

Базы данных

Лекция 00 – Вводная

Преподаватель: Поденок Леонид Петрович, 505а-5

+375 17 293 8039 (505а-5)

+375 17 320 7402 (ОИПИ НАНБ)

prep@lsi.bas-net.by

ftp://student:2ok*uK2@Rwox@lsi.bas-net.by

Кафедра ЭВМ, 2024

Оглавление

БАЗЫ ДАННЫХ. Вторая часть.....	3
Содержание учебной дисциплины:.....	4
Литература.....	8
Литература дополнительная.....	9
Основная вычислительная среда.....	10
Лабораторные работы.....	11
Что следует представить для сдачи лабораторной работы.....	13
Общие требования к оформлению ПЗ.....	13
Коротко об интерфейсе приложения.....	15
Пример базы данных.....	15
Базовые правила.....	16
Особенности формы TABLE.....	17
Особенности формы FORM.....	20

БАЗЫ ДАННЫХ. Вторая часть

Учебная программа учреждения образования по учебной дисциплине для специальности
1-40 02 01 Вычислительные машины, системы и сети

Курс 4, семестр 7.

Всего: 40
Лекций: 24
Лабораторных: 16
Аттестация: зачет

Перечень учебных дисциплин, усвоение которых необходимо
для изучения данной учебной дисциплины

№ п.п.	Название учебной дисциплины	Раздел, темы
1	Программирование на языках высокого уровня	Языки программирования С и С++
2	Основы алгоритмизации и программирования	Все
3	Операционные системы и системное программирование	Все

Содержание учебной дисциплины:

Первая часть

№ темы	Наименование разделов, тем	Содержание тем
1	Введение в системы баз данных	Системы баз данных и информационные системы. Компоненты систем баз данных и их виды. Архитектура ANSI-SPARC. Функции и структура СУБД. Архитектура информационных систем.
2	Проектирование баз данных	Жизненный цикл информационных систем. Общая классификация моделей данных. Методология проектирования баз данных.
3	Концептуальные модели данных	Семантическое моделирование данных. Виды концептуальных моделей данных. ER-модель и ER-диаграмма.
4	Реляционная модель данных	Дореляционные логические модели данных. Структура реляционных данных. Целостность реляционных данных. Преобразование ER-диаграммы в реляционную схему данных. Реляционная алгебра и реляционное исчисление. Нормализация реляционных данных.
5	Языки баз данных	Язык SQL: идентификаторы, функции, типы данных, создание схемы данных, создание таблиц и индексов, выборка и модификация данных, представления, транзакции, управление доступом к данным.
6	Физические модели данных	Файловые структуры данных и методы доступа: неупорядоченные, упорядоченные и хешированные файлы, индексы, очереди, сырые записи.

Вторая часть

№ темы	Наименование разделов, тем	Содержание тем
7	Прикладные интерфейсы баз данных использующих SQL	Технологии и прикладные программные интерфейсы для доступа к базам данных, использующих SQL. Использование SQL-операторов в прикладных программах.
8	Прикладные интерфейсы баз данных не использующих SQL	Технологии и прикладные программные интерфейсы для доступа к базам данных, не использующих SQL.
9	Разработка приложений	Виды приложений для работы с базами данных. Организация и структура прикладных программ, взаимодействующих с базами данных.
10	Серверное программирование	Триггеры и процедурные языки
11	Перспективы развития баз данных	Постреляционные и нереляционные СУБД. Хранилища данных. Интерактивная аналитическая обработка данных.
12	Настройка и обслуживание СУБД	Основы администрирования систем управления базами данных. Установка и настройка сервера БД. Управление пользователями (ролями). Управление базами данных. Резервное копирование и восстановление. Репликация.

