← **SQLite vs Postgres at my level of scale**

Output Capture

Prompt Used

Select an item...   + ⌄

Output

TT **B** *I* ~~T~~ ☰ ☷ 99 <> ⌗ ▦ ▣   Edit   Preview

SQLite, MySQL, and PostgreSQL each have their strengths and are suited for different use cases. The choice among them depends on your specific needs, particularly regarding scale, concurrency, and complexity.

SQLite

Pros:

*   **Lightweight:** SQLite is an embedded database, meaning it's a single file and requires minimal setup. It's ideal for small to medium-sized applications or those with low to moderate read/write activity.

*   **Portability:** Since it's just a file, it's easy to move or distribute.

*   **Simplicity:** No need for a server, which simplifies deployment and management.

Cons:

*   **Concurrency:** SQLite can handle multiple concurrent readers, but it has limitations with

Interface: save a prompt and associate it with an existing prompt – or create a new one

Content

 GPT Tags 🔖

☐ **Tag Name** ≡

☐ Custom GPT Data Management

☐ Data Analysis

☐ Data Architecture

☐ Data Architecture

☐ Database Architecture

☐ Database Design

☐ Database Optimization

☐ Databases

☐ Database Vs Spreadsheet

☐ Data Discovery

☐ Data Encryption

☐ Data Export

☐ Data (general)

Save a prompt to the prompt library and assign basic metadata items (rating, Boolean to show/hide in prompt library)

