**15.05.20**

***Уважаемые студенты, внимательно изучите тему урока и сделайте краткий конспект. Он, как бы, подводит итог по изучаемой теме. Теоретические вопросы по этой теме входят в итоговое тестирование по информатике.***

***Студенты, которые до сих пор не сдали практическую работу по* Access, выполните работу и отправьте на проверку**

Отчет о выполненной работе отправьте по электронной почте на yun707@yandex.ru. При отправлении выполненного задания укажите фамилию и группу, в Теме НАИМЕНОВАНИЕ ДИСЦИПЛИНЫ и НАЗВАНИЕ ВЫПОЛНЕННОЙ РАБОТЫ.

**Тема: Система управления базами данных (СУБД). Назначение и основные функции.**

**База данных** – это информационная модель, позволяющая упорядоченно хранить данные о группе объектов, обладающих одинаковым набором свойств.

Программное обеспечение, предназначенное для работы с базами данных, называется **система управления базами данных** **(СУБД)**. СУБД используются для упорядоченного хранения и обработки больших объемов информации.

СУБД организует хранение информации таким образом, чтобы ее было удобно:

просматривать, пополнять, изменять, искать нужные сведения, делать любые выборки,

осуществлять сортировку в любом порядке.

***Классификация баз данных:***

По характеру хранимой информации:

* Фактографические (картотеки)
* Документальные (архивы)

По способу хранения данных:

* Централизованные (хранятся на одном компьютере)
* Распределенные (используются в локальных и глобальных компьютерных сетях).

По структуре организации данных:

* Табличные (реляционные)
* Сетевые
* Иерархические.

Информация в базах данных структурирована на отдельные записи, которыми называют группу связанных между собой элементов данных. Характер связи между записями определяет два основных типа организации баз данных: иерархический и реляционный.

***В иерархической базе*** данных записи упорядочиваются в определенную последовательность, как ступеньки лестницы, и поиск данных может осуществляться последовательным «спуском» со ступени на ступень. Иерархическая база данных по своей структуре соответствует структуре иерархической файловой системы.

***Реляционная база данных***, по сути, представляет собой двумерную таблицу.

Столбцы таблицы называются полями: каждое поле характеризуется своим именем и топом данных. **Поле БД** – это столбец таблицы, содержащий значения определенного свойства.

В реляционной БД используются *четыре основных типов полей:* ***Числовой***, ***Символьный*** (слова, тексты, коды и т.д.), ***Дата*** (календарные даты в форме «день/месяц/год»), ***Логический*** (принимает два значения: «да» - «нет» или «истина» - «ложь»).

Строки таблицы являются записями об объекте. **Запись БД** – это строка таблицы, содержащая набор значения определенного свойства, размещенный в полях базы данных.

Системы управления базами данных позволяют объединять большие объемы информации и обрабатывать их, сортировать, делать выборки по определенным критериям и т. п.

 Современные СУБД дают возможность включать в них не только текстовую и графическую информацию, но и звуковые фрагменты и даже видеоклипы.

**Технология баз данных**

 Под обработкой данных понимается совокупность задач, осуществляющих преобразование массивов данных. Обработка данных включает в себя ввод данных в ЭВМ, отбор данных по каким-либо критериям и параметрам, преобразование структуры данных, перемещение данных , вывод данных в табличном или ином удобном для пользователя виде.

 Под управлением данных понимается весь круг операций с данными, которые необходимы для получения требуемого результата.

 *Пример базы данных*

 Допустим, что в институте, который размещается в разных корпусах, разрабатывается информационная система учебного процесса. Анализируя данную предметную область, разработчики выделили следующие объекты: 1) учебные корпуса; 2) факультеты; 3) кафедры; 4) учебные курсы; 5) преподаватели; 6) студенты; 7) студенческие группы. Данные об этих объектах должны быть включены в БД. На рис. 1 эти данные условно показаны в виде прямоугольников.

Между объектами существуют взаимосвязи, которые также должны найти свое отражение в БД. Так, между факультетами и студенческими группами существуют иерархические отношения включения (группы входят в факультеты, причем факультет объединяет несколько групп, а каждая группа входит ровно в один факультет). Между учебными предметами и студенческими группами существует взаимосвязь, выражаемая понятием «расписание экзаменов». На *рис. 1* взаимосвязи условно изображены в виде стрелок, возможно поименованных.

 

 Рис.1. Условное изображение объектов и их взаимодействие в Базе данных информационной модели ВУЗа

 В реальном мире все указанные выше объекты так или иначе связаны между собой. Однако в БД введена информация лишь о некотором минимальном наборе взаимосвязей, по которым тем не менее можно восстановить остальные взаимосвязи. Например, принадлежность студентов тому или иному факультету можно установить используя взаимосвязи «входит в факультет» и «входит в группу».

**Требования к организации данных.**

 Успешное функционирование БД может быть осуществлено только при выполнении ряда требований к ее организации. К таким основным требованиям можно отнести:

***Расширяемость базы данных.*** База данных должна обладать способностью к расширению, которое может быть за счет:

1) увеличения числа экземпляров однотипных данных, например количества данных о преподавателях;

2) введение в БД новых типов объектов или новых типов взаимосвязей, например между объектами «учебный предмет» и «студент» вводятся связи «оценка».

***Простота работы с базой данных.*** В условиях разработки и эксплуатации крупной системы значение приобретает простота работы с данными БД, т. е. необходимо, чтобы: 1) структура данных была логичной и ясной; 2) операции доступа к данным обладали ясными и четко очерченными функциями; 3) без больших трудозатрат выполнялись различные обслуживающие операции (копирование, перепись с носителя на носитель, расширение базы и др.).

