

## Creating Users

### Overview:

This document provides details of how to create, alter and delete Users within the Oracle database.

### Restricted Use:

For security reasons, some permissions have not been provided to students to avoid accidental deletion or modification of other user accounts.

### Security Implications for User Creation:

When creating Users in a database, (or for any application or system), you should always adhere to the principle of least privilege. Simply stated, this means that a user should be given no more privileges than they need to successfully complete their job or tasks. Although it may seem like it is convenient to give a developer DBA access and escalated privileges it is not a good idea and may cause security issues down the road.

Accessing a database requires a user to connect to the database instance using a valid user name defined in the database. In this class, you are asked to immediately change your password on first login. You are also on a restricted list to reduce the possibility of attacks. As discussed earlier this week, you can create profiles outlining password policies and other security guidelines. Next week we will be able discussing how roles can be used to further control user access and maintenance.

### Create User:

To create a User, you must have the Create User system privilege.

Figure 1 provides Oracle's diagram showing the major syntax for creating a User. Notice users can be authenticated using password, externally or globally. Externally and Globally authentications are associated with certificate and active directory accounts. For this class, we will create users with a password.

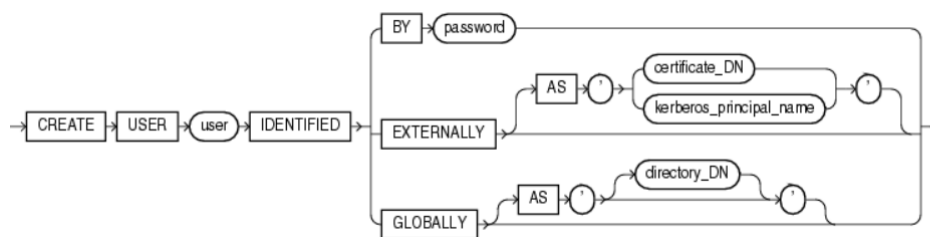


Figure 1 Create User Syntax

Figure 2 exposes the additional clauses available when creating a user. These options provide the ability to align with existing Profiles and Tablespaces, expire the initial password and set quotas. Each of these options help secure your database. For example, expiring the password is a best practice to ensure the user is in control of their credentials. Quotas help ensure a rogue User doesn't slow down your system by potentially filling up the tablespace they were assigned.

