**What is Rust?**

Rust is a systems programming language, which means it's used to create computer systems (such as OSs). It can, however, be employed in situations where performance and reliability are critical. Rust's syntax is equivalent to C++'s, it has performance comparable to modern C++, and it hits all the right notes for many experienced developers in terms of compilation and runtime model, type system, and deterministic finalization. Furthermore, Rust is built with the promise of memory safety without the requirement for garbage collection. So, why did we go for Rust for the latest Windows language projection? One factor is that year after year, Rust is the most popular programming language according to Stack Overflow's annual developer survey. While the language has a steep learning curve, it's difficult not to fall in love once you've gotten beyond it. Microsoft is a founding member of the Rust Foundation as well. The Foundation is a non-profit organization that has developed a novel strategy to sustain and grow a vast, participatory open-source ecosystem.

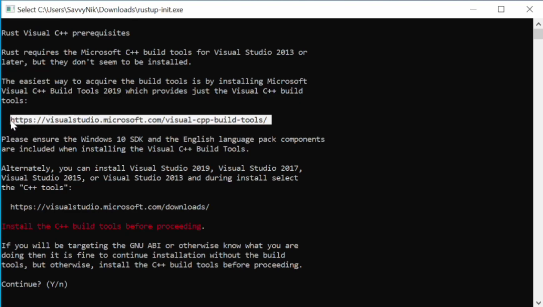
**The components of the Rust development ecosystem/toolkit**

In this section, we'll go through several Rust vocabulary and tools. You can come back to this page to review any of the descriptions.

* A crate is a compilation and linking unit in Rust. A crate can start off as source code, which can then be converted into a binary executable (binary for short) or a binary library (library for short).
* A package is a name for a Rust project. A package consists of one or more crates and a "Cargo.toml" file that defines how to assemble them.
* The Rust toolchain's installer and the updater is "rustup".
* The package management tool in Rust is called Cargo.
* Rust's compiler is called "rustic". The majority of the time, you'll call "rustic" indirectly through Cargo.
* crates.io ("https://crates.io/") is the Rust community's crate registry.

**How to Install Rust?**

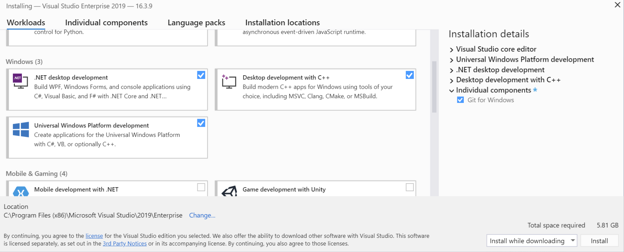
Rust can be downloaded from the [Rust website.](https://www.rust-lang.org/tools/install) When the website detects that you're using Windows, it provides 64- and 32-bit versions of the "rustup" program for Windows. After everything has finished downloading, browse to the download directory and select the "rustup-init" executable, which can then be launched. It will execute a command line that will assist us in configuring "RUST." It will state that "visual-cpp-build-tools" are required.



Copy the Highlighted Link above, and download the Visual Studio CPP build tools

**Install Visual Studio or the Microsoft C++ Build Tools**

This will take us to a page where we may download the Visual Studio installer. We recommend choosing from the following Windows workloads when downloading Visual Studio: NET desktop development, Desktop development using C++, and Universal Windows Platform development. Even if you don't think you'll need all three, we believe it's just easier to choose all three. Git is the default for new Rust projects. As a result, add Git for Windows as a separate component to the mix (use the search box to search for it by name).



After that, we'll be taken to the screen stating that the installation was successful. The visual studio build tools are now available. (In particular, the C++ tools).

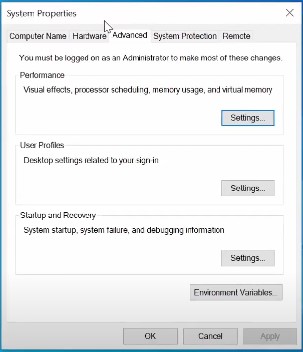
* Note:- The Command lines give information about where the cargo home directory is located as well as all the tools for rust including "rustic, rustup and other commands". The default location would be - (C:\Users\Username\.cargo\bin) directory. And we will have to use it later.
* This step is just to make sure that our directory is added to it.

Open the "rustup-init" installer once more. For a default installation, simply press "ENTER" or input "1." After a successful installation, leave the installer and search for "environment variables" in the Windows search box, then pick "Edit the system env variables.

Click on the Environment Variables on the right corner.

Inside the section of "Environment Variables" select "Path" and hit edit and we would see that (C:\Users\Username\.cargo\bin) directory is already added. Then hit OK on all three sections.

And with that, we have successfully installed Rust on our Windows.

****

**How to Install Rust Nightly?**

It's extremely simple to install nightly with rustup, the programme we installed in Chapter 1, Basics of Rust:

*“rustup default nightly”*

The nightly version of the tools will be installed if you run this command (cargo, rustc, and so on). It will also utilise the nightly version for the appropriate commands.

If you want to revert to the previous stable version, run the command:

*“rustup default stable”*

Because the nightly version is updated often, you may want to update it once a week or more frequently. To do so, use the following command:

*“rustup update”*

Now we are using Rust nightly.