

	ЧОУ ВО «Липецкий эколого-гуманитарный институт»	СИСТЕМА МЕНЕДЖМЕНТА КАЧЕСТВА	
	АННОТАЦИЯ ДИСЦИПЛИНЫ		
	Базы данных		Стр. 1 из 34

УТВЕРЖДАЮ

Ректор

_____ В.Ю. Филоненко

«2» сентября 2022 г.

АННОТАЦИЯ
к рабочей программе дисциплины
Базы данных
(наименование дисциплины)

Направление подготовки: 38.03.05 – Бизнес- информатика

Профиль подготовки: Электронный бизнес

Квалификация выпускника: бакалавр

Форма обучения: очная/очно-заочная

Липецк –2022 г.



1. Цели освоения дисциплины

Целями изучения дисциплины «Базы данных» являются:

- изучение студентами моделей информационных структур,
- методов организации хранения данных,
- приобретение навыков практической работы создания типизированных вычислительных модулей.

2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОП

Выпускник должен обладать следующими компетенциями:

умение проектировать и внедрять компоненты ИТ-инфраструктуры предприятия, обеспечивающие достижение стратегических целей и поддержку бизнес-процессов (ПК-13);

умение осуществлять планирование и организацию проектной деятельности на основе стандартов управления проектами (ПК-14).

В результате освоения дисциплины обучающийся должен:

знать:

понятия, связанные с реляционной моделью данных;

принципы построения и функционирования систем управления базами данных,

методы, используемые для проектирования БД: метод декомпозиции, категории стандартов, используемых при проектировании баз данных;

также основные возможности СУБД реляционного типа

уметь:

применять на практике формальные методы построения БД;

также уметь применять средства выбранной СУБД для реализации прикладного ПО

владеть:

методами проектирования баз данных.

3. Место дисциплины в структуре ОП ВО

Дисциплина «Базы данных» входит в базовый блок дисциплин направления (Б1.Б.27), обеспечивающих подготовку бакалавра по направлению «Бизнес-информатика». Читается на 2,3 семестрах – очная форма обучения, уст., 3,4 семестрах – заочная форма обучения.

Для освоения дисциплины в первую очередь студенты используют знания, умения, навыки, сформированные в процессе изучения дисциплин направления: «Теоретические основы информатики», «Вычислительные системы, сети и телекоммуникации», «Общая теория систем», «Информационная безопасность»; методологически и содержательно тесно связана с изучением дисциплины «Программирование», является предшествующей для дисциплин «Информационные системы и технологии», «Объектно-ориентированный анализ и программирование»; полученные знания помогут в проведении НИРС и выполнении ВКР.

4. Объем дисциплины (модуля) в зачетных единицах с указанием академических или астрономических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Объем дисциплины - 5 зачетных единиц, 180 часов,

Очная форма обучения: контактная работа – 90 час. (лекции – 36 час.; лабораторные занятия - 54 час.); экзамен – 36 час.; самостоятельная работа обучающихся – 54 часа.



Заочная форма обучения: контактная работа - 24 часа (лекции – 8 час.; лабораторные занятия – 4 час.; консультации – 12 час.); экзамен – 36 час.; самостоятельная работа обучающихся – 120 час.

5. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических или астрономических часов и видов учебных занятий

№ п/п	Раздел дисциплины	Семестр	Виды учебной работы, включая самостоятельную работу студентов (СРС) и трудоемкость (в часах) очная/заочная форма обучения				СРС	Формы текущего контроля Форма промежуточной аттестации
			лекции	лабор. занятия, конс.	интерактивные формы занятий			
1.	Вводный пример		1/0,5	2/-	Интерактивная лекция-визуализация с групповой дискуссией	-/2	Оценка участия в дискуссии.	
2.	Система баз данных		2/0,5	2/2	Разбор конкретных ситуаций по группам	0,5/4	Оценивание работы в группах	
3.	Базы данных		2/0,5	4/-	Интерактивная лекция-визуализация с групповой дискуссией	0,5/4	Оценка участия в дискуссии.	
4.	Назначение баз данных		2/0,5	2/-		0,5/4	Блиц-опрос.	
5.	Независимость данных. Реляционные и другие системы.		1/1	2/-	Дерево решений	0,5/4	Оценивание работы	
6.	Три уровня архитектуры		2/1	2/2	Разбор конкретных ситуаций по группам	-/4	Оценивание работы в группах	
7.	Внешний уровень		1/1	2/-		0,5/4	Опрос.	
8.	Концептуальный уровень		1/0,5	2/-		0,5/2	Опрос	
9.	Внутренний уровень		1/0,5	2/-		0,5/2	Опрос	
10.	Отображения		1/-	2/2	Разбор конкретных ситуаций по группам	0,5/2	Оценивание работы в группах	
11.	Администратор базы данных		1/0,5	2/-		0,5/2	Опрос	
12.	Система управления базой данных		1/0,5	4/2	Разбор конкретных ситуаций по группам	0,5/1/2	Оценивание лабораторных работ.	
13.	Система управления передачей данных		1/0,5	4/2		0,5/2	Опрос	
14.	Реляционная модель		1/0,5	4/2	Разбор конкретных ситуаций по группам	0,5/2	Оценивание работы в группах	
						12/12	Подготовка ПР, к зачету/ экз,	
<i>Итого за 2 семестр/уст., 3</i>			<i>18/8</i>	<i>36/12</i>		<i>18/52</i>	<i>Зачет/Экзамен (36)</i>	
15.	Отношения и переменные-отношения		2/-	2/0,3		1/4	Опрос	
16.	Смысл отношений		2/-	2/0,3		1/4	Опрос	
17.	Оптимизация		2/-	2/0,4		1/4	Устный опрос.	
18.	Каталог		2/-	2/0,5	Дерево решений	1/4	Оценивание работы	
19.	Базовые переменные-отношения и представления		2/-	2/0,5	Дерево решений	2/4	Оценивание работы	
20.	Обзор языка SQL		2/-	2/0,5	Подготовка презентаций в малых группах	2/4	Оценивание работы в группах	
21.	Информационная схема		2/-	2/0,5		2/4	Опрос	
22.	Представления		2/-	2/0,5		2/4	Оценивание лабораторной работы	
23.	Транзакции. Внедрение SQL-		2/-	2/0,5	Разбор конкретных	2/4	Оценивание работы	



операторов				ситуаций по группам		в группах
					22/32	Подготовка к экзамену/зачету Курсовая работа.
<i>Итого за семестр 4</i>		<i>18/-</i>	<i>18/4</i>	<i>-</i>	<i>36/68</i>	<i>Экзамен (36) /зачет Курсовая</i>

Распределение компетенций по темам дисциплины

№ п/п	Раздел дисциплины	Освоенные компетенции
1.	Вводный пример	ПК-13, ПК-14
2.	Система баз данных	ПК-13, ПК-14
3.	Базы данных	ПК-13, ПК-14
4.	Назначение баз данных	ПК-13, ПК-14
5.	Независимость данных. Реляционные и др. системы.	ПК-13, ПК-14
6.	Три уровня архитектуры	ПК-13, ПК-14
7.	Внешний уровень	ПК-13, ПК-14
8.	Концептуальный уровень	ПК-13, ПК-14
9.	Внутренний уровень	ПК-13, ПК-14
10.	Отображения	ПК-13, ПК-14
11.	Администратор базы данных	ПК-13, ПК-14
12.	Система управления базой данных	ПК-13, ПК-14
13.	Система управления передачей данных	ПК-13, ПК-14
14.	Реляционная модель	ПК-13, ПК-14
15.	Отношения и переменные-отношения	ПК-13, ПК-14
16.	Смысл отношений	ПК-13, ПК-14
17.	Оптимизация	ПК-13, ПК-14
18.	Каталог	ПК-13, ПК-14
19.	Базовые переменные-отношения и представления	ПК-13, ПК-14
20.	Обзор языка SQL	ПК-13, ПК-14
21.	Каталог	ПК-13, ПК-14
22.	Представления	ПК-13, ПК-14
23.	Транзакции. Внедрение SQL-операторов	ПК-13, ПК-14

Методические указания для преподавателей

Рекомендуемые средства, методы обучения, способы учебной деятельности, применение которых для освоения конкретных модулей рабочей учебной программы наиболее эффективно:

– обучение теоретическому материалу рекомендуется основывать на основной и дополнительной литературе, изданных типографским или электронным способом конспектах лекций; рекомендуется в начале семестра ознакомить студентов с программой дисциплины, перечнем теоретических вопросов для текущего промежуточного и итогового контроля знаний, что ориентирует и поощрит студентов к активной самостоятельной работе;

- рекомендуется проводить лекционные занятия с использованием мультимедийной техники (проектора). На первом занятии до студентов должны быть доведены требования по освоению материала, правила написания и сдачи лабораторных работ, проверочных работ / индивидуальных заданий и курсовой работы, перечень рекомендуемой литературы. Желательно провести обзор тем, которые будут изучены в течение семестра с тем, чтобы студенты более осознанно подходили к выполнению работ. Также часть занятий проводятся в активной и интерактивной форме (в соответствии с ПО 07.08-13-2013 Интерактивное обучение).



Учебный процесс, опирающийся на использование интерактивных методов обучения, организуется с учетом включенности в процесс познания всех студентов группы без исключения. Совместная деятельность означает, что каждый вносит свой особый индивидуальный вклад, в ходе работы идет обмен знаниями, идеями, способами деятельности. Организуются индивидуальная, парная и групповая работа, используется проектная работа, ролевые игры, осуществляется работа с документами и различными источниками информации и т.д.

Интерактивные методы основаны на принципах взаимодействия, активности обучаемых, опоре на групповой опыт, обязательной обратной связи. Создается среда образовательного общения, которая характеризуется открытостью, взаимодействием участников, равенством их аргументов, накоплением совместного знания, возможностью взаимной оценки и контроля.

Лекционные занятия (краткий конспект)

Лекция 1. Вводный пример

Система баз данных — это, по сути, не что иное, как *компьютеризированная система хранения записей*. Саму же **базу данных** можно рассматривать как подобие электронной картотеки, т.е. хранилище или контейнер для некоторого набора занесенных в компьютер файлов данных. Пользователям этой системы предоставляется возможность выполнять множество различных операций над такими файлами:

- добавлять новые пустые файлы в базу данных;
- вставлять новые данные в существующие файлы;
- получать данные из существующих файлов;
- изменять данные в существующих файлах;
- удалять данные из существующих файлов;
- удалять существующие файлы из базы данных.

Лекция 2. Система баз данных

Как отмечалось выше, система баз данных — это компьютеризированная система хранения записей, т.е. компьютеризированная система, основное назначение которой — хранить информацию, предоставляя пользователям средства ее извлечения и модификации. К информации может относиться все, что заслуживает внимания отдельного пользователя или организации, использующей систему.

Данные.

■ Под понятием **интегрированности** данных подразумевается возможность представить базу данных как объединение нескольких отдельных файлов данных.

Аппаратное обеспечение

К аппаратному обеспечению системы относится следующее.

■ Тома вторичной (внешней) памяти (обычно это магнитные диски), используемые для хранения информации, а также соответствующие устройства ввода вывода (дисководы и т.п.), контроллеры устройств, каналы ввода вывода и т.д.

■ Аппаратный процессор (или процессоры) вместе с основной (первичной) памятью, предназначенные для поддержки работы программного обеспечения системы баз данных (подробности — в следующем разделе).

Программное обеспечение.

Между собственно физической базой данных (т.е. данными, которые реально хранятся) и пользователями системы располагается уровень программного обеспечения, который можно называть по-разному: **менеджер базы данных** (database manager), **сервер базы данных** (database server) или, что более привычно, **система управления базами данных**, СУБД (database management system — DBMS).

