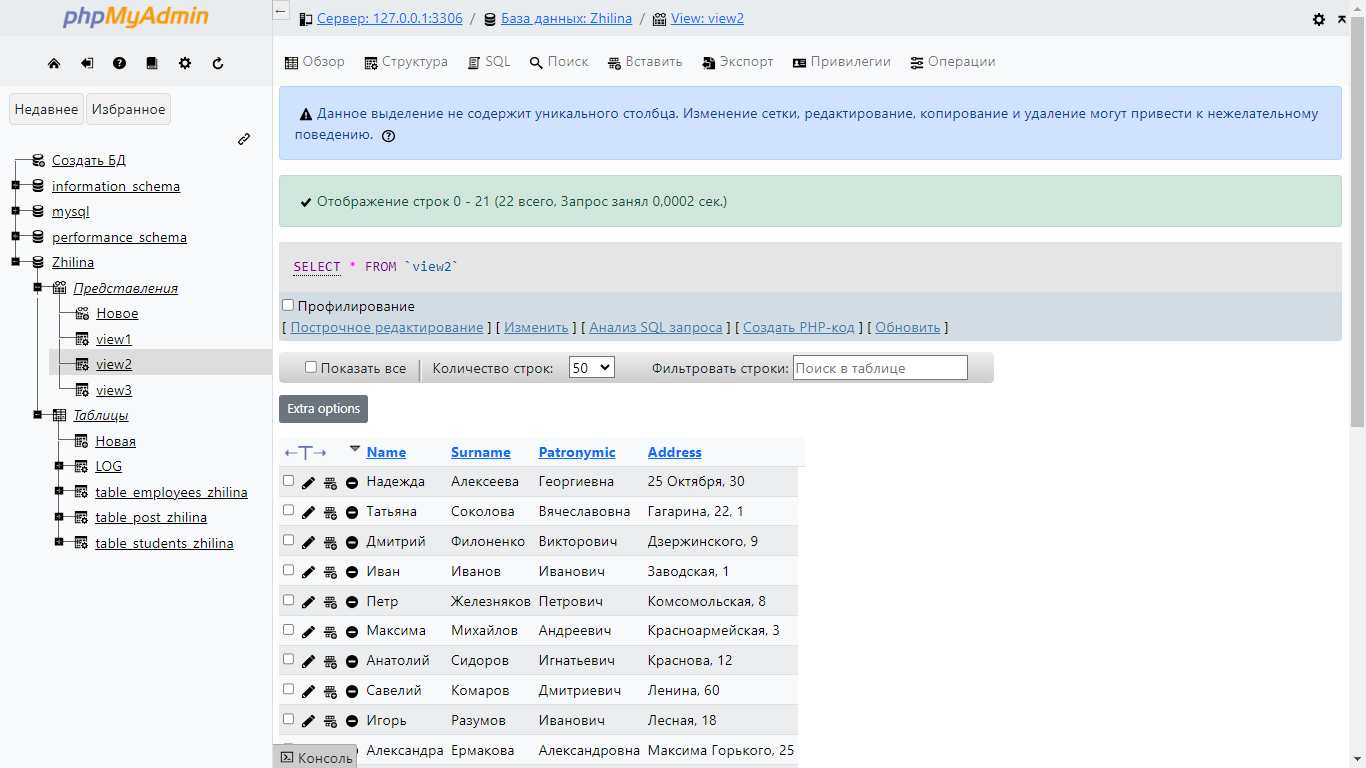
**

Рисунок 7 – представление view2

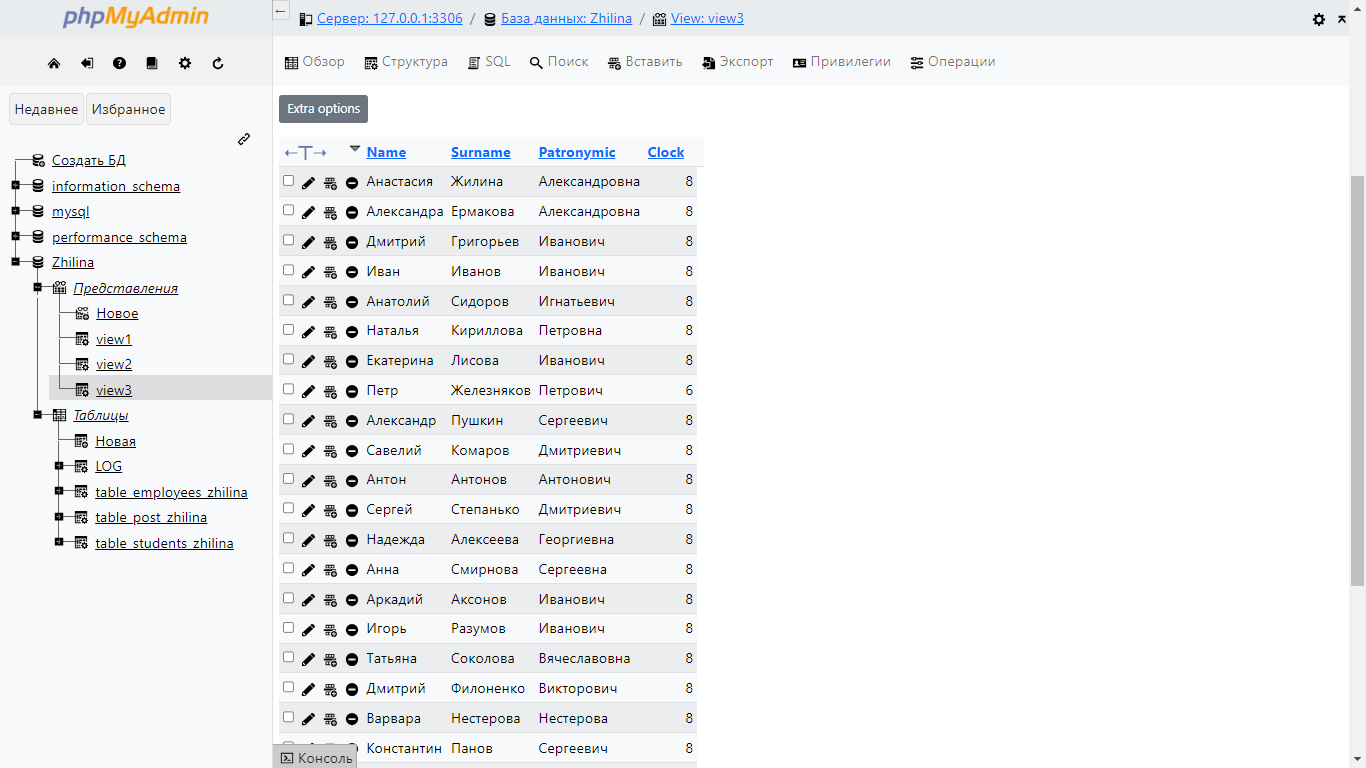
**

Рисунок 8 – представление view3

# Вывод

В результате работы были получены навыки работы с представлениями на языке *SQL* в таблицах СУБД *MySQL* в веб-приложении *phpMyAdmin*.

# Заключение

Результатом проведенной работы, являются:

1. Разработанная база данных.
2. Основные таблицы *table\_employees\_zhilina* и *table\_post\_zhilina*.
3. Представления запросов к таблицам *table\_employees\_zhilina* и *table\_post\_zhilina* в количестве 3 штук.

# Список используемой литературы

1. *Копейкин М.В., Спиридонов В.В., Шумова Е.О. Базы данных. Основы SQL реляционных баз данных: Учебное пособие. - СПб.: СЗТУ, 2005. - 160 с.*
2. *Э.В. Сысоев, Е.В. Бурцева. Базы данных: лекции к курсу. - Тамбов : Изд-во Тамб. гос. техн. ун-та, 2007. - 48 с.*
3. *Документация phpMyAdmin, FAQ phpMyAdmin.*