

# I'm Afraid of the Database!

Jay Miller

November 17, 2025



**aiven**

# Jay Miller

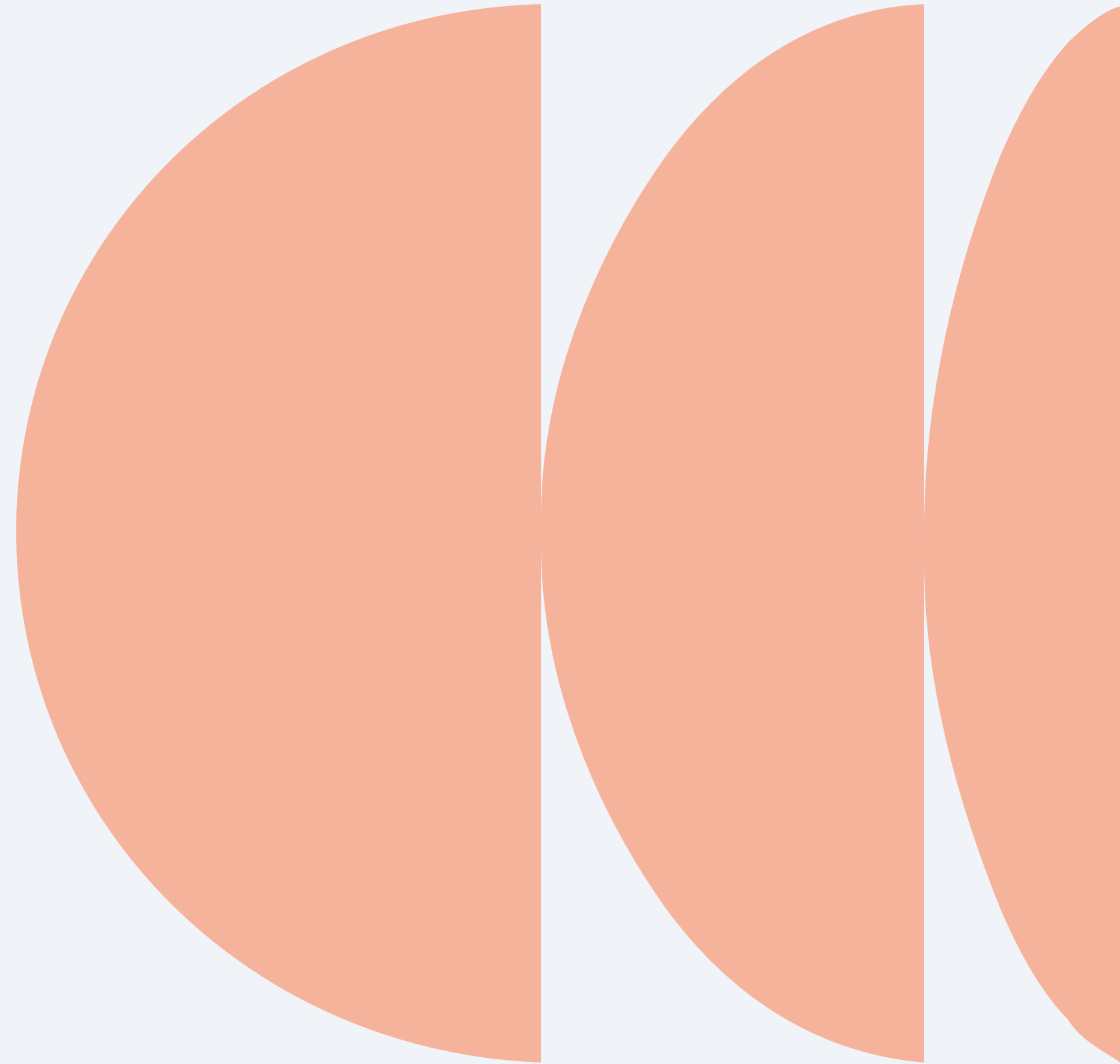
Staff Product Advocate @ Aiven  
Founder/Executor Black Python Devs

10+ years in Python

~ 1 Thinking about PostgreSQL



# Special Thanks





What makes  
databases scary for  
app developers?

# Why are databases scary

## Data Integrity & Security

It feels you always one configuration from leaking data.



