Granting Privileges in MySQL

pluralsight
hardcore developer training

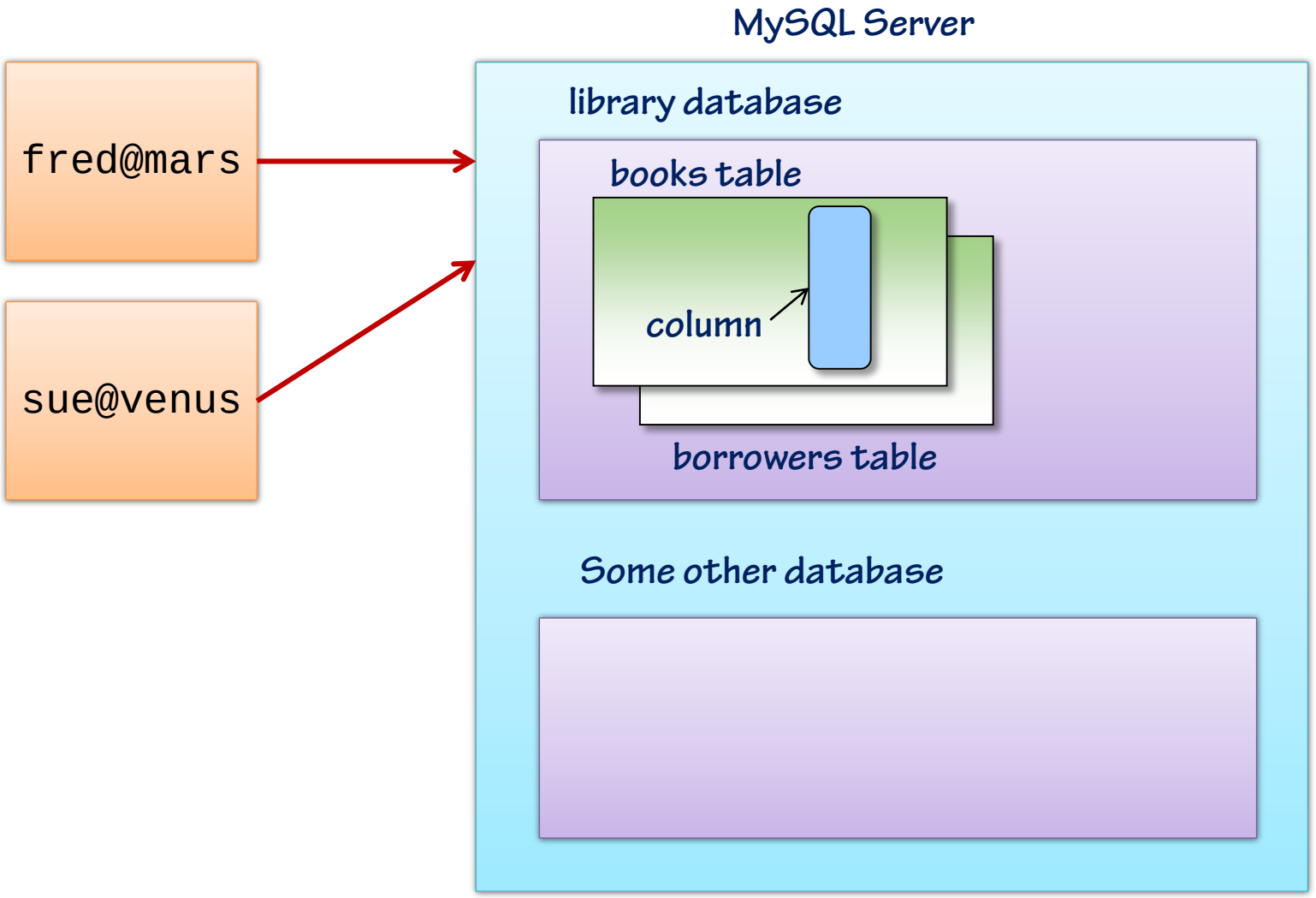
Lesson Objectives

- **In this lesson we will learn about the privilege-granting features of MySQL**
  - Creation of user accounts
  - Restricting login based on client machine
  - Granting access to database objects
- **We will create user accounts with limited privilege for our library application**
  - Apply the principle of least privilege

Why Bother?

