

RUST



SEASON 1 | EPISODE 1

The FOSS Files



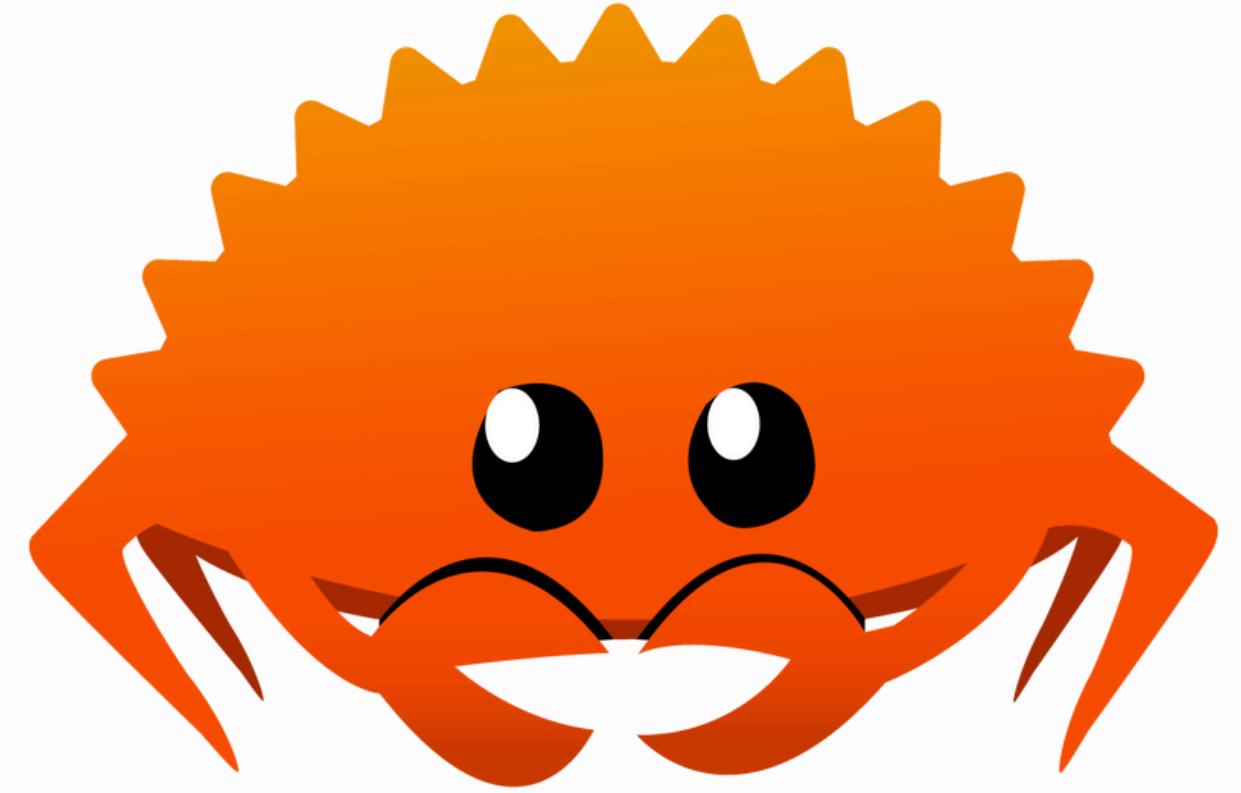
@Walchand Linux Users' Group



Rust

**THE PROGRAMMING LANGUAGE THAT EMPOWERS
EVERYONE TO BECOME A SYSTEMS
PROGRAMMER**

× × × ×



Rustaceans

The Rust Programmers

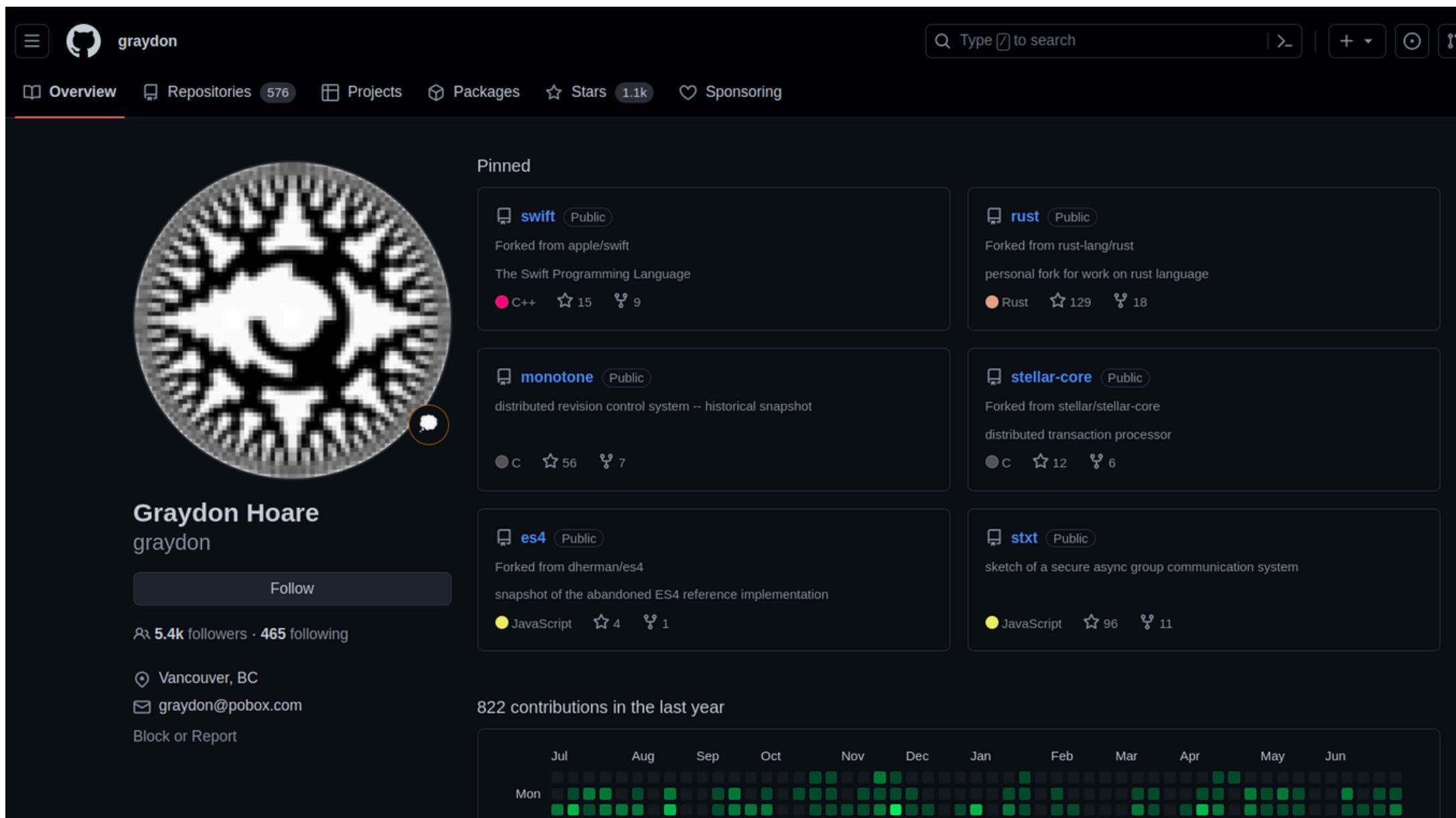


The Origin



Started as a side project

<https://github.com/graydon/rust>



A screenshot of the GitHub profile for the user 'graydon'. The profile features a large circular profile picture of a fractal-like tree or flame pattern. The user's name, 'Graydon Hoare', and GitHub handle, 'graydon', are displayed below the picture. A 'Follow' button is present. The profile shows 5.4k followers and 465 following. The location is listed as 'Vancouver, BC' and the email as 'graydon@pobox.com'. There is a 'Block or Report' link. The 'Overview' tab is selected, showing 576 repositories, 0 projects, 0 packages, 1.1k stars, and 0 Sponsoring. The 'Pinned' section contains five repositories: 'swift' (forked from apple/swift), 'rust' (forked from rust-lang/rust), 'monotone' (distributed revision control system -- historical snapshot), 'stellar-core' (forked from stellar/stellar-core), and 'es4' (forked from dherman/es4). Below the pinned section, it says '822 contributions in the last year' with a timeline bar showing activity from July to June.