## Query Language

Thinking about PostgreSQL the same way you think about Python is so hard they made ORMs for it.

## Cost

Why is it more expensive to store my data than run my app?



# Introducing Developer Tier for Aiven for PostgreSQL<sup>®</sup> services



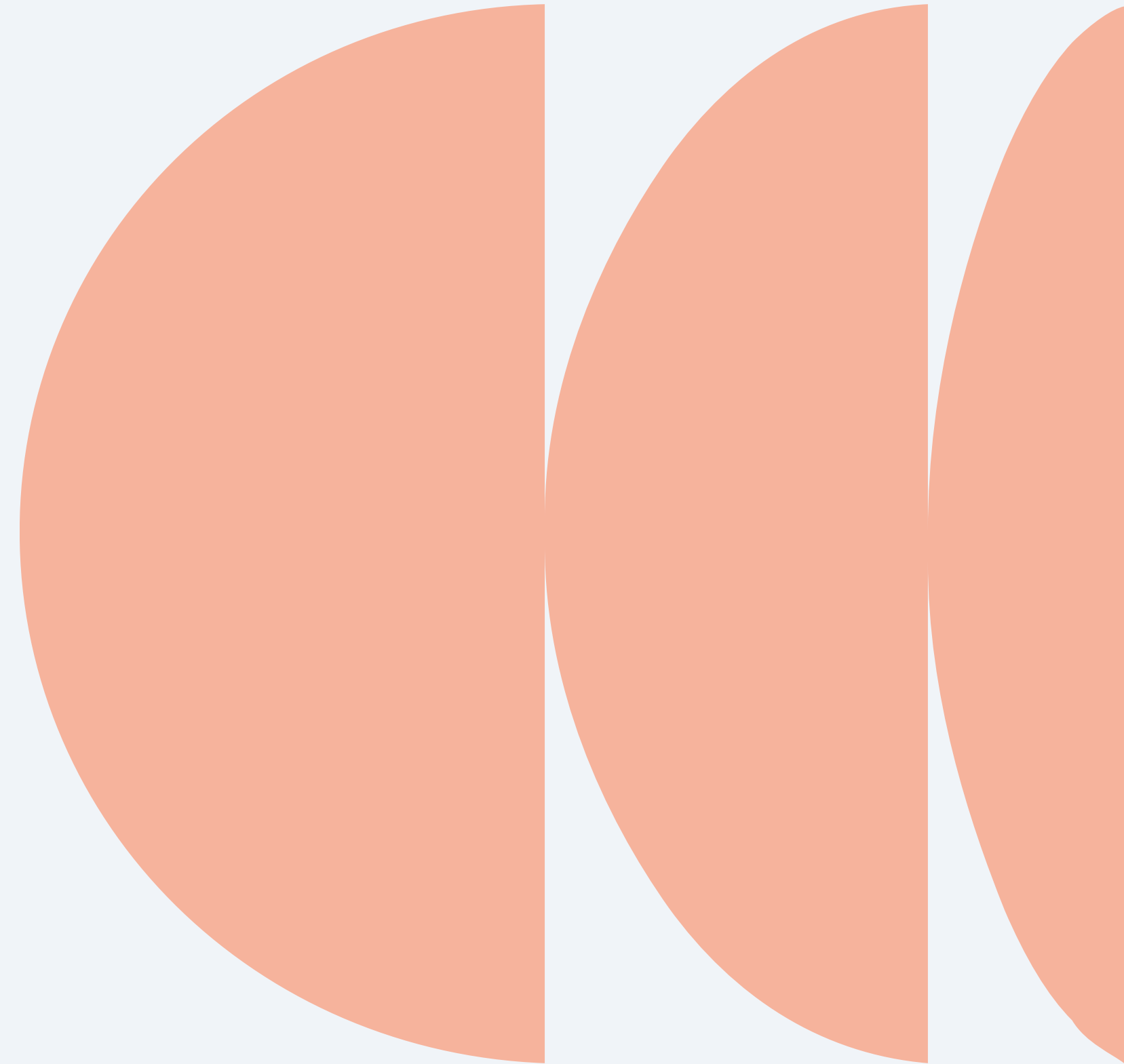
Get a \$300 credit for 30  
days!  
[aiven.io/jay](https://aiven.io/jay)

