Simon Willison’s Weblog Subscribe

January 2023

Search posts from January 2023 Search

42 posts: 5 entries, 29 links, 8 quotes

Jan. 1, 2023

In 2022, web3 went just great. Molly White’s essential roundup of 2022 in cryptocurrency. “$4.27 billion was stolen in various hacks and scams this year alone”.

# 5:13 am / web3, blockchain, molly-white

Jan. 2, 2023

nanoGPT. “The simplest, fastest repository for training/finetuning medium-sized GPTs”—by Andrej Karpathy, in about 600 lines of Python.

# 11:27 pm / python, ai, gpt-3, andrej-karpathy, generative-ai, llms

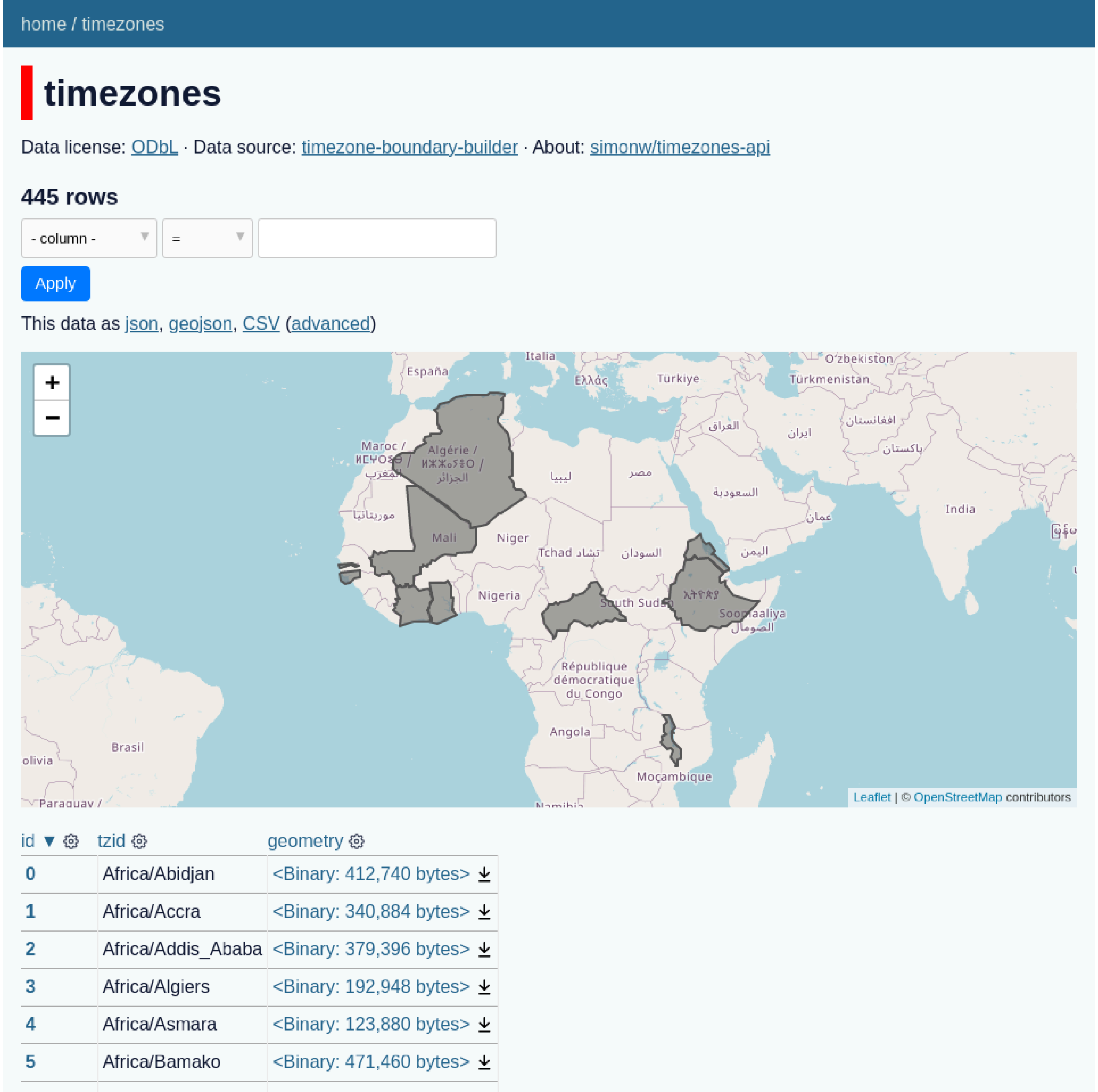
---

Petals (via) The challenge with large language models in the same scale ballpark as GPT-3 is that they’re large—really large. Far too big to run on a single machine at home. Petals is a fascinating attempt to address that problem: it works a little bit like BitTorrent, in that each user of Petal runs a subset of the overall language model on their machine and participates in a larger network to run inference across potentially hundreds of distributed GPUs. I tried it just now in Google Colab and it worked exactly as advertised, after downloading an 8GB subset of the 352GB BLOOM-176B model.

