

# Estructura de Datos y POO

Semana 3





## Contenido

- Programación Orientada a Objetos

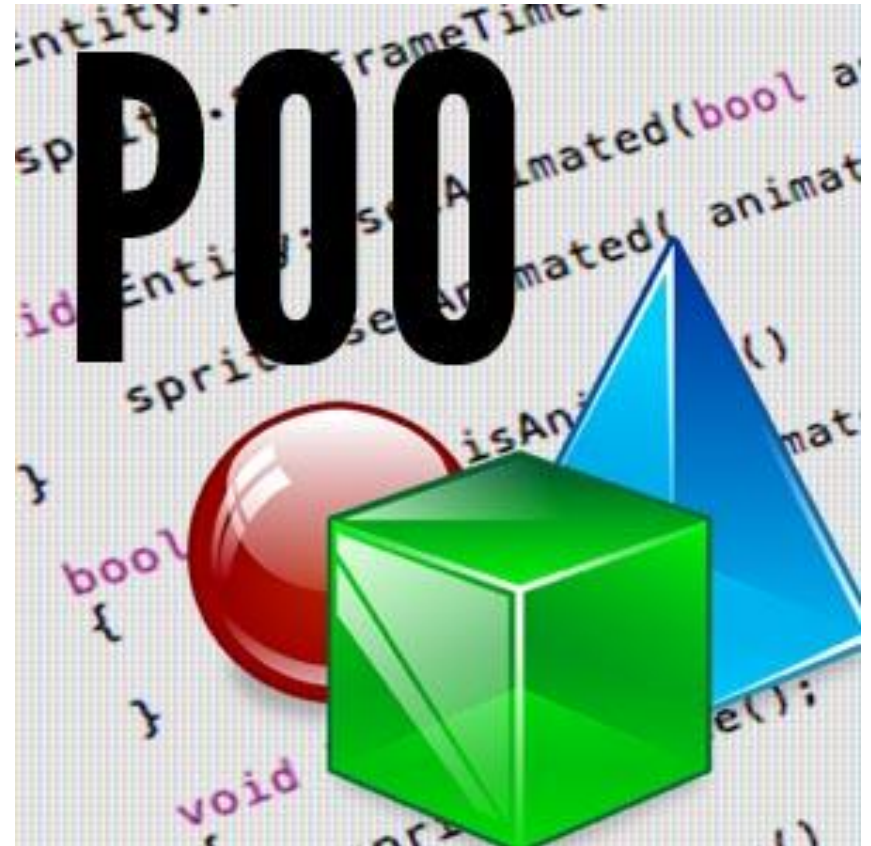


**PYTHON**



## Programación Orientada a Objetos

- ✚ La Programación Orientada a Objetos (**POO**) es un paradigma de programación que usa objetos y sus interacciones para diseñar aplicaciones y programas de computadora.
- ✚ La programación Orientada a objetos (**POO**) es una forma especial de programar, más cercana a como se expresan las cosas en la vida real que otros tipos de programación.



# Introducción a POO



## Ventajas de la POO

- + Fomenta la reutilización y ampliación del código.
- + Permite crear sistemas más complejos.
- + Agiliza el desarrollo de software.
- + Facilita el trabajo en equipo.



