Exercise 3: PostgreSQL with Docker

Duration: ~15 minutes

Objective

Run the same HR database on PostgreSQL using Docker. Verify that your SQL queries work identically on a production-grade database engine.

Part 1: Start PostgreSQL (~3 min)

```
cd bts-bdp-exercises/s5
docker compose up -d
```

This starts PostgreSQL 16 on `localhost:5432` and automatically creates the schema and seed data (via the mounted SQL files).

Verify it's running:

```
docker compose ps
```

> ***Troubleshooting:*** *If port 5432 is already in use (e.g., you have a local PostgreSQL installed), either stop the local service (* `brew services stop postgresql` *) or change the port mapping in* `docker-compose.yml` *to* `"5433:5432"` *and use port 5433 in all subsequent steps.*

Part 2: Connect VSCode Database Client to PostgreSQL (~3 min)

1. In VSCode Database Client, click the **+** button to create a new connection
2. Select **PostgreSQL**
3. Fill in:
   - **Host**: `localhost`
   - **Port**: `5432`
   - **Database**: `hr_database`
   - **Username**: `postgres`
   - **Password**: `postgres`
4. Click **Connect**

Part 3: Run Your Queries Again (~5 min)

Open a new SQL file connected to PostgreSQL and run the same queries from Exercise 2.

Try at least these:

```
-- Employees with department
SELECT e.first_name, e.last_name, d.name AS department
FROM employee e
INNER JOIN department d ON e.department_id = d.id
ORDER BY d.name, e.last_name;
```

```sql
-- Average salary by department
SELECT
    d.name AS department_name,
    ROUND(AVG(e.salary)::numeric, 2) AS avg_salary,
    COUNT(e.id) AS employee_count
FROM department d
INNER JOIN employee e ON d.id = e.department_id
GROUP BY d.name
ORDER BY avg_salary DESC;
```

> ***Note:*** *PostgreSQL requires* `::numeric` *for the* `ROUND()` *function because it cannot round a* `double precision` *value directly. SQLite handles this automatically. This is one of the few syntax differences you'll encounter between engines.*

Part 4: Run the Python Scripts Against PostgreSQL (~4 min)

```
pip install psycopg2-binary sqlalchemy

# Raw SQL with psycopg2
python scripts/create_hr_raw.py postgresql

# SQLAlchemy ORM
python scripts/create_hr_sqlalchemy.py postgresql
```

Compare the output with the SQLite versions from Exercise 1.

Notice how `create_hr_sqlalchemy.py` uses the **exact same ORM code** for both databases. Only the connection URL changes.

Cleanup

When you're done, stop PostgreSQL:

```
docker compose down
```

To also delete the data volume:

```
docker compose down -v
```

Deliverable

- Screenshot of a query result in VSCode Database Client connected to PostgreSQL
- Screenshot of `create_hr_raw.py postgresql` output