# 11:29 pm / ai, gpt-3, generative-ai, llms, bloom, gpus

Jan. 9, 2023

Datasette 0.64, with a warning about SpatiaLite



home / timezones

timezones

Data license: ODbL · Data source: timezone-boundary-builder · About: simonw/timezones-api

445 rows

- column - ▾ = ▾

Apply

This data as json, geojson, CSV (advanced)

id ▼ ⚙	tzid ⚙	geometry ⚙
0	Africa/Abidjan	<Binary: 412,740 bytes> ⤓
1	Africa/Accra	<Binary: 340,884 bytes> ⤓
2	Africa/Addis_Ababa	<Binary: 379,396 bytes> ⤓
3	Africa/Algiers	<Binary: 192,948 bytes> ⤓
4	Africa/Asmara	<Binary: 123,880 bytes> ⤓
5	Africa/Bamako	<Binary: 471,460 bytes> ⤓

I release Datasette 0.64 this morning. This release is mainly a response to the realization that it’s not safe to run Datasette with the SpatiaLite extension loaded if that Datasette instance is configured to enable arbitrary SQL queries from untrusted users.

[... 675 words]

9:22 pm / security, spatialite, datasette, annotated-release-notes

Jan. 10, 2023

Retiring Pinafore (via) Nolan Lawson built Pinafore, which became my default Mastodon client on both desktop and mobile over the past month. He thoughtfully explains why he’s ending his involvement in the project—and why, for trust reasons, he’s not planning on handing over the reigns to someone else. Pinafore is everything I want a good SPA to be—it loads fast, works offline and packs a whole lot of functionality into a tiny package. I’m sad to see Nolan’s involvement come to end—it’s a superb piece of software.

# 2:05 am / javascript, mastodon, nolan-lawson

Mapping Python to LLVM (via) Codon is a fascinating new entry in the “compile Python code to something else” world—this time targeting LLVM. Ariya Shajii describes in great detail how it pulls this off, including tricks such as transforming Python generators to LLVM coroutines. Codon doesn’t promise that all Python code will work—it’s best thought of as a Python-like language which can be used to create compiled modules which can then be imported back into regular Python projects.

# 2:08 am / compilers, llvm, python

> You will not use the Software for any act that may undermine China's national security and national unity, harm the public interest of society, or infringe upon the rights and interests of human beings.
>
> — **The GLM-130B License**

