

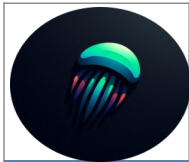


Practica II

Entornos de Desarrollo

Marcos "J." Pérez Gómez - Alias "Srta. Rottenmeier"

Fecha: 31 de oct de 24

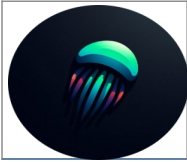


Practica II Entornos de Desarrollo

IES San Juan de la
Rambla

ÍNDICE

1	INSTALACIÓN DE HERRAMIENTAS EN LINUX (UBUNTU DESKTOP)	3
1.1	Visual Studio Code	3
1.2	Compilador de Java	4
1.3	Compilador C++	4
1.4	Compilador Pascal	4
1.5	Compilador COBOL	4
1.6	Compilador Python	4
1.7	Compilador GO	5
1.8	Compilador Lenguaje Ensamblador	5
2	Crea un programa que muestre por pantalla la frase “Madre mía, cómo me gusta la asignatura de ENTORNOS DE DESARROLLO 😊” en diferentes idiomas	6
2.1	C++	6
2.2	Python	6
2.3	Java	7
2.4	COBOL	7
2.5	Rust	7
2.6	Pascal	8
2.7	GO	8
2.8	Lenguaje ensamblador	8
3	¿Qué extensión tienen los archivos de código fuente de estos lenguajes?	9
4	INCLUIR OTROS LENGUAJES	10
4.1	LUA	10
4.2	Haxe	10
5	COMPILAR UN CÓDIGO EN UBUNTU LINUX	12
6	PRUEBAS CON CÓDIGO EN PYTHON	14
7	BIBLIOGRAFÍA Y REFERENCIAS	15



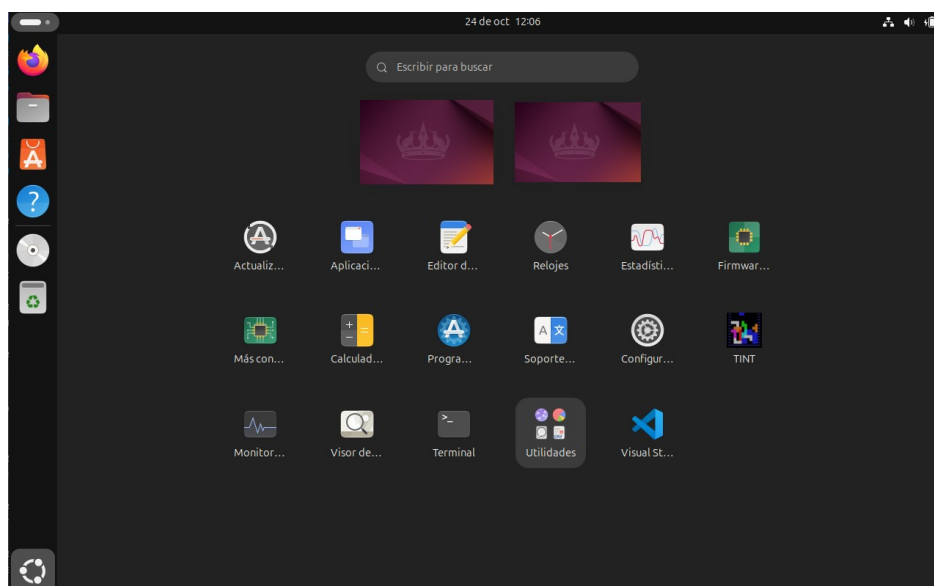
Practica II Entornos de Desarrollo

IES San Juan de la
Rambla

1 INSTALACIÓN DE HERRAMIENTAS EN LINUX (UBUNTU DESKTOP)

1.1 Visual Studio Code

```
marcos@marcos-VirtualBox: ~/Escritorio
marcos@marcos-VirtualBox:~/Escritorio$ sudo apt install ./code_1.94.2-1728494015_
amd64.deb
```





Practica II Entornos de Desarrollo

IES San Juan de la
Rambla

1.2 Compilador de Java

```
marcos@marcos-VirtualBox: ~/Escritorio
marcos@marcos-VirtualBox:~/Escritorio$ sudo apt install openjdk-11-jdk
```