- **The root MySQL account we have used so far has full privileges**
  - So why bother creating more accounts?
- **Principal of least privilege**
  - Only grant a user or program those privileges that are necessary for it to do its job
- **Improving security**
  - Restrict the damage that a "compromised" program can do (e.g. SQL injection)
  - Prevent a user's inadvertent actions from causing excessive damage

Levels of Access Control



Object Privileges

- These privileges control how an account can interact with an existing database

Privilege	Description
SELECT	Retrieve rows from tables
UPDATE	Update table rows
INSERT	Add new rows
DELETE	Delete rows
EXECUTE	Execute stored procedures and functions

Database Privileges

- These privileges control how an account can modify the database itself

Privilege	Description
`CREATE`	Create databases and tables
`DROP`	Remove databases, tables, etc.
`ALTER`	Alter tables and indexes
`INDEX`	Create or drop indexes
`CREATE ROUTINE`	Create a stored procedure or function
`TRIGGER`	Create or drop triggers

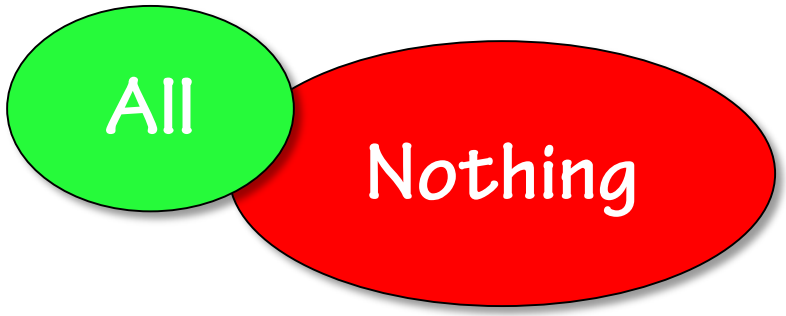
Administrative Privileges

- These privileges control how an account can administer the MySQL server

Privilege	Description
`CREATE USER`	Manage user accounts
`GRANT OPTION`	Grant privileges to other accounts
`SHOW DATABASES`	Display all the database names
`SHUTDOWN`	Shut down the server
`LOCK TABLES`	Lock tables (e.g. during backup)

