

Евгений Аристов

Первичный и внешний ключи



<https://aristov.tech>



Правила вебинара

<https://aristov.tech>

Задаем вопрос в чат

Вопросы вижу, отвечу в момент логической паузы

Если есть вопрос голосом - поставьте знак ? в чат

Если остались вопросы, можно их задать на следующем занятии

<https://aristov.tech>

Маршрут вебинара

<https://aristov.tech>

1. Первичный ключ (естественный и искусственный)
2. Внешний ключ

Первичный ключ

<https://aristov.tech>

Что нам говорит чатЖПТ:

- 1. Уникальность:** Первичный ключ гарантирует уникальность каждой записи в таблице. Каждая строка должна иметь уникальное значение первичного ключа, что помогает идентифицировать конкретную запись однозначно.
- 2. Упорядоченность:** Первичный ключ также определяет упорядоченность записей в таблице. Это может быть полезно при выполнении запросов или сортировке данных по первичному ключу.
- 3. Индексация:** По умолчанию в большинстве СУБД (систем управления базами данных) создается индекс для первичного ключа. Это повышает производительность при поиске и сортировке данных по первичному ключу.
- 4. Связи между таблицами:** Первичный ключ часто используется для установления связей между таблицами в базе данных, таких как связи один к одному или один ко многим.
- 5. Идентификация записей:** Первичный ключ позволяет идентифицировать и ссылаться на конкретные записи в таблице. Это особенно полезно при выполнении операций обновления, удаления или вставки данных.
- 6. Целостность данных:** Первичный ключ помогает обеспечить целостность данных в таблице, предотвращая появление дублирующихся записей и обеспечивая правильное функционирование базы данных.

<https://aristov.tech>

Первичный ключ (естественный)

<https://aristov.tech>

- ❖ Естественный ключ основан на существующем естественном свойстве или атрибуте. Например, в таблице "Сотрудники" естественным первичным ключом может быть поле "ИНН" (индивидуальный номер налогоплательщика) или комбинация полей, таких как "Фамилия" и "Имя".
- ❖ Преимущества:
 - Естественный ключ дает смысловую информацию и может легче запоминаться.
 - Он использует уже существующие атрибуты данных.
- ❖ Недостатки:
 - Могут возникнуть проблемы с уникальностью, так как данные, которые обычно считаются уникальными, могут фактически дублироваться.
 - Изменение естественного ключа (например, переименование поля) может потребовать изменений в других связанных таблицах.

Первичный ключ (искусственный)

<https://aristov.tech>

- ❖ Искусственный ключ создается специально для целей идентификации записей и не имеет отношения к смысловой информации данных. Например, это может быть автоинкрементируемое поле "ID".
- ❖ Преимущества:
 - Гарантирует уникальность без ссылки на комбинацию атрибутов.
 - Более стабилен, так как не изменяется при изменении других данных.
- ❖ Недостатки:
 - Может быть менее информативным для пользователя из-за отсутствия смысловой информации.
 - Может потребовать дополнительных усилий при установлении связей между таблицами.

Основан на последовательностях - SEQUENCE - разберем на следующей лекции

Классика - тип Integer - 4 байта, от -2млрд до +2млрд. Обычно начинается с 1

- ❖ **Сохранение целостности данных:** Внешние ключи помогают обеспечить целостность данных в базе. Они гарантируют, что данные, хранящиеся в столбце, связанном с внешним ключом, всегда ссылаются на существующие корректные записи в другой таблице.
- ❖ **Связи между таблицами:** Внешний ключ устанавливает связь между двумя таблицами, позволяя создавать связи один к одному, один ко многим или многие ко многим между данными в разных таблицах.
- ❖ **Поддержание целостности и безопасности данных:** Используя внешние ключи, можно гарантировать, что при попытке вставить, обновить или удалить данные, существуют ограничения и правила, которые не нарушат целостность данных во всей базе.
- ❖ **Улучшение производительности:** Индексы, создаваемые для внешних ключей, помогают ускорить выполнение операций соединения таблиц и поиска данных.
- ❖ **Обеспечение согласованности операций:** Использование внешних ключей помогает гарантировать согласованность операций между связанными таблицами. Например, при удалении записи из родительской таблицы, можно автоматически удалить или обновить связанные записи в дочерней таблице.

Основные правила

- ❖ такой же тип данных как и в родительской таблице (на самом деле может быть другой с прямой приводимостью - например integer -> bigint, главное уникальность и не нарушение диапазона)
- ❖ именование - я рекомендую:
 - fk_имя_родительской_таблицы
 - а в родительской таблице первичный ключ просто id
 - таким образом всё однотипно и очевидно
 - обычно в литературе рекомендуют **customerID** и **customerID_fk**
 - получаем масло масляное - **customer.customerID**, **sales.customerID_fk**
 - в моём варианте **customer.id**, **sales.fk_customer** - всё очевидно

Итоги

Итоги

Остались ли вопросы?

Увидимся на следующем занятии

Спасибо за внимание!

Когда дальше и куда?
скину в чате
материалы для бесплатного доступа будут появляться на ютубе

Аристов Евгений