СУБД — это наиболее важный, но не единственный программный компонент системы.

Пользователи

■ прикладные программисты,



- конечные пользователи.

Лекция 3. Что такое база данных. Перманентные данные

Обычно данные в базе данных называют перманентными или постоянными (хотя на самом деле они могут недолго оставаться таковыми!). Под словом *перманентные* (persistent) подразумеваются данные, которые отличаются от других, более изменчивых данных, таких как промежуточные результаты, входные и выходные данные, управляющие операторы, рабочие очереди, программные управляющие блоки и вообще все временные (transient) по своей сути данные. Точнее говоря, можно утверждать, что данные в базе остаются "перманентными", поскольку после того, как они были приняты средствами СУБД для помещения в базу, *удалить их из нее впоследствии можно лишь с помощью соответствующего явного запроса к базе данных*, но не как результат какого либо побочного эффекта от выполнения некоторой программы. Подобный взгляд на понятие перманентности позволяет точнее определить термин "база данных".

База данных— это некоторый набор перманентных (постоянных) данных, используемых прикладными системами какого либо предприятия.

Сущности и связи.

Свойства.

Данные и модели данных.

Лекция 4. Назначение баз данных

Компактность. Нет необходимости в создании и ведении многотомных бумажных картотек.

Скорость. Компьютер может выбирать и обновлять данные гораздо быстрее человека. В частности, с его помощью можно быстро получать ответы на *произвольные* вопросы, возникающие в процессе работы (например, "Какого вина у нас сейчас больше — Zinfandel или Pinot Noir?"), не затрачивая времени на визуальный поиск или поиск вручную.

Низкие трудозатраты. Нет необходимости в утомительной работе над картотекой вручную. Механическую работу машины всегда выполняют лучше.

Актуальность. В случае необходимости под рукой в любой момент имеется точная свежая информация.

Администрирование данных и администрирование базы данных.

Преимущества централизованного подхода к управлению данными.

Возможность совместного доступа к данным.

Совместный доступ к данным означает не только возможность доступа к ним нескольких существующих приложений базы данных, но и возможность разработки новых приложений для работы с этими же данными. Другими словами, требования новых приложений по доступу к данным могут быть удовлетворены без необходимости добавления новых данных в базу.

Сокращение избыточности данных.

В системах, не использующих базы данных, каждое приложение имеет свои файлы. Это часто приводит к избыточности хранимых данных и, следовательно, к расточительству пространства вторичной памяти. Например, как приложение, связанное с учетом персонала, так и приложение, связанное с учетом обучения служащих, могут иметь собственные файлы с ведомственной информацией о служащих. Как отмечено выше, эти два файла можно объединить с устранением избыточности (одинаковой информации) при условии, что администратор данных знает о том, какие данные нужны для каждого приложения, т.е. что на предприятии осуществляется необходимое общее управление.

Устранение противоречивости данных (до некоторой степени)

Возможность поддержки транзакций

Транзакция (transaction) — это логическая единица работы, обычно включающая несколько операций базы данных (в частности, несколько операций изменения).

Обеспечение целостности данных



*Организация защиты данных
Возможность балансировки противоречивых требований
Возможность введения стандартизации*

Лекция 5. Независимость данных. Реляционные и другие системы.

Для разных приложений требуются разные представления одних и тех же данных.

Администратор базы данных должен иметь неограниченные возможности изменять физическое представление или метод доступа к данным в случае изменения требований, причем без необходимости модифицировать существующие приложения.

Хранимое поле — это наименьшая единица хранимых данных.

Хранимая запись — это набор связанных хранимых полей. И снова мы различаем для них тип и экземпляр.

Хранимый файл — это набор всех существующих в настоящий момент экземпляров хранимых записей одного и того же типа.

Представление числовых данных

Представление символьных данных

Единицы измерения для числовых данных

Кодирование данных

Овеществление данных

Структура хранимых записей

Структура хранимых файлов

Реляционные и другие системы

Лекция 6. Три уровня архитектуры

Архитектура ANSI/SPARC включает три уровня: внутренний, внешний и концептуальный. В общих чертах они представляют собой следующее.

Внутренний уровень (также называемый *физическим*) наиболее близок к физическому хранилищу информации, т.е. связан со способами сохранения информации на физических устройствах.

Внешний уровень (также называемый *пользовательским логическим*) наиболее близок к пользователям, т.е. связан со способами представления данных для отдельных пользователей.

Концептуальный уровень (также называемый *общим логическим* или просто *логическим*) является "промежуточным" уровнем между двумя первыми.

Во-первых, концептуальный уровень в такой системе определенно будет реляционным в том смысле, что видимые на этом уровне объекты являются реляционными таблицами, а используемые операторы будут реляционными операторами.

Во-вторых, каждое внешнее представление, как правило, также будет реляционным или достаточно близким к нему. Например, объявления записей в PL/I и COBOL, представленные на рис. 2.2, упрощенно можно считать аналогами объявления реляционной таблицы в реляционной системе.

В-третьих, внутренний уровень *не* будет реляционным, поскольку объекты на этом уровне не будут реляционными таблицами (сохраняемыми); наоборот, они будут объектами такого же типа, как и находящиеся на внутреннем уровне объекты любой другой системы (сохраняемые записи, указатели, индексы, хешированные значения и т.п.). В действительности реляционная модель как таковая не может сказать ничего существенного о внутреннем уровне.

Лекция 7. Внешний уровень

Внешний уровень — это индивидуальный уровень пользователя, пользователь может быть прикладным программистом или конечным пользователем с любым уровнем профессиональной подготовки. (Особое место среди пользователей занимает администратор базы данных (АБД). В отличие от остальных пользователей, АБД интересуется также концептуальный и внутренний уровни. Об этом еще будет говориться в следующих двух разделах.)



У каждого пользователя есть свой язык общения.

Для прикладного программиста это либо один из распространенных языков программирования (например, PL/I, C++ или Java), либо специальный язык рассматриваемой системы. Такие оригинальные языки называют языками четвертого поколения на том (не вполне определенном!) основании, что машинный код, языка ссемблера и такие языки, как PL/I, можно считать языками трех первых "поколений", а оригинальные языки модернизированы по сравнению с языками третьего поколения так же, как языки третьего поколения улучшены по сравнению с языком ассемблера.

Для конечного пользователя это или специальный язык запросов, или язык специального назначения, который может быть основан на использовании форм и меню, разработан специально с учетом требований пользователя и интерактивно поддерживаться некоторым оперативным приложением.

Язык определения данных включает некоторые описательные структуры языка PL/I, необходимые для объявления объектов базы данных. Это сам оператор DECLARE (DCL), определенные типы данных языка PL/I, а также возможные специальные дополнения для языка PL/I, предназначенные для поддержки новых объектов, которые не поддерживаются существующей версией языка PL/I.

Лекция 8. Концептуальный уровень

Концептуальное представление — это представление всей информации базы данных в несколько более абстрактной форме (как и в случае внешнего представления) по сравнению с физическим способом хранения данных. Однако концептуальное представление существенно отличается от представления данных каким либо отдельным пользователем. Вообще говоря, концептуальное представление — это представление данных такими, какими они являются на самом деле, а не такими, какими их вынужден видеть пользователь в рамках, например, определенного языка или используемого аппаратного обеспечения.

Лекция 9. Внутренний уровень

Третьим уровнем архитектуры является внутренний уровень. **Внутреннее представление** — это низкоуровневое представление всей базы данных как базы, состоящей из некоторого множества экземпляров каждого из существующих типов **внутренних записей**. Термин "внутренняя запись" относится к терминологии ANSI/SPARC и означает конструкцию, иначе называемую *хранимой* записью (в дальнейшем мы будем использовать именно этот термин). Внутреннее представление, так же как внешнее и концептуальное, отделено от физического уровня, поскольку в нем не рассматриваются физические записи, обычно называемые **блоками** или **страницами**, и физические области устройства хранения, такие как цилиндры и дорожки. Другими словами, внутреннее представление предполагает наличие бесконечного линейного адресного пространства. Особенности методов отображения этого адресного пространства на физические устройства хранения в значительной степени зависят от используемой операционной системы и **по** этой причине не включены в общую архитектуру.

Лекция 10. Отображения

Отображение "*концептуальный внутренний*" устанавливает соответствие между концептуальным представлением и хранимой базой данных, т.е. описывает, как концептуальные записи и поля представлены на внутреннем уровне. При изменении структуры хранимой базы данных (т.е. при внесении изменений в определение структуры хранения) отображение "*концептуальный внутренний*" также изменяется, причем таким образом, чтобы концептуальная схема осталась неизменной. (Выполнение подобных изменений входит в обязанности администратора базы данных.) Иначе говоря, чтобы обеспечить независимость данных, результаты внесения любых изменений в схему хранения не должны быть видны на концептуальном уровне.



Отображение "внешний концептуальный" определяет соответствие между некоторым внешним представлением и концептуальным представлением. В целом, различия, которые могут существовать между этими двумя уровнями, подобны различиям между концептуальным представлением и хранимой базой данных. Например, данные полей могут быть разных типов, названия полей и записей могут быть изменены, несколько концептуальных полей могут быть объединены в одно (виртуальное) внешнее поле и т.д. В одно и то же время может существовать любое количество внешних представлений, причем одно и то же внешнее представление может принадлежать нескольким пользователям, а разные внешние представления могут перекрывать-ся.

Лекция 11. *Администратор базы данных*

Администратор *данных* (АД) — это человек, отвечающий за стратегию и политику принятия решений, связанных с данными предприятия, администратор *базы данных* (АБД) — это человек, обеспечивающий необходимую техническую поддержку с целью реализации принятых решений. Таким образом, АБД отвечает за общее управление системой на техническом уровне.

Определение концептуальной схемы.

Определение внутренней схемы.

Взаимодействие с пользователями.

Определение процедур резервного копирования и восстановления.

Управление производительностью и реагирование на изменяющиеся требования.

Лекция 12. *Система управления базой данных*

1. Пользователь выдает запрос на доступ к данным, применяя определенный подязык данных (обычно это язык SQL).

2. СУБД перехватывает этот запрос и анализирует его.

3. СУБД просматривает внешнюю схему (ее объектную версию) для этого пользователя, соответствующее отображение "внешний концептуальный", концептуальную схему, отображение "концептуальный внутренний" и определение структуры хранения.

4. СУБД выполняет необходимые операции в хранимой базе данных.

Обработка данных.

СУБД должна уметь обрабатывать запросы пользователя на выборку, изменение или удаление данных, уже существующих в базе, или на добавление в нее новых данных. Другими словами, СУБД должна включать в себя компонент процессора **ЯМЛ** или компилятора **ЯМЛ**, обеспечивающего поддержку языка манипулирования данными.

а) Планируемый запрос — это запрос, необходимость выполнения которого была предусмотрена заранее. Администратор базы данных, возможно, должен будет построить физический проект базы данных таким образом, чтобы гарантировать достаточное быстродействие выполнения подобных запросов.

б) Не планируемый запрос — это, наоборот, некоторый произвольный запрос, необходимость выполнения которого не была предусмотрена заранее и возникла по какой то особой причине. Физический проект базы данных может подходить, а может и не подходить для выполнения конкретного произвольного запроса.

Оптимизация и выполнение.

Защита и сохранение целостности данных.

Восстановление данных и поддержка параллельности.

Словарь данных.

Производительность.

Система управления файлами ничего не знает о внутренней структуре хранимых записей и, следовательно, не способна обрабатывать запросы, предполагающие знание этой структуры.

