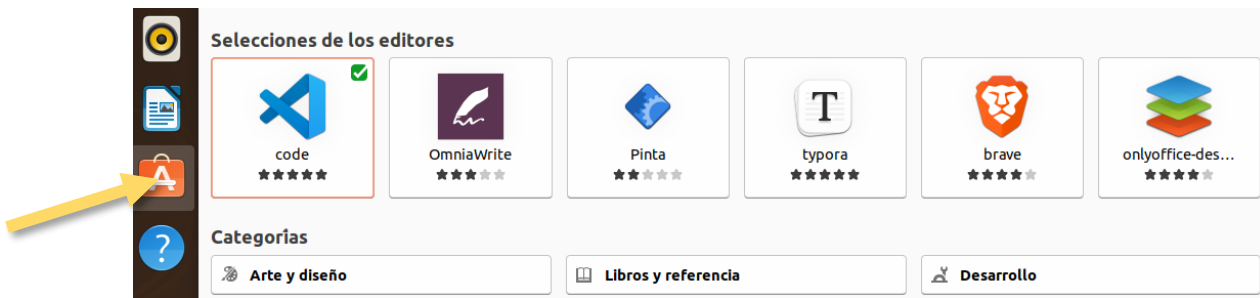
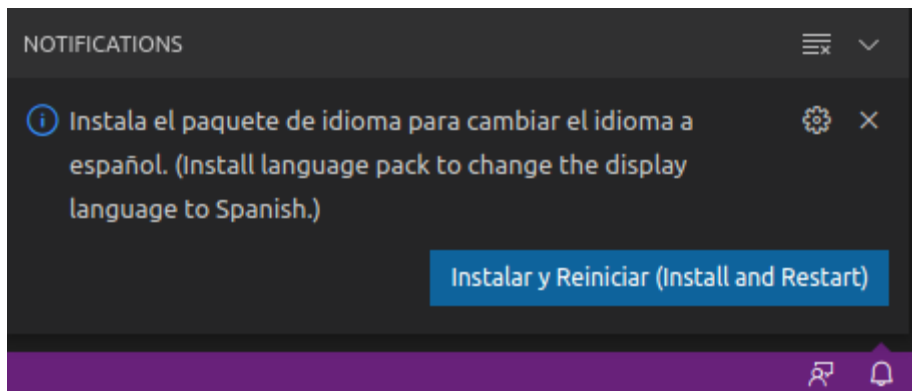


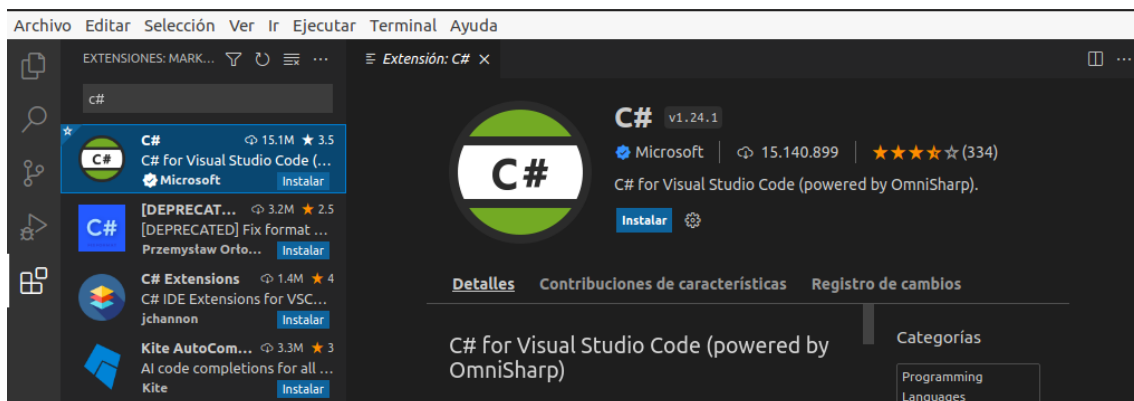
Instala *Visual Studio Code* desde *Ubuntu Software*.



