



Föreläsning 6 Programmeringsteknik och C DD1316

- Kommandotolkparametrar
- Klass
- Instans
- Metod
- Konstruktör



Parametrar från kommandotolken

Vid start av ett program kan man få in strängar till programmet genom att använda fördefinierade listan `sys.argv`.

datorn-> `python prg.py arg1 arg2...`



Exempel

```
import sys

print ("Du startade det här programmet med",
      len(sys.argv)-1, "parametrar" )
```

```
import sys

print ("Följande är en lista av
parametrar:")
for param in sys.argv:
    print (param)
```



Klass och instans (objekt)

- En *klass* är en mall för ett objekt
t.ex. bil, konto
- Ett *objekt* är en *instans* av en klass
t.ex. den röda bilen, mitt lönekonto



Klass och instans

Fördela följande till klasser och objekt

- Bok
- Boken Java från grunden
- Min kopia av Java från grunden
- Min hund
- Bilden av min hund medan han biter brevbäraren
- Bil
- En Chrysler



Definition av en klass

```
class Kurs:
    pass
```



Instansering

```
k1 = Kurs() # skapa instans av Kurs
k1.beteckning = "prge15"
print (k1.beteckning)

k2 = Kurs() # skapa ny instans av Kurs
k2.beteckning = "datae15"
print (k2.beteckning)
```



Metod

- Metoder har samma syntax som funktioner med den skillnaden att de definieras i en klass.

Två metodtyper:

- En *instansmetod* hör till instansen.
- En *klassmetod* hör till klassen (mer om detta i nästa föreläsning)



Konstruktormetod

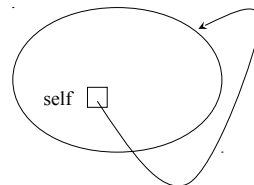
- En konstruktor är en metod som anropas då man skapar objekt av klassen.
- Konstruktorer är till för initiering av instansvariabler på ett smidigt sätt.
- En konstruktor definieras enligt nedan:

```
def __init__(self, namn, beteckning):
    self.namn = namn
    self.beteckning = beteckning
```



self

- Self är en referensvariabel som refererar inifrån instansen till själva instansen.



10/12



Skapa klass

```
class Kurs:
    def __init__(self, kod, beteckning):
        self.kurskod = kod
        self.beteckning = beteckning
        self.antal = 0

    def sattAntalStudenter (self, a):
        self.antal = a

    def geKod (self):
        return self.kurskod

    def geBeteckning (self):
        return self.beteckning

    def geAntal (self):
        return self.antal
```



Använda klassen

```
k1 = Kurs("DD1315", "prgi15")
k2 = Kurs("DD1316", "prge15")

print ("Antal studenter i kursen",
      k1.geBeteckning (),
      "är",
      k1.geAntal ())

print ("Kursen",
      k2.geBeteckning (),
      "har kurskoden",
      k2.geKod ())
```