Как правило, эти системы предоставляют слабую поддержку ограничений защиты и поддержки целостности данных или же не предоставляют ее вовсе.



Как правило, эти системы предоставляют недостаточную поддержку управления восстановлением данных и параллельным доступом к ним или же не предоставляют ее вовсе.

На уровне менеджера файлов не существует концепции настоящего словаря данных.

Как правило, отдельные файлы не "интегрированы" или не "разделяемы" в том смысле, который вкладывается в эти понятия применительно к базам данных. (Обычно они являются собственными файлами пользователя или приложения.)

Лекция 13. Система управления передачей данных

В этом разделе рассматривается передача данных. Запросы к базе данных от конечных пользователей в действительности передаются от рабочей станции пользователя (которая может быть физически удалена от самой системы баз данных) к некоторому интерактивному приложению (встроенному или нет), а от него — к СУБД в форме *коммуникационных сообщений*. Более того, ответы пользователю (от СУБД и оперативного приложения к станции пользователя) также передаются в форме подобных сообщений. Передача таких сообщений происходит под управлением другого программного компонента — менеджера передачи данных.

Менеджер передачи данных не является частью СУБД; он представляет собой автономную систему со своими правами. Однако, поскольку очевидно, что от менеджера передачи данных и СУБД требуется согласованная совместная работа, они иногда рассматриваются как равноправные партнеры в компоненте более высокого уровня, называемого системой базы данных и передачи данных. В этой системе СУБД отвечает за работу с базой данных, а менеджер передачи данных обрабатывает все сообщения, поступающие в СУБД и из нее, а точнее говоря, в то приложение, которое использует СУБД, и из него. Однако в этой книге об обработке сообщений нам осталось сказать относительно немного (поскольку такая обширная тема вполне заслуживает самостоятельного обсуждения).

Лекция 14. Реляционная модель

Операция выборки SELECT (или RESTRICT) предназначена для извлечения определенных строк из таблицы.

Операция проекции PROJECT предназначена для извлечения определенных столбцов из таблицы.

Операция объединения JOIN предназначена для соединения двух таблиц на основе общих значений в общих столбцах.

Во первых, отметим, что определение реляционной системы требует, чтобы база данных только *воспринималась пользователем* как набор таблиц. Таблицы в реляционной системе являются логическими, а не физическими структурами. На самом деле на физическом уровне система может использовать любую из существующих структур памяти (последовательный файл, индексирование, хеширование, цепочку указателей, сжатие и т.п.), лишь бы существовала возможность отображать эти структуры в виде таблиц на логическом уровне. Данное положение можно сформулировать и по другому: таблицы представляют собой *абстракцию* способа физического хранения данных, в которой множество деталей на уровне памяти (размещение хранимых записей, последовательность хранимых записей, кодировка хранимых данных, префиксы хранимых записей, хранимые структуры доступа, такие как индексы, и т.д.) *скрыто от пользователя*.

Во вторых, у реляционных баз данных есть одно замечательное свойство, называемое **информационным принципом**: *все информационное содержимое базы данных представлено одним и только одним способом, а именно — явным зада ниш значений, помещенных в позиции столбцов в строках таблицы*. Этот метод представления *единственно* возможный для реляционных баз данных (естественно, на логическом уровне). В частности, **нет никаких указателей**, связывающих одну таблицу с другой. Например, на рис. 3.1 существует определенная связь между строкой 'D1' в таблице DEPT и строкой 'E1' в таблице EMP, указывающая, что служащий с



номером 'E1' работает в отделе с номером 'D1'; но эта информация представлена не указателем, а наличием значения 'D1' в столбце DEPT строки'E1' в таблице EMP.

Лекция 15. Отношения и переменные-отношения

Принципы реляционной модели были изначально заложены в 1969 и 1970 годах Е.Ф. Коддом (E.F. Codd), который в то время был исследователем, работавшим в корпорации IBM. В конце 1968 года Кодд, математик по образованию, в первые пришел к заключению, что математические дисциплины можно использовать для привнесения в область управления базами данных строгих принципов и точности. Именно этих качеств не доставало данной области в то время.

С того времени эти идеи стали общепринятыми и оказали весьма существенное влияние на все аспекты технологии баз данных, а также на другие области, такие как искусственный интеллект, обработка естественных языков и разработка аппаратных систем.

Лекция 16. Смысл отношений

Во-первых, данное отношение ρ и заголовок отношения ρ представляют определенный предикат или логическую функцию.

Во-вторых, как упоминалось в главе 1, каждая строка в теле отношения ρ представляет собой определенное истинное высказывание, полученное из предиката путем подстановки определенных значений *аргументов* соответствующего типа вместо *местодержателей* или *параметров* этого предиката (реализация предиката).

Типы — это объекты (множества объектов), которые можно обсуждать;

Отношения — это факты (множества фактов), касающиеся объектов, которые можно обсуждать.

DEPT: "Отдел с номером $DEPT_i$ называется $DNAME$ и имеет бюджет $BUDGET$ ".

Выборка строк из DEPT, где $BUDGET > 8M$: "Отдел с номером $DEPT_i$ называется $DNAME$ и имеет бюджет $BUDGET$, который больше восьми миллионов долларов".

Извлечение столбцов $DEPT_t$ и $BUDGET$ из $DEPT$: "Отдел с номером $DEPT_f$ имеет какое то название и бюджет $BUDGET$ ".

Соединение переменных отношений $DEPT$ и EMP на основе столбца $DEPT_i$: "Отдел с номером $DEPT_t$ называется $DNAME$ и имеет бюджет $BUDGET$, а служащий с номером EMP_t по фамилии $ENAME$ работает в отделе с номером $DEPT_i$ и получает зарплату $SALARY$ " (заметим, что этот предикат имеет шесть параметров, а не семь, поскольку две ссылки на $DEPT_t$ обозначают один и тот же параметр).

Лекция 17. Оптимизация

Даже в этом простом примере существует по крайней мере два способа поиска необходимых данных.

1. Последовательный физический просмотр (хранимой версии) переменной отношения EMP, пока требуемая запись не будет найдена.

2. Если есть индекс для столбца EMP_i (в хранимой версии) переменной отношения (который, вероятно, действительно существует, поскольку этот столбец является уникальным, а большинство систем фактически *требует* создания индекса для обеспечения уникальности), то переход с помощью этого индекса непосредственно к данным служащего с номером 'E4'.

Оптимизатор выберет, какую из двух возможных стратегий следует применить. В общем случае для реализации любого конкретного реляционного запроса оптимизатор будет осуществлять выбор стратегии, исходя из соображений, подобных следующим:

на какие переменные отношения есть ссылки в запросе;

насколько велики эти переменные отношения;

какие существуют индексы;

насколько избирательны эти индексы;



как физически группируются данные на диске;
какие реляционные операции используются и т.д.

Лекция 18. Каталог

Каталог обычно размещается там, где сохраняются различные схемы (внешние, концептуальные, внутренние) и все, что относится к отображениям ("внешний концептуальный", "концептуальный внутренний"). Иначе говоря, в каталоге содержится детальная информация (иногда называемая метаданными), касающаяся различных объектов, которые имеют значение для самой системы. Примерами таких объектов могут служить переменные отношения, индексы, ограничения поддержки целостности, ограничения защиты и т.д. Метаданные необходимы для обеспечения правильной работы системы. Например, оптимизатор использует информацию каталога об индексах и других физических структурах хранения данных, а также иную информацию, необходимую ему для принятия решения о том, каким образом выполнить тот или иной запрос пользователя.

Замечательным свойством реляционных систем является то, что их *каталог также состоит из переменных отношений* (точнее, из *системных* переменных отношений, названных так для того, чтобы отличать их от обычных пользовательских). В результате пользователь может обращаться к каталогу так же, как к своим данным. Например, в каталоге обычно содержатся системные TABLES и COLUMNS, назначение которых — описание известных системе таблиц (т.е. переменных отношений) и столбцов этих таблиц. (Мы говорим "обычно", потому что каталоги в различных системах разные, так как существенная часть информации каталога является специфической для конкретной системы.)

Лекция 19. Базовые переменные-отношения и представления

Исходные (заданные) переменные отношения называются **базовыми переменными отношениями**, а присвоенные им значения называются **базовыми отношениями**. Отношение, которое получено или может быть получено из базового отношения в результате выполнения каких либо реляционных выражений, называется **производным** отношением.

Базовые переменные отношения "реально существуют" в том смысле, что они представляют данные, которые действительно хранятся в базе данных.

Представления, наоборот, "реально не существуют", а просто предоставляют различные способы просмотра "реальных" данных.

Лекция 20. Обзор языка SQL

В языке SQL имеются операции как определения данных, так и манипулирования ими.

В языке SQL поддерживается множество значений, принимаемых по умолчанию, сокращений и альтернативных написаний. Например, существует сокращение CHAR для определителя CHARACTER. Квадратные скобки "[" и "]" в принятой нотации используются для указания необязательных элементов (например, CHARACTER или BIT), при отсутствии которых будут использованы значения по умолчанию. (Это правило обычно подразумевается при описании в форме Бэкуса Наура, сокращенно — BNF).

Определив базу данных, можно начинать выполнять в ней различные операции, задаваемые с помощью операторов манипулирования данными языка SQL: SELECT, INSERT, UPDATE и DELETE. В частности, можно выполнять с данными реляционные операции выборки, проекции и соединения, причем во всех этих случаях следует использовать один и тот же оператор манипулирования данными языка SQL SELECT.

Лекция 21. Информационная схема

В языке SQL существует аналог того, что принято называть каталогом, — **информационная схема**. Знакомые нам термины "каталог" и "схема" действительно используются в языке SQL, но с особым, специфическим только для языка SQL смыслом. Нестрого говоря, **каталог** в



языке SQL состоит из дескрипторов (метаданных) для отдельной базы данных, а **схема** состоит из дескрипторов той части базы данных, которая принадлежит отдельному пользователю. Другими словами, в системе может быть любое число каталогов (по одному для каждой базы данных), каждый из которых делится на произвольное число схем. Однако каждый каталог должен содержать ровно одну схему с именем INFORMATION SCHEMA (информационная схема), которая, с точки зрения пользователя, и является схемой (как уже указывалось), выполняющей функции обычного каталога.

Таким образом, информационная схема состоит из набора SQL таблиц, содержимое которых фактически отражает (точно определенным образом) все определения из всех остальных схем рассматриваемого каталога. Точнее говоря, информационная схема по определению содержит набор представлений гипотетической "схемы определения". Для поддержки схемы определения реализация не требуется, но она требуется, во первых, для поддержки *некоторого* вида "схемы определения" и, во вторых, для поддержки представлений такой "схемы определения", которые имеют вид, подобный представлениям информационной схемы. Необходимо отметить следующие моменты.

1. Основная причина того, что требования состоят из двух отдельных требований, сформулированных выше, заключается в следующем. Существующие продукты, конечно, поддерживают нечто близкое к "схеме определения". Однако от одного продукта к другому диапазон различий таких схем будет очень широким (даже если эти продукты имеют одного изготовителя). Поэтому требование лишь того, чтобы в реализации поддерживались предопределенные представления "схемы определения", имеет смысл.

2. На самом деле нужно говорить "одна из информационных схем", а не просто "информационная схема", поскольку, как мы убедились, в каждом каталоге имеется такая схема. Поэтому в общем случае все данные, которые доступны некоторому пользователю, *не* будут описаны с помощью единственной информационной схемы. Однако, чтобы упростить изложение материала, мы будем продолжать считать, что в действительности существует только одна схема.

