Модель "Сущность-Связь". Классификация сущностей. Виды связей. Ограничения целостности.

Модель "Сущность-Связь":

Модель "Сущность-Связь" - это концептуальная модель, используемая для проектирования баз данных. Она представляет информацию в виде набора сущностей и связей между ними.

Один к одному (One-to-One): Каждая сущность в одном наборе связана с одной сущностью в другом наборе.

Один ко многим (One-to-Many): Каждая сущность в одном наборе связана с несколькими сущностями в другом наборе.

Многие ко многим (Many-to-Many): Множество сущностей в одном наборе связано с множеством сущностей в другом наборе.

Сущности:

Сущность - это объект или объектный класс, о котором хранится информация в базе данных. Сущности обычно представляют реальные или абстрактные объекты, например, "пользователь", "продукт" или "заказ". Каждая сущность имеет уникальный идентификатор, называемый первичным ключом.

Классификация сущностей:

Сущности могут быть классифицированы по их взаимодействию с другими сущностями. Вот некоторые распространенные типы сущностей:

Основные сущности: Такие сущности представляют самостоятельные объекты, которые не зависят от других сущностей.

Зависимые сущности: Эти сущности зависят от других сущностей и существуют только в контексте других сущностей.

Ассоциативные сущности: Такие сущности используются для связи между другими сущностями и содержат атрибуты, специфичные для этой связи.

Ограничения целостности:

Ограничения целостности применяются для обеспечения корректности данных в базе данных. Вот некоторые распространенные ограничения целостности:

Ограничение первичного ключа (Primary Key Constraint): Гарантирует уникальность идентификатора сущности.

Ограничение внешнего ключа (Foreign Key Constraint): Обеспечивает связь между двумя таблицами на основе значений столбцов.

Ограничение уникальности (Unique Constraint): Устанавливает ограничение на уникальность значений в столбце или группе столбцов.

Ограничение проверки (Check Constraint): Позволяет задать пользовательское условие проверки для столбца.

DDL

DDL (Data Definition Language) - язык определения данных, это часть языка SQL, который используется для создания, изменения и удаления структуры базы данных. Он позволяет определить таблицы, столбцы, индексы, ограничения целостности и другие объекты базы данных. Вот некоторые основные команды DDL:

CREATE: Команда CREATE используется для создания новых объектов базы данных, таких как таблицы, индексы, представления и т.д. Пример использования: CREATE TABLE, CREATE INDEX, CREATE VIEW.

ALTER: Команда ALTER позволяет изменять структуру существующих объектов базы данных. Например, можно добавить новый столбец в таблицу, изменить тип данных столбца или удалить объект. Пример использования: ALTER TABLE, ALTER COLUMN.

DROP: Команда DROP используется для удаления объектов базы данных. Это может быть удаление таблицы, индекса, представления и т.д. Пример использования: DROP TABLE, DROP INDEX, DROP VIEW.

TRUNCATE: Команда TRUNCATE используется для удаления всех данных из таблицы, при этом структура таблицы остается неизменной. Она эффективнее, чем использование команды DELETE для удаления всех записей из таблицы. Пример использования: TRUNCATE TABLE.

RENAME: Команда RENAME позволяет переименовывать объекты базы данных, такие как таблицы или столбцы. Пример использования: RENAME TABLE, RENAME COLUMN.

DDL позволяет администраторам базы данных определить и изменять структуру базы данных, что является важным аспектом управления данными.

DML

SELECT, INSERT, UPDATE, DELETE, MERGE

DDL

CREATE, DROP, ALTER, TRUNCATE, RENAME

CONSTRAINTS

NOT NULL, DEFAULT, UNIQUE, PRIMARY KEY, FOREIGN KEY

CHECK

SMALLINT

INTEGER

BIGINT

DECIMAL

NUMERIC

REAL

DOUBLE PRECISION

SMALLSERIAL

BIGSERIAL

CHAR

VARCHAR

TEXT