

# Uge 3

Tirsdag

# Dagens tekst

Typer

Strings

Lists

# Typy w Pythonie

integers: 1321

floats: 3.2324

booleans: True, False

strings: "hello world"

!

!

!

# Type konversion

```
f = 0.5  
i = 3  
j = 2  
b = True
```

float > integer > boolean

```
print i + b  
print f * i  
print f / i  
print j / i  
print j / float(i)
```

Strengge

# Hvad er en streng?

## Sekvens af karakterer

- bogstaver - store og små
- tal
- tegn
- specielle karakterer f.eks. '\n' og '\t'
- 'æ', 'ø', 'å' er specielle

# Definition af en streng

Valgfri anførelstegn: " " *eller* ' '

```
fruit = "banana"
```

```
numbers = "124452"
```

```
emptyString = ''
```

```
stringOfSpaces = '      '
```

# Gennemløb af en streng

```
fruit = "banana"
count = 0

for c in fruit:
    if c == 'a':
        count += 1

print count
```

Det fantastiske for loop





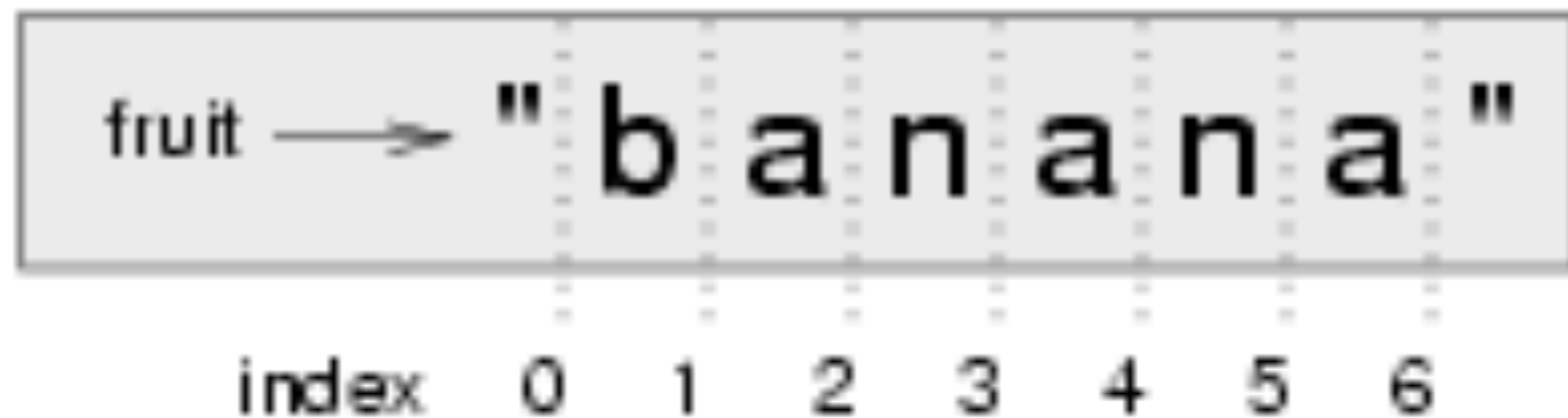
# Adgang til enkelte karakterer

```
fruit = "banana"  
letter = fruit[1]  
print letter
```

# Indeksering

Man angiver *indekset* for det element man vil have - ikke *nummeret* på elementet

I realiteten trækker man bare et fra nummeret...



# Opgave

Skriv et for loop der printer karaktererne fra en streng vha. indeksering.

Brug jeres bedste ven: `range( )`

ellers prøv

```
>>> help(range)
```

eller <http://docs.python.org/2/library/>

eller google: python range

# Løsning på opgave

```
s = "kasper"
```

```
# forwards:
```

```
for c in s:  
    print c
```

```
# forwards using indexing
```

```
for i in range(len(s)):  
    print s[i]
```

# Længde af en streng

```
fruit = "banana"  
length = len(fruit)  
print length
```

```
lastCharacter = fruit[length-1]
```

```
lastCharacter = fruit[-1]
```

# Opgave

Skriv et for loop der printer karaktererne fra en streng i omvendt rækkefølge.

