

# Třídy a objekty

```
class Zviratko:
```

```
    def __init__(self, jmeno):  
        self.jmeno = jmeno
```

```
    def snez(self, jidlo):  
        print(self.jmeno, 'jí', jidlo)
```

```
class Kotatko(Zviratko):
```

```
    def zamnoukej(self):  
        print(self.jmeno, 'mňau!')
```

```
class Prasatko(Zviratko):
```

```
    def zakvic(self):  
        print(self.jmeno, 'kví!')
```

```
    def snez(self, jidlo):  
        self.zakvic()  
        super().snez(jidlo)  
        print(self.jmeno,  
              'se po jídle válí v bahně!')
```

Vlastní třída, neboli typ

Inicializace objektu, může přebírat argumenty  
Nastavení atributu objektu

Metoda – první argument je self

Čtení atributu objektu

Dědičnost (kotátko je druh zvířátka)

Metoda specifická pro kotátka

Dědičnost (prasátko je druh zvířátka)

Metoda specifická pro prasátka

Přepisování metody předka

Volání vlastní metody

Volání metody jako na předkovi



```
prasatko = Prasatko('Ťulda')
```

```
prasatko.snez('žalud')
```

```
karlicka = Prasatko('Karlička')
```

```
print(karlicka.jmeno)
```

```
karlicka.jmeno = 'Karla'
```

```
mikes = Kotatko('Mikeš')
```

Vytvoření objektu (instance) třídy

Volání metody

Objektů jedné třídy může být víc

Můžeme se dívat na jejich atributy

A taky je měnit

```
for zviratko in [prasatko, karlicka, mikes, Kotatko('Mourek')]:  
    zviratko.snez('papání')
```

Nevíme jestli to je prasátko nebo kotátko,  
ale to nevadí, tohle funguje

## Polymorfismus

```
isinstance(zviratko, Zviratko)  
type(zviratko) == Zviratko
```

Je instancí zvířátka? Ano

Je třídou objektu Zviratko? Ne

## Typová kontrola