Лекция 22. Представления

Приведем пример определения представления на языке SQL.

```
CREATE VIEW GOOD SUPPLIER
AS SELECT Si, STATUS, CITY FROM S
WHERE STATUS > 15 ;
```

Пример SQL-запроса к этому представлению.

```
SELECT Si, STATUS FROM GOOD_SUPPLIER WHERE CITY = 'London' ;
```

Подставив определение представления вместо ссылки на имя представления, получим выражение, которое будет подобно приведенному ниже (обратите внимание на **вложенный подзапрос** в предложении FROM),

```
SELECT GOOD_SUPPLIER.Sf, GOOD_SUPPLIER.STATUS FROM ( SELECT St, STATUS, CITY FROM S
WHERE STATUS > 15 ) AS GOOD_SUPPLIER WHERE GOOD_SUPPLIER.CITY = 'London' ;
```

Это выражение может быть затем упрощено, например, так.

```
SELECT St, STATUS FROM S
WHERE STATUS > 15
AND CITY = 'London' ;
```

В последнем случае показан текст запроса, который фактически будет выполняться. В качестве второго примера рассмотрим следующую операцию DELETE.

```
FROM GOOD_SUPPLIER WHERE CITY ~='London' ;
```

Запрос на удаление, который будет выполняться **на самом деле**, выглядит так.

```
DELETE FROM S
WHERE STATUS > 15
```



AND CITY = 'London' ;

Лекция 23. Транзакции. Внедрение SQL-операторов

Для операторов COMMIT и ROLLBACK в языке SQL есть прямые аналоги. Это операторы COMMIT WORK и ROLLBACK WORK соответственно (в обоих операторах слово WORK необязательное). Но в языке SQL нет явного оператора, соответствующего оператору BEGINTRANSACTION. Неявно транзакция начинается всякий раз, когда программа выполняет оператор, способный "инициализировать транзакцию" (transactioninitiating), но только в том случае, когда никакая транзакция еще не выполняется. Рассматривать здесь список SQL операторов, способных инициализировать транзакцию, в наши планы не входит. Достаточно сказать, что таковыми являются практически все операторы, которые мы обсуждаем в этой главе (кроме, конечно, самих операторов COMMIT и ROLLBACK).

1. Внедренные SQL-операторы предваряются инструкцией EXEC SQL, так что их легко отличить от других операторов базового языка, и заканчиваются специальным завершающим символом (точка с запятой ";" для языка PL/I).

2. *Выполняемый* SQL-оператор.

3. SQL-операторы могут включать ссылки на базовые переменные (т.е. переменные базового языка).

4. Обратите внимание на предложение INTO оператора SELECT, представленного на рис. 4.3. Назначение этого предложения (как только что отмечалось) — указать результирующие (целевые) переменные, в которые будут возвращены выбранные значения. Каждая /-я целевая переменная, указанная в предложении INTO, соответствует извлекаемому значению, указанному в списке выборки предложения SELECT.

5. Все базовые переменные, на которые ссылаются внедренные SQL-операторы, должны быть определены (в PL/I это оператор DCL) в разделе объявлений внедренного языка SQL, который ограничивается операторами BEGIN DECLARESECTION и END DECLARE SECTION.

6. Каждая программа, содержащая внедренные SQL-операторы, должна включать базовую переменную с именем SQLSTATE.

7. Базовые переменные должны иметь типы данных, соответствующие значениям, для размещения которых эти переменные используются.

8. Базовые переменные для столбцов SQL-таблиц могут иметь те же имена, что и имена соответствующих столбцов.

9. Как уже упоминалось, выполнение каждого SQL-оператора, в принципе, должно сопровождаться проверкой значения, возвращаемого в переменной SQLSTATE. Для упрощения этого процесса предназначен оператор WHENEVER.

EXEC SQL WHENEVER <условие> <действие> ;

Лабораторные занятия, консультации

Лабораторная работа №1 - Система баз данных.

Информация. Данные. Интегрированность данных. Аппаратное обеспечение. Тома вторичной (внешней) памяти. Аппаратный процессор. Программное обеспечение. Менеджер базы данных. Сервер базы данных. Система управления базами данных. DBMS. Пользователи. Прикладные программисты. Конечные пользователи.

Лабораторная работа №2 - Три уровня архитектуры.

Архитектура ANSI/SPARC. Уровень внутренний. Уровень внешний. Уровень концептуальный. Уровень физическим. Уровень пользовательский. Уровень логически. Таблицы. а используемые операторы будут реляционными операторами.

**Лабораторная работа №3 – Отображения.**

Соответствие между концептуальным представлением и хранимой базой данных. Изменение структуры хранимой базы данных. Независимость данных. Внесения изменений в схему данных. Соответствие между представлением. Внешние представления.

Лабораторная работа №4 - Система управления базой данных.

Запрос на доступ к данным. Язык данных. Язык SQL. Перехват запроса. Просмотр внешней схемы. Концептуальная схема. Структура хранения. Хранимая база данных.

Лабораторная работа №5 - Обзор языка SQL.

Манипулирования данными. Значения, принимаемые по умолчанию. Сокращения. Форма Бэкуса Наура (BNF). SELECT, INSERT, UPDATE и DELETE. Выборка, проекция и соединение. Команды администрирования. Команды управления. Команды доступа к данным.

Лабораторная работа №6 - Транзакции. Внедрение SQL-операторов

COMMIT. ROLLBACK. COMMIT WORK. ROLLBACK WORK. BEGIN TRANSACTION. Неявно транзакция начинается. Инициализировать транзакцию. Список SQL операторов. Внедренные SQL-операторы. EXEC SQL. Языка PL/SQL). Выполняемый SQL-оператор. Предложение INTO оператора SELECT. Базовые переменные. BEGIN DECLARE SECTION и END DECLARE SECTION. Базовая переменная с именем SQLSTATE. Базовые переменные для столбцов SQL-таблиц. WHENEVER. EXEC SQL WHENEVER <условие> <действие>.

Лабораторный практикум, темы для консультаций**1. Методы управления базами данных***Вопросы для обсуждения*

1. Каковы преимущества использования системы баз данных?
2. Каковы недостатки использования системы баз данных?
3. Как вы понимаете термин *реляционная система*. Назовите различия между реляционной и не реляционными системами.
4. Как вы понимаете термин *модель данных*? Объясните различие между моделью данных и ее реализацией. Почему так важно это различие?
5. Напишите SQL-оператор для выполнения следующих операций в базе данных винного погреба.
 - а) Выберите номер ячейки, наименование вина и количество бутылок для всех вин производства 'Geyser Peak'.
 - б) Выберите номер ячейки (BIN) и наименование вина для всех вин, запас которых составляет более пяти бутылок.
 - в) Выберите номер ячейки (BIN) для всех красных вин.
 - г) Добавьте три бутылки (BOTTLES) в ячейку (BIN) с номером 30.
 - д) Удалите из всего запаса все вина производства компании 'Chardonnay'.
 - е) Добавьте данные нового поступления (12 бутылок): производитель— 'Gary Farrell', сорт— 'Merlot', ячейка номер 55, год выпуска— 1996, будет готово в 2001 году.

2. Архитектура системы баз данных*Вопросы для обсуждения*

1. Опишите последовательность шагов, применяемых при выборке определенного экземпляра внешней записи.
2. Перечислите главные функции, выполняемые СУБД.
3. Укажите различия между логической и физической независимостью от данных.
4. Как вы понимаете термин *метаданные*?
5. Перечислите главные функции, выполняемые АБД.



6. Укажите различия между СУБД и системой управления файлами.
7. Приведите несколько примеров инструментальных средств, предоставляемых различными поставщиками.
8. Приведите несколько примеров утилит базы данных.
9. Проанализируйте любую доступную вам систему баз данных. Попытайтесь представить ее в соответствии с архитектурой ANSI/SPARC, как описано в этой главе. Полностью ли она поддерживает три уровня архитектуры? Как определены отображения между уровнями? С чем схожи различные языки определения данных (внешний, концептуальный и внутренний)? Какой подязык данных поддерживает система? Какой язык является базовым? Кто выполняет функции АБД? Имеются ли какие-нибудь средства организации защиты и поддержания целостности данных? Существует ли в системе словарь? Описывает ли он сам себя? Какие приложения, предоставляемые поставщиками, поддерживает система? Какие утилиты входят в состав системы? Есть ли в системе отдельный компонент менеджера передачи данных? Имеются ли какие-либо возможности распределенной обработки?

3. Введение в реляционные базы данных

Вопросы для обсуждения

1. Опишите содержимое переменных-отношений каталога TABLE и COLUMN для базы данных поставщиков и деталей.
2. Вот запрос для базы данных поставщиков и деталей.
`RESULT := ((S JOIN SP) WHERE Pi = 'P2') { Si, CITY } ;`
 Что получится в результате его выполнения?
3. Предположим, что выражение, используется для определения представления.
`CREATE VIEW V AS
 ((S JOIN SP) WHERE Pi = 'P2') { Si, CITY } ;`
 Теперь рассмотрим следующий запрос.
`RESULT := { V WHERE CITY = 'London' } { Si } ;`
 Что получится в результате его выполнения? Поясните, какой компонент используется со стороны СУБД при выполнении запроса.
4. Как вы понимаете термины, характеризующие свойства транзакций: атомарность, продолжительность, изолированность и упорядоченность выполнения операций параллельных транзакций.

4. Введение в язык SQL

Вопросы для обсуждения

1. Ниже в таблице показаны примеры значений данных для расширенной формы базы данных поставщиков и деталей, которая называется базой данных поставщиков, деталей и проектов. Поставщики (S), детали (P) и проекты (J) однозначно определяются номером поставщика (S), номером детали (P) и номером проекта (J) соответственно. Значение строки SPJ (поставки) следующее: определенный поставщик поставляет определенную деталь для определенного проекта в определенном количестве (причем комбинация значений столбцов S#-P#-J# уникальна для отдельных строк). Запишите соответствующие определения данных на языке SQL для этой базы данных.

База данных поставщиков, деталей и проектов (значения для примера)

S	s#	SNAM	STA-	CITY	SPJ	s#	P#	J#	QTY
		E	TUS						
	SI	Smith	20	London		SI	PI	J1	200
	S2	Jones	10	Paris		SI	PI	J4	700
	S3	Black	30	Paris		S2	P3	J1	400



	S4	Clark	20	London		S2	P3	J2	200
	S5	Adams	30	Athens		S2	P3	J3	200
p						S2	P3	J4	500
						S2	P3	J5	600
	P#	PNAME	COLOR	WEIGHT	CITY	S2	P3	J6	400
	P1	Nut	Red	12.0	London	S2	P3	J7	800
	P2	Bolt	Green	17.0	Paris	S2	P5	J2	100
	P3	Screw	Blue	17.0	Rome	S3	P3	J1	200
	P4	Screw	Red	14.0	London	S3	P4	J2	500
J	P5	Cam	Blue	12.0	Paris	S4	P6	J3	300
	P6	Cog	Red	19.0	London	S4	P6	J7	300
	J#	JNAME	CITY			S5	P2	J2	200
	J1	Sorter	Paris			S5	P2	J4	100
	J2	Display	Rome			S5	P5	J5	500
	J3	OCR	Athens			S5	P5	J7	100
	J4	Console	Athens			S5	P6	J2	200
	J5	RAID	London			S5	PI	J4	100
	J6	EDS	Oslo			S5	P3	J4	200
	J7	Tape	London			S5	P4	J4	800
					S5	P5	J4	400	
					S5	P6	J4	500	

2. И снова исследуйте доступный вам продукт, поддерживающий язык SQL. Поддерживается ли в нем информационная схема? Если нет, то каким образом поддерживается каталог?