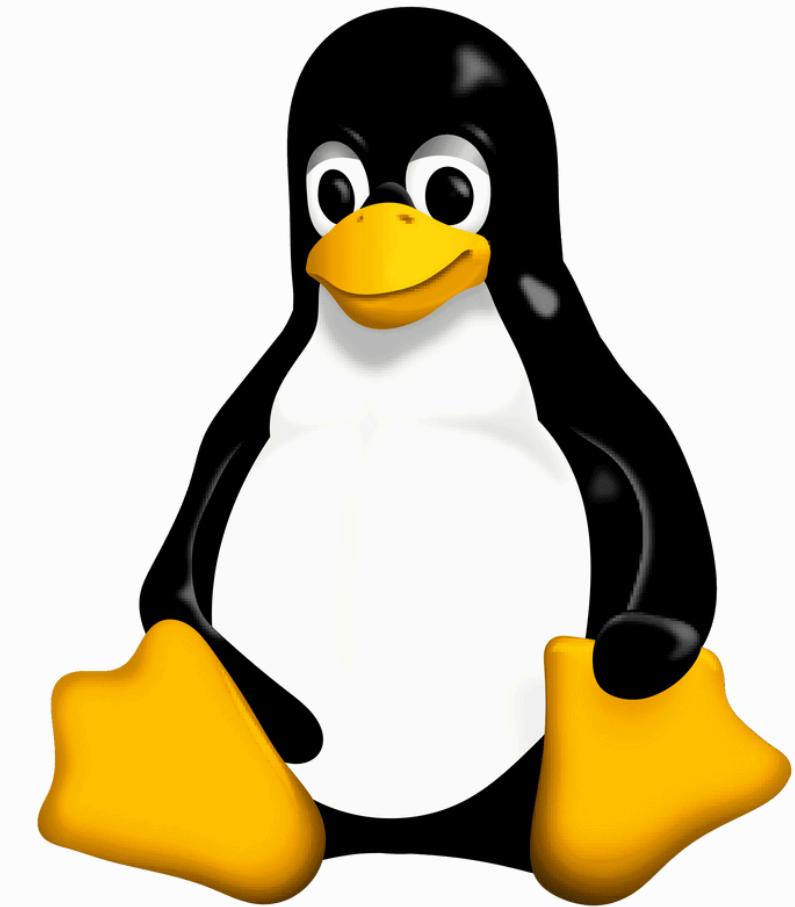
moz://a

SPONSORED IN 2009

Named after the Rust Fungus



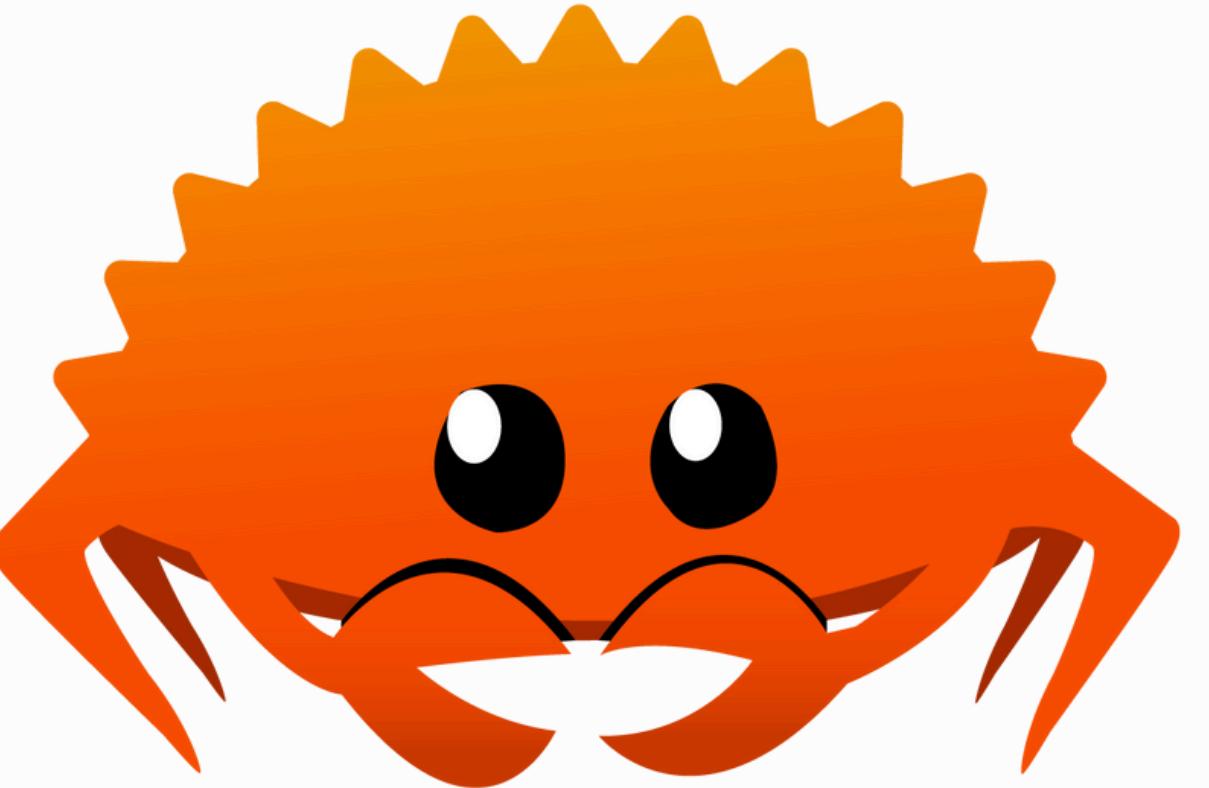
An experimental language for Linux



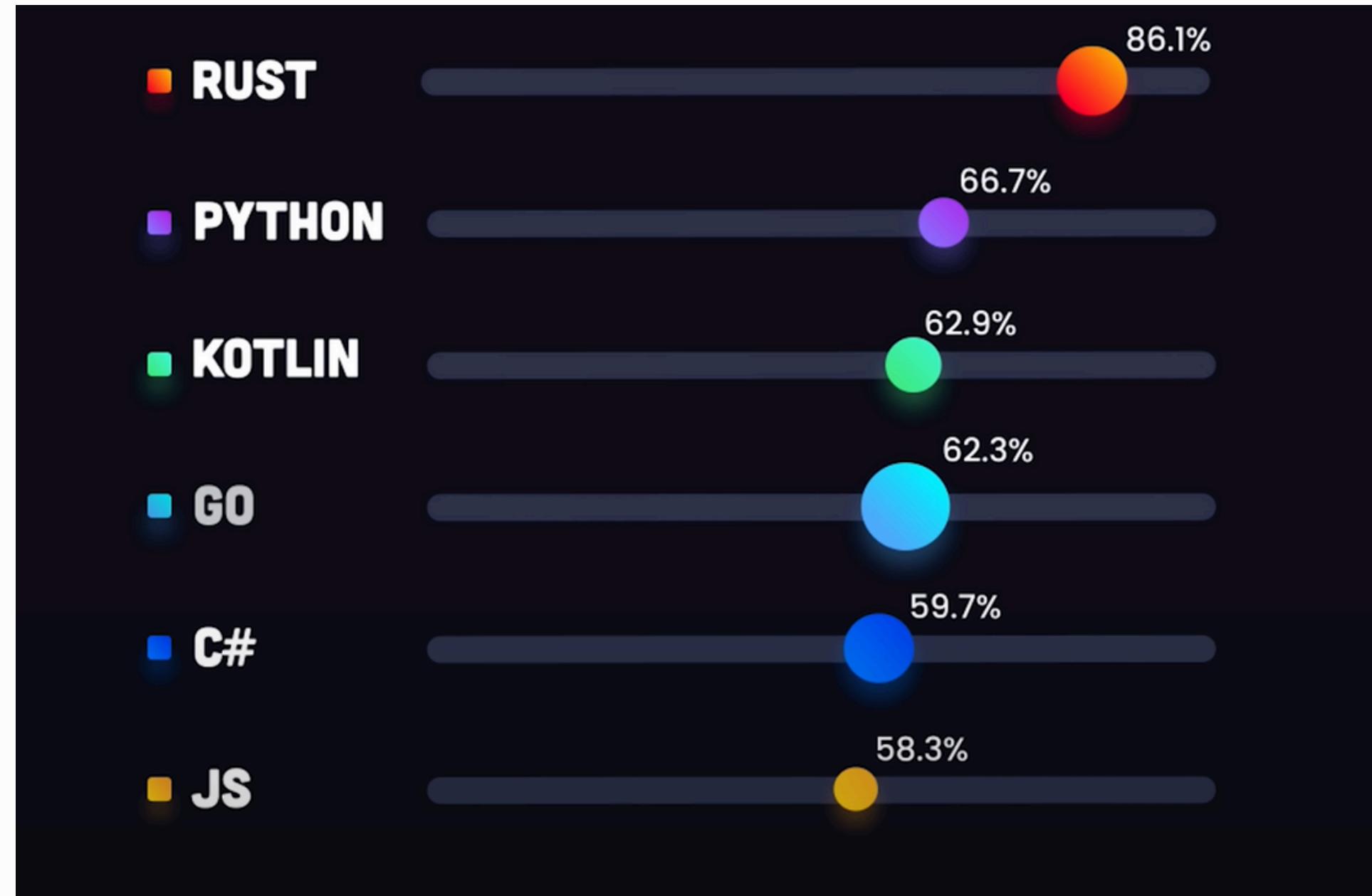
Popularity

StackOverflow

Most loved programming language since 2016



<https://survey.stackoverflow.co/2022/#most-loved-dreaded-and-wanted-language-love-dread>



Why Rust?

Provides high-level simplicity with low-level performance

**Memory
safety
without
garbage
collection**

**Concurrency
without
race
conditions**

**Performance
without
compromising
safety**

Garbage Collection



```
#include <iostream>
using namespace std;
int main() {
    // Dynamically allocate memory for an integer
    int* ptr = new int;
    *ptr = 10;

    // Perform some task

    // Deallocate the memory to avoid memory leaks
    delete ptr;

    return 0;
}
```



Features



No null pointer references

+

Zero cost abstractions

01



C++

```
● ● ●  
  
int main(){
    vector<int> v={1,2,3,4,5};
    auto maxi=max_element(v.begin(),v.end());
    cout<<"Max Element"<<*maxi<<"\n";
    return 0;
}
```

RUST

```
● ● ●  
  
fn main(){
    let numbers=vec![1,2,3,4,5];
    let maxi=numbers.iter().max();
    match maxi{
        Some(&max)=>println!("max element:{}",&max),
        None=>println!("The vector is empty"),
    }
}
```

