



Introduksjonskurs i Python

Muligheter for geologer og geoteknikkere

DPE	Kurs	11.2023
-----	------	---------

Hvorfor Python?

- Super populært
- Allsidig
- Stort felleskap – mange pakker og biblioteker



Hvorfor ANACONDA® ?

- Fri og åpen kildekode!
- Vitenskapelig og maskinlæring
- Hjelper med pakke-, avhengighets- og miljøhåndtering

Litt kode!

```
# Simple output (with Unicode)
```

```
>>> print(«Hei, jeg er din copper!")  
Hei, jeg er din copper!
```

```
# Input, assignment
```

```
>>> name = input('Hva heter du?\n')  
Hva heter du?  
Copper  
>>> print(f'Hei, {name}.')  
Hei, Copper.
```

← Kommentar har #

← Kaller på «print()» funksjonen

← Lager en variabel som holde det input() returnerer

- input()- funksjonen viser teksten du passerer og lagrer ditt svar i variabelen «name»

← Kaller på «print()» igjen, men gir inn variabelen i tillegg



Metoder/funksjoner

```
# Python 3: Fibonacci series up to n
>>> def fib(n):
>>>     a, b = 0, 1
>>>     while a < n:
>>>         print(a, end=' ')
>>>         a, b = b, a+b
>>>     print()
>>> fib(1000)
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987
```

>_

Matte og algebra

```
# Python 3: Simple arithmetic
>>> 1 / 2
0.5
>>> 2 ** 3
8
>>> 17 / 3 # classic division returns a float
5.666666666666667
>>> 17 // 3 # floor division
5
```

>_

Lister og sammensatte datatyper

```
# Python 3: List comprehensions
>>> fruits = ['Banana', 'Apple', 'Lime']
>>> loud_fruits = [fruit.upper() for fruit in
>>> fruits]
>>> print(loud_fruits)
['BANANA', 'APPLE', 'LIME']

# List and the enumerate function
>>> list(enumerate(fruits))
[(0, 'Banana'), (1, 'Apple'), (2, 'Lime')]
```

>_

Løkker og kontroll uttrykk

```
# For loop on a list
>>> numbers = [2, 4, 6, 8]
>>> product = 1
>>> for number in numbers:
...     product = product * number
...
>>> print('The product is:', product)
The product is: 384
```

>_

Dokumentasjon og hjelp!

- docs.python.org
- learning.anaconda.cloud
- Google problemet ditt!