Service tier

<p><b>Free</b> <input type="radio"/></p> <p>Explore and learn the platform at no cost.</p> <ul style="list-style-type: none"><li>✓ Free forever</li><li>✗ Automatically powered off when inactive</li><li>✗ No integrations or connection pooling</li></ul> <p><b>\$0</b></p>	<p><b>Developer</b> <input checked="" type="radio"/></p> <p>A cost-effective option for test and personal projects.</p> <ul style="list-style-type: none"><li>✓ Inactive services aren't powered off</li><li>✓ Basic support tier</li><li>✗ No integrations or connection pooling</li></ul> <p><b>\$8</b> <b>Free 30-day trial</b></p>	<p><b>Professional</b> <input type="radio"/></p> <p>For highly available business-critical workloads.</p> <ul style="list-style-type: none"><li>✓ Free 30-day trial with \$300 of credits</li><li>✓ Deploy across multiple clouds and regions</li><li>✓ 99.99% uptime SLA</li><li>✓ Automatic backups for disaster recovery</li></ul> <p><b>From \$12</b> <b>Free 30-day trial</b></p>
---	--	--



# Confronting Database Fears



# ORMS

**django**

**SQLAlchemy**

# ORMS

#36277 closed New feature (fixed)

## Add GeneratedField(db\_persist=False) support on PostgreSQL 18+

Reported by:	Adam Johnson	Owned by:
Component:	Database layer (models, ORM)	Version:
Severity:	Normal	Keywords:
Cc:	David Sanders, JaeHyuckSa, Ahmed Ibrahim	Triage Stage:
Has patch:	yes	Needs documentation:
Needs tests:	no	Patch needs improvement:
Easy pickings:	no	UI/UX:
Pull Requests:	19907 merged	

### Description

PostgreSQL 18, expected late 2025, [includes virtual generated column support](#).

When PostgreSQL 18 is testable, we can update the associated database feature to a cached property that checks the server version [d5c19f9b327fcc1f707e10de648fe9d4cdacb9b/django/db/backends/postgresql/features.py#L76](https://github.com/django/django/pull/19907)

comment:2 by Sarah Boyce, 8 months ago

**Triage Stage:** Accepted → Someday/Maybe

Going to update to "Someday/Maybe" until PostgreSQL 18 is available.

comment:6 by Adam Johnson, 7 weeks ago

**Triage Stage:** Someday/Maybe → Accepted

It came out yesterday: <https://www.postgresql.org/about/news/postgresql-18-released-3142/>

I think we can start working on this now.

comment:7 by arsalan64, 7 weeks ago

**Owner:** set to arsalan64

**Status:** new → assigned

Self-assigning so I can start working on this

comment:8 by arsalan64, 7 weeks ago

**Has patch:** set

PR : <https://github.com/django/django/pull/19907>

In your opinion should we add or edit test cases for this change ? (Personally I think the tests already present are enough)

Last edited 7 weeks ago by arsalan64 (previous) (diff)

```
=====
web-1      | SEARCH SQL - Full-Text Search (FTS):
web-1      | =====
=====
web-1      | SELECT "transcripts_transcript"."id", "t
ranscripts_transcript"."youtube_id", "transcripts_tra
nscript"."srt_content", "transcripts_transcript"."tex
t_content", "transcripts_transcript"."created_at", "t
ranscripts_transcript"."updated_at", "transcripts_tra
nscript"."search_vector", ts_rank(to_tsvector(COALESC
E(("transcripts_transcript"."search_vector")::text, )
), websearch_to_tsquery(english::regconfig, Django))
AS "rank" FROM "transcripts_transcript" WHERE "transc
ripts_transcript"."search_vector" @@ (websearch_to_ts
query(english::regconfig, Django)) ORDER BY 8 DESC LI
MIT 50
web-1      | =====
```

### Free SQL to Text AI Converter

Use Aiven's SQL-to-Text converter to turn complex SQL queries into simple English. [Are you interested in an API version?](#)

