

DD1310/DD1314/DA3009
PROGRAMMERINGSTEKNIK
Föreläsning 4

DATASTRUKTURER

Kap 4-5 i Dawson

• Strängar

'N' 'O' 'U' 'G' 'A' 'T'

• Listor

'blå' 'grön' 'röd' 'gul'

• Tupler

'blå' 'grön' 'röd' 'gul'

• Dictionary

'blå'	'grön'	'röd'	'gul'
'bleu'	'vert'	'rouge'	'jaune'

GEMENSAMT

- Lagrar flera värden
- Använd for-slinga för att gå igenom alla värden
- Har index
- Har inbyggda metoder

FOR-SLINGAN I REPRIS

Exempel	Vad blir i?
<pre>for i in range(5): print(i)</pre>	Heltal
<pre>for i in "abcde": print(i)</pre>	Bokstav
<pre>for i in ["do", "re", "mi"]: print(i)</pre>	Sträng

METODANROP

Exempel:

```
mening=input("Skriv en förolämpning: ")  
print("Så här ser den ut med versaler:")  
print(mening.upper())
```

*metod = method
anropa = invoke, call*

STRÄNGMETODER

Metod	Beskrivning
<code>upper()</code>	Returnerar versaler
<code>lower()</code>	...gemena
<code>swapcase()</code>	...byter A->a och a->A
<code>capitalize()</code>	Stor begynnelsebokstav
<code>title()</code>	"liten blå kanin"->"Liten Blå Kanin"
<code>strip()</code>	Tar bort blanktecken i början/slutet
<code>replace(x, y)</code>	Byter alla x mot y

STRÄNGMETODEN SPLIT()

- Det är enkelt att dela upp en mening i ord:
`mening.split()`
- Delar vid mellanslag
- Returnerar en *lista* med orden

PYTHONS DOKUMENTATION

Hur kan man veta vilka funktioner och metoder som finns i python?

- På docs.python.org/3/ finns dokumentation
- Se även "Pythonlänkar" på kurshemsidan

INDEX

- Strängar, listor och tupler har index
- Index anger ordningsnummer för element
- Index skrivs inom hakparenteser
- `låda[1]`



STRÄNGAR - INDEX

- Varje tecken i en sträng har index.
- Exempel:

`mat="pizza"`

p	i	z	z	a
0	1	2	3	4



mat[0]	mat[1]	mat[2]	mat[3]	mat[4]
"p"	"i"	"z"	"z"	"a"

STRÄNGAR: SKIVNING

mat[0] → "p"	Titta går bra...
mat[0] = "m"	...men inte ändra värdet.
mat[1:3] → "iz"	Delsträng (3:e ingår inte).
mat[:4] → "pizz"	Från början till 4.
mat[2:] → "zza"	Från 2 till slutet.

strängsskivning = string slicing

UPPGIFT: VAD BLIR S[3:]?

```
s = "aktie"  
print(s[3:])
```

akt

ie

t

tie

LISTAN

- En följd av data, omgiven av hakparenteser

```
tom = []
primaltal = [2,3,5,7,11,13,17,19]
dagar = ["tisdag", "onsdag", "fredag"]
```
- Kan innehålla data av valfri typ
- Har index precis som strängar
- Och en massa metoder

NÅGRA LISTMETODER

Metod	Beskrivning
append(x)	Lägger till x sist i listan.
sort()	Sorterar i stigande ordning.
reverse()	Vänder listan.
insert(i,x)	Stoppar in x på plats i.
pop(i)	Plockar ut elementet på plats i.
remove(x)	Tar bort första förekomsten av x.

ORDGISSNING

```
def spela():
    ordet = "RAMVERK"
    gissning = input("Gissa ordet: ")
    n = 1

    while gissning != ordet:
        if gissning > ordet:
            print("Ordet jag tänker på  
är före", gissning)
        elif gissning < ordet:
            print("Ordet jag tänker på  
är efter", gissning)
        gissning = input("Gissa igen: ")
        n += 1
    print("Rätt på bara", n, "gissningar!")
```

ORDGISSNING

```
def skapaLista():
    """ Skapar en lista med ord
    sorterad i alfabetisk ordning """
    lista = ["KYNNNE", \
            "NARIG", \
            "SCHABLON", \
            "ADAPTERA", \
            "NIMROD", \
            "SUBVERSIV", \
            "VOLATIL", \
            "ARMADA", \
            "MATRIKEL", \
            "KYMIG", \
            "BEFÄNGD", \
            "ODIÖS", \
            "FÖRTY"]

    lista.sort()
    return lista
```

DICTIONARY

- Skapas med klamrar {}:

```
engelska = {}
```
- Lagrar datapar:

```
engelska['trollstav'] = 'wand'
```
- Varje datapar har en *nyckel* och ett *värde*.
- Nycklarna kan t ex vara strängar eller tal
- Värdena kan vara av vilken typ som helst.
- En dictionary har inte ordning!

nyckel = key
värde = value

DICTIONARY - METODER

Metod	Beskrivning
if n in d:	Kollar om nyckeln n finns i d
d[n] = x	Stoppar in x med nyckel n
print(d[n])	Skriver ut värdet med nyckel n
keys()	Ger alla nycklar
values()	Ger alla värden
pop(n)	Plockar ut det värde som har nyckel n

LÄXA 2

Se "Labbar, föreläsningar och övningar" på kurswebbsidan.

- I labb 3 ska ni skriva ett glosförhörsprogram.
- Besök <http://pythontutor.com/visualize.html#py=3>, klistra in det givna programmet och kör steg för steg så att du förstår vad som händer.
- Bygg på programmet med en funktion som genomför ett glosförhör (det är OK att dela upp i flera funktioner).
- Läs även Kap 7 i kursboken (om filer) till nästa vecka.