

**Автономная некоммерческая организация профессионального образования  
«ПЕРМСКИЙ ГУМАНИТАРНО-ТЕХНОЛОГИЧЕСКИЙ КОЛЛЕДЖ»  
(АНО ПО «ПГТК»)**

УТВЕРЖДЕНА  
Педагогическим советом АНО ПО «ПГТК»  
(протокол от 05.02.2026 № 01)  
Председатель Педагогического совета, директор  
И.Ф. Никитина



**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ  
УЧЕБНОЙ ДИСЦИПЛИНЫ**

**МДК.02.01 УПРАВЛЕНИЕ И АВТОМАТИЗАЦИЯ БАЗ ДАННЫХ**

для специальности

**09.02.13 «Интеграция решений с применением технологий  
искусственного интеллекта»**  
(код и наименование специальности)

Квалификация выпускника

**Специалист по работе с искусственным интеллектом**

Форма обучения

**Очная**

Пермь 2026

Фонд оценочных средств учебной дисциплины МДК.02.01 УПРАВЛЕНИЕ И АВТОМАТИЗАЦИЯ БАЗ ДАННЫХ составлен в соответствии с требованиями Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.13 «Интеграция решений с применением технологий искусственного интеллекта» (утвержден приказом Министерства Просвещения Российской Федерации от 24 декабря 2024 г. N 1025).

Программа предназначена для студентов и преподавателей АНО ПО «ПГТК».

Автор – составитель: Могильникова Н.С., старший преподаватель.

Фонд оценочных средств учебной дисциплины рассмотрена и одобрена на заседании кафедры математических и естественно-научных дисциплин, протокол, № 01 от 04.02.2026.

## Содержание ФОС УД

1. Паспорт фонда оценочных средств
  - 1.1. Область применения фонда оценочных средств
  - 1.2. Организация текущего контроля успеваемости и промежуточной аттестации по итогам освоения учебной дисциплины
2. Контроль и оценка достижения запланированных результатов обучения
  - 2.1. Перечень вопросов и заданий для текущего контроля знаний
  - 2.2. Перечень вопросов и заданий для промежуточной аттестации
  - 2.3. Критерии оценивания ПА
3. Рекомендуемая литература и иные источники

## 1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

### 1.1. Область применения фонда оценочных средств

Фонд оценочных средств предназначен для оценивания достижений запланированных результатов по учебной дисциплины МДК.02.01 Управление и автоматизация баз данных программы подготовки специалистов среднего звена (далее ППССЗ) по специальности 09.02.13 «Интеграция решений с применением технологий искусственного интеллекта».

Фонд оценочных средств (ФОС) представляет собой комплект материалов для проведения промежуточной аттестации и текущего контроля.

Результаты обучения - это усвоенные знания и освоенные умения по дисциплине в целях овладения предусмотренных стандартом общих и профессиональных компетенций.

Фонд оценочных средств позволяет оценивать формирование элементов профессиональных компетенций (ПК) и элементов общих компетенций (ОК) через освоение умений, знаний и навыков.

Код ОК, ПК	Уметь	Знать
ПК 2.1 Выявлять проблемы, возникающие в процессе эксплуатации баз данных. ПК 2.2 Осуществлять процедуры администрирования баз данных. ПК 2.3 Проводить аудит систем безопасности баз данных с использованием регламентов по защите информации. ПК 2.4 Формировать требования хранения данных для обучения. ПК 2.5 Подготавливать данные для базы знаний. ОК 01 Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам	Производить идентификацию проблем, связанных с нормальным функционированием базы данных; Принимать решения по локализации проблем, связанных с нормальным функционированием базы данных; Документировать внештатные ситуации связанные с нормальным функционированием базы данных; Осуществлять основные функции по администрированию баз данных; Настраивать политики безопасности при работе с сервером баз данных Дать независимую оценку уровня безопасности Производить регламентное обновление программного обеспечения Разрабатывать перечень рекомендаций по дальнейшей эксплуатации БД с максимальной защитой хранящейся информации. Производить формирование требований к обработке данных и их извлечению;  Добавлять, удалять и изменять данные в базе данных; Производить операции по импорту и экспорту данных в различных форматах распознавать задачу и/или проблему в профессиональном и/или социальном контексте, анализировать и выделять её составные части	Основные коды ошибок при работе с базой данных; Методы и средства устранения ошибок, возникающих при работе с базой данных; Тенденции развития баз данных; Технология установки и настройки сервера баз данных; Требования к безопасности сервера базы данных;  Протоколы безопасности при работе с базой данных; Методы и средства защиты информации от несанкционированного доступа; Уровни угроз безопасности информации Формы документов, необходимых для формирования, ведения и использования баз данных Типы данных хранения информации в базе данных актуальный профессиональный и социальный контекст, в котором приходится работать и жить структура плана для решения задач, алгоритмы выполнения работ в профессиональной и смежных областях основные источники информации и ресурсы для решения задач и/или проблем в

<p>ОК 02 Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности</p> <p>ОК 05 Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста</p>	<p>определять этапы решения задачи, составлять план действия, реализовывать составленный план, определять необходимые ресурсы</p> <p>выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы</p> <p>владеть актуальными методами работы в профессиональной и смежных сферах</p> <p>оценивать результат и последствия своих действий (самостоятельно или с помощью наставника)</p> <p>определять задачи для поиска информации, планировать процесс поиска, выбирать необходимые источники информации</p> <p>выделять наиболее значимое в перечне информации, структурировать получаемую информацию, оформлять результаты поиска</p> <p>оценивать практическую значимость результатов поиска</p> <p>применять средства информационных технологий для решения профессиональных задач</p> <p>использовать современное программное обеспечение в профессиональной деятельности</p> <p>использовать различные цифровые средства для решения профессиональных задач</p> <p>грамотно излагать свои мысли и оформлять документы по профессиональной тематике на государственном языке</p> <p>проявлять толерантность в рабочем коллективе</p>	<p>профессиональном и/или социальном контексте</p> <p>методы работы в профессиональной и смежных сферах</p> <p>порядок оценки результатов решения задач профессиональной деятельности</p> <p>номенклатура информационных источников, применяемых в профессиональной деятельности</p> <p>приемы структурирования информации</p> <p>формат оформления результатов поиска информации</p> <p>современные средства и устройства информатизации, порядок их применения</p> <p>программное обеспечение в профессиональной деятельности, в том числе цифровые средства</p> <p>психологические основы деятельности коллектива</p> <p>правила оформления документов</p> <p>правила построения устных сообщений</p> <p>особенности социального и культурного контекста</p>
---	---	---

## **1.2. Организация текущего контроля успеваемости и промежуточной аттестации по итогам освоения программы учебной дисциплины**

В период обучения по образовательной программе СПО осуществляется текущий контроль успеваемости студентов, промежуточная аттестация по учебным дисциплинам и профессиональным модулям.

Текущий контроль осуществляется в пределах учебного времени, отведенного на учебную дисциплину, оценивается по пятибалльной шкале. Текущий контроль проводится с целью объективной оценки качества освоения программы дисциплины, а также стимулирования учебной деятельности студентов, подготовки к промежуточной аттестации и обеспечения максимальной эффективности учебного процесса. Для оценки качества подготовки используются различные формы и методы контроля. Текущий контроль учебной

дисциплины осуществляется в форме устного опроса; защиты практических заданий, реферата, творческих работ; выполнения контрольных и тестовых заданий; решения ситуационных задач и других форм контроля, предусмотренных программой учебной дисциплины.

Промежуточная аттестация проводится в форме, предусмотренной планом учебного процесса: дифференцированного зачета, экзамена.

В период сложной санитарно-эпидемиологической обстановки или других ситуациях невозможности очного обучения и проведения аттестации студентов колледж реализует образовательные программы или их части с применением электронного обучения, дистанционных образовательных технологий в предусмотренных законодательством формах обучения или при их сочетании, при проведении учебных занятий, практик, текущего контроля успеваемости, промежуточной аттестации обучающихся.

Форма промежуточной аттестации по учебной дисциплине МДК.02.01 Управление и автоматизация баз данных - дифференцированный зачет.

## **2. КОНТРОЛЬ И ОЦЕНКА ОСВОЕНИЯ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ**

### **2.1. Перечень вопросов и заданий для текущего контроля**

В результате текущей аттестации по учебной дисциплине МДК.02.01 Управление и автоматизация баз данных осуществляется проверка сформированности умений и знаний, направленных на формирование соответствующих ФГОС СПО общих и профессиональных компетенций.

#### **Практическое занятие «Построение схемы базы данных»**

Цель работы: Сформировать умения добавлять таблицы в базу данных с целью расширения ее функциональных возможностей.

Иногда в процессе разработки базы данных или в процессе опытной эксплуатации ее возникает необходимость добавления в нее новых таблиц. Очевидно, что спроектированная нами в предыдущей работе база данных Библиотека обладает очень ограниченными возможностями. Эта база данных, состоящая из трех таблиц: Издательства, Книги и Темы, не позволяет автоматизировать работу с читателями. В ней отсутствует информация о читателях.

В данной работе мы научимся добавлять таблицы в базу данных с целью расширения ее функциональных возможностей. Создание новых таблиц осуществляется точно так же, как это мы делали в предыдущей работе. Для добавления таблиц в ранее созданную схему данных и установления связи между таблицами используется кнопка Отобразить таблицу, размещенная на панели инструментов Связь.

#### **Задание**

1. Откройте базу данных Библиотека. Создайте в ней структуру таблицы Читатели, которая будет содержать следующие поля: Код читателя, Фамилию, Имя, Отчество, Домашний телефон, Домашний адрес. Типы данных для полей таблицы, их свойства определите самостоятельно по смыслу. В качестве ключа укажите поле Код читателя.

2. Аналогичным способом создайте структуру таблицы Выдача книг. В эту структуру включите три поля: Код читателя, Код книги, Дата заказа. В этой таблице ключевое поле не задавайте. Для поля Дата заказа укажите тип данных – Дата/время. Обратите внимание на то, что в последствии ключ Код читателя в таблице Читатели будет связываться с полем Код читателя в таблице Выдача книг. Поэтому эти поля должны иметь соответствующие типы данных и свойства.

3. Установите между добавленными таблицами: Читатели и Выдача книг, а также ранее созданными таблицами: Издательства, Книги и Темы, связи так, как это показано в окне Схема данных на рис. 1.

Напомним, что для установления связи между таблицами надо открыть окно Схема данных. При его

открытии появляется диалоговое окно Добавление таблицы, в котором надо выделить имена тех таблиц, между которыми будут устанавливаться связи. После этого нажимают кнопки Добавить и Заккрыть. Затем в окне Схема данных с помощью мыши перетаскивают ключевое поле одной таблицы на соответствующее поле в другой таблице. В появившемся окне Связи задают режим Обеспечение целостности данных и его подрежимы: каскадное обновление связанных полей и каскадное удаление связанных записей и нажимают кнопку Создать.

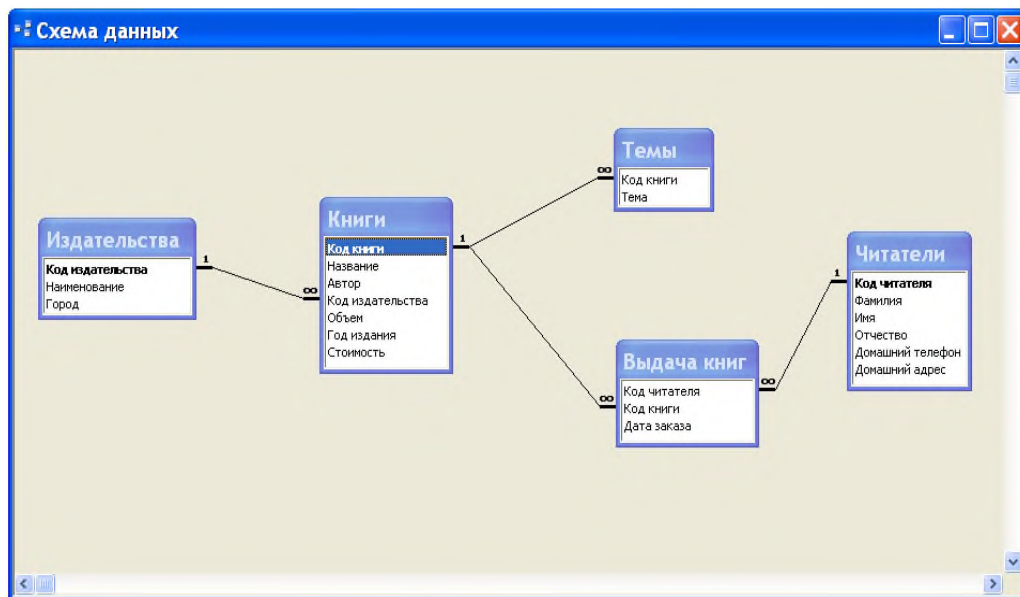


Рис. 1. Схема расширенной базы данных Библиотека.

