

October 2022

39 posts: [8 entries](#), [28 links](#), [3 quotes](#)

Oct. 1, 2022

[Supporting logical properties](#). A frustrating reminder from Jeremy Keith that Safari is not an evergreen browser: older iOS devices (1st gen iPad Air for example) get stuck on the last iOS version that supports them, which also sticks them with an old version of Safari, which means they will never get support for newer CSS properties such as inline-start and block-end. Jeremy shows how to use the `@supports` rule to hide this new syntax from those older browsers.

[1:03 am](#) / [css](#), [jeremy-keith](#), [safari](#), [web-standards](#), [ios](#)

Software engineering practices

Gergely Orosz [started a Twitter conversation](#) asking about recommended “software engineering practices” for development teams.

[... [1,557 words](#)]

[3:56 pm](#) / [documentation](#), [software-engineering](#), [version-control](#), [zero-downtime](#), [github-actions](#), [technical-debt](#), [gergely-orosz](#)

[Does Company ‘X’ have an Azure Active Directory Tenant?](#) ([via](#)) Neat write-up from Shawn Tabrizi about looking up if a company has Active Directory single-sign-on configured (which is based on OpenID) by checking for an OpenID configuration endpoint. I particularly enjoyed this new-to-me trick: Google’s “I’m Feeling Lucky” search button redirects to the first result, which means it can double as an unofficial API endpoint for returning the URL of the first matching search result.

[8:15 pm](#) / [google](#), [microsoft](#), [openid](#)

Oct. 2, 2022

[konstantint/SKompiler](#) ([via](#)) A tool for compiling trained SKLearn models into other representations—including SQL queries and Excel formulas. I’ve been pondering the most light-weight way to package a simple machine learning model as part of a larger application without needing to bundle heavy dependencies, this set of techniques looks ideal!

[11:56 pm](#) / [machine-learning](#), [sql](#)

Oct. 3, 2022

[Figure out how to serve an AWS Lambda function with a Function URL from a custom subdomain](#) ([via](#)) This took me five hours and 77 issue comments to figure out, but I finally managed to serve an AWS Lambda function running Datasette on a custom subdomain with an HTTPS certificate. I was going to write this up as a TIL but I’m exhausted so I decided to share my private notes thread instead.

12:29 am / [aws](#), [lambda](#), [datasette](#)

[Querying Postgres Tables Directly From DuckDB](#) ([via](#)) I learned a lot of interesting PostgreSQL tricks from this write-up of the new DuckDB feature that allows it to run queries against PostgreSQL servers directly. It works using COPY (SELECT ...) TO STDOUT (FORMAT binary) which writes rows to the protocol stream in efficient binary format, but splits the table being read into parallel fetches against page ranges and uses SET TRANSACTION SNAPSHOT ... in those parallel queries to ensure they see the same transactional snapshot of the database.

2:27 pm / [databases](#), [postgresql](#), [duckdb](#)

[You should have lots of AWS accounts](#) ([via](#)) Richard Crowley makes the case for maintaining multiple AWS accounts within a single company, because “AWS accounts are the most complete form of isolation on offer”.

6:36 pm / [aws](#), [richard-crowley](#), [security](#)

[Oct. 4, 2022](#)

[mod_wasm: run WebAssembly with Apache](#) ([via](#)) Brand new Apache module from a team at VMWare: mod_wasm builds on top of wasmtime to let you write WebAssembly programs that are exposed to the world by Apache, using a mechanism that looks similar to old CGI scripts (headers passed in environment variables, request body sent to standard input). Includes a demo Docker image that runs using Python-compiled-to-WebAssembly.

12:53 am / [apache](#), [webassembly](#)

[libsql](#) ([via](#)) A brand new Apache 2 licensed fork of SQLite. The README explains the rationale behind the project: SQLite itself is open source but not open contribution, and this fork aims to try out new ideas. The most interesting to me so far is a plan to support user defined functions implemented in WebAssembly. The project also intends to use Rust for new feature development.

4:13 pm / [open-source](#), [sqlite](#), [rust](#), [webassembly](#)

All these generative models point to the same big thing that’s about to alter culture; everyone’s going to be able to generate their own custom and subjective aesthetic realities across text, video, music (and all three) in increasingly delightful, coherent, and lengthy ways. This form of fractal reality is a double-edged sword – everyone gets to create and live in their own fantasies that can be made arbitrarily specific, and that also means everyone loses a further grip on any sense of a shared reality. Society is moving from having a centralized sense of itself to instead highly individualized choose-your-own adventure islands, all facilitated by AI. The implications of this are vast and unknowable. Get ready.