Al abrir Visual Studio Code aparecerá la siguiente notificación:



Instala la extensión C# en Visual Studio Code:



Ahora abre un **Terminal** de Ubuntu (*Ctrl+Alt+t*) e instala el **SDK .NET** siguiendo las instrucciones de sección **20.04** del siguiente enlace: <https://docs.microsoft.com/es-es/dotnet/core/install/linux-ubuntu>

En este mismo terminal crea en tu *directorio de usuario* una carpeta para el proyecto:

```
$ mkdir -p Proyectos/CS/saludo
```

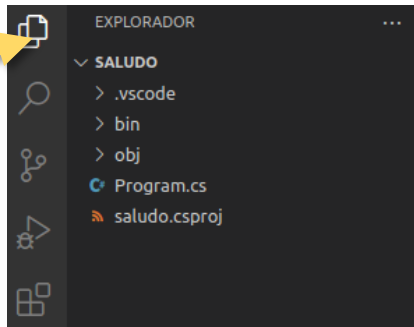
Puedes cerrar este terminal.

Vuelve a *Visual Studio Code* y abre un **Terminal** (*Ctrl+May+`*)

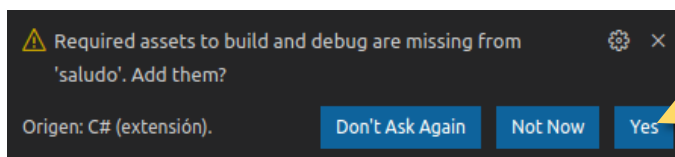
Utilizando este *Terminal* cámbiate al directorio creado: `$ cd Proyectos/CS/saludo`

Crea un proyecto de consola en este directorio: `$ dotnet new console`

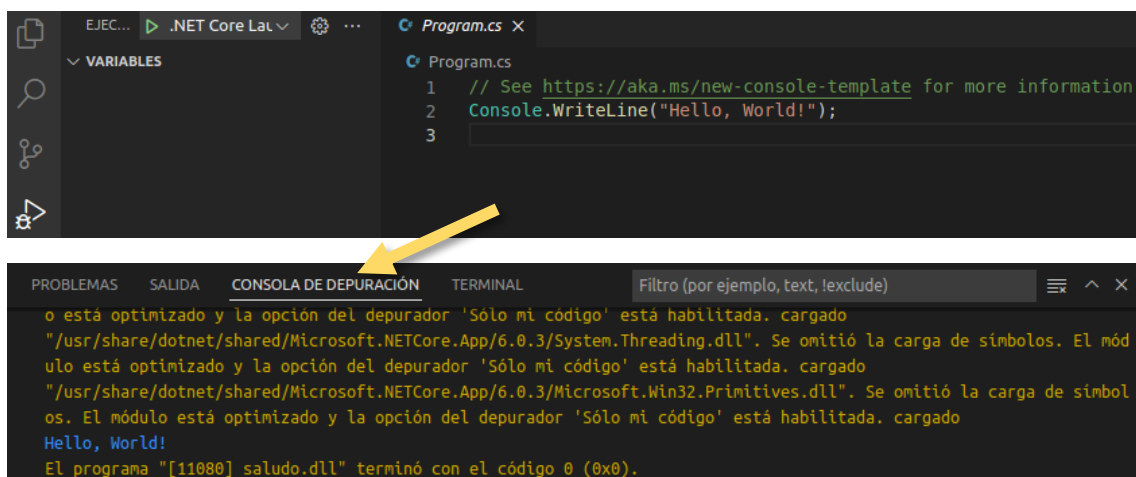
Desde el menú *Archivo* de **VS Code** selecciona *Abrir Carpeta...* y abre la carpeta *saludo* que creaste, la cual aparecerá en el Explorador de la siguiente manera:



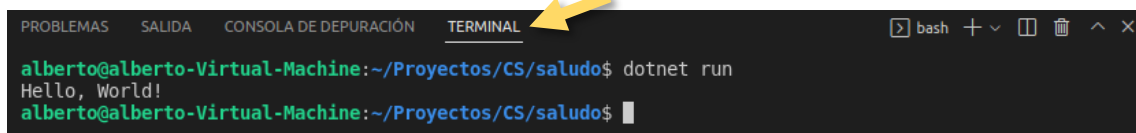
Cuando aparezca el siguiente aviso contesta *Yes*, y se creará el entorno para la depuración:



Ya puedes iniciar la *depuración y ejecución*. Se compilará y si está todo correcto se mostrará en la **CONSOLA DE DEPURACIÓN** el resultado de *Program.cs*



También puedes ejecutar el programa desde la carpeta *saludo*: **\$ dotnet run**



Desde un Terminal de Ubuntu puedes ejecutar el programa entrando en la carpeta *bin/Debug/net6.0* del proyecto:

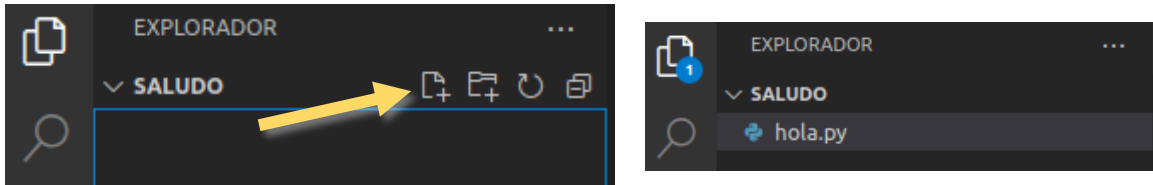
\$ dotnet saludo.dll ← *el runtime de dotnet ejecuta la librería*

\$./saludo ← *se lanza el código intermedio del "ejecutable"*

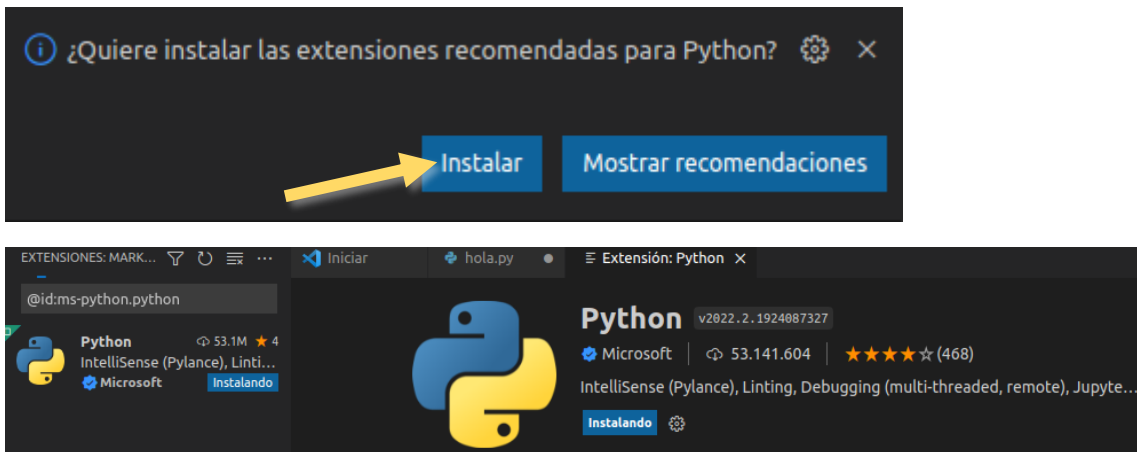
Crea un directorio dentro de *Proyectos* con el nombre *Py*, dentro de este crea *saludo*:

```
$ cd ~/Proyectos
$ mkdir -p Py/saludo
$ cd Py/saludo
$ code . ← Abrirá VS Code en la carpeta actual
```