4. Откройте таблицу Читатели и введите в нее данные, приведенные в таблице 1.

Таблица 1 Данные для ввода в таблицу Читатели

Код читателя	Фамилия	Имя	Отчество	Домашний телефон	Домашний адрес
1	Аксенов	Виктор	Сергеевич	252-88-13	ул. Есенина, 15-19
2	Голубева	Елена	Андреевна	220-99-29	ул. Чкалова, 7-38
3	Васильев	Игорь	Петрович	232-64-78	ул. Богдановича, 102-34
4	Кучеров	Валентин	Степанович	266-24-95	ул. Кнорина, 27-5
5	Мастяница	Вячеслав	Иванович	246-42-25	ул. Плеханова, 34-98
6	Победимская	Лариса	Анатолевна		ул. Чкалова, 9-10
7	Литвин	Борис	Николаевич	239-55-76	пр. Независимости, 46-54
8	Германович	Рита	Мироновна	278-31-51	ул. Казинца, 26-9
9	Бинцаровский	Теодор	Петрович		ул. Корженевская, 1-288

5. Введите в таблицу Выдача книг данные, приведенные в таблице 2.

Таблица 2 Данные для ввода в таблицу Выдача книг

Код читателя	Код книги	Дата заказа	Код читателя	Код книги	Дата заказа
1	1	1.09.07	4	3	7.01.08



1	3	5.07.08	4	4	25.10.07
1	4	21.10.07	5	2	23.04.08
2	1	4.11.07	6	1	18.06.08
3	2	3.08.08	7	3	20.01.08
8	7	25.12.07	9	6	2.02.08

Обратите внимание на то, что, если бы вы попробовали вначале ввести данные в таблицу Выдача книг, а затем в таблицу Читатели, то MS Access это не позволил бы сделать. Поэтому мы специально раньше установили связи между таблицами, а затем уже вводили данные в таблицы. В этом случае MS Access будет проверять целостность данных.

### Практическое занятие «Составление словаря данных»

Цель работы. Изучение средства создания словаря данных Access.

Теоретическая часть

Определив отдельные элементы данных, составляющие таблицы базы данных, и установив отношения между ними, необходимо подготовить описание базы данных, называемое словарем данных. Словари данных имеют огромное значение для баз данных. Недокументированной базой данных тяжело управлять и ее трудно обслуживать. Ошибки и промахи, допущенные на этапе ее разработки, во многих случаях можно обнаружить при подготовке предварительного словаря данных.

Завершив разработку структуры базы данных и убедившись в ее правильности, требуется довести до конца разработку подробного словаря данных. По мере добавления в приложение новых или модификации существующих форм и отчетов, необходимо соответствующим образом обновлять и словарь. Даже при создании базы данных для личного использования упрощенная версия словаря данных позволит получить большую пользу при малых затратах времени.

Стандартный словарь данных

Словарь данных содержит информацию о базе данных в целом, обо всех таблицах, полях, включенных в таблицы, о первичных и внешних ключах, а также включает толкование используемых кодов. В словаре также приводится назначение и описание каждого приложения, использующего базу.

Поскольку словари данных имеют иерархическую структуру, они могут храниться в традиционном формате, поддерживаемом текстовыми редакторами Windows. Ниже приводится структура стандартного словаря данных, в котором заданы традиционные заголовки:

1. БАЗА ДАННЫХ—Полное название базы и имя файла.

Описание назначения и общего содержания базы данных, а также лиц, которые могут ею пользоваться. Список приложений, работающих с базой, и информация о других базах данных, использующих данные из данной базы. Если имеются, то сюда же включаются диаграммы базы данных.

А. ОБЛАСТЬ ДАННЫХ — Название группы, к которой принадлежат таблицы.

Если таблицы классифицируются по группам, например, финансовая группа, то включается описание каждой группы.

1. ТАБЛИЦА — Таблицы, входящие в область данных.

- a) ДОСТУП — Права пользователей на доступ к таблице.
- b) ЗАПИСЬ — Общее определение элементов данных.
  - (1) ПЕРВИЧНЫЙ КЛЮЧ - Поле (поля) первичного ключа.
    - (a) ИНДЕКС — Описание индекса первичного ключа.
  - (2) ВНЕШНИЕ КЛЮЧИ - Внешние ключевые поля.
    - (a) ИНДЕКС - Индексы . внешних ключей.
  - (3) ПОЛЯ — Неключевые поля.
    - (a) ПЕРЕЧИСЛИМЫЕ МНОЖЕСТВА — Допустимые коды для полей.

За каждым заголовком идет текст, описывающий назначение элемента, базы данных, к которому относится заголовок. Последующие разделы словаря включают описания объектов, которые используют таблицы базы данных, с подзаголовками для запросов, форм и отчетов. Изображения, снятые с экрана, и копии отчетов также добавляются в словарь данных. Распечатки кода программ обычно приводятся в приложениях словаря. Подробные словари данных необходимы для обслуживания базы данных. Кроме того, описание словаря можно представить в табличной форме.

### Интегрированный словарь данных

Надстройка Архивариус (Documentor), которая впервые появилась в Access 2.0., позволяет создать отчет, где подробно описываются объекты и значения их свойств для текущей базы данных.

Средство Access 97 "Публикация в MS Word", запускаемое с помощью кнопки на панели инструментов, позволяет сохранить отчет в формате .RTF. Затем созданный файл можно импортировать в Microsoft Word или любой другой текстовый процессор, который обрабатывает файлы в формате RTF, например, WordPad. Кроме того, можно экспортировать отчет в формате BIFF, нажав на панели инструментов кнопку "Анализ в MS Excel".

Во многих случаях Архивариус (Documentor) сообщит вам больше, чем вы хотите знать о вашей базе данных; например, полный отчет обо всех объектах базы данных Борея (Northwind) составляет около 400 страниц. Однако чаще требуется поместить в словарь данных только информацию о таблицах и, возможно, запросах.

### Задание на выполнение работы

Для создания словаря данных с помощью Архивариуса (Documentor):

1. Откройте требуемую базу данных и выберите команду "Сервис, Анализ, Архивариус" (Tools, Analyze, Documentor).
2. Выберите вкладку с требуемым типом объектов, которые необходимо описать. Если выбрать вкладку "Все типы объектов" (All Object Types) и нажать кнопку "Выбрать все" (Select All), то можно создать описание для всех объектов базы данных.
3. Нажмите кнопку "Параметры" (Options) для вывода диалогового окна "Печать описания таблицы" (Print Table Definition). По умолчанию печатается наиболее подробное описание таблиц и индексов. Если база данных не имеет системы защиты, то можно сбросить флажок "Разрешения для пользователей и групп" (Permissions By User and Group), чтобы не выводить в отчете данные о правах доступа. Нажмите кнопку ОК для возврата в первое диалоговое окно Архивариуса (Documentor).

4. Выберите требуемые таблицы, нажимая кнопку "Выделить" (Select). При этом будет установлен флажок напротив имени указанной таблицы. Кроме того, можно выделить объект, дважды щелкнув по его названию. Если необходимо документировать все таблицы базы данных, нажмите кнопку "Выбрать все" (Select All).

5. Выберите вкладку "Запросы" (Queries) для вывода запросов базы данных. Нажмите кнопку "Параметры" (Options) для вывода диалогового окна "Печать описания запроса" (Print Table Definition). Сбросьте флажок "Разрешения для пользователей и групп" (Permissions By Group and Group), чтобы не выводить в отчете данные о правах доступа, и задайте нужные параметры, чтобы не выводить данные, дублирующие информацию о полях таблиц. Нажмите кнопку "ОК".

6. Выберите запрос "Сведения о заказах", дважды щелкнув по его имени, и нажмите кнопку ОК для создания отчета длиной 13 страниц.

7. Вскоре появляется окно предварительного просмотра "Описание объекта" (Object Definition). Обратите внимание на то, что отчет о трех объектах занимает одну страницу.

8. Нажмите на панели инструментов кнопку "Печать" (Print) для печати отчета либо щелкните по кнопке "Связи с Office" (Office Links) для создания файла в формате .RTF или .XLS

9. Нажмите на панели инструментов кнопку "Закрыть" (Close) либо дважды щелкните по кнопке системного меню для закрытия окна предварительного просмотра.

Документирование других объектов базы данных выполняется аналогичным образом. Можно вывести данные о самой базе данных, выбрав вкладку "Текущая база данных" (Current Database), или распечатать данные из выбранных форм, отчетов, макросов и модулей.

### **Практические занятия «Разработка технических требований к серверу баз данных», «Разработка требований к корпоративной сети».**

Цель: научиться разрабатывать технические требования к серверу баз данных, к корпоративным сетям.

Задание.

Проектирование (модернизация) корпоративной сети является составной частью разработки (модернизации) распределенной информационной системы управления (РИСУ) предприятия. Проектирование РИСУ включает:

- проектирование объектов данных, которые будут реализованы в базе данных;
- проектирование программ, экранных форм, отчетов, которые будут обеспечивать выполнение запросов к данным;
- проектирование конкретной сетевой среды или технологии, а именно: топологии сети, конфигурации аппаратных средств, используемой архитектуры (файл-сервер или клиент-сервер), параллельной обработки, распределенной обработки данных и т.п.

Параметры (показатели) корпоративной сети должны соответствовать целям проекта РИСУ. В разделе 1 было показано, что основное требование к корпоративной сети – это обеспечение всем

категориям пользователей доступа к разделяемым ресурсам сети с заданным качеством обслуживания (QoS – Quality of Service), причем качество обслуживания характеризуется следующими показателями:

- требуемая пропускная способность системы  $\nu$  [Мбит/с];
- требуемое время реакции системы на запрос  $T_{don}$  [с];
- безотказность работы системы;
- требуемый уровень информационной безопасности.

Одним из основных критериев качества обслуживания является производительность, причем в качестве показателей производительности используются время реакции, пропускная способность  $\nu$  (бит/с, пакетов/с) и задержка передачи.

Информационная безопасность – это защищенность сетевых ресурсов от несанкционированного доступа. В качестве показателей надежности используются: среднее время наработки на отказ  $T_{отк}$ , среднее время ремонта  $T_{рем}$  и коэффициент готовности  $K_g = T_{отк} / (T_{отк} + T_{рем})$ , определяющий вероятность работоспособного состояния сети в любой момент времени. Важным требованием к надежности вычислительных сетей является также отказоустойчивость, т.е. сохранение работоспособности при отказе отдельных элементов.

Ряд требований к компьютерным сетям связан с их эксплуатацией и развитием, а также с обеспечением удобства работы для пользователей.

Совместимость сетевого оборудования и программного обеспечения позволяет объединять разнообразные компоненты, приобретенные от разных производителей.

Расширяемость – это возможность расширения сети (добавления отдельных элементов, наращивания длины сегментов, замены оборудования на более мощное) без особых проблем. Масштабируемость – это возможность расширения сети в широких пределах без снижения производительности (с получением прогнозируемого эффекта).

Важным требованием, характеризующим удобство работы пользователей, является прозрачность доступа к сетевым ресурсам. Прозрачность означает, что при работе в сети пользователю не требуется знать детали устройства системы.

Требуемые пропускная способность и время реакции системы зависят от информационной нагрузки на сеть, которую создают пользователи. Для расчета требуемой пропускной способности  $\nu$  будем использовать аппарат теории систем массового обслуживания (СМО) и сетей массового обслуживания (СеМО) [13]. СеМО – это совокупность систем массового обслуживания, в которой заявки с выходов одних СМО могут поступать на входы других.

Например, в соответствии с теорией СеМО основным показателем информационной нагрузки на сервер сети является коэффициент загрузки сервера  $\rho = \lambda / \mu$ , где  $\lambda$  – интенсивность информационного потока, создаваемого пользователями, между рабочими станциями и сервером;  $\mu$  – интенсивность обслуживания сервера. В качестве единиц измерения  $\lambda$  и  $\mu$  используют число бит, пакетов или запросов в секунду [бит/с, пакетов/с, запросов/с].

При моделировании РИСУ в виде СеМО приходится исследовать системы, включающие в качестве подсистем многие системы массового обслуживания. Наиболее развита аналитическая теория экспоненциальных СеМО и точные методы расчета вероятностно-временных характеристик сложных систем. При моделировании РИСУ используют следующие разновидности моделей РИСУ: разомкнутая СеМО (РСеМО), замкнутая СеМО (ЗСеМО) и комбинированная СеМО. Для РСеМО характерно наличие одного или нескольких независимых внешних источников, которые генерируют заявки, поступающие в сеть, независимо от того, сколько заявок уже находится в сети, причем в любой момент времени в РСеМО может находиться произвольное число заявок (от 0 до  $\infty$ ). В ЗСеМО циркулирует фиксированное число заявок, а внешние независимые источники отсутствуют. В комбинированной СеМО постоянно циркулирует определенное число заявок, а также есть заявки, поступающие от внешних независимых источников.