— [Jack Clark](#)

5:29 pm / [ai](#), [generative-ai](#), [jack-clark](#), [llms](#)

[Oct. 5, 2022](#)

[The Illustrated Stable Diffusion](#) ([via](#)) Jay Alammar provides a detailed, clearly explained description of how the Stable Diffusion image generation model actually works under the hood..

[# 2:58 am](#) / [ai](#), [stable-diffusion](#), [generative-ai](#), [text-to-image](#)

[Getting tabular data from unstructured text with GPT-3: an ongoing experiment](#) (via) Roberto Rocha shows how to use a carefully designed prompt (with plenty of examples) to get GPT-3 to convert unstructured textual data into a structured table.

[# 3:03 am](#) / [data-journalism](#), [ai](#), [gpt-3](#), [openai](#), [prompt-engineering](#), [generative-ai](#), [llms](#)

Is the AI spell-casting metaphor harmful or helpful?



For a few weeks now I've been promoting spell-casting as a metaphor for prompt design against generative AI systems such as GPT-3 and Stable Diffusion.

[... [990 words](#)]

[8:40 pm](#) / [ethics](#), [machine-learning](#), [ai](#), [gpt-3](#), [openai](#), [prompt-engineering](#), [prompt-injection](#), [generative-ai](#), [llms](#), [terry-pratchett](#), [ai-ethics](#)

Oct. 7, 2022

[py2rs](#). Extremely useful document providing resources for learning Rust followed by an extensive collection of common Python tasks (building a list, opening a file, spawning a thread, running a simple web server) and their Rust equivalents.

[# 5:44 pm](#) / [python](#), [rust](#)

Stringing together several free tiers to host an application with zero cost using fly.io, Litestream and Cloudflare.

Alexander Dahl provides a detailed description (and code) for his current preferred free hosting solution for small sites: SQLite (and a Go application) running on Fly's free tier, with the database replicated up to Cloudflare's R2 object storage (again on a free tier) by Litestream.

[# 5:47 pm](#) / [hosting](#), [sqlite](#), [cloudflare](#), [fly](#), [litestream](#)

[Can :has Connect 4?](#) ([via](#)) Spectacular CSS demo by Jhey Tompkins, implementing a working 3D Connect 4 game using just CSS (brilliant trickery with the new `:has()` selector) and not a single line of JavaScript.

[# 5:49 pm](#) / [css](#)

Oct. 8, 2022

[Reasons Why I Think 50% Coding 50% Marketing is the Best Framework for Solo Tech Founders](#) ([via](#)) Jon Yongfook offers a deliciously simple recipe for splitting up the work of both developing and marketing a product: one week of development, then one week of marketing, then repeat. I really like this concept: I mix the two activities randomly at the moment and constantly find myself feeling guilty that I'm not spending enough focused time on either of them!

[# 3:43 pm](#) / [entrepreneurship](#), [marketing](#), [startup](#)

Oct. 10, 2022

[The AI that creates any picture you want, explained.](#) Vox made this explainer video about text-to-image generative AI models back in June, months before Stable Diffusion was released and shortly before the DALL-E preview started rolling out to a wider audience. It's a really good video—in particular the animation that explains at a high level how diffusion models work, which starts about 5m30s in.

[# 3:28 am](#) / [ai](#), [dalle](#), [stable-diffusion](#), [generative-ai](#), [text-to-image](#)

Oct. 12, 2022

Weeknotes: Publishing data using Datasette Cloud

My initial preview releases of [Datasette Cloud](#) (the SaaS version of my [Datasette](#) open source project) have focused on private data collaboration.

[... [582 words](#)]

[8:08 pm](#) / [plugins](#), [projects](#), [datasette](#), [weeknotes](#), [datasette-cloud](#)

Oct. 13, 2022

[How to implement a “dry run mode” for data imports in Django](#) ([via](#)) Adam Johnson describes in detail a beautiful pattern for implementing a dry-run mode for a Django management command, by executing ORM calls inside an `atomic()` transaction block, showing a summary of changes that are made and then rolling the transaction back at the end.

