

The of Programming

Matthias Endler





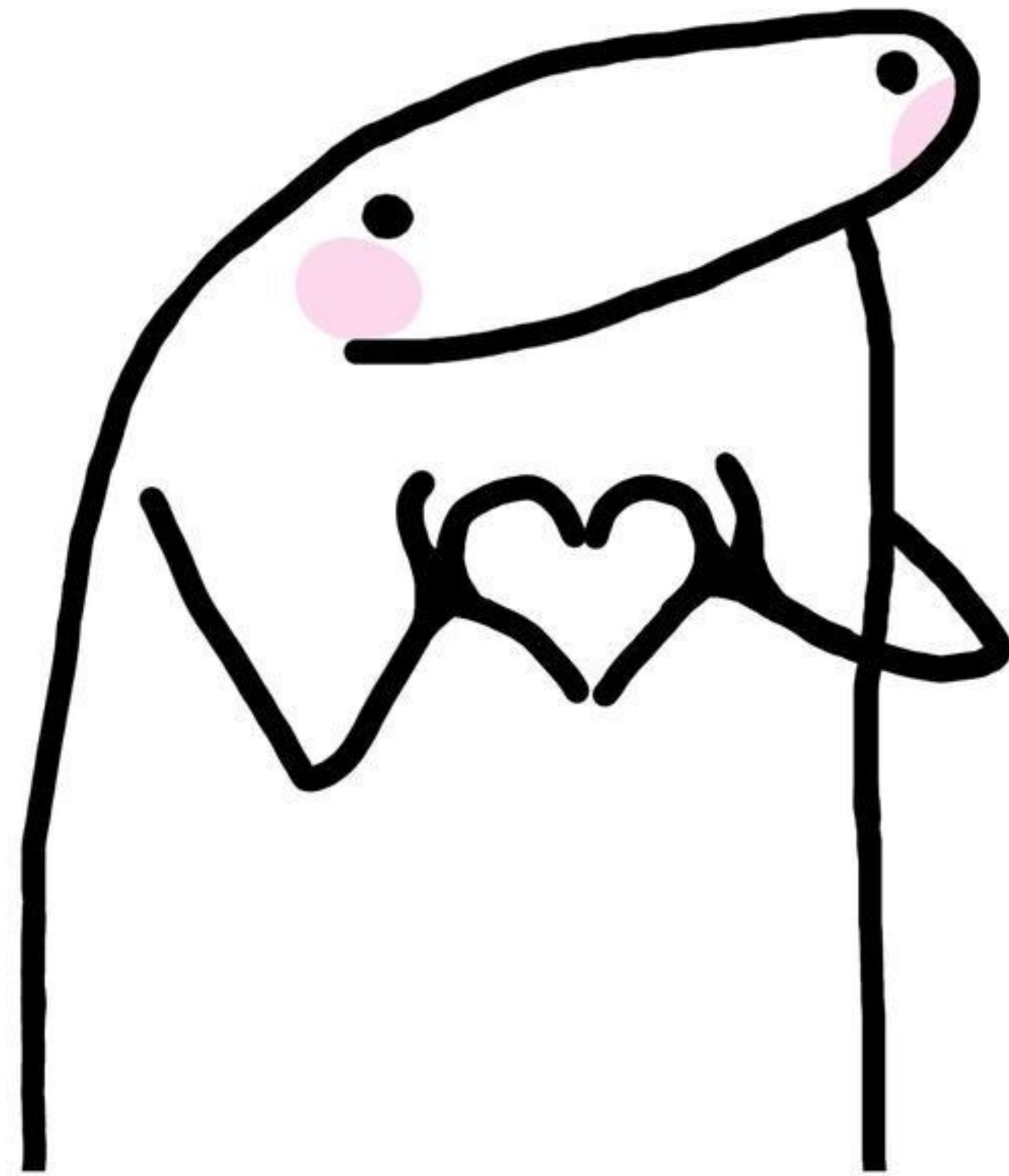
About me:

Rust consultant

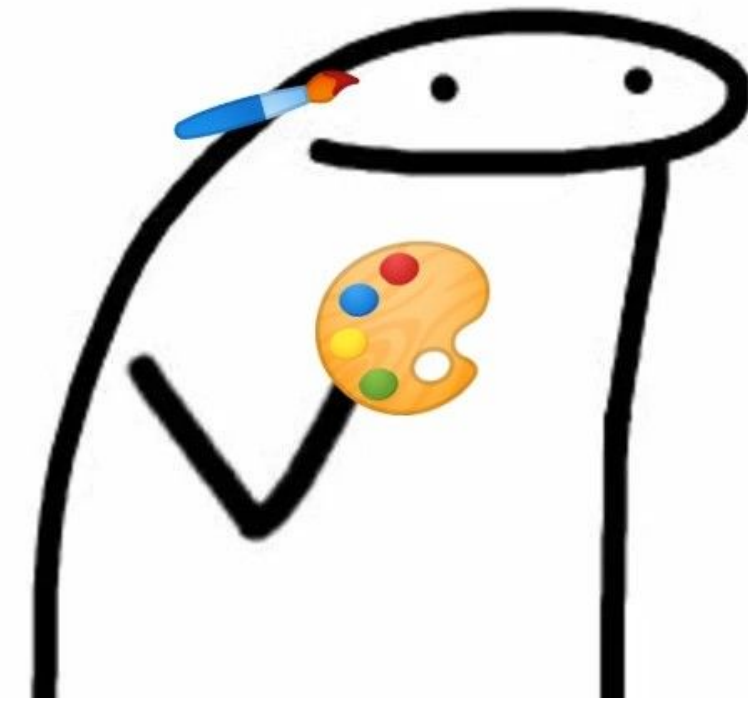
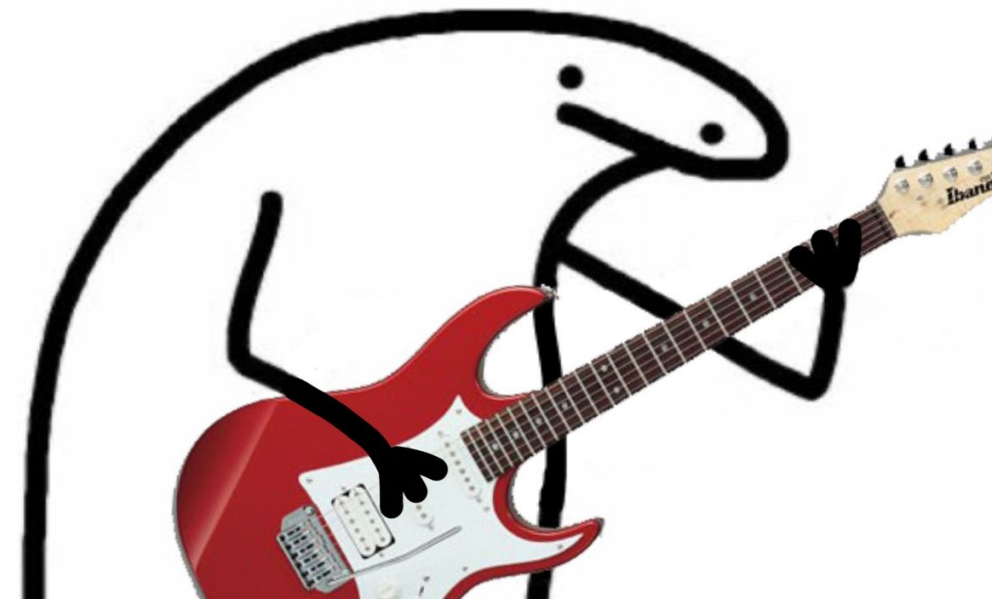
Podcast host

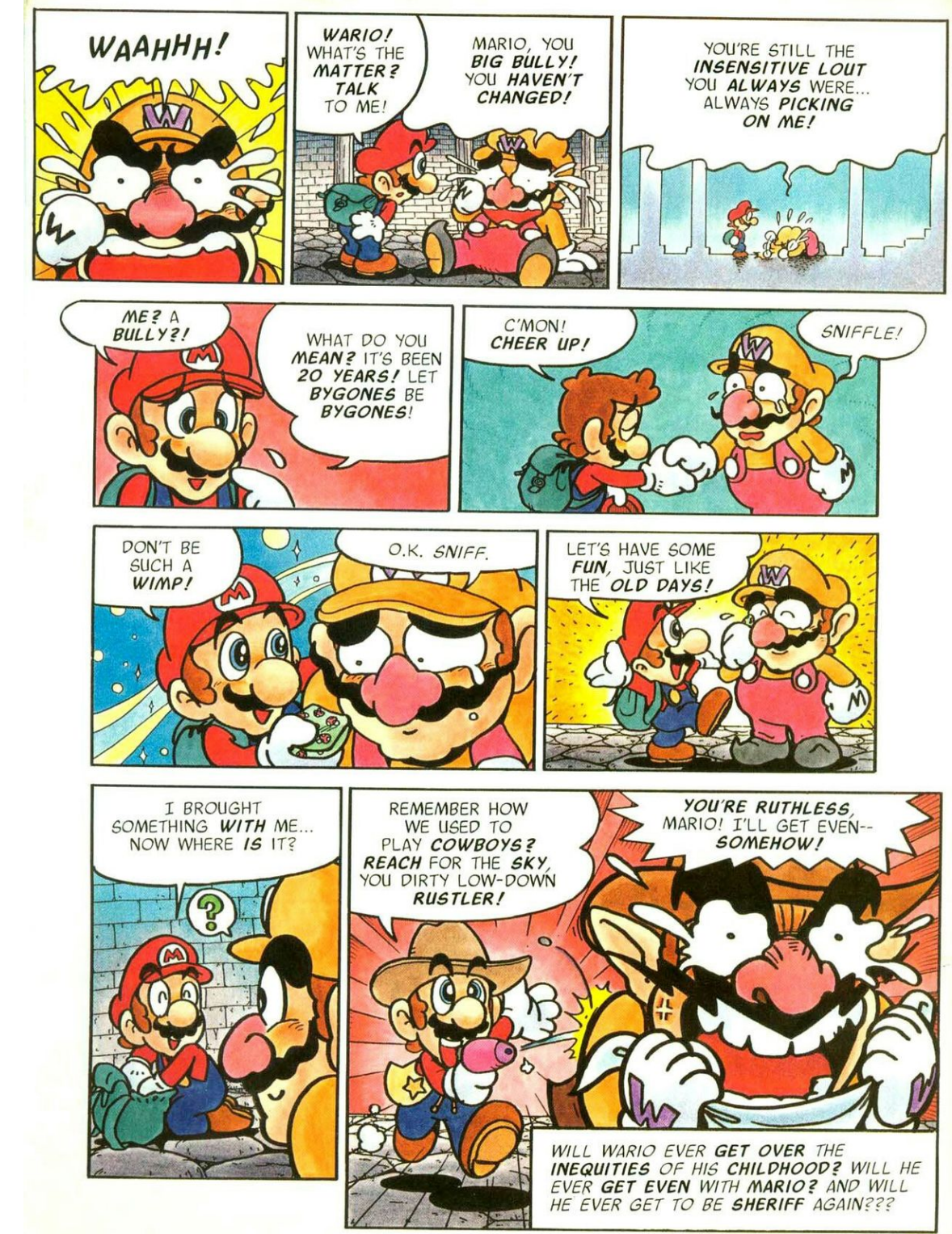
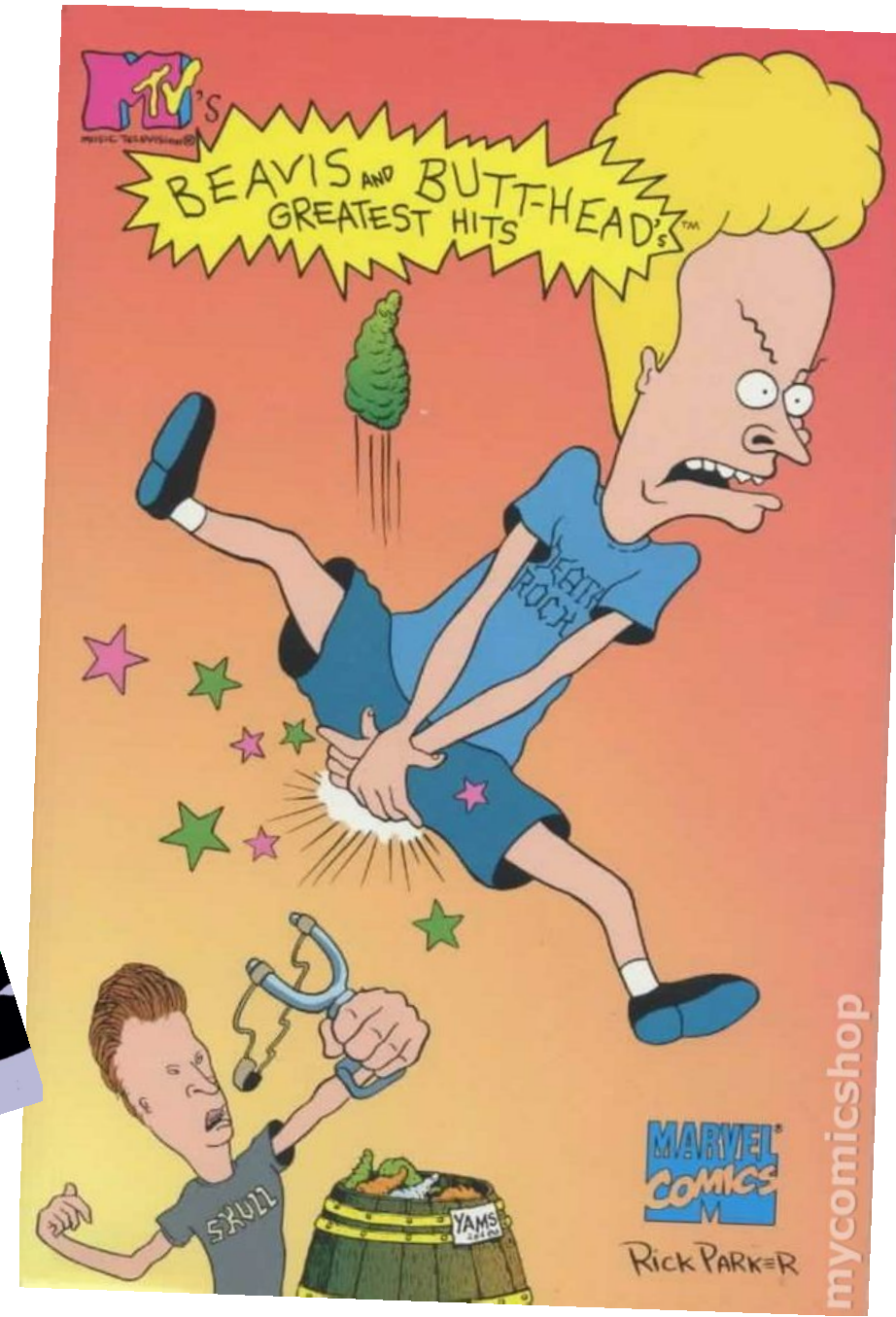
20 years of open source

