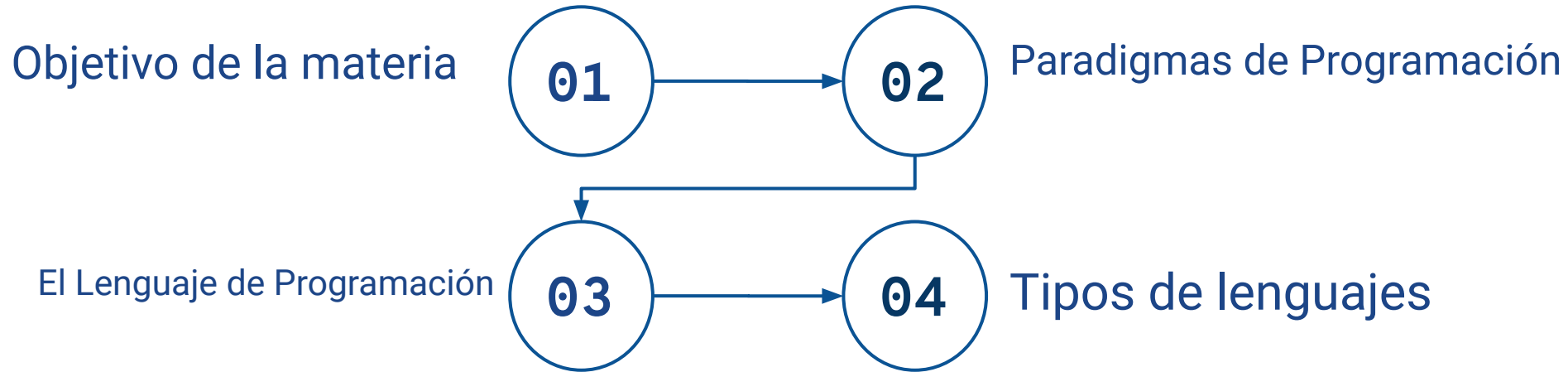


# Introducción a Materia

## Programación I

Versión '24

# Introducción a la Materia



# Objetivo de la materia

Nuestra materia tiene como principal objetivo obtener los conocimientos fundamentales de la programación , pudiendo desarrollar algoritmos independientemente de la tecnología, usando en nuestro caso el lenguaje Python.

**Advertimos de antemano que esto no es un curso de python.**

# Objetivos de la materia

## También tenemos los siguientes objetivos

- Aprender a programar en Python y conocer sus principales características y aplicaciones.
- Conocer y aplicar los conceptos fundamentales de la programación, como variables, tipos de datos, estructuras de control, funciones y programación orientada a objetos.
- Familiarizarse con las bibliotecas y herramientas más utilizadas en Python para la resolución de problemas.

# Objetivos de la materia

- Desarrollar habilidades en la creación de algoritmos y la resolución de problemas utilizando la programación en Python.
- Desarrollar un proyecto práctico utilizando las herramientas y técnicas aprendidas en el curso.
- Fomentar el trabajo en equipo y la colaboración en la resolución de problemas y proyectos prácticos.

# Paradigmas de programación

Dentro de la programación tenemos varios tipos de **paradigmas**

Se conoce como **paradigma** conjunto de valores, técnicas y procedimientos compartidos por una “comunidad científica” en un “Momento Histórico determinado”, para definir problemas y buscar soluciones.

En resumen se puede decir que un paradigma es un grupo de ideas sobre cómo hacer o pensar algo.

# Paradigmas de programación

## -Estructurado

*perspectiva algorítmica*

*enfatisa los procesos (funciones) del sistema*

## -Funcional

*perspectiva declarativa/matemática (Calculo lambda)*

*enfatisa más a la lectura literal (No existen bucles)*

## -Orientado a objetos

*perspectiva orientada a objetos*

*enfatisa los objetos que forman el sistema*

## -Estructurado

```
def sumar(numero_uno,numero_dos): #Entradas  
    suma = numero_uno + numero_dos #Procesos  
    return suma #Salidas
```

## -Funcional

```
lambda numero_uno, numero_dos : numero_uno + numero_dos #Calculo lambda
```



## -Orientado a objetos

```
class Calculadora: #Clase

    numero_uno = 0 #Atributo

    numero_dos = 0 #Atributo

    resultado = 0 #Atributo


    def sumar(self, numero_uno, numero_dos): #Metodo (Comportamiento)

        suma = numero_uno + numero_dos

        return suma
```

# El lenguaje de programación

Primero tengamos en cuenta la definición de **programa**

-Un programa informático es un conjunto de instrucciones que una vez ejecutadas realizarán una o varias tareas en una computadora. Sin programas, estas máquinas no pueden funcionar.

Nosotros como programadores tenemos como objetivos construir **programas** con un **paradigma** específico, usando un **lenguaje de programación** que nos va a permitir construirlo.

# Tipos de Lenguajes

Al igual que existen lenguajes para comunicarnos, existen lenguajes de programación para construir programas, estos mismos se clasifican en lenguajes de **Alto nivel** y **Bajo nivel**

**Alto nivel:** Son lenguajes en los que su sintaxis es más flexible para el programador , evitando a el mismo de tener que realizar acciones referidas a la comunicación hardware-software (Reservar espacio en memoria en la RAM), también me sirve para sintetizar una comunicación humano-maquina más efectiva y menos complicada

Como ejemplos de lenguajes de alto nivel tenemos a **Javascript, Python, C#, Java**, y cualquier lenguaje de programación actual.

# Tipos de Lenguajes

**Bajo nivel:** La comunicación se ejerce directamente sobre el hardware (pudiendo modificar libremente la asignación de memoria), son lenguajes por lo general más complicados, ya que su sintaxis no es tan flexible y entendible para el humano y más directa al lenguaje de maquina)

Algunos ejemplos de lenguajes de bajo nivel son **Lenguaje maquina, Asembler.**

También hay lenguajes de bajo nivel (comunicación directa con el hardware) pero con una sintaxis más amigable para el programador como por ejemplo: **C (Padre de todos los lenguajes actuales), Basic , COBOL.** A estos lenguajes también se los conoce como **Lenguajes de Medio Nivel.**

# Tipos de Lenguajes

Aparte de clasificar a los lenguajes por lenguajes de **Alto o Bajo Nivel**

También los lenguajes se clasifican en lenguajes **Compilados** y lenguajes **Interpretados**

**Lenguajes Compilados:** Son los lenguajes cuya implementación es convertida directamente a lenguaje maquina (código binario) mediante un Compilador (**MinGW**). El código que yo escribi se compila completamente, por lo que si hay un error en mi código el programa no podra ejecutarse.

Algunos ejemplos de lenguajes compilados pueden ser **C, C#, C++,Java, Go,BASIC**, entre otros.

# Tipos de Lenguajes

**Lenguajes Interpretados:** A diferencia de los lenguajes compilados, en los que se tiene que traducir y compilar todo mi código completo para poder ejecutarlo, los lenguajes interpretes no se compilan, sino que son traducidos por un segundo programa (Por ejemplo el navegador chrome) en el que se encarga de leer línea por línea de mi código traduciendo e interpretando en tiempo real.

Algunos ejemplos de estos pueden ser **Javascript, Python, PHP, Ruby**, entre otros.