Существует два основных режима работы сети: пакетный и диалоговый. Для диалогового режима важным параметром является среднее время обдумывания пользователя  $1/\eta$  [с] на один запрос к системе, где  $\eta$  – интенсивность обдумывания. Методика приближенного расчета пропускной способности магистрали сети приведена в разделе 1.

Для приложений с пакетным режимом работы величина  $1/\eta$  приблизительно равна среднему интервалу между моментами появления запросов к соответствующей сетевой службе. Интенсивность потока требований, генерируемых одним пользователем, можно вычислить по формуле  $\lambda_1 \approx 1/(1/\eta + T_{don})$ , где  $T_{don}$  – допустимое время реакции системы на запрос пользователя, например:  $1/\eta = 120$  с,  $T_{don} = 60$  с,  $\lambda_1 \approx 1/(120 + 60) = 0,0056$  запр/с.

## 8.2. Расчет параметров сети

### 8.2.1. Элементы теории СеМО для расчета параметров сети

Для оптимизации производительности сети ЭВМ используют методы и средства измерения, анализа и моделирования. Клиент-серверная архитектура и распределенная обработка данных в сети усложняют задачи моделирования.

Аналитическое моделирование основано на использовании моделей СМО и СеМО [13] и, как правило, связано со значительными упрощениями. Тем не менее результаты аналитического исследования могут быть очень ценными, даже если они не учитывают всех деталей реальной сети. Такие модели позволяют достаточно быстро получить приближенную инженерную оценку влияния характеристик оборудования и программного обеспечения на показатели производительности сети.

Модель сети строится из отдельных блоков, каждый из которых представляет один узел или канал передачи сети. Блок состоит из буферного накопителя заявок и обслуживающего элемента (рис.8.1). При анализе сетей ЭВМ в качестве заявок могут фигурировать запросы пользователей или отдельные пакеты, на которые разбиваются эти запросы. На вход блока поступает поток заявок, характеризуемый функцией распределения интервалов времени между моментами поступления заявок  $A(t)$ .

Интенсивность  $\lambda$  входного потока заявок – это среднее число заявок, поступающих на вход блока в единицу времени. Обратная величина  $1/\lambda$  – это среднее значение интервала между моментами поступления заявок, которое определяется интегралом  $\int_0^{\infty} t dA(t)$ .

Интенсивность обслуживания блока – это  $\mu$  среднее число обрабатываемых заявок в единицу времени. Обратная величина  $1/\mu$  – это среднее значение длительности обслуживания заявки, которое определяется интегралом  $\int_0^{\infty} t dB(t)$ , где  $B(t)$  – функция распределения длительности обслуживания. Отношение  $\rho = \lambda/\mu$  называется коэффициентом загрузки блока. Реальный блок имеет буфер ограниченной емкости  $r$  (см. рис.8.1,б). Идеализированный модуль может иметь неограниченный по емкости буфер (см. рис.8.1,а).



Рис. 8.1.

Блок М/М/1. Рассмотрим самую простую модель типа М/М/1 (один обслуживающий элемент, неограниченная емкость буфера, экспоненциальные законы распределения интервалов времени между моментами поступления заявок и времени обслуживания, дисциплина обслуживания FIFO) для блока, изображенного на рис.8.1, а. В этом случае  $A(t) = 1 - e^{-\lambda t}$ ,  $B(t) = 1 - e^{-\mu t}$ , среднее время задержки заявки в блоке

$$T = 1 / (\mu - \lambda), \quad (8.1)$$

а среднее число заявок в блоке (в очереди и в процессе передачи)

$$L = \lambda / (\mu - \lambda). \quad (8.2)$$

Среднее время ожидания в очереди  $W = T - (1/\mu) = \rho / (\mu - \lambda)$ , а среднее число заявок в очереди  $L_w = L - \rho = \rho\lambda / (\mu - \lambda)$ .

Блок M/G/1. Эта модель отличается от модели типа M/M/1 только тем, что распределение времени обслуживания  $B(t)$  может быть произвольным. Рассмотрим случай, когда распределение  $B(t)$  задается для блока двумя параметрами: интенсивностью обслуживания  $\mu$  и дисперсией времени обслуживания

$$D = \int_0^\infty t^2 dB(T) - 1/\mu^2.$$

Тогда среднее время нахождения заявки в очереди  $W = (1 + v^2) W^\Pi$ , где  $W^\Pi = (\rho/2\mu) \times (1-\rho)^{-1}$  – время нахождения заявки в очереди при постоянной длительности обслуживания;  $v^2 = \mu^2 D$  – квадрат коэффициента вариации времени обслуживания. Для постоянного времени обслуживания  $v=0$ , а для экспоненциального распределения времени обслуживания  $v=1$ . Для модели M/G/1 оценка времени пребывания заявки в блоке  $T = W + (1/\mu)$ , длины очереди в буфере  $L_w = \lambda W$  и общего числа заявок в блоке  $L = L_w + \rho$ .

Блоки M/M/1/r и M/G/1/r. Модель типа M/G/1/r для блока, изображенного на рис.8.1,б, отличается от модели M/G/1 тем, что емкость буфера ограничена величиной  $r$  (предполагается, что обрабатываемая заявка находится также в буфере). Эта модель характеризуется вероятностью потери заявки (отказа в обслуживании) [11]

$$P_{\text{отк}} \approx (1-\rho) \rho^{\Psi(r,v)} / (1-\rho^{\Psi(r,v)+1}), \quad (8.3)$$

где  $\Psi(r,v) = 2r/(1+v^2)$ , причем  $v$  – коэффициент вариации. Абсолютная пропускная способность блока M/G/1/r

$$\lambda_{\text{АБС}} = \lambda (1 - P_{\text{отк}}).$$

При  $v=1$  формула дает точное значение  $P_{\text{отк}}$  для экспоненциального распределения  $B(t)$ , т.е. для блоков M/M/1/r.

Сеть блоков M/M/1. Модель сети можно представить в виде СеМО, т.е. сети блоков [3]), содержащих буферы. Простые аналитические формулы можно получить для открытой сети блоков M/M/1, пример которой представлен на рис.8.2.

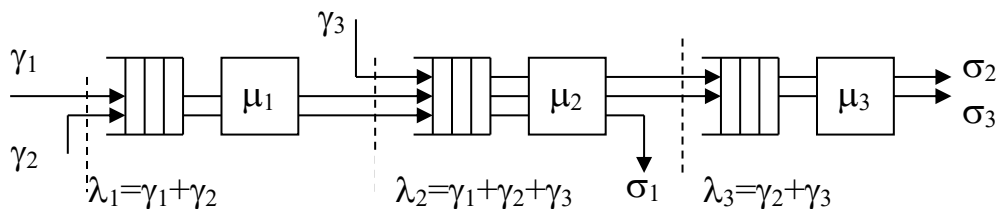


Рис. 8.2.

В этой сети, состоящей из трех блоков, три входных потока пакетов, имеющих интенсивности  $\gamma_1$ ,  $\gamma_2$  и  $\gamma_3$  соответственно. Требуется оценить среднюю задержку пакетов для каждого потока. Потoki, выходящие

из блоков, не ветвятся, причем выходные потоки  $\sigma_1=\gamma_1$ ,  $\sigma_2=\gamma_2$  и  $\sigma_3=\gamma_3$ , т.е. соблюдается баланс

$$\gamma_1+\gamma_2+\gamma_3=\sigma_1+\sigma_2+\sigma_3.$$

Очереди в этой сети можно рассматривать по отдельности [3], причем число пакетов в блоке  $j=1\dots 3$  оценивается по формуле (8.2), а именно

$$L_j = \lambda_j / (\mu_j - \lambda_j).$$

Интенсивность  $\lambda_j$  потока на входе каждого блока равна сумме интенсивностей элементарных потоков, поступающих на блок в соответствии с рис.3:  $\lambda_1 = \gamma_1 + \gamma_2$ ,  $\lambda_2 = \gamma_1 + \gamma_2 + \gamma_3$ ,  $\lambda_3 = \gamma_2 + \gamma_3$ .

Можно показать [3], что средняя задержка пакета в сети

$$T = \frac{1}{\gamma} \sum_{j=1}^{j=n} \frac{\lambda_j}{\mu_j - \lambda_j}, \quad (8.4)$$

где  $n$  – число блоков в системе;  $\gamma$  – сумма интенсивностей всех потоков, входящих в систему.

Формула (8.4) верна при следующих предположениях.

- Закон распределения интервалов времени между моментами поступления пакетов  $A(t)$  для отдельных потоков экспоненциальный, причем потоки являются независимыми процессами. Это предположение может быть выполнено на практике.
- Закон распределения времени обслуживания  $B(t)$  также экспоненциальный, причем процессы обслуживания в каждой очереди независимы. Это предположение не может быть выполнено, поскольку время обслуживания пакета пропорционально его длине, и, следовательно, нельзя говорить о независимости времен обслуживания в очередях.

Однако моделирование показывает [3], что применение формулы (8.4) дает приемлемую оценку средней задержки пакета в сети.

Имитационное моделирование позволяет имитировать поведение реальной сети ЭВМ. Имеется много программных средств для имитационного моделирования компьютерных сетей (GPSS, COMNET III фирмы Caci Products Co., BOnES Designer фирмы Cadence Inc., OPNET фирмы Modeler Mil3 Inc., ns2 и др.).

### 8.2.2. Расчет параметров сети в пакетном режиме

Для расчета параметров сети для пакетного режима обслуживания запросов используется теория РСеМО [13]. Предполагаем, что в результате анализа бизнес-процессов предприятия выявлен состав пользователей сети и состав приложений (сервисов). Для того чтобы обеспечить требуемое качество обслуживания (QoS) для различных категорий пользователей, распределим множество  $H$  всех пользователей по типам. Будем считать, что пользователи типа  $t \in T$ , где  $T$  – множество типов, характеризуются следующими параметрами:

$Q_{ts}$  [Кбайт] – средний объем информации, который необходимо передать по сети для работы пользователя типа  $t$  с сервисом  $s \in S$ , где  $S$  – множество сервисов, на интервале  $\Delta t$  [час];



$L_{ts}$  [Кбайт] – средний объем транзакции пользователя типа  $t$  по сервису  $s$ ;

$\tau_{ts.\partial on}$  [с] – допустимое время реакции сети для пользователя типа  $t$  по сервису  $s \in S$ .

Предполагаем, что на одном сервере (компьютере) может быть реализован один или несколько сервисов. Все пользователи распределены по рабочим группам, причем пользователи группы размещаются достаточно компактно: в одном или смежных помещениях размещается множество пользователей  $H_g = \bigcup_{t \in T} H_{tg}$  группы  $g \in G$ , где  $G$  – множество рабочих групп,  $H_{tg}$  – множество пользователей типа  $t$  в группе  $g$ .

Для того чтобы обеспечить требуемое качество обслуживания пользователей, необходимо подобрать параметры оборудования сети таким образом, чтобы они соответствовали информационной нагрузке всех типов пользователей всех рабочих групп.

Пусть  $E$  – множество элементов сетевого оборудования и  $E_{gs} \subset E$  – подмножество элементов, участвующих в передаче данных между сервисом  $s \in S$  и пользователями из группы  $g \in G$ .

Тогда, интенсивность трафика пользователя типа  $t$  по сервису  $s$  на интервале  $\Delta t$

$$\lambda_{ts} = \frac{8 * 10^3 * Q_{ts}}{3600 * \Delta t} [\text{бум} / \text{с}]. \quad (8.5)$$

Интенсивность трафика через элемент  $e$ :

$$\lambda_e = \sum_{g \in G} \sum_{t \in T} \sum_{h \in H_{tg}} \sum_{s \in S} x_{egs} \lambda_{ts}, \quad (8.6)$$

где

$$x_{egs} = \begin{cases} 1, & \text{если } e \in E_{gs}; \\ 0 & \text{в противном случае} \end{cases}$$

Интенсивность трафика сервиса  $s \in S$ :

$$\lambda_s = \sum_{g \in G} \sum_{t \in T} \sum_{h \in H_{tg}} \lambda_{ts}, \quad (8.7)$$

причем интенсивность обслуживания сервера  $\mu_s = \rho_s \lambda_s$ , где  $\rho_s$  – коэффициент загрузки сервера.

### 8.2.3. Расчет производительности серверов

В предыдущем разделе найдена интенсивность обслуживания сервера  $\mu_s$  [бит/с]. Требуется связать значение  $\mu_s$  с такими параметрами сервера, как  $\nu$  – производительность процессора сервера [оп/с],  $V_D$  – скорость передачи на НМД сервера [Кбайт/с] и  $t_D$  – время доступа к НМД сервера [с].

Среднее время обработки пакета процессором сервера можно определить по следующей формуле:

$$1/\mu_{\text{пак}} = \frac{q \cdot K_p}{\nu}, \quad (8.8)$$

где  $q$  – длина пакета в сети (байт);  $K_p$  – коэффициент расширения (число операций, приходящихся на 1 байт информации, берется по данным таблицы 8.1).