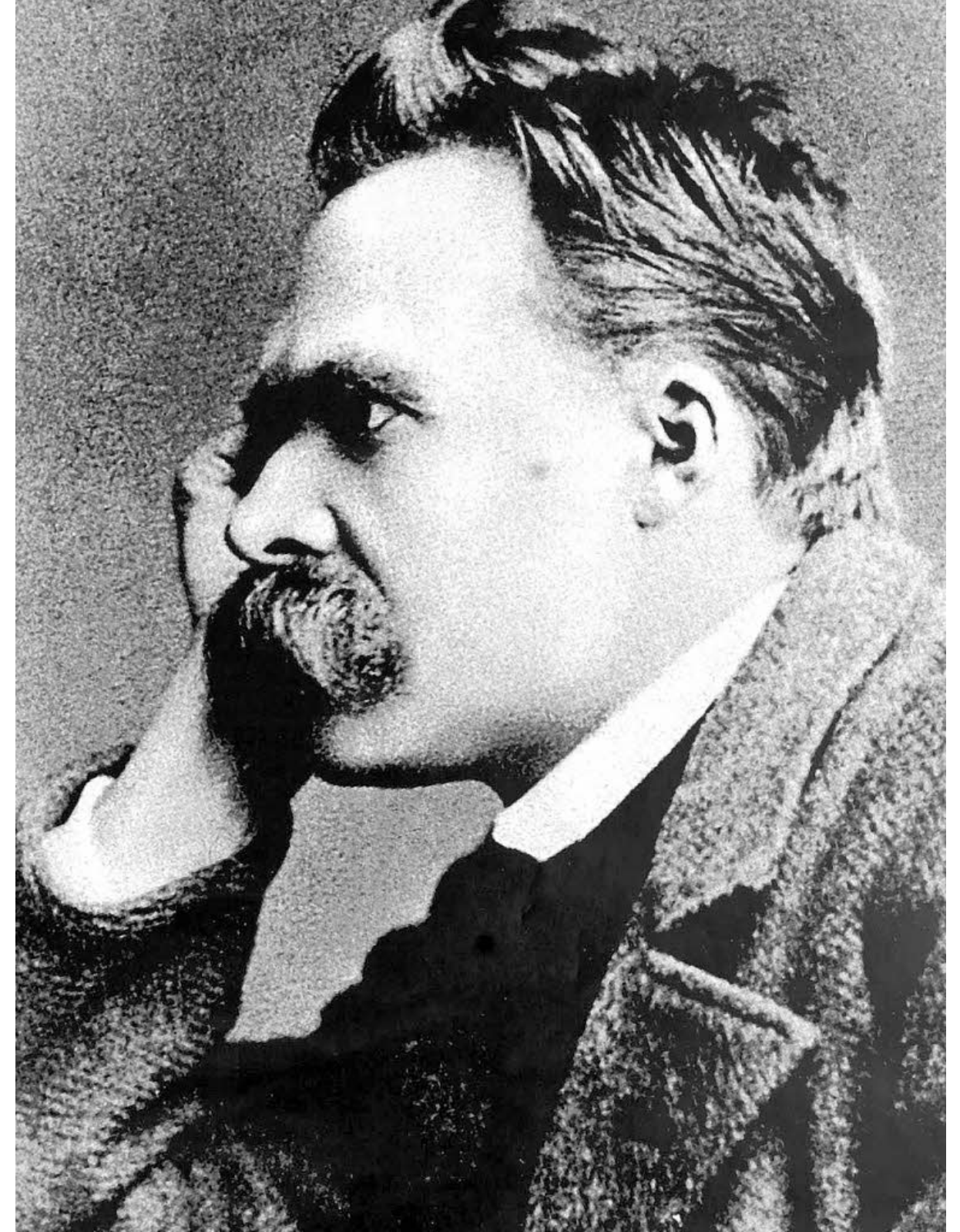




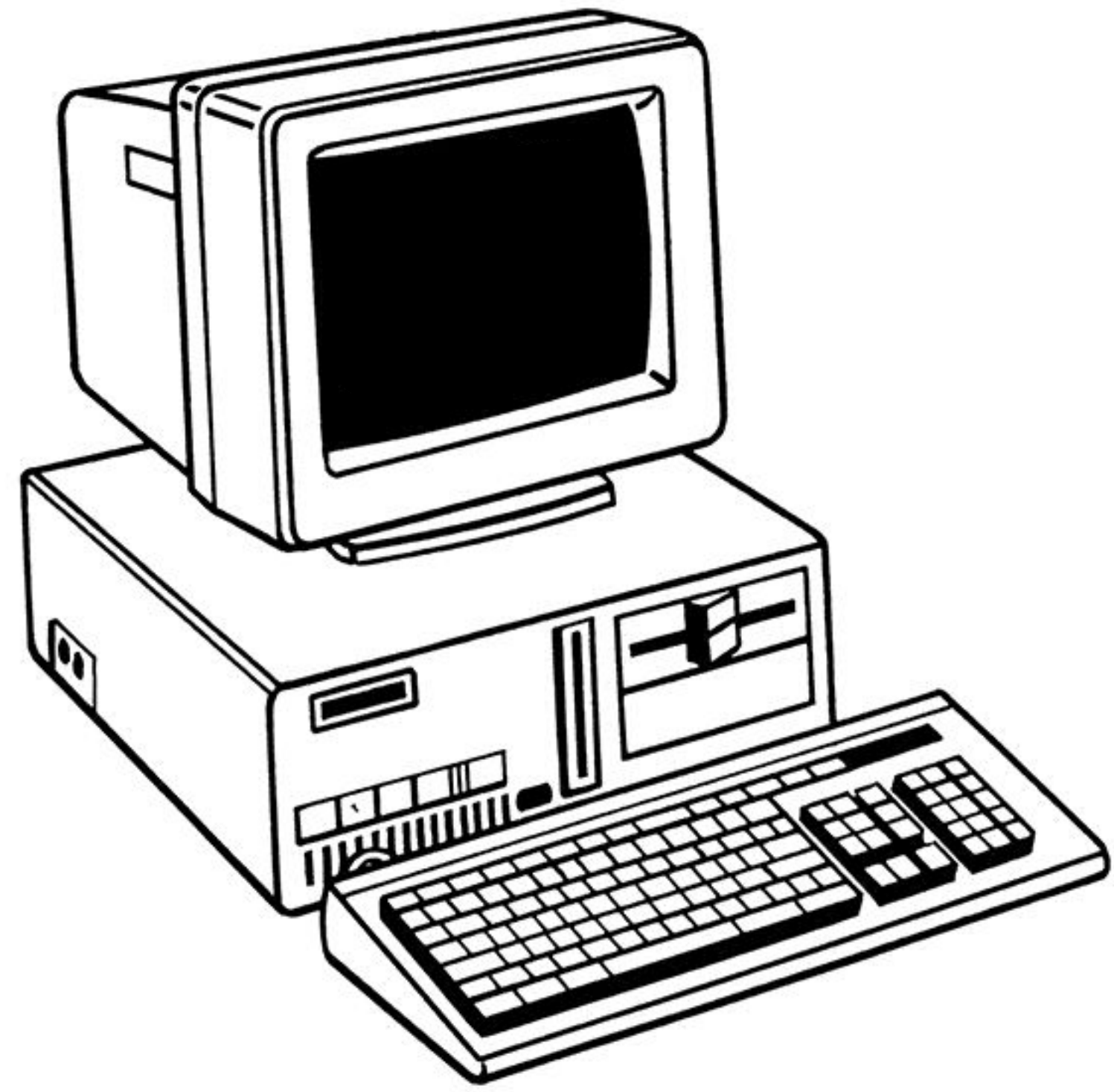
Art











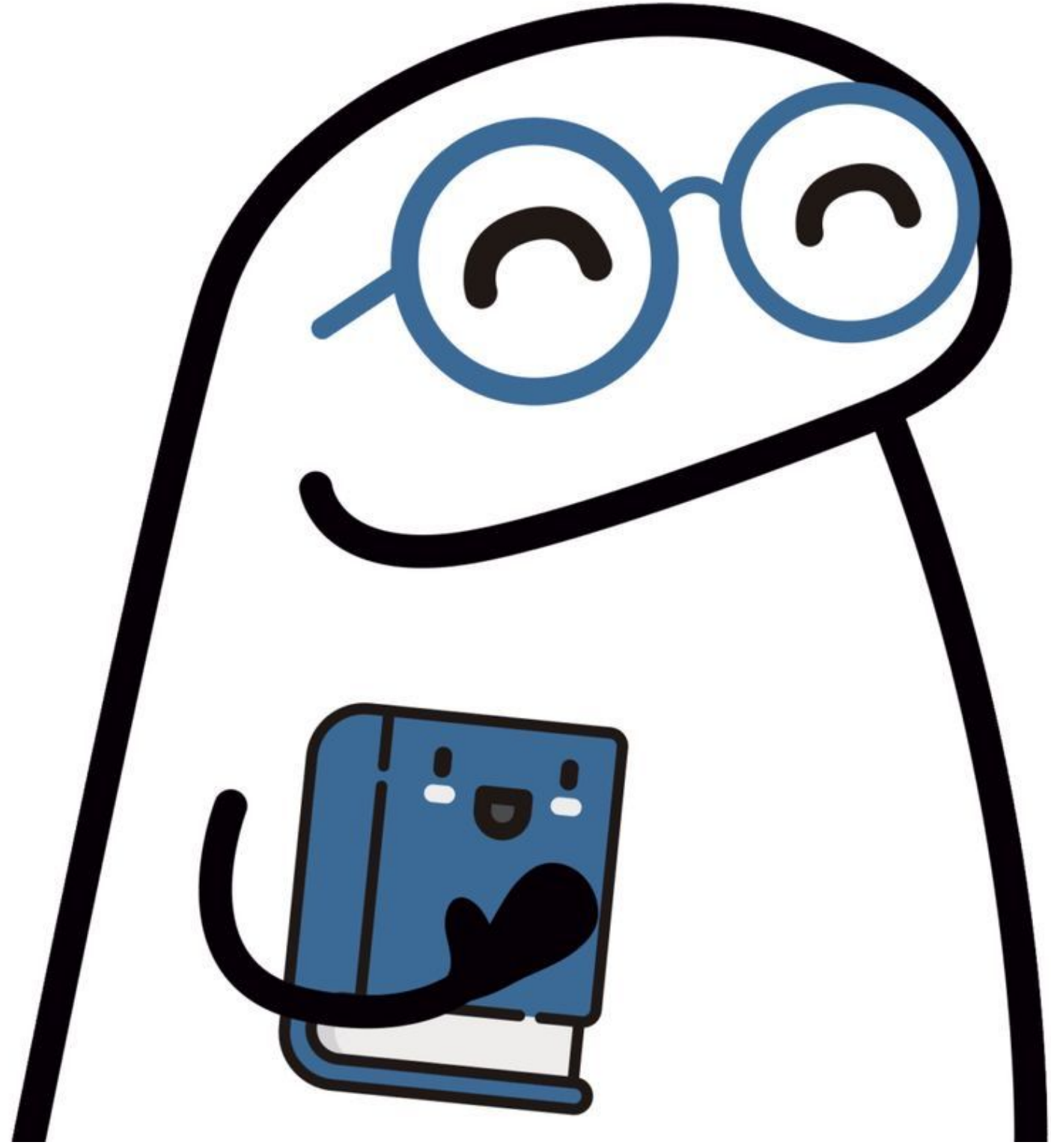
```
File Edit View Search Run Options Help
Untitled*
CLS
INPUT "What is your name"; name$
PRINT "Hello, "; name$; "!"
INPUT "How old are you"; age
INPUT "Which is your favourite Language?", language$
PRINT "Name:"; name$
PRINT "Age:"; age; " years old"
PRINT "favourite Language:"; language$

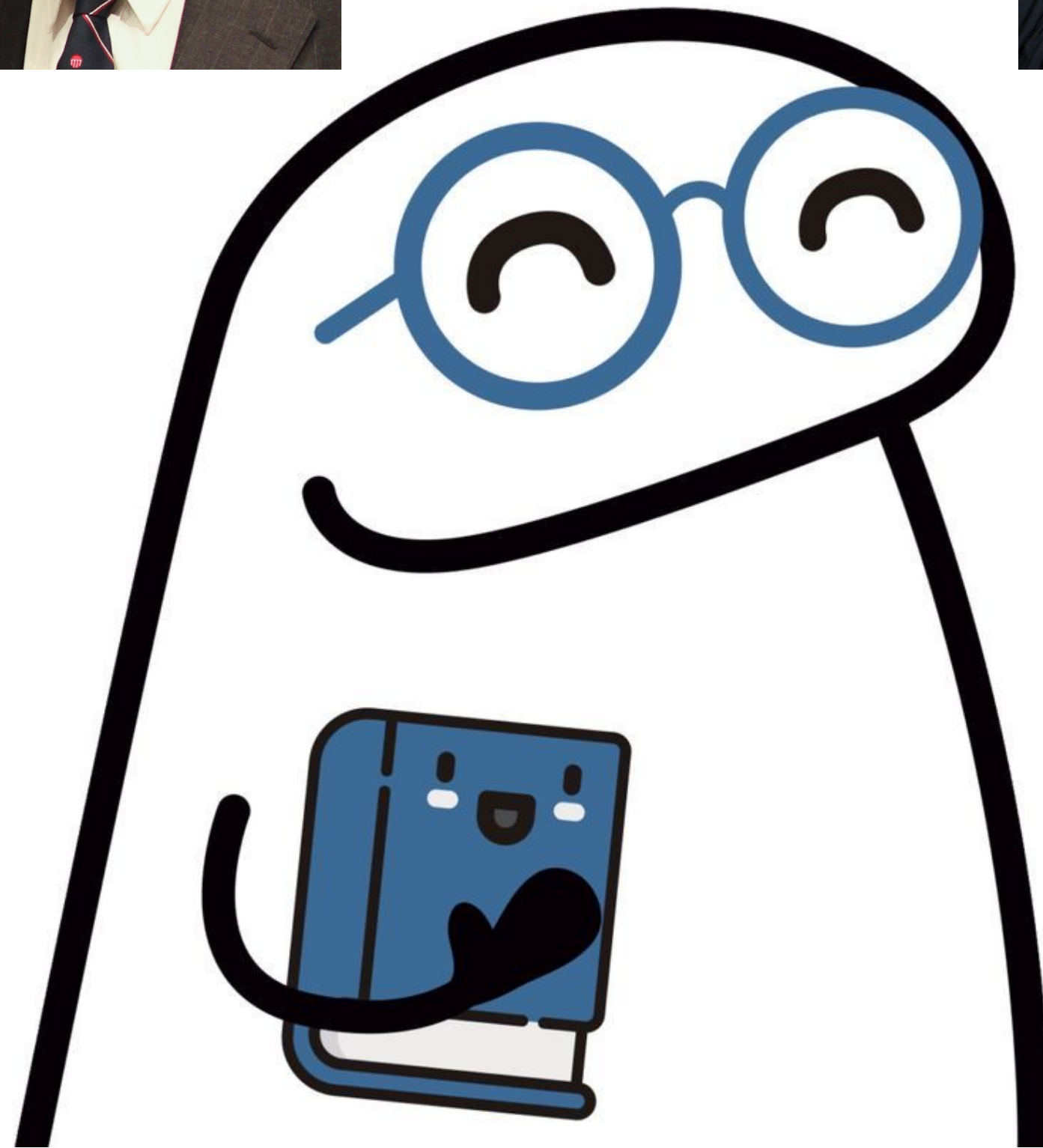
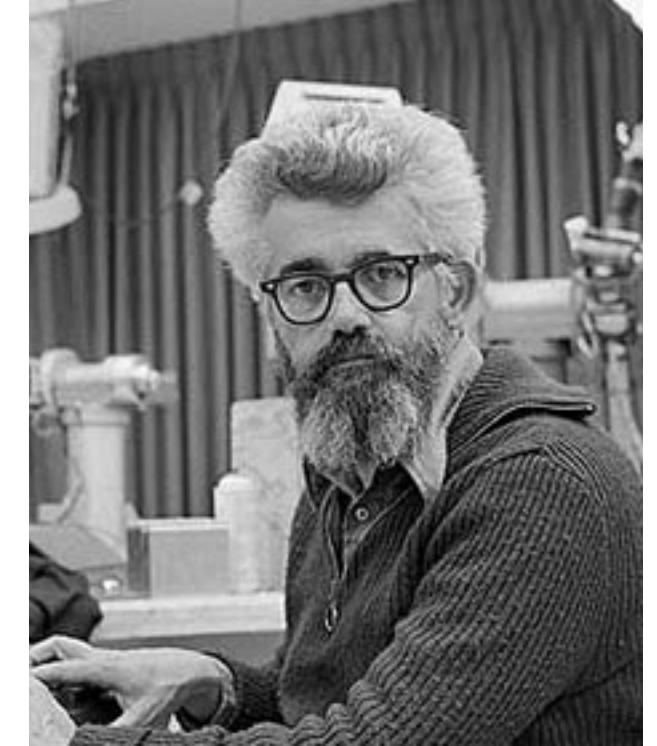
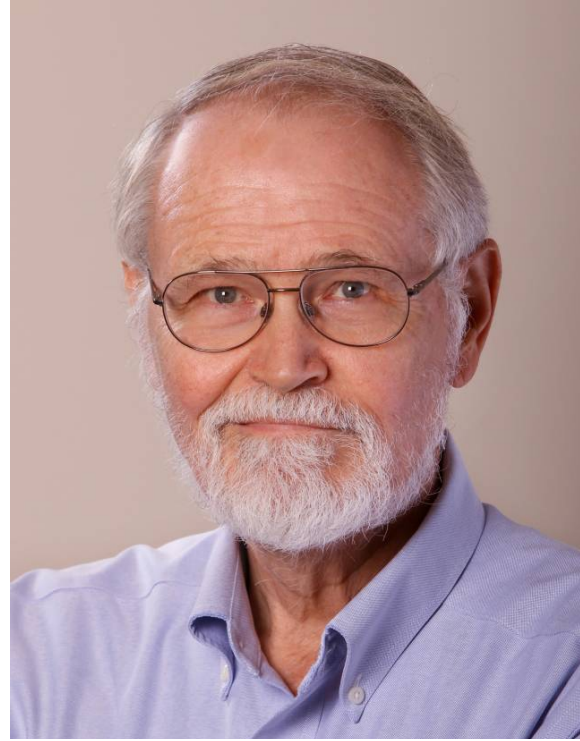
Status FindI
OK
1:39
```

THE
C
PROGRAMMING
LANGUAGE

Brian W. Kernighan • Dennis M. Ritchie

PRENTICE-HALL SOFTWARE SERIES





```
void swap(int* a, int* b) { int t = *a; *a = *b; *b = t; }

void quicksort(int arr[], int low, int high) {
    if (low < high) {
        int pivot = arr[high], i = (low - 1);
        for (int j = low; j ≤ high - 1; j++) {
            if (arr[j] < pivot) swap(&arr[++i], &arr[j]);
        }
        swap(&arr[i + 1], &arr[high]);
        int pi = i + 1;

        quicksort(arr, low, pi - 1);
        quicksort(arr, pi + 1, high);
    }
}
```