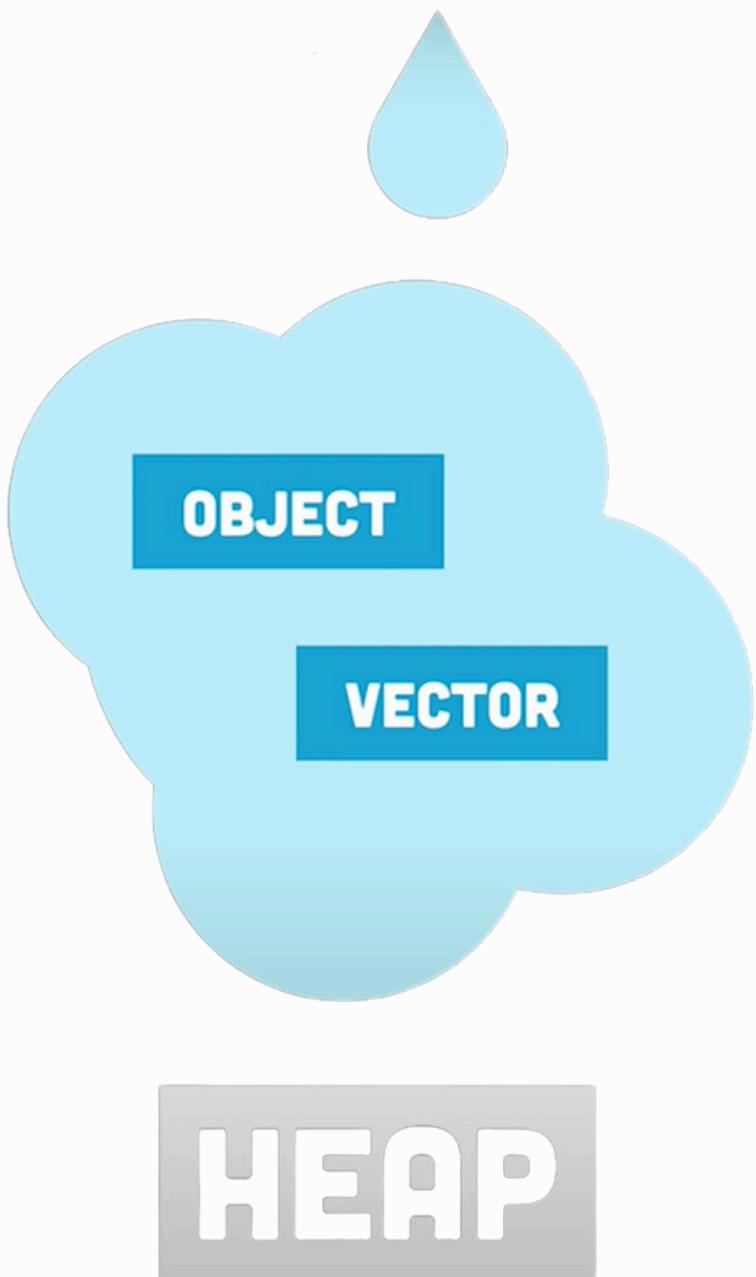


Rust OwnerShip, Borrowing and Lifetime (for the absence of garbage collection)

02



Mutability



```
let mut hello = "hi mom";
```

Rust Module System

03



Rust Thread Safety approach

04



Some awesome Rust Projects



RSTY



× × × ×



Rocket - Backend end points

```
● ● ●

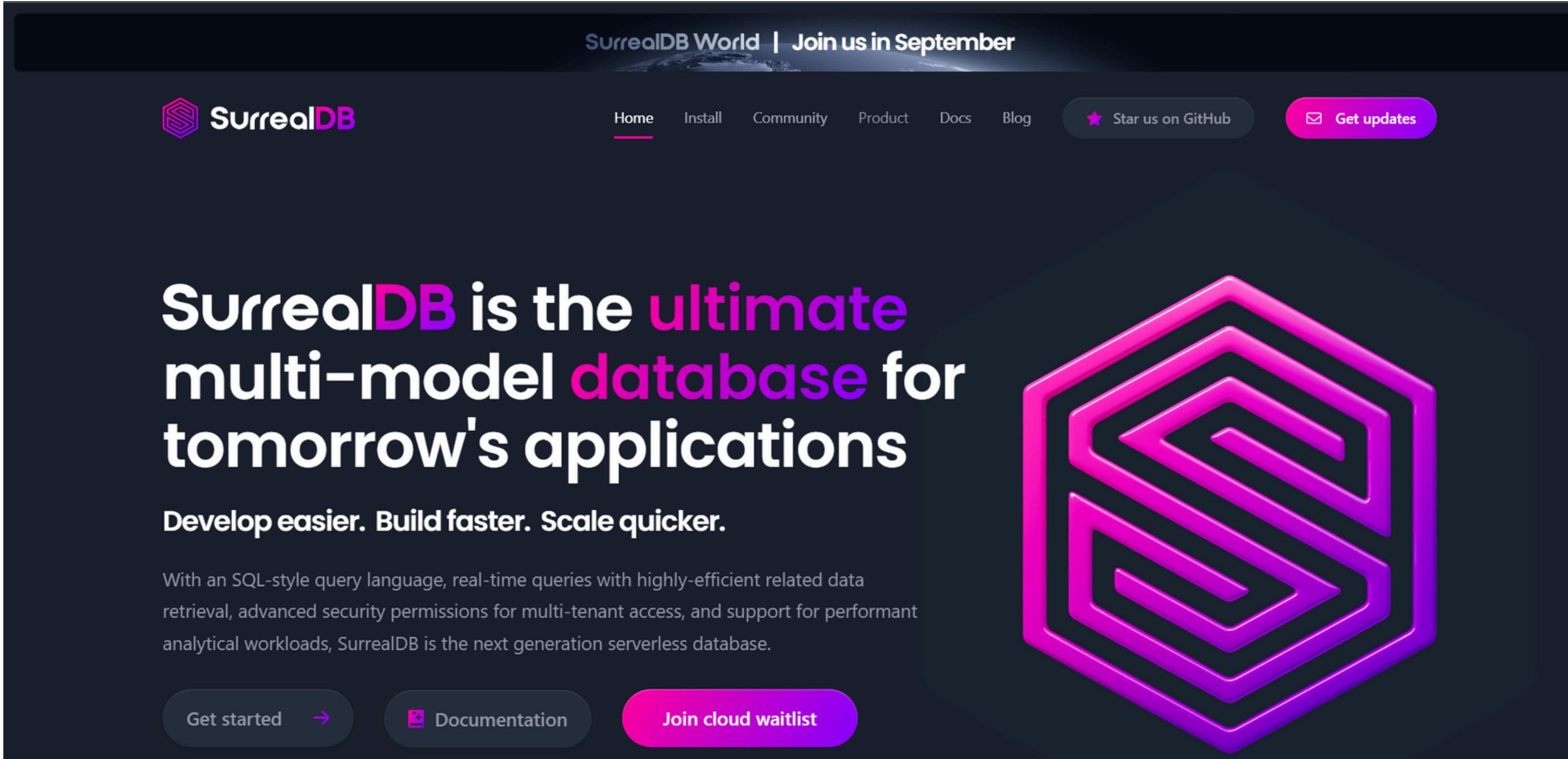
#[macro_use] extern crate rocket;

#[get("/hello/<name>/<age>")]
fn hello(name:&str,age:u8)->String{
    format!("Hello {} year old named {}",age,name);
}

#[launch]
fn rocket()->_{
    rocket::build().mount("/",routes![hello])
}
```



