

Skolan för Datavetenskap och kommunikation

PROGRAMMERINGSTEKNIK

FÖRELÄSNING 8

IDAG:

- ♥ Lista av objekt
- ♥ Nästlade for-slingor
- ♥ Interaktion mellan objekt
- ♥ Klassen Bur med Husdjur i lista

LISTA AV OBJEKT

- Flera objekt i samma program?

```
djur1 = Husdjur()
```

```
djur2 = Husdjur()
```

...

- Enklare att lägga husdjuren i en lista!



[0]

[1]

[2]

[3]

SKAPA LISTAN

```
lista = []  
for i in range(n):  
    nytt = Husdjur()  
    lista.append(nytt)
```

ANROPA METOD FÖR VARJE DJUR

```
for djur in lista:  
    djur.banna()
```

NÄSTLADE FOR-SLINGOR

Om man lägger en for-slinga inuti en annan säger vi att slingorna är nästlade. Den inre slingan går då igenom alla sina värden för varje värde i den yttre slingan. Exempel:

```
for i in range(3):  
    for j in range(3):  
        print(i,j)
```

```
0 0  
0 1  
0 2  
1 0  
1 1  
1 2  
2 0  
2 1  
2 2
```


SLUMPA HUSDJURSNAMN

```
from random import choice

def slumpnamn():
    namn = choice("BCFKR")+ \
           choice("iouy")+ \
           2*choice("nst")+ \
           choice("aey")
```

Bunny

Kotte

Russa

UTÖKAD HUSDJURSKLASS

```
def __init__(self):  
    """ Ger attributen slumpade värden """  
    self.namn = slumpnamn()  
    self.glad = randrange(10)  
    self.hunger = randrange(3)  
    self.kon = choice(("hona", "hane"))  
    self.preferens = choice(("samma", "annat"))
```


RITA UPP ETT OBJEKT

Husdjur-objekt:

namn

"Fussa"

glad

4

hunger

5

kön

"hona"

preferens

"samma"

NY KLASS: BUR

```
class Bur(object):
```

```
    def __init__(self, n):
```

```
        """Skapar en lista med n Husdjur"""
```

```
        self.lista = []
```

```
        for i in range(n):
```

```
            self.lista.append(Husdjur())
```

FLER METODER I KLASSEN BUR

```
def banna(self) :  
    for djur in self.lista:  
        djur.banna()
```

```
def mata(self, bullar) :  
    for djur in self.lista:  
        djur.mata(bullar)
```

```
def leka(self) :  
    for djur in self.lista:  
        djur.leka()
```

KLASSEN BUR

Attribut	Metoder
lista	__init__() banna() mata() leka() mingel() visa() avsked()

KLASSEN HUSDJUR

Attribut	Metoder
namn	<code>__init__()</code>
glad	<code>__str__()</code>
hunger	<code>banna()</code>
kon	<code>mata()</code>
preferens	<code>leka()</code>
	<code>avsked()</code>
	<code>kontakt()</code>

INTERAKTION

♥ Vi definierar en kontakt-metod som returnerar True om djur1 och djur2 får kontakt:

```
if djur1.kontakt(djur2):
```

♥ Tänk *kontakt(djur1, djur2)*




```
def kontakt(self, kompis):
    """Testar om kontakt uppstår mellan
       detta husdjur (self) och kompis"""
    if (self.kön == kompis.kön):
        if (self.preferens == "samma") and\
            (kompis.preferens == "samma"):
            print("Puss!")
            return True
    elif (self.kön != kompis.kön):
        if (self.preferens == "annat") and\
            (kompis.preferens == "annat"):
            print("Puss!")
            return True
    else :
        return False
```

MINGEL

```
n = len(lista)
for i in range(n-1):
    jag = lista[i]
    for j in range(i+1,n):
        du = lista[j]
        if jag.kontakt(du):
            lista.append(Husdjur(jag.namn, du.namn))
```

djur 0 & djur 1	djur 0 & djur 2	djur 0 & djur 3
djur 1 & djur 2	djur 1 & djur 3	
djur 2 & djur 3		