

Institución Universitaria Pascual Bravo

Herramientas de Programación I

Juan Camilo Cruz Peláez

CC : 1216718181

Grupo : 051

correo : juan.cruz181@pascualbravo.edu.co

Desarrollo de la tarea 1

2. Los dos **IDEs** de mi preferencia, de acuerdo con lo expuesto en clase, son:

1) Visual Studio

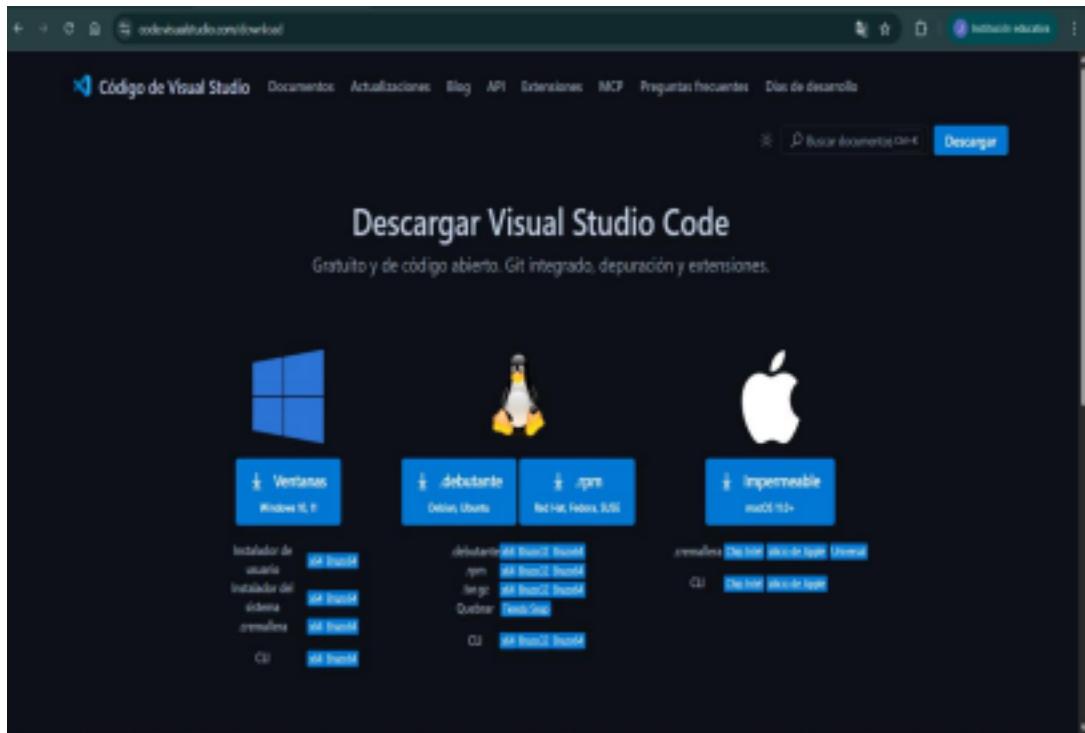
Visual Studio es un IDE desarrollado por Microsoft. Permite crear aplicaciones en C++, C#, Python y otros lenguajes, ofreciendo depuración, compilación y un entorno muy completo para proyectos grandes.

2) Visual Studio Code

Visual Studio Code, también de Microsoft, es un editor de código liviano y multiplataforma. Soporta múltiples lenguajes mediante extensiones y es ideal para proyectos más ágiles y variados.