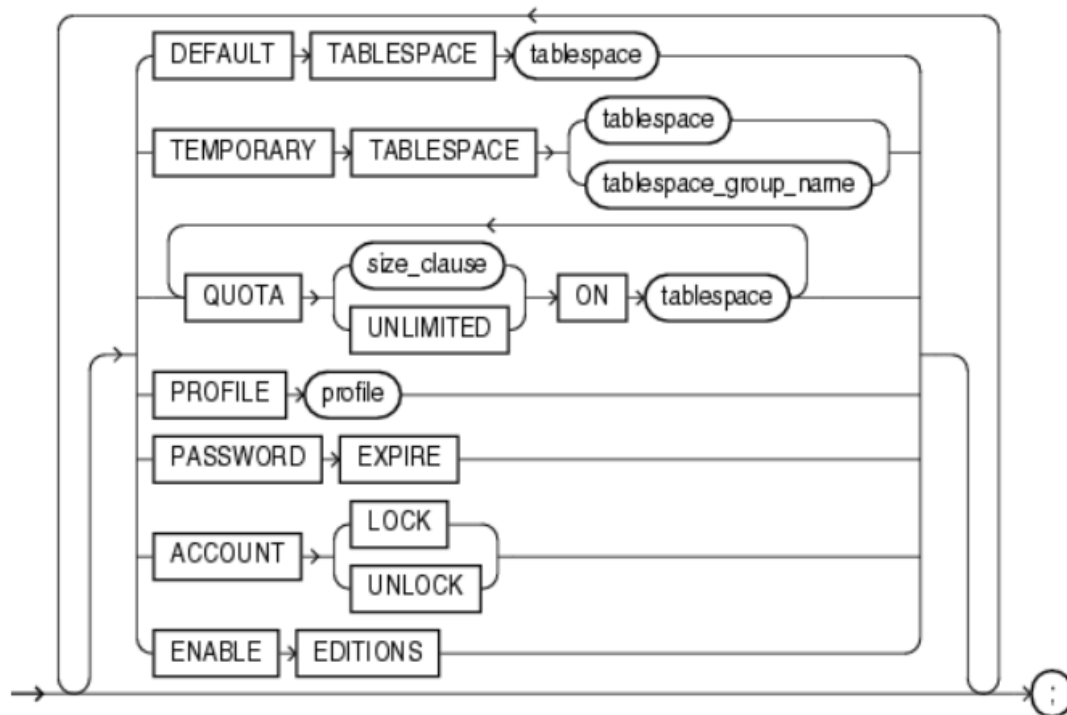


Figure 2 Additional Create User Options

The following are additional details about most of the Create User clauses:

- Password - Passwords are case sensitive and may contain numbers, letters and special characters. When creating an account within SQL Developer, often double quotes are needed to enclose the password if a special character is used.
- Default Tablespace – A default tablespace should be listed. If you don't provide one the user's objects are stored in the database default tablespace. Which may default to the SYSTEM tablespace. From a performance standpoint, it is not a good idea to use the SYSTEM tablespace as your user's tablespace as the SYSTEM has multiple processes and activities to take care of.
- Temporary Tablespace – This is where sorts and other activities occur for the user. If you don't specify a temporary tablespace, the default temporary space is used and possibly the SYSTEM tablespace. It is a best practice to assign a temporary tablespace for the users.
- Quota – Used to specify the maximum amount of space the user can allocate in the tablespace. UNLIMITED is a possibility but not recommended. Business rules and requirements may need to be reviewed to see how much space is required so the user is given enough but not too much.
- Profile - The profile limits the amount of database resources the user can use. Be sure to pay careful attention to the Password Complexity rules. The Oracle system does check to make sure password requirements specified are followed.
- Password Expire – This clause should be used when creating new accounts in most cases. This setting forces the user to change the password before the user can log in to the database.
- Account Lock – Forced the Account to be locked. This prevents the user access until such time the account is Unlocked. This is a good feature to use for old accounts that you want to keep but no longer use. However; be careful that default passwords are changed to prevent someone

with escalated privileges from unlocking the account and then logging in with the default password.

As an example create User statement, consider the following simple Create statement that creates a user named student1 on the default tablespace of Users, Temporary tablespace assignment of temp, limited by the Default profile and they must change their password on their first login.

```
CREATE USER student1
  IDENTIFIED BY "es32Q92G!s"
  DEFAULT TABLESPACE Users
  QUOTA 5M on Users
  TEMPORARY TABLESPACE temp
  PROFILE Default
  PASSWORD EXPIRE;
```

Figure 3 shows the results of successfully executing the Create User statement.