[# 4:22 pm](#) / [django](#), [transactions](#), [adam-johnson](#)

Oct. 14, 2022

Automating screenshots for the Datasette documentation using shot-scraper

Suggested facets: [state_rank](#), [start](#) (date), [end](#) (date)

state >10 ✕

- [NY](#) 4,187
- [PA](#) 3,270
- [OH](#) 2,257
- [CA](#) 2,175
- [IL](#) 2,030
- [TX](#) 1,730
- [MA](#) 1,677
- [VA](#) 1,660
- [NC](#) 1,365
- [MI](#) 1,299

party >10 ✕

- [Democrat](#) 20,709
- [Republican](#) 19,129
- [Whig](#) 1,208
- [Jackson](#) 878
- [Federalist](#) 821
- [Adams](#) 260
- [Ind. Republican-Democrat](#) 129
- [American](#) 84
- [Unionist](#) 77
- [Anti Masonic](#) 76

type 2 ✕

- [rep](#) 40,633
- [sen](#) 3,909

Link	rowid ▼ ⚙	legislator_id ⚙	type ⚙	state ⚙	start ⚙	end ⚙	class ⚙	party ⚙	district ⚙	h
1	1	Richard Bassett B000226	sen	DE	1789-03-04	1793-03-03	2	Anti-Administration		
2	2	Theodorick Bland B000546	rep	VA	1789-03-04	1791-03-03			9	
3	3	Aedanus Burke B001086	rep	SC	1789-03-04	1791-03-03			2	

I released [shot-scraper](#) back [in March](#) as a tool for keeping screenshots in documentation up-to-date.

[... [1,810 words](#)]

11:44 pm / [documentation](#), [datasette](#), [github-actions](#), [shot-scraper](#)

Oct. 15, 2022

[shot-scraper 1.0](#) ([via](#)) Only a minor release in terms of features, but I decided that I'm comfortable enough with the CLI design at this point that I'm ready to stamp a 1.0 on it and commit to not making backwards-incompatible changes (at least without shipping a 2.0 release, which I'd like to avoid if possible).

9:28 pm / [cli](#), [projects](#), [shot-scraper](#)

[Dumping the HTML of a page using shot-scraper](#). New in 1.0 is the “shot-scraper html URL” command, which outputs the HTML of a page once JavaScript has finished executing there. You can pass in additional custom JavaScript to run before the shapshot is taken, and you can also specify a CSS selector on the page to return just that fragment of HTML.

9:30 pm / [shot-scraper](#)

[How to create a Python package in 2022](#) ([via](#)) Fantastic tutorial on modern Python packaging by Rodrigo Girão Serrão. I've been meaning to figure out Poetry for a while now and this gave me exactly the information I needed to start figuring it out. Great coverage of GitHub Actions, Tox and pre-commit as well.

10:10 pm / [packaging](#), [python](#), [github-actions](#)

Oct. 16, 2022

Half Moon Bay Pumpkin Festival traffic on Saturday 15th October 2022 ([via](#)) It's the Half Moon Bay Pumpkin Festival this weekend... and its impact on the traffic between our little town of El Granada and Half Moon Bay—8 minutes drive away—is notorious. So I built a git scraper that archives estimated driving times from the Google Maps Navigation API, and used git-history to turn that scraped data into a SQLite database and visualize it on a chart.

3:56 am / [projects](#), [git-scraping](#), [git-history](#), [half-moon-bay](#)

Oct. 17, 2022

“You are GPT-3”. Genius piece of prompt design by Riley Goodside. “A long-form GPT-3 prompt for assisted question-answering with accurate arithmetic, string operations, and Wikipedia lookup. Generated IPython commands (in green) are pasted into IPython and output is pasted back into the prompt (no green).” Uses “Out[” as a stop sequence to ensure GPT-3 stops at each generated iPython prompt rather than inventing the output itself.

4:35 am / [gpt-3](#), [prompt-engineering](#), [generative-ai](#), [riley-goodside](#), [llms](#)

Oct. 18, 2022

Chris Amico's Python setup for 2022 ([via](#)) Homebrew to install pyenv, then pyenv to install specific Python versions. pipx and pipenv for package management. I need to habitually start using pyenv for everything.