Database type: PostgreSQL (Basic support) | Version: 18

```
Provide the SQL query to be explained:

1 SELECT "transcripts_transcript"."id", "transcripts_transcript"."youtube_id", "transcripts_transcript"."srt_content", "transcripts_transcript"."text_content", "transcripts_transcript"."created_at", "transcripts_transcript"."updated_at", "transcripts_transcript"."search_vector", ts_rank(to_tsvector(COALESCE(("transcripts_transcript"."search_vector")::text, ), websearch_to_tsquery(english::regconfig, Django)) AS "rank" FROM "transcripts_transcript" WHERE "transcripts_transcript"."search_vector" @@ (websearch_to_tsquery(english::regconfig, Django)) ORDER BY 8 DESC LIMIT 50
```

**Tip: Aiven can also help optimize your query**  
Receive indexing and SQL rewrites recommendations to speed up your queries. [Optimize now](#)

Explain Query

Query explained:  
Are you satisfied with the tool?  Yes, I'm satisfied  No, I'm not satisfied

The SQL query is structured as follows:

- SELECT Clause:** This specifies the columns to be retrieved.
  - "transcripts\_transcript"."id",
  - "transcripts\_transcript"."youtube\_id",
  - "transcripts\_transcript"."srt\_content",
  - "transcripts\_transcript"."text\_content",
  - "transcripts\_transcript"."created\_at",
  - "transcripts\_transcript"."updated\_at",
  - "transcripts\_transcript"."search\_vector" : These select the specified



### Logging raw SQL to the console in Django

...). Of course, turning on logging at the database is the most useful and what you ultimately want in...

[neilwithdata.com /](https://neilwithdata.com/)

[aiven.io/tools/sql-to-text](https://aiven.io/tools/sql-to-text)

# Migration Files – How they work

```
# Generated by Django 5.2.8 on 2025-11-12 21:28

from django.db import migrations

class Migration(migrations.Migration):

    dependencies = [
        ('transcripts', '0012_alter_srtsegment_embedding'),
    ]

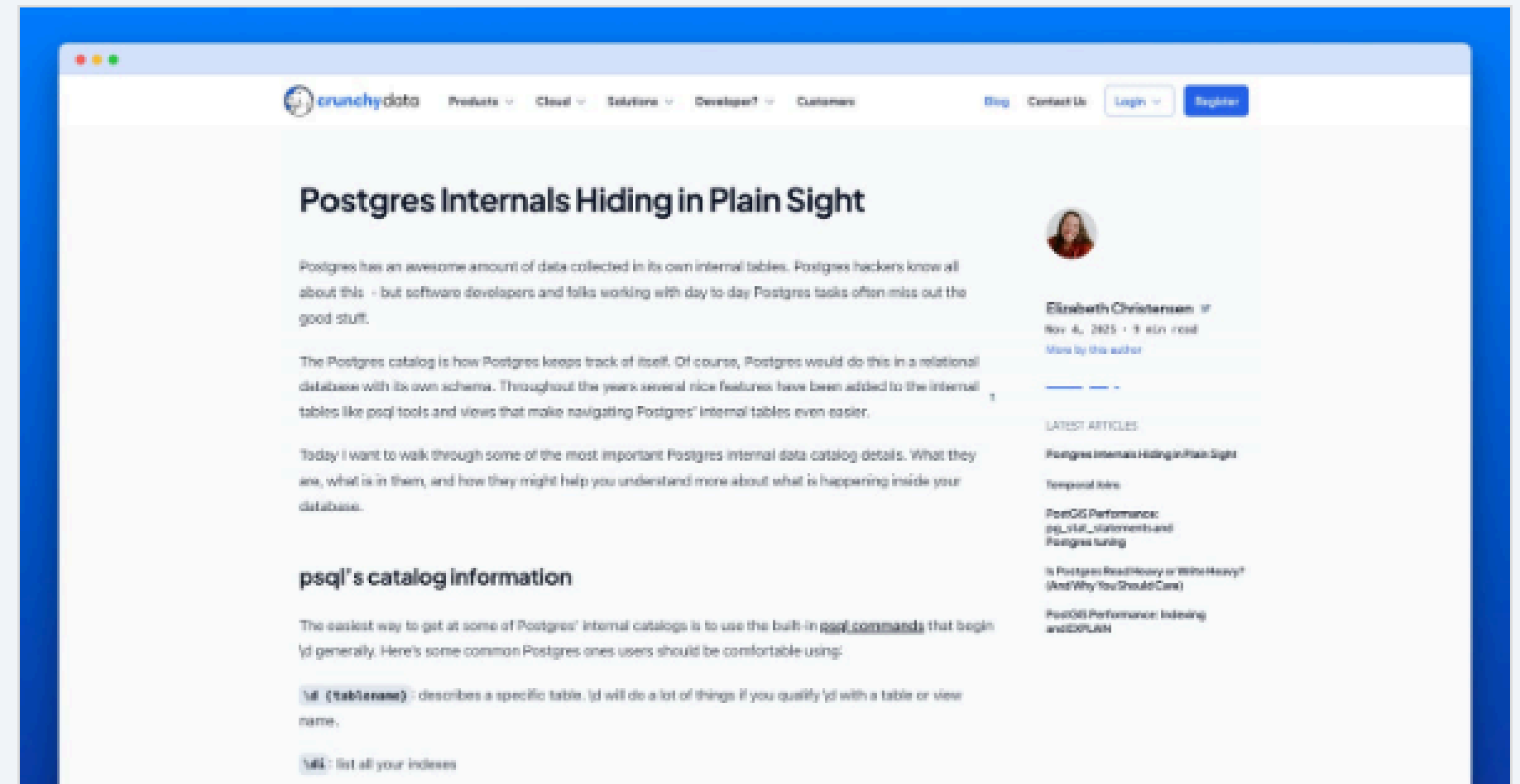
    operations = [
        migrations.RunSQL(
            sql="CREATE INDEX IF NOT EXISTS srtsegment_text_trgm_idx ON transcripts_srtsegment USING
gist (text gist_trgm_ops)",
            reverse_sql="DROP INDEX IF EXISTS srtsegment_text_trgm_idx CASCADE",
        ),
    ]
```