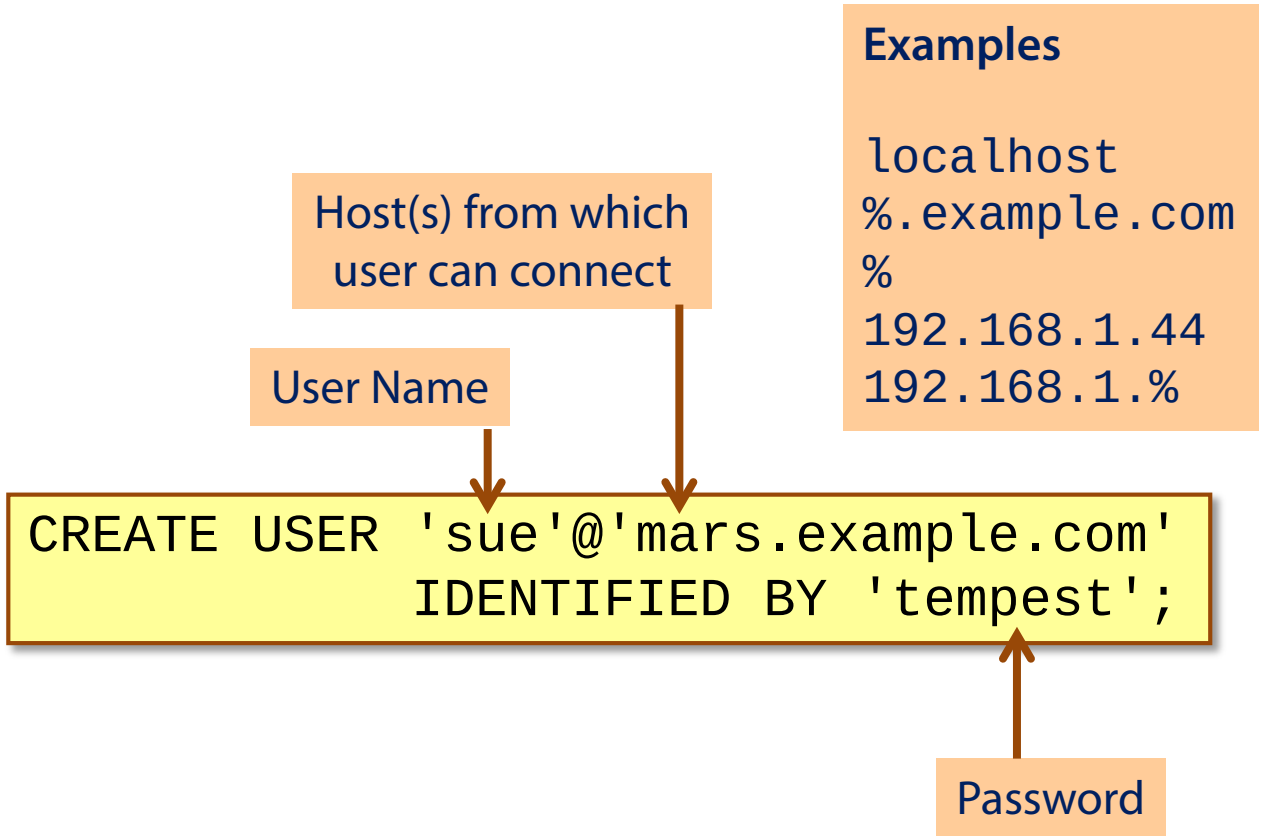
All or Nothing

- **The 'ALL' privilege allows all operations**
  - Except GRANT, which needs to be enabled explicitly
- **The ' USAGE' privilege allows no operations**
  - Useful when the command syntax requires a privilege name but no privilege change is intended