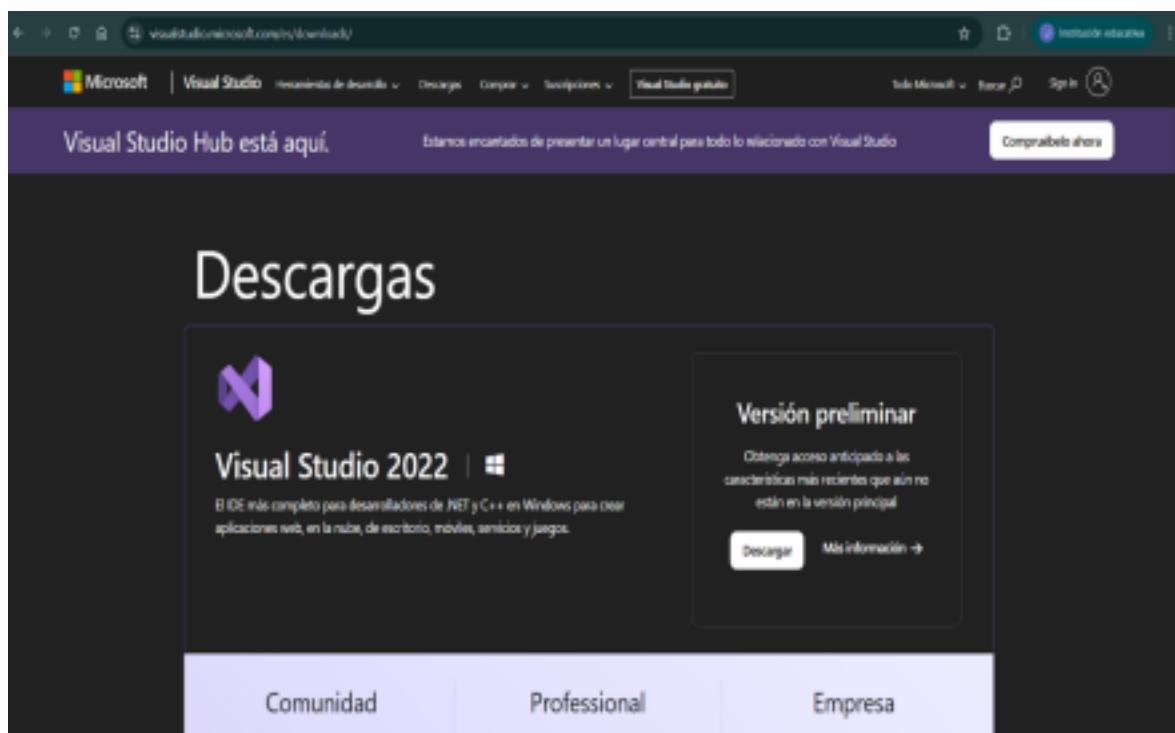
En los siguientes pantallazos demuestro en dónde se encuentra la página de descarga de los IDEs escogidos (**Visual Studio** y **Visual Studio Code**).

Visual Studio Code.



En la página oficial de Microsoft se encuentra la sección de descargas, donde se ofrecen versiones para Windows, macOS y Linux.

Visual Studio 2022



En la página oficial <https://visualstudio.microsoft.com/es/downloads/> se encuentra la sección de descargas de Visual Studio 2022. Desde allí se pueden descargar las ediciones **Comunidad, Profesional y Empresa**.

3. A continuación, se muestran los pantallazos de los programas ejecutados en ambos IDEs, donde se evidencia el código funcional y su respectiva salida en consola.

1) Ejercicios hechos en visual studio 2022.

The screenshot shows a Microsoft Visual Studio Code interface with several windows open. The main window displays a Python script named `PythonApplication1.py`. The script contains various print statements and function definitions. A terminal window titled "PythonApplication1.py" is active, showing the output of the script's execution. The output includes five exercises (EJERCICIO #1 to EJERCICIO #5) with their respective inputs and results. The terminal also prompts the user to press a key to continue. On the right side, there is a "Explorador de soluciones" (Solution Explorer) window showing the project structure, including files like `__init__.py`, `informe.py`, and `PythonApplication1.py`.

```
print("L0R3200 07")
variable = "abc_329"
print(variable)
print(variable[0])
print(variable[-1])
print(variable[0:3])

#0
print("L0R3200 07")
variable = "series datos"
variable = variable.upper()
print(variable)

variable = variable.lower()
print(variable)

print("L0R3200 07")
tupla = ("juan", 7.8, 1992, "medellin", 318834, ['univer', 18])
print(tupla)

lista = []
lista.append("univer", 36)
print(lista)

print(lista[0])
print(lista[0:3])
print(lista[0:3])
print(lista[-1])

#0
print("L0R3200 07")
a = 3
b = 5
c = 20
d = 30

suma = a + b
resta = d - c
multiplicacion = a * b
division = c / b
potenciacion = c ** a

print(suma)
print(resta)
print(multiplicacion)
print(division)
print(potenciacion)

#0
print("L0R3200 07")
print("EJERCICIO #1")
print("C")
print("E")

print("EJERCICIO #2")
print("AEIOU AEIOU")
print("aeiou aeiou")

print("EJERCICIO #3")
print("juan")
print("7.8")
print("1992")

print("EJERCICIO #4")
print("1992")
print("('juan', 7.8, 1992, 'medellin', 318834, ['univer', 18]))")
print("univer")
print("18")

print("EJERCICIO #5")
print("BB")
print("65")
print("500")
print("14.8")
print("52321875")

Prestione una tecla para continuar . . .
```

En estos pantallazos se presentan los ejercicios desarrollados durante la clase utilizando el entorno de desarrollo Visual Studio 2022. Se incluyen los pantallazos del código y de su ejecución, evidenciando la práctica de los conceptos vistos en clase.

