



# pyarmando academy

```
for second matrix: ";  
  
// Enter elements of first matrix.  
cout << endl << "Enter elements of matrix 1:" << endl;  
for(i = 0; i < r1; ++i)  
{  
    for(j = 0; j < c1; ++j)  
    {  
        cout << "Enter element a" << i + 1 << " " << j + 1 << " ";  
        cin >> a[i][j];  
    }  
}  
  
// Enter elements of second matrix.  
cout << endl << "Enter elements of matrix 2:" << endl;  
for(i = 0; i < r2; ++i)  
{  
    for(j = 0; j < c2; ++j)  
    {  
        cout << "Enter element b" << i + 1 << " " << j + 1 << " ";  
        cin >> b[i][j];  
    }  
}
```

# Bienvenidos

Siga las siguientes indicaciones para que su experiencia sea provechosa:

1. Encuentre un espacio adecuado
2. Tome nota si es necesario para realizar las preguntas luego.
3. Interactúe en la Videoconferencia.
4. Tener una conexión por cable o quitar las conexiones adicionales durante la videoconferencia.
5. Siempre este atento.

# Quien Soy?

- Ing .Armando Ruiz Rebollar
  - Developer Señor & Certificado Profesional Python



- Python Developer



- Android Developer



- API Developer



- CTO Dream Music
- CEO Pyarmando Academy

<https://www.instagram.com/aprendepyarmando/?hl=es-la>

<https://www.youtube.com/c/ArmandoRuiz2021>

<https://twitter.com/PyarmandoA>

<https://web.facebook.com/aprendepyarmando>

# Objetivo del Curso

- Un lenguaje fácil e intuitivo tan poderoso como los de los principales competidores.
- El código que es tan comprensible como el inglés simple
- De código abierto, para que cualquiera pueda contribuir a su desarrollo.
- Adecuado para tareas cotidianas, permitiendo tiempos de desarrollo cortos

# Contenido del Curso



PYTHON



FLASK



MYSQL



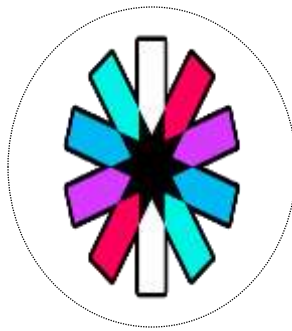
POSTGRESS



MONGO



API



JWT



POSTMAN



FIREBASE



HEROKU



VIRTUALBOX



TERMINAL

# Sesión 1

- Introducción a Python
- Instalación y configuración del entorno de desarrollo
- Aprendiendo la sintaxis del lenguaje:
  - Variables , Operadores , Tipos de Datos, Print, Input
- Estructuras de Control:
  - Condicionales , Bucles
- Ejercicios de Algoritmia con PYTHON

# Introducción a Python

- Python es un lenguaje de programación de alto nivel, interpretado, orientado a objetos y de uso generalizado con semántica dinámica, que se utiliza para la programación de propósito general.

# Características de PYTHON

- Multiplataforma
- Multiparadigma
- Código Abierto
- Interpretado
- Excelente Librería
- Totalmente Escalable



# En que industria puedo aplicar PYTHON?



# Hay Mas de un Python?

- PYTHON 3

- Es la versión más nueva (para ser precisos, la actual) del lenguaje. Está atravesando su propio camino de evolución, creando sus propios estándares y hábitos.
- Las Nuevas librerías de AI , y ciencia de datos generalmente estan desarrolladas en esta version.
- Si se va a comenzar un nuevo proyecto de Python, deberías usar Python 3, esta es la versión de Python que se usará durante este curso.

- PYTHON 2

- Su desarrollo se ha estancado intencionalmente, aunque eso no significa que no haya actualizaciones
- La ruta de desarrollo de Python 2 ya ha llegado a un callejón sin salida, pero Python 2 en sí todavía está muy vivo.

# Python es Escalable?

- Developer PYTHON
- Developer Web
- Cientifico de Datos
- Especialista en Inteligencia Artificial

# Que se Puede hacer con PYTHON



Aplicaciones Escritorio



Aplicaciones Web



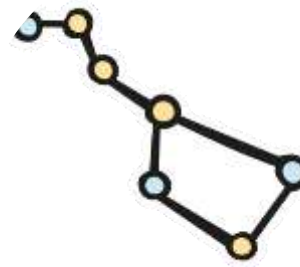
APIS



Aplicaciones Móviles



Inteligencia Artificial



Ciencia de Datos



Scraping

# ¿Quiénes usan Python?



<https://www.python.org/about/success/>

[https://en.wikipedia.org/wiki/List\\_of\\_Python\\_software](https://en.wikipedia.org/wiki/List_of_Python_software)

# Algunos conceptos Importantes para empezar

- **Programa Fuente**

- Es una serie de instrucciones lógicas que se le da al computador para que cumpla un propósito por ejemplo hacer un factura electrónica

- **Programa Objeto**

- Es un programa fuente que es traducido a lenguaje de maquina por un compilador . para que el computador lo pueda entender.

- **Compilador.**

- Son traductores que convierten un programa fuente a objeto , además el compilador espera hasta terminar la compilación de todo el programa para generar un informe de errores.

- **Interprete.**

- traduce instrucciones de alto nivel y lo ejecuta directamente. detecta si el programa tiene errores y permite su corrección durante el proceso de ejecución

- **Sintaxis.**

- Son las reglas del lenguaje y deben ser obedecidos.