Лабораторные
Первая часть – 32 часа

№ темы по п.1	Наименование лабораторной работы	Содержание
3	1) Разработка концептуальной модели данных.	Анализ предметной деятельности в рамках заданной индивидуальной темы. Разработка ER-модели данных в виде текстового описания и оформление ER-диаграммы по описанию ER-модели.
4	2) Разработка реляционной модели данных	Преобразование ER-диаграммы модели данных в реляционную схему данных и ее описание.
5	3) Реализация модели данных в виде базы данных	Создание таблиц базы данных модели данных согласно реляционной схемы и их наполнение.
5	4) Формирование запросов к базе данных	Формирование простых запросов к базе данных
5	5) Формирование сложных запросов к базе данных	Формирование сложных запросов к базе данных, содержащих подзапросы и предложение WITH

Вторая часть – 16 часов

№ темы по п.1	Наименование лабораторной работы	Содержание
7	6) Разработка пользовательского интерфейса прикладной программы.	Разработка спецификаций пользовательского интерфейса приложения. Знакомство с прикладным интерфейсом СУБД PostgreSQL и программирование клиентской компоненты приложения.
7	7) Разработка серверной части прикладной программы.	Разработка спецификаций серверной части (backend) программы. Программирование серверной части с использованием прикладного интерфейса СУБД PostgreSQL.
8	8) Разработка NoSQL базы данных и спецификаций прикладной программы.	Освоение прикладного интерфейса СУБД BerkeleyDB. Разработка конвертора базы данных Postgres в набор баз данных BerkeleyDB. Разработка спецификаций приложения.
8	9) Разработка прикладной программы с использованием NoSQL базы данных.	Программирование прикладной программы с использованием прикладного интерфейса СУБД BerkeleyDB.

SQL базы данных– Postgres

NoSQL базы данных «ключ-значение» – BerkeleyDB, Tokio Cabinet, Kioto Cabinet

Литература

1. Рогов, Е. PostgreSQL изнутри / Е. Рогов. – Москва : ДМК Пресс, 2024. – 664 с.
2. Коннолли, Т. Базы данных: проектирование, реализация и сопровождение. Теория и практика / Т. Коннолли, К. Бегг, А. Страчан. – К.; М.; СПб.: ИД «Вильямс», 2003. – 1440 с.
3. Дейт, К. Дж. SQL и реляционная теория. Как грамотно писать код на SQL / К. Дж. Дейт. - СПб.: Символ-Плюс, 2010. – 480 с.
4. Дейт, К. Дж. Введение в системы баз данных / К. Дж. Дейт. – К.; М.; СПб.: ИД «Вильямс», 2006. – 1328 с.
5. Гарсиа-Молина, Г. Системы баз данных. Полный курс / Г. Гарсиа-Молина, Д. Ульман, Д. Уидом. - К.; М.; СПб.: ИД «Вильямс», 2003. – 1088 с.
6. Yadava H. The Berkeley DB Book. Apress – 2007
7. Васильев А.Ю. Работа с PostgreSQL: настройка и масштабирование. – 2017.
8. Моргунов Е. П. PostgreSQL. Основы языка SQL: учеб. пособие / Е. П. Моргунов; под ред. Е. В. Рогова, П. В. Лузанова. – СПб.: БХВ-Петербург, 2018. – 336 с.: ил.
9. Лузанов П. и др. PostgreSQL для начинающих. – 2017.
10. PostgreSQL : Документация – <https://postgrespro.ru/docs/postgresql>
11. PostgreSQL : Документация [Электронный ресурс]. – Режим доступа : <https://postgrespro.ru/docs/postgresql>
12. Официальная страница русскоязычной документации PostgreSQL и Postgres Pro [Электронный ресурс]. – Режим доступа : <https://postgrespro.ru/docs>
13. Садаладж, П. Д. NoSQL. Новая методология разработки нереляционных баз данных / П. Д. Садаладж, М. Фаулер. – Москва : Вильямс, 2017. – 192 с.

Литература дополнительная

- 1) ISO/IEC 9899-2011[2012] – Programming languages – C, 702 p.
- 2) ISO/IEC 14882-2017[2017] – Programming languages – C++, – 1622 p.
- 4) ГОСТ 19 Система программной документации (серия стандартов).
- 5) ГОСТ 2 Система конструкторской документации (серия стандартов).
- 6) Агальцов, В.П. Базы данных. В 2-х т.Т. 1. Локальные базы данных: Учебник / В.П. Агальцов. - М.: ИД ФОРУМ, НИЦ ИНФРА-М, 2013. - 352 с.
- 7) Агальцов, В.П. Базы данных. В 2-х т. Т. 2. Распределенные и удаленные базы данных: Учебник / В.П. Агальцов. – М.: ИД ФОРУМ, НИЦ ИНФРА-М, 2013. - 272 с.
- 8) Голицына, О.Л. Базы данных: Учебное пособие / О.Л. Голицына, Н.В. Максимов, И.И. Попов. - М.: Форум, 2012. - 400 с.
- 9) Кириллов, В.В. Введение в реляционные базы данных / В.В. Кириллов, Г.Ю. Громов. - СПб.: БХВ-Петербург, 2012. - 464 с.
- 10) Карпова, И.П. Базы данных: Учебное пособие / И.П. Карпова. - СПб.: Питер, 2013. - 240 с.
- 11) Пирогов, В.Ю. Информационные системы и базы данных: организация и проектирование: Учебное пособие / В.Ю. Пирогов – СПб.: БХВ-Петербург, 2009. - 528 с.
- 12) Редько, В.Н. Базы данных и информационные системы / В.Н. Редько, И.А. Бассараб - М.: Знание, 2011. – 602 с.

