

Introducción a la Programación

Guía de instalación de Haskell, Python, VS Code y extensiones

[Programas a instalar:](#)

[Instalacion de Python3](#)

[Ubuntu/Debian](#)

[Windows](#)

[OSX](#)

[Verificando la instalación](#)

[Instalacion Haskell](#)

[Ubuntu/Debian](#)

[Windows](#)

[OSX](#)

[Verificando la instalación](#)

[Instalacion Visual Studio Code](#)

[Ubuntu/Debian](#)

[Windows](#)

[OSX](#)

[Instalacion extensión para Python](#)

[Prueba de instalación](#)

[Instalacion extensión para Haskell](#)

[Prueba de Instalación](#)

[Linux: Instalación Alternativa de Extensiones vía línea de comando](#)

Programas a instalar:

- Haskell
- Python3
- Visual Studio Code
- Extensiones de VSCode: Python, Haskell, Haskell Syntax Highlighting

Para simplificar la instalación vamos a usar los siguientes gestores de paquetes:

- en Ubuntu vamos a usar *apt-get*
- en Windows vamos a usar *chocolatey*
- en OSX vamos a usar *brew*

Instalacion Haskell

Ubuntu/Debian

En la terminal ingresar:

```
sudo apt update
sudo apt upgrade
```

Para instalar actualizaciones. Luego:

```
sudo apt install ghc
```

Windows

Instalar chocolatey siguiendo los pasos de <https://chocolatey.org/install>. Elegir la opcion 'Individual':

1. Choose How to Install Chocolatey:



Abrir una terminal como administrador: *Inicio -> cmd.exe*

Desde la terminal ejecutar:

```
choco install ghc -y
```

OSX

Abrir una terminal y ejecutar el siguiente comando¹:

```
alias curl='curl -k'
curl --proto 'https' --tlsv1.2 -sSf https://get-ghcup.haskell.org | sh
```

El instalador va a realizar una serie de preguntas. Seleccionar siempre la opción default (apretar ENTER en cada pregunta).

Al finalizar la instalación cerrar la terminal actual ya que el PATH no queda bien configurado. Cuando usemos una nueva terminal va a funcionar correctamente.

¹ Lo ideal seria instalarlo a través de brew, pero al día de hoy (Febrero 2023) el paquete falla. Actualización marzo 2023: brew install ghc funciona!

Verificando la instalación

Nuevamente, al finalizar la instalación se puede abrir una consola de Haskell ejecutando:

```
ghci
```

```
francisco@francisco-TP-X13-G2a:~$ ghci
GHCi, version 9.2.5: https://www.haskell.org/ghc/  :? for help
ghci> 1==2
False
ghci> :q
Leaving GHCi.
```

Luego se pueden escribir comandos como en el ejemplo. Para salir de la consola escribir `:q`

Instalacion de Python3

Ubuntu/Debian

En la terminal (se abre con Ctrl+Shift+T) ingresar:

```
sudo apt update
sudo apt upgrade
```

Para instalar actualizaciones.

Luego:

```
sudo apt install -y python3 python3-pip
```

Windows

Instalar chocolatey siguiendo los pasos de <https://chocolatey.org/install>. Elegir la opcion 'Individual':

1. Choose How to Install Chocolatey:



Abrir una terminal como administrador: Inicio -> cmd.exe

Desde la terminal ejecutar:

```
choco install python3 --version=3.8.3 -y
```

OSX

Abrir una terminal y ejecutar

```
brew install python3
```

En el caso en que brew no esté instalado se puede instalar con los siguientes comandos:

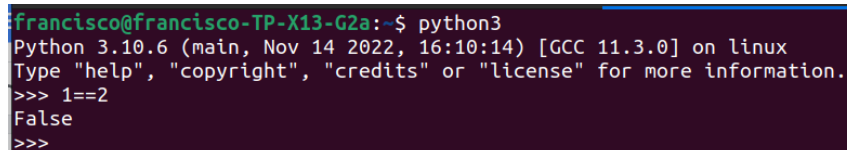
```
curl https://raw.githubusercontent.com/Homebrew/install/master/install.sh -o install.sh  
sudo bash ./install.sh
```