# Embracing PostgreSQL



# PostgreSQL Commands app devs should consider learning

```
\dt+  
\du+ – users (detailed info)  
\dv – views  
\df – functions  
pg_stat_statements
```



## Postgres Internals Hiding in Plain Sight | Crunchy Data Blog

Get under the hood of Postgres by looking at psql, system views, and internal tables.

 Crunchy Data

# PostgreSQL Conferences need App Developer's Stories

## Postgres CFP Deadlines

CFP = CALL FOR PROPOSALS

CFP CLOSING ON

	<b>Prague PostgreSQL Dev Day (P2D2)</b> 27-28 Jan 2026   Prague, Czech Republic	<b>Nov 14</b>
	<b>PgDay Dallas</b> 19 Feb 2026   Dallas, TX	<b>Nov 30</b>
	<b>PostgreSQL@SCaLE 23x</b> 05-08 Mar 2026   Pasadena, CA	<b>Dec 01</b>
	<b>CERN PGDay</b> 06 Feb 2026   Meyrin, Switzerland	<b>Dec 07</b>
	<b>FOSSASIA PGDay</b> 10 Mar 2026   Bangkok, Thailand	<b>Dec 15</b>
	<b>PGConf Germany</b> 21-22 Apr 2026   Essen, Germany	<b>Dec 19</b>
	<b>pgDay Paris</b> 26 Mar 2026   Paris, France	<b>Dec 19</b>

## Postgres CFP Deadlines

CFP = CALL FOR PROPOSALS

CFP CLOSING ON

	<b>PG DATA (formerly PGDay Chicago)</b> 4-5 Jun 2026   Chicago, IL	<b>Jan 05</b>
	<b>Nordic PGDay</b> 24 Mar 2026   Helsinki, Finland	<b>Jan 06</b>
	<b>PGConf.dev</b> 19-22 May 2026   Vancouver, Canada	<b>Jan 16</b>
	<b>POSETTE: An Event for Postgres</b> 16-18 Jun 2026   Virtual 🌐	<b>Feb 01</b>
	<b>PgDay BOSTON (CFP will open on Dec 1st)</b> 09 Jun 2026   Boston, MA	<b>Feb 28</b>

Source: Claire Giordano – LinkedIn



[2026.pg-data.org](http://2026.pg-data.org)