Таблица 8.1. Число операций на 1 байт информации (коэффициент пересчета)

Тип задач	Коэффициент пересчета
Учетные	$200 \div 500$
Планирования	$1000 \div 5000$
Прогнозирования	$10000 \div 20000$

Время обработки пакета на магнитном диске сервера складывается из двух составляющих: времени доступа и времени передачи, т.е. может быть определено из выражения:

$$1/\mu_D = \frac{q}{V_D} + t_D \quad (8.9)$$

где  $V_D$  – скорость передачи НМД,  $t_D$  – время доступа к НМД.

На рис. 8.4 изображена эквивалентная схема сервера, рассматриваемого как СеМО. На этом рисунке  $p$  – вероятность обращения к НМД при обработке входящих пакетов,  $\mu_{\text{пак}}$  – интенсивность обработки пакетов процессором сервера,  $\mu_D$  – интенсивность обработки пакета на магнитном диске.

Время произвольного доступа (random access time) – от 3 до 15 мс. Скорость передачи данных для внутренней зоны диска от 44,2 до 74,5 Мбайт/с и от 60,0 до 111,4 Мбайт/с для внешней зоны.

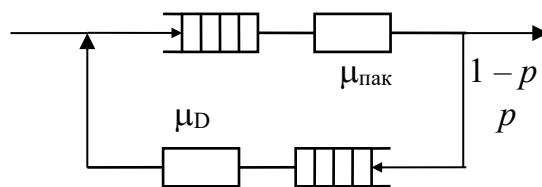


Рис. 8.4. Эквивалентная схема сервера

Схему на рис. 8.4 можно заменить одной СМО с интенсивностью обслуживания  $\mu_p = \mu_s / q$  [пак/с], определяемой формулой:

$$\frac{1}{\mu_p} = \frac{1}{\mu_{пак}} + p \frac{1}{\mu_D} \quad (8.10)$$

Уравнение (8.10) получено на основе уравнений баланса с учетом простых свойств слияния и разветвления потоков [10]: при слиянии  $n$  потоков заявок с интенсивностями  $\lambda_1, \dots, \lambda_n$  образуется поток, имеющий интенсивность  $\lambda = \lambda_1 + \dots + \lambda_n$ . При ветвлении потока с интенсивностью  $\lambda$  на  $n$  направлений, вероятности перехода заявки в которые равны  $p_1, \dots, p_n$ , образуется  $n$  потоков с интенсивностями  $\lambda p_1, \dots, \lambda p_n$  соответственно.

Подставим  $1/\mu_{пак}$  из формулы (8.8) в (8.10) и выразим требуемую производительность процессора сервера  $V$  через остальные параметры:

$$V = \frac{K_p q}{(1-p)/\mu_p - p(q/V_D + t_D)} \quad (8.11)$$

### 8.3. Обеспечение информационной безопасности

Безопасная система – это система, которая, во-первых, надежно хранит информацию и всегда готова предоставить ее своим пользователям, а во-вторых, система, которая защищает эти данные от несанкционированного доступа (<http://www.citfotum.ru/nets/spsmp>).

При проектировании корпоративной сети следует рассмотреть расширенный набор технических средств обеспечения информационной безопасности, например,:

1. Системы контроля доступа, включающие средства аутентификации и авторизации пользователей.
2. Средства аудита.
3. Системы шифрования информации.
4. Системы цифровой подписи, используемые для аутентификации документов.
5. Средства доказательства целостности документов (использующие, например, дайджест-функции).
6. Системы антивирусной защиты
7. Пакетная фильтрация в маршрутизаторах.
8. Пакетная фильтрация в межсетевых экранах.

## 9. Сервисы-посредники (proxyservices).

Из расширенного набора следует отобрать оптимальный состав средств обеспечения информационной безопасности для конкретной проектируемой системы.

## 8.4. Применение стандартов при проектировании корпоративных сетей

Стандартизация играет ключевую роль в создании и внедрении инфо-телекоммуникационных систем различных уровней и назначений. Основы стандартизации в области вычислительных сетей и средств телекоммуникаций были рассмотрены в первом разделе. Современные информационные системы (ИС) основаны на интеграции информационных, вычислительных и телекоммуникационных ресурсов. Эффективная интеграция современных информационных систем (ИС) – это сложная, комплексная межотраслевая проблема, которая решается на основе методов функциональной стандартизации и применении технологии открытых ИС. На таких ИС базируется информатизация всех сфер современного общества России: органов государственного управления, финансово-кредитной сферы, информационного обслуживания предпринимательской деятельности, производства сельского хозяйства, науки, медицины, образования и т. д.

Развитие и распространение технологии открытых ИС неразрывно связаны с применением функциональных стандартов. Методы функциональной стандартизации позволяют идентифицировать профили, т.е. группы базовых стандартов для выполнения функций, реализуемых конкретными ИС в разных предметных областях деятельности.

Профиль – это совокупность нескольких базовых стандартов (или подмножество одного базового стандарта) с четко определенными и гармонизированными (согласованными) подмножествами обязательных и факультативных возможностей, предназначенная для реализации заданной функции или группы функций ИС.

Технология открытых систем является основой инфраструктуры всех уровней – от уровня предприятия до уровня отрасли и национальной информационной инфраструктуры. Эта технология обеспечивает интеграцию с мировым информационным пространством и с мировой экономикой. Суть технологии открытых систем состоит в использовании стандартных интерфейсов между разнородными аппаратными и программными компонентами систем, в стандартизации и сертификации ИТ.

Существуют следующие виды стандартов в области ИТ: международные стандарты; национальные стандарты; стандарты специальных объединений и комитетов и стандарты

отдельных фирм. В Российской Федерации действует около 500 стандартов области ИТ, которые разделяются на межгосударственные (ГОСТ) и государственные стандарты (ГОСТ Р).

Стандарты на кабельные системы зданий. При разработке локальных (ЛВС) и корпоративных вычислительных сетей (КВС) необходимо ориентироваться, прежде всего, на международные стандарты на проектирование кабельных систем поскольку в настоящее время не существует российских стандартов на структурированные кабельные системы зданий. Существует три группы международных стандартов на кабельные системы зданий: американские, международные и европейские. Поскольку за основу были взяты американские требования, эти стандарты различия, в основном, в терминологии.

Отметим четыре основных стандарта: TIA/EIA 568A – стандарт телекоммуникационных кабельных систем коммерческих зданий; TIA/EIA 569 – стандарты прокладки телекоммуникационных каналов коммерческих зданий; TIA/EIA 606 – стандарт администрирования телекоммуникационной инфраструктуры коммерческих зданий; TIA/EIA 607 – требования по заземлению и электрическим соединениям телекоммуникационных систем коммерческих зданий.

### **Практическое занятие «Конфигурирование сети»**

Цель занятия: изучение инструментов конфигурирования сети в UNIX, включающих настройку параметров TCP/IP-сети.

*Начальные условия:* Командная строка суперпользователя после входа в систему.

Получить сведения обо всех настроенных сетевых интерфейсах с помощью команды `ifconfig -a`:

```
desktop ~ # ifconfig -a
eth0  Link encap:Ethernet  HWaddr 00:0D:60:8D:42:AA
      inet addr:192.168.1.5  Bcast:192.168.1.255  Mask:255.255.255.0
      UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
      RX packets:204779 errors:0 dropped:0 overruns:0 frame:0
      TX packets:107606 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:302429520 (288.4 Mb)  TX bytes:9177476 (8.7 Mb)
      Base address:0x8000 Memory:c0220000-c0240000

lo    Link encap:Local Loopback
      inet addr:127.0.0.1  Mask:255.0.0.0
      UP LOOPBACK RUNNING  MTU:16436  Metric:1
      RX packets:228 errors:0 dropped:0 overruns:0 frame:0
      TX packets:228 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:0
      RX bytes:17724 (17.3 Kb)  TX bytes:17724 (17.3 Kb)
```

Проверить возможность соединения с локальной машиной с помощью команды `ping 127.0.0.1`.

```
desktop ~ # ping 127.0.0.1
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.052 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.051 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.055 ms
```

Перед конфигурированием интерфейса `eth0` необходимо убедиться, что он отключен. Отключение сетевого интерфейса `eth0` производится командой `ifconfig eth0 down`.

```
desktop ~ # ifconfig eth0 down
```

```
desktop ~ # ifconfig -a
```

```
lo      Link encap:Local Loopback
        inet addr:127.0.0.1  Mask:255.0.0.0
        UP LOOPBACK RUNNING  MTU:16436  Metric:1
        RX packets:228 errors:0 dropped:0 overruns:0 frame:0
        TX packets:228 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0
        RX bytes:17724 (17.3 Kb)  TX bytes:17724 (17.3 Kb)
```

Для связи сетевого интерфейса *eth0* с IP-адресом 192.168.1.1 выполним команду `ifconfig eth0 192.168.1.1 up`.

```
desktop ~ # ifconfig eth0 192.168.1.1 up
```

```
desktop ~ # ifconfig -a
```

```
eth0    Link encap:Ethernet  HWaddr 00:0C:F1:2E:0E:F9
        inet addr:192.168.1.1  Bcast:192.168.1.255  Mask:255.255.255.0
        UP BROADCAST MULTICAST  MTU:1500  Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
        Interrupt:11 Base address:0x2000 Memory:c0210000-c0210fff
```

```
lo      Link encap:Local Loopback
        inet addr:127.0.0.1  Mask:255.0.0.0
        UP LOOPBACK RUNNING  MTU:16436  Metric:1
        RX packets:228 errors:0 dropped:0 overruns:0 frame:0
        TX packets:228 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0
        RX bytes:17724 (17.3 Kb)  TX bytes:17724 (17.3 Kb)
```

При этом по умолчанию используется сеть класса *C*, т.е. маска сети «255.255.255.0».

Для задания специфической маски подсети используется параметр `netmask`. Например, данная команда задаёт параметры сети класса *A*: `ifconfig eth0 10.10.1.1 netmask 255.0.0.0 up`.

```
desktop ~ # ifconfig eth0 10.10.1.1 netmask 255.0.0.0 up
```

```
desktop ~ # ifconfig -a
```

```
eth0    Link encap:Ethernet  HWaddr 00:0C:F1:2E:0E:F9
        inet addr:10.10.1.1   Bcast:10.255.255.255  Mask:255.0.0.0
        UP BROADCAST MULTICAST  MTU:1500  Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
        Interrupt:11 Base address:0x2000 Memory:c0210000-c0210fff
```

```
lo      Link encap:Local Loopback
        inet addr:127.0.0.1  Mask:255.0.0.0
        UP LOOPBACK RUNNING  MTU:16436  Metric:1
        RX packets:228 errors:0 dropped:0 overruns:0 frame:0
        TX packets:228 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0
        RX bytes:17724 (17.3 Kb)  TX bytes:17724 (17.3 Kb)
```

С помощью команды `arp` можно узнать текущую ARP-таблицу операционной системы (соответствие MAC-адресов канального уровня IP-адресам). Таблица автоматически поддерживается операционной системой в процессе сетевого обмена.

desktop ~ # `arp`

Address	HWtype	HWaddress	Flags	Mask	Iface
gate.localnet	ether	00:02:44:8F:16:B7	C		eth0

## Практическое занятие «Установка и настройка сервера MySQL»

**ЦЕЛЬ:** ознакомиться установкой, запуском MySQL Server 5.0, научиться подключаться к серверу баз данных MySQL в локальной сети, управлять учетными записями пользователей, ознакомиться с возможностями программ-утилит MySQL Administrator, Navicat for MySQL и phpMyAdmin по администрированию баз данных. ¶Теоретические сведения

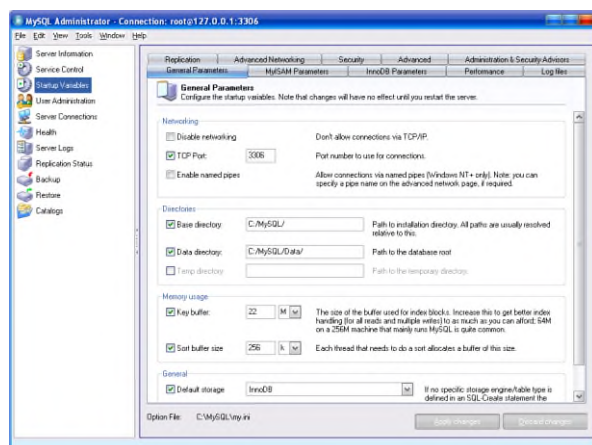
### 1. Вспомогательные утилиты MySQL

Различные вспомогательные утилиты MySQL располагаются в подкаталоге `bin` корневого каталога MySQL. Далее приведен список наиболее часто используемых утилит:

- `myisampack`— сжимает таблицы типа MyISAM, уменьшая их в размере и делая доступными только для чтения;
- `mysql` — консольный клиент для доступа к MySQL-серверу, позволяет выполнять SQL-запросы и осуществлять администрирование сервера;
- `mysqladmin`— утилита для выполнения административных функций, таких как создание или удаление баз данных, получения информации с сервера о номере версии, процессах, состоянии сервера и т. п.;
- `mysqlbinlog`— данная утилита используется для чтения содержимого журнала двоичной регистрации при восстановлении данных в нештатных ситуациях;
- `mysqlcheck` — утилита, используемая для описания, проверки, оптимизации и восстановления таблиц;
- `mysqldump`— выводит содержимое базы данных MySQL в виде файла с SQL-операторами или в виде текстовых файлов с символом табуляции в качестве разделителя. Такие файлы часто называют дампом (dump) базы данных;
- `mysqlhotcopy` — утилита для создания резервной копии таблиц без остановки сервера MySQL, т. е. создания "горячей" копии базы данных;
- `mysqlimport` — выполняет перенос информации из текстового файла в таблицы базы данных;
- `mysqlshow` — отображает информацию о существующих базах данных, таблицах, полях и индексах.

