ОБЕСПЕЧЕНИЕ ЦЕЛОСТНОСТИ ДАННЫХ. ЦЕЛОСТНОСТЬ ОТНОШЕНИЯ. ДОМЕННАЯ, ССЫЛОЧНАЯ, СЕМАНТИЧЕСКАЯ ВИДЫ ЦЕЛОСТНОСТИ

Лектор: PhD, и.о. доцента Турарбек Әсем Турарбекқызы

Результаты обучения

По завершении лекции студент должен:

1. Различать целостность отношения, доменную, ссылочную и семантическую целостность.
2. Объяснять назначение ограничений PRIMARY KEY, UNIQUE, NOT NULL, FOREIGN KEY и CHECK.
3. Применять SQL-ограничения для предотвращения некорректного ввода данных.

Цель лекции

Изучить основные виды целостности данных и механизмы обеспечения достоверности и непротиворечивости информации в реляционных базах данных.

Целостность данных

Основные виды целостности:

1.*Целостность отношений* — гарантирует, что каждая строка в таблице уникальна и полная. Обычно обеспечивается с помощью первичного ключа.

2. *Доменная целостность* — каждое значение должно соответствовать допустимому диапазону, типу данных и ограничениям (например, CHECK, NOT NULL).

3. *Ссылочная целостность* — гарантирует, что значения внешнего ключа ссылаются на существующие значения в другой таблице. Нарушение ссылки недопустимо.

4. *Семантическая целостность* — проверяет соответствие бизнес-правилам и логике предметной области (например, возраст студента не может быть отрицательным).

Целостность данных — это свойство, обеспечивающее достоверность, непротиворечивость и корректность информации в базе данных.

```
CREATE TABLE Students (
    student_id INT PRIMARY KEY,
    name VARCHAR(100)
);
```

Нарушить нельзя:

- нельзя вставить две строки с одним и тем же student_id;
- нельзя вставить строку с student_id = NULL.

Механизмы обеспечения целостности в SQL:

Целостность отношения реализуется через ограничения (constraints), поддерживаемые СУБД:

- **PRIMARY KEY,**
- **UNIQUE,**
- **NOT NULL,**
- **FOREIGN KEY,**
- **CHECK.**

Роль СУБД в обеспечении целостности

```
CREATE TABLE Courses (
    course_id INT PRIMARY KEY,
    course_name VARCHAR(100) NOT NULL,
    credits INT CHECK (credits >= 1 AND credits <=
5)
);
```

- credits должен быть в пределах 1–5,
- course_name не может быть пустым.

Современные СУБД (PostgreSQL, Oracle, SQL Server, MySQL и др.) предоставляют встроенные механизмы для автоматического контроля целостности, включая:

декларативные ограничения (CONSTRAINTS),

триггеры,

процедуры,

пользовательские функции проверки правил.

ЦЕЛОСТНОСТЬ ОТНОШЕНИЯ

Целостность отношения — это набор правил, определяющих допустимые значения и связи между кортежами в отношениях (таблицах).

Эти правила включают:

- обязательное наличие первичных ключей,
- запрет на дублирующиеся строки,
- корректность внешних ключей,
- допустимость нулевых значений (NULL).

ВИДЫ ЦЕЛОСТНОСТИ

Доменная целостность (Domain Integrity)

Определение: доменная целостность требует, чтобы значения в столбце таблицы соответствовали допустимому набору значений, определённому для данного домена (типа данных).

Реализуется через:

Типы данных (INT, VARCHAR, DATE и др.),

Ограничения CHECK (например: CHECK (Возраст >= 0 AND Возраст <= 150)),

Ограничение NOT NULL.

- **Пример**: поле Возраст должно принимать только значения типа INTEGER в диапазоне от 0 до 150.

Ссылочная целостность (Referential Integrity)

Определение: обеспечивается корректностью связей между отношениями, выраженных через внешние ключи (FOREIGN KEY). Ссылочная целостность требует, чтобы каждый внешний ключ указывал на существующий первичный ключ в другой (или той же) таблице.

Механизмы СУБД:

FOREIGN KEY с опциями ON DELETE CASCADE, ON UPDATE SET NULL и др.,

автоматический контроль целостности при вставке, обновлении, удалении.

⚠ Нарушение ссылочной целостности приводит к **"висячим ссылкам"**, что делает данные неполными или неконсистентными.

- **Пример**: в таблице Orders внешний ключ CustomerID должен ссылаться на существующую запись ID в таблице Customers.

Семантическая целостность (Semantic Integrity)

Определение: это набор правил, отражающих *логические, предметные и бизнес-ограничения*, которые должны соблюдаться в конкретной предметной области. Эти правила не могут быть полностью выражены только через стандартные SQL-ограничения и часто реализуются на уровне приложения или триггеров.

Способы реализации:

Триггеры (TRIGGER) на добавление/обновление данных,

Хранимые процедуры (STORED PROCEDURES),

Приложения на уровне бизнес-логики.

- **Пример**:
- сумма процентов скидок не может превышать 100%;
- дата окончания события не может быть раньше даты начала.

Сравнения

Вид целостности	Уровень	Примеры реализации
Доменная	Атрибут	Типы данных, NOT NULL, CHECK
Ссылочная	Между отношениями	FOREIGN KEY, ON DELETE
Семантическая	Предметная область	Триггеры, процедуры, логика приложения

Заключение

Целостность данных обеспечивает корректность и согласованность информации. Ограничения SQL позволяют контролировать допустимые значения, уникальность записей и связи между таблицами. Соблюдение правил целостности повышает надёжность базы данных.

Вопросы для самопроверки

1. Что понимается под целостностью данных?
2. Какие существуют виды целостности?
3. Как обеспечивается доменная целостность?
4. Для чего используется ограничение FOREIGN KEY?
5. Чем семантическая целостность отличается от ссылочной?

Список использованных источников

1. Elmasri R., Navathe S. B. *Fundamentals of Database Systems*. 7th ed. Pearson, 2016.
2. Silberschatz A., Korth H. F., Sudarshan S. *Database System Concepts*. 7th ed. McGraw-Hill, 2020.
3. Connolly T., Begg C. *Database Systems: A Practical Approach to Design, Implementation, and Management*. 6th ed. Pearson, 2015.
4. Date C. J. *An Introduction to Database Systems*. 8th ed. Addison-Wesley, 2003.
5. Ramakrishnan R., Gehrke J. *Database Management Systems*. 3rd ed. McGraw-Hill, 2003.
6. PostgreSQL Global Development Group. *PostgreSQL Documentation: Constraints*. https://www.postgresql.org/docs/current/ddl-constraints.html
7. PostgreSQL Global Development Group. *PostgreSQL Documentation: CREATE TABLE*. https://www.postgresql.org/docs/current/sql-createtable.html
8. Oracle Corporation. *MySQL 8.4 Reference Manual: CREATE TABLE Statement*. https://dev.mysql.com/doc/refman/8.4/en/create-table.html

СПАСИБО ЗА ВНИМАНИЕ