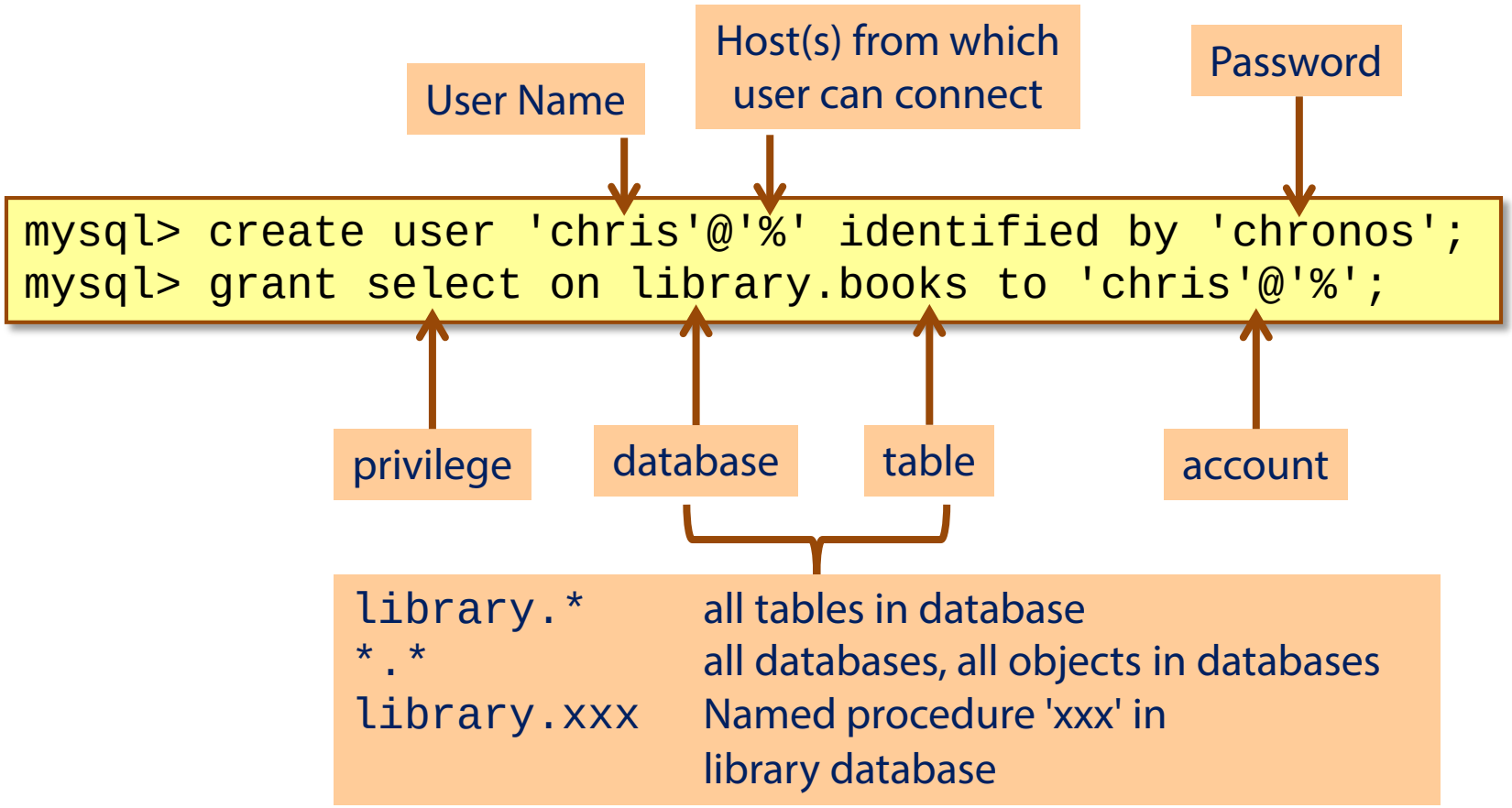
Creating an Account

- **Accounts are created with the CREATE USER statement**
  - An account specifies both a user name and a client host location
  - '%' and '_' wildcards can be used in host matching