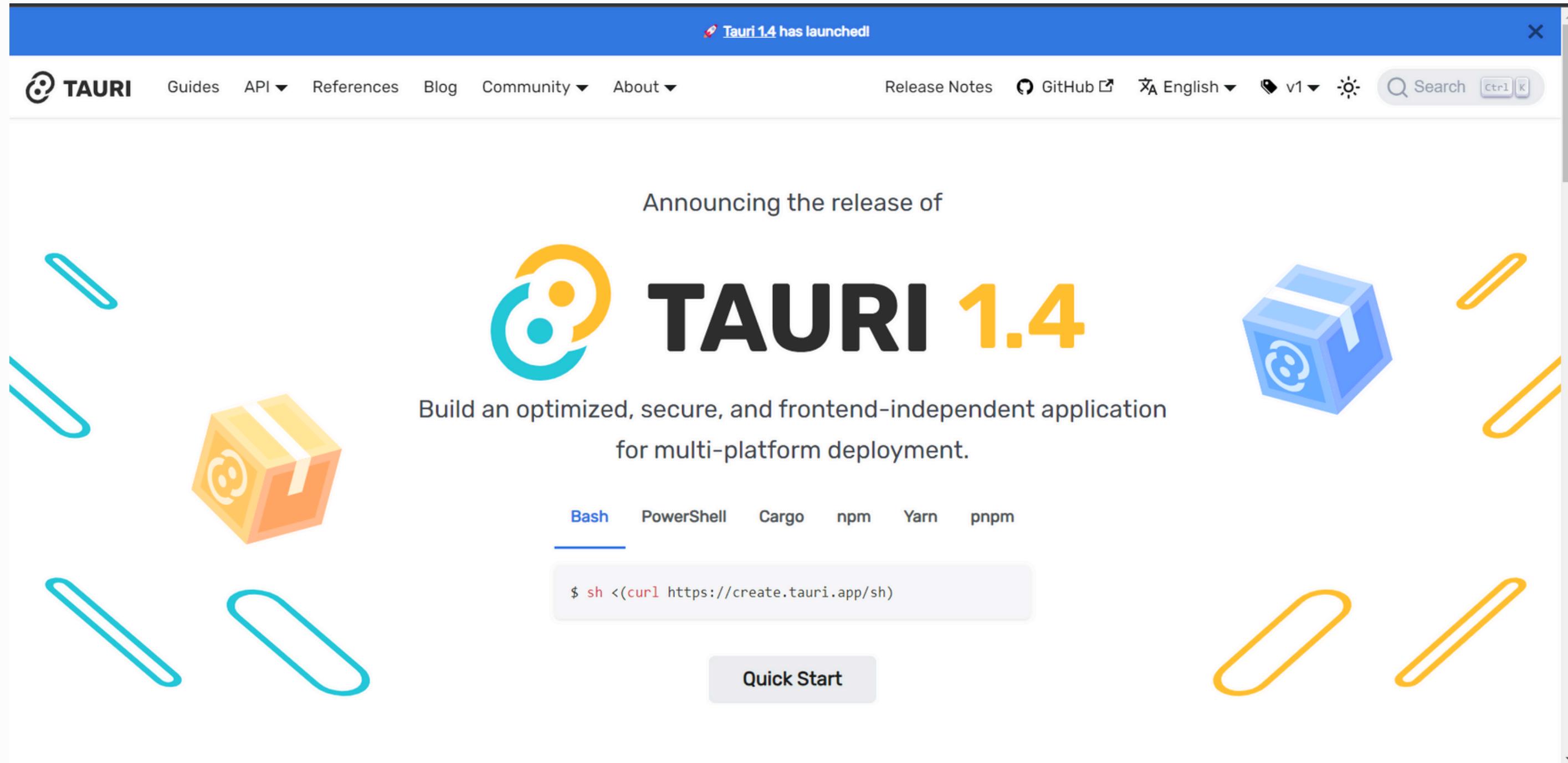
Surreal DB - Surreal SQL



The screenshot shows the official website for SurrealDB. At the top, there's a banner with the text "SurrealDB World | Join us in September". Below the banner is the SurrealDB logo and navigation links for Home, Install, Community, Product, Docs, and Blog. There are also buttons to "Star us on GitHub" and "Get updates". The main headline reads: "SurrealDB is the ultimate multi-model database for tomorrow's applications". Below the headline is the tagline "Develop easier. Build faster. Scale quicker." A detailed description follows: "With an SQL-style query language, real-time queries with highly-efficient related data retrieval, advanced security permissions for multi-tenant access, and support for performant analytical workloads, SurrealDB is the next generation serverless database." At the bottom are three calls-to-action: "Get started", "Documentation", and "Join cloud waitlist". To the right of the text is a large, stylized purple hexagonal logo consisting of nested, rounded hexagons.



