Управление привилегиями, пользователи и роли в СУБД Interbase/Firebird

Пользователи и роли

Разграничения прав доступа к БД и таблицам, в клиент-серверных СУБД осуществляется на уровне **пользователей (users)** и **ролей (roles)**. Каждый пользователь имеет **имя** и **пароль**, с помощью которых и осуществляется подключение к базам данных.

Права на создание пользователей есть только у администраторов сервера БД (**SYSDBA** и пользователей с административными привилегиями).

Списки пользователей хранятся в специальной базе на сервере БД. Роли создаются внутри каждой отдельной базы данных. **Роли используются** для назначения привилегий **группам пользователей**.

Создание пользователя

Команда создания пользователя выглядит следующим образом

```
create user имя_пользователя password 'пароль'
    [FIRSTNAME 'имя']
    [MIDDLENAME 'отчество']
    [LASTNAME 'фамилия']
    [GRANT ADMIN ROLE];
```

имя_пользователя и **пароль** указываются обязательно.
'фамилия', 'имя' и **'отчество'** являются необязательными параметрами.
параметр [GRANT ADMIN ROLE] указывает, что создаваемый пользователь будет иметь привилегии администратора БД (**SYSDBA**)

Создание пользователя

Перед созданием пользователя, необходимо подключиться к серверу с правами администратора.

```
connect 'имя_бд' user 'SYSDBA' password 'masterkey';
create user TestUser password 'test'
    firstname 'Тестовый'
    middlename 'Пользователь' lastname 'Системы';
commit;
```

В данном примере создается пользователь **TestUser**.
Это простой пользователь, т.е. от его имени можно создать и управлять базой данных, но невозможно администрировать сервер (создавать новых пользователей и т.п.)

Создание ролей

Команда создания роли выглядит следующим образом

```
create role имя;
```

После того, как роль создана, ее можно назначить группе пользователей командой **grant**

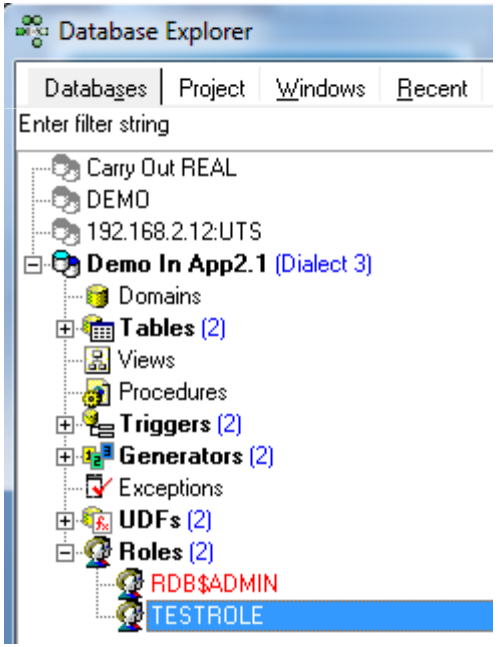
```
grant роль1 [,роль2,…] to имя1 [,имя2,…] ;
```

Например:

```
create role TestRole;
grant TestRole to TestUser ;
```

Создание ролей

Роль можно создавать только **при наличии активного соединения** с базой данных (т.е. после выполнения команды **Connect**).
Для проверки, необходимо зарегистрировать базу данных и соединиться с ней.
На вкладке ролей (Roles) появиться роль **TESTROLE**.



Назначение привилегий

Распределение прав доступа между пользователями и ролями к объектами базы данных осуществляется командой **grant**.

```
grant привилегии on таблица to объекты [WITH GRANT OPTION];
```

объекты в данном случае – это список пользователей и ролей

таблица – это таблица или просмотр, к которому устанавливается доступ

привилегии – это список действий над таблицей, который будет разрешен для указанных объектов.

опция **WITH GRANT OPTION** показывает, что объекты получают права на изменение привилегий над таблицей.

Назначение привилегий

привилегии –могут иметь следующие значения (или их комбинацию):

SELECT – выполнение запроса select к таблице

DELETE – разрешение на удаление записей

INSERT - разрешение на добавление записей

UPDATE [(col [, col …])] - разрешение на изменение указанных столбцов

REFERENCES [(col [, col …])] – разрешение на доступ к внешним ключам (по ссылке FOREIGN KEY или REFERENCES)