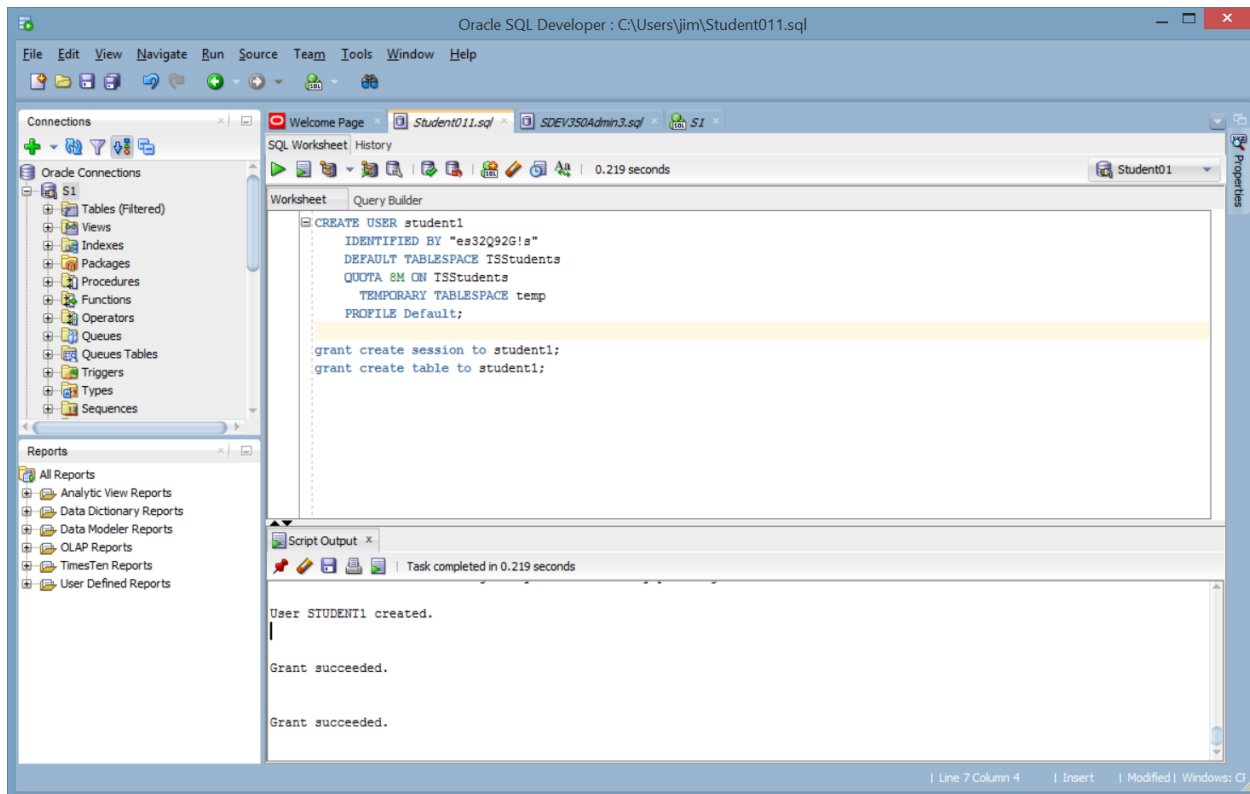


Figure 3 Creating a Student

Since you have the ability to Create Profiles, you are welcome to use a Profile you previously created.

You can use the Data dictionary views to verify the users were created. For example, the following query will show the username, last login and expiration date for a user named 'STUDENT1':

```
select username, last_login, expiry_date from dba_users where username =
'STUDENT1';
```

Figure 4 shows the results of running this query using a student account in SQL Developer.