June 4-5 2026

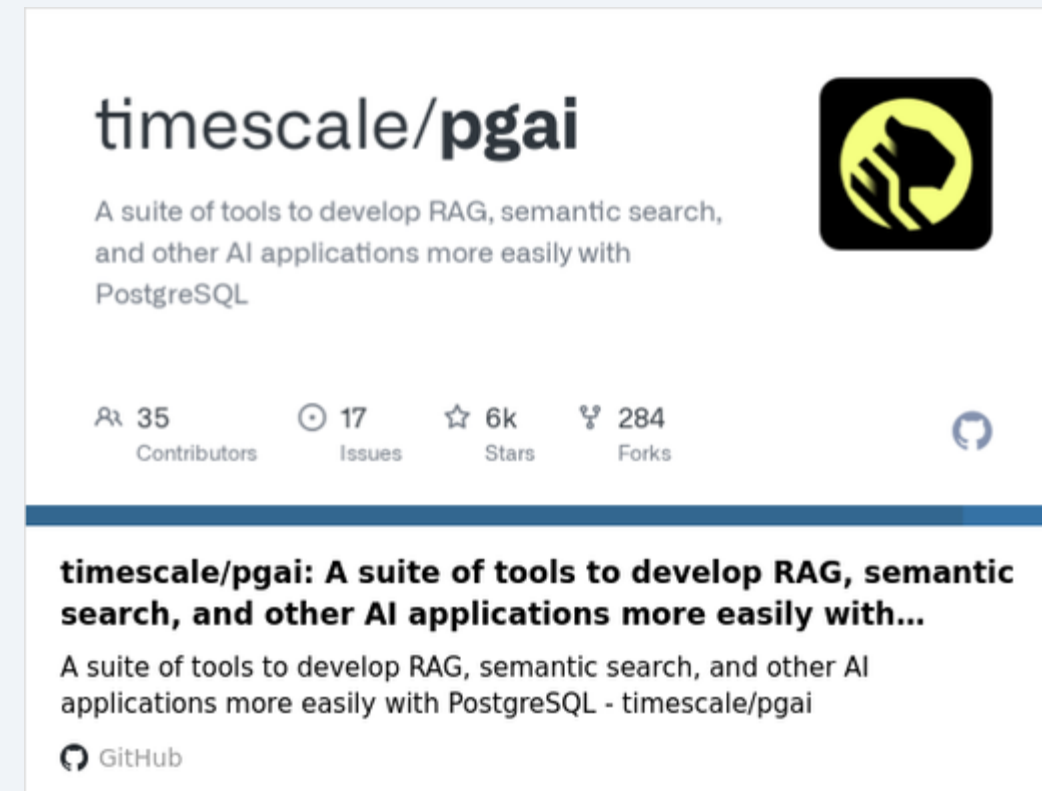
You should submit a talk!!

# AI (because it's 2025)

Don't 🙌 LEAK 🙌 PII 🙌 OR 🙌  
PASSWORDS 🙌 TO 🙌 AI

MCP works well but honestly I  
don't think it's necessary

`/clear` frequent and often



The screenshot shows the GitHub repository page for `timescale/pgai`. The repository name is displayed in a large font, followed by a brief description: "A suite of tools to develop RAG, semantic search, and other AI applications more easily with PostgreSQL". To the right of the text is a circular logo with a yellow background and a black silhouette of a hand holding a globe. Below the description, there are statistics for the repository: 35 Contributors, 17 Issues, 6k Stars, and 284 Forks. A blue horizontal bar separates the repository header from the main content area. The main content area features a bold title: "timescale/pgai: A suite of tools to develop RAG, semantic search, and other AI applications more easily with..." followed by a truncated description: "A suite of tools to develop RAG, semantic search, and other AI applications more easily with PostgreSQL - timescale/pgai". A GitHub logo is visible at the bottom left of the main content area.

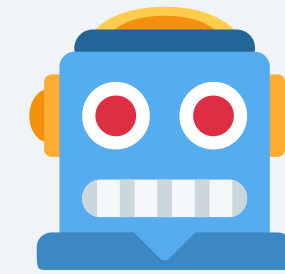


What can we do to  
make PostgreSQL  
less scary for App  
Developers

# Build more Apps with PostgreSQL at the front



+



# Did you know that PostgreSQL



**django**

I'm looking for Python  
Developers that would like  
to help work on PG Sites





## Watch video on YouTube

Error 153

Video player configuration error



# Questions

# Thank You!



Get a \$300 credit for 30  
days!  
[aiven.io/jay](https://aiven.io/jay)