## 2. Графическая утилита MySQL Administrator

1. Назначение утилиты MySQL Administrator – собственная графическая утилита MySQL полноценного администрирования СУБД.¶

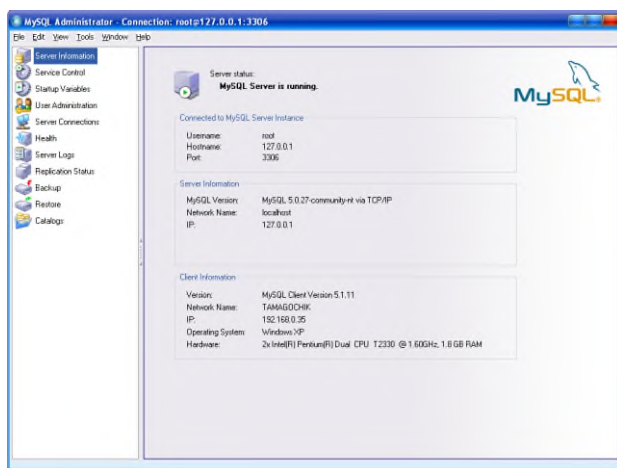


2.

Пакет ^ MySQL Administrator предоставляет продвинутые возможности настройки всех параметров сервера. ¶ Утилита предназначена для администрирования сервера, а также может выполнять операции резервного копирования и восстановления информации (как над отдельными, так и над всеми базами, причем есть встроенный планировщик заданий). Кроме СУБД MySQL поддерживается Oracle или другие БД через ODBC (не проверялось, но понятно, что в таком случае будут отключены все возможности, специфические именно для MySQL, да и программа разрабатывалась главным образом для MySQL, а все остальное – как бесплатный довесок). ¶ Основные функции утилиты сгруппированы в 11 пунктов графического меню в стиле MS Windows 2003. Утилита распространяется на правах OpenSorce GPL и доступна для платформы Win32 и Linux как в бинарном виде, так и в исходных кодах. Текущая версия 1.0.1 alpha работает довольно стабильно, хотя есть некоторые недочеты в интерфейсной части и часть функций не реализована (только интерфейсная часть).

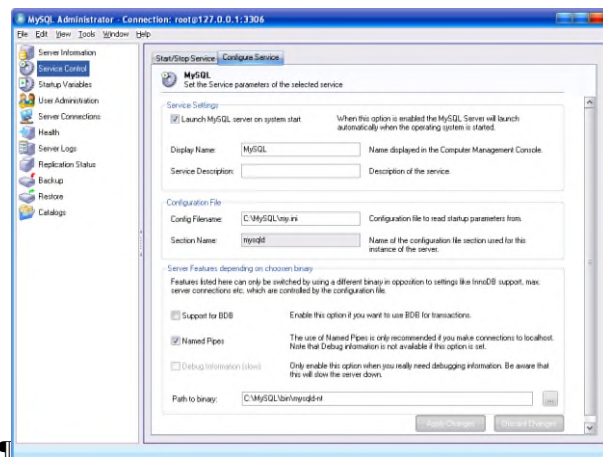
## 2. Меню и главные функции программы

Server Information. Первый пункт стандартен – Server Information. Здесь можно посмотреть на какой платформе запущен сервер, хост/порт, текущий аккаунт пользователя, IP, операционную систему, процессор и размер доступной памяти. Тут же показывается текущий статус сервера – работает или остановлен.¶



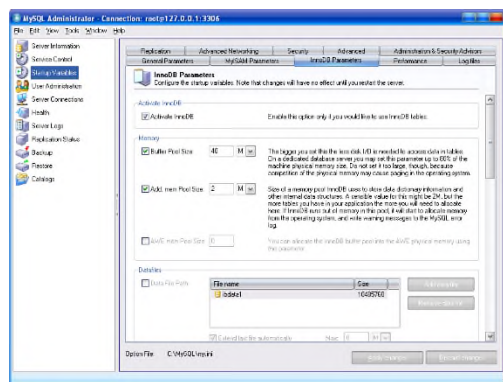


Service Control. Позволяет запустить или перезапустить сервер, просмотреть лог загрузки. Вторая вкладка позволяет настроить основные параметры – пути к файлу конфигурации, директории с бинарными файлами и настроить сервер для поддержки специфических расширений (например, таблиц формата InnoDB с поддержкой транзакций или использования для работы именованных каналов). В разных версиях сервера (mysqld-opt, mysqld-nt, mysqld-max, mysqld-max-nt), кроме различий в производительности, еще и по-разному реализована поддержка расширений (таблицы InnoDB поддерживаются в mysqld-opt, но не поддерживаются

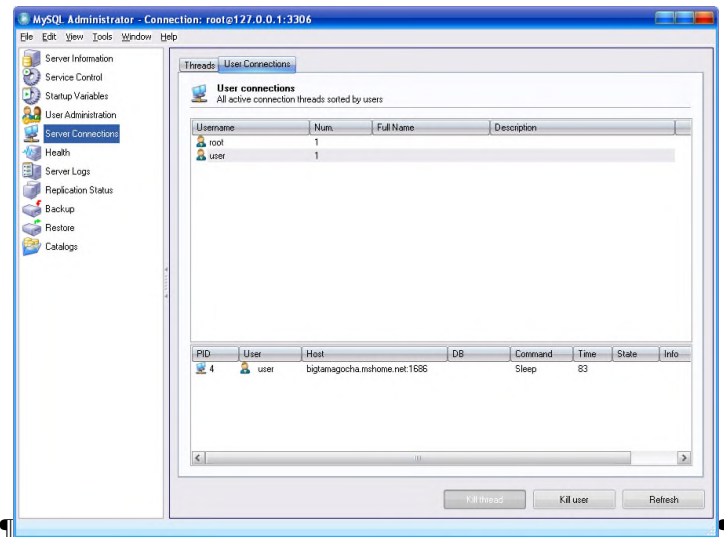


в стандартном mysqld).

Startup Variables. Этот раздел позволяет настроить множество тонких опций, влияющих на производительность сервера. Кроме базовых настроек (поддержка сетевых соединений, настройка портов, пути к служебным каталогам, настройка работы с памятью), отдельно можно настроить опции для различных типов таблиц (MyISAM, InnoDB). Для увеличения производительности есть настройки кеширования, для работы с несколькими серверами есть настройки репликации. Вкладка Advanced позволяет очень тонко настроить работу сервера, а некоторые опции нельзя настроить традиционным путем при помощи командной строки. На отдельных вкладках сгруппированы настройки безопасности и сетевые возможности. Как некоторый недочет - очень небольшое количество настроек безопасности, а ведь при серьезной работе в коммерческих приложениях безопасности уделяется повышенное внимание.

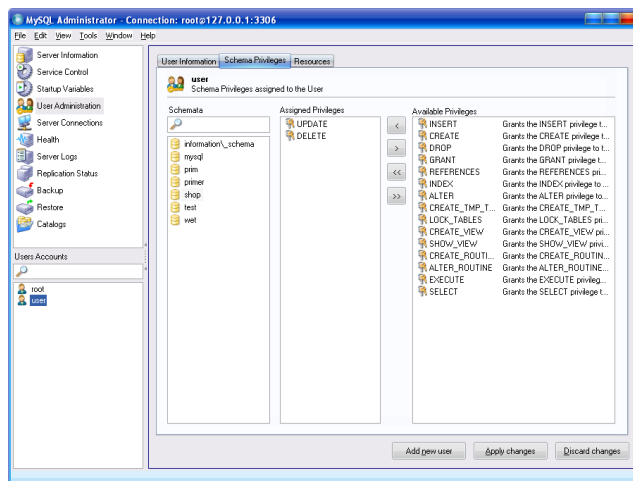


Server Connections. В этом меню можно просмотреть все текущие подключения клиентов к серверу



и при необходимости завершить любое из них.

User Administration Позволяет настроить привилегии для каждого пользователя и оперативно управлять ими (раньше для этого надо было вручную править служебные таблицы MySQL с помощью SQL запросов).



При использовании MySQL Administrator все операции над пользователями проходят в графическом режиме и в любое время за считанные секунды можно настроить привилегии для любого пользователя или базы данных. Эта функция одна из самых полезных и придется по вкусу администраторам и разработчикам. Health Меню Health чисто информационное и позволяет отслеживать производительность сервера в реальном времени, расход памяти и сетевой трафик, а также показывает статистику по типам запросов и эффективность задействования ключей. Эти возможности можно использовать, чтобы наблюдать изменения в работе сервера при изменении настроек, тестировать их влияние на производительность и таким образом постепенно настроить сервер на максимум для конкретного круга задач.

^ Server Logs Показывает стандартные логи сервера – общий лог и записи об ошибках.

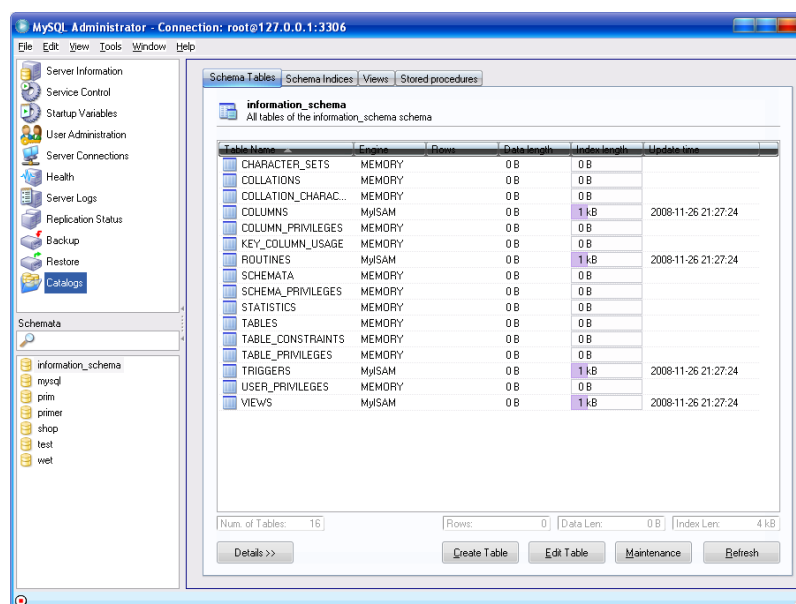
Вот самое интересное и полезное - это реализованные функции резервирования и восстановления БД. Реализован удобный планировщик, и теперь можно делать отдельное резервирование отдельных БД и таблиц – одни БД один раз в сутки, другие в конце недели и т.д. Опции резервирования также гибко настраиваются – для этого есть вкладка Advanced Options. Пока для резервирования доступен лишь один формат - простой текстовый файл с SQL запросами, без сжатия и т.д. Но будем надеяться, что в следующих версиях будет хоть какой-то алгоритм сжатия.

К сожалению, вся вкладка Schedule пока неактивна – видимо, разработчики еще не до конца протестировали эту возможность. Все-таки, этот релиз обозначен как альфа. В любом случае, путь, по которому пошли разработчики не может не радовать – наконец СУБД MySQL обзаводится мощными средствами конфигурирования и настройки и теперь ее можно применять и в корпоративных приложениях, а не только в Web-проектах. ¶

При восстановлении из резервной копии также можно выбрать отдельные таблицы и базы данных, которые подлежат восстановлению.

