

# C# tarea inicial

Jhostyn Manuel Coats Lerebours, 2025-1865, Friday Afternoon (18:00-21:59)

- Declarar variable de los diferentes tipos, asignarles valor e imprimir el valor.

The screenshot shows a Microsoft Visual Studio interface. On the left is the code editor with a file named 'Program.cs' containing C# code that declares variables of different types (string, int, decimal, float, double, bool, char) and prints them to the console. On the right is the 'Microsoft Visual Studio Debug' window showing the output of the program's execution. The output window displays the values of each variable: 'Hello World', '18', '5.3463853743498', '12.343', '89.9384948949', 'False', 'True', 'a'. Below the code editor, the 'Output' window shows the build log, indicating a successful build at 4:28 PM with no errors or warnings.

```
1 string hello = "Hello World";
2 int number = 18;
3 decimal numberdecimal = 5.3463853743498m;
4 float numberfloat = 12.343f;
5 double numberdouble = 89.9384948949d;
6 bool trueorfalse = false;
7 bool falseorfalse = true;
8 char character = 'a';
9 Console.WriteLine(hello);
10 Console.WriteLine(number);
11 Console.WriteLine(numberdecimal);
12 Console.WriteLine(numberfloat);
13 Console.WriteLine(numberdouble);
14 Console.WriteLine(trueorfalse);
15 Console.WriteLine(falseorfalse);
16 Console.WriteLine(character);
```

Output

```
Show output from: Build
Build started at 4:28 PM...
1>----- Build started: Project: Questionnaire, Configuration: Debug Any CPU -----
1>Skipping analyzers to speed up the build. You can execute 'Build' or 'Rebuild'
1>Questionnaire -> C:\Users\jhost\source\repos\Questionnaire\Questionnaire\bin\Debug\net8.0\Questionnaire.exe
===== Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped =====
===== Build completed at 4:28 PM and took 04.588 seconds =====
```

Microsoft Visual Studio Debug

```
Hello World
18
5.3463853743498
12.343
89.9384948949
False
True
a

C:\Users\jhost\source\repos\Questionnaire\Questionnaire\bin\Debug\net8.0\Questionnaire.exe
ode 0 (0x0).
Press any key to close this window . . .
```

- Buscar cómo se declara una constante en C#

Constants are immutable values which are known at compile time and do not change for the life of the program. Constants are declared with the const modifier. Only the C# built-in types may be declared as const.  
e imprimir el valor.

The screenshot shows a Microsoft Visual Studio interface. On the left is the code editor with a file named 'Program.cs' containing C# code that declares a constant 'PI' using the 'const' modifier and prints it to the console. On the right is the 'Microsoft Visual Studio Debug' window showing the output of the program's execution. The output window displays the value '3.141592653589793'. Below the code editor, the 'Output' window shows the build log, indicating a successful build at 4:30 PM with no errors or warnings.

```
19 const double PI = Math.PI;
20 Console.WriteLine(PI);
21
22
23
24
25
```

Output

```
Show output from: Build
Build started at 4:30 PM...
1>----- Build started: Project: Questionnaire, Configuration: Debug Any CPU -----
1>Skipping analyzers to speed up the build. You can execute 'Build' or 'Rebuild'
1>Questionnaire -> C:\Users\jhost\source\repos\Questionnaire\Questionnaire\bin\Debug\net8.0\Questionnaire.exe
===== Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped =====
===== Build completed at 4:30 PM and took 00.796 seconds =====
```

Microsoft Visual Studio Debug

```
3.141592653589793

C:\Users\jhost\source\repos\Questionnaire\Questionnaire\bin\Debug\net8.0\Questionnaire.exe
ode 0 (0x0).
Press any key to close this window . . .
```

Probar de cambiar su valor luego y ver que es lo que pasa.

The screenshot shows a Microsoft Visual Studio interface. On the left is the code editor with a file named 'Program.cs' containing C# code that declares a local variable 'PI' and prints it to the console. A comment in the code suggests changing its value later. On the right is a 'Microsoft Visual Studio' dialog box asking if the user wants to continue and run the last successful build, with 'Yes' and 'No' buttons. Below the code editor, the 'Output' window shows the build log, indicating a build error CS0128 (a local variable or function named 'PI' is already defined in this scope) and a warning CS0219 (the variable 'PI' is assigned but its value is never used). The status bar at the bottom shows 'Project Questionnaire File Program.cs Line 23 Suppression State'.

```
19 const double PI = Math.PI;
20 Console.WriteLine(PI);
21 //Try changing its value later and see what happens.
22 const double PI = 3.1416;
23 Console.WriteLine(PI);
24
25
```

