

## Rechteverwaltung

### GRANT und REVOKE

In SQL ist es möglich neue Benutzer anzulegen und denen Rechte zu geben, so dass sie nur auf bestimmte Teile einer Datenbank Zugriff haben.

#### Verwaltung von Benutzern:

Einen neuen Benutzer anlegen:

**CREATE USER Benutzername IDENTIFIED BY 'Passwort';**

(Das IDENTIFIED BY kann auch weggelassen werden, wenn kein Passwort nötig ist)

Einen Benutzer löschen:

**DROP USER Benutzername;**

Den Benutzer wechseln:

**EXIT;**

(darüber wird der Benutzer ausgeloggt)

**mysql.exe -u Benutzername -p;**

(Falls der Benutzer kein Passwort hat kann das -p weggelassen werden)

#### Anlegen von Rechten:

Ein neues Recht vergeben:

**GRANT Privileg ON Tabelle TO Benutzername;**

(Privileg ist ein Befehl. Möglich sind: SELECT, DELETE, UPDATE, INSERT und viele weitere oder ALL PRIVILEGES für alle Rechte.)

Die Rechte eines Benutzers anzeigen lassen:

**SHOW GRANTS FOR Benutzername;**

(Wenn das „FOR Benutzername“ weggelassen wird, werden die eigenen Rechte angezeigt.)

Ein Recht wegnehmen:

**REVOKE Privileg ON Tabelle FROM Benutzername;**

(Achtet darauf, dass es hier FROM heißt. Im GRANT-Befehl heißt es TO !)

## Rechteverwaltung

### Beispiel

Legen Sie einen lokalen Benutzer an, der Anne heißen soll. Anne soll sich nur über das sehr sichere Passwort 123 einloggen können.

**CREATE USER Anne@localhost IDENTIFIED BY '123';**      Im Admin-Account.  
(das @localhost sorgt für einen lokalen Benutzer!)

Wechsel Sie in den Account von Anne und lassen Sie sich die Tabelle Kosten anzeigen.

**EXIT;**  
**mysql.exe -u Anne -p;**  
**Password: 123**  
**USE NoName;**      ← Hier kommt es zum Fehler. Anne hat keine Rechte.

Anne soll jetzt die Tabell Kosten sehen und Einträge verändern dürfen.

**GRANT Select, Update ON Kosten TO Anne@localhost;**      ← Im Admin-Account.

Wechseln Sie in den Account von Anne und lassen Sie sich ihre Rechte anzeigen. Danach versuchen Sie die Tabelle Kosten auszugeben.

(Wechsel wie oben.)  
**SHOW GRANTS;**      ← Die Ausgabe sieht so aus:

```
MariaDB [(none)]> show grants;
+-----+
| Grants for Anne@localhost |
+-----+
| GRANT USAGE ON *.* TO 'Anne'@'localhost' IDENTIFIED BY PASSWORD |
| GRANT SELECT, UPDATE ON `noname`.`kosten` TO 'Anne'@'localhost' |
+-----+
```

(Die erste Zeile ist das Auswählen der Datenbank. In der zweiten Zeile steht das Recht SELECT und UPDATE auf der Tabelle Kosten in der Datenbank NoName für Anne@localhost.)

**USE NoName;**      ← Hier gibt es keine Fehlermeldung mehr.  
**SELECT \* FROM Kosten;**      ← Wird korrekt angezeigt.

Anne braucht das Update-Recht nicht. Nehmen Sie es ihr wieder weg.

**REVOKE Update ON Kosten FROM Anne@loclahost;**      ← Im Admin-Account.

Anne soll eine Zeile updaten.

(Wechsel wie oben.)  
**USE NoName;**  
**UPDATE Kosten SET Betrag=0 WHERE KostenID=1;**      ← Zugriff verweigert!

