

OOP: konstruktor (výchozí, implicitní, obecný), výchozí parametry, metody, druhy dědičnosti

Výchozí konstruktor (implicitní konstruktor)

Každá třída v jazyce má výchozí konstruktor, který se volá při vytváření nové instance třídy, pokud není explicitně definován jiný konstruktor. V mnoha programovacích jazycích je výchozí konstruktor prázdný a nevyžaduje žádné parametry.

```
class Auto:
    def __init__(self):
        self.znacka = "Neznama znacka"
        self.model = "Neznamy model"
        self.rok_vyroby = 2022
```

Obecný konstruktor

Používá se k inicializaci objektu s určitými hodnotami. Tento konstruktor může mít různé parametry.

```
class Auto:
    def __init__(self, znacka, model, rok_vyroby):
        self.znacka = znacka
        self.model = model
        self.rok_vyroby = rok_vyroby
```

Výchozí parametry

Konstruktor může být volán bez určitých parametrů a použije se výchozí hodnota.

```
class Auto:
    def __init__(self, znacka = "skoda"):
        self.znacka = znacka
```

Metody

Metody jsou funkce, které jsou součástí třídy a provádějí operace na objektech této třídy. Tyto metody mohou být volány na instanci třídy a mohou manipulovat s atributy objektu nebo provádět jiné operace

Druhy dědičnosti

- **Jednoduchá dědičnost**

Třída může dědit pouze od jedné jiné třídy. Každá třída může mít pouze jednoho předka.

- **Víceúrovňová dědičnost**

Třída může dědit od jiné třídy, která sama dědí od jiné třídy, vytvářejíc tak řetězec dědičnosti.

- **Vícenásobná dědičnost**

Třída může dědit od více než jedné jiné třídy současně.