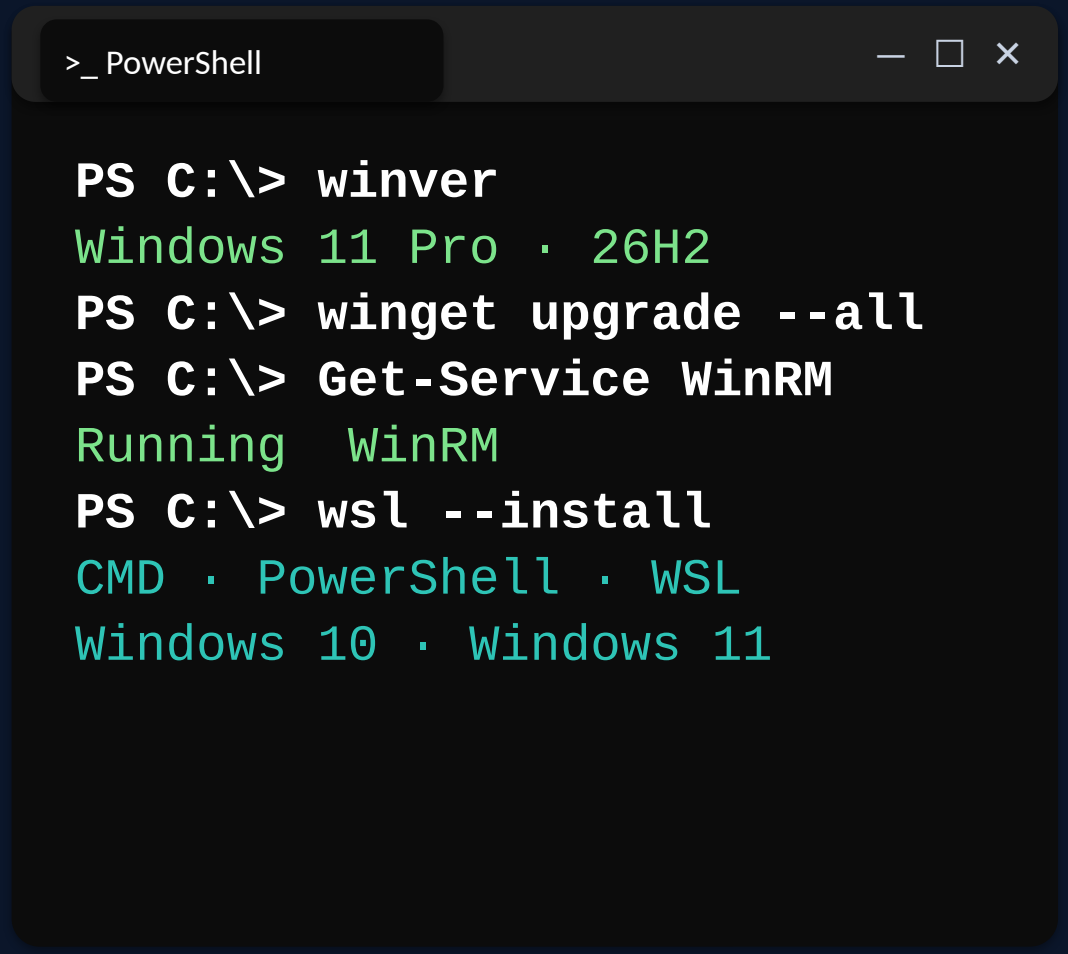


LESSON 7 OF 12

Windows 10 & 11

Desktop · Workstation · Laptop

Databases on Windows: SQL Server, SQLite, MySQL & PostgreSQL

Full Course · Windows 11 26H2 · Windows 10 22H2

What You Will Learn in Lesson 7

- Choosing a database: embedded file vs database server
- SQLite: a complete database in one file
- SQL Server Express and Developer editions, SSMS and sqlcmd
- PostgreSQL and MySQL / MariaDB on Windows
- Database services, ports and the firewall
- Backups: BACKUP DATABASE, pg_dump and mysqldump
- Connecting from Python

Which Database?

Database	Type	Cost / license	Great for
SQLite	Embedded file	Free, public domain	Apps, prototypes, data science
SQL Server Express	Server	Free (size-limited)	Small apps, learning T-SQL
SQL Server Developer	Server	Free, not for production	Full-feature development
PostgreSQL	Server	Free, open source	Web apps, analytics, GIS
MySQL / MariaDB	Server	Free community editions	Web apps, WordPress, LAMP skills

SQLite: A Database in One File

```
PS C:\> winget install -e --id SQLite.SQLite
PS C:\Lab> sqlite3 school.db
sqlite> CREATE TABLE students (id INTEGER PRIMARY KEY, name TEXT, grade INTEGER);
sqlite> INSERT INTO students (name, grade) VALUES ('Maryam', 95), ('Ahmad', 88);
sqlite> SELECT name, grade FROM students ORDER BY grade DESC;
Maryam|95
Ahmad|88
sqlite> .tables
sqlite> .quit
```