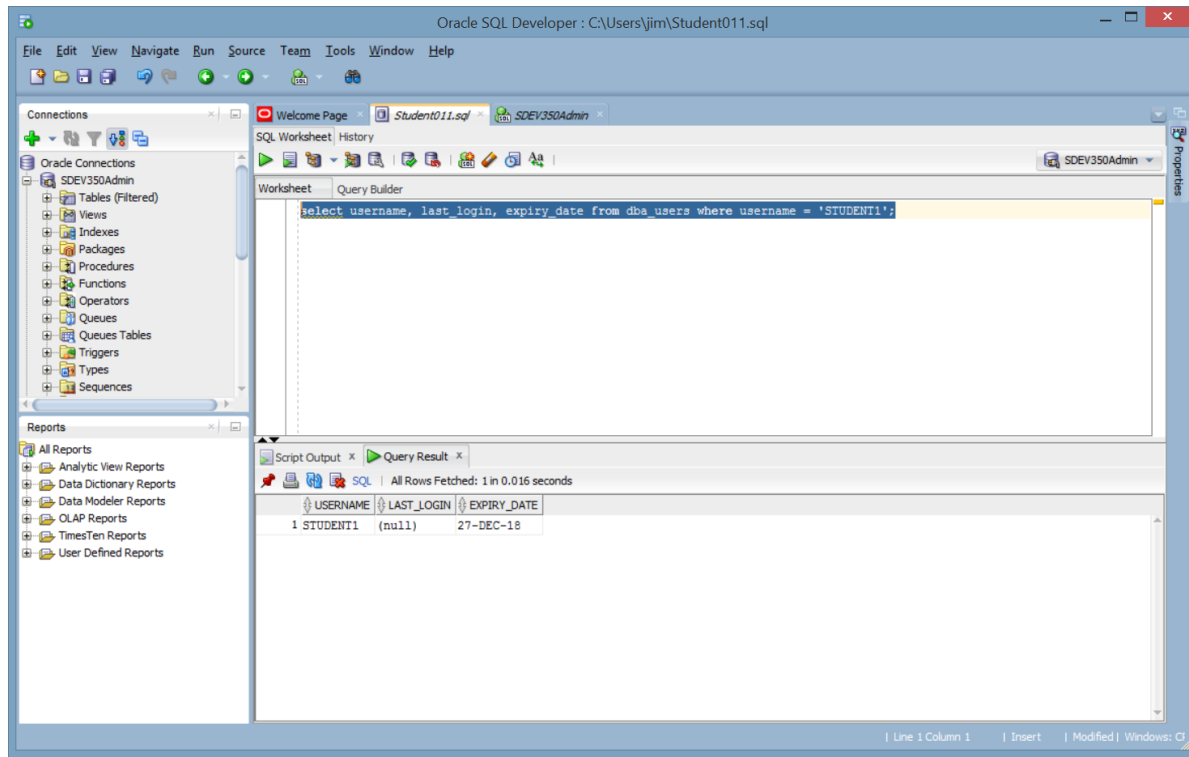


Figure 4 Verifying the User Creation with the Data dictionary

Similar to previous data dictionary view queries, be careful with the case of your string. Although the username is Student1 in this example, the UpperCase equivalent (e.g. STUDENT1) is stored in the data dictionary view. If you don't use upper case in your query, you won't receive the expected results.

### Altering a User:

The Alter User statement can be used to alter an existing User. One of the most common uses of this statement is to change the password. For example:

```
Alter user Student1 identified by "qs62A92G_c"  
Password expire;
```

You do need the Alter User permissions for a successful Alter call. If provided with the Alter User privileges, running this statement would yield a successful response as shown in figure 5.

Note: Be aware that Microoft word often changed quotes to curly quotes (""") versus typical quotes (""") which won't work in database commands. If needed copy from a text editor to avoid this issue.