Prompt Library
Creating Item in Prompt Library

• **Prompt Text**

This was the prompt that I used to generated this output. Now, I'm going to capture it into my prompt library and also relate it to this output!
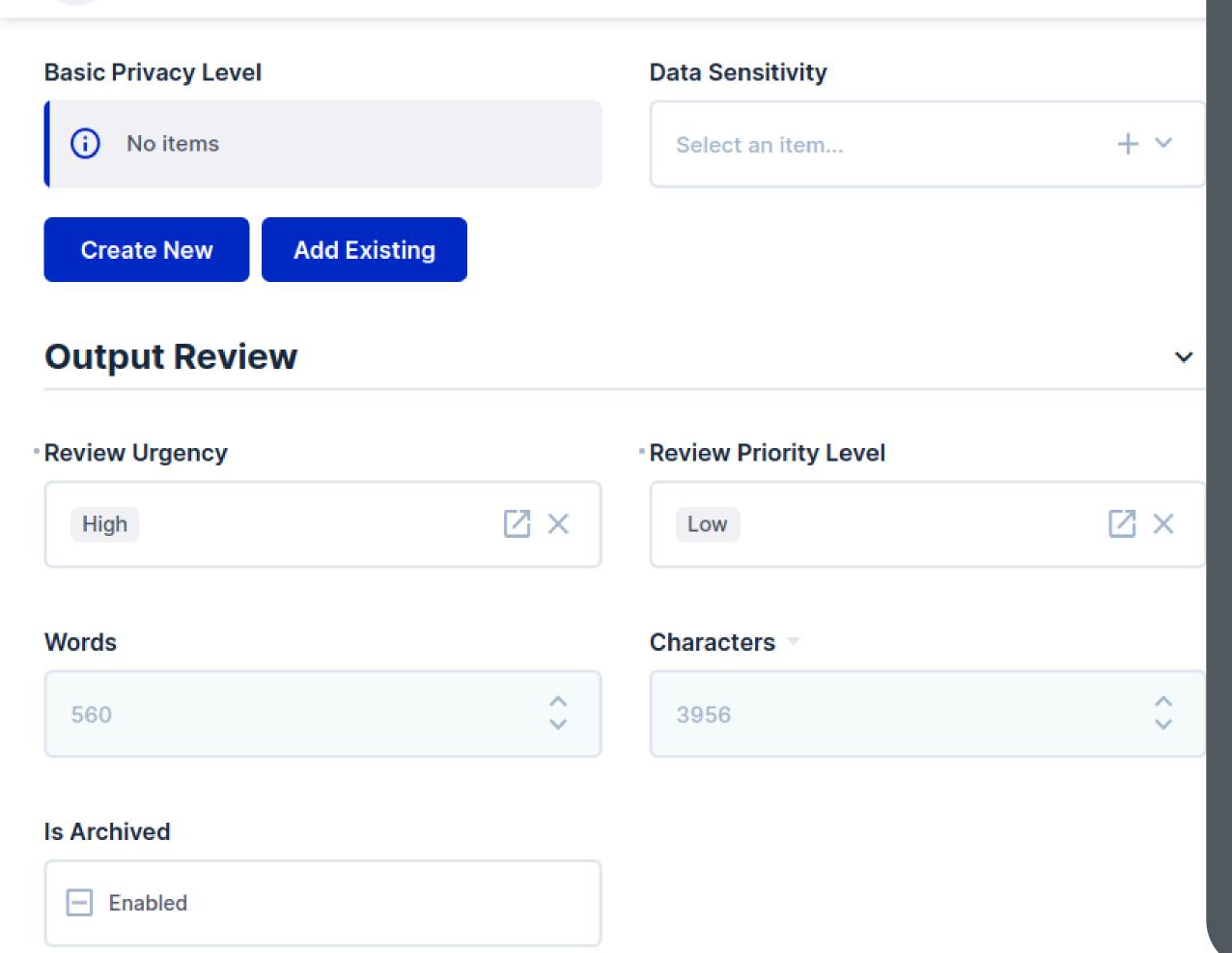
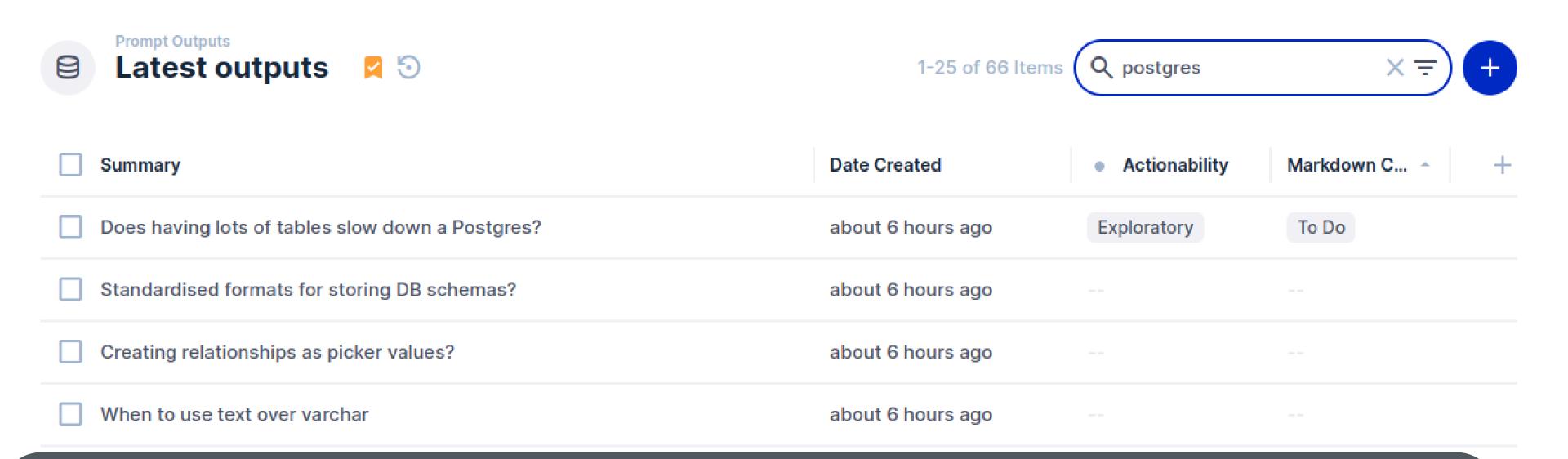**Engineered**

☐ Enabled

Rating

Show In Prompt Library?

