

# Programación en Julia: Primeros pasos

CADI

Héctor Medel

Benjamín Pérez

Tecnológico de Monterrey

January 14, 2024

# Toma de asistencia

# Objetivo

Introducir, de manera práctica, las [bases de programación en Julia](#). A través de ejemplos se revisarán los elementos principales de este lenguaje de programación y paquetes más útiles.

# Fechas y horarios

- ▶ Sesiones sincrónicas: Del 15 al 19 de enero (Lunes a Viernes) de 09:00 a 13:00 hrs.
- ▶ Modalidad: Virtual.

# Políticas para acreditar el curso


- ▶ **Asistencia** de al menos el 80% del taller.
- ▶ A lo largo de la semana se encargarán aproximadamente **4 tareas**. Las tareas serán entregadas en equipos de 3 integrantes.
- ▶ La correcta solución de las tareas deberá ser entregada a más tardar el **viernes 19 de enero a las 23:59 hrs.**

# Temario del curso

1. Motivación de Julia
2. Instalación y editores
3. Paquetes
4. Variables y tipos
5. Flujo de control
6. Funciones
7. Gráficas simples
8. Input/Output
9. Structs
10. Despacho múltiple

# Instalación de Julia

- ▶ Para descargar Julia ingresa al sitio <https://julialang.org/downloads/>
- ▶ Descarga instala el archivo correspondiente a tu sistema operativo.

 [Download](#) [Documentation](#) [Learn](#) [Blog](#) [Community](#) [Contribute](#) [JSOC](#) [Star](#) 43,853 [Sponsor](#)

## Install

Install the latest Julia version ([v1.10.0](#) December 25, 2023) by running this in your terminal:

```
curl -fsSL https://install.julialang.org | sh
```

It looks like you're using a Unix-type system. For Windows instructions [click here](#).

Once installed **julia** will be available via the command line interface.

This will install the **Juliaup** installation manager, which will automatically install julia and help keep it up to date. The command **juliaup** is also installed. To install different julia versions see [juliaup --help](#).

Please star us [on GitHub](#). If you use Julia in your research, please [cite us](#). If possible, do consider [sponsoring us](#).


## Supported platforms


Different OSes and architectures have varying [tiers of support](#).

Operating System	OS Version	Architecture	Support Tier
macOS	10.10+	x86-64 (64-bit)	Tier 1
	11.4+	ARMv8 (64-bit)	Tier 1
	10.6+	i686 (32-bit) / x86-64 (64-bit)	Tier 3
Windows	10+	x86-64 (64-bit)	Tier 1


# Instalación de Julia

- ▶ Por ahora, es recomendable dejar la configuración por defecto durante el proceso de instalación.
- ▶ Dependiendo de tu OS, sigue las instrucciones del instalador.

[Download](#)[Documentation](#)[Learn](#)[Blog](#)[Community](#)[Contribute](#)[JSoC](#)

 Star

43,853

 Sponsor

## Install

Install the latest Julia version ([v1.10.0](#) December 25, 2023) by running this in your terminal:

```
curl -fsSL https://install.julialang.org | sh
```

It looks like you're using a Unix-type system. For Windows instructions [click here](#)

Once installed **julia** will be available via the command line interface.

This will install the **Juliaup** installation manager, which will automatically install julia and help keep it up to date. The command **juliaup** is also installed. To install different julia versions see [juliaup --help](#).

Please star us [on GitHub](#). If you use Julia in your research, please [cite us](#). If possible, do consider [sponsoring us](#).