Verificando la instalación

Luego de esto, para asegurarse que la instalación fue correcta, se puede abrir una consola de Python, escribiendo en la terminal:

Linux/OSX: `python3`

Windows: `python`



```
francisco@francisco-TP-X13-G2a:~$ python3  
Python 3.10.6 (main, Nov 14 2022, 16:10:14) [GCC 11.3.0] on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>> 1==2  
False  
>>>
```

Luego se pueden escribir comandos como en el ejemplo. Para salir de la consola: `Ctrl + D`

Las librerías permiten utilizar tipos de datos y funciones ya desarrolladas. Para utilizar una librería es necesario instalarla. Para ello es preciso haber instalado Pip3.

Para verificar la correcta instalación ejecutar desde una terminal:

```
pip3 --version
```

Debería aparecer la versión instalada.

Para instalar una librería luego hacer

```
pip3 install <nombreLibrería>
```

Instalacion Visual Studio Code

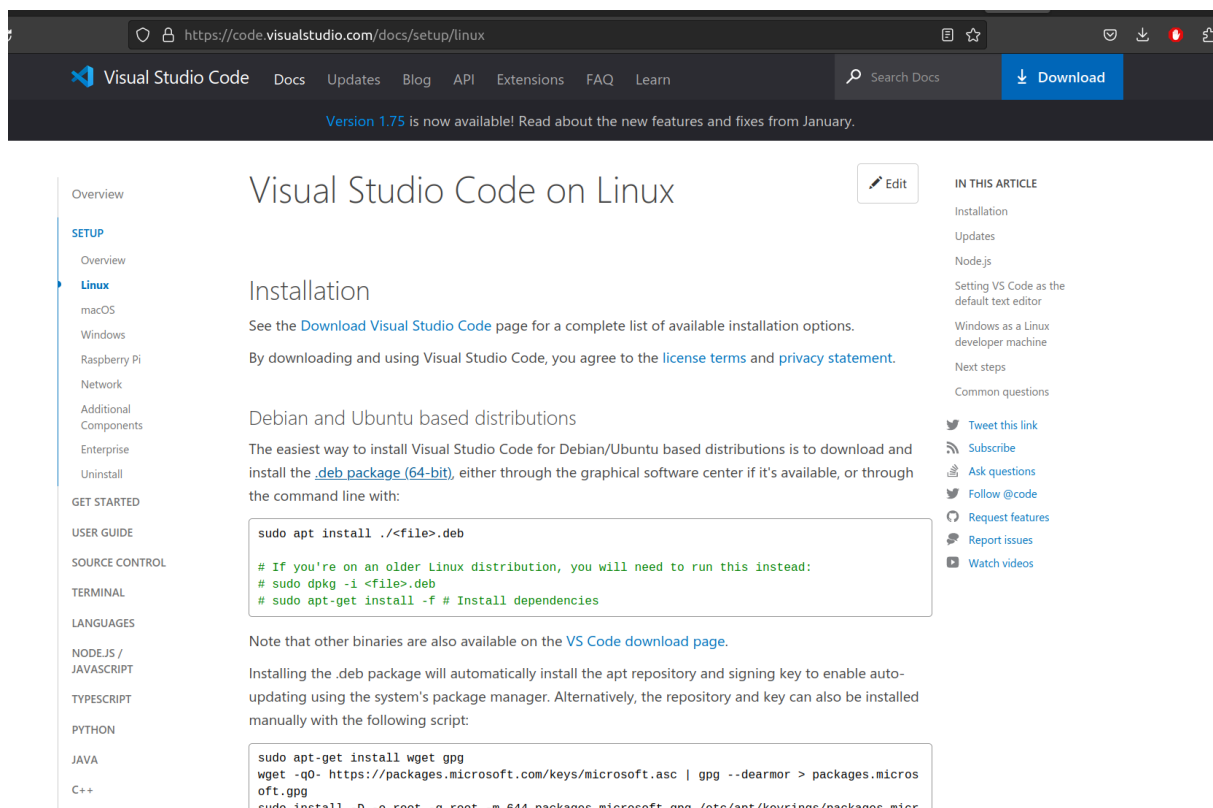
Ubuntu/Debian

Fuente: <https://code.visualstudio.com/docs/setup/linux>

Primero actualizamos el SO:

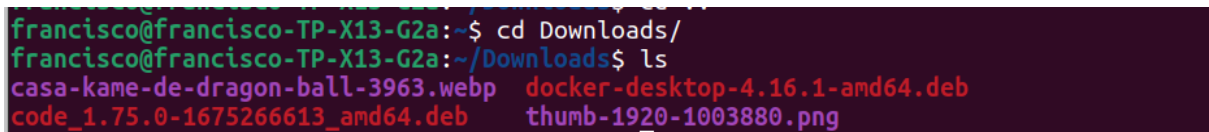
```
sudo apt update
sudo apt upgrade
```

Descargar de <https://code.visualstudio.com/docs/setup/linux> el paquete .deb (64bits)



Luego desde la terminal

```
cd /Downloads
```



(o al directorio donde se haya descargado el paquete). El siguiente comando deberá incluir el nombre de dicho paquete:

```
sudo apt install ./code_1.75.0-1675266613_amd64.deb
```

Finalmente se actualiza la caché y se instala el paquete de la siguiente manera:

```
sudo apt install apt-transport-https  
sudo apt update  
sudo apt install code
```

Windows

Desde una terminal como administrador ejecutar:

```
choco install vscode-insiders.install -y
```

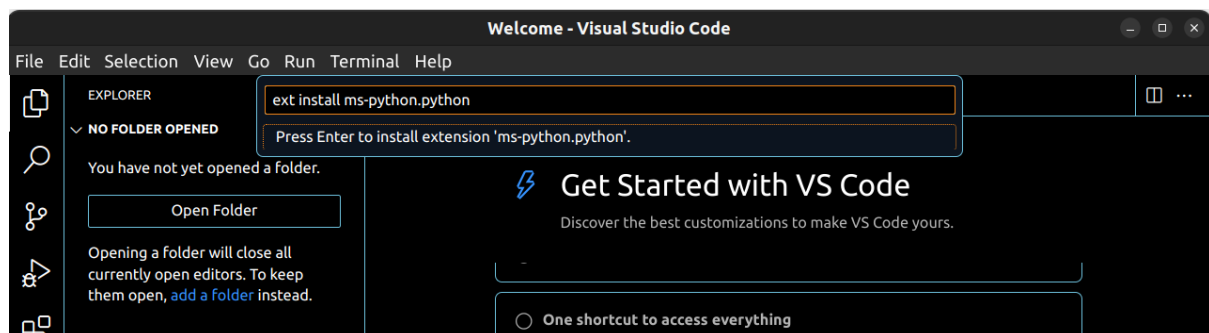
OSX

Descargar el instalador desde: <https://code.visualstudio.com/download>

Abrir el .zip e instalar el .app como cualquier otra aplicación de Mac.

Instalacion extensión para Haskell

En la ventana de Inicio acceder al buscador (al cual se llega con Ctrl+P / Command+P)



Para instalar la extensión de Python escribir:

```
ext install haskell.haskell
```

en el buscador y apretar Enter, se instalará la extensión de Haskell.