☐ Enabled

SQLite vs Postgres at my level of scale

Basic Privacy Level

ⓘ No items

Create New **Add Existing**

Output Review ⌄

• **Review Urgency**

| High | ⬀ ✕ |

• **Review Priority Level**

| Low | ⬀ ✕ |

Words

| 560 | ⌄ |

Characters ▾

| 3956 | ⌄ |

Is Archived

⊟ Enabled

Data Sensitivity

Select an item... + ⌄

Core output management interface marking up metadata for compliance and internal review with autogenerated word & char counts

Latest outputs

1-25 of 66 Items

postgres

	Summary	Date Created	● Actionability	Markdown C... ▲	+
☐	Does having lots of tables slow down a Postgres?	about 6 hours ago	Exploratory	To Do	
☐	Standardised formats for storing DB schemas?	about 6 hours ago	--	--	
☐	Creating relationships as picker values?	about 6 hours ago	--	--	
☐	When to use text over varchar	about 6 hours ago	--	--	

Review latest saved outputs, search through them by keywords, and filter them by undocking filters. Parameters shown here are actionability lookup and markdown conversion status lookup

The "agent groups" metadata table is designed to allow administrators to group custom GPT agents (or fine-tuned LLM models). This could in turn be used for access control, allowing specific business functions to access only the agents needed for their operations

Agent Groups
Select Item

	Group Name	Description
☐	Customer Support	Agents specializ...
☐	Content Creation	Agents focused ...
☐	Data Analysis	Agents dedicate...
☐	Education and Training	Agents designed...
☐	Marketing and Advertising	Agents specializ...
☐	Technical Support	Agents equipped...
☐	Research and Development	Agents focused ...
☐	Legal and Compliance	Agents trained t...
☐	Translation and Localization	Agents focused ...
☐	Internal Operations	Agents designed...

Batch Editing 5 Items ✓

☐ GPT Used

Select an item... + ⌄

☐ Date Created

Key Details

☐ Summary

☑ Tags

Postgres 🗑

Postgre SQL 🗑

Create New **Add Existing**

☐ Associated Projects

ⓘ No items

Create New Add Existing

**Applying batch edits to saved outputs.
One could batch-assign outputs to a specific GPT/LLM/agent, associate them with tags, associate them with projects, or associate them with a calendar date**

← **Backoffice Development Copilot**

GPT Name

Backoffice Development Copilot

GPT Library

Daniel (personal)   ✕

Create New   **Add Existing**

Summary

A GPT which guides users through the process of developing effective backoffice and administrative syst

GPT Link

https://chatgpt.com/g/g-gdDolOeNm-backoffice-development-copilot

Creation Date

📅

The custom GPT module (which could be renamed 'custom agent') allows the user to perform simple (digital) inventory management on a "fleet" of custom GPTs or fine-tuned LLM agents maintained for business purposes. This module is associated with the prompt library and the outputs repository through M2M relationships.

Custom GPTs 🔖

8 Items 🔍 database ✕ ⇝ ⊕

☐ GPT Name ≡	GPT Link	GPT Optimised?	Importance	Access UI	⊕
☐ Databases > Spreadsheets	↗ https://chatgpt.com/g/g-bf	--	4	Web UI	
☐ Data Organisation Genie	↗ https://chatgpt.com/g/g-2A	--	4	Web UI	
☐ Data Relationships Utility	↗ https://chatgpt.com/g/g-Ql	--	0	Web UI	
☐ GPT Workbench Development Assistance	↗ https://chatgpt.com/g/g-VI	--	0	Web UI	
☐ Graph Database Stack Assistant	↗ https://chatgpt.com/g/g-M	--	0	Web UI	
☐ PostgreSQL Expert					
☐ Schema Genie					
☐ The Database Matchmaker	↗ https://chatgpt.com/g/g-uN	--	0	Web UI	

List view, custom GPTs module

← **Backoffice Development Copilot**

Configuration Text

A GPT whose purpose is to guide both technical and business users on the creation and development of effective backoffice solutions for administering businesses as well as enhancing their job function and managing their personal life.