## Supported platforms

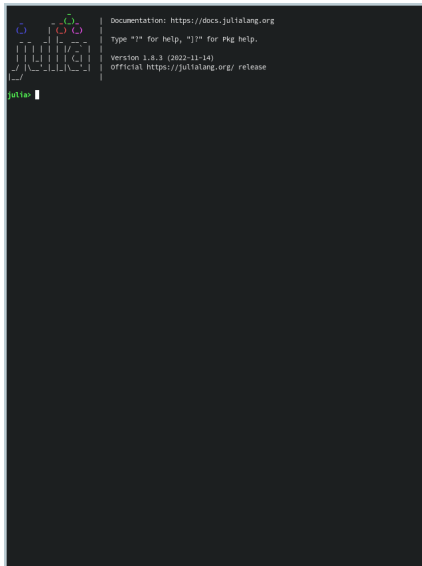
Different OSes and architectures have varying [tiers of support](#).

Operating System	OS Version	Architecture	Support Tier
macOS	10.10+	x86-64 (64-bit)	Tier 1
	11.4+	ARMv8 (64-bit)	Tier 1
	10.6+	i686 (32-bit) / x86-64 (64-bit)	Tier 3
Windows	10+	x86-64 (64-bit)	Tier 1



# Trabajando en Julia – Real Evaluate Print Loop (REPL)

- Al correr el archivo ejecutable de Julia, se abrirá una ventana similar a la siguiente.

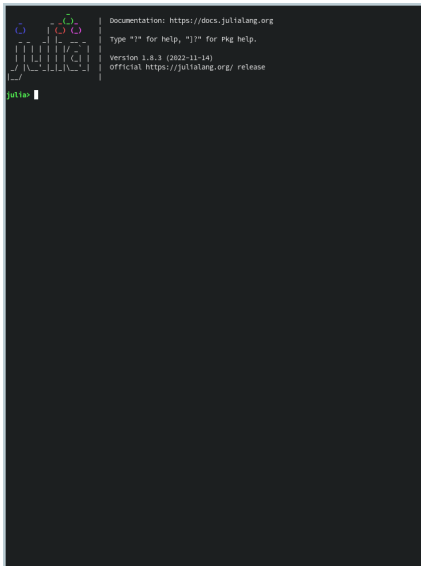
A screenshot of the Julia REPL (Real Evaluate Print Loop) window. The window has a dark background. In the top left corner, there is a small logo consisting of four colored circles (blue, green, red, yellow) arranged in a square. To the right of the logo, the text "Documentation: https://docs.julialang.org" is displayed. Below this, the text "Type '?' for help, ']' for pkg help." is shown. Further down, the text "Version 1.8.3 (2022-11-14)" and "official https://julialang.org/ release" are visible. At the bottom left, the prompt "julia>" is shown with a cursor. The rest of the window is empty.

```
Documentation: https://docs.julialang.org
Type '?' for help, ']' for pkg help.
Version 1.8.3 (2022-11-14)
official https://julialang.org/ release

julia> |
```

# Trabajando en Julia – Real Evaluate Print Loop (REPL)

- Sigamos en la terminal.



```
Documentation: https://docs.julialang.org
Type "?" for help, "?>" for pkg help.
Version 1.8.3 (2022-11-14)
official https://julialang.org/ release

julia> |
```

## Probemos los siguientes comandos

```
julia> 6 * 7  
42  
julia> ans  
42  
julia> ans + 10  
52
```

Si por alguna razón no queremos que se despliegue el resultado, agregamos ; al final.

Podemos **asignar** un valor a una variable

```
julia> a = 6 * 7  
42  
julia> b = "Hola"  
"Hola"
```

## Algunos comandos básicos en el REPL

- ▶ Flecha hacia arriba/abajo nos ayudan a navegar en el historial de comandos ejecutados.
- ▶ Borrar pantalla CTRL+L
- ▶ Interrumpir la ejecución de un comando CTRL+C

## Accesar a la [documentación/ayuda](#)

Cuando ingresamos el caracter ? en el REPL, notemos que cambia de la siguiente manera

```
help?>
```

Busquemos ayuda acerca de la función coseno.

## Accesar a la [documentación/ayuda](#)

Cuando ingresamos el caracter ? en el REPL, notemos que cambia de la siguiente manera

```
help?>
```

Busquemos ayuda acerca de la función coseno.

```
help?> cos
```

```
search: cos cosh cosd cosc cospi acos acosh acosd sincos sincosd sincospi
```

```
cos(x)
```

Compute cosine of x, where x is in radians.