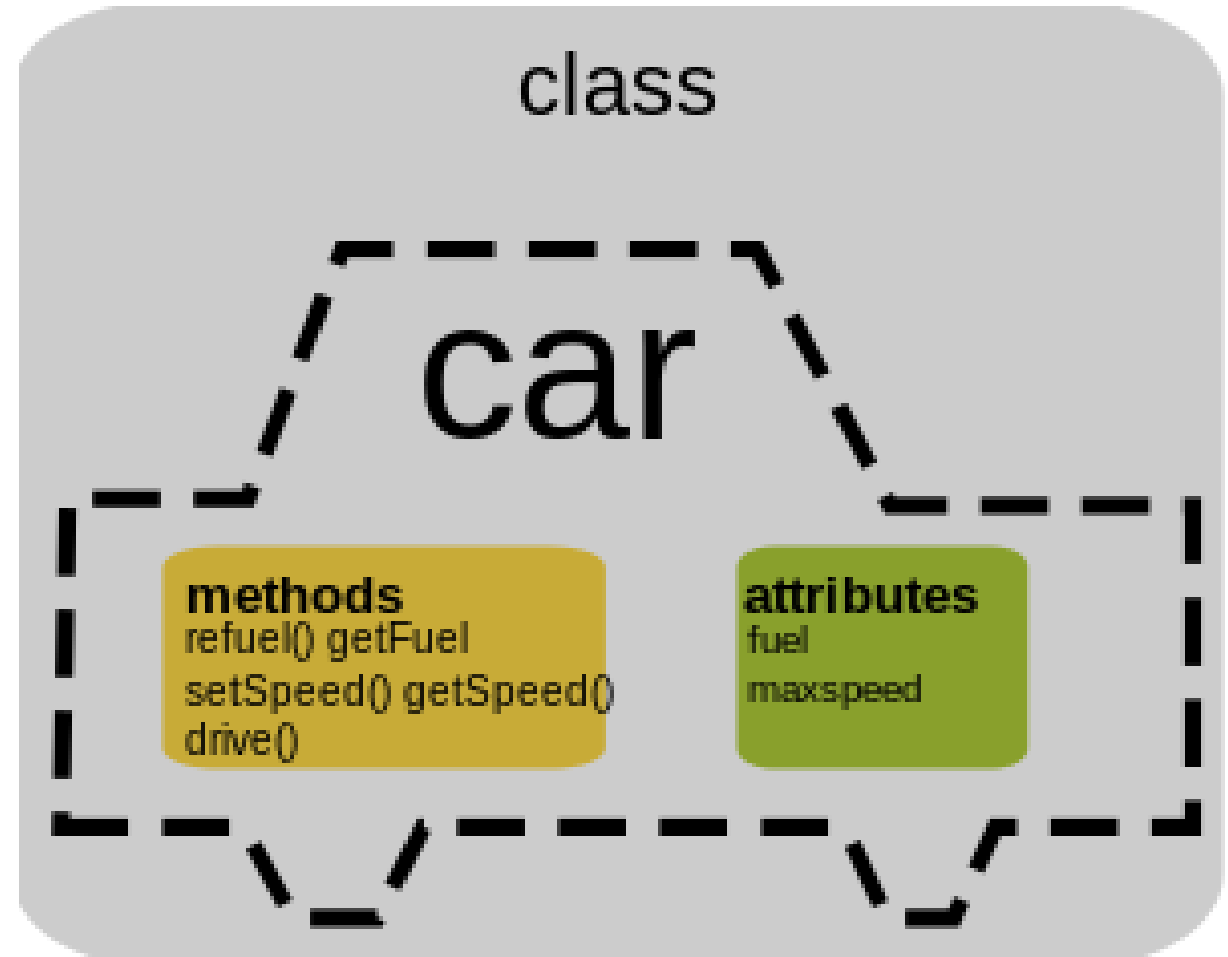
# Introducción a POO



## Conceptos de la POO

✚ Conceptos fundamentales que sustentan la POO:

- Clase
- Objeto
- Atributos
- Métodos



# Introducción a POO



## Clase

- Una clase es como un tipo de dato creado por el usuario, que posee dos categorías de miembros:

- **Atributos**

- **Métodos**

- En general, es posible crear una clase a partir de cualquier objeto que esté a nuestro alrededor. Por ejemplo:

- Persona
- Automóvil
- Mascota



# Introducción a POO



## Objetos (instancias)

- Una **instancia** es una ocurrencia de la **clase**
- Al momento de crear un objeto se produce la instanciación
- Un **objeto** es una instancia de una **clase** específica

### Ejemplo:

Supongamos que existe la clase **Vehículo**

- El auto con placa BAW-461 es una instancia de la clase **Vehículo**, es decir, es un objeto de esa clase
- La camioneta con placa ZTU-983 es otra instancia de la clase **Vehículo**

### CLASES

- Automóviles (cuatro puertas o cupés)
- 4X2 Y 4X4
- Utilitarios
- Utilitarios deportivos
- Vans de PASAJEROS
- ANTIGUOS
- TUNING







## Atributos

- Son los datos que caracterizan a los objetos de una clase

- Marca
- Año
- Color
- Placa, etc.





# Introducción a POO



## Métodos

- Representan todas aquellas **acciones** que se pueden realizar **sobre un objeto** de cierta clase
- En la implementación, estos métodos son segmentos de código en la forma de funciones
- La clase **Vehículo** puede incluir los métodos:
  - Encender
  - Acelerar
  - Girar
  - Frenar



# Introducción a POO



## Clases y objetos en Python

Para crear una clase se hace escribiendo la palabra **class** seguida del nombre de la clase y un par de paréntesis. Luego de creada la clase se deben definir sus **atributos** y **métodos**.

Para diferenciar un **método** de una función lo hacemos escribiendo dentro de sus paréntesis el parámetro **self**.

Toda clase debe definir el método **\_\_init\_\_()** (método constructor que le da el estado inicial a una clase).

Cuando la clase está lista ya se puede empezar a crear objetos que pertenezcan a esa clase.

```
class Vehiculo():  
    def __init__(self, ruedas):  
        self.__ruedas=ruedas  
  
    def numeroRuedas(self):  
        print("Nro ruedas: {}".format(self.__ruedas))
```

```
objVehiculo = Vehiculo(4)  
objVehiculo.numeroRuedas()
```



# Gracias