3. Сформулируйте на языке SQL следующие операции обновления для базы данных поставщиков, деталей и проектов.

а) Вставить нового поставщика 'S10' в таблицу S; имя поставщика— 'Smith', город— 'New York', статус еще неизвестен.

б) Изменить цвет всех красных деталей ('red') на оранжевый ('orange').

в) Удалить все проекты, для которых нет поставок.

4. Используя базу данных поставщиков, деталей и проектов, напишите программу с вложенными SQL-выражениями для выдачи списка всех строк поставщиков по порядку их номеров. За каждой строкой поставщика должны непосредственно следовать строки проектов, обеспечиваемых этим поставщиком, по порядку номеров проектов.

5. Даны таблицы PART и PART_STRUCTURE, определенные таким образом.

```
CREATE TABLE PART
(Pi ... , DESCRIPTION ... , PRIMARY KEY ( Pi ) ) ; CREATE TABLE
PART_STRUCTURE
(MAJOR Pi .7 , MINOR Pi ... , QTY ... , PRIMARY KEY ( MAJOR Pi , MINOR Pi ), FOREIGN KEY ( MAJOR~Pi ) REFERENCES PART, FOREIGN KEY ( MINOR~Pi ) REFERENCES PART ) ;
```

В таблице PART_STRUCTURE показано, какие детали (MAJOR_Pi) содержат другие детали (MINOR_Pi) как компоненты первого уровня. Напишите программу на языке SQL для получения списка всех компонентов данной детали на всех имеющихся уровнях (задача разувложения деталей).

Таблица PART STRUCTURE (значения для примера)

MAJOR P#	MINOR P#	QTY
----------	----------	-----



P1	P2		М	С
P1	P3	4		
P2	P3	1		
P2	P4		О	С
P3	P5	9		
P4	P5	8		
P5	P6	3		

6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

Рекомендуемый режим и характер различных видов учебной, в том числе самостоятельной, работы:

– изучение теоретического материала определяется рабочей учебной программой дисциплины, включенными в нее календарным планом изучения дисциплины и перечнем литературы; настоятельно рекомендуется при подготовке к очередной лекции освежить в памяти, по указанию лектора, материал предшествующих дисциплин рабочего учебного плана, на который опирается изучаемый раздел данной дисциплины;

– проверочная работа / индивидуальное задание (ПР/ИЗ) и курсовая работа выполняются в соответствии с изданными типографским или электронным способом методическими указаниями, регламентирующими все этапы выполнения и сдачи работ, определяют свой вклад в рейтинговую оценку;

Планирование времени на самостоятельную работу, необходимого на изучение настоящей дисциплины, лучше всего осуществлять на весь семестр (в соответствии с ПО 07.08-12-2013 Организация самостоятельной работы студентов), предусматривая при этом регулярное повторение пройденного материала. Материал, законспектированный на лекциях, необходимо регулярно дополнять сведениями из литературных источников, представленных в списке рекомендуемой литературы. По каждой из тем для самостоятельного изучения, приведенных в программе дисциплины, следует сначала прочитать рекомендованную литературу и при необходимости составить краткий конспект основных положений, терминов, сведений, требующих запоминания и являющихся основополагающими в этой теме и нужных для освоения последующих разделов.

Для расширения знаний по дисциплине рекомендуется использовать Интернет-ресурсы: проводить поиск в различных поисковых системах, таких как www.rambler.ru, www.yandex.ru, www.google.ru, www.yahoo.ru и использовать материалы сайтов, рекомендованных преподавателем на лекционных занятиях.

При подготовке к зачету/экзамену следует руководствоваться перечнем вопросов для подготовки к промежуточному контролю. При этом, прежде всего, следует уяснить суть основных понятий дисциплины, проработать учебные материалы основной и дополнительной литературы, а также литературы из электронно-библиотечной системы, рекомендованных для изучения дисциплины.

Распределение времени на самостоятельную работу студента

№ п/п	Вид самостоятельной работы	Количество времени (часы) очная/заочная форма обучения
2/Уст., 3 семестр		
1	Проработка материала лекций, учебных материалов. Самостоятельная проработка тем	2/18
2	Подготовка к лабораторным занятиям, консультациям. Самостоятельная проработка тем	4/22



3	Подготовка к выполнению ПР/ИЗ	4/4
4	Подготовка к зачету/ экзамену	8/8
	Итого	18/52
3/4 семестр		
1	Проработка материала лекций, учебных материалов. Самостоятельная проработка тем	6/14
2	Подготовка к лабораторным занятиям, консультациям. Самостоятельная проработка тем	8/22
3	Подготовка к выполнению курсовой работы	14/24
4	Подготовка к экзамену/зачету	8/8
	Итого	36/68

Вопросы для самостоятельной работы студентов

1. Языковые средства современных СУБД.
2. Основы реляционной алгебры.
3. Реляционное исчисление.
4. Оптимизация реляционных выражений.
5. Язык запросов SQL.
6. Доступ по первичным ключам.
7. Хешированные файлы.
8. Организация индексно-последовательных файлов.
9. Организация индексных таблиц в виде иерархических структур.
10. Бинарные деревья.
11. В-деревья.
12. Доступ по вторичным ключам. Инвертированные списки.
13. Использование систем управления базами данных.
14. СУБД MS Access и ее возможности.
15. Проверка правильности введенного в поля значения.
16. Формирование текстового представления поля.
17. Открытие и закрытие набора данных.
18. Программный доступ к записям.

Образовательные технологии

В ходе освоения дисциплины в аудиторной работе (лекции, лабораторные работы, консультации) используются следующие образовательные технологии с применением интерактивных методов обучения:

– компьютерные технологии: сопровождение лекций демонстрацией слайдов из интернета, подготовленных преподавателем и студентами (в качестве домашнего задания) по обозначенной тематике, а также при проведении лабораторных работ – для показа подготовленных студентами презентаций;

– метод проблемного (дискуссионного) и диалогового изложения лекций;

– метод решения и обсуждения конкретных ситуаций (презентаций) в ходе лекций и лабораторных работ.

Внеаудиторная работа (ПР/индивидуальное задание) оценивается на практическом занятии или консультации.

Предполагается возможность внеаудиторных он-лайн коммуникаций преподавателя со студентами, а также распространения необходимых материалов и осуществления контроля посредством использования возможностей Интернета.



7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине (модулю)

Перечень вопросов к зачету

1. Информация и данные. База данных (БД).
2. Пользователи БД.
3. Основные требования к БД.
4. Автоматизированные информационные системы (АИС).
5. Автоматизированные банки данных (АБД). Жизненный цикл БД.
6. Уровни абстракции и этапы проектирования БД.
7. Средства и методы проектирования БД.
8. Схемы и подсхемы.
9. Инфологический подход при проектировании БД.
10. Модель "сущность-атрибут-связь".
11. Моделирование локальных представлений.
12. Объединение моделей локальных представлений: идентичность, агрегация, обобщение.
13. Проверка корректности инфологической модели.
14. Даталогическое моделирование БД.
15. Модели данных.
16. Организация данных и ограничения целостности БД.
17. Основные операции над данными.
18. Система управления базами данных (СУБД), архитектура СУБД.
19. Языки описания данных (ЯОД).
20. Языки манипулирования данным. (ЯМД).
21. Выбор СУБД. Реляционные СУБД.
22. Организация процессов обработки данных.
23. Языковые средства современных СУБД.
24. Основы реляционной алгебры.
25. Реляционное исчисление.
26. Оптимизация реляционных выражений.
27. Язык запросов SQL.
28. Доступ по первичным ключам.
29. Хешированные файлы.
30. Организация индексно-последовательных файлов.
31. Организация индексных таблиц в виде иерархических структур: бинарные деревья, В-деревья.
32. Доступ по вторичным ключам. Инвертированные списки
33. Использование систем управления базами данных. СУБД MS Access и ее возможности.

Перечень вопросов к экзамену

1. Понятие базы данных, СУБД. Методология проектирования базы данных. Концептуальное проектирование.
2. Задачи логического проектирования базы данных. Этапы доработки модели на основе реляционной модели данных.
3. Определение требований целостности данных.
4. Модель «Сущность-связь». Сущности, атрибуты, связи.
5. Расширенная ER-модель: суперклассы и подклассы сущностей, наследование, специализация.



6. Реляционная алгебра. Объединение, пересечение, разность отношений. Интерпретация операций в языке SQL.
7. Реляционная алгебра. Соединение отношений. Интерпретация операции в языке SQL.
8. Реляционная алгебра. Выборка, проекция, деление отношений. Интерпретация операций в языке SQL.
9. Реляционное исчисление. Интерпретация в языке SQL.
10. Понятие нормализации отношений. Цель нормализации.
11. Функциональные зависимости. Декомпозиция отношений по функциональным зависимостям.
12. Процесс нормализации. Первая нормальная форма. Вторая нормальная форма.
13. Третья нормальная форма. Нормальная форма Бойса-Кодда.
14. Методология физического проектирования реляционных баз данных.
15. Технология клиент-сервер. Использование SQL Server. Понятие технологий OLAP и OLTL.
16. Уровни архитектуры клиент-сервер.
17. Структурированный язык запросов SQL. Типы команд.
18. Разработка физической модели данных. Создание таблицы как основного объекта для хранения данных.
19. Ограничения целостности. Ограничения первичного и внешнего ключа. Реализация бизнес - правил.
20. Оператор SELECT и предложение FROM. Использование логических условий для выбора данных.
21. Внутренне и внешнее соединение таблиц в запросе.
22. Расчет значения результирующих столбцов. Агрегатные функции.
23. Наложение ограничений на группировку записей.
24. Реализация вложенных подзапросов.
25. Построение подзапросов, возвращающих единичные значения.
26. Построение подзапросов, возвращающих множественные значения.
27. Добавление, изменение, удаление записей. Ограничения декларативной ссылочной целостности. Каскадные воздействия.
28. Понятие просмотра и способы его формирования.
29. Модифицируемые и немодифицируемые просмотры.
30. Понятие хранимой процедуры. Создание хранимых процедур.
31. Функции пользователя. Типы функций. Создание и использование пользовательских функций.
32. Создание триггеров для обеспечения целостности данных.
33. Использование триггеров для реализации бизнес - правил.

Критерии оценок промежуточной аттестации

Оценка за работу в течение семестра складывается из результатов текущего контроля знаний и работы в течение семестра.

Текущий контроль успеваемости по дисциплине может учитывать следующее:

- выполнение студентом всех видов работ, предусмотренных программой дисциплины (в том числе ответы на семинарах, коллоквиумах, при тестировании; подготовка докладов и рефератов; выполнение лабораторных, курсовых и проверочных работ, индивидуальных заданий, участие в деловых играх и т.п.);
- посещаемость;
- самостоятельная работа студента;
- исследовательская работа и т.д.

Текущий контроль осуществляется в виде устного опроса на лекциях и оценки активности участия в дискуссии или обсуждении: в конце лекции выдается контрольный вопрос для



текущего контроля, на ответ дается 10 минут. Предусмотрено тестирование по дисциплине на карточках или с применением электронных тестов (в компьютерном классе института).

Оценка должна носить комплексный характер и учитывать достижения студента по основным компонентам учебного процесса.

Оценка знаний по 100-балльной шкале в соответствии с критериями института реализуется следующим образом:

- менее 53 балла – «неудовлетворительно»;
- от 53 до 79 баллов – «удовлетворительно»;
- от 80 до 92 баллов – «хорошо»;
- 93 балла и выше – «отлично».