Основная вычислительная среда

ОС: Linux, MS Windows

СУБД: PostgreSQL, Berkeley DB

«Фреймворки» используем только для реализации пользовательского интерфейса. Все остальное пишем используя базовые API.

Qt

wxWidgets

gtk/gtk++

Что понадобится:

- ПЭВМ с операционной системой Linux или Windows в конфигурации разработки, подключенная к локальной сети;
- установленный штатный дистрибутив СУБД PostgreSQL (сервер, клиент, документация);
- дистрибутив СУБД PostgreSQL в исходных кодах;
- дистрибутив СУБД BerkeleyDB в исходных кодах;
- GNU Compiler Collection (C, C++) версия, достаточная для сборки Postgres и BDB (12++);
- отладчик GDB (для отладки кода на C и C++);
- анализатор использования памяти valgrind.
- библиотеки для разработки пользовательского интерфейса (те самые фреймворки);

Лабораторные работы

1) Лабораторная работа выполняется на основе темы, выбранной в предыдущем семестре.

2) Схема должна содержать основную таблицу, объем данных в которой должен быть достаточен для демонстрации работы пользовательского интерфейса.

Для первичного наполнения таблицы тестовыми данными можно использовать генератор, написанный специально для этой цели.

3) Помимо основной таблицы схема должна содержать не менее трех таблиц-справочников (LUT) для заполнения и верификации соответствующих полей в основной таблице.

4) Не менее, чем два справочника должны быть связаны с основной таблицей отношением «многие ко многим».

5) Все таблицы должны поддерживать добавление, изменение и удаление данных в интерактивном и пакетном режимах.

5) Помимо основного приложения следует разработать утилиты для выгрузки/загрузки БД и конвертации БД из Postgres в BDB.

6) Все лабораторные должны быть правильно оформлены.

7) Спецификации (технические требования).

1 (6) Разработка требований к схеме данных и пользовательскому интерфейсу прикладной программы.

Разработка спецификаций (технических требований) к схеме БД и пользовательскому интерфейсу прикладной программы для интерактивной работы с ней.

Практическое знакомство с прикладным интерфейсом СУБД PostgreSQL.

2 (7) Разработка серверной части прикладной программы.

Программирование клиентской компоненты приложения и утилит.

Программирование серверной части БД.

3 (8) Разработка NoSQL базы данных и спецификаций прикладной программы.

Освоение прикладного интерфейса СУБД BerkeleyDB. Разработка конвертора базы данных Postgres в набор баз данных BerkeleyDB. Адаптация спецификаций приложения.

4 (9) Разработка прикладной программы с использованием NoSQL базы данных.

Программирование прикладной программы с использованием прикладного интерфейса СУБД BerkeleyDB.

Что следует представить для сдачи лабораторной работы

1) Пояснительная записка в формате PDF для каждой из лабораторных работ.

На титульном листе обязательно должно присутствовать название выбранной темы и название лабораторной работы. Номер темы необязателен.

2) Исходный код проекта лабораторной. Исходный код должен быть под контролем системы управления версиями, например git.

Общие требования к оформлению ПЗ

Формат страницы текстовой части А4. Все листы кроме первого нумеруются внизу по центру.

Поля:

левое — 3 см;

правое — 1 см;

верхнее — 2 см;

нижнее — 2 см.

Абзац начинается с абзацного отступа размером в 5 букв 'н' (как учили в школе).

Межстрочный интервал одинарный.

Шрифт абзаца пропорциональный типа serif (Times New Roman, PT Astra Serif, Liberation Serif, ...), кегль 12-14 pt.