Replication Status. Функция Replication Status скорее всего еще до конца не реализована, присутствует только вкладка Server Information. Для реализации и настройки репликации надо перейти на вкладку Replication раздела Startup variables, где можно детально настроить параметры репликации. ¶

Catalogs. Последний раздел дает доступ к отдельным базам и таблицам, позволяет осуществлять операции оптимизации, проверки и восстановления таблиц, а также показывает сведения об индексах и их параметрах, а также пользователей, которые имеют доступ к конкретной базе или таблице. ¶



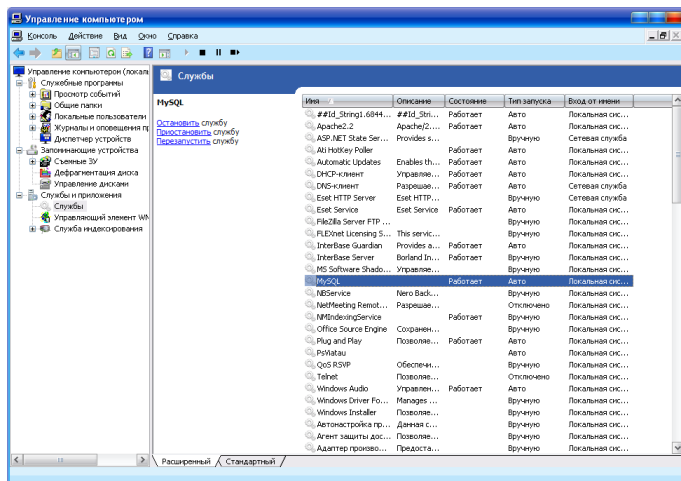
В целом, MySQL Administrator способна обеспечить все потребности администратора или разработчика, как по управлению пользователями, так и по тонкой настройке производительности, анализе работы сервера, и может служить инструментом для резервирования и репликации баз данных. Для полноценного использования, думаю, еще надо подождать следующего релиза, где будут исправлены все недочеты и увеличена функциональность (в основном это касается функций резервирования). ¶ Распространение такого мощного средства на условиях OpenSource будет способствовать продвижению СУБД и теперь можно ожидать, что появятся вполне серьезные разработки корпоративного масштаба, основанные на MySQL. Версия сервера – 5.0 выполняет поддержку хранимых процедур. ¶^

## **ПОСЛЕДОВАТЕЛЬНОСТЬ ВЫПОЛНЕНИЯ РАБОТЫ:**

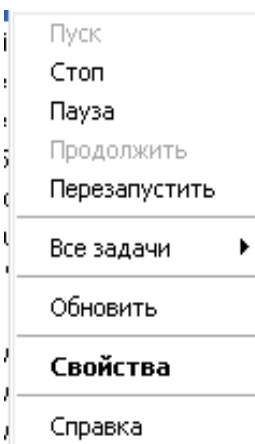
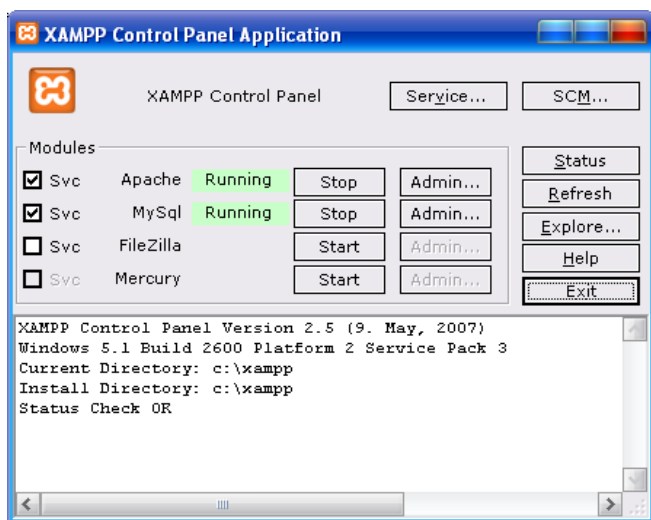
### **Часть 1 – Установка, конфигурирование, запуск MySQL Server 5.0 и XAMPP**

1. ¶ Установить MySQL Server 5.0 (или XAMPP, в составе которого также есть сервер MySQL) на компьютер пользователя (если он установлен).

2. ¶Проверить состояние (статус) сервера на данный момент, задать настройки запуска сервера MySQL на компьютере пользователя – открыть консоль управления сервисами, выполнив команду Пуск | Настройка | Панель управления | Администрирование | Службы или вызвать окно Управление компьютером. В результате этого откроется окно, представленное на рис. В консоли управления сервисами необходимо найти сервис MySQL.¶



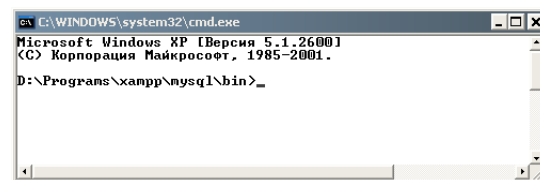
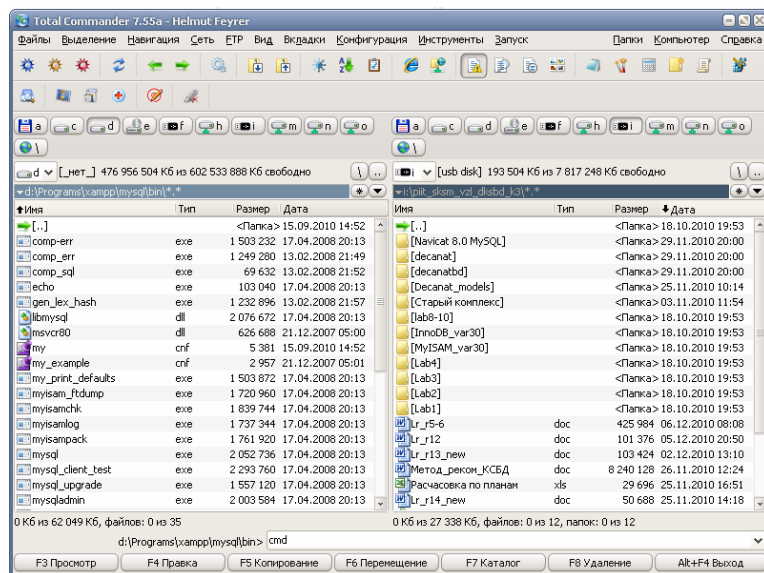
¶Рис. Консоль управления сервисами¶



¶ЗАМЕЧАНИЕ: Если поле Состояние данного сервиса пусто, то он не запущен. Для его запуска следует выбрать в Контекстном меню пункт Пуск. Для остановки сервиса необходимо выбрать пункт Стоп. В столбце Тип запуска: значение Авто сообщает Windows о необходимости запускать сервис при старте операционной системы, значение Вручную означает необходимость запуска сервиса пользователем через консоль управления сервисами. Можно изменить режим запуска сервиса, выбрав в контекстном меню сервиса пункт Свойства. ¶Также можно воспользоваться диалоговым окном XAMPP Control Panel для просмотра сведений о состоянии сервера. Если сервер не запущен, то выполнить перезапуск XAMPP (файл xampp\_restart.exe), и в случае успешного перезапуска, проверить состояние сервера, например, в диалоговом окне управляющей панели.

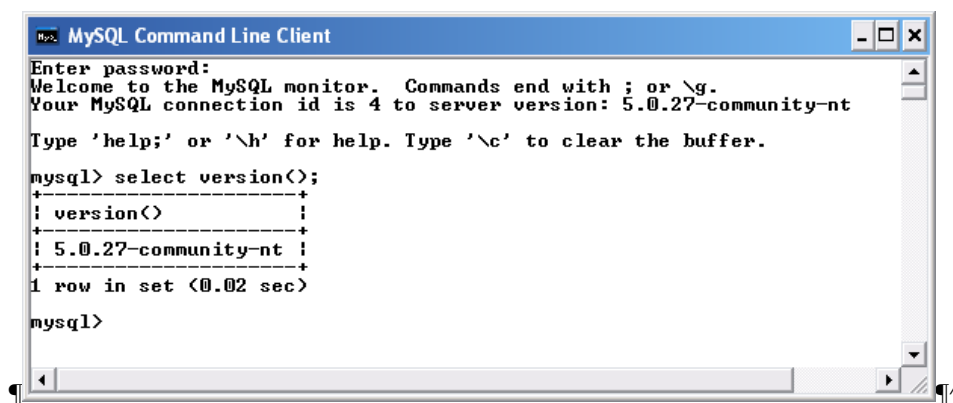
3. ¶Запустить командную строку.

**ЗАМЕЧАНИЕ:** для удобства последующего запуска сервера MySQL рекомендуется для запуска командной строки выполнить следующие действия: запустить программу-надстройку TotalCommander, в одной из панелей войти в папку ...MYSQL\BIN, затем в командной строке внизу экрана программы прописать команду cmd и нажать ENTER. При этом каталог BIN в командной строке в приглашении MS-DOS будет сразу установлен рабочим.



...MYSQL\BIN, если он не является рабочим.

4. ¶Получить сведения об IP-адресе компьютера, выполнив команду ipconfig. Полученный адрес выписать в тетрадь. Получить сведения об IP-адресах еще 4-х компьютеров в локальной сети и записать.
5. ¶Сделать рабочим каталог
6. ¶Подключиться к локальному серверу MySQL, как пользователь root
7. ¶Получить сведения о версии сервера – после того, как появилось приглашение mysql>, ввести команду SELECT VERSION();



**Часть 2 – Работа с утилитой MySQL. Управление учетными записями пользователей: создание, удаление, переименование**

8. ¶Получить сведения о существующих учетных записях на локальном сервере.
9. ¶Создать по указанию преподавателя следующие учетные записи пользователей:
  1. ¶Имя пользователя – фамилия студента, возможность обращения только с локального сервера, задать пароль 123456;

2. ¶Имя пользователя – user, возможность обращения с любого хоста только в локальной сети той аудитории, где проводится занятие, например, 153;

3. ¶3 учетные записи, в которых имя пользователя – фамилии студентов группы с номерами мониторов, например, petrov04 (поочередно), возможность обращения только с хостов данных студентов группы, пароли можно не задавать.

¶ЗАМЕЧАНИЕ: выписать на доску имена пользователей, созданных учетных записей для одноклассников.

10. ¶Получить сведения о существующих учетных записях на локальном сервере и их привилегиях (грантах).

11. ¶Выполнив выход из-под пользователя root, подключиться локально на свой сервер под своей фамилией (авторизировать пользователя).

¶ЗАМЕЧАНИЕ: для захода под паролем, следует ввести команду следующего шаблона¶...\\MYSQL\\BIN> mysql –u <имя пользователя> –p¶Имя реального пользователя в угловые скобки не брать!

12. ¶Выполнить выход.

13. ¶Подключиться, как user к любому хосту в локальной сети 153.

¶ЗАМЕЧАНИЕ: для захода под паролем, следует ввести команду следующего шаблона¶...\\MYSQL\\BIN> mysql –u <имя пользователя> --host= ¶IP-адрес ввести в виде, полученном по команде ipconfig¶Дописать в конце команды –p (если есть пароль на вход).

14. ¶Выполнить выход.

15. ¶Подключиться к серверу на хосте студента группы, который зарегистрировал Вас.  
Выполнить выход.

16. ¶Авторизировать пользователя root.

17. ¶Переименовать учетную запись user в superuser.

¶ЗАМЕЧАНИЕ: задать в команде старое имя пользователя с указанием хоста.

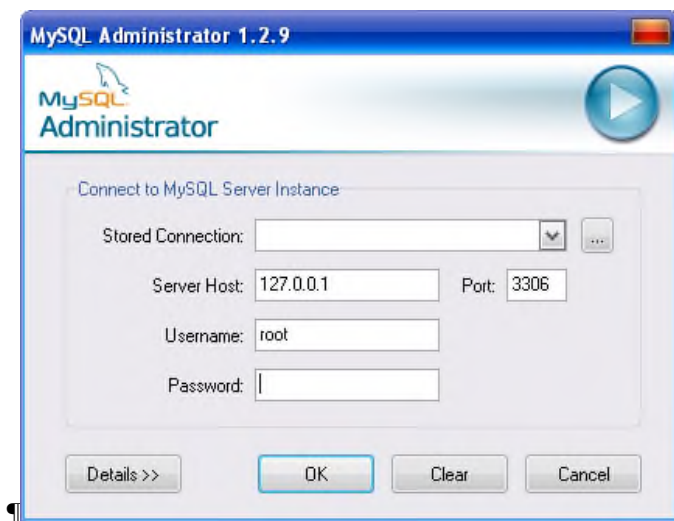
18. ¶Удалить учетную запись с фамилией студента на локальном сервере.

¶ЗАМЕЧАНИЕ: задать в команде имя пользователя с указанием хоста.

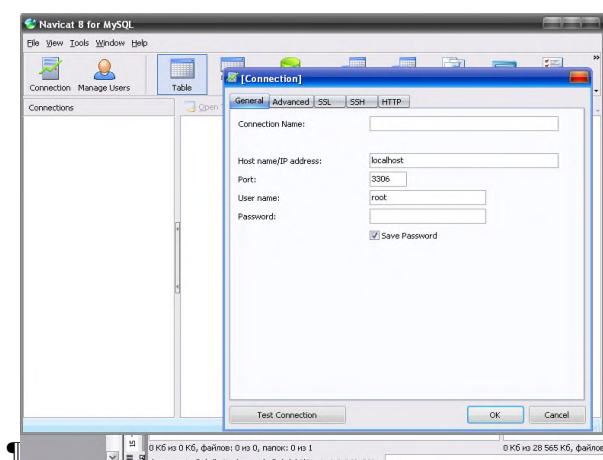
19. ¶Завершить работу с окном клиента.