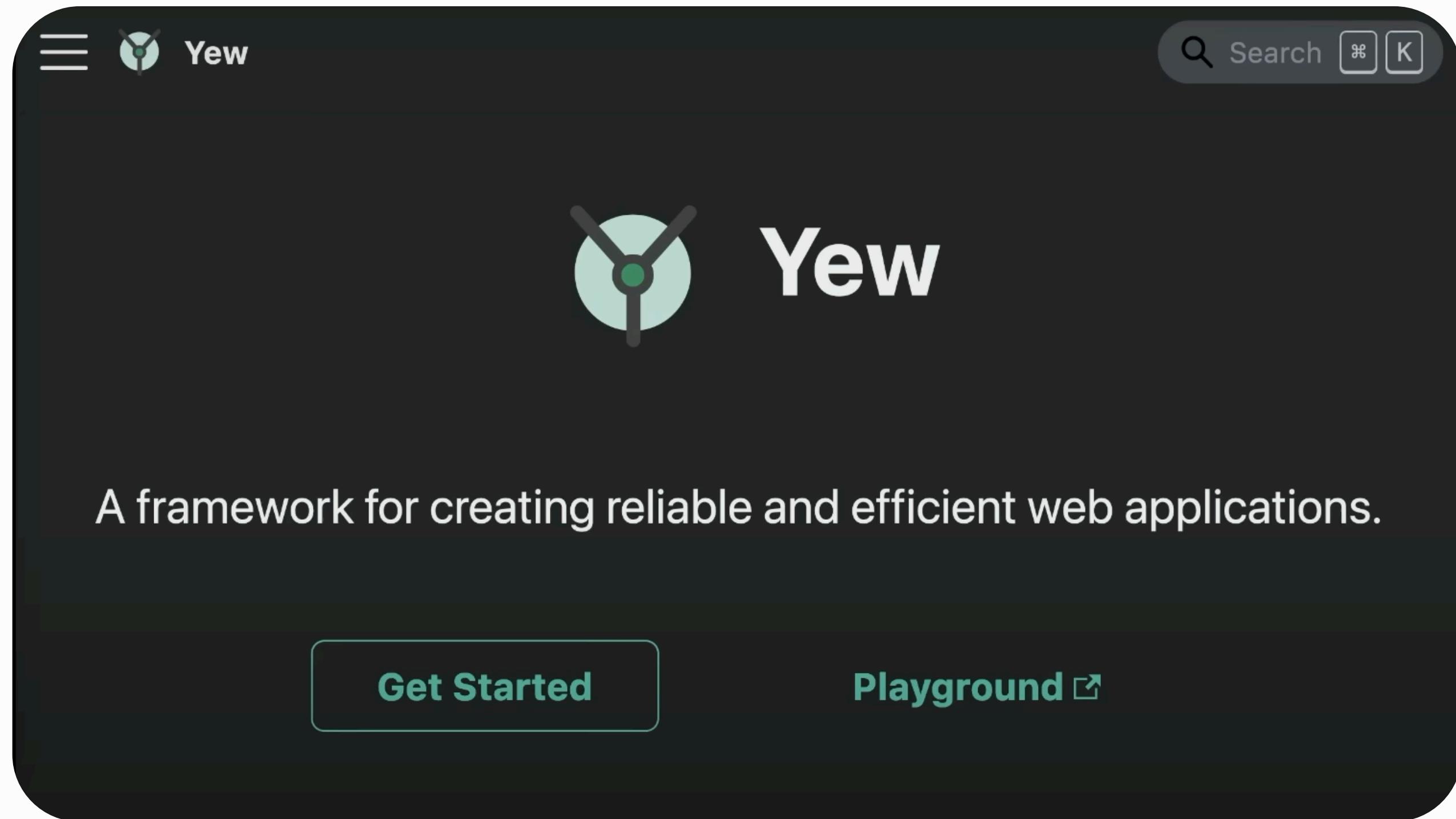
Tauri - Desktop Applications



The screenshot shows the Tauri website homepage. At the top, a blue banner displays the message "Tauri 1.4 has launched!" with a small rocket icon. Below the banner is the Tauri navigation bar with links for Guides, API, References, Blog, Community, About, Release Notes, GitHub, English, v1, and a search bar. The main content area features the Tauri logo and the text "Announcing the release of TAURI 1.4". It also includes the subtext "Build an optimized, secure, and frontend-independent application for multi-platform deployment." Below this, there are tabs for Bash (which is selected), PowerShell, Cargo, npm, Yarn, and pnpm. A code snippet in a box shows the command "\$ sh <(curl https://create.tauri.app/sh)". At the bottom of the page is a "Quick Start" button.



Yew - Frontend



Yew - example

```
● ● ●

use yew::prelude::*;

#[function_component]
fn App->Html{
    html!{"hello world"}
}

fn main(){
    yew::Renderer::  

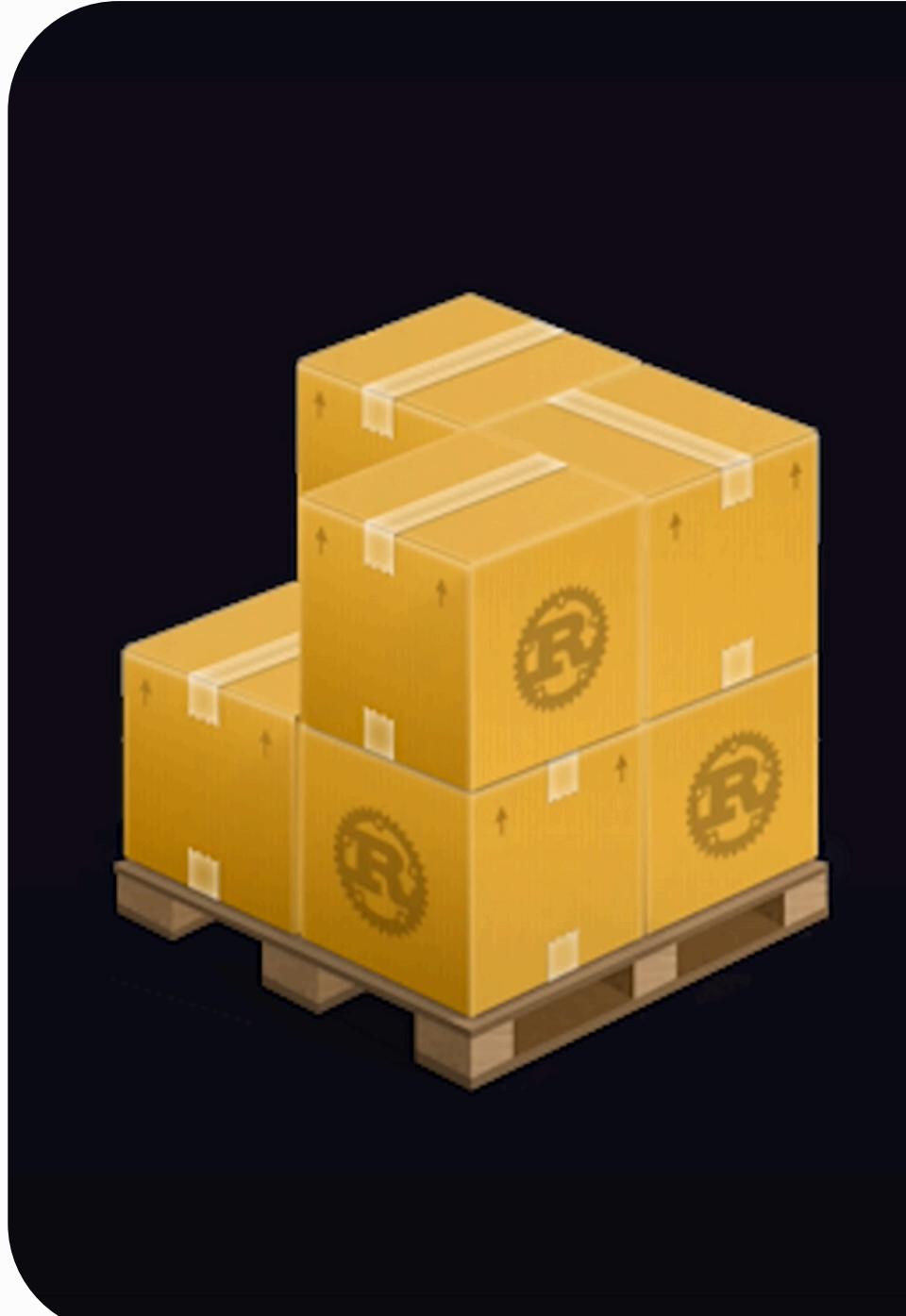
App>::new().render();
}
```