***Целостность базы данных.*** Под целостностью базы данных в общем случае понимается ее готовность к работе.

***Секретность данных.*** Под этим термином понимается в общем случае защита данных от несанкционированного доступа.

**Виды баз данных**

Виды баз данных определяются МОДЕЛЬЮ ДАННЫХ. Ядро любой БД — модель данных, представляющая собой множество структур данных, ограничений целостности и операций манипулирования данных. В настоящее время наибольшее применение получили иерархическая, сетевая и реляционная модели данных.

***Иерархическая модель***. Часто объекты находятся в отношениях, которые принято называть иерархическими: например, отношение часть - целая (например, автомобиль состоит из кузова, двигателя, колес и т.д.); родовидовое отношение (например, автомобили бывают грузовые, легковые и др.); отношения подчиненности (например, ректор — декан и многие другие).

Объекты, связанные иерархическими отношениями, образуют дерево «ориентированный граф», у которого имеется только одна вершина, не подчиненная никакой другой вершине (эту вершину принято называть корнем дерева); любая другая вершина графа подчинена лишь только одной другой вершине.



 Рис. 2. Иерархическая концептуальная схема информационной модели вуза

Все типы связей этой модели принадлежат к виду «один ко многим» и изображаются в виде стрелок. Такой тип связи означает, что одна запись (на ее тип указывает одно острие) соединена со многими подчиненными записями (на их тип указывает двойное острие). В силу единственности пути к любой вершине дерева необходимость в наименовании типов связей в иерархической модели отпадает.

Пример иерархической концептуальной схемы приведен на *рис. 2*. Прямоугольниками на этой схеме изображены типы записей (ФАКУЛЬТЕТ, КОРПУС, ГРУППА, ЭКЗАМЕН). Атрибуты записаны внутри соответствующих прямоугольников. Ключевые атрибуты подчеркнуты.

В иерархической модели ключ для нижних уровней иерархии всегда составной. Он состоит из ключа данной записи и ключей записей, стоящих по иерархии выше данной. Например, полный ключ записи типа ЭКЗАМЕН состоит из атрибутов ШИФР ФАКУЛЬТЕТА, НОМЕР ГРУППЫ и НАИМЕНОВАНИЕ ПРЕДМЕТА.

Основное достоинство иерархических баз данных состоит в экономичном использовании ресурсов памяти и соответственно высоком быстродействии системы. Недостатком является жесткие связи и при изменении модели возникает необходимость в перепрограммировании БД.

***Сетевая модель.*** В сетевой модели данных понятия главного и подчиненных объектов несколько расширены. Любой объект может быть и главным и подчиненным (в сетевой модели главный объект обозначается термином «владелец набора», а подчиненный — термином «член набора»). Один и тот же объект может одновременно выступать и в роли владельца, и в роли члена набора, т.е. каждый объект может участвовать в любом числе взаимосвязей. Схема сетевой модели приведена на рис. 3.

|  |  |
| --- | --- |
| image001 | relacion |
| Рис. 3. Схема сетевой модели данных | Рис.4. Схема реляционной модели данных |

***Реляционная модель.*** В реляционной модели данных объекты и взаимосвязи между ними представляются с помощью таблиц. Взаимосвязи также рассматриваются в качестве объектов. Каждая таблица представляет один объект и состоит из строк и столбцов. В реляционной модели реализуются связи между объектами по принципу каждый с каждым (рис.4.).

В реляционной базе данных каждая таблица должна иметь первичный ключ (ключевой элемент) — поле или комбинацию полей, которые единственным образом идентифицируют каждую строку в таблице. Благодаря своей простоте и естественности представления реляционная модель получила наибольшее распространение в СУБД для персональных компьютеров. Классическим представителем реляционной СУБД является Access.

**Объекты Access**

Исходное окно Access отличается простотой и лаконичностью. Шесть вкладок этого окна представляют шесть видов объектов, с которыми работает программа.

*Таблицы* - основные объекты базы данных. В них хранятся данные. Реляционная база данных может иметь много взаимосвязанных таблиц.

 *Запросы* - это специальные структуры, предназначенные для обработки данных базы. С помощью запросов данные упорядочивают, фильтруют, отбирают, изменяют, объединяют, то есть обрабатывают.

 *Формы* - это объекты, с помощью которых в базу вводят новые данные или просматривают имеющиеся.

 *Отчеты* - это формы "наоборот". С их помощью данные выдают на принтер в удобном и наглядном виде.

 *Макросы* - это макрокоманды. Если какие-то операции с базой производятся особенно часто, имеет смысл сгруппировать несколько команд в один макрос и назначить его выделенной комбинации клавиш.

 *Модули* - это программные процедуры, написанные на языке Visual Basic.

***Режимы работы с Access***

Режим работы с базой выбираются с помощью командных кнопок: ***Открыть, Конструктор, Создать.***

* Кнопка ***Открыть*** открывает избранный объект. Если это таблица, то ее можно просмотреть, внести новые записи или изменить те, что были внесены ранее.
* Кнопка ***Конструктор*** тоже открывает избранный объект, но иначе. Она открывает его структуру и позволяет править не содержимое, а устройство. Если это таблица, в нее можно вводить новые поля или изменять свойства существующих полей. Если это форма, в ней можно изменять или создавать элементы управления.
* Действие командной кнопки ***Создать*** соответствует ее названию. Она служит для создания новых объектов. Этот элемент управления предназначен для проектировщиков базы. Таблицы, запросы, формы и отчеты можно создавать несколькими разными способами: автоматически, вручную или с помощью Мастера.