



Universidad Autónoma del Estado de México

Facultad de Geografía



Programación con Python

Orientado a la automatización de procesos en QGIS

Ing. Adonai Emmanuel
Nicanor Bautista

Introducción a la Programación Orientada a Objetos (POO)

- La Programación Orientada a Objetos es un paradigma de programación y Python es un lenguaje multiparadigma.
- **Clase:** Es un tipo de *"plantilla"* que permite crear diferentes objetos con ciertas características en común. La creación de un objeto se llama **Instancia**.
- **Objetos:** Son un tipo de dato el cual contiene atributos y métodos.
- **Atributos (Clase o instancia):** Pueden ser cualquier otro tipo de datos como números, strings, etc. Además hay tres tipos: **Públicos**, **protegidos** (`_`) y **privados** (`__`).

Introducción a la Programación Orientada a Objetos (POO)

- **Métodos:** Son funciones interactúan con los objetos y los atributos que contiene, permitiendo generar nueva información.
- Las clases se crean con la palabra reservada **class** seguido del nombre de la clase, la cual debe iniciar con **Mayúscula**.
- El Método **Constructor** sirve para inicializar valores cuando la clase ha sido instanciada, así como realizar los procesos y operaciones necesarias para generar objetos, este método puede recibir o no valores. Este método siempre sera definido como **`__init__`**.
- La palabra reservada **self** permite tener acceso a los atributos y métodos dentro de una clase.
- Con **del** podemos eliminar una instancia.

Introducción a la Programación Orientada a Objetos (POO)

Palabra reservada y
nombre de la clase

Atributos de clase

Método constructor

Métodos

```
1 class Example:
2     """
3     Esta es un clase de ejemplo
4     """
5     saludo = "Hola a todos"
6
7     def __init__(self):
8         self.num_alumnos = 25
9         self._protegido = 15
10        self.__privado = 40
11
12        def imprime_saludo(self):
13            print(self.saludo)
14
15        def imprime_alumnos(self):
16            print(self.num_alumnos)
```

Docstring
(Documentación)

Atributos de instancia
Públicos
Protegidos
Privados

Introducción a la Programación Orientada a Objetos (POO)

- La **Herencia (Extend)** sirve para aprovechar el máximo código posible en objetos con estructuras similares. Como su nombre lo indica, hereda propiedades de una clase a otra.
- La clase que brinda estas propiedades se denomina **Clase Padre**.
- La nueva clase (**Clase Hija**) puede incluir atributos y métodos adicionales a los que le han sido heredados.
- Para acceder a los atributos de la clase padre, es necesario usar la palabra reservada **super**.
- También existe la herencia múltiple, es decir, una clase hija puede incluir distintas clases padre, dando prioridad de izquierda a derecha según sean declaradas.

Introducción a la Programación Orientada a Objetos (POO)

```
1 class Example:
2     saludo = "Hola a todos"
3
4     def __init__(self):
5         self.num_alumnos = 25
6
7     def imprime_saludo(self):
8         print(self.saludo)
9
10    def imprime_alumnos(self):
11        print(self.num_alumnos)
```

```
class Curso(Example):
    def __init__(self):
        super().__init__()
    def nom_curso(self):
        self.n_curso = "Programación con Python"
    def imprime_curso(self):
        print(self.n_curso)
```