

Ferris the Rustacean

**Rust Info**

Rust is a modern systems programming language focusing on safety, speed, and concurrency. It accomplishes these goals by being memory safe without using garbage collection.

It wasn’t always so clear, but the Rust programming language is fundamentally about empowerment: no matter what kind of code you are writing now, Rust empowers you to reach farther, to program with confidence in a wider variety of domains than you did before.

Programmers who need to “dip down” into lower-level control can do so with Rust, without taking on the customary risk of crashes or security holes, and without having to learn the fine points of a fickle toolchain

Who is Rust for?

Teams of Developers, Students, Companies, Open Source Developers, People who value speed and stability

Rust also brings contemporary developer tools to the systems programming world:

* Cargo, the included dependency manager and build tool, makes adding, compiling, and managing dependencies painless and consistent across the Rust ecosystem.
* Rustfmt ensures a consistent coding style across developers.
* The Rust Language Server powers Integrated Development Environment (IDE) integration for code completion and inline error messages.

By using these and other tools in the Rust ecosystem, developers can be productive while writing systems-level code.

Rust is for students and those who are interested in learning about systems concepts. Using Rust, many people have learned about topics like operating systems development. The community is very welcoming and happy to answer student questions. Through efforts such as this book, the Rust teams want to make systems concepts more accessible to more people, especially those new to programming.

Hundreds of companies, large and small, use Rust in production for a variety of tasks. Those tasks include command line tools, web services, DevOps tooling, embedded devices, audio and video analysis and transcoding, cryptocurrencies, bioinformatics, search engines, Internet of Things applications, machine learning, and even major parts of the Firefox web browser.

Rust is for people who want to build the Rust programming language, community, developer tools, and libraries. We’d love to have you contribute to the Rust language.

Rust is for people who crave speed and stability in a language. By speed, we mean the speed of the programs that you can create with Rust and the speed at which Rust lets you write them.

Make a program

Cmd

Cargo new \*project name\*

Cd into project, then into src

\*Type in rustc main.rs\*

Run main.exe

Type cargo run or main.exe in the src directory in cmd

**Installing Rust**

Add C++ extension in Visual Studio.

Go to rustling.org

Click Install at the top of the screen

Download the .exe, save and run

Windows might not like it but oh well,

Install with the default (1)

Copy Cargo bin directory (%USERPROFILExxxxxxx) (right click to copy) (close)

Windows search bar environment click Environment variables button in advanced tab

Click path, edit, new, then paste the copy. Apply and exit all that.

To check and see if that’s working Open cmd

Make new directory (mkdir Rust), cd into it

To use Rust project generator type in: cargo new hello\_world (makes package)

(show file explorer, added new folder and inside, there is a .gitignore, Cargo.toml (settings), src holds main.rs )

Cd into hello\_world then into source folder

Then type rustc main.rs (will compile program, generates exec)

Run it (main.exe)

**In VS Code**

Install rls in extensions

Install Code Runner

Add folder to project folder

Then you should be able to run it

**Ready to go!**

To obtain user input and then print the result as output. We need to bring in the io (input.output) library into the project. It comes from the standard library (also known as std (not the best name))

**RUST Sources**

https://doc.rust-lang.org/book/ch00-00-introduction.html

https://doc.rust-lang.org/rust-by-example/index.html

Install Rust: https://www.youtube.com/watch?v=SigbqtVQnyI