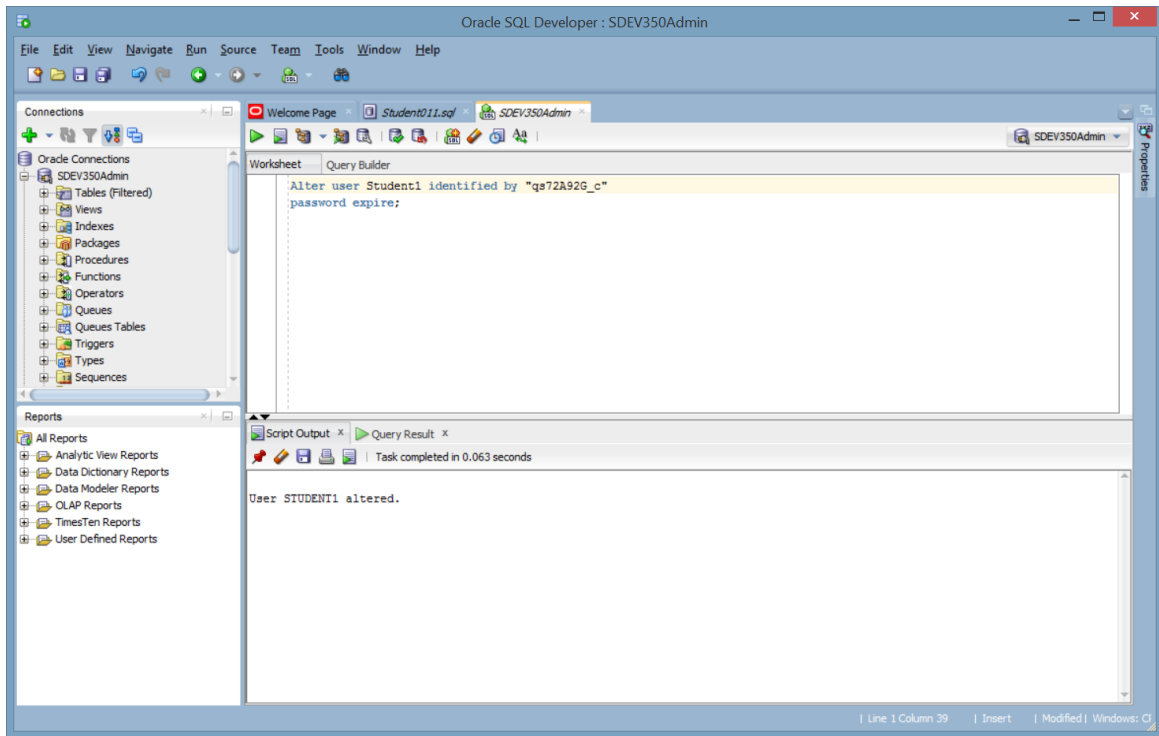
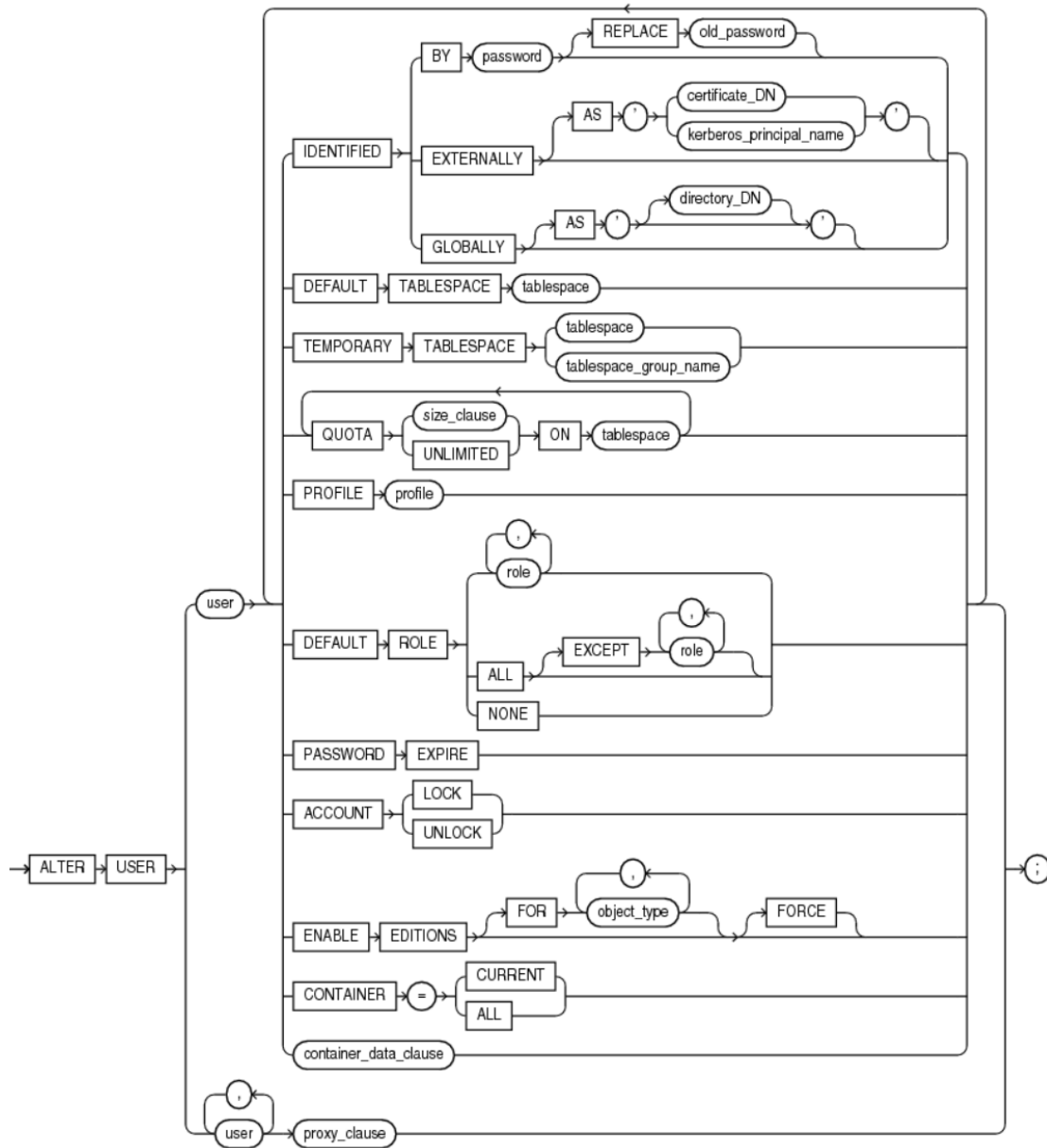


