Getting Started with Rust

Contents

Getting Started with Rust 1
Install Rust in Windows 1
Install VSCode Extension Pack for Rust 1
Tutorial 1: Hello World 2
Tutorial 2: Square Calculator 3
Tutorial 3: Guessing Game 5
Assignment: Create Your Own Rust Program 5
  Assignment Submission 5

Time required: 90 minutes

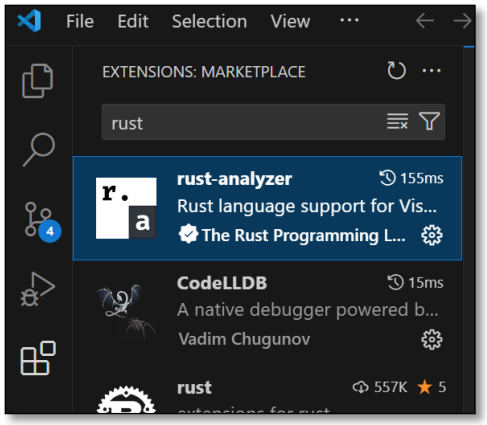
Install Rust in Windows

Before you can compile and run any Rust programs on Windows, you must download and install the Rust programming language and compiler.

1. Download and install the https://visualstudio.microsoft.com/visual-cpp-build-tools/
    - Select Desktop Development with C++
    - Click Install.
2. Go to https://www.rust-lang.org/learn/get-started
3. Download and install **Rustup-Init.exe (64-Bit)**.

Install VSCode Extension Pack for Rust

1. Launch Visual Studio Code.
2. Go to the **View** menu → **Extensions** command.
3. Search for **rust-analyzer** → Install it.



Tutorial 1: Hello World

This is the go to reference for Rust: https://doc.rust-lang.org/book/

Time to write our first rust program.

1. Open a command prompt. cargo is a general purpose tool for managing Rust applications.

```
# Create folder and rust application
cargo new hello_world
```

2. This will create a hello_world folder with several files and folders in it.
3. Open this folder with Visual Studio Code.
4. Open the **src** folder → open **main.rs**
5. There is a boilerplate Rust hello world program.
6. Open a command prompt in the hello_world folder.

```
# Build our program
cargo build
```

7. Go to the target/debug folder.
8. There will be an executable file, **hello_world.exe**
9. Time to run our new program.

```
# Run our program
cargo run
```

Tutorial 2: Square Calculator

1. Open a command prompt..

```
# Create folder and rust application
cargo new square_calculator
```

2. Open this folder with Visual Studio Code.
3. Open the **src** folder → open **main.rs**
4. Add the following code:

```rust
use std::io;

▶ Run | ⚙ Debug
fn main() {
    // Print a nice title for our program
    println!("--------------------------------");
    println!("|  Susie's Square Calculator   |");
    println!("--------------------------------");

    // Create a mutable string to store user input
    let mut input = String::new();

    // Prompt the user to enter the length of a side
    println!("Enter the length of a side: ");

    // Read the user input from the standard input (keyboard)
    // store it in the 'input' variable
    io::stdin().read_line(&mut input).expect("Failed to read line: ");

    // Trim any whitespace from the input,
    // parse it as a 64-bit floating-point number,
    // store it in the 'side' variable
    let side: f64 = input.trim().parse().expect("Enter a valid number: ");

    // Calculate the area of the square (side length squared)
    let area = side * side;

    // Calculate the perimeter of the square (side length multiplied by 4)
    let perimeter = side * 4.0;

    // Display the calculated area and permimeter
    println!("     Area: {}", area);
    println!("Perimeter: {}", perimeter);
}
```

5. Open a command prompt in the square_calculator folder.

```
# Build our program
cargo build
```

6. Time to run our new program.

```
# Run our program
cargo run
```

Example run:

```
------------------------------
|  Susie's Square Calculator   |
------------------------------
Enter the length of a side:
2.3
     Area: 5.289999999999999
Perimeter: 9.2
```

Tutorial 3: Guessing Game

Here is a tutorial from the rust book.

https://doc.rust-lang.org/book/ch02-00-guessing-game-tutorial.html

Assignment: Create Your Own Rust Program

Take one of your programs from a previous assignment. Recreate it in Rust.

Assignment Submission

1. Use pseudocode or TODO.
2. Comment your code to show evidence of understanding.
3. Attach the program files.
4. Attach screenshots showing the successful operation of the program.
5. Submit in Blackboard.