2) Ejercicios hechos en Visual Studio Code

The screenshot shows a Visual Studio Code interface with a Python file named 'pp.py' open. The code demonstrates various comparison operators and logical operators. The output terminal shows the results of these operations.

```
print('*' * 50)
# Variables
x = 50
y = 5
z = 17
n = 20

# COMPARACIONES
print("x > y      : ", x > y)
print("x < y      : ", x < y)
print("x == z     : ", x == z)
print("x != y     : ", x != y)
print("x >= z    : ", x >= z)
print("y <= n     : ", y <= n)

# OPERADORES LOGICOS
print("x > y and x == z : ", x > y and x == z)
print("x > y or x < y   : ", x > y or x < y)
print("not(x < y)       : ", not(x < y))
print("(x > y) and (y > z) : ", (x > y) and (y > z))
print("(x < y) or (z == 10) : ", (x < y) or (z == 10))
print("not(x == y)       : ", not(x == y))

x > y      : True
x < y      : False
x == z     : False
x != y     : True
x >= z    : True
y <= n     : True
x > y and x == z : False
x > y or x < y   : True
not(x < y)       : True
(x > y) and (y > z) : False
(x < y) or (z == 10) : False
not(x == y)       : True
```

En este pantallazo se desarrollaron ejercicios enfocados en el uso de **operadores lógicos y comparaciones** en Python. Se trabajaron ejemplos con los operadores and, or, not y con comparadores como `>`, `<`, `>=`, `<=`, `==`, `!=`.

En los pantallazos se puede observar el código escrito y su salida en consola, lo cual permite comprender cómo estos operadores permiten tomar decisiones y evaluar condiciones dentro de un programa.

4.

Error intencional en Visual Studio 2022

The screenshot shows a Visual Studio Code interface with a Python application open. The code contains several intentional syntax errors, such as missing colons and mismatched parentheses. The output window displays a syntax error message: "File "C:/Users/User/Desktop/PythonApplication1/PythonApplication1.py", line 38 syntaxError: invalid syntax". A blue oval highlights this error message. The status bar at the bottom indicates "Python analysis done".

```
1 print("LAMBERTO R")
2 variable = "abc 123"
3 print(variable)
4 print(variable+5)
5 print(" ")
6 print("LAMBERTO R")
7 variable = variable.upper()
8 print(variable)
9 variable = variable.lower()
10 print(variable)
11 print(" ")
12 print("LAMBERTO R")
13 variable = "juan, 7.0, 1992, medellin, 300034"
14 print(variable)
15 print(" ")
16 print("LAMBERTO R")
17 print("juan, 7.0, 1992, medellin, 300034")
18 print(" ")
19 print("LAMBERTO R")
20 print("juan, 7.0, 1992, medellin, 300034")
21 print(" ")
22 print("LAMBERTO R")
23 print("juan, 7.0, 1992, medellin, 300034")
24 print(" ")
25 print("LAMBERTO R")
26 print("juan, 7.0, 1992, medellin, 300034")
27 print(" ")
28 print("LAMBERTO R")
29 print("juan, 7.0, 1992, medellin, 300034")
30 print(" ")
31 print("LAMBERTO R")
32 print("juan, 7.0, 1992, medellin, 300034")
33 print(" ")
34 print("LAMBERTO R")
35 print("juan, 7.0, 1992, medellin, 300034")
36 print(" ")
37 print("LAMBERTO R")
38 print("juan, 7.0, 1992, medellin, 300034")
39 print(" ")
40 print("LAMBERTO R")
41 print("juan, 7.0, 1992, medellin, 300034")
42 print(" ")
43 print("LAMBERTO R")
44 print("juan, 7.0, 1992, medellin, 300034")
45 print(" ")
46 print("LAMBERTO R")
47 a = 1
48 b = 2.3
49 c = 33
50 d = 100
51
52 suma = a + b
53 resta = a - c
54 multiplicacion = a * c
55 division = c / b
56 potenciacion = c ** a
57
58 print(suma)
59 print(resta)
60 print(multiplicacion)
61 print(division)
62 print(potenciacion)
63 print(" ")
```

En este ejercicio se introdujo de manera intencional un error de sintaxis dentro del código en Python. Al ejecutar el programa, Visual Studio muestra el error en la consola de salida (SyntaxError: invalid syntax) indicando la línea en la que ocurrió el problema.

