МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ   
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ОРЛОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
имени И.С. ТУРГЕНЕВА»

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

по направлению подготовки 01.03.02 Прикладная математика и информатика

направленность (профиль) Системное программирование   
и компьютерные технологии

Студента Тютюнова Игоря Андреевича шифр 150710

Факультет физико-математический

Тема выпускной квалификационной работы

РАЗРАБОТКА МЕТОДОВ ОПТИМИЗАЦИИ ВЗАИМОДЕЙСТВИЯ ВЕБ-САЙТА С БАЗОЙ ДАННЫХ

Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Тютюнов И.А.

Научный   
руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Федяев Ю.С.

Зав. кафедрой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Дорофеева В.И.

Орёл 2022

**Содержание**

[ВВЕДЕНИЕ 3](#_Toc102406910)

[ГЛАВА 1. ТЕОРИТИЧЕСКИЕ СВЕДЕНИЯ ОБ ОПТИМИЗАЦИИ РАБОТЫ С БАЗОЙ ДАННЫХ 7](#_Toc102406911)

[1.1. Общая архитектура реляционных СУБД 7](#_Toc102406912)

[1.1.1. Структура данных 7](#_Toc102406913)

[1.1.2. Кластеризация 7](#_Toc102406914)

[1.1.3. Индексирование 7](#_Toc102406915)

[1.1.4. Кэширование в базах данных 7](#_Toc102406916)

[1.2. Оптимизация запросов 7](#_Toc102406917)

[1.2.1. Синтаксическая оптимизация запросов 7](#_Toc102406918)

[1.2.2. Семантическая оптимизация запросов 7](#_Toc102406919)

[1.2.3. Генерация и выбор плана выполнения 7](#_Toc102406920)

[1.2.4. Практика написания эффективных SQL-запросов 7](#_Toc102406921)

[ГЛАВА 2. РАЗРАБОТКА ИНСТРУМЕНТА ДЛЯ РАСПРЕДЕЛЕНИЯ НАГРУЗКИ НА БАЗУ ДАННЫХ В ВЫСОКОНАГРУЖЕННЫХ ИНФОРМАЦИОННЫХ СИСТЕМАХ 8](#_Toc102406922)

[2.2.1. Теоретические сведения об использовании брокера очередей RabbitMQ 8](#_Toc102406923)

[2.2.2. Разработка инструмента для распределения нагрузки на базу данных с использованием брокера очередей RabbitMQ 8](#_Toc102406924)

[ЗАКЛЮЧЕНИЕ 9](#_Toc102406925)

[СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ 10](#_Toc102406926)

[ПРИЛОЖЕНИЕ 11](#_Toc102406927)

# ВВЕДЕНИЕ

**Обоснование выбора темы и ее актуальность**

Нужно обязательно изменить

Кандидат на удаление

На заре повсеместного внедрения информационных технологий в научные, производственные, образовательные и другие сферы человеческой деятельности, очень остро начал подниматься вопрос хранения и эффективной обработки больших объемов данных. Технологии хранения данных, на тот период времени, доставляли множество неудобств разработчикам программного обеспечения, т.к. разработка каждой прикладной программы, требующей сохранения данных на внешний носитель, требовала собственной реализации логической структуры хранения блоков данных.

Концептуально новым шагом в развитии информационных технологий стало внедрение понятия файла, как именованной области данных, и, файловых систем, как регламента, определяющего способ организации и хранения файлов, а также доступа к содержащемся в них данным [5]. Файловые системы связывают физическое расположение данных на носителе информации с прикладными программами, посредствам интерфейса программирования приложений (Application Programming Interface - API), предоставляемым драйвером файловой системы.

Однако эти средства все же не позволяли полностью избавиться от описанных выше проблем. В связи с тем, что структуры данных, которыми оперировали новые программные комплексы, постоянно усложнялись, разработчики программного обеспечения были вынуждены, как и раньше, создавать индивидуальные средства структуризации данных для каждого своего программного продукта. Для того чтобы повысить эффективность разработки программных продуктов, работающих со сложно структурированными данными, было необходимо средство, позволяющее обобщить элементы информационных систем (ИС), отвечающих за работу с этими данными. Это логичное и вполне рациональное стремление положило начало введения нового термина - Базы данных.

Итак, База данных — это совокупность связанных данных, организованных по определенным правилам, предусматривающим общие принципы описания, хранения и манипулирования, независимая от прикладных программ.

Очевидно, что такая методология является целесообразным решением задачи обобщения принципов работы с данными. Изначально данные в базах данных хранились в виде древовидных иерархических структур, в которых малые компоненты являются частью более крупных, которые, в свою очередь, могли быть частью других - таким образом, собиралась единая система данных.

Сегодня SQL является стандартным инструментом управления данными, включающим в себя операторы определения данных (Data Definition Language, DDL), операторы манипуляции данными (Data Manipulation Language, DML), операторы определения доступа к данным (Data Control Language, DCL), операторы управления транзакциями (Transaction Control Language, TCL).

В современном мире трудно переоценить значимость баз данных. Базы данных прочно укрепили свои позиции (и продолжают все активнее применяться) повсеместно в промышленных, образовательных, здравоохранительных, правоохранительных и в др. общественно важных структурах, а также в сферах бизнеса.

Язык SQL, как основной инструмент работы с базами данных, играет первостепенную роль в вопросе разработки СУБД, а также администрирования баз данных. Так как базы данных во многих сферах могут достигать очень больших объемов (как по количеству таблиц, записей и полей, так и по общему объему хранимой информации), то естественно встает вопрос об эффективной, с точки зрения быстродействия, работы с этими данными.

К сожалению, многие разработчики не так часто задумываются о оптимизации SQL-запросов, напрасно полагая, что современное аппаратное обеспечение поможет компенсировать лишние затраты, возникающие при выполнении неэффективных SQL-запросов. Современные СУБД имеют встроенные средства оптимизации запросов, перед их непосредственным выполнением. Понимание принципов работы этих средств необходимо для написания эффективных запросов. Поэтому изучение данного вопроса является актуальным на сегодняшний момент. В данной работе будут рассмотрены встроенные в современные СУБД инструменты оптимизации работы с базой данных, будут описаны общие рекомендации по написанию эффективных SQL-запросов, будет разработан инструмент кеширования результатов запросов к базе данных, а также будет разработан инструмент для распределения нагрузки на базу данных в высоконагруженных информационных системах.

**Предмет исследования**

Оптимизация работы веб-сайта.

**Объект исследования**

Оптимизация взаимодействия веб-сайта с базой данных

**Цель работы**

Разработать инструмент для распределения нагрузки на базу данных в высоконагруженных системах.

**Основные задачи исследования**

1. Рассмотреть встроенные в современные СУБД инструменты оптимизации работы с базой данных;
2. Описать общие рекомендации по написанию эффективных SQL запросов;
3. Разработать инструмент для распределения нагрузки на базу данных в высоконагруженных системах;

**Структура работы**

Работа состоит из введения, двух глав, заключения и списка источников.

Во введении рассматривается актуальность работы, ставится цель и формулируются задачи, необходимые для достижения поставленной цели.

Первая глава представляет собой совокупность теоретических данных об изучаемом предмете исследования.

Во второй главе описывается практическое применение инструмента для оптимизации клиентской и серверной части веб-сайтов.

В заключении делаются выводы по проделанной работе.

В конце работы приводится список использованных источников

# ГЛАВА 1. ТЕОРИТИЧЕСКИЕ СВЕДЕНИЯ ОБ ОПТИМИЗАЦИИ РАБОТЫ С БАЗОЙ ДАННЫХ

## Общая архитектура реляционных СУБД

### Структура данных

### Кластеризация

### Индексирование

### Кэширование в базах данных

## Оптимизация запросов

### Синтаксическая оптимизация запросов

### Семантическая оптимизация запросов

### Генерация и выбор плана выполнения

### Практика написания эффективных SQL-запросов

# ГЛАВА 2. РАЗРАБОТКА ИНСТРУМЕНТА ДЛЯ РАСПРЕДЕЛЕНИЯ НАГРУЗКИ НА БАЗУ ДАННЫХ В ВЫСОКОНАГРУЖЕННЫХ ИНФОРМАЦИОННЫХ СИСТЕМАХ

### 2.2.1. Теоретические сведения об использовании брокера очередей RabbitMQ

### 2.2.2. Разработка инструмента для распределения нагрузки на базу данных с использованием брокера очередей RabbitMQ

# ЗАКЛЮЧЕНИЕ

# СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

# ПРИЛОЖЕНИЕ