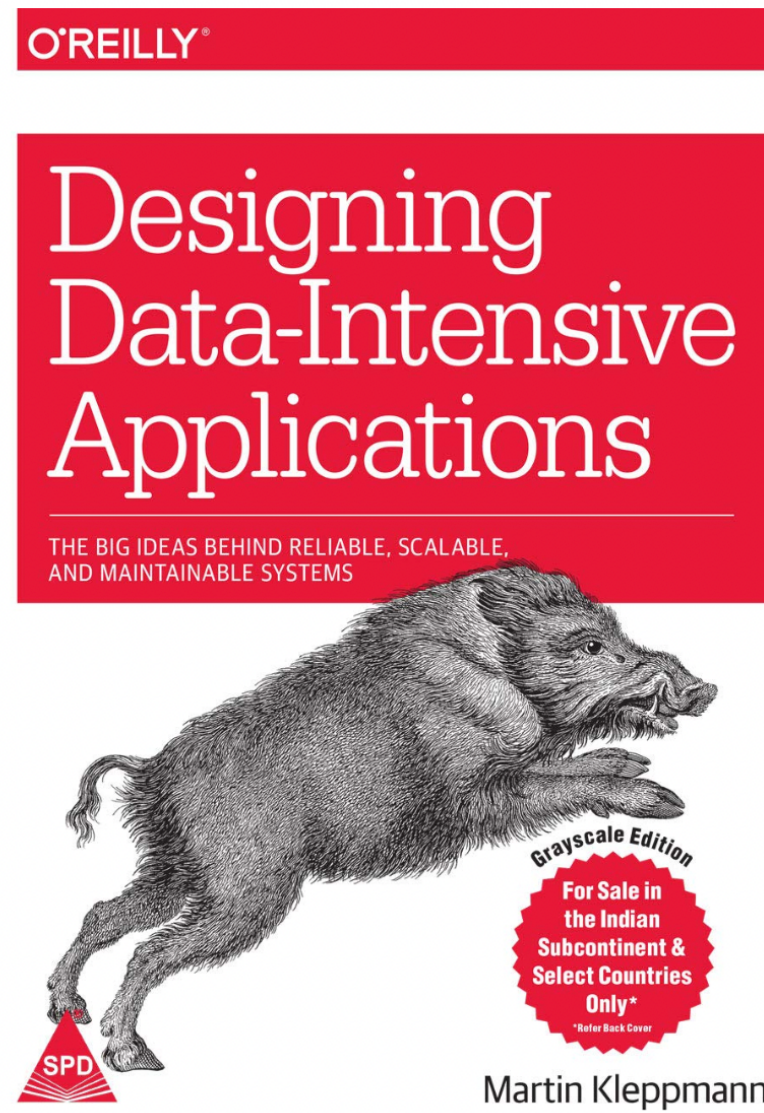
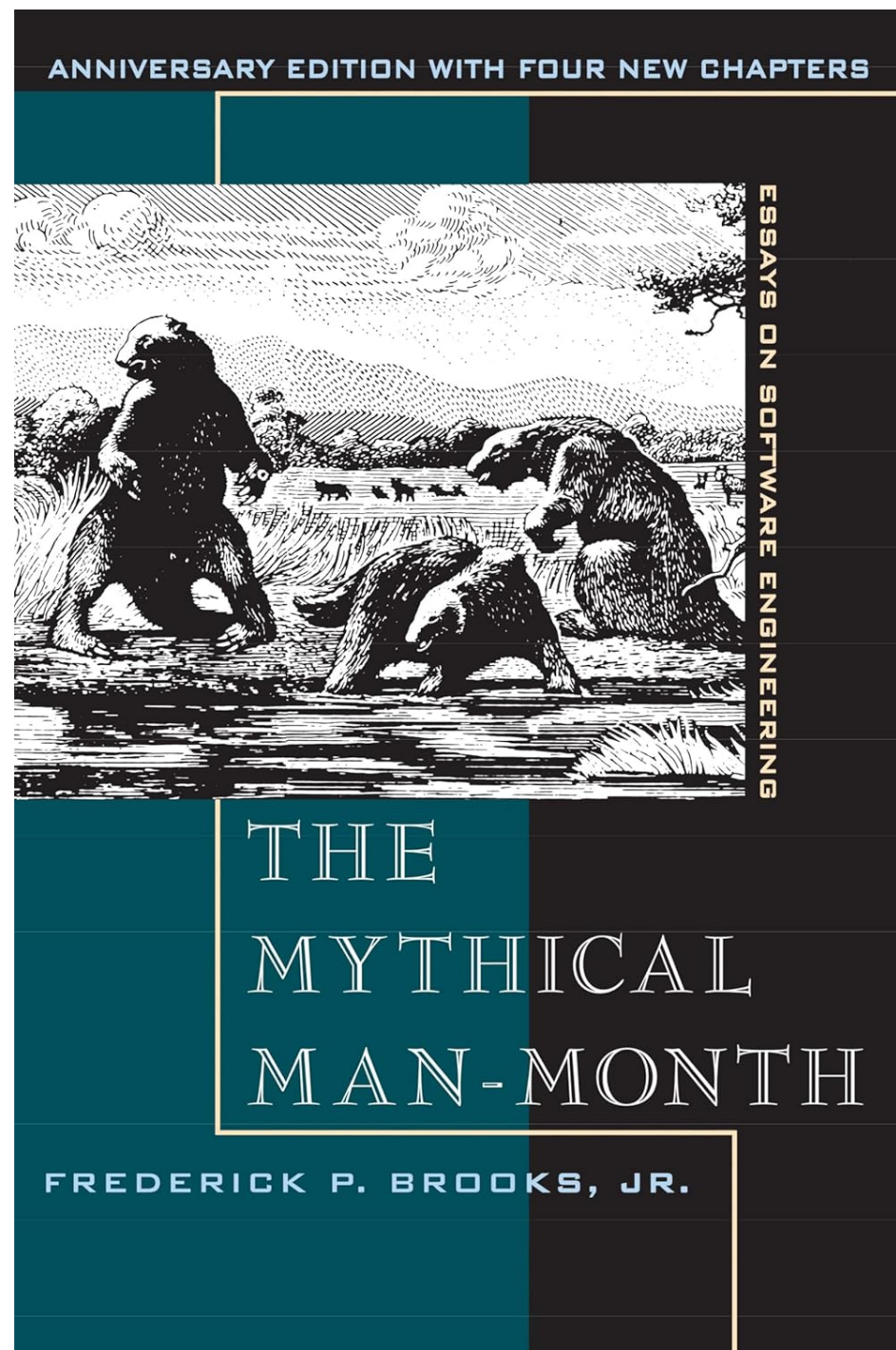
```
quicksort :: (Ord a) => [a] -> [a]
quicksort [] = []
quicksort (x:xs) =
    let smaller = quicksort [a | a <- xs, a <= x]
        bigger  = quicksort [a | a <- xs, a > x]
    in smaller ++ [x] ++ bigger
```

>>>>>>>, [> ,] < [[>>> + <<<-] > [<+>] <+<] > [<<<<<<<<+>>>>>>>>-] <<<<<<<< [[>>+
>+>>+<<<<<-] >> [<<+>>] < [>+>>+>>+<<<<<-] > [<+>] >>>> [-<] +< [>] +<< [>>]
<< [-]]] > [<+>] > [<<+>>] <+< [->] << [-] < [-] << [-] < [[>+<] <] >> [>] <+>>>>] > [<->
<+ [- [>+<] < [>+<] >>>>>>> [<<<<<<<<+>>>>>>>-] <<<<<<] << [>>+<<] > [>] >+>
>+<<<-] > [<+>] >>>>>> [<+<+>>] < [>+<] <<< [>+] [<-] < [<] >> [<<+> [-] +>] <->
] >> [-] +<<< [>>+<<] >> [->] <<<<< [>+<] < [>+<] >>>>>>> [<<<<<<<<+>>>>>>-]
<<] > [[-] <<<<<< [>>+>>>>>+<<<<<<<-] >> [<<+>>] >>>>> [- [>> [<<<+>>>-] < [>+<]
< [>+<] >] << [[>>+<<] <]] >] <<<<<<-] > [>>>>>>+<<<<<<-] << [[>>>>>>+<<<<<<<
-] > [<+>] <+<] < [[>>>>>>>>+<<<<<<<-] >> [<+>] <+<<] >+> [<-> [>+<] < [<] > [[<
+>] >] >>> [<<<<+>>>>-] << [<+>] >>] < [<->] >>>] <<<<<<] >>>>>>>>>>>>>>> [. >]

qsort ← {1 ≥ ≠ U ω : ω · p ← ω [? ≠ ω] · (∇(ω ≤ p) / ω) , ∇(ω > p) / ω}



Engineering



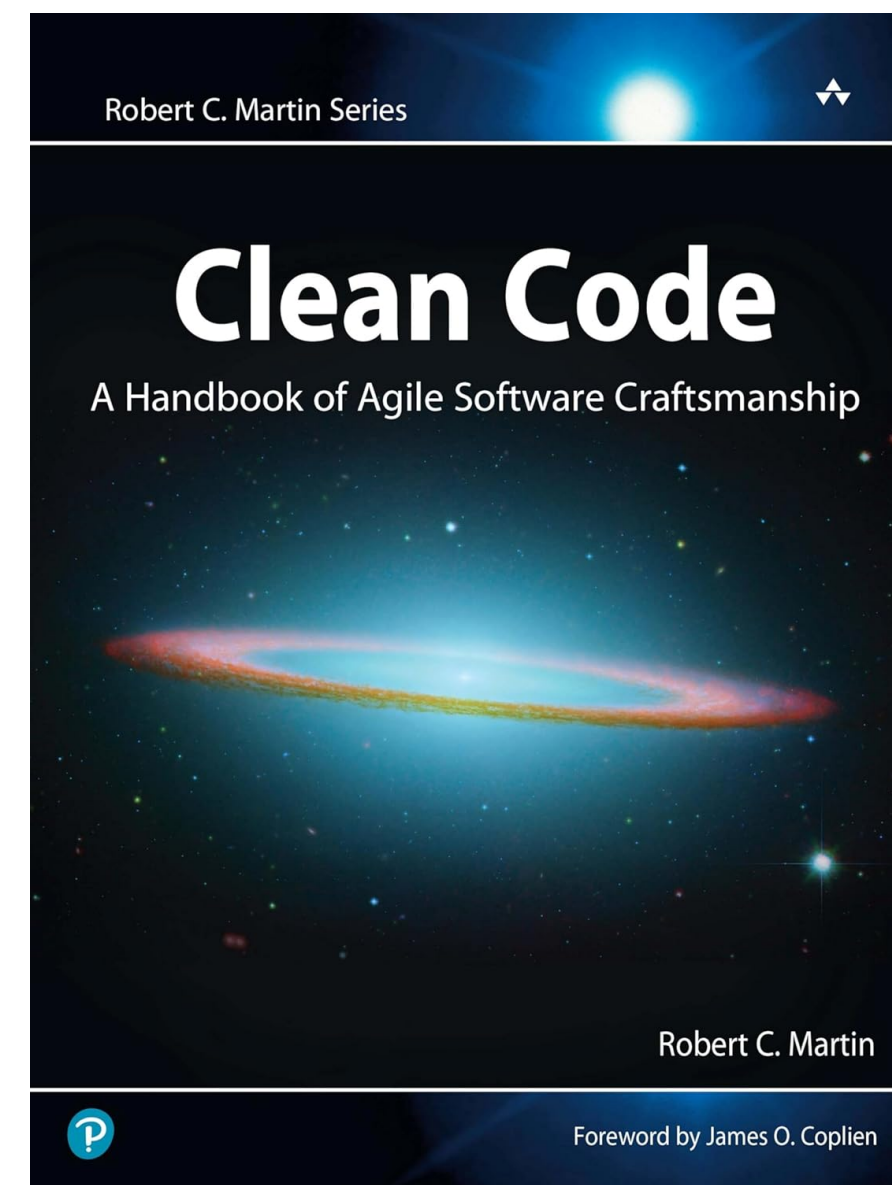
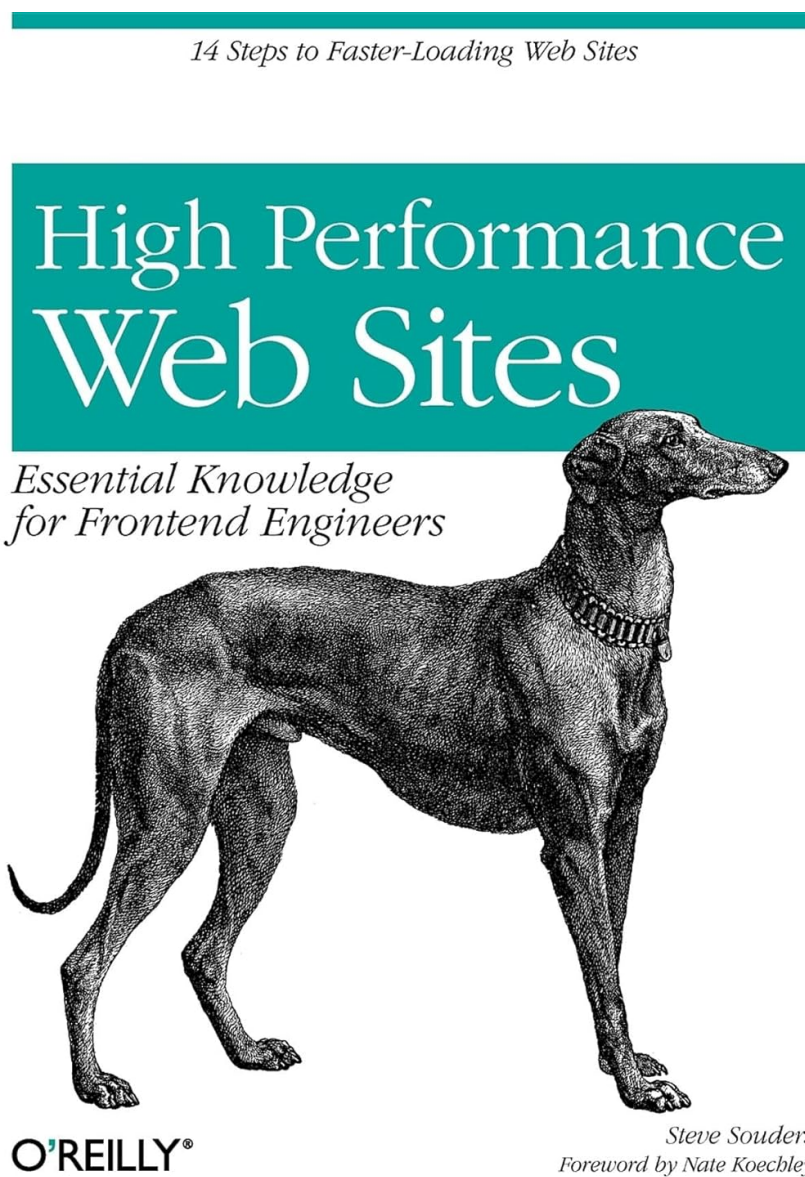
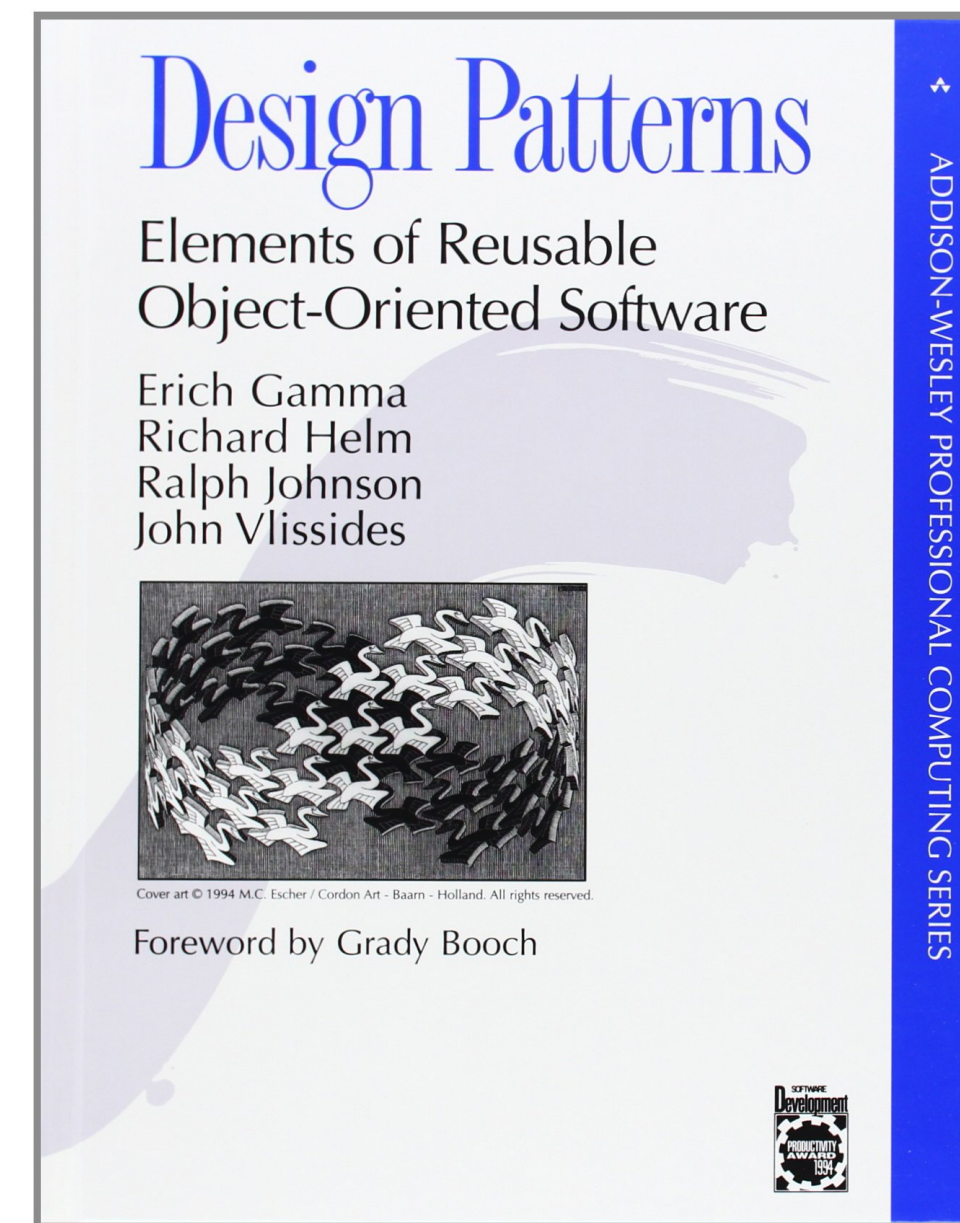
From the authors of *The Visible Ops Handbook*

