

1. Привилегии баз данных
2. Синонимы
3. Представления

# Привилегии в СУБД могут быть разделены на две категории:

- **Системные привилегии** позволяют пользователям выполнять определенное действие на уровне системы или над конкретным типом объектов. К системным привилегиям можно отнести, например, привилегию на создание таблиц (CREATE TABLE) или привилегию на удаление строки из любой таблицы в базе данных (DELETE ANY TABLE). Большинство системных привилегий должны быть доступны только администраторам и разработчикам приложений.
- **Объектные привилегии** позволяют пользователям выполнять определенные действия с конкретным объектом. К объектным привилегиям, например, относятся привилегии на удаление строки в определенной таблице или запуск определенной процедуры. Объектные привилегии назначаются конечным пользователям для того, чтобы они могли работать с приложениями и базами данных без разрешения администраторов.

# СИНОНИМЫ

— псевдоним (дополнительное имя) для объекта (таблицы, представления, последовательности и пр.).

Они используются для:

- маскировки действительного имени и владельца объекта;
- обеспечения общего доступа к объекту;
- достижения прозрачности местоположения для таблиц, представлений или программных единиц удаленной базы данных;
- упрощения кодирования предложений SQL для пользователей базы данных.

Синтаксис предложения для создания синонима имеет вид:

***CREATE [ OR REPLACE ] [ PUBLIC ] SYNONYM***

***[схема.]синоним***

***FOR [схема.]объект;***

где объектом может быть:

- таблица;
- представление;
- материализованное представление;
- последовательность;
- хранимая процедура (функция или пакет);
- СИНОНИМ.

Синоним может быть общим (PUBLIC) или личным.

Обычный пользователь может создать личный синоним, который будет доступен только этому пользователю.

Администраторы баз данных чаще всего создают общие синонимы, благодаря которым объекты базовых схем становятся доступными для общего пользования всех пользователей базы данных

Преимуществом общедоступных синонимов является то, что они могут создаваться и поддерживаться в одном месте.

Если во время создания синонима определена схема, то пользователям не нужно указывать имя схемы при выполнении запросов по отношению к таблице.

# Представление

— это пустая именованная таблица, определяемая перечнем тех столбцов таблиц и признаками тех их строк, которые хотелось бы в ней увидеть.

Представление является как бы "окном" в одну или несколько базовых таблиц.

Представление создается с помощью предложения CREATE VIEW (создать представление).

Синтаксис:

```
CREATE [OR REPLACE] VIEW имя_представления  
{[(столбец [,...])]}  
AS предложение_select  
[WITH CHECK OPTION];
```



При выполнении этого предложения создается представление (виртуальная таблица) с именем, указанным в *имя\_представления*.

Фраза *OR REPLACE* позволяет заменить существующее представление с именем *имя\_представления* на новое.

Список имен столбцов (*столбец [...]*), количество которых должно совпадать с количеством столбцов, генерируемых *предложением\_select*, должен быть определен лишь в тех случаях, когда:

- хотя бы один из столбцов *предложения\_select* не имеет имени, так как создается с помощью выражения, SQL-функции или константы;
- два или более столбцов подзапроса имеют одно и то же имя.

Если же список отсутствует, то представление наследует имена столбцов из подзапроса.

Необязательная фраза *WITH CHECK OPTION* (с проверкой) указывает, что для операций *INSERT* и *UPDATE* над этим представлением должна осуществляться проверка, обеспечивающая удовлетворение *WHERE*-фразы подзапроса.

## Задачи, которые решают представления:

1. обеспечение независимости пользовательских программ от изменения логической структуры базы данных при ее расширении и (или) изменении размещения столбцов, возникающего, например, при расщеплении таблиц.
2. представления дают возможность различным пользователям по-разному видеть одни и те же данные, возможно, даже в одно и то же время.
3. от определенных пользователей могут быть скрыты некоторые данные, невидимые через предложенное им представление. Таким образом, принуждение пользователя осуществлять доступ к базе данных через представления является простым, но эффективным механизмом для управления санкционированием доступа

# Разграничение доступа к записям таблицы

- с использованием представлений;
- с использованием процедур, функций, пакетов.

Пользователь работает только со своими строками, например которые он сам и создал.

Доступ пользователя к записям таблицы определяется более сложным образом, например, рядовые сотрудники имеют один уровень доступа, более продвинутые — имеют значительно расширенный доступ, а единицы — обладают правом работы со всей таблицей (мандатный контроль доступа).