

Московский государственный технический университет им. Н.Э.
Баумана
Факультет «Информатика и системы управления»
Кафедра ИУ5 «Системы обработки информации и управления»

Лекции

по базам данных

Работу
выполнил:
Цыпышев Т.А.
Группа: ИУ5-41Б
Преподаватель:
Имя
преподавателя

Москва
2024

Содержание

I	Лекция №1 07.02.24	3
1.	Введение в базы данных	4
1.1.	Контрольные мероприятия	4
1.2.	Введение	4
1.2.1.	Реляционная модель данных	4
1.2.2.	Правильное именование сущностей	4
1.3.	Группы операторов SQL	5
1.3.1.	DDL (Data Definition Language)	5
1.3.2.	DML (Data Manipulation Language)	5
1.3.3.	DCL (Data Control Language)	5
1.3.4.	TCL (Transaction Control Language)	6

Глава I

Лекция №1 07.02.24

Глава 1

Введение в базы данных

1.1. Контрольные мероприятия

ЛР(6+1) + РК + Курсовая работа(Продолжение макета)

Распределённый экзамен: 60 = 3 (Удов.) 71 = 4 (Хор.) 85 = 5 (Отл.)

1.2. Введение

База данных - это упорядоченный набор структурированной информации или данных.

Виды баз данных:

- Дореляционные
- Реляционные
- Постреляционные

1.2.1. Реляционная модель данных

Реляционная база данных - это составленная по реляционной модели база данных, в которой данные, занесенные в таблицы, имеют изначально заданные отношения. Сами таблицы в такой базе данных также соотносятся друг с другом строго определенным образом. Реляционные базы данных используют целый комплекс инструментов, которые обеспечивают целостность данных, т. е. их точность, полноту и единообразие.

Данные в реляционной базе данных формируют отношения - двумерные таблицы с информацией о сущностях, т.е. объектах. Строка такой таблицы называется кортежем. Кортежи содержат множество атрибутов одной сущности, категории которых задаются в столбцах.

Типы данных:

- Реляционные
- Постреляционные

1.2.2. Правильное именование сущностей

При выборе имени сущности разумно придерживаться таких правил:

- Имя должно быть существительным (полным, сокращенным либо аббревиатурой) в единственном числе.

- Имя должно быть как можно короче. Оптимально - 2-4 буквы, максимум до 10.
- Имя должно быть уникальным в пределах базы данных.
- Имя должно быть мнемонически понятным проектантам без заглядывания в словарь (но словарь такой хорошо бы составить).
- Желательно, чтобы имена не начинались и не заканчивались на другие имена сущностей.

1.3. Группы операторов SQL

Операторы базы SQL подразделяются на несколько основных групп по признаку типа задач, которые можно решить с их помощью.

1.3.1. DDL (Data Definition Language)

Представляют собой группу операторов для определения данных. Они работают с целыми таблицами. Такие операторы SQL используются в тех случаях, когда нужно внести в базу новую таблицу или, напротив, удалить старую. Они включают в себя следующие командные слова:

- **CREATE** - создание нового объекта в существующей базе.
- **ALTER** - изменение существующего объекта.
- **DROP** - удаление объекта из базы.

1.3.2. DML (Data Manipulation Language)

Эти операторы языка SQL предназначены для манипуляции данными. С их помощью меняется наполнение таблиц. Они позволяют изменять значение строк, столбцов и прочих атрибутов. Такие операторы SQL, например, позволяют удалить информацию о сотруднике, который больше не работает в компании, или исправить данные действующих специалистов. Эти операторы SQL представлены следующими командными словами:

- **SELECT** - позволяет выбрать данные в соответствии с необходимым условием.
- **INSERT** - осуществляют добавление новых данных.
- **UPDATE** - производит замену существующих данных.
- **DELETE** - удаление информации.

1.3.3. DCL (Data Control Language)

Это операторы SQL, предназначенные для определения доступа к данным. С их помощью можно закрыть или открыть для пользователей работу с базой. Такие операторы необходимы, чтобы ограничить кого-либо из сотрудников в доступе к информации или, наоборот, позволить работать с базой новому специалисту.

- **GRANT** - предоставляет доступ к объекту.
- **REVOKE** - аннулирует выданное ранее разрешение на доступ.
- **DENY** - запрет, который прекращает действие разрешения.

1.3.4. TCL (Transaction Control Language)

Предназначен для управления транзакциями, то есть таким сочетанием команд, которые выполняются в определённом алгоритме. Транзакция проведена успешно, если все необходимые команды выполнены пошагово. Если же в какой-либо из них произошёл сбой, то вся операция, включая предыдущие команды, отменяется. Простым и понятным примером таких операторов SQL является проведение банковских платежей.

При этом вы сначала вводите сумму, а затем подтверждаете отправку платежа кодом, который вам присылает банк. Если операция не будет подтверждена, то транзакция отменится автоматически.

- `BEGIN TRANSACTION` - начало транзакции.
- `COMMIT TRANSACTION` - изменение команд транзакции.
- `ROLLBACK TRANSACTION` - отказ в транзакции.
- `SAVE TRANSACTION` - формирование промежуточной точки сохранения внутри операции.