Para mejorar la edición instalar también la extensión Haskell Syntax Highlighting:

```
ext install justusadam.language-haskell
```

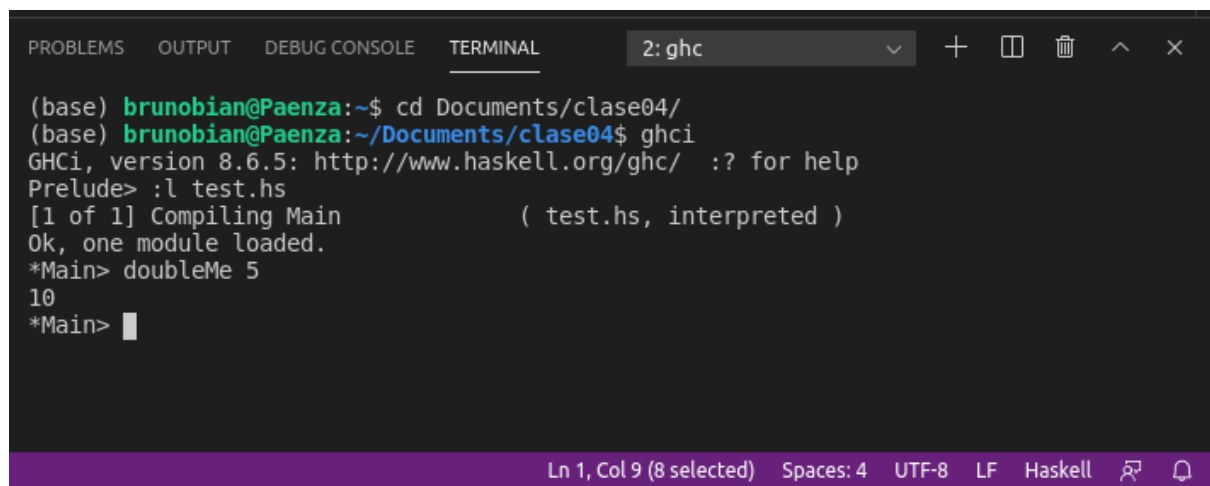
Para poder lanzar el intérprete de Haskell desde adentro del vscode:

```
ext install Meowcolm024.runner2
```

Prueba de Instalación

Luego es posible editar y ejecutar el programa desde la terminal que aparece abajo siguiendo estos pasos:

- Abrir un archivo nuevo File > New File
- Definir nuestra primera función: `doubleMe x = x + x`
- Guardar el archivo como `test.hs` (File > Save)
 - Es importante recordar dónde lo guardamos
 - Vamos a guardarlo en una carpeta "clase04" dentro de Documentos (Documents en inglés) así: Documentos/clase04/
- Abrir una Terminal: menu > Terminal > New Terminal
- En la terminal asegurarse que estemos en el directorio donde guardamos el archivo
 - `cd ~/Documentos/clase04/`²
- Abrir el intérprete interactivo de Haskell: `ghci`
- Dentro del intérprete tenemos que pedirle que cargue nuestro archivo: `:l test.hs`
- Ahora nuestra función ya existe y podemos usarla. Ej: `doubleMe 5`

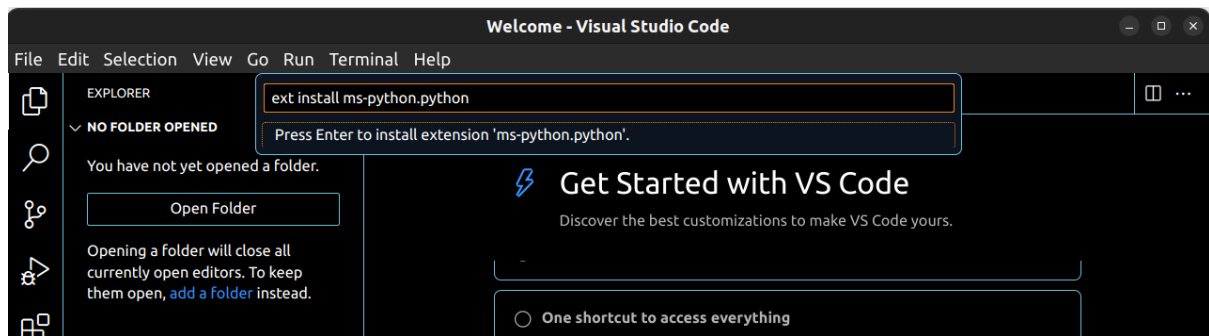
A screenshot of a VS Code terminal window. The terminal title bar shows '2: ghc'. The terminal content shows a user navigating to the directory ~/Documents/clase04/ and running 'ghci'. The GHCi prompt shows the file 'test.hs' being loaded and interpreted. The user then enters 'doubleMe 5' and the output '10' is displayed. The status bar at the bottom indicates 'Ln 1, Col 9 (8 selected)', 'Spaces: 4', 'UTF-8', 'LF', and 'Haskell'.