1.3 Compilador C++

```
marcos@marcos-VirtualBox: ~/Escritorio
marcos@marcos-VirtualBox:~/Escritorio$ sudo apt install g++~
```

1.4 Compilador Pascal

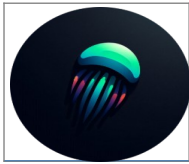
```
marcos@marcos-VirtualBox: ~/Escritorio
marcos@marcos-VirtualBox:~/Escritorio$ sudo apt install fpc
```

1.5 Compilador COBOL

```
marcos@marcos-VirtualBox: ~/Escritorio
marcos@marcos-VirtualBox:~/Escritorio$ sudo apt install gnuccobol
```

1.6 Compilador Python

```
marcos@marcos-VirtualBox: ~/Escritorio
marcos@marcos-VirtualBox:~/Escritorio$ sudo apt install python3
```



Practica II
Entornos de Desarrollo

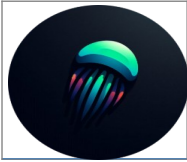
**IES San Juan de la
Rambla**

1.7 Compilador GO

```
marcos@marcos-VirtualBox: ~/Escritorio
marcos@marcos-VirtualBox:~/Escritorio$ sudo zypper install go
```

1.8 Compilador Lenguaje Ensamblador

```
24 de oct 12:18
marcos@marcos-VirtualBox: ~/Escritorio
marcos@marcos-VirtualBox:~/Escritorio$ sudo dnf install nasm
```



Practica II

Entornos de Desarrollo

IES San Juan de la
Rambla

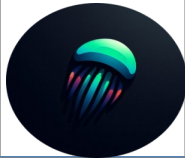
2 Crea un programa que muestre por pantalla la frase “Madre mía, cómo me gusta la asignatura de ENTORNOS DE DESARROLLO 😊” en diferentes idiomas.

2.1 C++

```
1 #include <iostream>
2
3 int main() {
4     std::cout << "Me gusta la asignatura Entornos de Desarrollo :)" << std::endl;
5     return 0;
6 }
```