Output

```
Show output from: Build
Build started at 4:31 PM...
1>----- Build started: Project: Questionnaire, Configuration: Debug Any CPU -----
1>Skipping analyzers to speed up the build. You can execute 'Build' or 'Rebuild'
1>C:\Users\jhost\source\repos\Questionnaire\Questionnaire\Program.cs(2,14,23,16): error CS0128: A local variable or function named 'PI' is already defined in this scope
1>C:\Users\jhost\source\repos\Questionnaire\Questionnaire\Program.cs(23,14,23,16): warning CS0219: The variable 'PI' is assigned but its value is never used
1>Done building project "Questionnaire.csproj" -- FAILED.
===== Build: 0 succeeded, 1 failed, 0 up-to-date, 0 skipped =====
===== Build completed at 4:31 PM and took 00.538 seconds =====
```

Microsoft Visual Studio

There were build errors. Would you like to continue and run the last successful build?

Yes No

Project	File	Line	Suppression State
Questionnaire	Program.cs	23	
Questionnaire	Program.cs	1	
Questionnaire	Program.cs	23	

# C# tarea inicial

3. Declara un entero, incrementarlo, decrementarlo, hacer operaciones con el.

The screenshot shows the Microsoft Visual Studio interface. On the left is the code editor with the following C# code:

```
28 int numberinteger = 10;
29 Console.WriteLine(numberinteger);
30 numberinteger++;
31 Console.WriteLine(numberinteger);
32 numberinteger--;
33 Console.WriteLine(numberinteger);
34 numberinteger = numberinteger + numberinteger;
35 Console.WriteLine(numberinteger);
36 numberinteger = numberinteger - numberinteger;
37 Console.WriteLine(numberinteger);
38 numberinteger = 10;
39 numberinteger = numberinteger * numberinteger;
40 Console.WriteLine(numberinteger);
41 numberinteger = numberinteger / 10;
42 Console.WriteLine(numberinteger);
```

The output window on the right shows the execution results:

```
10
11
10
20
0
100
10

C:\Users\jhost\source\repos\Questionnaire\Questionnaire\bin\Debug\net8.0\Questionnaire.exe 0 (0x0).
Press any key to close this window . . .
```

The build output window at the bottom shows the build process:

```
Build started at 4:35 PM...
1>---- Build started: Project: Questionnaire, Configuration: Debug Any CPU ----
1>Skipping analyzers to speed up the build. You can execute 'Build' or 'Rebuild' command to r
1>Questionnaire -> C:\Users\jhost\source\repos\Questionnaire\Questionnaire\bin\Debug\net8.0\Questi
===== Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped ======
===== Build completed at 4:35 PM and took 00.573 seconds ======
```

4. Declarar un float con valor=10152466.25. Declara un byte que es igual a 5 + el float.

The screenshot shows the Microsoft Visual Studio interface. On the left is the code editor with the following C# code:

```
44
45 float numberFloat = 10152466.25f;
46 Console.WriteLine(numberFloat);
47 byte numberbyte = 5;
48 Console.WriteLine(numberbyte + numberFloat);
```

The output window on the right shows the execution results:

```
10152466
10152471

C:\Users\jhost\source\repos\Questionnaire\Questionnaire\bin\Debug\net8.0\Questionnaire.exe 0 (0x0).
Press any key to close this window . . .
```

The build output window at the bottom shows the build process:

```
Build started at 4:37 PM...
1>---- Build started: Project: Questionnaire, Configuration: Debug Any CPU ----
1>Skipping analyzers to speed up the build. You can execute 'Build' or 'Rebuild' command to r
1>Questionnaire -> C:\Users\jhost\source\repos\Questionnaire\Questionnaire\bin\Debug\net8.0\Questi
===== Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped ======
===== Build completed at 4:37 PM and took 00.501 seconds ======
```

5. Adjuntar comentario de una y de varias líneas en su código. Imprimir la fecha y hora del sistema.

The screenshot shows the Microsoft Visual Studio interface. On the left is the code editor with the following C# code:

```
50 //This is a single-line comment
51 /*These are
52 * multi-line
53 * comments
54 */
55 Console.WriteLine(DateTime.Now);
```

The output window on the right shows the execution results:

```
2/10/2026 4:37:58 PM

C:\Users\jhost\source\repos\Questionnaire\Questionnaire\bin\Debug\net8.0\Questionnaire.exe 0 (0x0).
Press any key to close this window . . .
```

The build output window at the bottom shows the build process:

```
Build started at 4:37 PM...
1>---- Build started: Project: Questionnaire, Configuration: Debug Any CPU ----
1>Skipping analyzers to speed up the build. You can execute 'Build' or 'Rebuild' command to r
1>Questionnaire -> C:\Users\jhost\source\repos\Questionnaire\Questionnaire\bin\Debug\net8.0\Questi
===== Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped ======
===== Build completed at 4:37 PM and took 00.713 seconds ======
```