Nota: Para salir del intérprete interactivo tipear `:q`

Instalacion extensión para Python

² cd: Comando de Linux para cambiar el directorio actual a DIR. El DIR predeterminado es el valor de la variable de shell INICIO (HOME).

En la ventana de Inicio acceder al buscador (al cual se llega con Ctrl+P / Command+P)

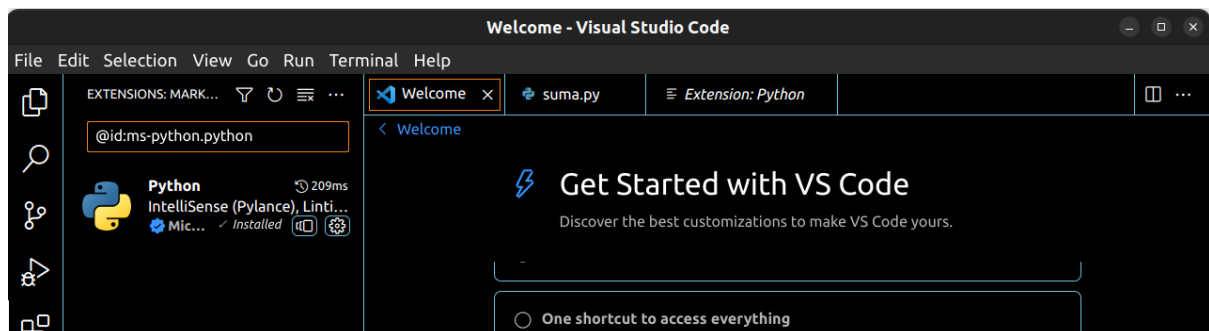


Para instalar la extensión de Python escribir:

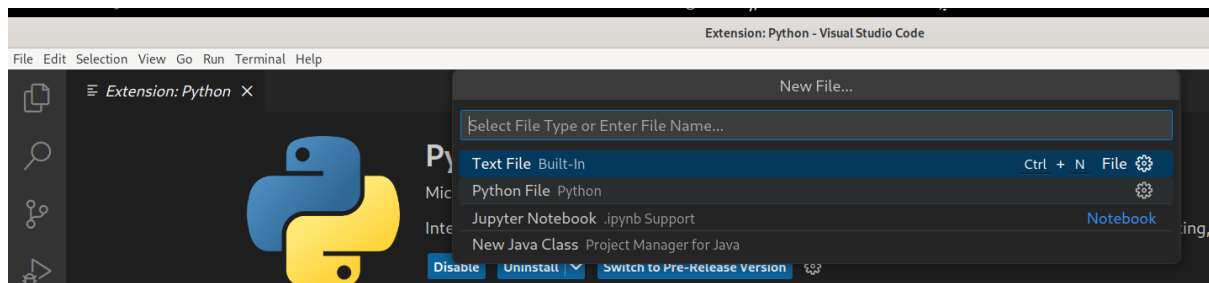
```
ext install ms-python.python
```

Luego Enter.

Cuando la instalación termina debería aparecer en la pestaña de extensiones. Tener en cuenta que es posible que haya que "recargar" (cerrar y volver a abrir el VS Code) cuando se termina de instalar, nos lo indicará en la ventana de la extensión.