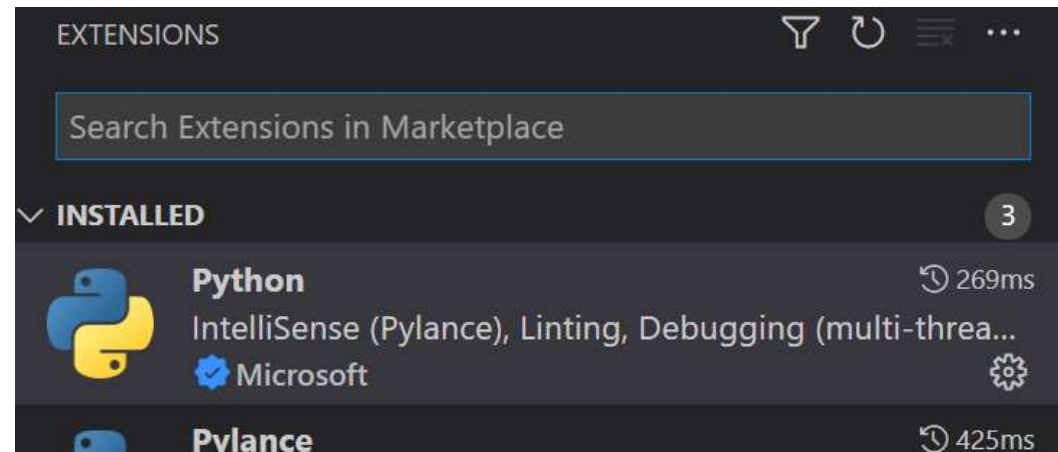
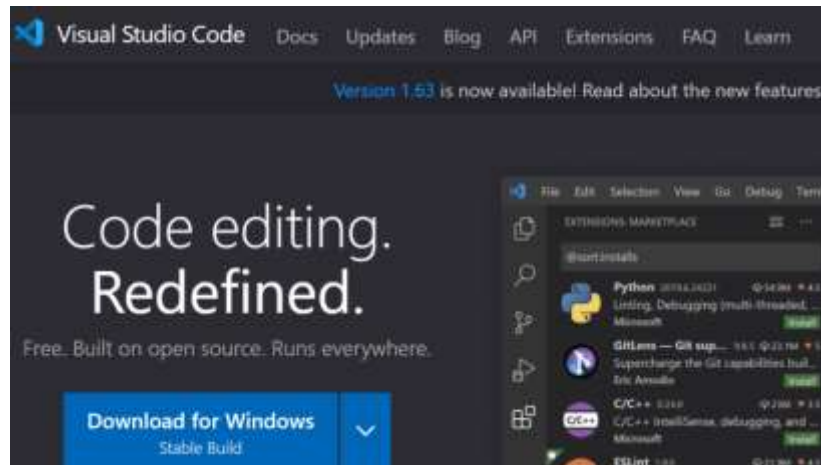
# Instalación y configuración del entorno de desarrollo

- Instalar el Interpretador de PYTHON:
  - <https://www.python.org/downloads/>





- Instalar Visual Studio Code
  - <https://code.visualstudio.com/>
- Extensiones Python



```
holamundo.py
1 print("Hola Mundo")
```

# Variables

- Las variables almacenan datos en la memoria de una computadora
- Reglas de nombres de variables de Python
  - Un nombre de variable debe comenzar con una letra o el carácter de subrayado
  - Un nombre de variable no puede comenzar con un número
  - Un nombre de variable solo puede contener caracteres alfanuméricos y guiones bajos (Az, 0-9 y \_)
  - Los nombres de las variables distinguen entre mayúsculas y minúsculas (firstname, Firstname, FirstName y FIRSTNAME) son variables diferentes)

- Usaremos el estilo estándar de nomenclatura de variables de Python que ha sido adoptado por muchos desarrolladores de Python
- (snake\_case). Usamos un carácter de subrayado después de cada palabra para una variable que contiene más de una palabra

```
first_name = 'Armando'
last_name = 'Ruiz'
country = 'Perú'
city = 'Lima'
age = 46
is_married = True
skills = ['HTML', 'CSS', 'JS', 'React', 'Python']
person_info = {
    'firstname': 'Armando',
    'lastname': 'Ruiz',
    'country': 'Perú',
    'city': 'Lima'
}
```

# Tipo de datos

- En Python existen a comparación con otros lenguajes pocos tipos de datos llamados primitivos. Para identificar el tipo de datos, usamos la función incorporada de *type* .
  - Numéricos(int,float)
  - Cadena(str)
  - Lógico(bool)
- **Cast:** convertir un tipo de datos a otro tipo de datos. Usamos `int()` , `float()` , `str()` , `list` , `set`

```
edad = 28
# Variable Cadena
nombre = "Armando Ruiz"
# Variable Decimal
precio = 20.8
# Variable Logica
Estado = True
# Funcion type me devuelve el tipo de dato que tiene la variable
print(type(edad))
print(type(nombre))
print(type(precio))
print(type(Estado))
```

# Condicionales

- En Python y otros lenguajes de programación, la palabra clave *if* se usa para verificar si una condición es verdadera y ejecutar el código de bloque. Recuerde la sangría después de los dos puntos.
- Si la condición es verdadera, se ejecutará el primer bloque, si no, se ejecutará la otra condición.
- Recuerde la sangria(indentation) despues de los dos puntos.

```
edad = int(input("Ingresa la edad de la persona:"))
if edad > 18:
    print("La Persona es mayor edad")
else:
    print("La Persona es menor de edad")
```

# Bucles

- Una palabra clave *for* se usa para hacer un Loop y se usa para iterar sobre una secuencia (es decir, una lista, una tupla, un diccionario, un conjunto o una cadena etc)

```
for num in range(1,20):  
    print(num,"Hola Soy Un Bucle!!")
```

[www.pyarmando.com](http://www.pyarmando.com)