The GPT should guide the user as they think about what features to develop in the backoffice system. This might include the selection of effective development frameworks, the brainstorming of

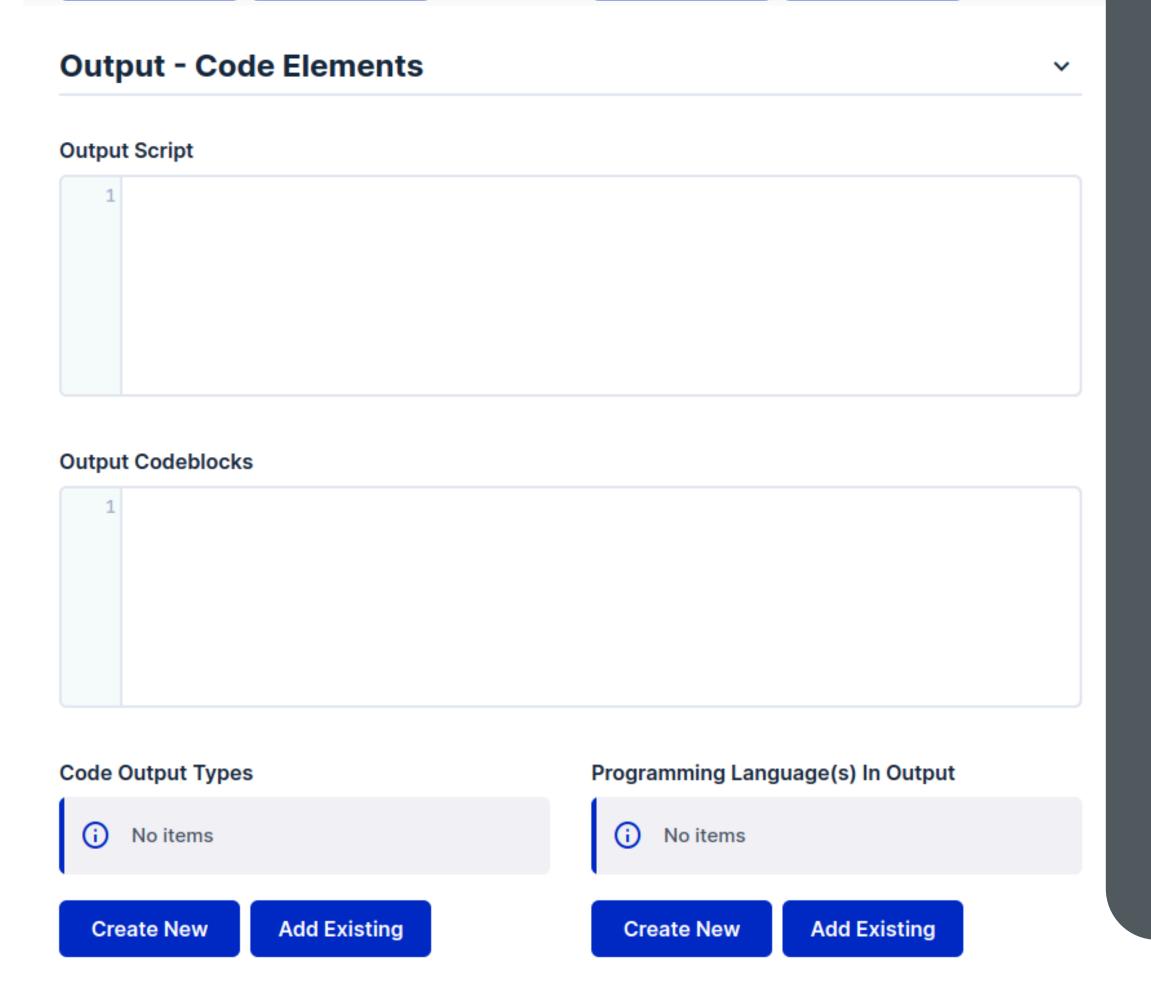**GPT Privacy Setting**

ⓘ   No items

Github Release Plan

Select an item...   + ⌄

Create New   **Add Existing**

Output - Code Elements ⌄

Output Script

```
1
```

Output Codeblocks

```
1
```

Code Output Types

ⓘ No items

Programming Language(s) In Output

ⓘ No items

Create New   **Add Existing**

Create New   **Add Existing**

Fields for recording generated code including scripts and snippets. M2M tables for code output types and programming language specification

Data structure: 3 core modules + lookup tables. A deeply interconnected design