

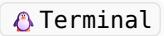
## Instalación de Rust

Rust se instala mediante rustup, el instalador oficial que gestiona versiones del compilador, toolchains y herramientas asociadas. Este proceso es prácticamente idéntico en todos los sistemas operativos.

### Linux y macOS

1. Abre una terminal.
2. Ejecuta el siguiente comando para descargar e iniciar el instalador de Rust:

```
curl --proto '=https' --tlsv1.2 -sSf https://sh.rustup.rs | sh
```



Este comando descarga el instalador de **rustup**, que configurará:

1. El compilador Rust **rustup**
2. **Cargo**, el gestor de paquetes y herramientas de construcción
3. La documentación estándar
4. Las herramientas necesarias para trabajar con Rust

Durante el proceso, se te preguntará si deseas continuar con la instalación por defecto o personalizarla.

La opción recomendada es presionar Enter para aceptar la configuración estándar.

Verás algo como esto:

```
--snip--  
Rust is installed now. Great!  
  
To get started you may need to restart your current shell.  
This would reload your PATH environment variable to include  
Cargo's bin directory ($HOME/.cargo/bin).  
  
To configure your current shell, you need to source  
the corresponding env file under $HOME/.cargo.
```



Reinicia tu terminal o ejecuta:

```
source $HOME/.cargo/env
```



### Windows

Rust en Windows requiere el compilador **MSVC** de Microsoft para generar archivos ejecutables **.exe**. Debes instalar **Visual Studio Build Tools** antes de instalar Rust.

1. Instalar Visual Studio Build Tools

Descarga Visual Studio desde:

<https://visualstudio.microsoft.com/es/downloads/>

2. Componentes requeridos

Rust necesita el workload **Desktop Development with C++** que incluye todas las herramientas esenciales para compilar programas nativos.

3. Instalación minimalista

Si no planeas desarrollar en C++ y quieres instalar solo lo indispensable:

Ve a la pestaña **Componentes individuales** y selecciona únicamente:

- **MSVC v143 – VS 2022 C++ x64/x86 build tools**
- **Windows 11 SDK (10.0.22621.0)**

Después de seleccionar los componentes, haz clic en Instalar y espera a que finalice.

#### 4. Descargar Rust desde la página oficial

Accede a la página oficial:

<https://www.rust-lang.org/tools/install>

Aquí encontrarás tres instaladores según la arquitectura de tu sistema:

- rustup-init.exe (32 bits)
- rustup-init.exe (x64 / 64 bits)
- rustup-init.exe (ARM64)

Selecciona el que corresponda a la arquitectura de tu sistema.

#### 5. Ejecutar el instalador

Ejecuta el archivo **rustup-init.exe** y sigue las instrucciones de la consola.

Cuando aparezca el menú, simplemente presiona Enter para la instalación estándar:

```
--snip--  
1) Proceed with standard installation (default - just press enter)  
2) Customize installation  
3) Cancel installation  
>1
```



En la mayoría de casos es necesario cerrar y volver a abrir la terminal para que Windows reconozca los nuevos comandos **rustc cargo rustup** y sus todas sus herramientas.

#### 6. Verificación de la Instalación

Ejecuta los siguientes comandos en tu terminal:

```
rustc --version # rustc 1.89.0 (29483883e 2025-08-04)  
cargo --version # cargo 1.89.0 (c24e10642 2025-06-23)  
rustup --version # rustup 1.27.0 (2024-08-14)
```



#### 7. Solución de Problemas Comunes

##### 1. Linux/macOS: “comando no encontrado”

Si después de instalar Rust sigues viendo command not found:

```
# Verifica que $HOME/.cargo/bin esté en tu PATH  
echo $PATH | grep .cargo  
  
# Si no aparece, añádelo manualmente  
echo 'export PATH="$HOME/.cargo/bin:$PATH"' >> ~/.bashrc
```



```
source ~/ .bashrc
```

)

### 1. Windows: “rustc no se reconoce como comando”

- Reinicia tu terminal. Este es el problema más común.
- Verifica que la instalación se completó sin errores.
- Verifica manualmente que existe:

```
dir $env:USERPROFILE\.cargo\bin\rustc.exe
```

 Terminal

## Gestión de Versiones con Rustup

### 1. Actualizar Rust

```
rustup update
```

 Terminal

### 1. Cambiar entre canales

```
rustup default stable    # Canal estable (recomendado)
rustup default beta      # Canal beta
rustup default nightly   # Canal nightly (experimental)
```

 Terminal

No necesitas instalar **nightly** a menos que uses características experimentales o herramientas específicas que lo requieran.

## Cargo

Cargo es el sistema de compilación y gestor de paquetes oficial de Rust. Es una herramienta fundamental que simplifica todo el ciclo de desarrollo, desde la creación de proyectos hasta la gestión de **dependencias** y la **compilación**.

Cargo se instala automáticamente cuando instalas Rust a través de rustup.

- Crea y estructura proyectos
- Compila tu código
- Descarga y gestiona dependencias
- Ejecuta tests
- Genera documentación
- Publica paquetes en crates.io