2.2 Python

```
1 print("Me gusta la asignatura Entornos de Desarrollo :)")
```

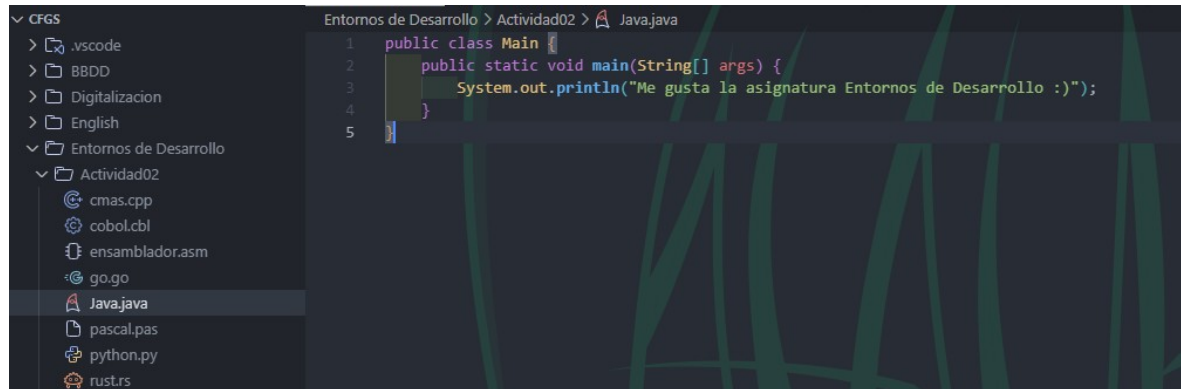


Practica II

Entornos de Desarrollo

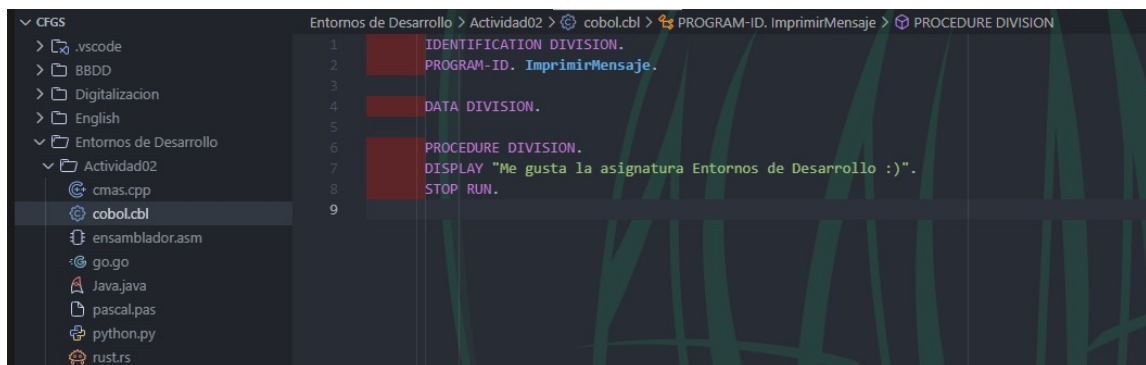
IES San Juan de la
Rambla

2.3 Java



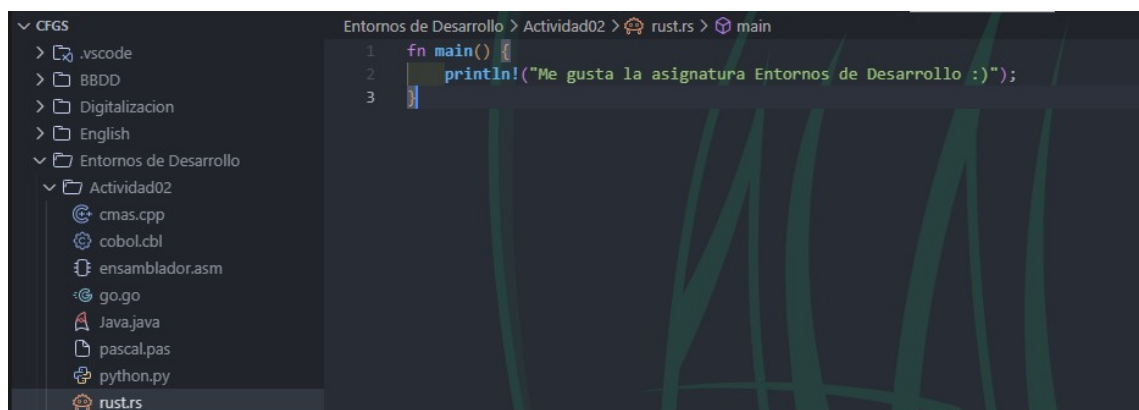
```
Entornos de Desarrollo > Actividad02 > Java.java
1 public class Main {
2     public static void main(String[] args) {
3         System.out.println("Me gusta la asignatura Entornos de Desarrollo :)");
4     }
5 }
```

2.4 COBOL

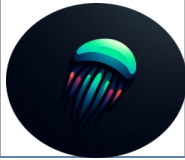


```
Entornos de Desarrollo > Actividad02 > cobol.cbl > PROGRAM-ID: ImprimirMensaje > PROCEDURE DIVISION
1 IDENTIFICATION DIVISION.
2 PROGRAM-ID. ImprimirMensaje.
3
4 DATA DIVISION.
5
6 PROCEDURE DIVISION.
7 DISPLAY "Me gusta la asignatura Entornos de Desarrollo :)".
8 STOP RUN.
9
```

2.5 Rust



```
Entornos de Desarrollo > Actividad02 > rust.rs > main
1 fn main() {
2     println!("Me gusta la asignatura Entornos de Desarrollo :)");
3 }
```

Practica II Entornos de Desarrollo

IES San Juan de la
Rambla

2.6 Pascal

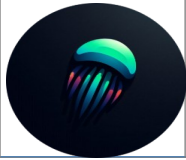
```
Entornos de Desarrollo > Actividad02 > pascal.pas
1  program ImprimirMensaje;
2
3  begin
4      writeln('Me gusta la asignatura Entornos de Desarrollo :');
5  end.
```

2.7 GO

```
Entornos de Desarrollo > Actividad02 > go.go
1  package main
2
3  import "fmt"
4
5  func main() {
6      fmt.Println("Me gusta la asignatura Entornos de Desarrollo :)")
7  }
```