Текущий контроль знаний:

№ п/п	Форма текущего контроля	Баллы
1.	Выполнение студентом всех видов работ, предусмотренных программой дисциплины (в том числе ответы на семинарах, коллоквиумах, при тестировании; подготовка докладов и рефератов; выполнение лабораторных, курсовых и проверочных работ, индивидуальных заданий, участие в деловых играх и т.п.)	45
2.	Выполнение проверочной работы (ПР)	15

Итого: текущий контроль знаний – 60 баллов.

Оценка за работу в семестре:

1. Присутствие и работа на лекции (конспект) – 1 балл;
2. Присутствие на лабораторном занятии, консультации – 1 балл;
3. Ответы на лабораторных занятиях, защита лабораторной работы – 2 балла;
4. Активность на лабораторных занятиях – 1 балл;
5. Самостоятельная работа (выполнение ПР/индивидуального задания) – 15 баллов;

Выполнение курсовой работы с последующей защитой - 100 баллов.

Итого: оценка за работу в семестре – 40 баллов.

Результаты текущего контроля успеваемости оцениваются по 100-балльной системе. Аттестованным считается студент, набравший 53 балла и выше.

Курсовая работа – 100 баллов.

Промежуточная аттестация проводится в форме зачета, экзамена, которые проводятся в устной форме в виде ответов на вопросы билета.

При этом оценка знаний студентов осуществляется в баллах с учетом:

- оценки по итогам текущего контроля знаний;
- оценки промежуточной аттестации в ходе зачета, экзамена.

Содержание билета:

1-е задание – 50 баллов;

2-е задание – 50 баллов;

Итого: за промежуточную аттестацию (результат в ходе зачета, экзамена) – 100 баллов.

Темы проверочных работ / индивидуальных заданий (ПР/ИЗ)

1. Понятие БД и СУБД.
2. Архитектура СУБД.
3. Модели БД. Реляционные БД.
4. Необходимость оптимизации хранения данных. Понятие о нормализации. 1,2,3 нормальные формы
5. Таблицы БД и связи между ними.
6. Первичные ключи и индексы.
7. Механизм BDE.



8. Создание таблиц.
9. Типы полей. Контроль за содержимым полей.
10. Таблица подстановки.
11. Вторичные индексы.
12. Ссылочная целостность.
13. Создание псевдонима БД.
14. Модуль данных.
15. Создание объектов-полей.
16. Создание объектов-столбцов.
17. Объекты для реальных полей.
18. Объекты для подстановочных полей.
19. Объекты для вычисляемых полей.
20. Объекты для пустых полей.
21. Обращение к значению поля.
22. Проверка правильности введенного в поля значения.
23. Формирование текстового представления поля.
24. Открытие и закрытие набора данных.
25. Программный доступ к записям.

Темы курсовых работ

1. Разработка базы данных «Сессия».
2. Разработка базы данных «Магазин меховых изделий».
3. Разработка базы данных «Школьная библиотека».
4. Разработка базы данных «Учет успеваемости в школе».
5. Разработка базы данных «Листок учета студентов».
6. Разработка базы данных «Видеотека».
7. Разработка базы данных «Кадровое агентство».
8. Разработка базы данных «Магазин книг».
9. Разработка базы данных «Магазин автозапчастей».
10. Разработка базы данных «Кредиты».
11. Разработка базы данных «Мебельный магазин».
12. Разработка базы данных «Овощная база».
13. Разработка базы данных «Магазин компьютерных книг».
14. Разработка базы данных «Известные люди».
15. Разработка базы данных «Расписание экзаменов».
16. Разработка базы данных «Отель, регистрация клиентов».
17. Разработка базы данных «Автомобильные салоны».
18. Разработка базы данных «Автостоянка».
19. Разработка базы данных «Расчет родительской платы в детском саду».
20. Разработка базы данных «Студенческий отдел кадров».
21. Разработка базы данных «Рекламное агентство».
22. Разработка базы данных «Классный журнал».
23. Разработка базы данных «Туристическое агентство».
24. Разработка базы данных «Зоопарк».
25. Разработка базы данных «Студенты».

Примерный перечень тестов для промежуточного контроля

Студент должен выбрать один или несколько правильных ответов.

Вариант 1

1) Возможные действия с данными в СУБД



- 1 Добавить
- 2 Удалить
- 3 Просмотреть
- 4 Изменить
- 5 *Все перечисленное*
- 2) **Для чего нужна команда SELECT?**
 - 1 *для извлечения определенных строк из таблицы*
 - 2 для извлечения определенных столбцов из таблицы
 - 3 для объединения двух таблиц
- 3) **Для чего нужна команда JOIN?**
 - 1 для извлечения определенных столбцов из таблицы
 - 2 *для объединения двух таблиц*
 - 3 для извлечения определенных строк из таблицы
- 4) **Что такое оवेशествленные вычисления?**
 - 1 *промежуточные результаты полностью оवेशествлены*
 - 2 оवेशествления промежут. результатов не происходит
 - 3 промежут. операции передаются по частям
- 5) **Что такое конвейерные вычисления?**
 - 1 оवेशествления промежут. результатов не происходит
 - 2 *промежут. операции передаются по частям*
 - 3 промежуточные результаты полностью оवेशествлены
- 6) **Одна из отличительных особенностей реляционных сис...**
 - 1 невозможность обработки множеств
 - 2 *возможность обработки множеств*
 - 3 обработка только одной строки
- 7) **СУБД это**
 - 1 *Комплекс программных средств и файлов данных*
 - 2 Программный модуль
 - 3 Файлы данных
- 8) **Формат данных в базе данных**
 - 1 Для всех СУБД одинаков
 - 2 В каждой СУБД индивидуальный
 - 3 *Возможно наследование свойств для некоторых СУБД*
- 9) **Сколько уровней существует в архитектуре ANSI/SPAR...**
 - 1 1
 - 2 2



3 3

4 5

10) Внутренний уровень

- 1 *Связан со способами сохранения информ.*
- 2 *Связан со способами представления данных*
- 3 *Является "промежуточным" уровнем между двумя первы*

11) Внешний уровень

- 1 *Связан со способами сохранения информации*
- 2 *Связан со способами представления данных*
- 3 *Является "промежуточным" уровнем между двумя первы*

12) Концептуальный уровень

- 1 *Связан со способами сохранения информации*
- 2 *Связан со способами представления данных*
- 3 *Является "промежуточным" уровнем*

13) Что такое система баз данных?

- 1 *компьютеризированная система хранения записей*
- 2 *затрудняюсь ответить*
- 3 *это особого рода документированные данные*

14) База данных - это

- 1 *Набор данных, собранных на одной дискете*
- 2 *данные, предназначенные для работы программы*
- 3 *совокупность взаимосвязанных данных, организованны*

15) Реляционная база данных - это

- 1 *БД, табличное хранение данных*
- 2 *БД, в которой элементы в записи упорядочены.*
- 3 *БД, где записи расположены в произвольном порядке*

Вариант 2р

16) Иерархическая база данных - это

- 1 *БД, в которой информация виде таблиц*
- 2 *БД, в которой элементы в записи упорядочены*
- 3 *БД, записи расположены в произвольном порядке*

17) Какой язык поддерживает практически все современные СУБД

- 1 *DDL*
- 2 *DCL*
- 3 *PL/I*
- 4 *SQL*



18) Язык DDL-это

- 1 *Язык определения данных*
- 2 Язык обработки данных
- 3 Язык изменения данных
- 4 Язык сортировки данных

19) Язык DML-это

- 1 Язык определения данных
- 2 *Язык обработки данных*
- 3 Язык сортировки данных
- 4 Язык изменения данных

20) Внутренний уровень архитектуры СУБД

- 1 *Наиболее близок к физическому*
- 2 Наиболее близок к пользователю

21) Внешний уровень архитектуры СУБД

- 1 Наиболее близок к физическому
- 2 *Наиболее близок к пользователю*

22) Концептуальный уровень

- 1 Наиболее близок к физическому
- 2 Наиболее близок к пользователю
- 3 *Переходный от внутреннего к внешнему*

23) Проектированием БД занимается

- 1 Администратор БД
- 2 Программист БД
- 3 Пользователь БД
- 4 *Проектировщик БД*

24) Под понятием интегрированности данных подразумевается

- 1 *представить БД как объединение файлов данных*
- 2 использования отдельных элементов БД различн. пользоват.

25) Под понятием разделяемости данных подразумевается ...

- 1 *использования элементов БД несколькими пользоват.*
- 2 представить БД как объединение неск. файлов данных

26) Один из компонентов реляционной модели

- 1 ограниченный набор скалярных типов
- 2 *генератор типов отношений*
- 3 ограниченный набор реляционных операторов

27) Вместо какого термина применяется слово "кортеж"



1 таблица

2 запись

3 первичный ключ

28) Какому термину соответствует термин "отношение"

1 строка

2 запись

3 таблица

29) Какому термину соответствует термин "кортеж"

1 таблица

2 строка

3 столбец

30) Реляционная модель данных предполагает что

1 Данные представлены посредством строк в таблицах

2 Для обработки строк данных предостав. операторы

3 Оба варианта верны

31) Могут ли пользователи создавать собственные типы данных

1 да

2 нет

32) Почему реляционную модель называют непроцедурной?

1 пользователь указывает что делать, а не как

2 пользователь указывает как делать, а не что

8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

8.1.Основная учебная литература

1. Щелоков С.А. Базы данных: учебное пособие. - Оренбургский государственный университет, 2014. - 298 с. // <http://www.knigafund.ru/books/185133>

8.2.Дополнительная литература

1. Голицына О.Л. и др. Базы данных: учебное пособие / Голицына О.Л., Максимов Н.В., Попов И.И. – М.: ФОРУМ: ИНФРА-М, 2006. – 352 с. (гриф)

2. Советов Б.Я., Цехановский В.В., Чертовской В.Д. Базы данных: теория и практика: учебник. – 2-е изд., испр. и доп. – М.: Высшая школа, 2007. (гриф)

В соответствии с договором студентам и преподавателям института предоставляется право доступа к электронному периодическому изданию Электронно-библиотечной системы «КнигаФонд» (www.knigafund.ru).

1. [Кузнецов С. Введение в реляционные базы данных](#) Национальный Открытый Университет «ИНТУИТ» • 2016 год • 248 страниц

2. [Щелоков С. А. Базы данных: учебное пособие](#) Оренбургский государственный университет • 2014 год • 298 страниц



3. [Щелоков С.А. Разработка и создание баз данных средствами СУБД Access и SQL Server](#) Оренбургский государственный университет • 2014 год • 109 страниц
4. [Стасьшин В.М. Проектирование информационных систем и баз данных: учебное пособие](#) НГТУ • 2012 год • 100 страниц

9. Перечень ресурсов информационно-телекоммуникационной сети Интернет, необходимых для освоения дисциплины (модуля)

1. Издательство «Открытые системы»: <http://www.osp.ru>
2. Мир ПК. – Электронный журнал. <http://www.pcworld.ru>
3. LAN. Электронный журнал. <http://www.osp.ru/lan>
4. База ГОСТов, ЕСПД: <http://www.gostbaza.ru/>

10. Методические указания для обучающихся по освоению дисциплины (модуля)

Планирование и организация времени, необходимого для изучения дисциплины

Важным условием успешного освоения дисциплины является создание системы правильной организации труда, позволяющей распределить учебную нагрузку равномерно в соответствии с графиком образовательного процесса. Большую помощь в этом может оказать составление плана работы на семестр, месяц, неделю, день. Его наличие позволит подчинить свободное время целям учебы, трудиться более успешно и эффективно. С вечера всегда надо распределять работу на завтрашний день. В конце каждого дня целесообразно подвести итог работы: тщательно проверить, все ли выполнено по намеченному плану, не было ли каких-либо отступлений, а если были, по какой причине они произошли. Нужно осуществлять самоконтроль, который является необходимым условием успешной учебы. Если что-то осталось невыполненным, необходимо изыскать время для завершения этой части работы, не уменьшая объема недельного плана. Все задания к практическим занятиям, а также задания, вынесенные на самостоятельную работу, рекомендуется выполнять непосредственно после соответствующей темы лекционного курса, что способствует лучшему усвоению материала, позволяет своевременно выявить и устранить «пробелы» в знаниях, систематизировать ранее пройденный материал, на его основе приступить к овладению новыми знаниями и навыками.