## Rechteverwaltung

### Aufgaben

1) Erklären Sie was im folgenden Code passiert: CREATE USER safepass@localhost IDENTIFIED BY 'Karl';	*
2) Legen Sie folgende lokalen Benutzer an: a) Peter mit Passwort whats7 b) Rechnungsstelle mit Passwort 223intern c) 778 mit Passwort 778 Beurteilen Sie danach die Nützlichkeit dieser Benutzer.	* * * **
3) Vergeben Sie folgende Rechte: a) Select-Recht auf die Tabelle Rechnungen für den Benutzer Meyer. b) Select-, Update- und Insert-Recht auf die Tabelle Rechnungen für Schmidt. c) Delete-Recht auf die Tabelle Anwalt für Schmidt d) Schmidt soll auch die Tabelle Rechnungen ändern dürfen. Schauen Sie nach welchen Befehl es dafür gibt und geben Sie ihm das Recht.	* ** * **
4) Wir wollen die Rechte des Benutzers Lara@localhost ändern. Erkläre in eigenen Worten was hier passiert. a) GRANT Insert ON 'kanzlei'. 'anwalt' TO 'Lara'@'localhost'; b) GRANT Delete ON 'kanzlei'. 'anwalt' TO 'Lara'@'localhost'; c) REVOKE Select ON 'kanzlei'. 'anwalt' TO 'Lara'@'localhost'; d) GRANT Update ON 'kanzlei'. 'anwalt' FROM 'Lara'@'localhost'; Bei den Befehlen c) und d) meckert SQL. Warum?	* * * ** **
5) Karl hat den SHOW GRANTS-Befehl eingegeben und erhält folgende Ausgabe: GRANT USAGE ON *.* TO 'Karl'@'localhost' GRANT SELECT, UPDATE ON 'kanzlei'. 'anwalt' TO 'Karl'@'localhost' Danach hat er folgenden Befehl eingegeben: UPDATE Rechnungen SET Betrag=50; Was wird er als Antwort erhalten? Warum?	**
6) Schauen Sie sich die Rechte von Thompson und Miller an. Wer darf mehr tun?	**
7) Entziehen Sie folgende Rechte: a) Entziehen Sie Thompson das Insert-Recht in der Tabelle Rechnungen. b) Nehmen Sie Miller das Select-Recht auf die Tabelle Anwalt weg. Schauen Sie sich jetzt seine Rechte an. Erklären Sie Ihre Entdeckung!	* * **

<p>8) Leichter Auftrag:</p> <p>Legen Sie einen neuen Benutzer Weiler mit dem Passwort ‚true‘ an und geben Sie ihm das Select und das Update-Recht auf die Tabelle Rechnungen. Dann löschen Sie ihn wieder.</p>	*
<p>9) Leider verstorben:</p> <p>Leider haben wir heute früh erfahren, dass unser Kollege Smith verstorben ist. Smith war ein guter Anwalt und war der einzige, der neue Anwälte einstellen durfte. Diese Aufgabe übernimmt jetzt Kollege Miller. Bitte geben Sie Miller alle Rechte die er dafür benötigt. Schauen Sie bei Smith nach, welche das sind. Dann entziehen Sie Smith alle seine Rechte und löschen Sie seinen Account. Danke.</p>	**
<p>10) Die Kanzlei wurde von einem Trojaner angegriffen. Er hat folgenden Code als Admin eingegeben:</p> <pre>show grant for Smith@localhost; show grants for Miller@localhost; show grants for Thompson@localhost; revoke select on kanzlei.rechnungen from Smith@localhost; revoke select on kanzlei.rechnungen from Miller@localhost; drop user Thompson@localhost; create user spion identified by 'xxx'; grant all privileges on kanzlei.rechnungen to spion; exit; mysql -u spion -p; xxx use kanzlei; select * from rechnungen; delete * from rechnungen; exit;</pre> <p>Die Kanzlei möchte wissen was hier passiert ist. Wie kann das rückgängig gemacht werden? Welcher Schaden entstand der Kanzlei?</p>	***
<p>11) Neuer Anwalt:</p> <p>Die Kanzlei hat einen neuen Anwalt eingestellt. Er wurde am 3.10.1981 geboren, heißt Bruno und ist auf Unternehmensrecht spezialisiert. Sie sollen ihm einen neuen Account anlegen. Als Passwort wählt er immer den Namen seiner Frau Trudy. Da er schon in seiner alten Firma mit Rechnungen zu tun hatte wird er auch hier daran arbeiten. Geben Sie ihm die Rechte, die er dafür braucht. Um die Rechnungen ordnungsgemäß zu erstellen muss er auch die Anwälte sehen können. Sorgen Sie dafür. Falls es Ihnen hilft: Er spielt gerne Golf.</p>	***