

Атрибути класів

Якщо викликати список атрибутів об'єкта, то можна побачити наступне:

```
['__class__', '__delattr__', '__dict__', '__dir__', '__doc__', '__eq__',  
'__format__', '__ge__', '__getattribute__', '__gt__', '__hash__', '__init__',  
'__init_subclass__', '__le__', '__lt__', '__module__', '__ne__', '__new__',  
'__reduce__', '__reduce_ex__', '__repr__', '__setattr__', '__sizeof__', '__str__',  
'__subclasshook__', '__weakref__', 'make', 'model', 'name', 'start', 'stop']
```

Атрибути можуть бути віднесені до *атрибутів класа* та *атрибутів екземпляра*.

Атрибути класу діляться серед усіх об'єктів класу, *атрибути екземплярів* є власністю екземпляра. *Атрибути екземпляра* оголошуються всередині будь-якого методу (функції класу), тоді як *атрибути класу* оголошуються поза будь-яким методом.

Одним з найбільше використовуваних атрибутів є `__init__`. Це є конструктором полів екземпляру. Він записується аналогічно функції класу, але спрацьовує при створенні нового об'єкту.

```
class Person:  
    def set_name(self, n, s):  
        self.name = n  
        self.surname = s
```

Таким чином можна встановити окремі поля одразу при створенні нового об'єкту. У такому випадку треба створювати екземпляри з вказанням значень полів у якості параметрів.

```
>>> p1 = Person()  
>>> p1.set_name("Bill", "Ross")  
>>> p1.name, p1.surname  
( 'Bill', 'Ross')
```

Можна встановити параметри по замовчуванню. Для цього можна одразу встановити значення полів.

```
class Rectangle:  
    def __init__(self, w=0.5, h=1):  
        self.width = w  
        self.height = h
```

Також існує атрибут `__del__`. Він виконується при видаленні об'єкта. Немає сенсу вводити туди нові поля, але є сенс змінювати поля класа, виводити написи або вертати повідомлення.

`__call__` – автоматично викликається при зверненні до об'єкта як до функції.

Також є атрибути перегрузки. *Перегрузка функції* – це заміна дії інтерфейсу для окремого класу або групи класів.

`__add__` – застосовується для заміни інтерфейса додавання (“+”);

`__sub__` – заміна інтерфейса віднімання (“-”);

`__mul__` – заміна множення (“*”);

`__truediv__` – заміна ділення.

Також є наступні атрибути об'єктів:

- `__class__` – вертає назву класу об'єкта.
- `__doc__` – вертає документацію класу.
- `__getattr__` – вертає назву атрибуту у вигляді рядка.
- `__str__` – вертає об'єкт у строковому вигляді.

Контрольні питання:

- 1) Що таке конструктор класу?
- 2) Коли викликається атрибут `__init__`? `__del__`? `__call__`?
- 3) Як писати атрибути в клас?
- 4) Що таке перегрузка функції? Навести приклад.