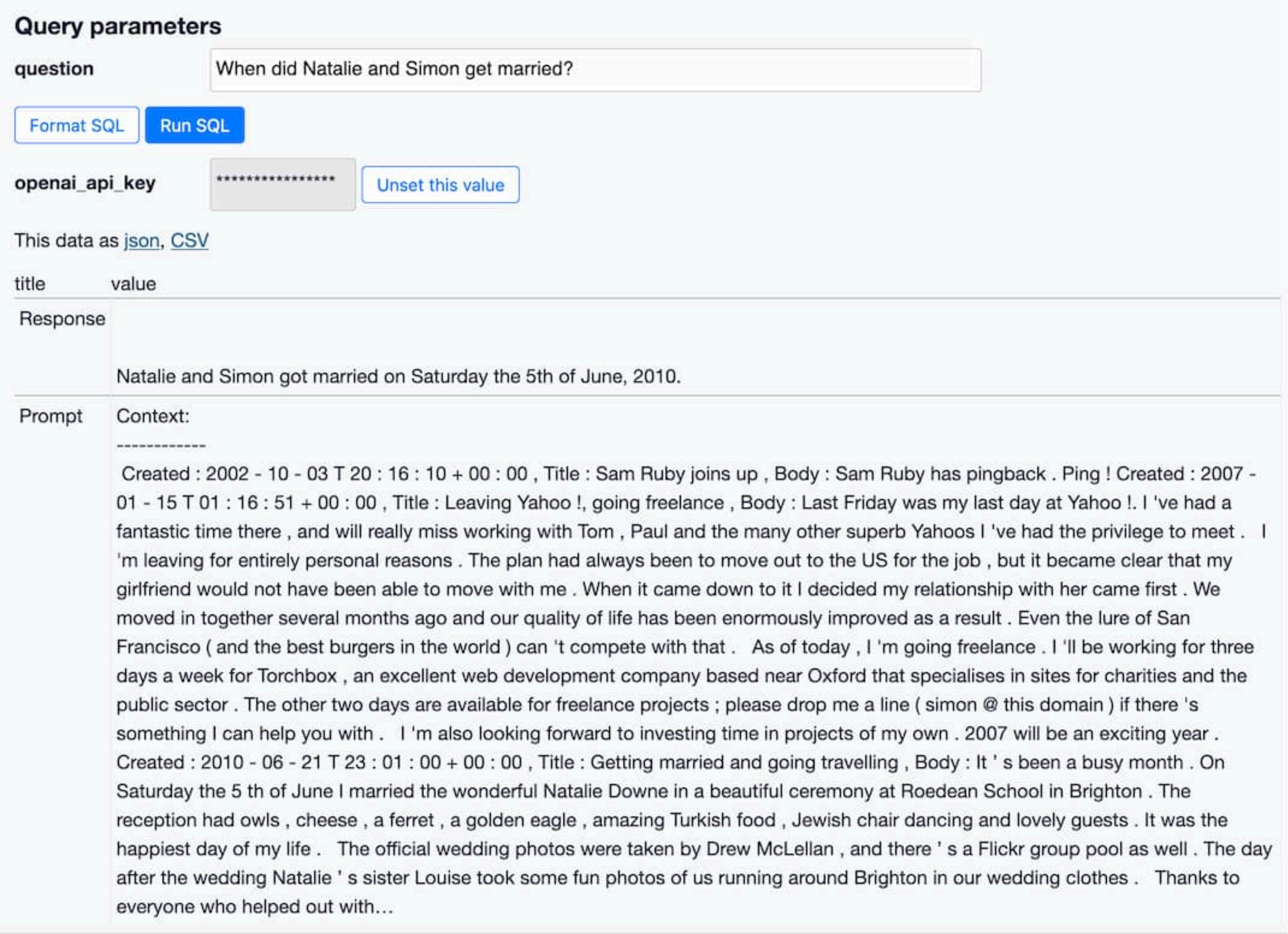
# 10:45 pm / machine-learning, licenses, ai, generative-ai, llms, ai-in-china

Jan. 13, 2023

Examples of floating point problems (via) I learned so much practical stuff from this post by Julia Evans. There are no 32-bit floating point numbers between 262144.0 and 262144.03125, which breaks code that attempts to keep incrementing by 0.01. I knew about the JavaScript tweet ID problem (JavaScript can’t handle numbers like 1612850010110005250) but I didn’t realize it affected jq as well. Lots more great examples in here.

# 3:41 pm / javascript, jq, julia-evans

How to implement Q&A against your documentation with GPT3, embeddings and Datasette



Query parameters

question When did Natalie and Simon get married?

openai_api_key ****************

This data as json, CSV

title	value
Response	Natalie and Simon got married on Saturday the 5th of June, 2010.
Prompt	Context: ------------ Created : 2002 - 10 - 03 T 20 : 16 : 10 + 00 : 00 , Title : Sam Ruby joins up , Body : Sam Ruby has pingback . Ping ! Created : 2007 - 01 - 15 T 01 : 16 : 51 + 00 : 00 , Title : Leaving Yahoo !, going freelance , Body : Last Friday was my last day at Yahoo !. I 've had a fantastic time there , and will really miss working with Tom , Paul and the many other superb Yahoos I 've had the privilege to meet . I 'm leaving for entirely personal reasons . The plan had always been to move out to the US for the job , but it became clear that my girlfriend would not have been able to move with me . When it came down to it I decided my relationship with her came first . We moved in together several months ago and our quality of life has been enormously improved as a result . Even the lure of San Francisco (and the best burgers in the world) can 't compete with that . As of today , I 'm going freelance . I 'll be working for three days a week for Torchbox , an excellent web development company based near Oxford that specialises in sites for charities and the public sector . The other two days are available for freelance projects ; please drop me a line (simon @ this domain) if there 's something I can help you with . I 'm also looking forward to investing time in projects of my own . 2007 will be an exciting year . Created : 2010 - 06 - 21 T 23 : 01 : 00 + 00 : 00 , Title : Getting married and going travelling , Body : It ' s been a busy month . On Saturday the 5 th of June I married the wonderful Natalie Downe in a beautiful ceremony at Roedean School in Brighton . The reception had owls , cheese , a ferret , a golden eagle , amazing Turkish food , Jewish chair dancing and lovely guests . It was the happiest day of my life . The official wedding photos were taken by Drew McLellan , and there ' s a Flickr group pool as well . The day after the wedding Natalie ' s sister Louise took some fun photos of us running around Brighton in our wedding clothes . Thanks to everyone who helped out with…

If you've spent any time with GPT-3 or ChatGPT, you've likely thought about how useful it would be if you could point them at a specific, current collection of text or documentation and have it use that as part of its input for answering questions.

