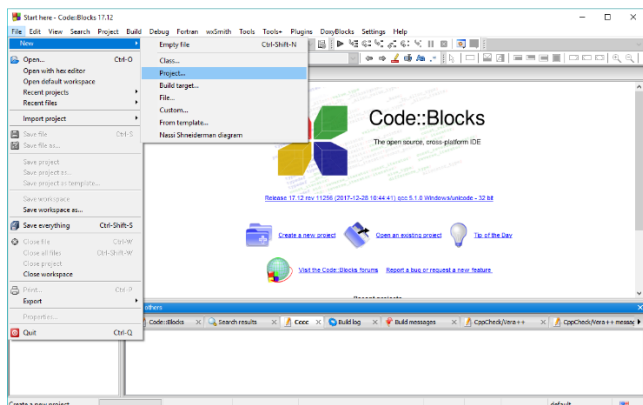
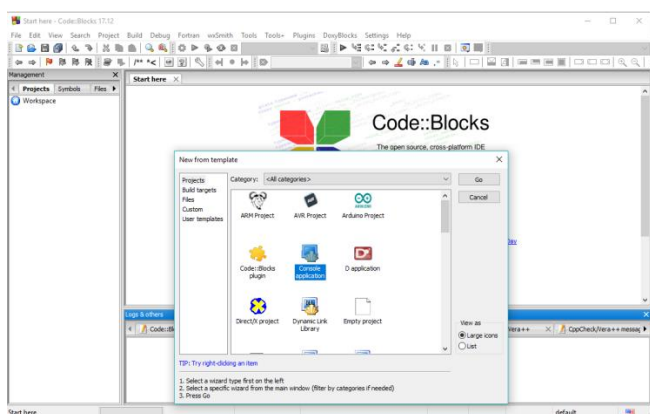


El entorno Code::Blocks. Empezando a programar en C.

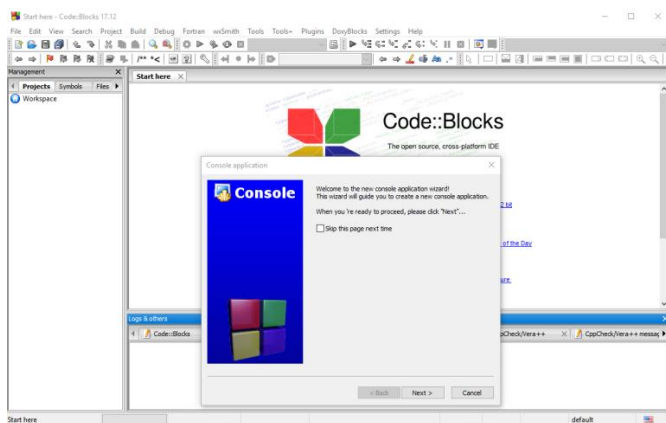
1. Ejecutar la aplicación Code Blocks y crear un nuevo proyecto en un directorio de tu pendrive, para ello seleccionar **File /New / Project**:



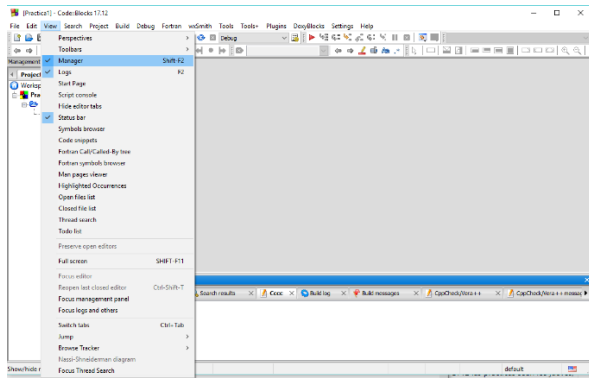
Seleccionar **Console Application**:



Continuar..., seleccionar como **Lenguaje C**, como título por ejemplo **practica1** e indica el directorio dónde quieres almacenarlo:

