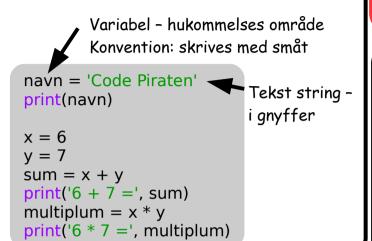
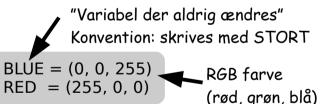
Dokumentation: http://docs.python.org

Variabel



Konstant



Type

En variabel har en 'type'

```
navn = 'Børge' # string (tekst)
tal_string = str(7)# konverter int til string
x = 7 # integer (heltal)
y = int('7') # konverter string til int
z = 1.4142 # float (komma tal)
sandt = True / False # (boolsk udtryk)
```



Coding Pirates

Hvis så

Basis

Betingelse (conditional)

```
Blokke indrykkes 4 tegn
Husk kolon (:) efter if, elif, else

alder = int(input('Angiv din alder: '))
if alder < 6:
    print('Du går i børnehave')
elif 6 <= alder < 18:
    print('Du går i skole')
else:
    print('Du er voksen')
```

operatorer

==	Lig med
!=	Forskellig fra
<	Mindre end
>	Større end

and/or tabeller

and	٢	Æ
٦	۲	F
F	F	F

or	T	F
Т	T	Т
F	Т	F

Løkker (loops)

Kode der skal udføres flere gange Blokke indrykkes 4 tegn Husk kolon (:) efter for, while

```
# for - gennemløb et antal gange
# gennemløb fra 0-4 (5 gange)
for i in range(0, 5):
  print('i er ', i)
# while - gennemløb så længe
# 'betingelse' er opfyldt
kl = 7
print('klokken er', kl, ', tid at stå op')
while kl < 21:
  print('tant og fjas')
  kl += 1
print('klokken er', kl, ', sengetid')
# break - ud af løkke 'før tid'
# continue - spring til 'næste'
for i in range(0, 10):
  if i == 2:
     continue
  elif i == 5:
     break
  print('i er ', i)
```



Dokumentation: http://docs.python.org

Liste

Liste med flere data elementer
Index baseret - starter i index 0

```
# liste med to string elementer
farver = ['rod', 'grøn']

# læs og skriv det første element
print(farver[0])
farver[0] = 'rød' # erstat 'rod' med 'rød'
print(farver)

# tilføj (append) et element i enden
farver.append('blå')
print(farver)
```

Map / Dictionary

'key' mapper til 'værdi' (som i en ordbog) Opslag med 'key'

```
# map med to opslag:
# - fra 'børge' til Børges alder (13)
# - fra 'piraten' til hans alder (317!)
elev_aldre = { 'børge': 13, 'piraten': 317 }

# opslag af alder for 'børge'
print('Børge er', elev_aldre['børge'], 'år')

# øg Børges alder med 1 år
elev_aldre['børge'] += 1
print('Børge er nu', elev_aldre['børge'], 'år')
```



I/O - input/output

Konsol I/O

```
# input - læs fra konsollen
alder = int(input('Angiv din alder: '))
# print - skriv til konsollen
print('Din alder er:', alder)
```

Fil I/O

```
# skriv fil - open med 'w' (for write)
fil = open('testfil.txt', 'w')
```

\n er linie skift (newline)
fil.write('coding pirates\n')
fil.write('python test fil\n')

en fil skal altid lukkes fil.close()

læs fil – open med 'r' (for read)
fil = open('testfil.txt', 'r')

læs linier - rstrip fjerner \n
linie = fil.readline().rstrip()
print('Første linie:', linie)
linie = fil.readline().rstrip()
print('Anden linie:', linie)

luk filen fil.close()

Funktioner

Definer små grupper af funktionalitet i en 'funktion', der så kan kaldes flere gange

```
# definition af funktion der skriver
# en liste af strings til en fil.
# argumenter:
# - navn: navn på fil
# - data: en liste af strings, hver
    string skrives på en linie i filen
def skrivfil(navn, data):
  fil = open(navn, 'w')
  for linie in data:
     fil.write(linie + '\n')
  fil.close()
# kald funktionen med to
# argumenter
skrivfil('aldre.txt', ['12', '13', '11'])
# kald funktionen igen, med to
# andre argumenter
skrivfil('navne.txt', ['børge', 'helge'])
```

Moduler

Et modul er en samling af kode som andre har skrevet - eksempler:

import random, time, calendar

Python Standard Library:

https://docs.python.org/3/library