Система обучения основывается на рациональном сочетании нескольких видов учебных занятий (в первую очередь, лекций и практических занятий), работа на которых обладает определенной спецификой.

Подготовка к лекциям

Знакомство с дисциплиной происходит уже на первой лекции, где от студента требуется не просто внимание, но и самостоятельное оформление конспекта. При работе с конспектом лекций необходимо учитывать тот фактор, что одни лекции дают ответы на конкретные вопросы темы, другие – лишь выявляют взаимосвязи между явлениями, помогая студенту понять глубинные процессы развития изучаемого предмета как в истории, так и в настоящее время.

Конспектирование лекций – сложный вид вузовской аудиторной работы, предполагающий интенсивную умственную деятельность студента. Конспект является полезным тогда, когда записано самое существенное и сделано это самим обучающимся. Не надо стремиться записать дословно всю лекцию. Такое «конспектирование» приносит больше вреда, чем пользы. Целесообразно вначале понять основную мысль, излагаемую лектором, а затем записать ее. Желательно запись осуществлять на одной странице листа или оставляя поля, на которых позднее, при самостоятельной работе с конспектом, можно сделать дополнительные записи, отметить непонятные места.

Конспект лекции лучше подразделять на пункты, соблюдая красную строку. Этому в большой степени будут способствовать вопросы плана лекции, предложенные преподавателям.



Следует обращать внимание на акценты, выводы, которые делает лектор, отмечая наиболее важные моменты в лекционном материале замечаниями «важно», «хорошо запомнить» и т.п. Можно делать это и с помощью разноцветных маркеров или ручек, подчеркивая термины и определения.

Целесообразно разработать собственную систему сокращений, аббревиатур и символов. Однако при дальнейшей работе с конспектом символы лучше заменить обычными словами для быстрого зрительного восприятия текста.

Работая над конспектом лекций, всегда необходимо использовать не только учебник, но и ту литературу, которую дополнительно рекомендовал лектор. Именно такая серьезная, кропотливая работа с лекционным материалом позволит глубоко овладеть теоретическим материалом.

Подготовка к практическим (лабораторным) занятиям

Подготовку к каждому практическому занятию студент должен начать с ознакомления с планом практического занятия, который отражает содержание предложенной темы. Тщательное продумывание и изучение вопросов плана основывается на проработке текущего материала лекции, а затем изучения обязательной и дополнительной литературы, рекомендованной к данной теме. Все новые понятия по изучаемой теме необходимо выучить наизусть и внести в глоссарий, который целесообразно вести с самого начала изучения курса.

Результат такой работы должен проявиться в способности студента свободно ответить на теоретические вопросы практикума, его выступлении и участии в коллективном обсуждении вопросов изучаемой темы, правильном выполнении практических заданий и контрольных работ.

В процессе подготовки к практическим занятиям, студентам необходимо обратить особое внимание на самостоятельное изучение рекомендованной литературы. При всей полноте конспектирования лекции в ней невозможно изложить весь материал из-за лимита аудиторных часов. Поэтому самостоятельная работа с учебниками, учебными пособиями, научной, справочной литературой, материалами периодических изданий и Интернета является наиболее эффективным методом получения дополнительных знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала, формирует у студентов свое отношение к конкретной проблеме.

Рекомендации по написанию проверочных работ / индивидуальных заданий, курсовой работы

Рекомендации по работе с литературой

Работу с литературой целесообразно начать с изучения общих работ по теме, а также учебников и учебных пособий. Далее рекомендуется перейти к анализу монографий и статей, рассматривающих отдельные аспекты проблем, изучаемых в рамках курса, а также официальных материалов и неопубликованных документов (научно-исследовательские работы, диссертации), в которых могут содержаться основные вопросы изучаемой проблемы.

Работу с источниками надо начинать с ознакомительного чтения, т.е. просмотреть текст, выделяя его структурные единицы. При ознакомительном чтении закладками отмечаются те страницы, которые требуют более внимательного изучения.

В зависимости от результатов ознакомительного чтения выбирается дальнейший способ работы с источником. Если для разрешения поставленной задачи требуется изучение некоторых фрагментов текста, то используется метод выборочного чтения. Если в книге нет подробного оглавления, следует обратить внимание ученика на предметные и именные указатели.

Избранные фрагменты или весь текст (если он целиком имеет отношение к теме) требуют вдумчивого, неторопливого чтения с «мысленной проработкой» материала. Такое чтение предполагает выделение: 1) главного в тексте; 2) основных аргументов; 3) выводов. Особое внимание следует обратить на то, вытекает тезис из аргументов или нет.



Необходимо также проанализировать, какие из утверждений автора носят проблематичный, гипотетический характер и уловить скрытые вопросы.

Понятно, что умение таким образом работать с текстом приходит далеко не сразу. Наилучший способ научиться выделять главное в тексте, улавливать проблематичный характер утверждений, давать оценку авторской позиции – это сравнительное чтение, в ходе которого студент знакомится с различными мнениями по одному и тому же вопросу, сравнивает весомость и доказательность аргументов сторон и делает вывод о наибольшей убедительности той или иной позиции.

Если в литературе встречаются разные точки зрения по тому или иному вопросу из-за сложности прошедших событий и правовых явлений, нельзя их отвергать, не разобравшись. При наличии расхождений между авторами необходимо найти рациональное зерно у каждого из них, что позволит глубже усвоить предмет изучения и более критично оценивать изучаемые вопросы. Знакомясь с особыми позициями авторов, нужно определять их схожие суждения, аргументы, выводы, а затем сравнивать их между собой и применять из них ту, которая более убедительна.

Следующим этапом работы с литературными источниками является создание конспектов, фиксирующих основные тезисы и аргументы. Можно делать записи на отдельных листах, которые потом легко систематизировать по отдельным темам изучаемого курса. Другой способ – это ведение тематических тетрадей-конспектов по одной какой-либо теме. Большие специальные работы монографического характера целесообразно конспектировать в отдельных тетрадях. Здесь важно вспомнить, что конспекты пишутся на одной стороне листа, с полями и достаточным для исправления и ремарок межстрочным расстоянием (эти правила соблюдаются для удобства редактирования). Если в конспектах приводятся цитаты, то непременно должно быть дано указание на источник (автор, название, выходные данные, № страницы). Впоследствии эта информация может быть использована при написании текста реферата или другого задания.

Таким образом, при работе с источниками и литературой важно уметь:

- сопоставлять, сравнивать, классифицировать, группировать, систематизировать информацию в соответствии с определенной учебной задачей;
- обобщать полученную информацию, оценивать прослушанное и прочитанное;
- фиксировать основное содержание сообщений; формулировать, устно и письменно, основную идею сообщения; составлять план, формулировать тезисы;
- готовить и презентовать развернутые сообщения типа доклада;
- работать в разных режимах (индивидуально, в паре, в группе), взаимодействуя друг с другом;
- пользоваться реферативными и справочными материалами;
- контролировать свои действия и действия своих товарищей, объективно оценивать свои действия;
- обращаться за помощью, дополнительными разъяснениями к преподавателю, другим студентам.
- пользоваться лингвистической или контекстуальной догадкой, словарями различного характера, различного рода подсказками, опорами в тексте (ключевые слова, структура текста, предваряющая информация и др.);
- использовать при говорении и письме перифраз, синонимичные средства, словоописания общих понятий, разъяснения, примеры, толкования, «словотворчество»;
- повторять или перефразировать реплику собеседника в подтверждении понимания его высказывания или вопроса;
- обратиться за помощью к собеседнику (уточнить вопрос, переспросить и др.);
- использовать мимику, жесты (вообще и в тех случаях, когда языковых средств не хватает для выражения тех или иных коммуникативных намерений).



При подготовке к промежуточной аттестации целесообразно:

- внимательно изучить перечень вопросов и определить, в каких источниках находятся сведения, необходимые для ответа на них;
- внимательно прочитать рекомендованную литературу;
- составить краткие конспекты ответов (планы ответов).

11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем

Windows 8, Microsoft Office 2007, Windows 8 (Microsoft Word 2007 - Текстовый процессор; Microsoft Excel 2007 - Табличный процессор; Microsoft Access 2007 - Система управления базами данных; Microsoft PowerPoint 2007 - Создание и показ презентаций); Microsoft Visio 2016 – редактор диаграмм и блок-схем.

ГАРАНТ-Мастер - Информационно-правовая система.

12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Для проведения занятий по дисциплине «Базы данных» кафедра располагает необходимой материально-технической базой, обеспечивающей проведение всех видов занятий, предусмотренных данной программой и соответствующей действующим санитарным и противопожарным правилам и нормам:

1. Специализированной аудиторией для проведения лекционных и семинарских занятий, оснащенной ЖК-телевизором, проектором Nec NP-V260G, стационарным экраном «Digis Optimal-C»;
2. Специализированной аудиторией для проведения практических занятий, семинаров, курсового проектирования, консультаций, текущего контроля и промежуточной аттестации, оснащенной ЖК-телевизором, проектором Benq MS504, стационарным экраном «Digis Optimal-C»;
3. Специализированной аудиторией для самостоятельной работы обучающихся, оснащенной ноутбуками «Lenovo B590» с выходом в сеть Интернет и доступом к электронной информационно-образовательной среде ЛЭГИ;
4. Специализированным компьютерным классом, оснащенным ноутбуками «Lenovo B590»;
5. Учебниками, учебными пособиями и методической литературой библиотеки ЛЭГИ, наборами учебно-наглядных пособий по основным разделам программы.

**Лист согласований**

СОГЛАСОВАНО

РАЗРАБОТАНО

Представитель руководства по СМК

Доцент кафедры ПИЭ

_____ Н.Ю. Филоненко

_____ Т.В. Лаврухина

« _____ » 2016 г.

« _____ » 2016 г.



ПРИЛОЖЕНИЕ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ КУРСОВОЙ РАБОТЫ

ВВЕДЕНИЕ

Курсовая работа является одной из форм подготовки специалистов высшей квалификации.

Целью написания курсовой работы является выработка у студентов навыков самостоятельного изучения отдельных тем и исследования проблем, глубокое усвоение положений, выводов, законов, приобретение опыта самостоятельного получения и накопления знаний, что необходимо для подготовки в дальнейшем дипломной работы, а будущему дипломированному специалисту – в его трудовой деятельности.

Тематика курсовой работы может быть выбрана из двух направлений:

1. Теоретическое исследование структур системы управления базы данных.
2. Создание и проектирование автоматизированной системы с функциональным информационным наполнением.

Теоретическое исследование по выбранной теме должно обладать научной новизной, выполняться по мало изученной или дискуссионной проблеме (или по отдельному ее аспекту).

Курсовая работа по созданию автоматизированной системы, должна содержать информационную модель исследуемого процесса и представлена в виде проектной документации на разработку программы.