See also [cosd], [cospi], [sincos], [cis].

# Podemos correr `scripts` en el REPL

Generemos un archivo llamado `miscript.jl`, e incluyamos lo siguiente

```
# Script que suma a y b  
a = 1  
b = 2  
suma = a+b
```

Posteriormente, dentro del REPL ejecutemos lo siguiente

```
julia> include("miscript.jl")  
3
```



# Manejo de paquetes (pkg)

Cuando ingresamos el caracter ] en el REPL, notemos que cambia de la siguiente manera

```
(@v1.10) pkg>
```

Esto es conocido como el modo pkg. Dentro de este entorno es donde instalamos paquetes.

## Instalemos un paquete para graficar

Ejecutemos las siguientes líneas dentro del modo pkg

```
(@v1.10) pkg> add Plots
```

## Instalemos un paquete para graficar

Ejecutemos las siguientes líneas dentro del modo pkg

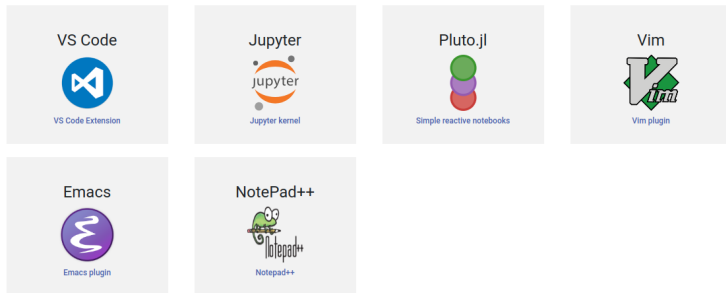
```
(@v1.10) pkg> add Plots
```

Posteriormente, salimos del modo Pkg, y graficaremos una función

```
julia> using Plots  
julia> f(x) = x^2  
julia> xs = 0.0:0.01:2.0  
julia> ys = f.(xs)  
julia> plot(xs, ys)
```

# Por ahora hemos interactuado con Julia vía el REPL...

Existen diversos **IDEs y Editores**, por ejemplo



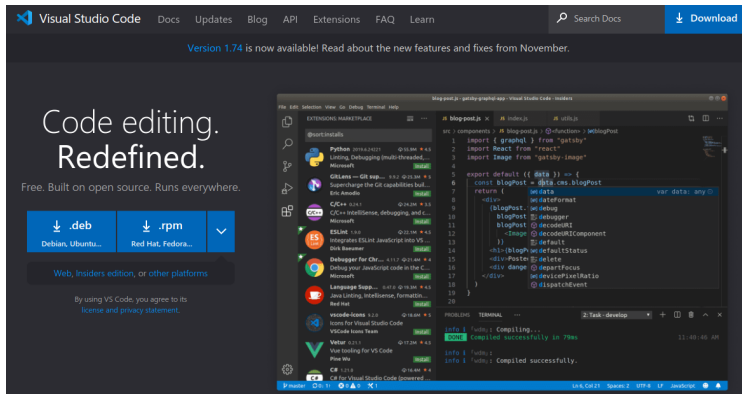
Veremos la instalación de VSCode.

# ¿Qué es VSCode?

- ▶ Editor de código multiplataforma.
- ▶ Soporta varios lenguajes, entre ellos Julia.

# Instalación de VSCode

- ▶ Para descargar VSCode ingresa al sitio <https://code.visualstudio.com/>
- ▶ Descarga instala el archivo correspondiente a tu sistema operativo.



# Extensión de Julia

- ▶ Dentro de VSCode, instalaremos la extensión para Julia.
- ▶ Abre el menú de extensiones que se encuentra en la barra vertical de la izquierda.
- ▶ En el cuadro de búsqueda escribe `julia`, e instala la extensión.