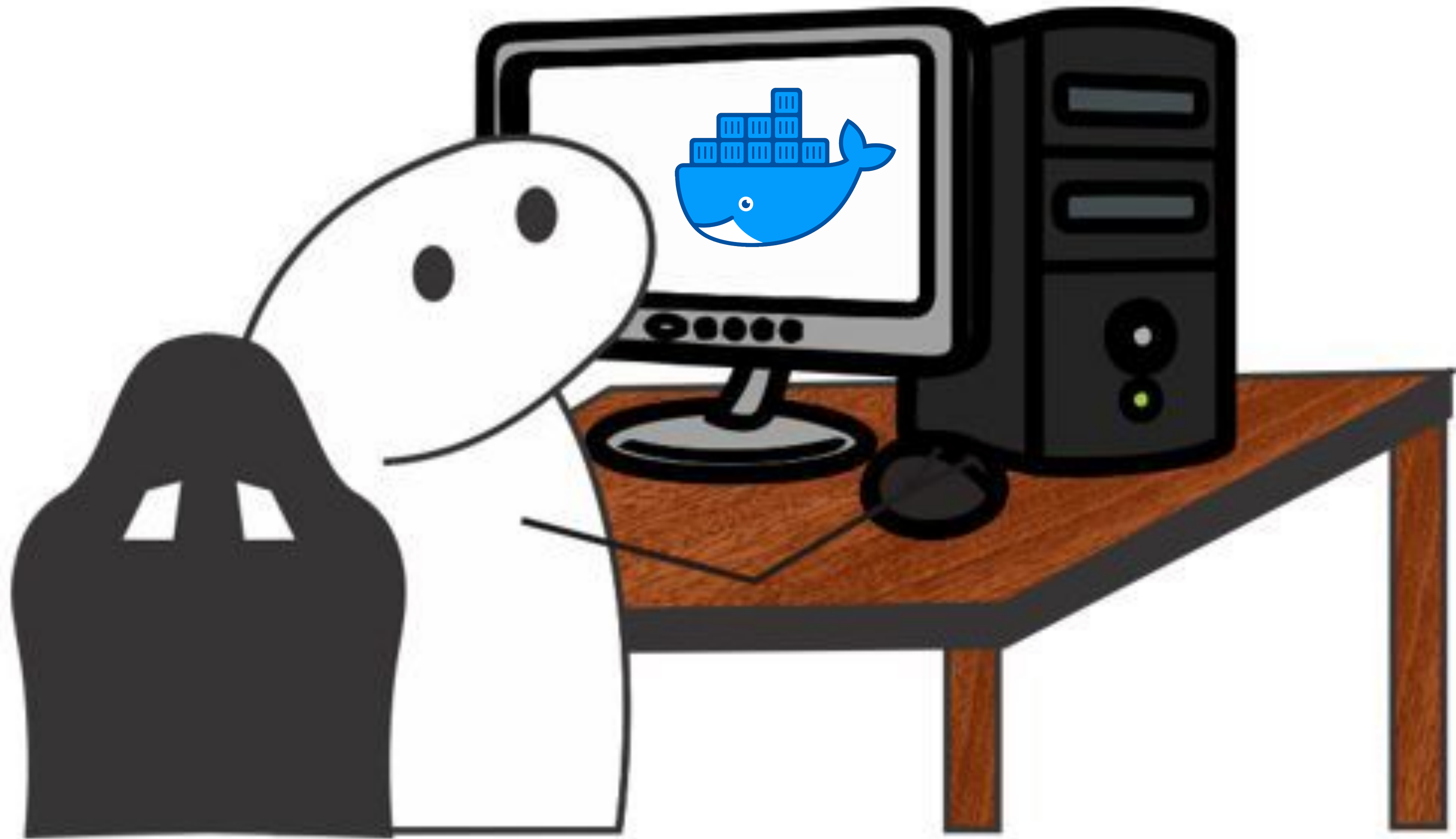


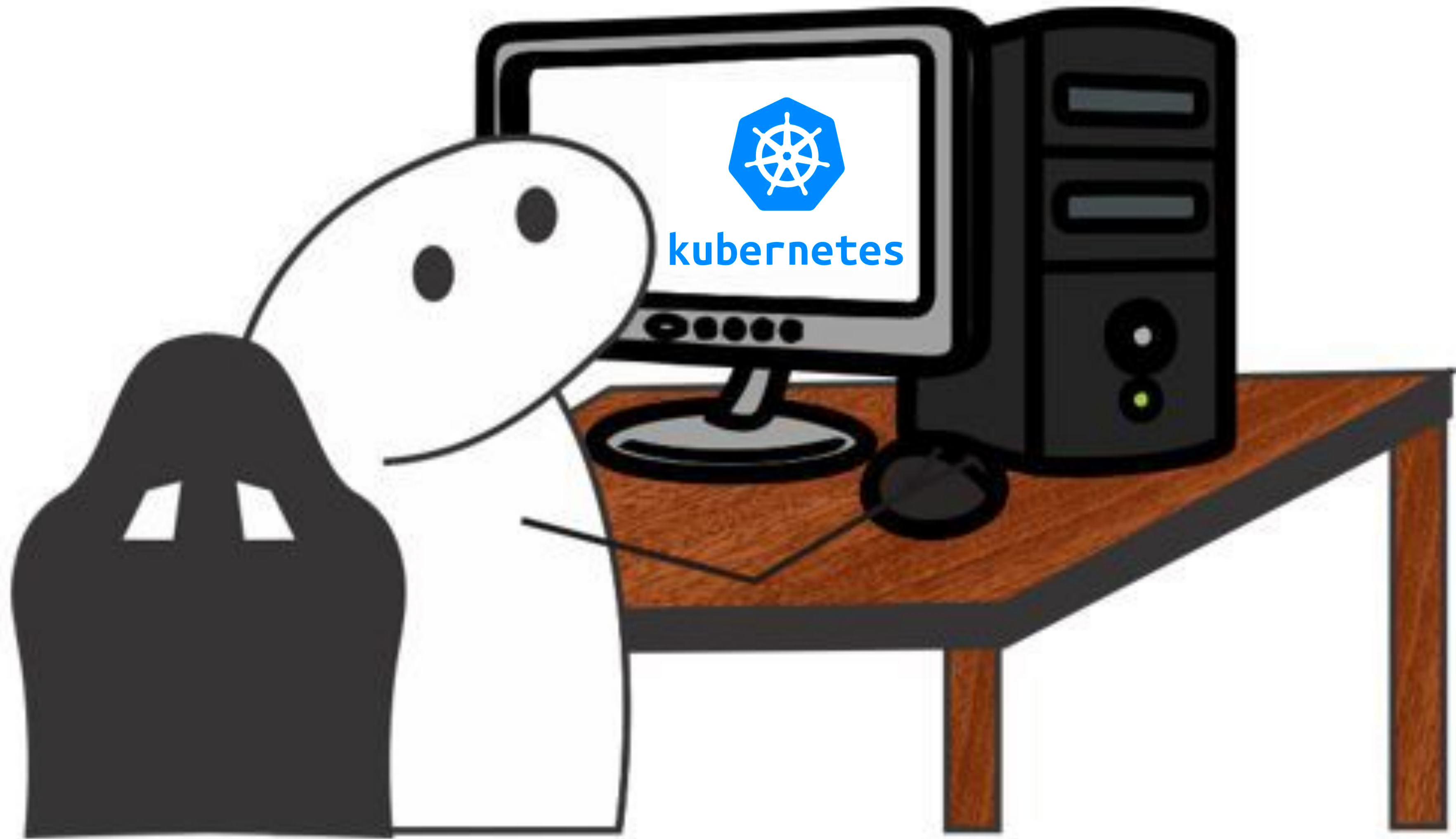
The Phoenix Project

A Novel About IT, DevOps, and Helping Your Business Win

Gene Kim, Kevin Behr, and George Spafford









Google Cloud Platform

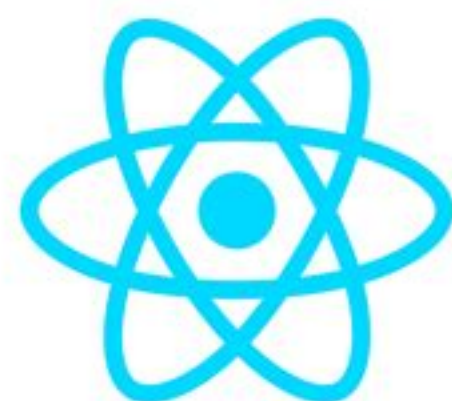


argo



HashiCorp
Terraform

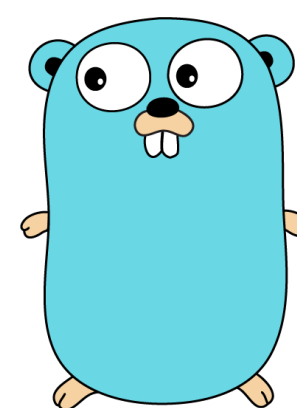
TypeScript



React

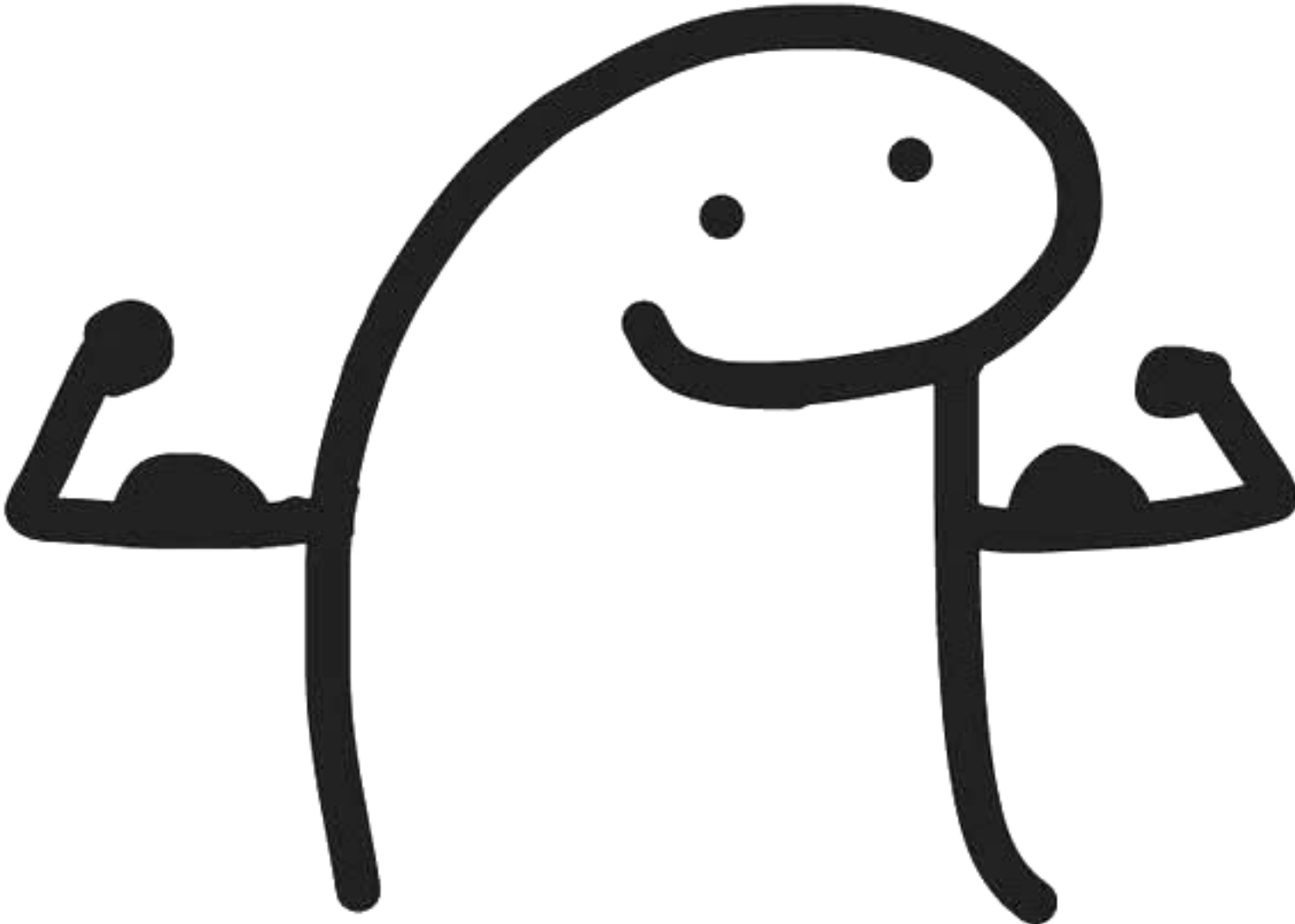


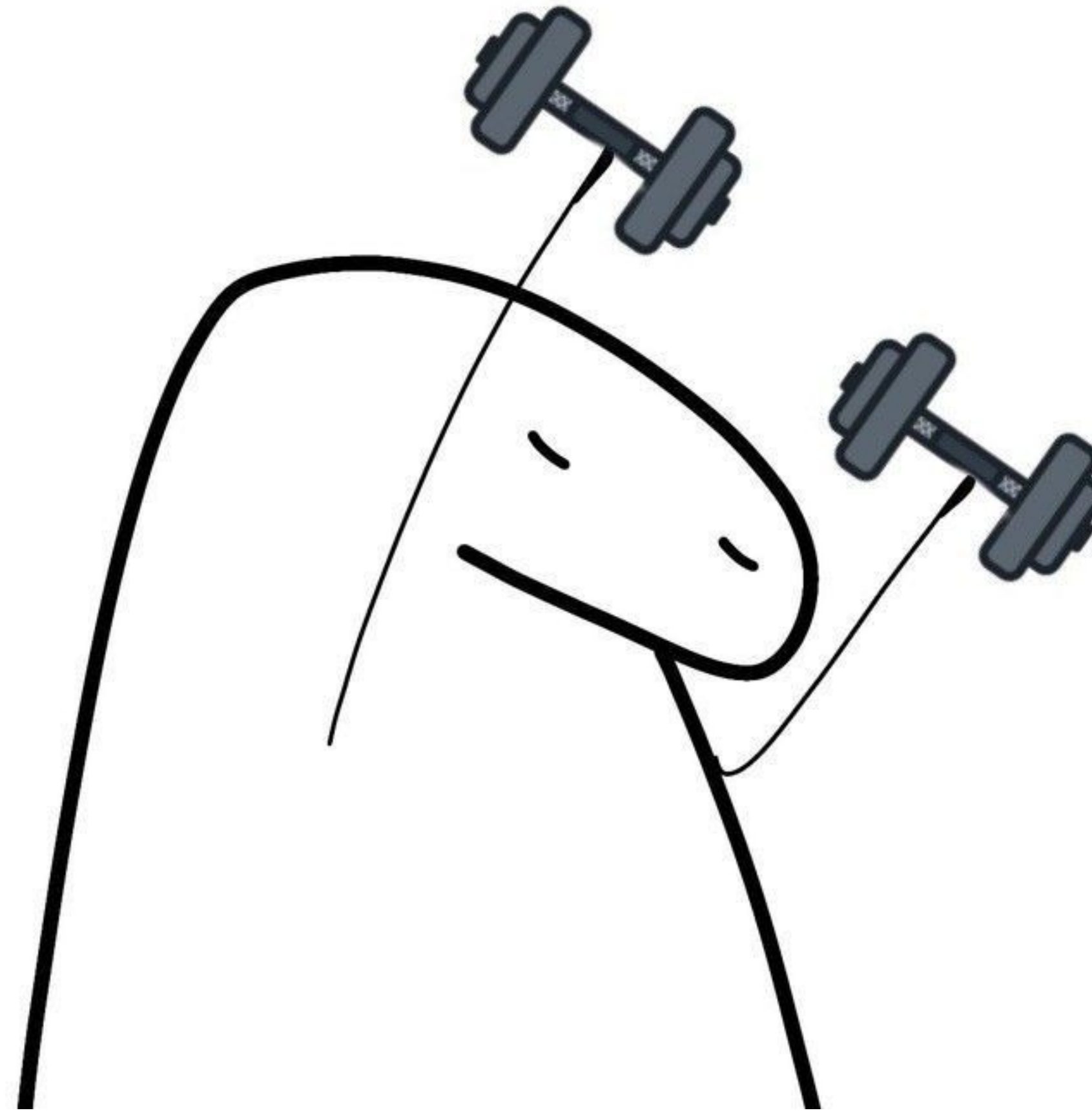
Microservices

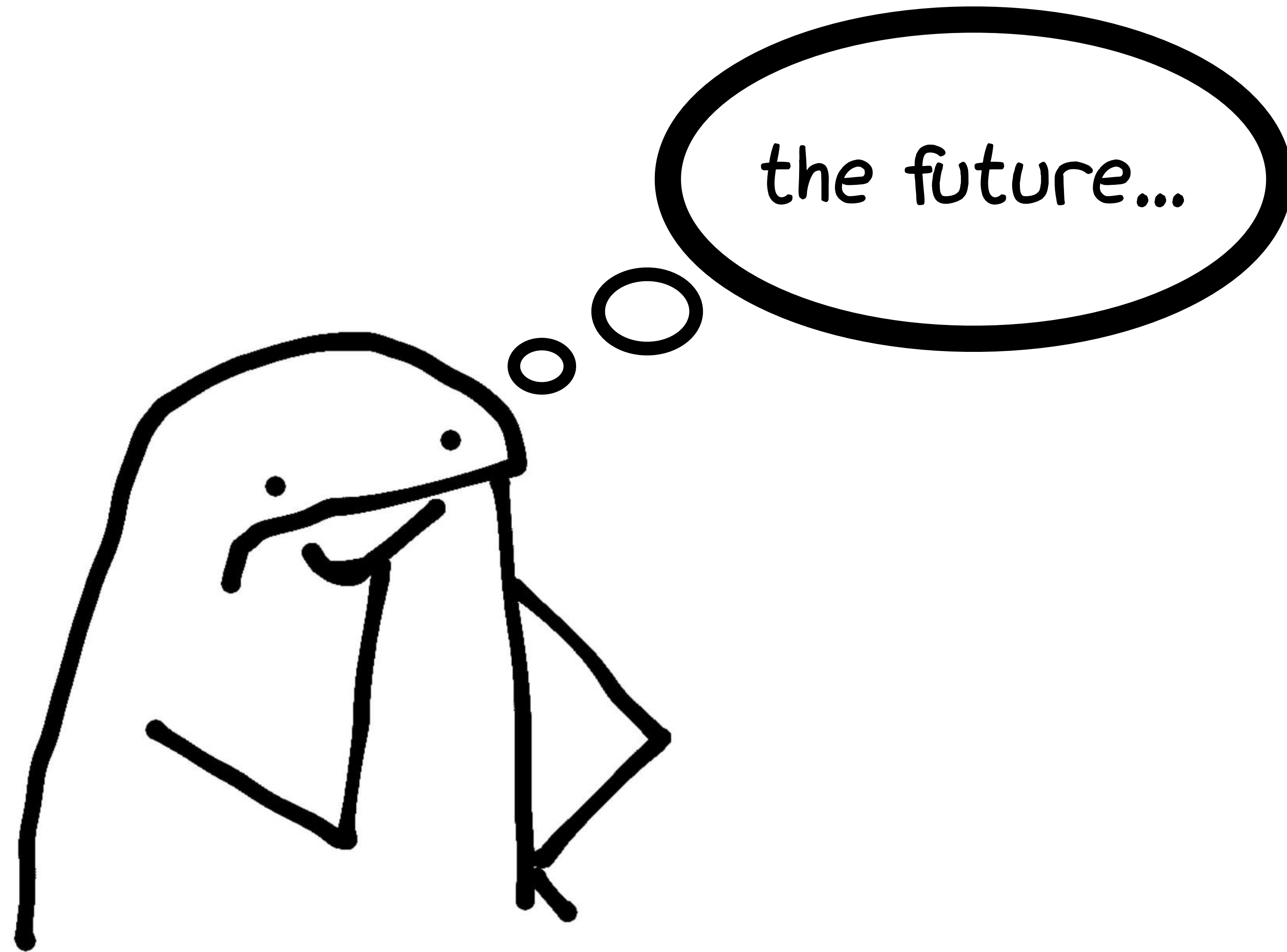


Golang

php7







the future...



PIENSE