2.8 Lenguaje ensamblador

```
Entornos de Desarrollo > Actividad02 > ensamblador.asm
1  section .data
2      msg db "Me gusta la asignatura Entornos de Desarrollo :)", 0xA ; Cadena de texto con un salto de línea
3      len equ $ - msg ; Longitud de la cadena
4
5  section .text
6      global _start
7
8      _start:
9          ; Escribir el mensaje
10         mov eax, 4 ; sys_write
11         mov ebx, 1 ; file descriptor 1 (salida estándar)
12         mov ecx, msg ; Dirección de la cadena a imprimir
13         mov edx, len ; Longitud de la cadena
14         int 0x80 ; Llamada al sistema
15
16         ; Salir del programa
17         mov eax, 1 ; sys_exit
18         xor ebx, ebx ; Código de salida 0
19         int 0x80 ; Llamada al sistema para salir
20
```

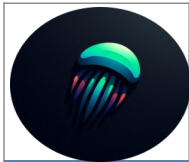



Practica II
Entornos de Desarrollo

**IES San Juan de la
Rambla**

3 ¿Qué extensión tienen los archivos de código fuente de estos lenguajes?

- **C++:** .cpp
- **Python:** .py
- **Java:** .java
- **Cobol:** .cbl
- **Rust:** .rs
- **Pascal:** .pa
- **Go:** .go
- **Ensamblador:** .asm



Practica II Entornos de Desarrollo

IES San Juan de la
Rambla

4 INCLUIR OTROS LENGUAJES

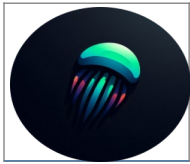
4.1 LUA

```
1 print("Me gusta la asignatura de ENTORNOS DE DESARROLLO :)")
```

El interprete de **LUA** en Linux, ironicamente, es **LUA**. Se instala con el siguiente comando: `sudo apt install lua5.3`. La extensión de archivo de **LUA** es **.lua**.

4.2 Haxe

```
1 class Main {
2     static public function main() {
3         trace("me gusta la asignatura de ENTORNOS DE DESARROLLO :");
4     }
5 }
```



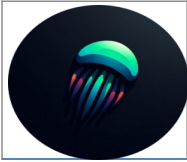
Practica II

Entornos de Desarrollo

IES San Juan de la
Rambla

Para instalar el interprete de **Haxe**, primero hay que agregar a nuestra máquina el repositorio oficial, con el comando `"sudo add-apt-repository ppa:haxe/releases"`. Hay que tener en cuenta que para instalar el interprete es recomendable tener actualizado el sistema. Ahora, se instala con `"sudo apt install haxe"`. Seguidamente se instala **Neko**, el interprete para terminal de **Haxe**, se instala de la siguiente manera `"sudo apt install neko"`.

Para ejecutar este código, hay que dirigirse a la terminal de **Linux**, y ejecutar el siguiente comando: `haxe --main ./haxe.hx -interp`.



5 COMPILAR UN CÓDIGO EN UBUNTU LINUX

```
marcos@marcos-VirtualBox: ~/Escritorio
marcos@marcos-VirtualBox:~/Escritorio$ mcs ./HolaUser.cs
```

Se compila un código adjuntado a la actividad en un archivo .exe, el cuál Linux nativamente no puede ejecutar.

```
marcos@marcos-VirtualBox: ~/Escritorio
marcos@marcos-VirtualBox:~/Escritorio$ mono HolaUser.exe
Hola qué tal amigo mio, ¿cómo te llamas?
```

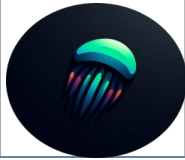
También se puede cambiar el código para cambiar el resultado:

```
marcos@marcos-VirtualBox: ~/Escritorio
marcos@marcos-VirtualBox:~/Escritorio$ mono HolaUser.exe
¿Qué tal amigo, estamos en la increíble clase de Entornos de Desarrollo, ¿cómo te llamas?
Gonzalo
Bienvenido Gonzalo que pases un bien día
```

Si se intenta ejecutar el programa sin compilarlo, sucede lo siguiente:

```
marcos@marcos-VirtualBox:~/Escritorio$ mono HolaUser.cs
Cannot open assembly 'HolaUser.cs': File does not contain a valid CIL image.
```

La solución es obvia y simple, con compilar el código antes de ejecutarlo es suficiente, en su defecto, se puede instalar un interprete como **G++** para poder ejecutarlo directamente desde la terminal sin necesidad de compilar.