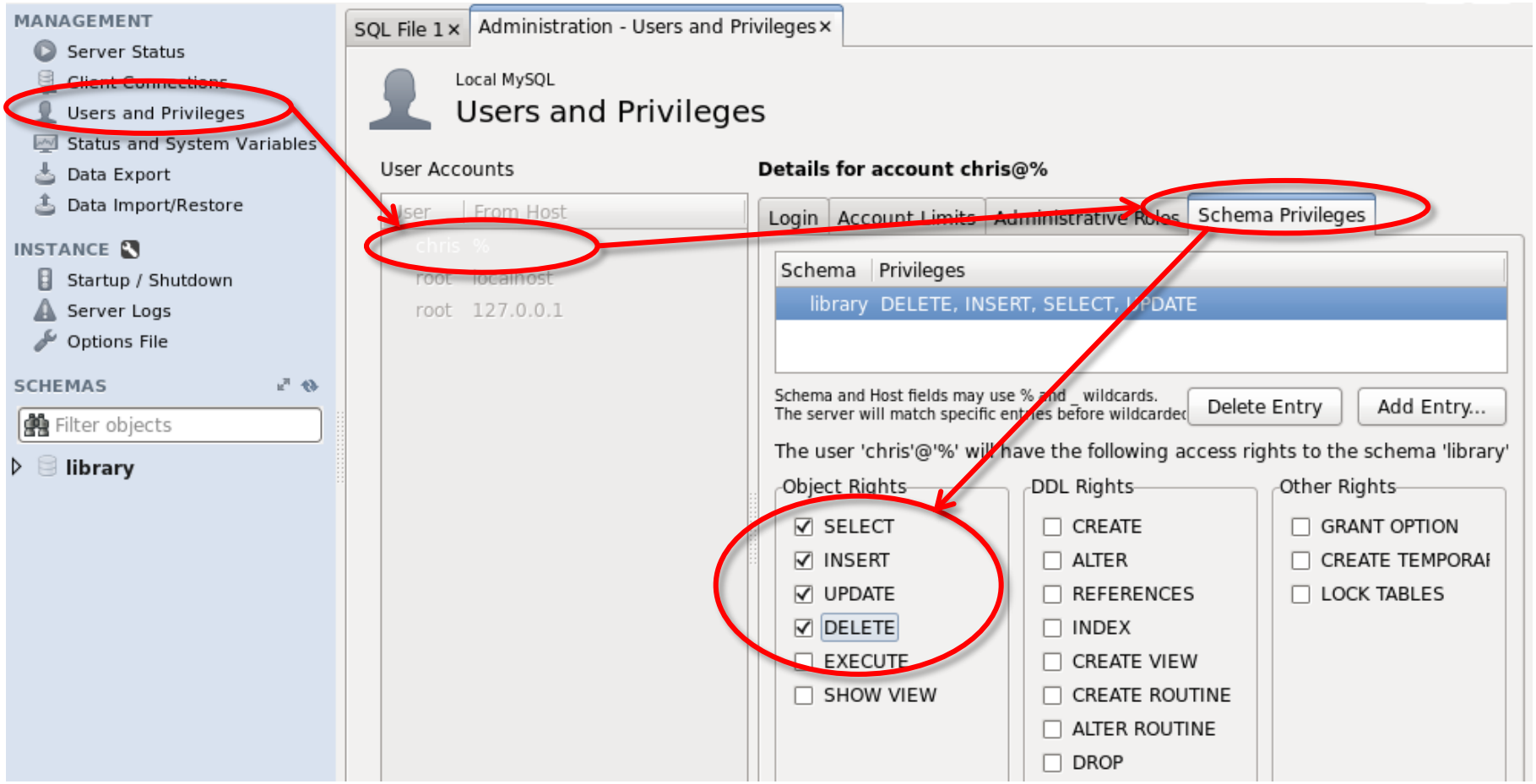


Granting Privilege from the Command Line

- The `grant` command is used to assign privileges



Granting Privilege using MySQL Workbench



Displaying Privileges

- Privileges may be displayed using the `show grants` command:

```
mysql> show grants for chris;
+-----------------------------------------------------------------+
| Grants for chris@%                                              |
+-----------------------------------------------------------------+
| GRANT USAGE ON *.* TO 'chris'@'%' IDENTIFIED BY PASSWORD |
|  '*BC564BD66372EAF77C090367C027C5B8C3CE7075'                    |
| GRANT SELECT, INSERT ON `library`.`books` TO 'chris'@'%' |
+-----------------------------------------------------------------+
```

Revoking Privilege

- Privileges may be revoked using syntax similar to the grant command

```
mysql> revoke delete,update on library.* from 'chris'@'%';
```

MySQL Accounts for Library Application

- **We will define four accounts for our library application:**

`root`: Create and administer database

`librarian`: Add new books and borrowers

`assistant`: Check books out and in

`borrower`: Browse books

Lesson Summary

- **We have**
  - Created user accounts for MySQL
  - Granted privileges at the database, table and column level
- **Using**
  - MySQL command line tool
  - MySQL Workbench
- **And for the library application**
  - Defined and created user accounts
  - Granted privileges to follow the principle of least privileges

Coming up in Lesson 9:

Putting it all together

… in which we create a complete web application for our library