Шрифт имен таблиц, полей и прочих идентификаторов по тексту – моноширинный с отличием единицы от буквы 'l' и с плотностью, равной плотности основного шрифта абзаца (Hack, Fira Code, gost, Intel One Mono).

Переносы в тексте включены, в заголовках отключены.

Перечисления

Ненумерованные перечисления (перечни, списки) ничем не обозначаются, либо используется дефис, после которого идет неразрывный пробел.

Именованные перечисления оформляются, как ненумерованные без дефиса, имя при этом выделяется гарнитурой или полужирным.

Нумерованные перечисления нумеруются цифрой со скобкой или малой буквой русского алфавита, после которой идет неразрывный пробел.

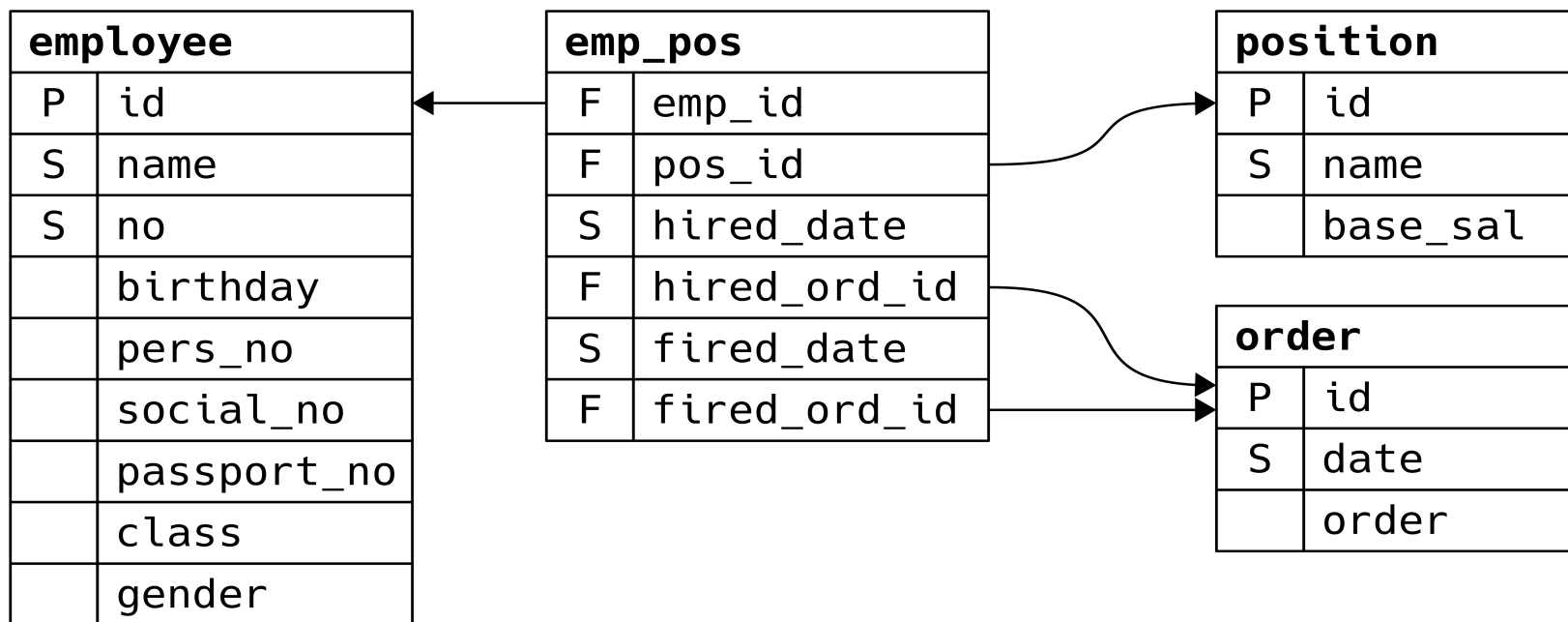
Оформление схем

Толщина линий блоков и соединительных линий – 0,25..0,3 мм.

Надписи в блоках шрифт типа sans (Arial, PT Sans, PT Astra Sans, Liberation Sans, ...) тем же размером, что и текст абзаца.