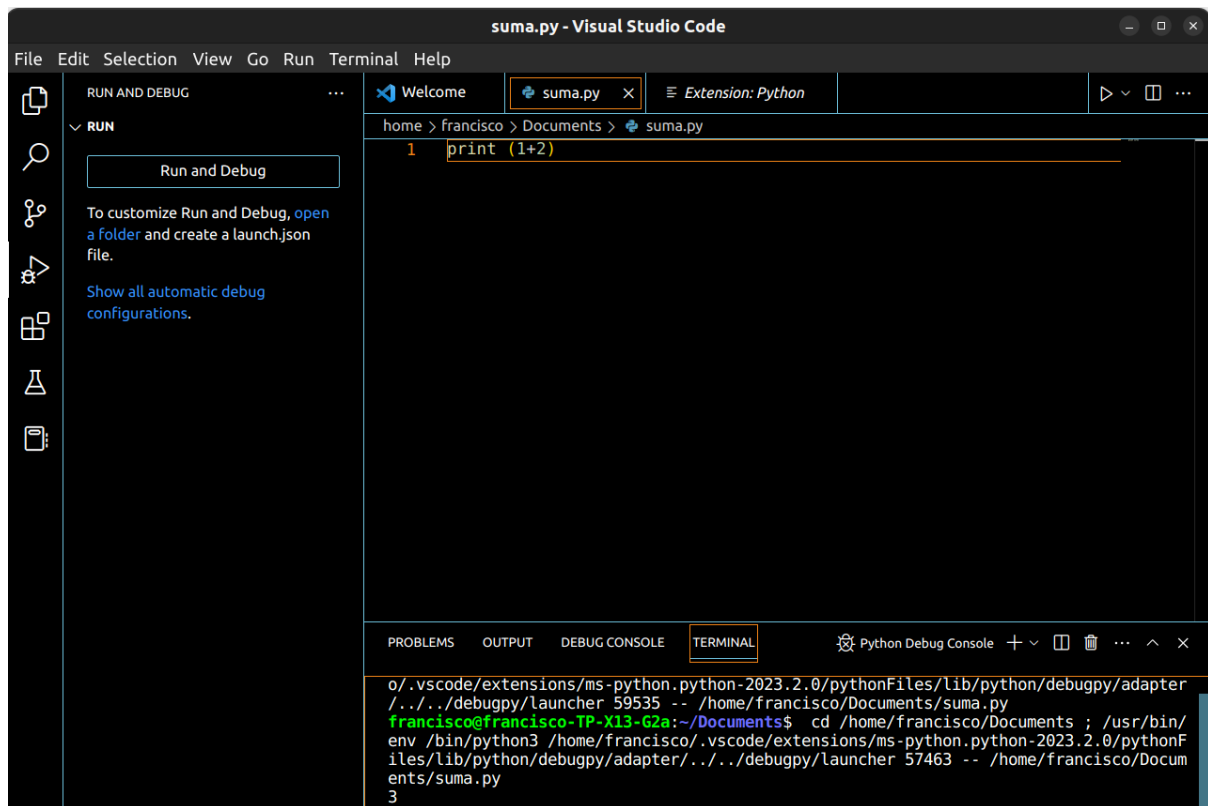


Prueba de instalación



1. Ir a File> New File > Python File
2. Escribir en el área de texto print (1 + 2)
3. Guardar el archivo con nombre: suma.py
4. Presionar el botón de "ejecutar" que se encuentra arriba a la derecha.

Al ejecutar un archivo de Python, el resultado aparecerá en la consola de abajo.



Linux: Instalación Alternativa de Extensiones vía línea de comando

En Linux es posible instalar las extensiones desde línea de comando. En la terminal ingresar:

```
code --install-extension <nombreDeLaExtensión>
```

Por ejemplo para instalar Haskell Syntax Highlighting sería:

```
code --install-extension justusadam.language-haskell
```