# Løsning på opgave

```
s = "kasper"

# forlæns
for i in range(len(s)):
    print s[i]

# baglæns:
for i in range(1, len(s)+1):
    print s[-i]

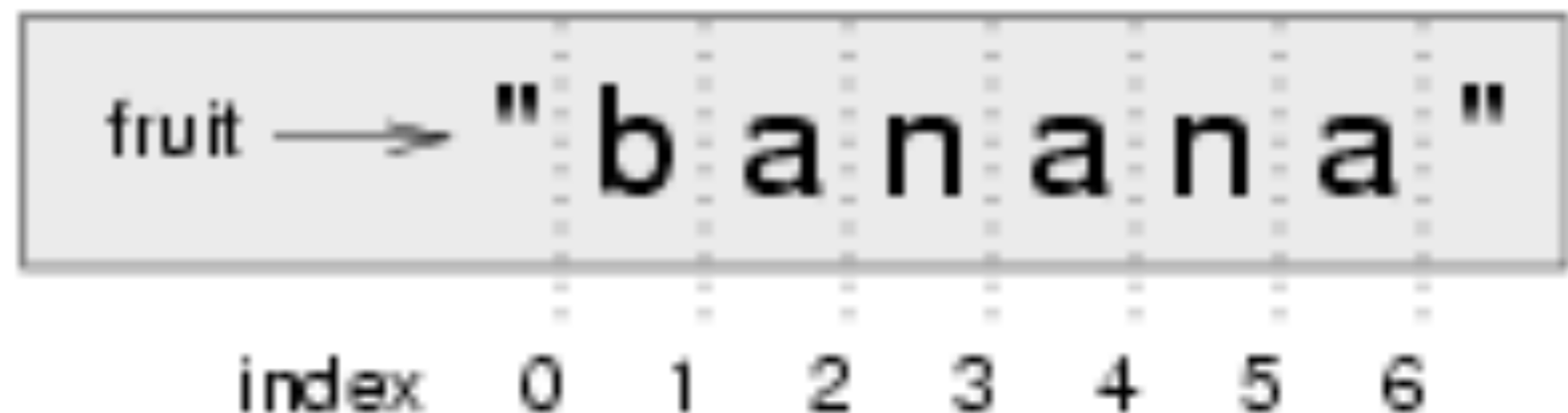
# baglæns alternativt:
for i in range(-1, -len(s)-1, -1):
    print s[i]
```

# Slicing

```
name = "Aarhus Universitet"
```

```
print name[0:7]
```

```
print name[7:18]
```





# Streng sammenligning

```
someString = "banana"
```

```
otherString = "Banana"
```

```
someString == otherString
```

```
someString != otherString
```

```
someString > otherString
```

# “in” operatoren

```
s = "Petersen"
```

```
ss = "sen"
```

```
if ss in s:
```

```
    print s, 'is a "sen name" '
```

# Strengene er "immutable" - "uændrelige"

```
s = "banana"
```

```
s[0] = 'B' # not legal !!
```

Lister

# Hvad er en liste?

Sekvens af hvad som helst:

- integers
- floats
- booleans
- strings
- lister
- dictionaries

Hvornår bruger i lister?

En liste er en “container” type



# Definition af en liste

```
lst = []
```

```
lst = [9, 3, 5]
```

```
lst = ["kasper", "munch"]
```

```
lst = [{"potato", "apple"}, [3, 5]]
```

# Adgang til listens elementer - indeksering

```
numbers = [1,2,3,4,5]
```

```
print numbers[0]
```

```
print numbers[-1]
```



# Længde af en liste

```
lst = [[2,7,5], ["pizza", "burger"]]
```

```
len(lst)
```

Hvad er længden af listen?