Este ejemplo demuestra cómo el IDE identifica y señala los errores en el código, permitiendo al programador corregirlos más fácilmente.

Error intencional en Visual Studio Code

The screenshot shows a Visual Studio Code interface with a Python file named 'py' open. The code contains several print statements demonstrating comparison operators and logical operators. A specific line of code, 'print('g1225')', is highlighted with a blue rectangle. In the terminal tab at the bottom, an error message is displayed:

```
Traceback (most recent call last):
  File "C:/Users/User/Desktop/py", line 26, in <module>
    g1225
NameError: name 'g1225' is not defined
```

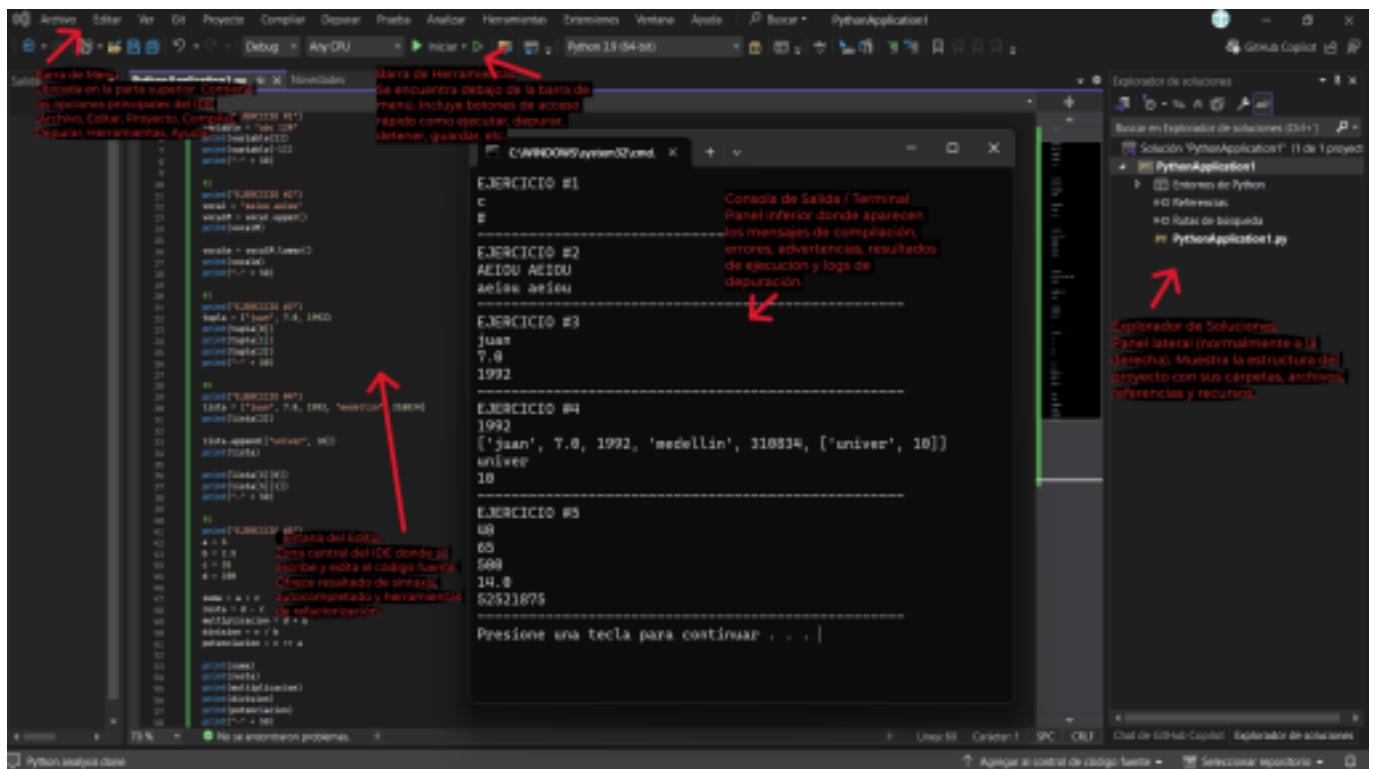
En este ejercicio se insertó intencionalmente un error escribiendo una variable inexistente (g1225).

Al ejecutar el programa, Visual Studio Code muestra en la terminal el mensaje de error: *NameError: name 'g1225' is not defined*.

Este resultado permite evidenciar cómo el IDE identifica que la variable no está definida y señala la línea del código donde ocurre el problema.

5. Pantallazos de los componentes principales de los dos IDEs:

[Visual Studio 2022](#)



Visual Studio Code