Подготовка и написание курсовой работы состоит из нескольких этапов:

1. Выбор темы и ее согласование с научным руководителем.
2. Обоснование структуры работы.
3. Составление библиографии, ознакомление с законодательными актами, нормативными документами, другими источниками и литературой, относящимися к теме курсовой работы.
4. Сбор теоретического и статистического материала.
5. Обработка и анализ полученной информации с применением современных математико-статистических методов.
6. Формулирование выводов, а в случае теоретического исследования - научно-обоснованной разработкой или альтернативной интерпретацией тех или иных концепций или позиций по теме работы.
7. Оформление курсовой работы в соответствии с установленными требованиями.

2. ТЕМЫ КУРСОВЫХ РАБОТ

Тематика курсовых работ рассматривается и утверждается на заседании кафедры, а затем предлагается студентам.

Студенты могут предложить свою тему курсовой работы с учетом ранее выполненных работ по другим дисциплинам, учитывая соответствие этой проблемы содержанию тех дисциплин, по которым учебными планами предусматривается написание курсовых работ. Такая преемственность обеспечит возможность исследования интересующей проблемы с разных сторон и последовательную подготовку к написанию ВКР.

Выбрав тему курсовой работы, студент согласует ее с научным руководителем – преподавателем соответствующей учебной дисциплины.

В отдельных случаях тема студенту может быть предложена кафедрой для разработки тема теоретической, исследовательской направленности.



2.1. Список тем курсовых работ теоретической направленности

- 1) Модели БД. Реляционные БД.
- 2) Необходимость оптимизации хранения данных.
- 3) Таблицы БД и связи между ними.
- 4) Первичные ключи и индексы.
- 5) Типы полей. Контроль за содержимым полей.
- 6) Таблица подстановки.
- 7) Вторичные индексы.
- 8) Ссылочная целостность.
- 9) Создание псевдонима БД.
- 10) Модуль данных. 15. Создание объектов-полей.
- 11) Проверка правильности введенного в поля значения.
- 12) Формирование текстового представления поля.
- 13) Фильтрация записей.
- 14) Блокировка таблиц в многопользовательском режиме.
- 15) Смена текущего индекса.
- 16) Добавление нового индекса.
- 17) Удаление индекса.
- 18) Составные индексы.
- 19) Эксклюзивный доступ к таблице.
- 20) Точный и неточный поиск записей в таблице.

2.2. Список тем курсовых работ по созданию информационных систем

- 1) Разработка базы данных «Сессия»
- 2) Разработка базы данных «Магазин меховых изделий»
- 3) Разработка базы данных «Школьная библиотека»
- 4) Разработка базы данных «Учет успеваемости в школе»
- 5) Разработка базы данных «Листок учета студентов»
- 6) Разработка базы данных «Видеотека»
- 7) Разработка базы данных «Кадровое агентство»
- 8) Разработка базы данных «Магазин книг»
- 9) Разработка базы данных «Магазин автозапчастей»
- 10) Разработка базы данных «Кредиты»
- 11) Разработка базы данных «Мебельный магазин»
- 12) Разработка базы данных «Овощная база»
- 13) Разработка базы данных «Магазин компьютерных книг»
- 14) Разработка базы данных «Известные люди»
- 15) Разработка базы данных «Расписание экзаменов»
- 16) Разработка базы данных «Отель, регистрация клиентов»
- 17) Разработка базы данных «Автомобильные салоны»
- 18) Разработка базы данных «Автостоянка»
- 19) Разработка базы данных «Расчет родительской платы в детском саду»
- 20) Разработка базы данных «Студенческий отдел кадров»
- 21) Разработка базы данных «Рекламное агентство»
- 22) Разработка базы данных «Классный журнал»
- 23) Разработка базы данных «Туристическое агентство»
- 24) Разработка базы данных «Зоопарк»
- 25) Разработка базы данных «Студенты»



3. СТРУКТУРА И СОДЕРЖАНИЕ КУРСОВОЙ РАБОТЫ

3.1. Содержательная часть работы

Курсовая работа должна иметь:

- титульный лист, оформленный в соответствии с установленными требованиями (пример оформления приведен на сайте института);
- содержание;
- введение;
- разделы (главы) и подразделы;
- заключение;
- список использованных источников и литературы;
- приложения (в случаях необходимости).

Общий объем курсовой работы должен составлять не менее 35-40 страниц машинописного текста, напечатанного через полтора интервала шрифтом №14 Times New Roman или Arial №12 (включая титульный лист, лист содержания, список источников и литературы).

Приложения в общий объем работы не входят.

Содержание (оглавление) включает введение, наименования разделов (глав), подразделов, заключение, список источников и литературы, приложения с указанием номера их начальной страницы.

Во введении:

- обосновывается актуальность избранной темы;
- определяется степень разработанности проблемы исследования на данный момент времени;
- формулируются цель и задачи курсовой работы;
- определяются хронологические границы исследования;
- дается характеристика информационной базы;
- дается обзор источников и соответствующей отечественной и зарубежной литературы с анализом авторских концепций по исследуемой проблеме. В курсовой работе теоретического характера обзор источников и литературы может быть выделен в специальный раздел.
- определяются методы исследования и анализа информации;
- дается краткая характеристика структуры курсовой работы (краткое изложение рассмотренных вопросов в каждом разделе).

Объем введения должен составлять примерно 2-3 страницы.

Цель исследования – это конечный результат выполнения курсовой работы. Задачи - это пути (что необходимо сделать) для достижения цели исследования.

Предмет исследования – это та проблема (процесс), на которую направлено исследование.

Изложение содержания работы должно быть строго логичным, а разделы – взаимосвязанными в рамках общей логики изложения материала. Особое внимание следует обратить на переход от одной главы к другой.

Каждый раздел (глава) курсовой работы должен заканчиваться краткими выводами, в которых излагаются обобщенно наиболее качественные результаты исследования. Как правило, эти выводы либо предопределяют необходимость и содержание далее излагаемого материала, либо могут быть использованы для более глубокого его понимания.

Текст введения, каждого раздела (главы), заключения и списка источников и литературы следует начинать с нового листа.

Объем содержательной части курсовой работы составляет примерно 70-80 % общего объема работы (примерно 30-35 страниц).

При цитировании источников и литературы по тексту работы внизу страницы делается подстрочная.



Заключение посвящено изложению основных результатов выполненной работы. В нем следует в концентрированном виде изложить итог решения тех задач, которые были поставлены в курсовой работе, обобщить ранее сформулированные выводы и сделать общий вывод. В курсовой работе теоретической направленности следует также охарактеризовать научную ценность результатов работы, указать перспективы дальнейшей разработки темы. Объем заключения составляет примерно 2-3 страницы.

Список имеет следующую структуру с обязательным заголовком для каждого раздела.

В этом разделе списка располагаются в алфавите авторов и заглавий монографии, статьи, рецензии, авторефераты, электронные ресурсы. В описании статей обязательно указываются название журнала или сборника, где они опубликованы, год, номер и страница, в других позициях литературы указывается также общее количество страниц в публикации.

Приложения помещают после списка источников и литературы в порядке их упоминания в тексте.

3.2. Структура курсовой работы по теоретической направленности

В курсовых работах, представляющих теоретическое исследование, можно изложить исторические аспекты проблемы (явления), опыт разных стран (регионов); охарактеризовать степень ее изученности; изложить и сопоставить дискуссионные вопросы по теме исследования и альтернативные взгляды разных авторов, уточнить формулировки (понятийный аппарат); рассмотреть существующие методические подходы к анализу данной проблемы. Необходимо при этом не просто пересказать существующие точки зрения на сущность данного явления или методологические основы, а творчески осмыслив и проанализировав их, обосновать в результате собственную позицию, аргументировать ценность результатов исследования, возможность их использования в практической деятельности.

3.3. Структура курсовой работы по созданию информационных систем

В курсовой работе, касающейся прикладных аспектов создания автоматизированных систем с информационной моделью должен быть спроектирован процесс формирования структуры базы данных и системы управления базы данных.

Студентом необходимо проработать и создать комплект документации по следующим нормативным структурам:

- 1) функциональный состав системы в виде диаграммы IDF0;
- 2) информационное обеспечение системы в виде концептуальной модели (схема IDF3, DFD или диаграмма Чена);
- 3) физическая модель базы данных;
- 4) программные структуры управления базой данных (триггеры, процедуры);
- 5) механизмы соблюдения целостности и ссылочности.

Так же необходимо протестировать функциональность системы на входном наборе данных и полученные результаты оформить в виде полученных результатов экранных форм.

4. ОФОРМЛЕНИЕ КУРСОВОЙ РАБОТЫ

4.1. Требования к оформлению содержательной части

Курсовая работа должна быть подготовлена в одном экземпляре и сброшюрована.

Оглавление (содержание), которое располагают после титульного листа, печатается шрифтом Times New Roman № 14 или Arial №12 через полтора интервала, разделы отделяются пробелом в два интервала



Текст дипломной работы следует печатать шрифтом № 14 Times New Roman или Arial №12 через полтора интервала, соблюдая следующие размеры полей: левое - не менее 30 мм, правое - не менее 10 мм, верхнее - не менее 15 мм, нижнее - не менее 20 мм.

Разделы и подразделы должны иметь заголовки. Заголовки разделов оформляют симметрично тексту, заголовки подразделов - с абзаца. Расстояние между заголовками и текстом должно быть увеличено для выделения заголовка.

Заголовки разделов печатаются прописными буквами, заголовки подразделов - строчными буквами. Заголовки не подчеркиваются, в конце их точки не ставятся.

Заголовки разделов и подразделов нумеруются арабскими цифрами. Номер подраздела состоит из номера раздела и подраздела, разделенных точкой.

Список использованных источников и литературы печатается через два интервала, каждое название начинается с абзаца.

Каждое приложение следует начинать с нового листа, в правом верхнем углу которого пишется слово «Приложение» и номер, обозначенный арабской цифрой (без знака №), например: Приложение 1. В левом нижнем углу следует указать, на основании каких источников составлено приложение.

4.2. Требования к оформлению ссылок на литературные источники

Ссылки на печатные издания.

Заголовок (автор). Основное заглавие: сведения, относящиеся к заглавию / первые сведения об ответственности ; последующие сведения об ответственности. – Сведения об издании. – Место издания : издатель, дата издания. – Объем. – (Основное заглавие серии ; номер выпуска серии). – Примечания.

Источником информации для описания является титульный лист (для книг), первая страница текста (для статей), титульный экран (для электронных ресурсов).

Ссылки на электронные ресурсы.

Заголовок (автор). Основное заглавие [Электронный ресурс] : сведения, относящиеся к заглавию / сведения об ответственности. – Сведения об издании. – Обозначение вида ресурса. – Место издания : издатель, дата издания. – Специфическое обозначение материала и количество физических единиц (*только для ресурса локального доступа*). – (Основное заглавие серии). – Примечание (указать режим доступа для ресурса Интернет).

5. ПОРЯДОК ЗАЩИТЫ КУРСОВОЙ РАБОТЫ

Выполненная студентом курсовая работа сдается на кафедру в сроки, отведенные для рецензирования научным руководителем.

В ходе защиты курсовой работы задача студента – показать углубленное понимание вопросов конкретной темы, хорошее владение материалом по теме.

Процедура защиты включает следующие этапы:

- сообщение студента об основном содержании работы;
- ответы студента на вопросы.

Студент должен тщательно подготовиться к защите курсовой работы. Общая схема доклада (на 5-7 минут):

- следует дать краткое обоснование темы, показать ее актуальность;
- указать, какова цель работы;
- раскрыть, какие результаты достигнуты в ходе исследования и что сделано лично студентом;
- изложить вытекающие из проведенного исследования основные выводы.