

Python. Programowanie obiektowe. Dziedziczenie klas

Dziedziczenie to proces, w którym jedna klasa przejmuje atrybuty i metody innej. Nowo utworzone klasy nazywane są klasami potomnymi, a klasy, z których pochodzą klasy potomne, nazywane są klasami nadrzędnymi.

Oryginalna klasa nazywa się klasą bazową lub superklasą, zaś nowa nazywana jest klasą pochodną lub podklasą.

Klasy potomne mogą przesłonić lub rozszerzyć atrybuty i metody klas nadrzędnych. Innymi słowy, klasy potomne dziedziczą wszystkie atrybuty i metody rodzica, ale mogą również określać atrybuty i metody, które są dla siebie unikalne.

Przykład:

```
class bird:
```

```
    def __init__(self, gatunek, szybkość):
        self.gatunek = gatunek
        self.szybkość = szybkość
```

```
    def fly(self):
        print ("Tu", self.gatunek, ". Lecimy i osiągniemy prędkość", self.szybkość)
```

```
class eagle (bird):
```

```
    def __init__(self, szybkość):
        super().__init__("orzeł", szybkość)
```

```
    def attack(self):
        print ("Tu", self.gatunek, ". Do atakuuuuu!")
```

```
class penguin (bird):
```

```
    def __init__(self, szybkość):
        super().__init__("pingwin", szybkość)
```

```
    def slide(self):
        print ("Tu", self.gatunek, ". Bziuuuuuum")
```

```
    def fly(self):
        print ("Tu", self.gatunek, ". Nie potrafię latać :((")
```

```
orzel = eagle(90)
```

```
orzel.fly()
```

```
orzel.attack()
```

```
pingwin = penguin(50)  
pingwin.slide()  
pingwin.fly()
```

Zdefiniowaliśmy klasę 'bird', tak jak to robiliśmy w przypadku każdej innej klasy oraz klasę eagle oraz penguin. Po nazwie, w nawiasach, umieściliśmy informację, że klasy te dziedziczą od klasy 'bird'.

Ponieważ pingwiny nie latają musimy nadpisać metodę fly w klasie penguin.

Funkcja super()

Wywołane konstruktory klas potomnych, inicjuje konstruktor klasy od której dziedziczą, za pomocą funkcji super ()

```
def __init__(self, szybkość):  
    super().__init__("orzeł", szybkość)
```

Dzięki tej funkcji eliminujemy konieczność podawania nazwy ptaka, która powinna być oczywista.

Lista zadań:

1. Napisz klasę Pojazd, która będzie klasą bazową oraz dwie podklasy : samochód oraz motocykl. Utwórz prosty konstruktor oraz destruktor. Podklasy będą dziedziczyły puste pole : nr_tablica odpowiadający za przypisanie tablicy rejestracyjnej. Destruktor będzie odpowiadał za usunięcie obiektów z pamięci procesora po wykonaniu programu. Dodaj metody, które według Ciebie pozwolą w pełni pokazać dziedziczenie.
2. Napisz program, w którym utworzysz klasę podstawową : Zwierzę oraz klasy potomne: kot, pies, ptak, ryba. Zaproponuj odpowiednie metody, pola i dziedziczenie.
3. Napisz program, w którym utworzysz klasę podstawową Figura oraz klasy potomne: kwadrat, prostokąt, trójkąt, koło, romb, trapez. Zaproponuj odpowiednie metody, pola i dziedziczenie. Program powinien obliczać obwód i pole stworzonego obiektu na podstawie wprowadzonych przez użytkownika danych.