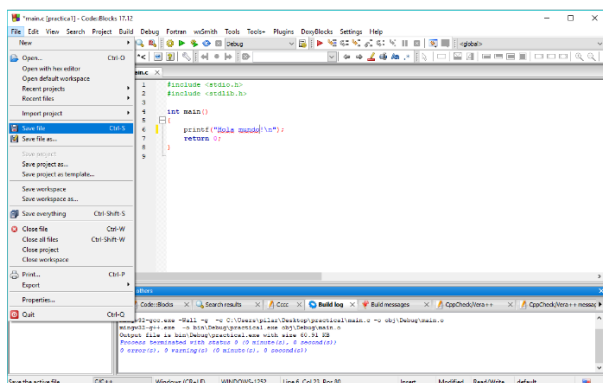


Cuando has creado el proyecto, si no puedes ver la ventana **Management** para trabajar con los ficheros del proyecto, puedes seleccionarla desde **View /Manager**:



2.- Abrir la carpeta **Sources** y el archivo **main.c** para editarlo, compilarlo y posteriormente ejecutarlo (bien desde la opción **Build** o desde los iconos):

Menús:



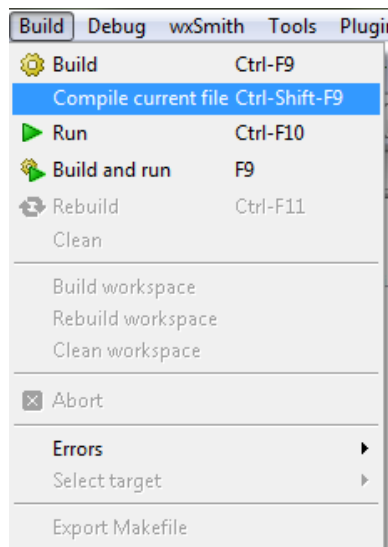
Iconos:



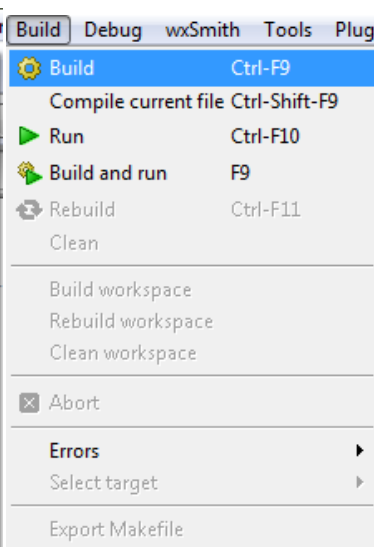
Nuevo Abrir Guardar Guardar
fichero todo

\n provoca un salto de línea


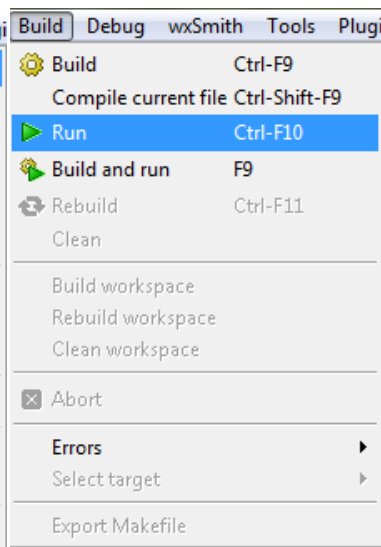
Compilar y ejecutar el programa anterior, desde la barra de menús opción **Build**. Corregir errores si fuese necesario.

A) Menú Build>Compile


Abreviado: Ctrl+Shift+F9

B) Menú Build>Build

Abreviado: Ctrl+F9

Icono: **C) Menu Build>Run**

Abreviado: Ctrl+F10

Icono: **B) y C) en un paso: Menú Build> Build and run**, abreviado F9, icono 

Comprueba que se han creado los ficheros **.o** y **.exe** en las carpetas correspondientes del proyecto creado.

3. Escribir un programa, compilarlo, si hubiera errores corregirlos y volverlo a compilar, guardarlo y ejecutarlo.

Guardar el programa, opciones de *salvar* (**Save**) del menú **File** de la barra de menús

```
// este comentario ocupa una línea
/* este comentario ocupa
varias líneas */

#include <stdio.h>

int main()
```

```
{  
    // instrucciones  
}
```

Incluir el programa en el siguiente recuadro:

```
#include <stdio.h>  
  
#include <stdlib.h>  
  
int main()  
{  
    int n;  
    printf("Introduce un numero: ");  
    scanf("%i", &n);  
    n+=1;  
    printf("El numero posterior es %i.", n);  
    return 0;  
}
```

4. Utilizar el debugger en el programa anterior.

Explicar en el siguiente recuadro, o mediante capturas de pantalla, cómo se ha utilizado el debugger:

