LangMem/LangGraph: Postgres Memory Store Setup

==========

DBA-style checklist to get a dedicated database + role (with password) ready for use as a LangMem/LangGraph memory store.

0) Confirm server & psql

```
brew services start postgresql@14
psql -d postgres
```

1) Create a dedicated login role (user) with password

```
-- In psql, connected to the "postgres" database:
CREATE ROLE langmem_app
```

WITH LOGIN

PASSWORD 'CHANGEME' -- pick a strong password

NOSUPERUSER NOCREATEDB NOCREATEROLE NOINHERIT

CONNECTION LIMIT -1;

```
-- (optional) set a default search_path this app should use
ALTER ROLE langmem_app SET search_path TO memory, public;
```

2) Create a dedicated database owned by that role

```
CREATE DATABASE langmem_db OWNER langmem_app;
```

3) Enable pgvector in that database (needed for semantic search)

```
\c langmem_db
CREATE EXTENSION IF NOT EXISTS vector;
```

4) (Optional) Put app objects in a dedicated schema

```
-- still in langmem_db
CREATE SCHEMA IF NOT EXISTS memory AUTHORIZATION langmem_app;

-- make the DB default to that schema for all sessions
ALTER DATABASE langmem_db SET search_path TO memory, public;
```

5) (Optional hardening)

```
-- If you don’t want everyone to be able to create in public:
REVOKE CREATE ON SCHEMA public FROM PUBLIC;
```

6) Quick smoke test

```
psql "postgresql://langmem_app:CHANGEME@localhost:5432/langmem_db" -c "SELECT current_user, current_dat
psql "postgresql://langmem_app:CHANGEME@localhost:5432/langmem_db" -c "CREATE EXTENSION IF NOT EXISTS v
```

7) App connection string (use this in Python code)

```
-----------------------------------------------
postgresql://langmem_app:CHANGEME@localhost:5432/langmem_db
```

8) What your app will do next

```
----------------------------
```

- Point your Postgres-backed Store at that connection string.
- Call the store's setup() once (creates the tables/indexes it needs).
- Keep your manage_memory_tool / search_memory_tool calls the same.

Oracle ↔ Postgres mental mapping

```
-------------------------------
```

Oracle concept | Postgres equivalent

```
-----------------------------|---------------------
```

Create user with password | CREATE ROLE ... WITH LOGIN PASSWORD ...

Create a dedicated schema | CREATE SCHEMA memory AUTHORIZATION langmem_app

Make user own the database | CREATE DATABASE ... OWNER langmem_app

Enable special feature | CREATE EXTENSION vector;

Default schema for a session | ALTER ROLE/DB SET search_path TO memory, public