



# Depurar código C++ en VS Code

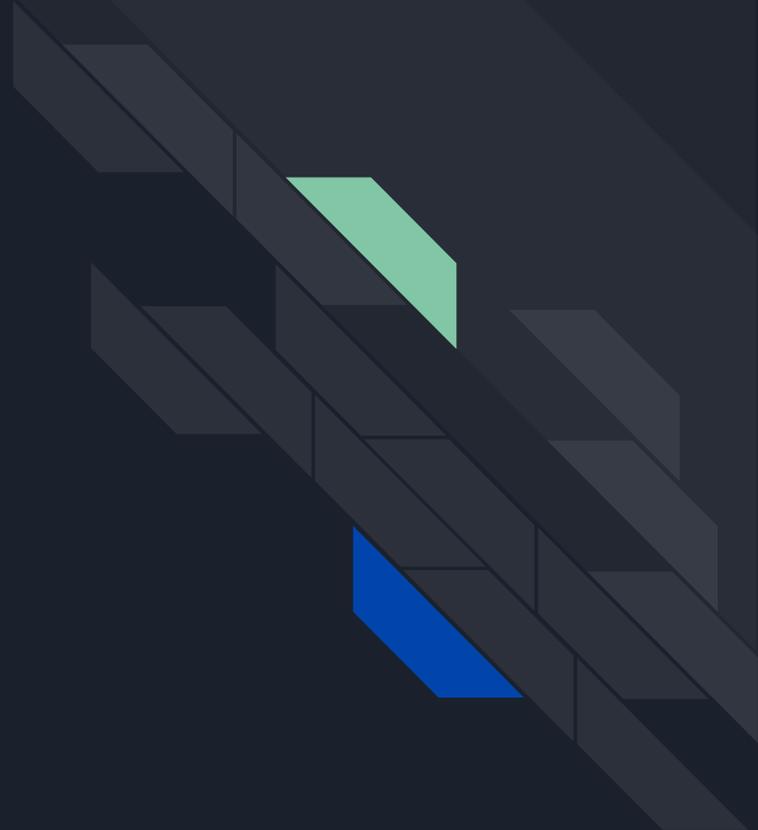
# Índice

[Pre-requisitos](#)

[Configuración de depuración](#)

[Definición de tareas por defecto](#)

[Asignación de tareas al proceso de depuración](#)





# Pre-requisitos

En este documento se presenta como crear una configuración de compilación para Visual Studio Code, en proyectos C++ que usen un Makefile para la compilación de sus ejecutables.

Por ello, se presupone que el en proyecto existe un fichero Makefile que compila el proyecto de forma correcta.

**IMPORTANTE:** se deben indicar la bandera -g durante la compilación para habilitar la inclusión de los símbolos de depuración.



# Configuración de depuración

En VS Code:

- Vamos a la sección de depuración 
- Creamos un nueva configuración launch.json
- Seleccionamos GDB y Configuración Predeterminada
- En el fichero JSON, cambiamos el valor de la clave “program” por el nombre de nuestro ejecutable, como se ve en el ejemplo:

```
"program": "${workspaceFolder}/<ejecutable_proyecto>",
```



# Definición de tareas por defecto

Con este enfoque debemos lanzar el makefile de forma manual para compilar el proyecto antes de comenzar la depuración.

Si queremos automatizar también este proceso, podemos hacer haciendo uso de “preLaunchTask”

Para ello hacemos lo siguiente:

- Usamos el ShortCut Ctrl+Shift+P para abrir el buscador de acciones
- Escribimos “Build” y seleccionamos la acción “Configure Default Build Task”
- Pulsamos Intro en “Create tasks.json from template”
- Seleccionamos Others



# Definición de tareas por defecto

Tras el proceso nos queda un fichero `tasks.json` que se ve como sigue:

```
{
  // See https://go.microsoft.com/fwlink/?LinkId=733558
  // for the documentation about the tasks.json format
  "version": "2.0.0",
  "tasks": [
    {
      "label": "echo",
      "type": "shell",
      "command": "echo Hello"
    }
  ]
}
```



# Definición de tareas por defecto

Ahora tenemos que configurar la tarea, en nuestro caso, compilar el proyecto

El atributo **label** es solo un nombre para la tarea, podemos poner el que queramos.  
(Build o Compilar son buenos ejemplos para nuestro caso)

Dejamos **type** en shell

Cambiamos el **command** al comando que queremos ejecutar, es decir, **make**

```
{  
    "label": "Build",  
    "type": "shell",  
    "command": "make"  
}
```



# Definición de tareas por defecto

Para asignar esta tarea como la tarea por defecto para la compilación del proyecto hacemos:

- Click en Terminal > Run Build Task (Shortcut: Ctrl+Shift+B)
- Hacemos click en Configure Default Build Task
- Seleccionamos la tarea que acabamos de crear (Build)



# Definición de tareas por defecto

```
{  
  // See https://go.microsoft.com/fwlink/?LinkId=733558  
  // for the documentation about the tasks.json format  
  "version": "2.0.0",  
  "tasks": [  
    {  
      "label": "Build",  
      "type": "shell",  
      "command": "make",  
      "problemMatcher": [],  
      "group": {  
        "kind": "build",  
        "isDefault": true  
      }  
    }  
  ]  
}
```



# Asignar tarea al proceso de depuración

Enlazamos la tarea de compilación al proceso de depuración, de forma que se compile el proyecto de manera automática antes de comenzar a depurar.

Añadimos la siguiente propiedad a la configuración de depuración en el fichero `launch.json`

```
"preLaunchTask": "Build",
```