Коротко об интерфейсе приложения

Пример базы данных



employee – Сотрудник

emp.name – Фамилия, имя и отчество
emp.no – Учетный номер
emp.pers_no – Персональный номер
emp.social_no – Номер соцстраха
emp.passport_no – Номер паспорта
emp.class – Класс служащего

position – Должность

pos.name – Наименование должности
pos.base_sal – Базовая ставка

order – Приказ

ord.date – Дата приказа
ord.order – Текст приказа

emp_pos –

ep.emp_id – Ссылка на сотрудника
ep.pos_id – Ссылка на должность
ep.hired_date – Дата назначения
ep.hired_ord_id – Ссылка на приказ
ep.fired_date – Дата увольнения
ep.fired_ord_id – Ссылка на приказ

Базовые правила

1) Первичные ключи записей не должны нести никакой информации, которая могла бы представлять интерес для пользователя. Это значит:

- первичные ключи не должны отображаться в приложении – они используются только для связи таблиц друг с другом (Consistency/Согласованность);
- ~~- несмотря на уникальность каких-либо информационных полей, они не должны использоваться в качестве первичных ключей записи.~~

2) Внешние ключи, как не несущие никакой информации, которая могла бы представлять интерес для пользователя, не должны отображаться в приложении.

3) Каждая таблица, содержащая информацию, представляющую интерес для пользователя, должна, как минимум, иметь следующие диалоговые окна:

форма TABLE – основное окно для отображения и навигации по таблице;

форма FORM – форма для детального просмотра полей записи таблицы и ее редактирования.

4) Приложение должно иметь элемент Pull-Down меню, содержащее вызов формы TABLE для всех таблиц, являющихся справочниками (LUT – LookUp Table).

5) Любое приложение всегда имеет хотя бы одну основную таблицу. Если основная таблица одна, ее форма TABLE автоматически должна вызываться при старте приложения.

Если основных таблиц несколько, Pull-Down меню приложения должно иметь элемент с вызовами их форм TABLE. При старте такого приложения либо вызывается последняя активная таблица, либо ни одна не вызывается автоматически.

Основная таблица выводится в отсортированном порядке ключа сортировки S.

Если таблица имеет несколько ключей сортировки, должна быть возможность один из ключей назначить ключом по умолчанию.

Если в разных случаях необходимо отображать различный набор столбцов, может использоваться скроллинг по горизонтали (дорого), либо можно иметь несколько вариантов формы TABLE с разным набором столбцов (дешевле).

Особенности формы TABLE

Служащие (редактирование)

Фамилия Имя Отчество	Учетный №	Класс	
			▲
			▼

Локатор

Редактирование Вставка Удаление Печать

1) Форма должна вызываться для работы в следующих режимах:

- просмотр с редактированием;
- просмотр без редактирования;
- просмотр для удаления;
- выбор записи.

2) Заголовок окна должен отображать имя таблицы и режим ее работы.

3) Форма во всех режимах содержит скроллируемый виджет, отображающий диапазон записей таблицы БД в порядке установленной сортировки. Данный виджет должен иметь курсор – полосу выбора, синхронизированную в отношении доступа к выбранной записи в виджете и курсором доступа в таблице БД.

4) Во всех режимах может присутствовать локатор – поле, ввод в которое устанавливает курсор в виджете на соответствующую запись или остается на последней соответствовавшей.

Если перемещение курсора выбора происходит в пределах отображаемой области и устанавливается не на первую/последнюю из строк отображения, отображение не изменяется.

В противном случае содержимое отображения изменяется таким образом, чтобы курсор выбора располагался со второй до предпоследней строки.

Во всех режимах присутствует кнопка «Отмена» (на картинках опущена).

5) В режиме «просмотр с редактированием» присутствуют кнопки (button):

Редактирование;

Вставка;

Удаление;

Отмена;

Печать (не обязательно).

6) В режиме «просмотр без редактирования» присутствуют кнопки:

Просмотр;

Отмена;

Печать (не обязательно).

6) В режиме «просмотр для удаления» присутствуют кнопки:

Удалить;

Отмена;

и чекбокс «каскадное удаление».

7) В режиме «выбор записи» присутствуют кнопки:

Выбрать;

Отмена.

В режиме выбора (режим LUT) в TABLE выводится только та информация, которая требуется для выбора.

Пример формы TABLE с горизонтальным скроллингом

Служащие

[illegible]

Локатор

Просмотр

Редактирование

Вставка

Удаление

Печать

Особенности формы FORM