# Opgave

Skriv et for loop der beregner og printer kvadratet på 0, 1, 2, 3, 4, 5, 6, 7, 8 og 9.

# Løsning på opgave

```
for i in range(10):  
    x = i**2  
    print x
```

# Operationer på en liste

```
lst1 = [4, 5, 3]
```

```
lst2 = [2, 45, 23]
```

```
lst3 = lst1 + lst2
```

```
lst4 = lst1 * 3
```

# Slicing

```
guys = ["d'Artagnan",  
        "Athos",  
        "Porthos",  
        "Aramis"]  
  
musketees = guys[1:4]  
  
# shorthand:  
musketees = guys[1:]
```

# Lister er “mutable”

```
lst = [24, 562, 94]
```

```
lst[1] = 0.529
```

```
del lst[1]
```

# Gennemløb af (og ændring i) en liste

```
myList = range(5)
```

```
for i in myList:
```

```
    i = 0
```

```
print myList
```

```
# Versus
```

```
for i in range(len(myList)):
```

```
    myList[i] = 0
```

```
print myList
```

# Liste metoder

`l.append(x)`  
`l.extend(lst)`



Nye bedste venner

`l.sort()`  
`l.reverse()`

`l.insert(i, x)`  
`l.pop(i)`  
`l.remove(x)`  
`l.index(x)`  
`l.count(x)`



# Nestede lister

```
listOfLists = [[234234, 121, 5255],  
               [199238, 3721, 245],  
               [345602, 656, 5255]]
```

```
listOfLists[2][1] = 656
```

# En liste fra en streng - og tilbage

```
s = "banana"
```

```
lst = list(s)
```

```
lst[0] = "n"
```

```
s = "".join(lst)
```

# Tuples - listers imutible fætre

```
t = ("first", "second", "third")
```

```
lst = list(t)
```

```
t = tuple(lst)
```

# Typy w Pythonie

integers: 1321

floats: 3.2324

booleans: True, False

strings: "hello world"

lists: [ value, value, value ]

tuples: (value, value, value)

!

Uge 3

Torsdag

# Dagens tekst

Moduler og namespaces

Sidste uges aflevering

Mere om strenge

Precedence

# Importer af moduler

```
import math
```

```
math.sqrt(4)
```

```
from math import sqrt
```

```
sqrt(4)
```

```
dir(math)
```

# Opgave

Skriv noget kode der skriver alle math modulets funktioner og konstanter ud.

Hint: `help(dir)`



# Løsning på opgave

```
for s in dir(math):  
    if not s.startswith('_'):  
        print s
```

Sidste uges opgave...

# Streng metoder - giver nye strenge

```
s.capitalize()
```

```
s.upper()
```

```
s.lower()
```

```
s.replace()
```

```
s.strip()
```

```
s.split()
```

```
s.splitlines()
```

```
s.join()
```

# Streng metoder - testing

`s.startswith()`

`s.endswith()`

`s.isalpha()`

`s.isdigit()`

`s.islower()`

`s.isupper()`

`s.find()`

# Opgave

`s = " one two three "`

Fjern flankerende whitespace og newlines (`strip`).

Lav om til store bogstaver (`upper`).

Bryd op til en liste af ord (`split`).

Sæt sammen til: `"ONEandTWOandTHREE"` (`join`)

# Løsning på opgave

```
s = " one two three  "  
s = s.strip()  
s = s.upper()  
lst = s.split()  
s = "and".join(lst)
```

```
# or shorter:
```

```
s = "and".join(s.strip().upper().split())
```

# Streng formatting

```
"%s is %d years old" % ("He", 42)
```

```
"The result is: %f" % (8305.38)
```

```
"%e is as big as %.2e" % (100000)
```

# Opgave

`s = "bananas"`

`n = 7`

Lav følgende streng: "I have 7 bananas"

Hint: %s: string, %d: integer, %f: float



# Løsning på opgave

```
s = "bananas"
```

```
n = 7
```

```
print "I have %d %s" % (n, s)
```