Figure 5 Altering a User

As shown in figure 6, the Alter user syntax has multiple options and clauses and can yield a lengthy and complex statement. For example, the following alter user statement Alters the Student1 user and assigns the default tablespace to Users, the temporary tablespace to TSStudents and the Profile to my\_profile :

```
ALTER USER Student1
  DEFAULT TABLESPACE Users
  TEMPORARY TABLESPACE TSStudents
  PROFILE my_profile;
```



Description of the illustration "alter\_user.gif"

Figure 6 Alter User Syntax

### Drop a User:

With the proper Drop User permissions, a user can be dropped (deleted). You do need the Drop User permissions for a successful Drop call. **If a user has already created objects (e.g. tables), the cascade clause must be used**

```
Drop User Student1;
```

Successfully dropping the profile will remove it from the DBA\_Users view as shown in Figure 7.

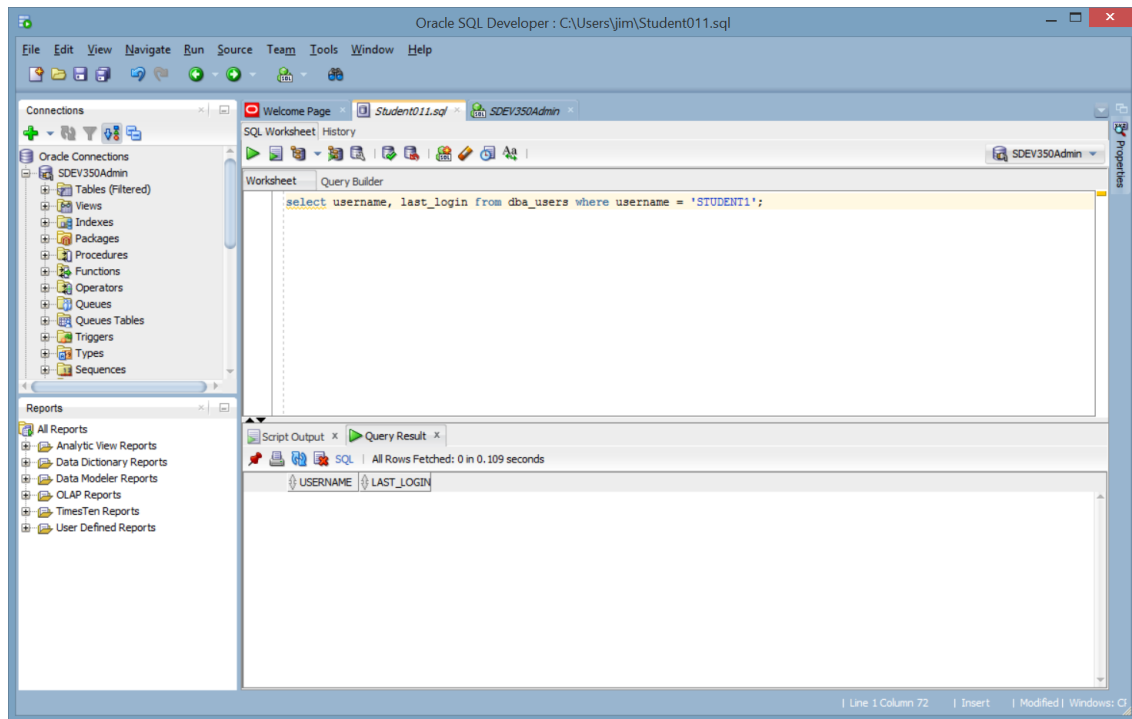


Figure 7 Verifying the Student was Dropped

### Granting Permissions to your Users:

When you create a user, by default they don't have any permissions. In fact, they don't have any permissions to even connect to the database. Therefore, be sure to provide permissions to create session. This gives the user the ability to connect. In addition, if you want them to create a table, you would grant them the create table permission. The following SQL statements are needed to allow the create session and create table permissions:

```
grant create session to student1;  
grant create table to student1;
```

In this example, create session and create table permissions have been granted to user student1. Student accounts have limited permissions, but they do have the proper permissions to grant these privileges using the above statements.