



CICLO 01

[FORMACIÓN POR CICLOS]
Fundamentos de
Programación

Clases derivadas, herencia,
polimorfismo



Programación orientada a objetos

Clases derivadas

Herencia

La programación orientada a objetos permite definir nuevas clases con base en clases previamente definidas.

La clase original se denomina “clase base” o “superclase”.

La nueva clase se denomina “clase derivada” o “subclase”.

La principal ventaja es que los objetos pertenecientes a la clase derivada pueden usar los métodos de la clase base. Esta propiedad hace que el desarrollo de la clase derivada sea más eficiente:

- Evita la duplicidad de métodos
- Propicia la reutilización de código
- Facilita el mantenimiento
- Facilita la creación de nuevas aplicaciones

Esa propiedad de que los objetos de la clase derivada puedan utilizar los métodos definidos en la clase base se denomina **herencia**.

Adicionalmente diremos que también se definen clases por composición.

Una clase puede definirse como un conjunto de clases definidas previamente.

Por ejemplo, la clase vehículo se puede definir como un composición de la clase motor, la clase carburador, la clase llanta, etc.

Es importante entonces diferenciar entre lo que es una clase derivada de una clase por composición.

En nuestro curso nos ocuparemos de las clases derivadas

En general, si se tiene una clase base cualquiera y se quiere definir una clase derivada, la forma general en Python es:

```
class derivada(claseBase):  
    claseBase.__init__(self, n)  
    otrasInstrucciones
```

Y así se estaría definiendo la clase derivada y su constructor.

Si se requieren operaciones adicionales en el constructor de la clase derivada, se escriben en el sitio donde está **otrasInstrucciones**.

otrasInstrucciones es opcional.

En nuestro curso definiremos una clase derivada de la clase vector, la cual llamaremos *altaPrecision*.

En esta clase representaremos números enteros con cualquier cantidad de dígitos. Es decir, enteros con 5000 dígitos, 15000 dígitos, etc.

En cada casilla del vector se representa un dígito.

Todo objeto de la clase *altaPrecision* podrá hacer uso de los métodos definidos en la clase vector.

Se definen los métodos correspondientes a las operaciones que se efectúan con enteros y que no se ejecutan normalmente sobre vectores. Es decir, métodos exclusivos de la clase derivada.

Adicionalmente, se podrán definir métodos con el mismo nombre que tienen en la clase base, dando lugar a una propiedad de la programación orientada a objetos, la cual es el polimorfismo dinámico.

Se presentará también otra cualidad importante, que es la sobrecarga de operadores, lo que posibilita elaborar algoritmos más elegantes.

Y, por último, presentaremos el concepto de variable global y su aplicación.