Practica II

Entornos de Desarrollo

IES San Juan de la
Rambla

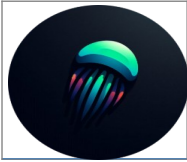
Vamos a realizar otro ejemplo, haciendo que se solicite también la edad del usuario:

```
HolaUser.cs x
home > marcos > Escritorio > HolaUser.cs
1  using System;
2
3  class HolaUser
4  {
5      static void Main(string[] args)
6      {
7          Console.WriteLine("¿Qué tal amigo? Estamos en la increíble clase de Entornos de Desarrollo. ¿Cómo te llamas?");
8          string nombre = Console.ReadLine();
9
10         Console.WriteLine("Bienvenido " + nombre + ", que pases un buen día.");
11
12         Console.WriteLine("Por cierto, ¿cuántos años tienes?");
13         string edad = Console.ReadLine();
14
15         Console.WriteLine("Tienes " + edad + " años. ¡Gracias por compartirlo, " + nombre + "!");
16     }
17 }
18
```

```
marcos@marcos-VirtualBox: ~/Escritorio$ mono HolaUser.exe
¿Qué tal amigo? Estamos en la increíble clase de Entornos de Desarrollo. ¿Cómo te llamas?
Gonzalo Tercero
Bienvenido Gonzalo Tercero, que pases un buen día.
Por cierto, ¿cuántos años tienes?
2
Tienes 2 años. ¡Gracias por compartirlo, Gonzalo Tercero!
```

En el caso de ejecutarlo en otra carpeta, es decir, compilándolo hacia otra carpeta, el resultado no cambia, ejecutándose normalmente como en los pasos anteriores.

```
marcos@marcos-VirtualBox: ~/Descargas$ mono HolaUser.exe
¿Qué tal amigo? Estamos en la increíble clase de Entornos de Desarrollo. ¿Cómo te llamas?
Kimberly
Bienvenido Kimberly, que pases un buen día.
Por cierto, ¿cuántos años tienes?
-3
Tienes -3 años. ¡Gracias por compartirlo, Kimberly!
```



6 PRUEBAS CON CÓDIGO EN PYTHON

Se hará un código parecido al realizado en **C#**, pero con el lenguaje interpretado **Python**.

```
• HolaUser.py  
~/Escritorio  
Abrir ▾   
  
nombre = input("¿Qué tal, cómo te llamas? ")  
  
print("Hola! " + nombre)
```

```
marcos@marcos-VirtualBox:~/Escritorio$ python3 HolaUser.py  
¿Qué tal, cómo te llamas? Elsa Pato  
Hola! Elsa Pato
```

Ahora, como en el apartado de **C#**, se le pedirá la edad al usuario.

```
• HolaUser.py  
~/Escritorio  
Abrir ▾   
  
nombre = input("¿Qué tal, cómo te llamas? ")  
  
edad = input("¿Cuántos años tienes? ")  
  
print("Hola, " + nombre + "! Tienes " + edad + " años.")|
```

```
marcos@marcos-VirtualBox:~/Escritorio$ python3 HolaUser.py  
¿Qué tal, cómo te llamas? Helen Chufe  
¿Cuántos años tienes? si  
Hola, Helen Chufe! Tienes si años.
```

La principal diferencia entre estos dos lenguajes es que en **C#** es necesario definir clases como **Main**, mientras que en **Python** se puede escribir el código directamente, lo cual es más rápido y sencillo para programas pequeños.

	<p align="center">Practica II Entornos de Desarrollo</p>	<p align="center">IES San Juan de la Rambla</p>
---	--	--

7 BIBLIOGRAFÍA Y REFERENCIAS

Título	Enlace
Logo	DALL·E 2 OpenAI. (s. f.). Recuperado 30 de septiembre de 2024, de https://openai.com/index/dall-e-2/
Wikipedia Srta. Rottenmeier	Señorita Rottenmeier. (2024). En <i>Wikipedia, la enciclopedia libre</i> . https://es.wikipedia.org/w/index.php?title=Se%C3%B1orita_Rottenmeier&oldid=160875880