2025/09/04 08:56 1/2 LU01d - DB

## LU01d - DB

Damit wir im späteren Verlauf des Semesters Objekte persistieren können, benötigen wir eine Datenbank.

Folgende Datenbankmanagementsysteme (DBMS), beziehungsweise Database Engines sind in der Software-Entwicklung gebräuchlich:

- MySQL weit verbreitetes Open-Source-DBMS, häufig für Webanwendungen im Einsatz. https://www.mysgl.com/
- **PostgreSQL** objektrelationales Open-Source-DBMS mit starkem Fokus auf Standards und Erweiterbarkeit. https://www.postgresql.org/
- **Oracle Database** kommerzielles, leistungsfähiges Enterprise-DBMS mit umfassenden Features für große Unternehmen. https://www.oracle.com/database/
- **Microsoft SQL Server** relationales DBMS von Microsoft, oft in Kombination mit dem .NET-Ökosystem genutzt. https://www.microsoft.com/en-us/sql-server
- **SQLite** leichtgewichtiges, dateibasiertes DBMS, das häufig in mobilen Apps und Embedded-Systemen verwendet wird. https://sqlite.org/
- **H2** schlankes, Java-basiertes In-Memory-DBMS, oft in Testszenarien und kleineren Projekten eingesetzt. https://www.h2database.com/html/main.html
- **MongoDB** dokumentenorientiertes NoSQL-DBMS, das JSON-ähnliche Datenstrukturen speichert und hohe Flexibilität bietet. https://www.mongodb.com/

In diesem Modul werden wir PostgreSQL einsetzen. Laden Sie dazu die Version **17** herunter und installieren Sie diese.

https://www.postgresgl.org/download/

Die Installation kann mit folgendem Befehl überprüft werden.

```
psql --version
```

```
Peter@PETER-LAPTOP MINGW64
$ psql --version
psql (PostgreSQL) 17.5
```

Der Standardbenutzer heisst postgres. Damit könnten Sie bereits einzelne DQL-Statements ausführen.

Für die lokale Entwicklung erlauben wir den lokalen Zugriff **ohne Passwort**, indem wir die Datei pg\_hba.conf entsprechend anpassen.

psql -U postgres -c "SHOW hba file;"



Nur für lokale Entwicklung, niemals für produktive Datenbanken!

Ändern Sie in der Datei pg\_hba.conf folgende Zeilen.

| Old: `host sha-256`    | all | all | 127.0.0.1/32 | scram- |
|------------------------|-----|-----|--------------|--------|
| New: `host             | all | all | 127.0.0.1/32 | trust` |
| Old: `host<br>sha-256` | all | all | ::1/128      | scram- |
| New: `host             | all | all | ::1/128      | trust` |

Damit die Änderung wirksam wird, muss der Datenbankserver mit Admin-Rechten neu gestartet werden.

```
net stop postgresql-x64-17 && net start postgresql-x64-17
```

Zeigen Sie die aktuellen Benutzer an.

```
psql -U postgres -c "SELECT * FROM pg_catalog.pg_user;"
```

Erstellen Sie einen neuen Benutzer mit einem Namen und Passwort Ihrer Wahl.

```
psql -U postgres -c "CREATE USER localuser WITH LOGIN;"
```

Dieser Benutzer sollte nun aufgelistet sein.

```
psql -U postgres -c "SELECT * FROM pg_catalog.pg_user;"
```

Sie können nun eine Datenbank anlegen und anstatt postgres Ihren eigenen Benutzer als Besitzer definieren. Damit trennen Sie die Serveradministration von der Datenbankadministration.

```
psql -U postgres -c "CREATE DATABASE localdb OWNER localuser;"
```

Diese Datenbank ist nun bereit und kann aus einem Backend aufgerufen werden.

From:

https://wiki.bzz.ch/ - BZZ - Modulwiki

Permanent link:

https://wiki.bzz.ch/de/modul/ffit/3-jahr/java/learningunits/lu01/db?rev=175555553

Last update: 2025/08/19 00:18



https://wiki.bzz.ch/ Printed on 2025/09/04 08:56