

Getting Started

Learn to Code with Rust / Section Review

Rust

- Rust is a programming language. A programmer writes **source code** in Rust.
- The Rust **compiler** is a tool that translates your source code into machine code that the computer can understand.
- The Rust installation includes the compiler (**rustc**), a project manager (**cargo**), and utility tools (**rustup** + **rustfmt**).
- A dependency is a requirement that a piece of software needs to run.

Cargo

- Cargo is a command line tool that assists with Rust project management.
- **cargo new** creates a new Cargo/Rust project.
- **cargo build** compiles the executable from the source code.
- **cargo run** compiles and runs the executable.
- **cargo check** checks the code for violations without compiling the code.
- **cargo fmt** styles/formats the project code according to community conventions.

Functions

- A **function** is a collection of steps to execute in order.
- Every Rust program must have a **main** function, which will automatically run.
- A **function** consists of:
 - The **fn** keyword
 - A function name written in **snake_case**
 - Parentheses to accept inputs
 - Curly braces
 - The body

Outputting Text

- The **println!** macro outputs text to the screen.
- All macros end with an exclamation mark.
- Use parentheses to invoke/run the macro.
- The **println!** macro accepts a string argument.
- A **string** is a piece of text. It consists of characters (alphabetic, digits, symbols, etc).
- End the line/command with a semicolon.

Comments

- Two slashes (//) create a comment, a line ignored by the compiler.
- Use /* */ for multi-line comments. The compiler treats any content between the starting and ending symbols as a comment.

Compilation

- The Rust compiler can build the executable in two models: **debug** and **release**.
 - **debug** is for development.
 - **release** is for the final program.
- The **Run** button from **rust-analyzer** runs **cargo run** behind the scenes, compiling and running your executable in debug mode.