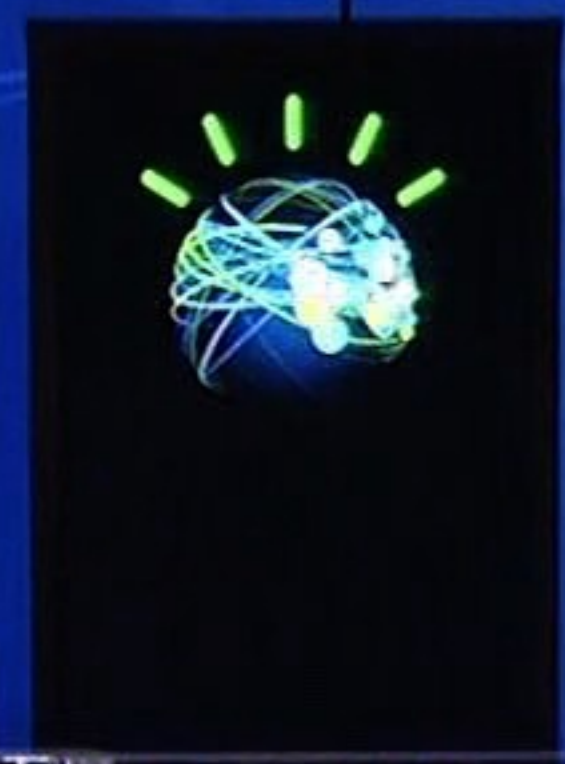
ΣΚΕΨΟΥ

THINK

DENKE

SMACHMIS

PENSER



\$24,000

\$77,147

\$21,600

Who is Stoker?
(FOR ONE WELCOME OUR
NEW COMPUTER OVERLORDS)

Who is Bram
Stoker?

WHO IS
BRAM STOKER?

\$1,000

\$ 17,973

\$5600



DeepMind





LEE SE DOL
02:58:01

ALPHAGO
02:58:55

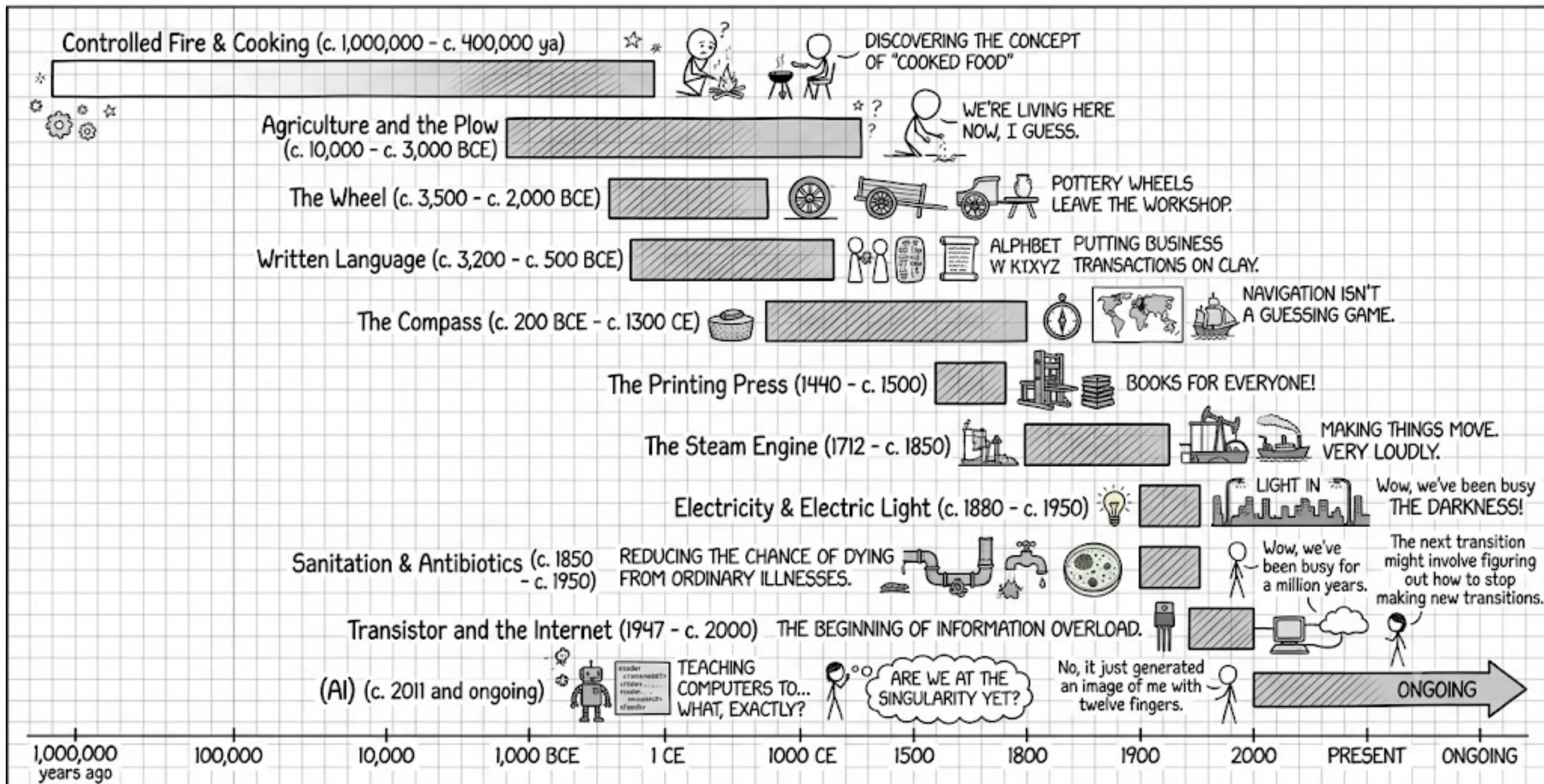
Chess: 1997, IBM Deep Blue

Jeopardy: 2011, IBM Watson

Go: 2017, AlphaGo

StarCraft II: 2019, AlphaStar

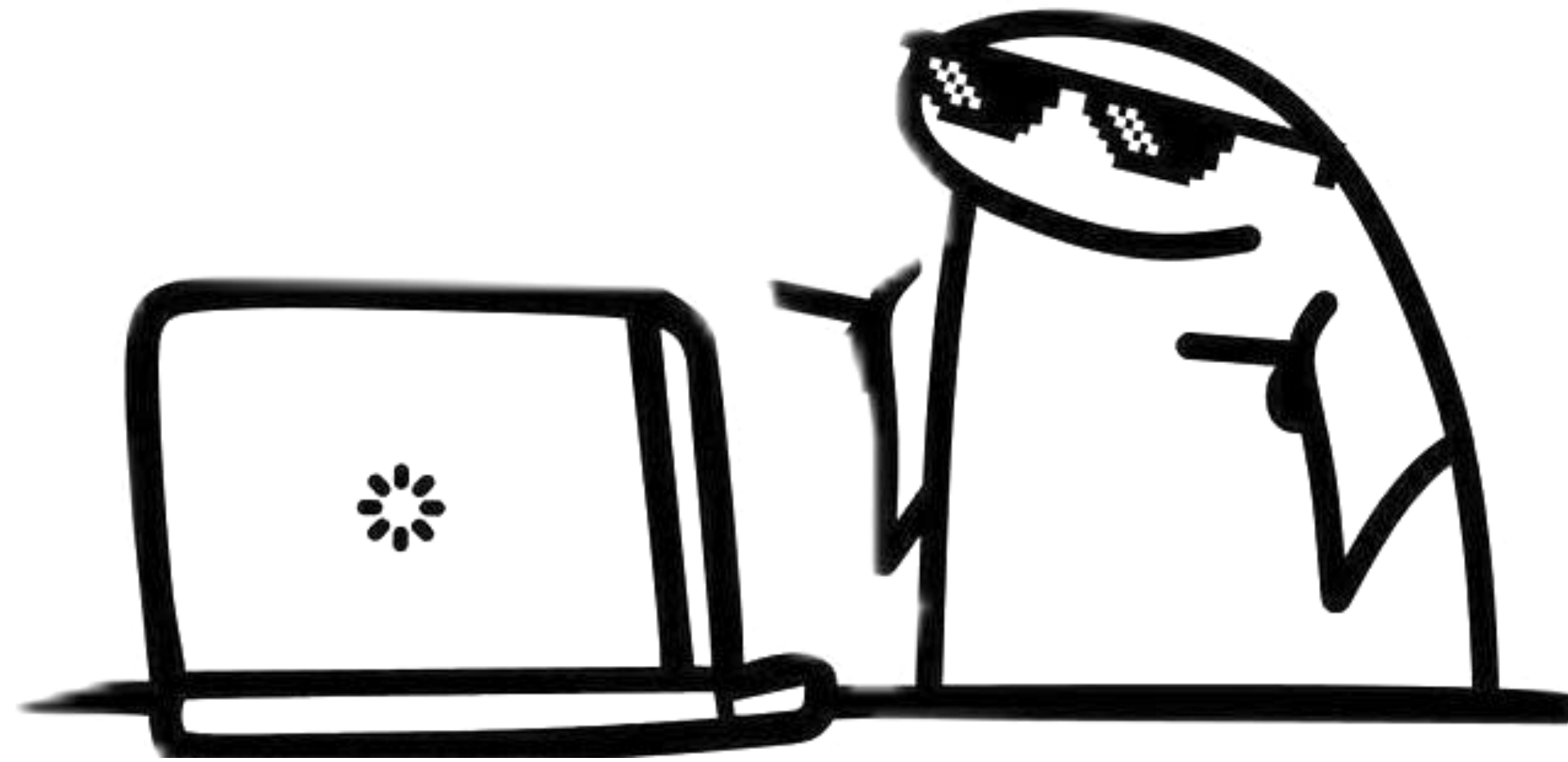
A BRIEF HISTORY OF HUMAN INVENTIONS (LOG SCALE) [TIME SPAN IS LOGARITHMIC]