Cargo and Crates.io



Each package as crate



crates.io Click or press 'S' to search... Browse All Crates | Docs | Log in with GitHub

The Rust community's crate registry

[Install Cargo](#) [Getting Started](#)

Instantly publish your crates and install them. Use the API to interact and find out more information about available crates. Become a contributor and enhance the site with your work.

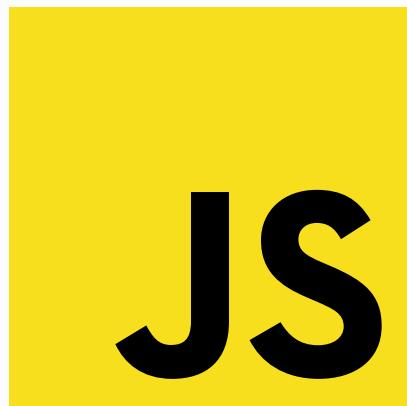
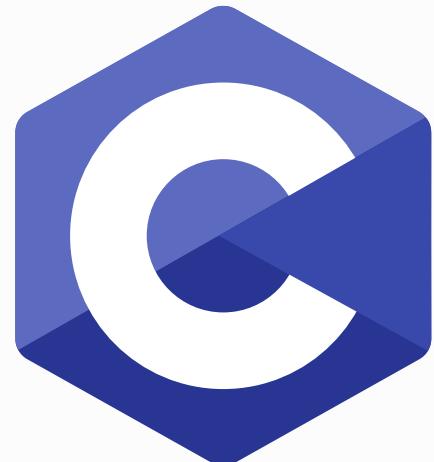
9,970,325,485 Downloads 

68,889 Crates in stock 

New Crates	Most Downloaded	Just Updated
clang-tidy-sarif v0.2.19	rand	mlua v0.6.6
msrv v0.0.0	syn	termin-8 v0.1.2
radio_wav_fix v0.15.0	rand_core	deca v0.0.4
reverie-engine v0.0.3	libc	sysinfo v0.20.5
stretto v0.1.0	quote	union-utils v0.1.4
sql-parser	efc_if	fluid-let



How is Rust different from other languages?



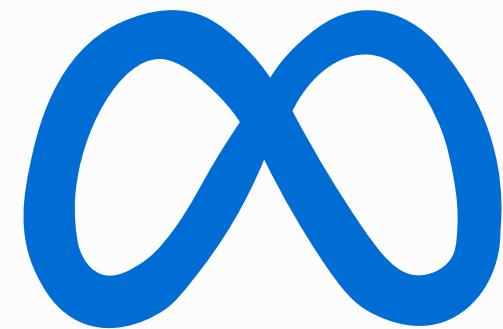
What can Rust be used for?

- Operating Systems
- Databases
- Game Engines
- Server-side Apps
- Embedded Programs
- Web Assembly Programs