¶^ Часть 3 – Работа с графическими программами-утилитами MySQL Administrator, Navicat 8 for MySQL и phpMyAdmin

20. ¶ Установить утилиту MySQL Administrator на компьютер пользователя в папку, с установленным сервером MySQL, если она не установлена.
21. ¶ Скопировать в папку Data учебную базу с заданным именем по указанию преподавателя.
22. ¶ Подключиться к серверу локально, как пользователь root

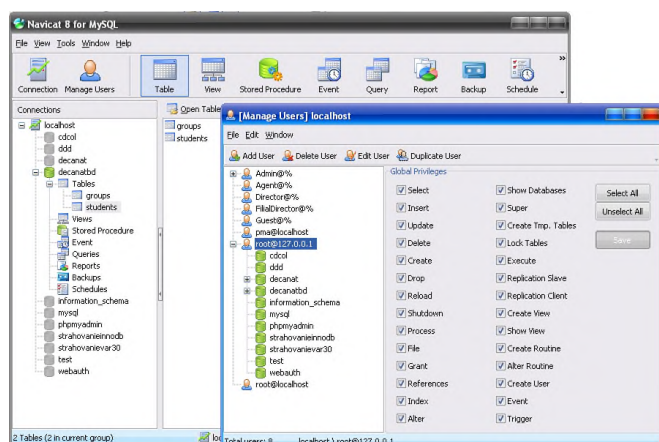


23. ¶ Ознакомиться с экраном программы, пунктами графического меню. Просмотреть учетные записи пользователей, привилегии, а также содержимое таблиц, выложенных на сервере.
24. ¶ Выполнить выход File \ Reconnect и подключиться к серверу на хосте студента группы, который зарегистрировал Вас. Ознакомиться с его учетными записями. Выполнить выход.
25. ¶ Запустить программу-утилиту Navicat for MySQL (если она не установлена, то можно запустить, не устанавливая, локально из папки по указанию преподавателя).
26. ¶ Подключиться к серверу локально, как пользователь root, нажав на кнопку Connection и заполнив поля на вкладке General. Можно также выполнить тест подключения (кнопка Test Connection).





27. Ознакомиться с экраном программы, пунктами графического меню. Просмотреть учетные записи пользователей (кнопка Manage Users), привилегии, а также содержимое таблиц, выложенных на сервере.



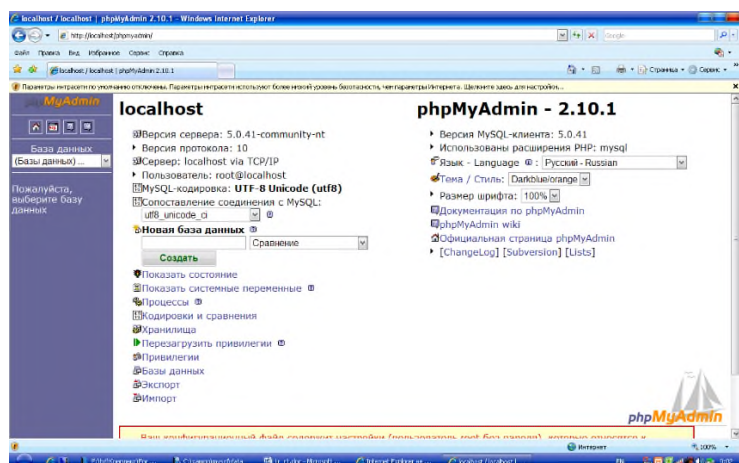
28. Выполнить выход File \ Close Connection и подключиться к серверу на хосте студента группы, который зарегистрировал Вас (кнопка Connection). Ознакомиться с его учетными записями. Выполнить выход (File \ Close).

29. Запустить браузер и в строке адреса ввести localhost

**ЗАМЕЧАНИЕ:** предварительно убедиться, что сервер Apache (в диалоговом окне XAMPP) запущен.

30. Выбрать при первом запуске английский язык, перейти по ссылке [phpMyAdmin](http://localhost/phpMyAdmin)

31. Аналогично ознакомиться с окном, пунктами графического меню и т.д.



**ЗАМЕЧАНИЕ:** обратите внимание, что при подключении к любому хосту в локальной сети (например, если прописать в строке адреса в браузере 192.168.153.102), происходит подключение к серверу на заданном хосте под именем root, что автоматически дает полный доступ с Вашего ПК к базам данных сервера (как администратора). На странице загрузки будет выведено соответствующее сообщение и рекомендация о желательной смене пароля пользователя root с заданного по умолчанию на другой. В реальных рабочих условиях это необходимо, но в учебных целях пароль (пустая строка) следует оставить без изменения. Для корректной замены пароля следует зайти в главное окно XAMPP в левой панели по ссылке Security и в появившемся окне выбрать ссылку



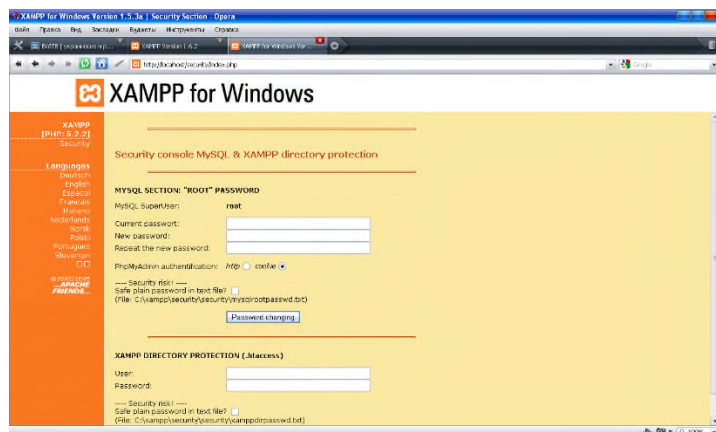
<http://localhost/security/xamppsecurity.php>,

затем

поменять

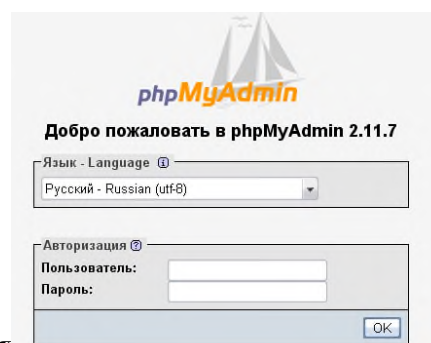
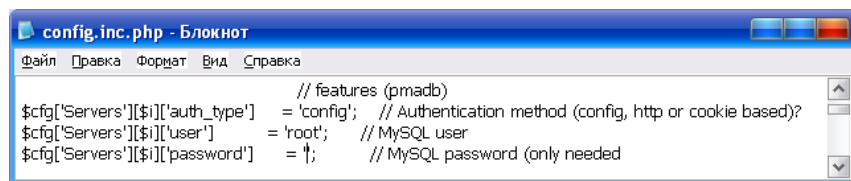
пароль.

¶



¶Если выполнить смену пароля иначе, например,

исправить учетную запись на странице Привилегии, то в дальнейшем следует в файле конфигурации <путь к каталогу>\xampp\phpMyAdmin\config.inc.php пароль изменить вручную в соответствующей строке (на рисунке — это строка, где установлен текстовый курсор ).¶



Если

пароль на root был изменен, то нужно помнить, что новый пароль будет действителен во всех утилитах при обращении к серверу.

32. Для создания диалогового окна входа нужно в файле config.inc.php дописать следующие строки, если их нет: ¶\$cfg['Servers'][\$i]['auth\_type'] = 'cookie';¶\$cfg['Servers'][\$i]['host'] = 'localhost';¶Строки из предыдущего диалогового окна закомментировать. Вместо localhost можно указать IP-адрес нужного сервера. Предварительно создать учетные записи.

32. ¶Оформить отчет.

### Система оценивания:

Содержание и объем материала, подлежащего проверке в контрольной работе, определяется программой. При проверке усвоения материала выявляется полнота, прочность усвоения учащимися теории и умение применять ее на практике в знакомых и незнакомых ситуациях.

Отметка зависит также от наличия и характера погрешностей, допущенных учащимися.

- *грубая ошибка* - полностью искажено смысловое значение понятия, определения;
- *погрешность* отражает неточные формулировки, свидетельствующие о нечетком представлении рассматриваемого объекта;
- *неДочет* - неправильное представление об объекте, не влияющего кардинально на знания

определенные программой обучения;

- *мелкие погрешности* - неточности в устной и письменной речи, не искажающие смысла ответа или решения, случайные описки и т.п.

Эталоном, относительно которого оцениваются знания учащихся, является обязательный минимум содержания информатики и информационно-коммуникационных технологий.

Исходя из норм (четырёхбалльной системы), заложенных во всех предметных областях выставляете отметка:

«5» ставится при выполнении всех заданий полностью или при наличии 1 -2 мелких погрешностей;

«4» ставится при наличии 1-2 недочетов или одной ошибки;

«3» ставится при выполнении 2/3 от объема предложенных заданий;

«2» ставится, если допущены существенные ошибки, показавшие, что учащийся не владеет обязательными умениями поданной теме в полной мере (незнание основного программного материала)

## **2.2. Перечень вопросов и заданий для промежуточной аттестации**

**Форма: дифференцированный зачет**

### **Контрольные вопросы.**

1. Распределенные и централизованные базы данных. Архитектура файл-сервер. Архитектура клиент-сервер.
2. Иерархическая и сетевая модели данных.
3. Реляционная модель данных. История развития. Основные понятия (тип данных, домен, отношение, кортеж, атрибут, ключ).
4. Реляционная база данных.
5. Функции системы управления базами данных (СУБД): управления данными во внешней памяти, управление буферами оперативной памяти, управление транзакциями.
6. Функции системы управления базами данных: журнализация, поддержка языков баз данных.
7. Типовая организация современной СУБД.
8. Базовые средства манипулирования реляционными данными.
9. Реляционная алгебра. Общая интерпретация реляционных операций.
10. Особенности теоретико-множественных операций реляционной алгебры.
11. Реляционное исчисление.
12. Схема отношения. Функциональные зависимости. Декомпозиция отношений, транзитивные зависимости.
13. Нормализация отношений. Проектирование с использованием метода сущность - связь.
14. 1, 2, 3 и 4 нормальные формы. Нормальная форма Бойса-Кодда. Приведение базы данных к нормализованному виду.
15. Целостность сущностей и ссылок.

16. История развития SQL. Функции и основные возможности SQL. ANSI SQL; T-SQL; PL/SQL; Jet SQL
17. Выражения в SQL.
18. СУБД в архитектуре клиент-сервер. Открытые системы.
19. Клиенты и серверы локальных сетей.
20. Системная архитектура клиент-сервер. Удаленный вызов процедур.
21. Сервера баз данных.
22. Типичное распределение функций между клиентом и сервером. Распределенные базы данных.
23. Создание и модификация базы данных в MS SQL Server.
24. Сортировка и поиск данных в MS SQL Server.
25. Язык T-SQL. Числовые и денежные типы данных. Типы данных для хранения информации о времени.
26. Язык T-SQL. Символьные и текстовые типы данных.
27. Язык T-SQL. Специальные типы данных. Конвертирование типов данных.
28. Процесс проектирования таблиц в реляционной базе данных. Определение идентификационной колонки.
29. Создание таблиц средствами T-SQL.
30. Изменение структуры таблицы средствами T-SQL. Удаление таблиц.
31. Добавление данных в таблицу средствами T-SQL. Использование INSERT и SELECT...INTO.
32. Извлечение данных средствами T-SQL. Команда SELECT. Разделы SELECT и INTO.
33. Извлечение данных средствами T-SQL. Команда SELECT. Раздел FROM.
34. Извлечение данных средствами T-SQL. Команда SELECT. Разделы WHERE, GROUP BY, HAVING, ORDER BY.
35. Изменение данных в таблице средствами T-SQL. Команда UPDATE.
36. Удаление данных средствами T-SQL. Команда DELETE.
37. Transact-SQL ODBC и MS SQL Server.
38. Использование представлений.
39. Хранимые процедуры. Этапы создания.
40. Создание, модификация и удаление хранимых процедур средствами T-SQL.
41. Создание, изменение и удаление представлений средствами T-SQL.
42. Современные промышленно-сопровождаемые СУБД
43. Системы управления базами данных следующего поколения.

***Критерии оценивания устного ответа на экзамене***

**Оценка «5» («отлично»)** соответствует следующей качественной характеристике: «изложено правильное понимание вопроса и дан исчерпывающий на него ответ, содержание раскрыто полно, профессионально, грамотно».