March 2023

April 2025







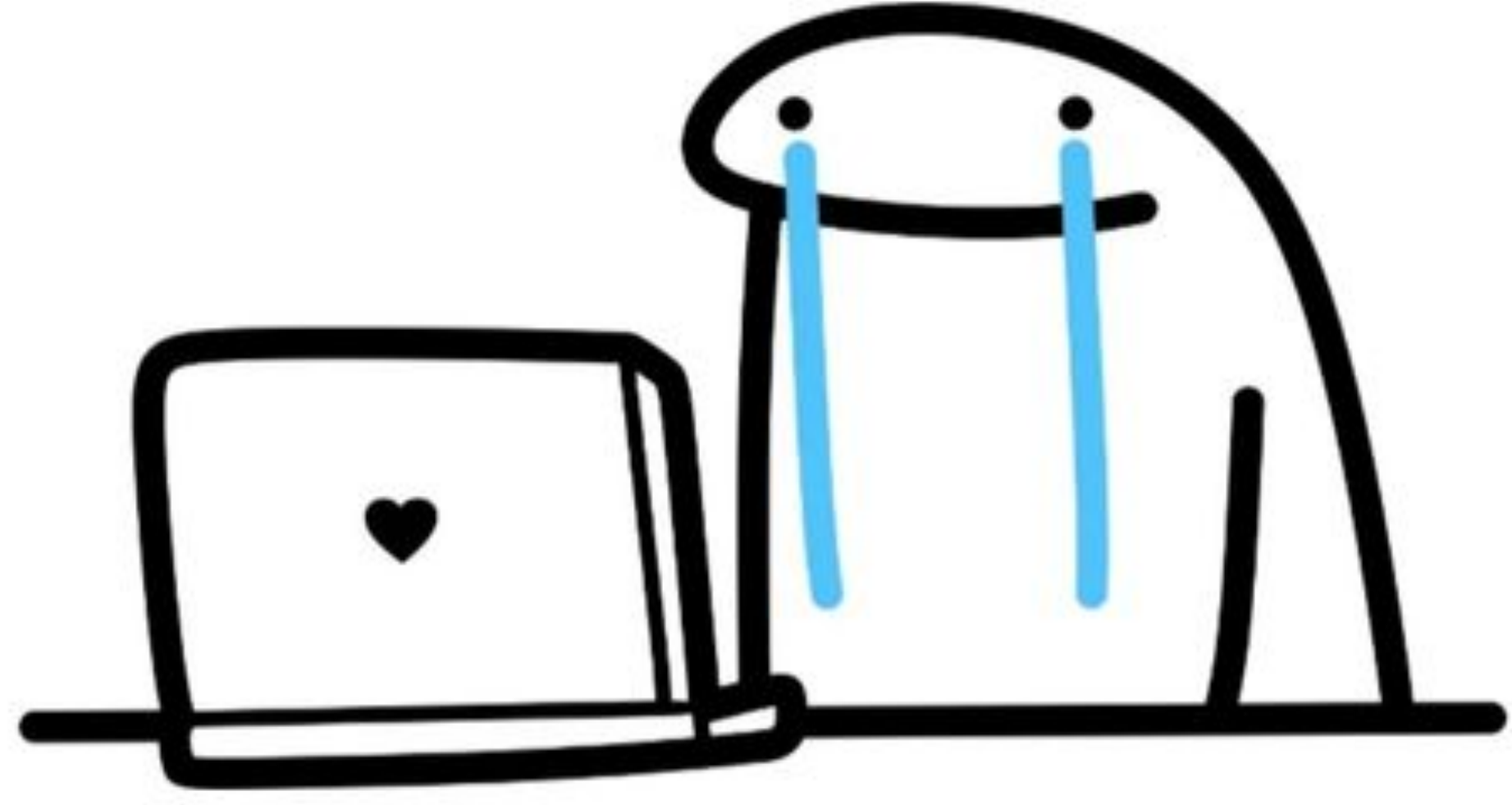
ME


Claude

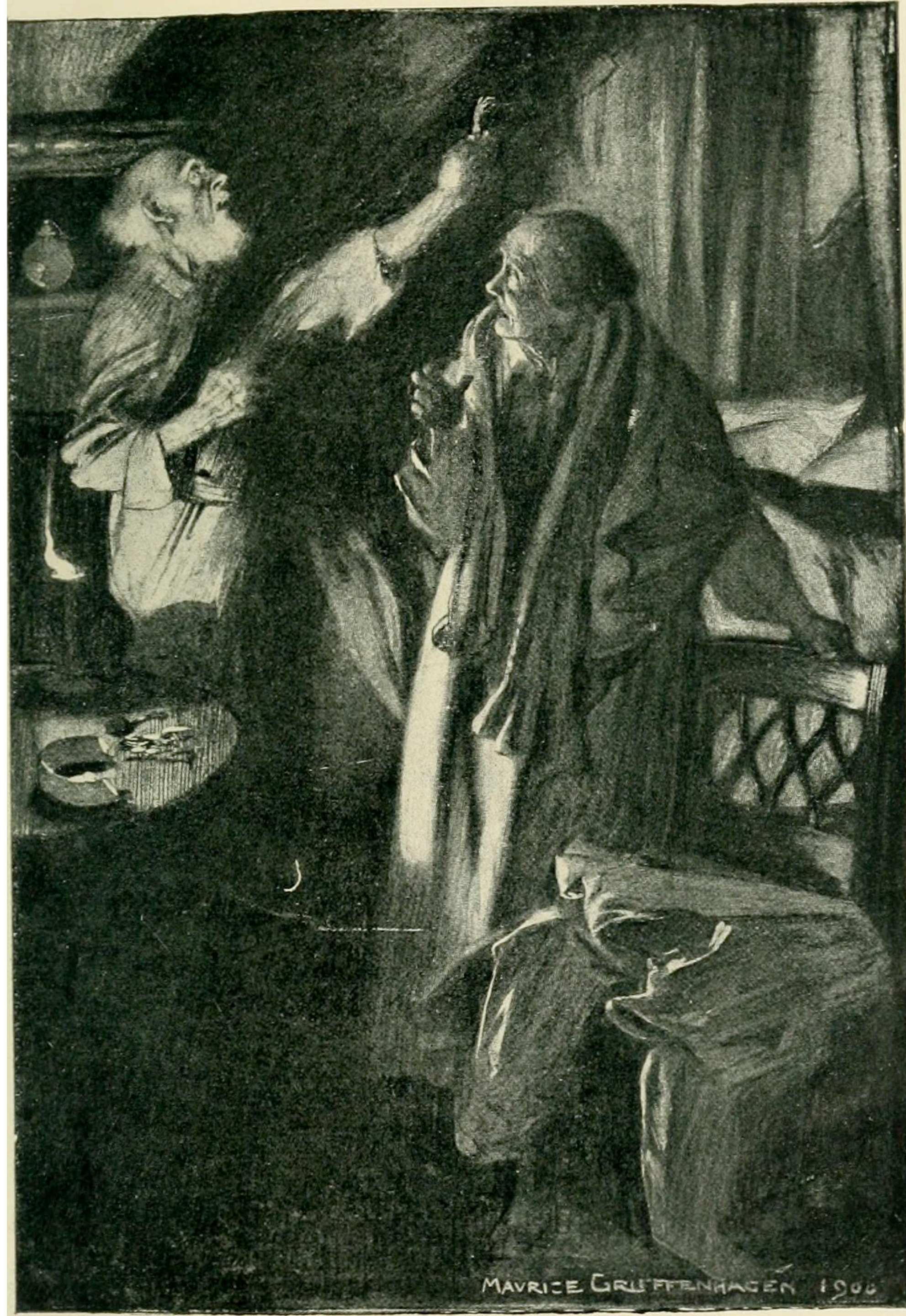


"One more prompt"

Something has changed



Computers stopped being fun







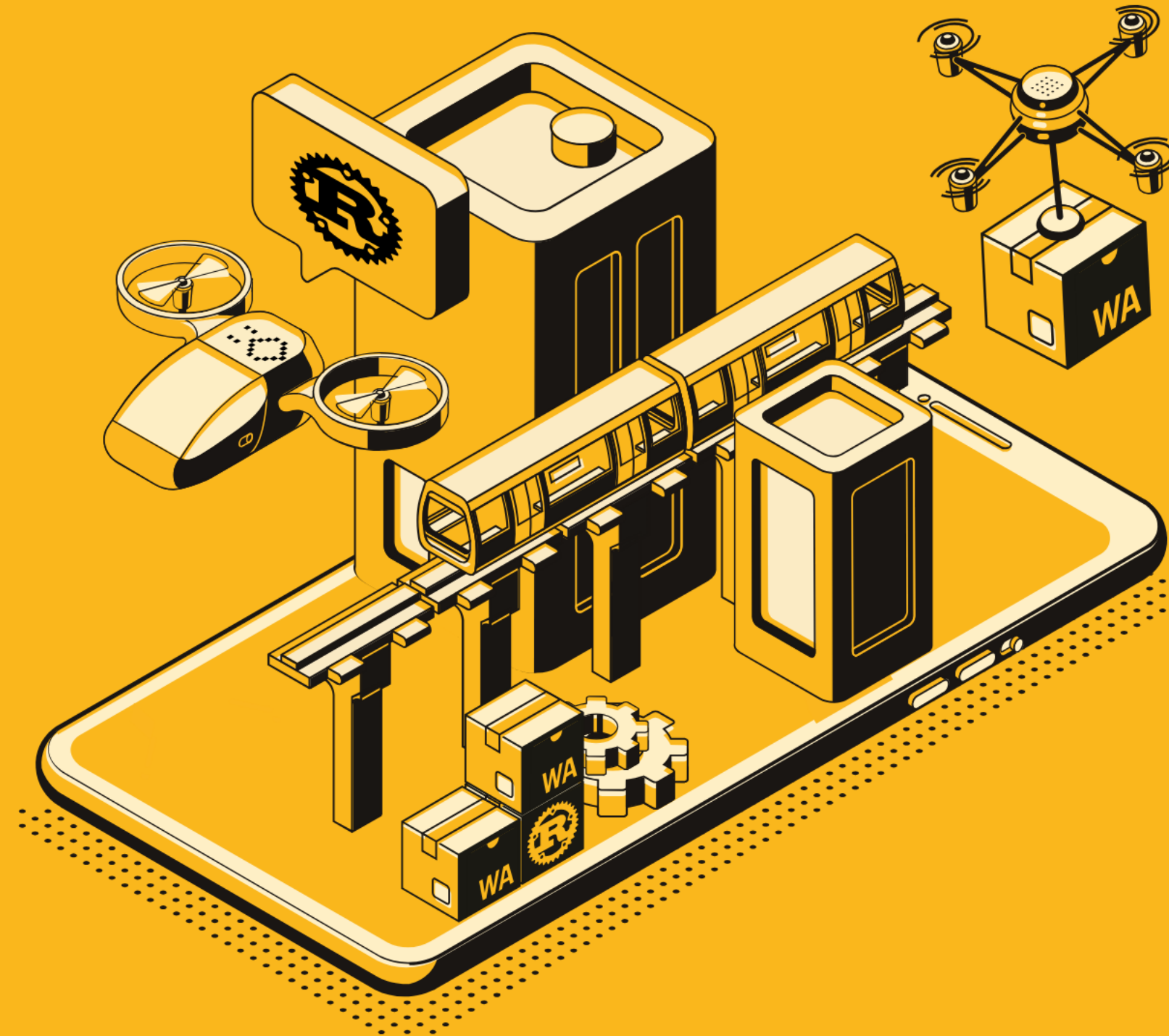
Careful what
you wish for...



Business



**Friendly,
professional
Rust
Consulting**



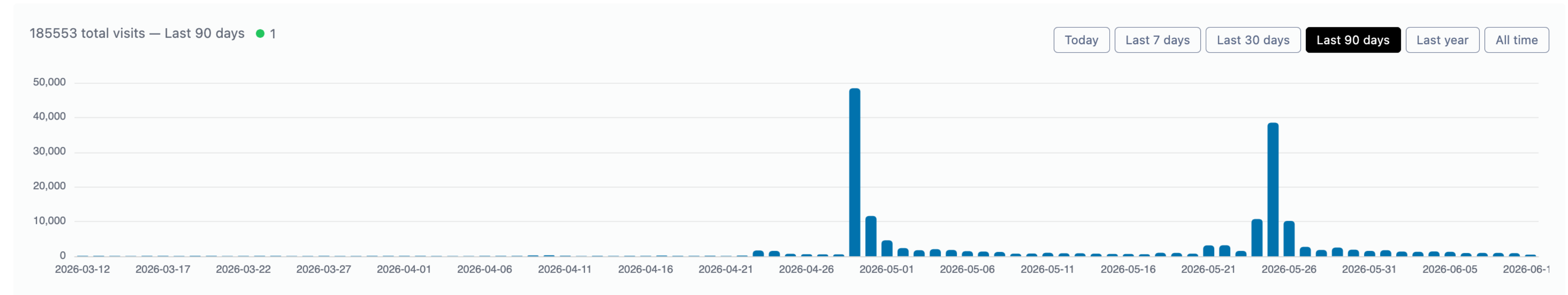
TRUSTED BY ENGINEERING TEAMS WORLDWIDE

Helping companies ship production Rust — on time and with confidence



Dashboard

Tracking <https://corrode.dev>



Pages

PATH	VISITS	VISITS/DAY	AVG. DURATION	SHARE
/blog/bugs-rust-wont-catch	67187	746.5	17min 11s	36.2
● /learn/migration-guides/go-to-rust	64252	713.9	16min 07s	34.6
/	7985	88.7	2min 14s	4.3
/podcast/s06e02-helsing	4988	55.4	14min 47s	2.7
/podcast/s06e04-rust4linux	4344	48.3	12min 58s	2.3
/blog	3338	37.1	4min 24s	1.8
/blog/defensive-programming	3190	35.4	26min 47s	1.7
/podcast	1934	21.5	4min 40s	1.0
/blog/async	1858	20.6	18min 52s	1.0
/blog/tips-for-faster-rust-compile-times	1844	20.5	28min 59s	1.0
/blog/rust-learning-resources-2026	1769	19.7	21min 56s	1.0
/blog/idiomatic-rust-resources	1765	19.6	16min 01s	1.0
/learn	1523	16.9	7min 02s	0.8
/blog/flattening-rusts-learning-curve	1384	15.4	27min 39s	0.7
/services	1298	14.4	3min 17s	0.7
/learn/migration-guides/rust-to-rust	1114	12.2	14min 08s	0.5

Start with a stack **All tools**

Collapse all

Jump to

Category...

Sort

Relevance

License

All licenses

MSRV





Any MSRV

Hide deprecated

 Staff picks only

 **Toolchain** 3

Installing, switching, and managing Rust toolchains and components.

TOOL	NOTES	METRICS
<p>rustup</p> <p> rust-lang/rustup</p> <p>Apache-2.0</p> <p> Staff pick Essentials</p>	<p>The official Rust toolchain installer and version manager. Installs and switches between stable, beta, and nightly toolchains, manages components (clippy, rustfmt, rust-src) and cross-compilation targets, and pins a per-project toolchain via <code>rust-toolchain.toml</code>.</p> <p>Effectively mandatory: almost every other tool here assumes a rustup-managed toolchain. The only time you'd skip it is a fully pinned distro/Nix or CI image that provisions toolchains some other way.</p> <p>ALTERNATIVES <code>rustup-toolchain-install-master</code> <code>Nix (oxalica/rust-overlay)</code></p>	<p>↓ — downloads</p> <p>☆ 6.9k stars</p> <p>🕒 1w ago updated</p> <p>Maintained</p>
<p>cargo-binstall</p> <p> cargo-bins/cargo-binstall</p> <p><code>install</code></p> <p>v1.19.1 · GPL-3.0-only</p> <p>by Félix Saparelli, ryan, Jiahao XU, Binstall</p> <p>Essentials</p>	<p>Installs Rust binaries from pre-built GitHub release artifacts instead of compiling from source, turning a multi-minute <code>cargo install</code> into a download. Falls back to building from source when no prebuilt binary is published.</p> <p>Reach for it in CI and on dev machines to install tools fast. Skip it if you need a build from source for provenance/audit reasons, or for crates that ship no release binaries (where it gains you nothing over <code>cargo install</code>).</p> <p>ALTERNATIVES <code>cargo-install</code> <code>cargo-quickinstall</code></p>	<p>↓ 100.4k downloads</p> <p>☆ 2.7k stars</p> <p>🕒 1w ago updated</p> <p>Maintained</p>
<p>cargo-quickinstall</p> <p> cargo-bins/cargo-quickinstall</p> <p><code>install</code></p>	<p>Installs Rust binaries from precompiled artifacts instead of building from source. It fetches prebuilt packages from a community build service, falling back to <code>cargo install</code> when no prebuilt binary exists for your target.</p> <p>Useful for skipping long compile times in CI and local setup. It is also used as a backend by <code>cargo-binstall</code>, which adds discovery of binaries published directly on a project's own releases.</p> <p>ALTERNATIVES <code>cargo-binstall</code></p>	<p>↓ — downloads</p> <p>☆ — stars</p> <p>🕒 — updated</p> <p>Maintained</p>

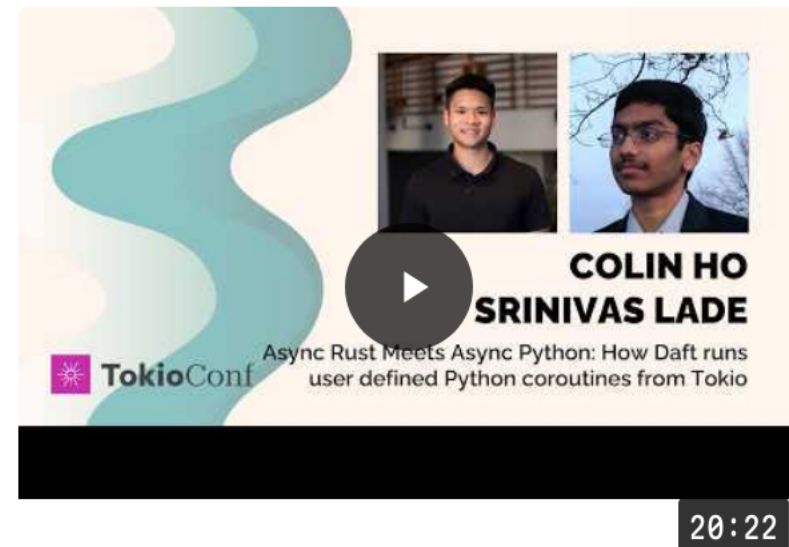
Rust Search

Find the best content on Rust, curated by the community; a search engine for Rustaceans.

 ✕ 🔍

[ALL](#) [ARTICLES](#) [PODCASTS](#) [VIDEOS](#)

ANY TIME ▾



Async Rust Meets Async Python: How Daft runs...

Async Rust Meets Async Python: How Daft runs user defined Python coroutines from Tokio by Colin Ho, Srinivas Lade Daft is a data engine for multimodal and AI workloads, letting users build pipelines that process text, images, embeddings, and model calls in a single Python API. A common pattern is...

TokioConf | Talk | 2026-04-21

[Video](#)



RustConf 2022 - ASYNC RUST: PAST, PRESENT, AND FUTURE GENERAL by Nick Cameron

RustConf 2022 - ASYNC RUST: PAST, PRESENT, AND FUTURE GENERAL by Nick Cameron Today, `async` and `await` are stable parts of the language, but you can't actually run `async` code without a third party runtime. Unlike most languages, an `async` function has to be awaited before any work gets done...

 www.youtube.com | Video | 2022-09-05



Async Rust: Portability and Interoperability



Async Rust: Portability and Interoperability - Nick Cameron - Rust Linz, April 2022

Async Rust: Portability and Interoperability - Nick Cameron - Rust Linz, April 2022 Async Rust: Portability and Interoperability I'll talk about the portability and interoperability

 Customize

Humans

Hackathons





Open Source

Because Code Quality Matters

We list **the best static analysis tools and linters** that can help you improve code quality. All tools are peer-reviewed by fellow engineers. Avoid bugs in production, outages, and angry customers.

Show Tools

Source Code



Hero image by upklyak on Freepik

Latest from our Blog

Introducing Pixee as Our Sponsor: Automated Security and Code Quality with Pixeebot

25/05/2024

As developers, maintaining secure, high-quality code is challenging. Security vulnerabilities, performance...

[Read more](#)

A Closer Look At BugProve

19/11/2023

I have never been a huge fan of IoT devices. Granted, they make our life easier, but they also open the door t...

[Read more](#)

Welcoming Bearer as Our Sponsor: Simple Ruby and JavaScript App Security

06/04/2023

As a developer, I have faced my fair share of security mishaps. I recall times when I accidentally exposed...