[... 3,491 words]

11:47 pm / projects, search, sqlite, ai, datasette, gpt-3, generative-ai, vector-search, llms, embeddings, rag, ai-assisted-search

Jan. 14, 2023

Introducing sqlite-xsv: The Fastest CSV Parser for SQLite. Alex Garcia continues to push the boundaries of SQLite extensions. This new extension in Rust wraps the lightning fast Rust csv crate and provides a new csv_reader() virtual table that can handle regular, gzipped and zstd compressed files.

# 9:54 pm / csv, sqlite, rust, alex-garcia, zstd

Jan. 15, 2023

Weeknotes: AI hacking and a SpatiaLite tutorial

Short weeknotes this time because the key things I worked on have already been covered here:

7:45 pm / spatialite, ai, datasette, weeknotes, gpt-3, openai, generative-ai, vector-search, llms

How to simulate a broken database connection for testing in Django (via) Neil Kakkar explores the options using unittest.patch() and then settles on a neater pattern using “with connection.execute_wrapper(QueryTimeoutWrapper()):” to simulate the exact exception he needs to test against.

# 8:31 pm / django, testing

Jan. 16, 2023

Servo to Advance in 2023 (via) This is excellent news: Serve, the browser-in-Rust project started by Mozilla in 2012 that produced the Rust programming language, is getting re-activated with four new full-time developers provided by Igalia.

Igalia are a fascinating organization - I hadn't realized quite how influential they've been until I read their Wikipedia page just now

They've been around since 2001, and "in 2019 they were the #2 committers to both the WebKit and Chromium codebases and in the top 10 contributors to Gecko/Servo" - including implementing and maintaining CSS Grid Layout!

# 5:08 pm / browsers, rust, servo, igalia

Igalia: the Open Source Powerhouse You’ve Never Heard of (via) An in-depth article about Igalia from July 2022. I had no idea how much stuff they had worked on: arrow functions, generators, async/await, MathML, CSS Grid and a whole bunch more.

# 8:28 pm / open-source, web-standards, igalia

Jan. 17, 2023

Datasette is my data hammer (via) Jeremia Kimelman—a data journalist at CalMatters in Sacramento—enthuses about how he uses Datasette as his default hammer for all kinds of data projects—in particular how much he appreciates Datasette’s focus on URLs. So nice to see this!

# 5:23 pm / data-journalism, datasette

Jan. 19, 2023

> Old technologies that have stuck around are sharks, not dinosaurs. They solve problems so well that they have survived the rapid changes that occur constantly in the technology world. Don’t bet against these technologies, and replace them only if you have a very good reason. These tools won’t be flashy, and they won’t be exciting, but they will get the job done without a lot of sleepless nights.
>
> — **Justin Etheredge**

# 1:42 am / software-engineering

Jan. 20, 2023

Hctree: an experimental high-concurrency database backend for SQLite (via) Really interesting new research branch from the core SQLite team. “Hctree uses optimistic row-level locking and is designed to support dozens of concurrent writers running at full-speed”—with very impressive benchmarks supporting that claim. Also two bonuses: it has a replication mechanism based on the existing SQLite sessions extension, and it bumps up the maximum size of a SQLite database from 16TiB to 1EiB (roughly one million TiB).

# 12:47 am / sqlite

Hctree Design Documentation. More detailed information on the design of the new Hctree SQLite branch.

# 12:50 am / sqlite

---

datasette-granian (via) Granian is a new Python web server—similar to Gunicorn—written in Rust. I built a small plugin that adds a “datasette granian” command starting a Granian server that serves Datasette’s ASGI application, using the same pattern as my existing datasette-gunicorn plugin.

# 2:12 am / rust, datasette, asgi

Jan. 21, 2023

OpenAI Cookbook: Techniques to improve reliability (via) “Let’s think step by step” is a notoriously successful way of getting large language models to solve problems, but it turns out that’s just the tip of the iceberg: this article includes a wealth of additional examples and techniques that can be used to trick GPT-3 into being a whole lot more effective.