Выставляется студенту,

- усвоившему взаимосвязь основных понятий дисциплины в их значении для приобретаемой профессии, проявившему творческие способности в понимании, изложении и использовании учебно-программного материала;

- обнаружившему всестороннее систематическое знание учебно-программного материала, четко и самостоятельно (без наводящих вопросов) отвечающему на вопрос билета.

**Оценка «4» («хорошо»)** соответствует следующей качественной характеристике: «изложено правильное понимание вопроса, дано достаточно подробное описание предмета ответа, приведены и раскрыты в тезисной форме основные понятия, относящиеся к предмету ответа, ошибочных положений нет».

Выставляется студенту,

- обнаружившему полное знание учебно-программного материала, грамотно и по существу отвечающему на вопрос билета и не допускающему при этом существенных неточностей;

- показавшему систематический характер знаний по дисциплине и способному к их самостоятельному пополнению и обновлению в ходе дальнейшей учебы и профессиональной деятельности.

**Оценка «3» («удовлетворительно»)**

Выставляется студенту,

- обнаружившему знание основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, справляющемуся с выполнением заданий, предусмотренных программой;

- допустившему неточности в ответе и при выполнении экзаменационных заданий, но обладающими необходимыми знаниями для их устранения под руководством преподавателя.

**Оценка «2» («неудовлетворительно»)**

Выставляется студенту,

- обнаружившему существенные пробелы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий;

- давшему ответ, который не соответствует вопросу экзаменационного билета.

## **Тестовое задание.**

1 Предметная область - это:

- 1) совокупность таблиц, состоящих из записей и полей; информации об индексах и связях; хранимых процедур;
- 2) совокупности таблиц, объединенных связями; экранных форм, отчетов, запросов
- 3) некоторая часть реально существующей системы, функционирующая как самостоятельная единица;
- 4) поименованная совокупность структурированных данных, относящихся к определенной предметной области;

- 5) набор правил, обеспечивающих соответствие ключевых значений в связанных таблицах.

## 2 Система управления базой данных (СУБД) - это:

- 1) регулярная структура, состоящая из однотипных записей, разбитых на поля;
- 2) комплекс программных и языковых средств, необходимых для создания и модификации базы данных;
- 3) поименованная совокупность структурированных данных, относящихся к определенной предметной области;
- 4) служебная информация, содержащая упорядоченные сведения о ключевых значениях;
- 5) программно-аппаратный комплекс, предназначенный для хранения и обработки информации какой-либо предметной области.

## 3. База данных - это:

- 1) комплекс программных и языковых средств, необходимых для добавления, модификации, удаления, поиска и отбора информации;
- 2) совокупности таблиц, объединенных связями; экранных форм, отчетов, запросов;
- 3) некоторая часть реально существующей системы, функционирующая как самостоятельная единица;
- 4) поименованная совокупность структурированных данных, относящихся к определенной предметной области;
- 5) программно-аппаратный комплекс, предназначенный для хранения и обработки информации какой-либо предметной области.

## 4 Реляционная модель базы - это:

- 1) совокупность таблиц, состоящих из записей и полей; информации об индексах и связях; хранимых процедур;
- 2) совокупности таблиц, объединенных связями; экранных форм, отчетов, запросов;
- 3) некоторая часть реально существующей системы, функционирующая как самостоятельная единица;
- 4) поименованная совокупность структурированных данных, относящихся к определенной предметной области;
- 5) набор правил программно-аппаратный комплекс, предназначенный для хранения и обработки информации какой-либо предметной области.

## 5 Таблица базы данных - это:

- 1) регулярная структура, состоящая из однотипных записей, разбитых на поля;
- 2) комплекс программных и языковых средств, необходимых для создания и модификации базы данных;

- 3) поименованная совокупность структурированных данных, относящихся к определенной предметной области;
- 4) служебная информация, содержащая упорядоченные сведения о ключевых значениях;
- 5) функциональная зависимость между объектами.

6. Ключ таблицы базы данных - это:

- 1) поле или строковое выражение, образованное из значений нескольких полей, по которому можно определить значения других полей для одной или нескольких записей таблицы;
- 2) поле или строковое выражение, образованное из значений нескольких полей, по которому можно однозначно идентифицировать строку в таблице;
- 3) программный модуль, сохраняемый в базе данных для выполнения определенных операций с информацией базы;
- 4) поименованная совокупность структурированных данных, относящихся к определенной предметной области;
- 5) набор правил, обеспечивающих связи между таблицами в базе данных.

7. Отношение в теории реляционных баз данных - это:

- 1) основной объект базы данных, состоящий из кортежей и имеющий определенный набор свойств – атрибутов;
- 2) набор всех допустимых значений, которые может содержать атрибут;
- 3) формальный метод анализа отношений на основе их первичного ключа и существующих функциональных зависимостей;
- 4) функциональная зависимость между объектами;
- 5) математические принципы, вытекающие из теории множеств и логики предикатов/

8. Связи между ключевыми значениями в реляционной модели бывают:

- 1) "один к одному", "один ко многим", "многие ко многим";
- 2) только "один к одному";
- 3) только "один ко многим";
- 4) только "многие ко многим".

9. Сетевая модель данных состоит из:

- 1) набора экземпляров одного типа, образующих дерево с одним корневым объектом;
- 2) набора записей и набора связей с любым числом других записей;
- 3) совокупности таблиц со связями по ключевым значениям;
- 4) многомерных таблиц, созданных с использованием объектно-ориентированных методов;
- 5) множества баз данных, управляемых одной СУБД.

10. Реляционная модель данных состоит из:

- 1) набора экземпляров одного типа, образующих дерево с одним корневым объектом;

- 2) набора записей и набора связей с любым числом других записей;
  - 3) совокупности таблиц со связями по ключевым значениям;
  - 4) многомерных таблиц, созданных с использованием объектно-ориентированных методов;
  - 5) множества баз данных, управляемых одной СУБД.
11. Иерархическая модель данных состоит из:
- 1) набора экземпляров одного типа, образующих дерево с одним корневым объектом;
  - 2) набора записей и набора связей с любым числом других записей;
  - 3) совокупности таблиц со связями по ключевым значениям;
  - 4) многомерных таблиц, созданных с использованием объектно-ориентированных методов;
  - 5) множества баз данных, управляемых одной СУБД.
12. Использование каких моделей данных наиболее эффективно в системах класса OLAP:
- 1) Реляционных;
  - 2) Объектно-ориентированных;
  - 3) Многомерных;
  - 4) Сетевых.
13. Предметно-ориентированный, интегрированный, неизменяемый и поддерживающий хронологию набор данных, предназначенный для обеспечения принятия управленческих решений, называется:
- 1) Банком данных;
  - 2) Информационным массивом;
  - 3) Хранилищем данных;
  - 4) Информационной системой.
14. Какое из перечисленных высказываний не является верным по отношению к объектно-ориентированным базам данных (ООБД):
- 1) При ссылке на объекты необходимо повторять пользовательские ключи;
  - 2) Все объекты ООБД идентифицируются одинаковым образом;
  - 3) Идентификаторы никогда не изменяются до тех пор, пока существуют объекты, которые они идентифицируют;
  - 4) Идентификаторы не характеризуются излишней сложностью.
15. В случае, если СУБД по отношению к базе данных выполняет не только те действия, которые явно указывает пользователь, но и дополнительные действия в соответствии с правилами, заложенными в саму СУБД, база данных называется:
- 1) Многомерной;
  - 2) Активной;
  - 3) Реляционной;

- 4) Дедуктивной.
16. Оператор SQL, выполняющий проверку на диапазон значений:
- 1) FROM...TO;
  - 2) BETWEEN...AND;
  - 3) FROM...AND;
  - 4) BETWEEN...TO.
17. Оператор IN в языке SQL выполняет:
- 1) Проверку выражения на NULL;
  - 2) Проверку выражения на совпадение с любым из элементов списка;
  - 3) Проверку выражения на совпадение со всеми элементами списка;
  - 4) Логическую импликацию выражений.
18. Какая команда SQL осуществляет выбор пяти первых фамилий студентов, упорядоченных по учебным группам:
- 1) SELECT Имя, Фамилия FROM Студент[Группа] LIMIT 5;
  - 2) SELECT Имя, Фамилия ORDER BY [Группа] FROM Студент LIMIT 5;
  - 3) SELECT Имя, Фамилия FROM Студент ORDER BY[Группа] LIMIT 5;
  - 4) SELECT Имя, Фамилия ORDER BY[Группа] WHERE Студент LIMIT 5.
19. Чем отличаются подходы, применяемые в реляционной алгебре и реляционном исчислении?
- 1) Реляционная алгебра использует описательный подход, а реляционное исчисление предписывающий;
  - 2) Оба подхода описательные;
  - 3) Оба подхода предписывающие;
  - 4) Реляционная алгебра использует предписывающий подход, а реляционное исчисление описательный.
20. Какой из перечисленных тестов для СУБД измеряет насколько быстро СУБД может выполнять однотабличный запрос, ответ на который содержит определенную процентную долю строк таблицы:
- 1) Полное сканирование;
  - 2) Обновление;
  - 3) Чтение с произвольной выборкой;
  - 4) Выборка.

***Критерии оценивания устного ответа на экзамене***

**Оценка «5» («отлично»)** соответствует следующей качественной характеристике:



«изложено правильное понимание вопроса и дан исчерпывающий на него ответ, содержание раскрыто полно, профессионально, грамотно».

Выставляется студенту,

- усвоившему взаимосвязь основных понятий дисциплины в их значении для приобретаемой профессии, проявившему творческие способности в понимании, изложении и использовании учебно-программного материала;
- обнаружившему всестороннее систематическое знание учебно-программного материала, четко и самостоятельно (без наводящих вопросов) отвечающему на вопрос билета.

**Оценка «4» («хорошо»)** соответствует следующей качественной характеристике: «изложено правильное понимание вопроса, дано достаточно подробное описание предмета ответа, приведены и раскрыты в тезисной форме основные понятия, относящиеся к предмету ответа, ошибочных положений нет».

Выставляется студенту,

- обнаружившему полное знание учебно-программного материала, грамотно и по существу отвечающему на вопрос билета и не допускающему при этом существенных неточностей;
- показавшему систематический характер знаний по дисциплине и способному к их самостоятельному пополнению и обновлению в ходе дальнейшей учебы и профессиональной деятельности.

**Оценка «3» («удовлетворительно»)**

Выставляется студенту,

- обнаружившему знание основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, справляющемуся с выполнением заданий, предусмотренных программой;
- допустившему неточности в ответе и при выполнении экзаменационных заданий, но обладающими необходимыми знаниями для их устранения под руководством преподавателя.

**Оценка «2» («неудовлетворительно»)**

Выставляется студенту,

- обнаружившему существенные пробелы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий;
- давшему ответ, который не соответствует вопросу экзаменационного билета.

### 3. Рекомендуемая литература и иные источники

#### Основные источники:

1. Данилова, Л. Ф. Организация баз данных : учебно-методическое пособие / Л. Ф. Данилова, А. Н. Полетайкин. — Новосибирск : Сибирский государственный университет телекоммуникаций и информатики, 2022. — 152 с. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/138776.html>

2. Разработка баз данных : учебное пособие / А. С. Дорофеев, Р. С. Дорофеев, С. А. Рогачева, С. С. Сосинская. — Москва : Ай Пи Ар Медиа, 2024. — 240 с. — ISBN 978-5-4497-2576-9. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/134888.html>

#### Дополнительная литература:

1. Блам, Р. Администрирование почтовых серверов sendmail : учебное пособие / Р. Блам. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 702 с. — ISBN 978-5-4497-0857-1. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/101989.html> — Режим доступа: для авторизир. пользователей

2. Введение в СУБД MySQL : учебное пособие / . — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 228 с. — ISBN 978-5-4497-0912-7. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/102004.html> — Режим доступа: для авторизир. пользователей

3. Королев, Е. Н. Администрирование СУБД : учебное пособие / Е. Н. Королев, Б. Н. Тишуков, А. В. Мандрыкин. — Воронеж : Воронежский государственный технический университет, ЭБС АСВ, 2021. — 156 с. — ISBN 978-5-7731-0947-1. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/118629.html> — Режим доступа: для авторизир. пользователей

4. Костюк, А. И. Администрирование баз данных и компьютерных сетей : учебное пособие / А. И. Костюк, Д. А. Беспалов. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2020. — 127 с. — ISBN 978-5-9275-3577-4. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/107941.html> — Режим доступа: для авторизир. пользователей

5. Оптимизация работы серверов баз данных Microsoft SQL Server 2005 : учебное пособие / . — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 372 с. — ISBN 978-5-4497-0901-1. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/102024.html> — Режим доступа: для авторизир. пользователей

## ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

<b>№ п.п.</b>	<b>Содержание изменения</b>	<b>Дата, номер протокола заседания кафедры, подпись зав.кафедрой</b>
1	2	3
1		
2		
3		
4		