SQL Server on Windows

Piece	What it is
Download	microsoft.com/sql-server: choose Express or Developer; 'Basic' install
Instance names	Express: localhost\SQLEXPRESS; Developer: localhost (default)
SSMS	SQL Server Management Studio: the main graphical tool
VS Code + MSSQL extension	Query SQL Server from your editor
sqlcmd	Command line client: winget install -e --id Microsoft.Sqlcmd
Authentication	Windows authentication (recommended) or SQL logins

T-SQL with sqlcmd

```
PS C:\> sqlcmd -S localhost\SQLEXPRESS -E -Q "SELECT @@VERSION"

PS C:\> sqlcmd -S localhost\SQLEXPRESS -E
1> CREATE DATABASE School;
2> GO
1> USE School;
2> CREATE TABLE Students (Id INT IDENTITY PRIMARY KEY, Name NVARCHAR(50), Grade INT);
3> INSERT INTO Students (Name, Grade) VALUES (N'Maryam', 95), (N'Ahmad', 88);
4> SELECT * FROM Students WHERE Grade > 90;
5> GO
```

PostgreSQL on Windows

```
PS C:\> winget search PostgreSQL           # pick the current major version
# The installer sets a password for the 'postgres' user and port 5432
PS C:\> Get-Service postgresql*
PS C:\> psql -U postgres -h localhost
postgres=# CREATE DATABASE school;
postgres=# \c school
school=# CREATE TABLE students (id SERIAL PRIMARY KEY, name TEXT, grade INT);
school=# INSERT INTO students (name, grade) VALUES ('Maryam', 95);
school=# \dt          -- list tables      \q  -- quit
```

MySQL and MariaDB on Windows

```
PS C:\> winget search MySQL ; winget search MariaDB
# Installer: set the root password; service listens on port 3306
PS C:\> Get-Service MySQL*, MariaDB*
PS C:\> mysql -u root -p
mysql> CREATE DATABASE school;
mysql> CREATE USER 'student'@'localhost' IDENTIFIED BY 'Use-A-Strong-Password!';
mysql> GRANT ALL PRIVILEGES ON school.* TO 'student'@'localhost';
mysql> SHOW DATABASES;
mysql> exit
```

Database Services, Ports and Safety

Server	Windows service	Default port
SQL Server	MSSQL$SQLEXPRESS / MSSQLSERVER	1433 (TCP)
PostgreSQL	postgresql-x64-<version>	5432
MySQL / MariaDB	MySQL<version> / MariaDB	3306
Keep it local	Listen on localhost on a laptop or desktop	Do not open the port
If you must share	Firewall rule for one subnet; TLS; strong passwords	Never to the internet

Backups for Every Database

```
# SQL Server (T-SQL)
PS C:\> sqlcmd -S localhost\SQLEXPRESS -E -Q "BACKUP DATABASE School TO DISK='C:\Backup\School.bak'"

# PostgreSQL
PS C:\> pg_dump -U postgres -h localhost -F c -f C:\Backup\school.dump school

# MySQL / MariaDB
PS C:\> cmd /c "mysqldump -u root -p school > C:\Backup\school.sql"

# SQLite: copy the file while no program is writing to it
PS C:\> Copy-Item C:\Lab\school.db C:\Backup\
```

Connect from Python

```
import sqlite3
con = sqlite3.connect(r"C:\Lab\school.db")
for name, grade in con.execute("SELECT name, grade FROM students"):
    print(name, grade)

# Other servers (pip install in your venv):
#   SQL Server -> pyodbc or mssql-python     PostgreSQL -> psycopg
#   MySQL      -> mysql-connector-python     All of them -> SQLAlchemy, pandas.read_sql
import pandas as pd
print(pd.read_sql("SELECT * FROM students", con))
```

Summary & Homework

- SQLite = one file; SQL Server, PostgreSQL, MySQL = services on ports
- SQL Server Express (production, limited) vs Developer (full, non-production)
- Use Windows authentication or separate app users — never root/sa for apps
- Keep databases on localhost; back up and test restores
- Homework: create the School database in SQLite and one server
- Homework: back it up, delete it, and restore it

Thank You! 🙏

Free videos & PDFs: www.MyWebUniversity.com

Slides for this lesson: MyWebUniversity.com/presentations

YouTube: @MyWebUniversity · @MyWebUniversity-Dari

Wahidullah Lutfy — Founder and Director of IT of MyWebUniversity, and CS teacher

Book: The Direct Path to Linux Ubuntu (Amazon · Barnes & Noble)

Learn it, then pass it on.