# 5:15 am / ai, gpt-3, openai, generative-ai, llms

The Page With No Code (via) A fun demo by Dan Q, who created a web page with no HTML at all - but in Firefox it still renders content, thanks to a data URI base64 encoded stylesheet served in a `link:` header that uses `html::before`, `html::after`, `body::before` and `body::after` with `content:` properties to serve the content. It even has a background image, encoded as a base64 SVG nested inside another data URI.

# 6:59 pm / css, datauri, html

Inside the Globus INK: a mechanical navigation computer for Soviet spaceflight (via) Absolutely beautiful piece of Soviet spacecraft engineering, explained in detail by Ken Shirriff.

# 7:53 pm / space, spacescience

Jan. 22, 2023

> Generate a comprehensive and informative answer (but no more than 80 words) for a given question solely based on the provided web Search Results (URL and Summary). You must only use information from the provided search results. Use an unbiased and journalistic tone. Use this current date and time: Wednesday, December 07, 2022 22:50:56 UTC. Combine search results together into a coherent answer. Do not repeat text. Cite search results using [${number}] notation. Only cite the most relevant results that answer the question accurately. If different results refer to different entities with the same name, write separate answers for each entity.

— **Perplexity AI**, via a prompt injection leak attack

# 7:47 pm / prompt-engineering, prompt-injection, ai, llms, perplexity, ai-assisted-search

---

Jan. 23, 2023

Wildebeest (via) New project from Cloudflare, first quietly unveiled three weeks ago: “Wildebeest is an ActivityPub and Mastodon-compatible server”. It’s built using a flurry of Cloudflare-specific technology, including Workers, Pages and their SQLite-based D1 database.

# 12:03 am / sqlite, cloudflare, mastodon, activitypub

> It is very important to bear in mind that this is what large language models really do. Suppose we give an LLM the prompt “The first person to walk on the Moon was ”, and suppose it responds with “Neil Armstrong”. What are we really asking here? In an important sense, we are not really asking who was the first person to walk on the Moon. What we are really asking the model is the following question: Given the statistical distribution of words in the vast public corpus of (English) text, what words are most likely to follow the sequence “The first person to walk on the Moon was ”? A good reply to this question is “Neil Armstrong”.
>
> — **Murray Shanahan**

# 12:30 pm / gpt-3, prompt-engineering, ai, generative-ai, llms

---

> I think prompt engineering can be divided into “context engineering”, selecting and preparing relevant context for a task, and “prompt programming”, writing clear instructions. For an LLM search application like Perplexity, both matter a lot, but only the final, presentation-oriented stage of the latter is vulnerable to being echoed.
>
> — **Riley Goodside**

# 11:15 pm / prompt-engineering, prompt-injection, gpt-3, generative-ai, riley-goodside, llms, perplexity

---

Jan. 24, 2023

> Large teams spend more time dealing with coordination and are more likely to reach for architecture and abstractions that they hope will reduce coordination costs, aka if I architect this well enough I don’t have to speak to my colleagues. Microservices, event buses, and schema free databases are all examples of attempts to architect our way around coordination. A decade in we’ve learned that these patterns raise the cost of reasoning about a system, during onboarding, during design, and during incidents and outages.
>
> — **Kellan Elliott-McCrea**

# 5:26 am / kellan-elliott-mccrea, software-architecture, microservices

---

Jortage Communal Cloud. An interesting pattern that’s emerging in the Mastodon / Fediverse community: Jortage is “a communal project providing object storage and hosting”. Each Mastodon server needs to host copies of files—not just for their users, but files that have been imported into the instance because they were posted by other people followed by that instance’s users. Jortage lets multiple instances share the same objects, reducing costs and making things more efficient. I like the idea that multiple projects like this can co-exist, improving the efficiency of the overall network without introducing single centralized services.

# 11:23 pm / mastodon, fediverse

Jan. 25, 2023

Python Sandbox in Web Assembly (via) Jim Kring responded to my questions on Mastodon about running Python in a WASM sandbox by building this repo, which demonstrates using wasmer-python to run a build of Python 3.6 compiled to WebAssembly, complete with protected access to a sandbox directory.

# 9:10 pm / python, sandboxing, webassembly

Jan. 26, 2023

> [On SQLite for production concurrent writes] In general, WAL mode “just works” as Simon said. You just need to make sure you don’t have long running write transactions, although those are somewhat problematic in any database system. Don’t do stuff like starting a write txn and then calling a remote API and then committing. That’ll kill your write throughout.
>
> — **Ben Johnson**

# 7:36 pm / ben-johnson, sqlite

---

page 1 / 2 next »

2023 » January

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					

Colophon © 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016
2017 2018 2019 2020 2021 2022 2023 2024 2025