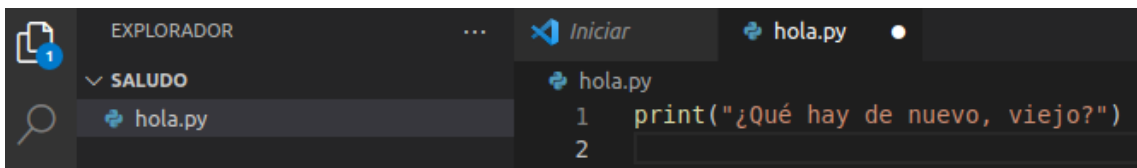
Selecciona la carpeta SALUDO y crea un archivo llamado *hola.py*



Reconocerá la extensión *.py* de **Python** y saldrá el siguiente aviso:

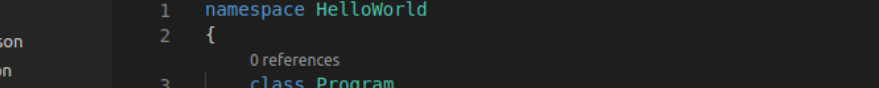


Ahora introduce el código para saludar y guarda el archivo.



Abre un *terminal* en VS Code y ejecuta el archivo: `$ python3 hola.py`

Crea un nuevo proyecto de *consola .NET core*:



```
1 namespace HelloWorld
2 {
3     0 references
4     class Program
5     {
6         0 references
7         static void Main(string[] args)
8         {
9             var maquina=System.Environment.MachineName;
10            var usuario=System.Environment.UserName;
11            Console.WriteLine($"Hola {usuario} \u001b[31m [{maquina}]");
12        }
13    }
```

Modifica el archivo *saludo.py*:

```
hola.py
1 import socket
2 import os
3 from colorama import Fore, Back, Style
4
5 print("¿Qué hay de nuevo, viejo "+os.getlogin()+"? "+"'\033[94m'+socket.gethostname())
6 print(f"\033[37m¿Qué hay de nuevo, viejo {os.getlogin()}? \033[94m[{socket.gethostname()}]")
7 print(f"{Style.RESET_ALL}¿Qué hay de nuevo, viejo {os.getlogin()}? {Fore.BLUE}[{socket.gethostname()}]")
8
```

OTROS PROYECTOS

El comando **dotnet new -list** nos muestra la lista de plantillas disponibles, entre ellas:

- console. Aplicación de consola .NET 6
- web. Aplicación Web ASP.NET vacía.
- webapp. Aplicación Web ASP.NET (páginas web dinámicas) con framework *Razor*.
- blazorserver. Aplicación servidor con el framework *Blazor*.
- angular. Aplicaciones Web dinámicas (a diferencia de los anteriores, centrado en *JavaScript*). Es necesario tener instalado *Node.js*
 - En lugar de crear el proyecto con dotnet...

```
$ sudo npm install -g @angular/cli ← instala Angular en el equipo
```

\$ ng new proyectoAngular ← crea el proyecto en el directorio actual

\$ ng server ← levanta el servicio en *localhost:puerto*

```

PROBLEMAS          SALIDA          CONSOLA DE DEPURACIÓN          TERMINAL
ng - angular1
alberto@alberto-Virtual-Machine:~/Proyectos/angular/angular1$ ng serve
✓ Browser application bundle generation complete.

Initial Chunk Files | Names | Raw Size
vendor.js            | vendor | 1.97 MB
polyfills.js         | polyfills | 294.79 kB
styles.css, styles.js | styles | 173.22 kB
main.js              | main | 49.79 kB
runtime.js           | runtime | 6.51 kB
Initial Total | 2.48 MB

Build at: 2022-03-27T09:28:39.878Z - Hash: 30ab86449ae26899 - Time: 10227ms

** Angular Live Development Server is listening on localhost:4200, open your browser on
http://localhost:4200/ **

✓ Compiled successfully

```