2:04 pm / [python](#), [chris-amico](#)

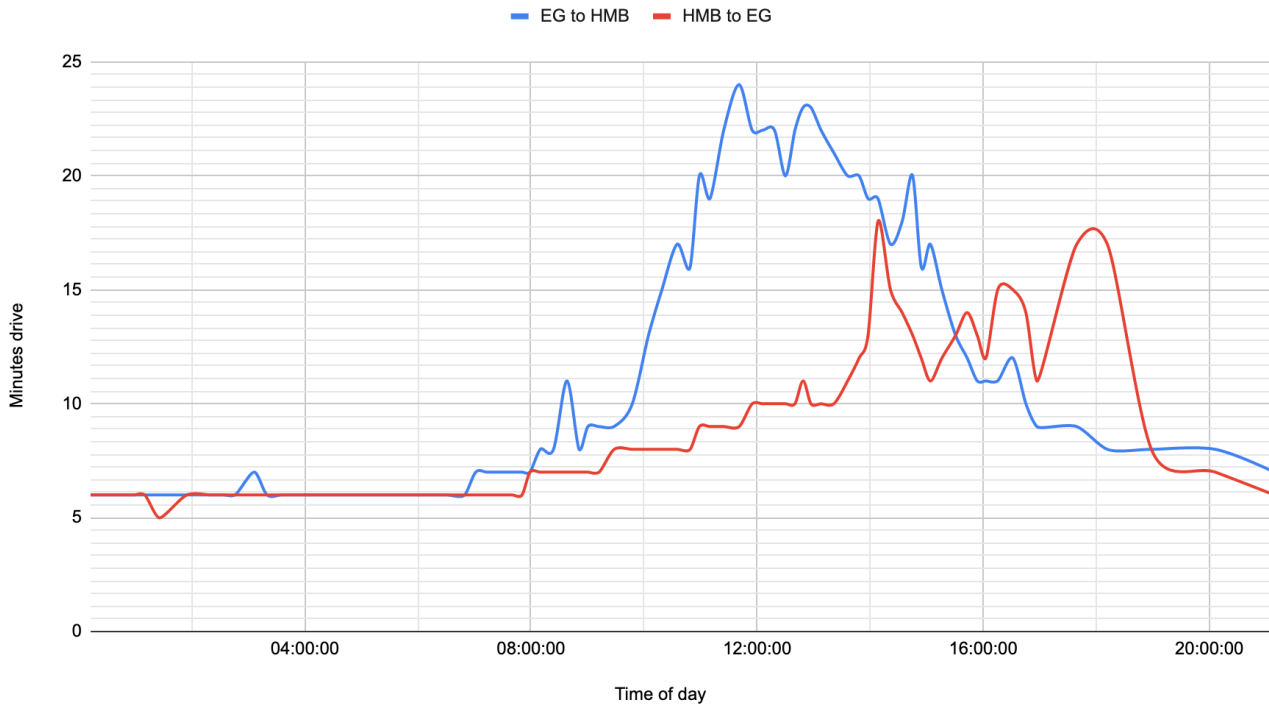
Oct. 19, 2022

Simple, Fast, and Scalable Reverse Image Search Using Perceptual Hashes and DynamoDB. Christopher Bong provides a clear explanation of how perceptual hashes can be used to create a string representing the visual content of an image, such that similar images can be identified by calculating a hamming distance between those hashes. He then explains how they built a large-scale system for this at Canva on top of DynamoDB, by splitting those strings into smaller hash windows and using those for efficient bulk lookups of similar candidates.

3:04 pm / [images](#), [search](#)

Measuring traffic during the Half Moon Bay Pumpkin Festival

Pumpkin Festival traffic on Sunday 16th October 2022



This weekend was the [50th annual Half Moon Bay Pumpkin Festival](#).

[... [2,693 words](#)]

[3:41 pm](#) / [data-journalism](#), [natalie-downe](#), [projects](#), [sqlite](#), [datasette](#), [git-scraping](#), [git-history](#), [datasette-lite](#), [half-moon-bay](#)

[Oct. 21, 2022](#)

[The Commodordion](#) ([via](#)) The Commodordion is “an 8-bit accordion primarily made of C64s, floppy disks, and gaffer tape” by Linus Åkesson. It’s absolutely beautiful.

[11:36 pm](#) / [commodore](#), [hacks](#), [music](#)

page 1 / 2 [next »](#)

[2022](#) » October

M	T	W	T	F	S	S
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