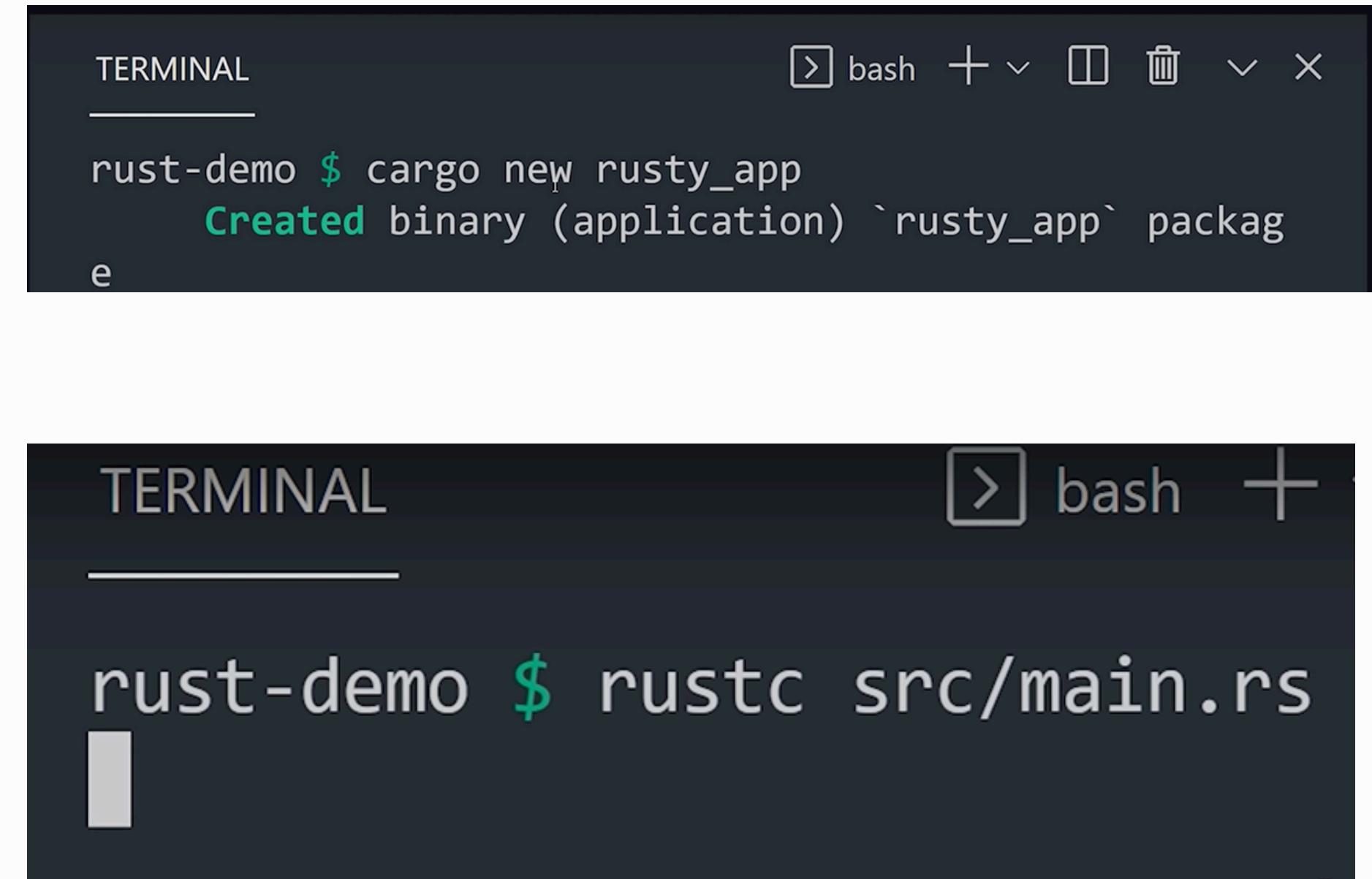
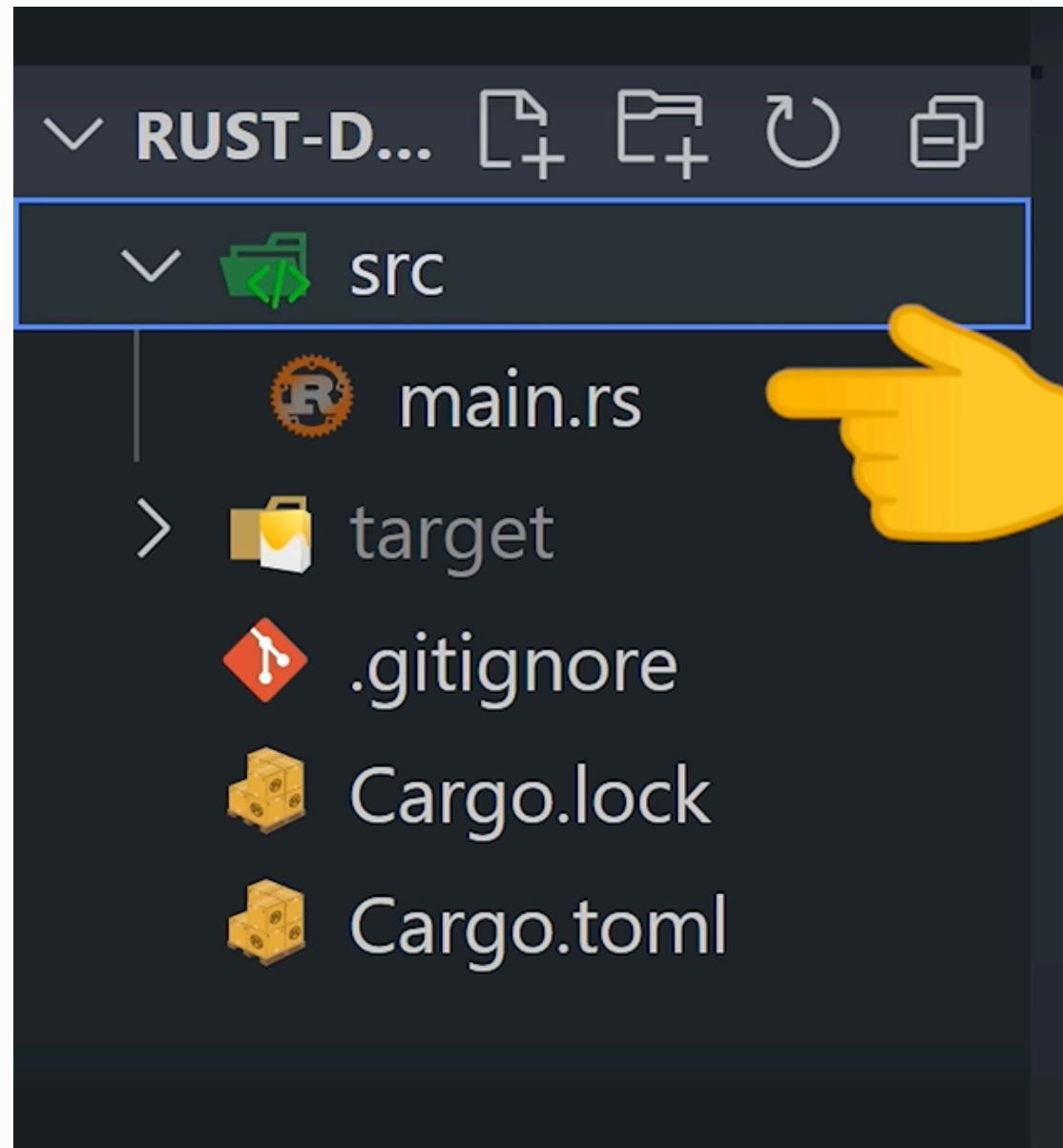


Industry Adoption

<https://github.com/omarabid/rust-companies>



Get Started



The image shows two terminal windows. The top terminal window displays the output of the command "cargo new rusty_app", which has created a new binary application named "rusty_app". The bottom terminal window shows the command "rustc src/main.rs" being run.

TERMINAL

```
rust-demo $ cargo new rusty_app
Created binary (application) `rusty_app` package
```

TERMINAL

```
rust-demo $ rustc src/main.rs
```

Should you learn rust in 2023?





Thank You

See you again on the next
episode of ***THE FOSS Files!***