EXECUTE ON PROCEDURE proc – разрешение на выполнение хранимой процедуры

Назначение привилегий

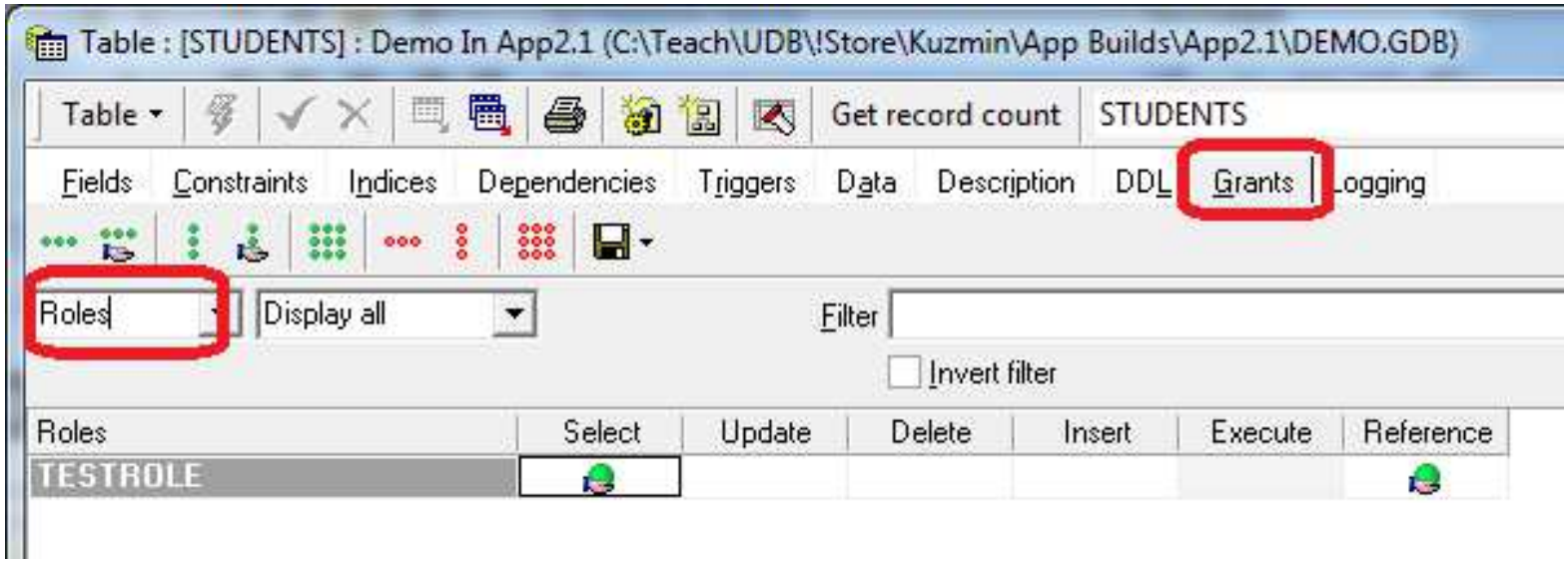
Установим привилегии на просмотр данных в таблицах **GROUPS** и **STUDENTS** для роли TestRole.

```
grant SELECT,REFERENCES on GROUPS to TestRole;
grant SELECT,REFERENCES on STUDENTS to TestRole with
grant option;
grant EXECUTE ON PROCEDURE GroupCount To TestRole;
```

Здесь назначаются права на просмотр таблицы **GROUPS**, без права изменения привилегий и парава на просмотр таблицы **STUDENTS**, с возможностью изменения привилегий доступа для роли **TestRole**.
Учитывая, что пользователь TestUser получил роль **TestRole**, при подключении к он сможет просматривать данные, но не будет иметь права на их редактирование.
Так же будут получены права на выполнение хранимой процедуры **GroupCount**.

Назначение привилегий

Посмотреть установленные привилегии для ролей можно на вкладке **Grant -> Roles**, после соединения с БД и открытия таблицы.



Проверка установленных привилегий

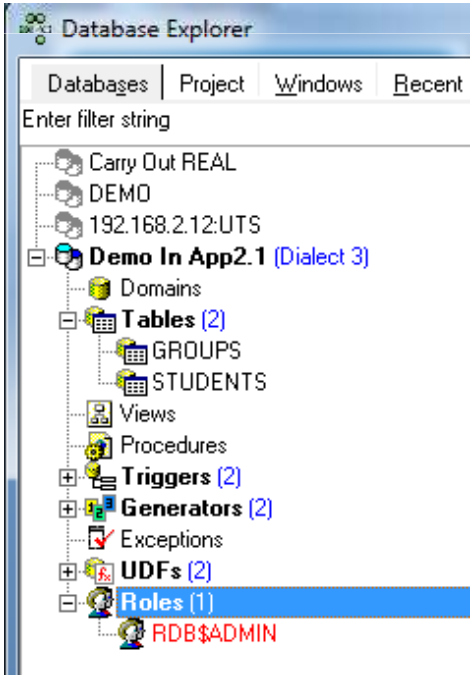
Для проверки правильности установленных ролей, подключимся к базе данных от имени TestUser и попробуем выполнить различные запросы (**Tools->SQL Editor**).
Для проверки, необходимо выполнить запросы select, insert, delete, update и execute procedure.

Удаление ролей и пользователей

По окончании эксперимента, удалим созданного пользователя и роль

```
drop user имя_пользователя;
drop role имя_роли;
```

Для проверки подключимся к базе данных и откроем вкладку **Roles**. После удаления, роль **TestRole** должна исчезнуть.



Утилита gsec

Управлять пользователями можно с помощью утилиты **gsec.exe**, входящей в комплект Firebird.

```
gsec.exe опции команда параметры;
```

опции задают имя пользователя и пароль администратора
команда задает действие, которое необходимо произвести
параметры задают дополнительные опции для команды