



SQL Server: Grant/Revoke Privileges

Learn how to **grant and revoke privileges** in SQL Server (Transact-SQL) with syntax and examples.

Description

You can GRANT and REVOKE privileges on various database objects in SQL Server. We'll look at how to grant and revoke privileges on tables in SQL Server.

Grant Privileges on Table

You can grant users various privileges to tables. These permissions can be any combination of SELECT, INSERT, UPDATE, DELETE, REFERENCES, ALTER, or ALL.

Syntax

The syntax for granting privileges on a table in SQL Server is:

```
GRANT privileges ON object TO user;
```

privileges

The privileges to assign. It can be any of the following values:

Privilege	Description
SELECT	Ability to perform SELECT statements on the table.
INSERT	Ability to perform INSERT statements on the table.
UPDATE	Ability to perform UPDATE statements on the table.
DELETE	Ability to perform DELETE statements on the table.
REFERENCES	Ability to create a constraint that refers to the table.
ALTER	Ability to perform ALTER TABLE statements to change the table definition.
ALL	ALL does not grant all permissions for the table. Rather, it grants the ANSI-92 permissions which are SELECT, INSERT, UPDATE, DELETE, and REFERENCES.

object

The name of the database object that you are granting permissions for. In the case of granting privileges on a table, this would be the table name.

user

The name of the user that will be granted these privileges.

Example

Let's look at some examples of how to grant privileges on tables in SQL Server.

For example, if you wanted to grant SELECT, INSERT, UPDATE, and DELETE privileges on a table called *employees* to a user name *smithj*, you would run the following GRANT statement:

```
GRANT SELECT, INSERT, UPDATE, DELETE ON employees TO smithj;
```

You can also use the ALL keyword to indicate that you wish to grant the ANSI-92 permissions (ie: SELECT, INSERT, UPDATE, DELETE, and REFERENCES) to a user named *smithj*. For example:

```
GRANT ALL ON employees TO smithj;
```

If you wanted to grant only SELECT access on the *employees* table to all users, you could grant the privileges to the public role. For example:

```
GRANT SELECT ON employees TO public;
```

Revoke Privileges on Table

Once you have granted privileges, you may need to revoke some or all of these privileges. To do this, you can run a revoke command. You can revoke any combination of SELECT, INSERT, UPDATE, DELETE, REFERENCES, ALTER, or ALL.

Syntax

The syntax for revoking privileges on a table in SQL Server is:

```
REVOKE privileges ON object FROM user;
```

privileges

It is the privileges to assign. It can be any of the following values:

Privilege	Description
SELECT	Ability to perform SELECT statements on the table.
INSERT	Ability to perform INSERT statements on the table.
UPDATE	Ability to perform UPDATE statements on the table.
DELETE	Ability to perform DELETE statements on the table.
REFERENCES	Ability to create a constraint that refers to the table.
ALTER	Ability to perform ALTER TABLE statements to change the table definition.

Privilege	Description
ALL	ALL does not revoke all permissions for the table. Rather, it revokes the ANSI-92 permissions which are SELECT, INSERT, UPDATE, DELETE, and REFERENCES.

object

The name of the database object that you are revoking privileges for. In the case of revoking privileges on a table, this would be the table name.

user

The name of the user that will have these privileges revoked.

Example

Let's look at some examples of how to revoke privileges on tables in SQL Server.

For example, if you wanted to revoke DELETE privileges on a table called *employees* from a user named *anderson*, you would run the following REVOKE statement:

```
REVOKE DELETE ON employees FROM anderson;
```

If you wanted to revoke ALL ANSI-92 permissions (ie: SELECT, INSERT, UPDATE, DELETE, and REFERENCES) on a table for a user named *anderson*, you could use the ALL keyword as follows:

```
REVOKE ALL ON employees FROM anderson;
```

If you had granted SELECT privileges to the public role (ie: all users) on the *employees* table and you wanted to revoke these privileges, you could run the following REVOKE statement:

```
REVOKE SELECT ON employees FROM public;
```

Copyright TechOnTheNet.com