[Read more](#)

Redesigning Analysis Tools

29/03/2023

We are happy to announce that we completely rebuilt analysis-tools.dev

Popular Static/Dynamic Analysis Tools by Language

Show all (18)

Typescript Static Analysis Tools

Most Popular Typescript Linters

- 22 67% upvoted TypeScript Call Graph
- 10 78% upvoted TypeScript ESLint
- 7 63% upvoted zod

Most Popular Typescript Formatters

Help make this list better
[Suggest Tools](#)

Rust Static Analysis Tools

Most Popular Rust Linters

- 95 93% upvoted clippy
- 35 89% upvoted cargo-audit
- 12 93% upvoted MIRAI

Most Popular Rust Formatters

- 40 93% upvoted rust-analyzer
- 25 96% upvoted rustfmt
- 6 64% upvoted Rust Language Server

Sql Static Analysis Tools

Most Popular Sql Linters

Most Popular Sql Formatters

Filters is:pr is:open | Labels 17 | Milestones 0 | New pull request

<input type="checkbox"/>	41 Open ✓ 1,184 Closed	Author	Label	Projects	Milestones	Reviews	Assignee	Sort
<input type="checkbox"/>	Add adocfmt ✓ #1839 opened 5 days ago by dheid (First-time contributor)							
<input type="checkbox"/>	Add BreakShield CI — AST-based breaking change detection for APIs #1838 opened last week by vojtsprime11 (First-time contributor)							
<input type="checkbox"/>	Add proof-of-commitment supply chain risk scorer ✓ #1837 opened last week by piiico (First-time contributor)							
<input type="checkbox"/>	Add ansible-security-scanner ✓ #1833 opened 2 weeks ago by cpeoples (First-time contributor)							
<input type="checkbox"/>	Add CodeGuard to Python static analysis tools ✓ #1832 opened 2 weeks ago by mohameden19961 (First-time contributor)							
<input type="checkbox"/>	Add pgrls (Postgres Row-Level Security static analyzer) ✓ #1829 opened 3 weeks ago by dmitrymaranik (First-time contributor)							
<input type="checkbox"/>	add go-mutesting ✓ #1828 opened 3 weeks ago by jonbaldie (First-time contributor)							
<input type="checkbox"/>	Add vastlint to Multiple languages ✗ #1825 opened on May 11 by aleksUIX (First-time contributor)							
<input type="checkbox"/>	Add GauntletCI to static analysis tools list ✗ #1824 opened on May 9 by EricCogen (First-time contributor)							
<input type="checkbox"/>	Add zizmor static analysis tool ✓ #1821 opened on May 6 by musaabhasan (First-time contributor)							
<input type="checkbox"/>	feat: add nyx linter tool with multi-language support ✓ #1820 opened on May 6 by elicpeter (First-time contributor) 5 tasks done							
<input type="checkbox"/>	Add sem and weave: entity-level code analysis and merge tools ✓ #1818 opened on Apr 25 by rs545837 (First-time contributor)							

Filters is:pr is:open Labels 17 Milestones 0 New pull request

<input type="checkbox"/>	41 Open	1,184 Closed	Author	Label	Projects	Milestones	Reviews	Assignee	Sort
<input type="checkbox"/>	Add adocfmt ✓		#1839 opened 5 days ago by dheid (First-time contributor)						
<input type="checkbox"/>	Add BreakShield CI — AST-based breaking change detection for APIs		#1838 opened last week by vojtsprime11 (First-time contributor)						
<input type="checkbox"/>	Add proof-of-commitment supply chain risk scorer ✓		#1837 opened last week by plilico (First-time contributor)						
<input type="checkbox"/>	Add ansible-security-scanner ✓		#1833 opened 2 weeks ago by cpeoples (First-time contributor)						
<input type="checkbox"/>	Add CodeGuard to Python static analysis tools ✓		#1832 opened 2 weeks ago by mohameden19961 (First-time contributor)						
<input type="checkbox"/>	Add pgrls (Postgres Row-Level Security static analyzer) ✓		#1829 opened 3 weeks ago by dmitrymaranik (First-time contributor)						
<input type="checkbox"/>	add go-mutesting ✓		#1828 opened 3 weeks ago by jonbaldie (First-time contributor)						
<input type="checkbox"/>	Add vastlint to Multiple languages ✗		#1825 opened on May 11 by aleksUIX (First-time contributor)						
<input type="checkbox"/>	Add GauntletCI to static analysis tools list ✗		#1824 opened on May 9 by EricCogen (First-time contributor)						
<input type="checkbox"/>	Add zizmor static analysis tool ✓		#1821 opened on May 6 by musaabhasan (First-time contributor)						
<input type="checkbox"/>	feat: add nyx linter tool with multi-language support ✓		#1820 opened on May 6 by elicpeter (First-time contributor) 5 tasks done						
<input type="checkbox"/>	Add sem and weave: entity-level code analysis and merge tools ✓		#1818 opened on Apr 25 by rs545837 (First-time contributor)						
<input type="checkbox"/>	Add fallow ✓		#1814 opened on Apr 23 by BartWaardenburg (First-time contributor)						
<input type="checkbox"/>	Add SecLens benchmark ✓		#1811 opened on Apr 16 by subho007 (First-time contributor)						
<input type="checkbox"/>	Add repowise ✓		#1809 opened on Apr 14 by RaghavChamadiya (First-time contributor)						
<input type="checkbox"/>	Add malware-check: static/dynamic malware analysis tool ✓		#1806 opened on Apr 13 by momenbase1 (First-time contributor) 5 tasks done						2
<input type="checkbox"/>	Add CodeSafer — AI code security scanner ✓		#1804 opened on Apr 11 by goldmembrane (First-time contributor) 4 tasks done						1
<input type="checkbox"/>	Add ffmt - fast Fortran formatter written in Rust ✓		#1802 opened on Apr 11 by sbryngelson (Contributor)						2
<input type="checkbox"/>	Add VulnHawk - AI-powered code security scanner ✓		#1801 opened on Apr 8 by momenbase1 (First-time contributor) 6 tasks done						2
<input type="checkbox"/>	Add Inkog — AI agent security scanner ✓		#1800 opened on Apr 8 by cloakmaster (First-time contributor)						1
<input type="checkbox"/>	Add nestjs-doctor - static analysis tool for NestJS ✓		#1799 opened on Apr 8 by Franciscuo (First-time contributor) 1 task done						
<input type="checkbox"/>	Add cognium — semantic taint-tracking SAST for Java, JS, TS, Python, Rust, Bash		#1795 opened on Apr 3 by cogglyadmin (First-time contributor)						
<input type="checkbox"/>	Add Git AutoReview — AI code review for VS Code ✓		#1792 opened on Apr 3 by vitalli4reva (First-time contributor) 6 tasks done						2
<input type="checkbox"/>	Add HefestoAI — AI-generated code quality guardian (21 formats) ✓		#1785 opened on Mar 31 by artvepa80 (First-time contributor)						8
<input type="checkbox"/>	Add sloppy-joe — multi-ecosystem dependency supply chain security ✓		#1784 opened on Mar 29 by brennhill (First-time contributor)						

< Previous 1 2 Next >

ProTip! Click a checkbox on the left to edit multiple issues at once.

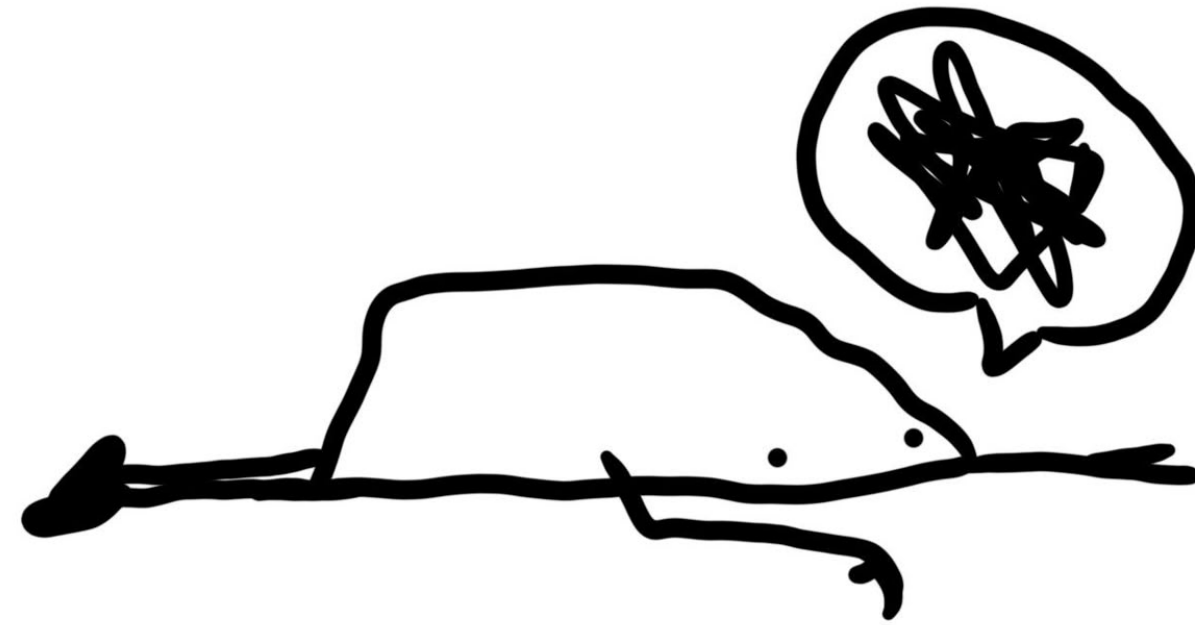
Filters is:pr is:open Labels 17 Milestones 0 New pull request

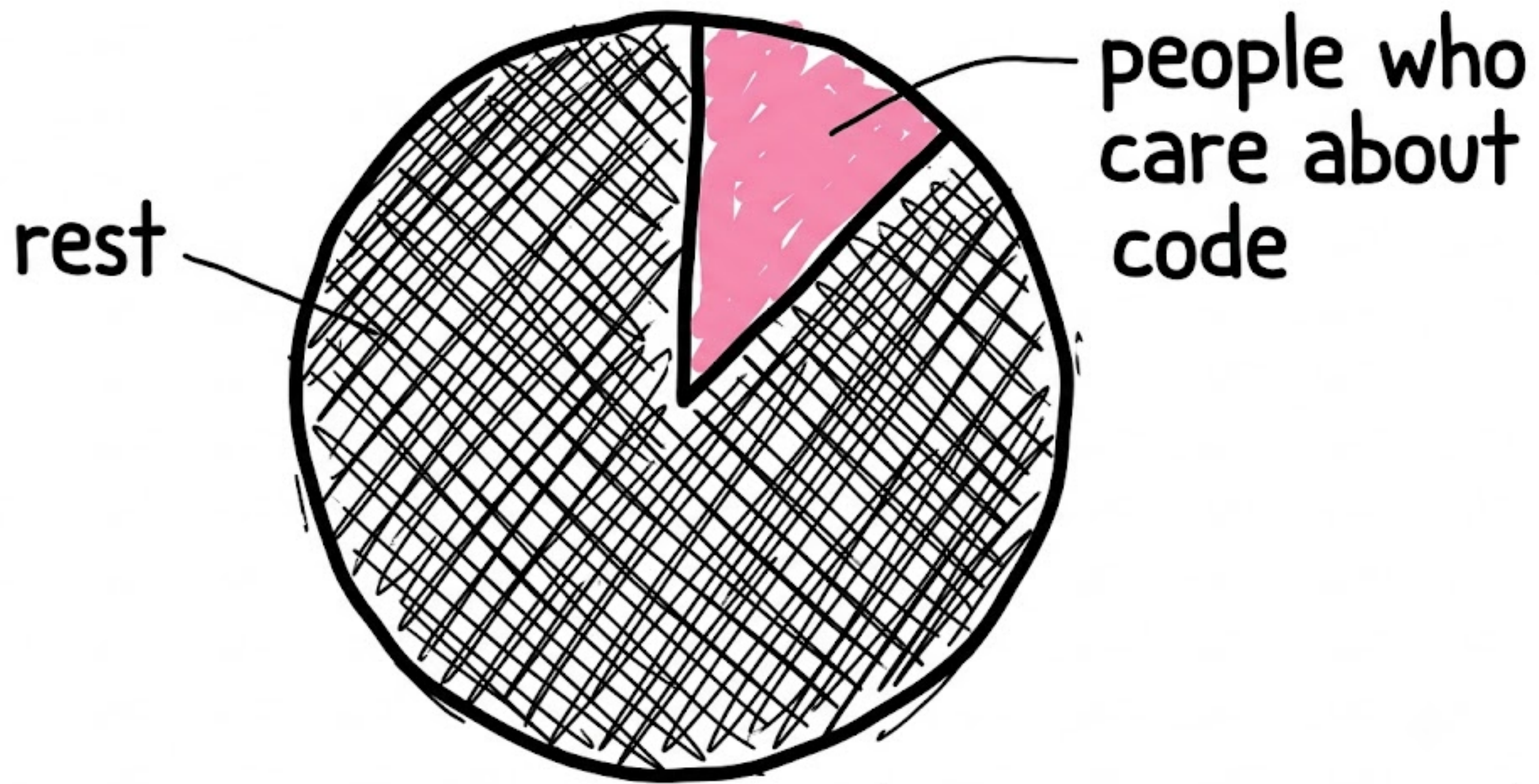
<input type="checkbox"/>	41 Open	1,184 Closed	Author	Label	Projects	Milestones	Reviews	Assignee
<input type="checkbox"/>	Add pyscn to Python tools ✓		#1780 opened on Mar 24 by DaisukeYoda (First-time contributor) 6 tasks done					
<input type="checkbox"/>	Add ClawSec - Security analysis for AI agent skills ✓		#1778 opened on Mar 22 by huacheng (First-time contributor)					
<input type="checkbox"/>	Add LucidShark - unified code quality tool for AI agents ✓		#1773 opened on Mar 19 by toniantunovi (First-time contributor)					
<input type="checkbox"/>	Add Reporecall — multi-language code indexer with call graphs ✗		#1772 opened on Mar 19 by nillo (First-time contributor)					
<input type="checkbox"/>	Add CodeHawk — AI GitHub PR review bot ✓		#1771 opened on Mar 18 by CrossGen-ai (First-time contributor)					
<input type="checkbox"/>	Add Open Code Review to Multiple languages section ✗		#1770 opened on Mar 15 by raye-deng (First-time contributor)					
<input type="checkbox"/>	Add codesize - file and function size linter with tree-sitter		#1768 opened on Mar 14 by ChrisGVE (First-time contributor)					
<input type="checkbox"/>	Add SlowQL - SQL static analyzer ✓		#1766 opened on Mar 10 by makroumi (First-time contributor) 1 task done					
<input type="checkbox"/>	Add g0 — AI agent security analysis tool ✗		#1765 opened on Mar 9 by JBAhire (First-time contributor)					
<input type="checkbox"/>	Add Sentinel AI ✗		#1764 opened on Mar 8 by MaxwellCalkin (First-time contributor)					
<input type="checkbox"/>	Add NAAb — polyglot language with LLM governance engine ✓		#1761 opened on Mar 5 by b-macker (First-time contributor)					
<input type="checkbox"/>	Add config-guard — MCP security linter ✓		#1756 opened on Mar 2 by KGT24k (First-time contributor)					
<input type="checkbox"/>	Add cursor-doctor - Cursor AI IDE config linter ✓		#1751 opened on Feb 27 by nedcodes-ok (First-time contributor)					
<input type="checkbox"/>	Add aikido: static security analyzer for Aiken smart contracts ✗		#1750 opened on Feb 26 by Bajuzjefe (First-time contributor)					
<input type="checkbox"/>	Add prodlint — production readiness for AI-generated JS/TS apps ✓		#1744 opened on Feb 20 by Anthony-Marcovecchio (First-time contributor)					
<input type="checkbox"/>	Add rma-cli - Fast SAST scanner with Semgrep compatibility		#1736 opened on Feb 5 by bumahkib7 (First-time contributor)					

< Previous 1 2 Next >

ProTip! Exclude your own issues with `-author:mre`.







Pair programming

RUN ▶ ... DEBUG ▾ STABLE ▾ ...

SHARE TOOLS ▾ CONFIG ▾ ?

```
1 fn main() {
2     println!("{code:?}", code = HttpStatus::NotFound);
3     let code1 = HttpStatus::Ok;
4     let code2 = HttpStatus::Ok;
5     let is_identical = code1 == code2;
6     println!("{is_identical}");
7 }
8
9 #[derive(Debug, PartialEq)]
10 enum HttpStatus {
11     Ok,
12     NotFound,
13     InternalServerError,
14 }
15
```

Execution

Format

A Beginner's Guide To Rust

or, "wrangling with the borrow checker until one of you gives up"

Glad you're here. You're going to write Rust in a browser tab. No setup, no project structure to wrangle, just small problems that build into real fluency. Each chapter has a short note, a working code editor, and tests that turn green when your code is right. Most take five to ten minutes.

Along the way you'll meet **ownership and borrowing**, **pattern matching**, `Option` and `Result`, **iterators**, **structs and methods**, **modules**, and four hands-on mini-projects: a password validator, a word counter, an env-file parser, and a CSV parser. Skip ahead if a chapter feels familiar; come back when something clicks.

Let's get acquainted

Here's your first piece of Rust. Hit **Run** below to send it off to the compiler — if everything's wired up, you'll see `Hello, Rust!` appear underneath. Then we'll sign you up and start chapter 1.

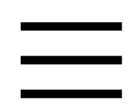
HELLO.RS

▶ Run

```
1 /// Prints a friendly greeting.
2 fn hello() {
3
4 }
5
6 fn main() {
7     hello();
8 }
```

```
>>> lists = [[]] * 3
>>> lists
[[], [], []]
>>> lists[0].append(3)
```

```
>>> lists = [[]] * 3
>>> lists
[[], [], []]
>>> lists[0].append(3)
>>> lists
[[3], [3], [3]]
```



3.14.6



Quick search

Go

drops below zero).

Notes:

1. While the `in` and `not in` operations are used only for simple containment testing in the general case, some specialised sequences (such as [str](#), [bytes](#) and [bytearray](#)) also use them for subsequence testing:

```
>>> "gg" in "eggs"  
True
```

2. Values of `n` less than `0` are treated as `0` (which yields an empty sequence of the same type as `s`). Note that items in the sequence `s` are not copied; they are referenced multiple times. This often haunts new Python programmers; consider:

```
>>> lists = [[]] * 3  
>>> lists  
[[], [], []]  
>>> lists[0].append(3)  
>>> lists  
[[3], [3], [3]]
```

What has happened is that `[[]]` is a one-element list containing an empty list, so all three elements of `[[]] * 3` are references to this single empty list. Modifying any of the elements of `lists` modifies this single list. You can create a list of different lists this way:

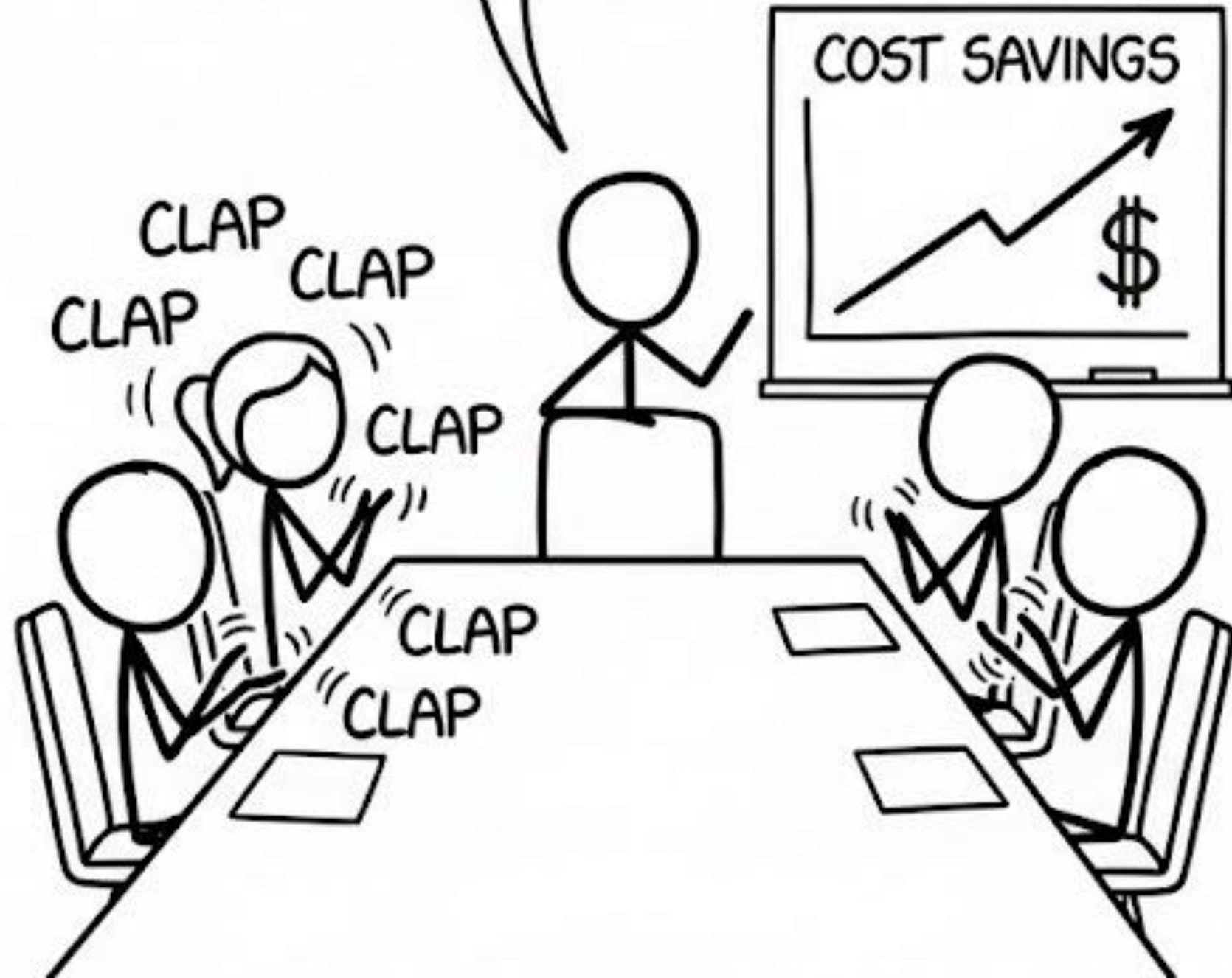
```
>>> lists = [[] for i in range(3)]  
>>> lists[0].append(3)  
>>> lists[1].append(5)  
>>> lists[2].append(7)  
>>> lists  
[[3], [5], [7]]
```

Further explanation is available in the FAQ entry [How do I create a multidimensional list?](#).

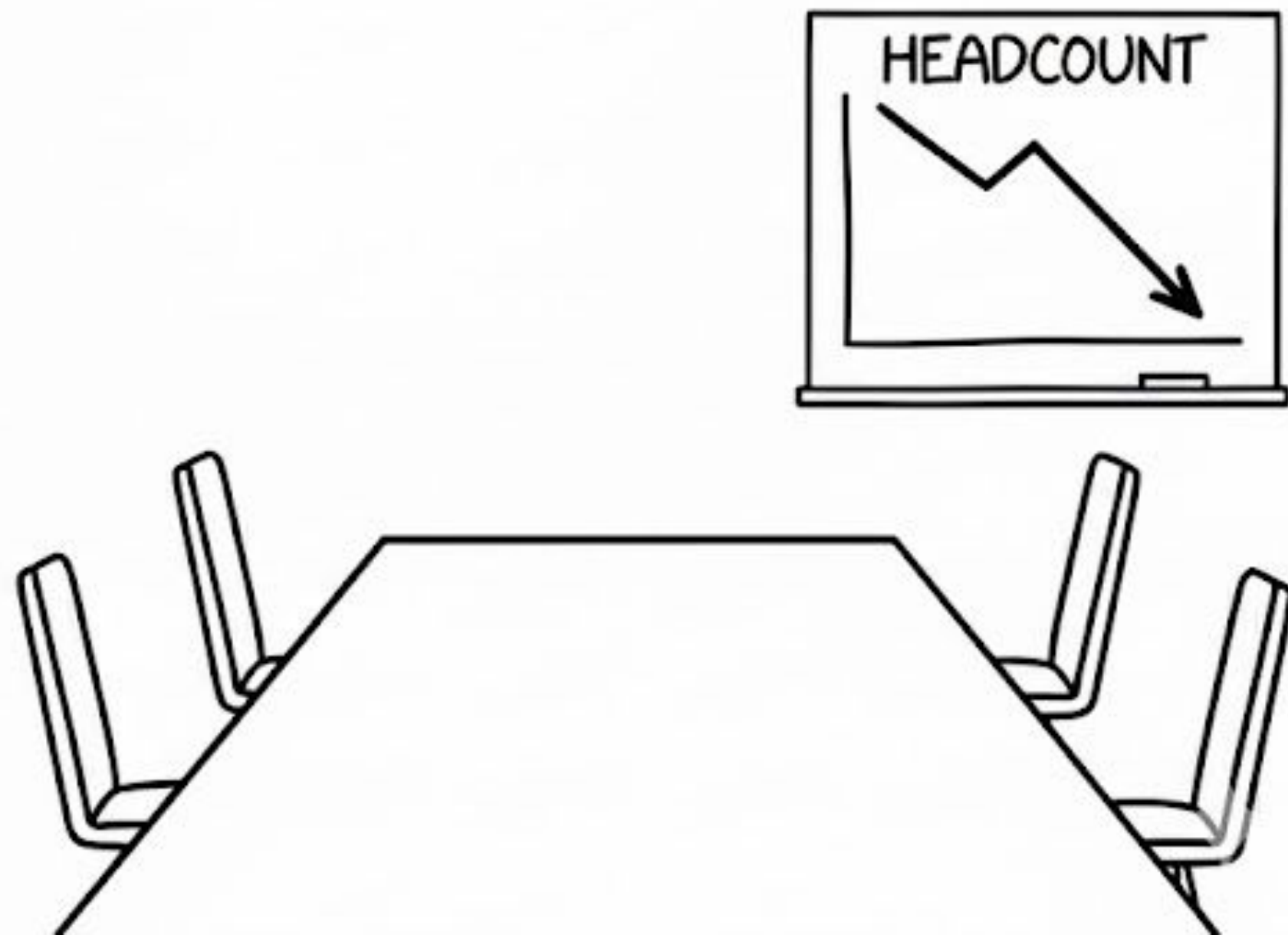
3. If `i` or `j` is negative, the index is relative to the end of sequence `s`: `len(s) + i` or `len(s) + j` is substituted. But note that `-0` is still `0`.

Juniors

Starting today, we're no longer hiring junior developers. AI will handle all entry-level coding tasks.

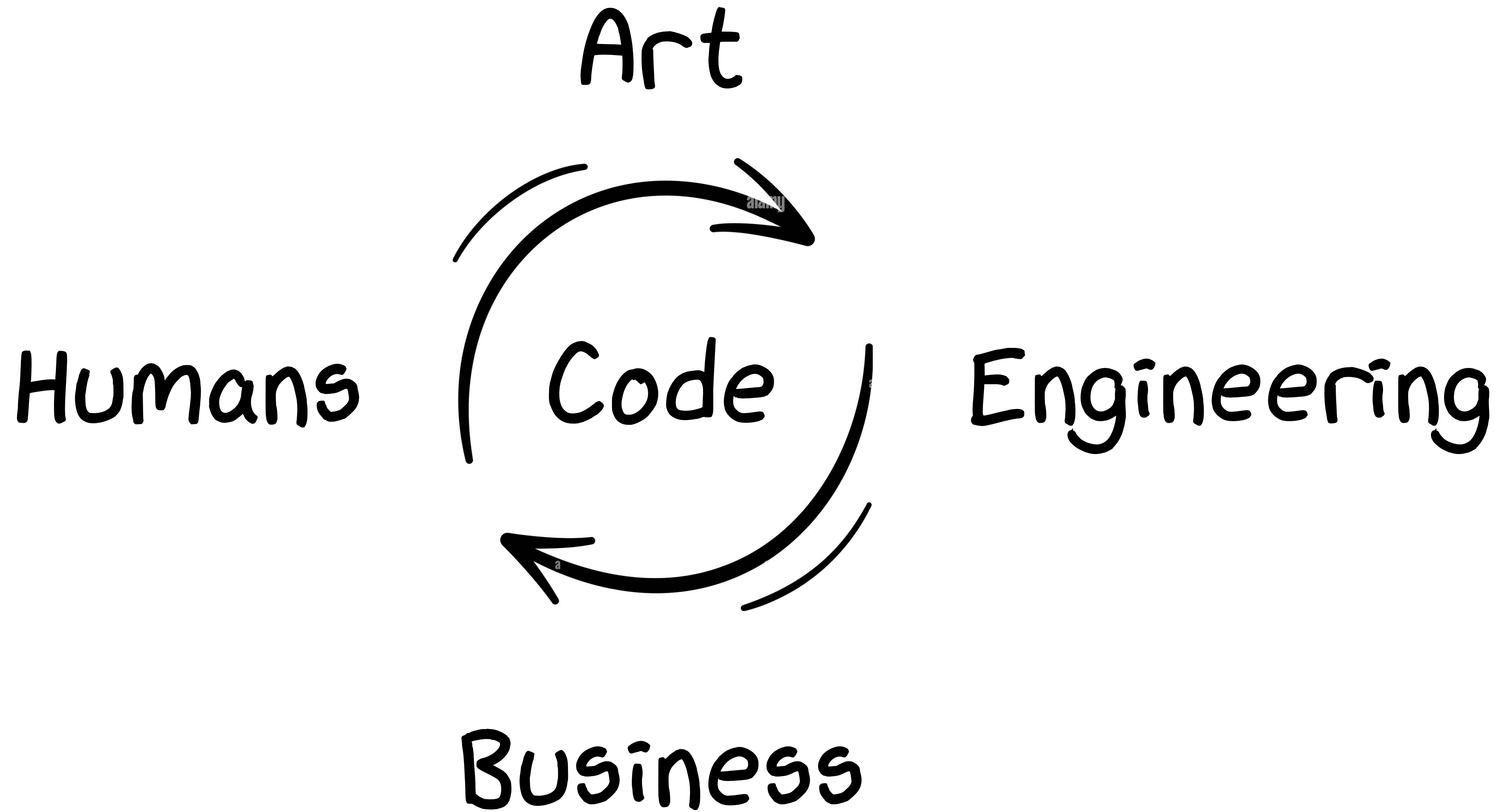


5 years later...



Summary





Art

- Write code by hand sometimes.
- Have fun!
- Write a game?
- Don't stop exploring.
- Express yourself
- It's okay to fail.

Engineering

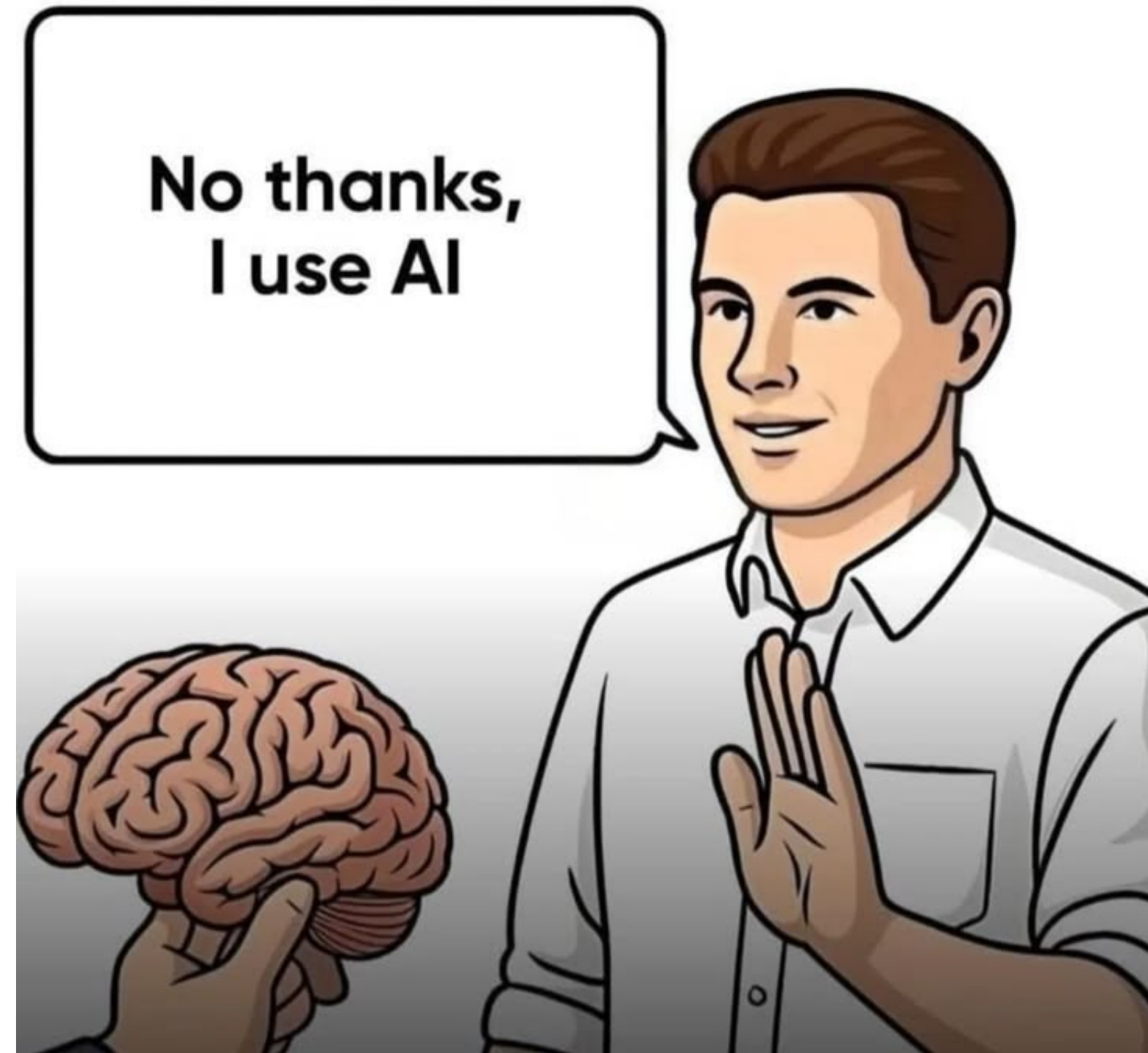
- Take time to think deeply
- Go slowly
- Read the code of other people
- Don't stop reviewing code
- Simple solutions are best
- Refactor
- Document
- Read the reference
- Care about your craft

Business

- Automation is good
- Embrace AI, aim higher
- You constantly have to adapt
- Find new areas of business
- Be a problem solver

Humans

- Write good documentation for other humans
- Write text for yourself and others
- Teach others
- Have empathy
- Do more things together
- Go to local events



Don't switch off your brain just now

Anything worthwhile takes a long time

People > profits, code, productivity

Half our life is spent trying to find something to do
with the time we have rushed through life trying to save

