



Kræsjskurs

aaa

Ola Nordmann
, NTNU
1 January 1970

Tilordningsoperatoren



Gitt to variabler a og b , bytt om innholdet.
Tilordningsoperatoren =
Original verdi blir overskrevet.

Tilordningsoperatoren



```
1  a = "hei"  
2  b = 3.14  
3  a = b  
4  b = a
```

	1	2	3	4
a				
b				

Tilordningsoperatoren



```
1  a = "hei"  
2  b = 3.14  
3  a = b  
4  b = a
```

	1	2	3	4
a	"hei"			
b	?			

Tilordningsoperatoren



```
1  a = "hei"  
2  b = 3.14  
3  a = b  
4  b = a
```

	1	2	3	4
a	"hei"	"hei"		
b	?	3.14		

Tilordningsoperatoren



```
1  a = "hei"  
2  b = 3.14  
3  a = b  
4  b = a
```

	1	2	3	4
a	"hei"	"hei"	3.14	
b	?	3.14	3.14	

Tilordningsoperatoren



Problem: Gammel verdi blir overskrevet. Vi kan bare endre en ting av gangen.

```
1  a = "hei"  
2  b = 3.14  
3  a = b  
4  b = a
```

	1	2	3	4
a	"hei"	"hei"	3.14	3.14
b	?	3.14	3.14	3.14

Tilordningsoperatoren



To vanlige løsninger

```
1  a = "hei"  
2  b = 3.14  
3  old_a = a    # ofte kalt temporary / temp  
4  a = b  
5  b = old_a
```

	1	2	3	4	5
a	"hei"	"hei"	"hei"	3.14	3.14
b	?	3.14	3.14	3.14	"hei"
a_old	?	?	"hei"	"hei"	"hei"

Tilordningsoperatoren



To vanlige løsninger

```
1  a = "hei"  
2  b = 3.14  
3  (b, a) = (a, b)  
4  # bruker tupler  
5  # kommer tilbake til dette senere
```

Tilordningsoperatoren



Flere tilordninger etter hverandre

```
1 a = b = c = 3.14
```

Tilordningsoperatoren



Flere tilordninger etter hverandre

Triks: Parenteser.

```
1  a = (b = (c = 3.14))  
2  # ikke riktig syntaks, kun for illustrere
```

Flere tilordningsoperatorer



Finnes flere tilordningsoperatorer.

VIKTIG: Bortsett fra standard tilordning, må variabelen som tilordnes har en verdi fra før av.

Operator	Eksempel	Det samme som
=	a = 1	a = 1
--	a -= 1	a = a - 1
/=	a /= 1	a = a / 1
//=	a //= 1	a = a // 1
+=	a += 1	a = a + 1
*=	a *= 1	a = a * 1
%=	a %= 1	a = a % 1
=	a **= 1	a = a1

Aritmetiske operatører



Viktig: Holde tunga rett i munnen med `//`, `%` og `**`

Operator	Eksempel
<code>+</code>	<code>1 + 2</code>
<code>-</code>	<code>1 - 2</code>
<code>/</code>	<code>1 / 2</code>
<code>//</code>	<code>1 // 2</code>
<code>*</code>	<code>1 * 2</code>
<code>%</code>	<code>1 % 2</code>
<code>**</code>	<code>1 ** 2</code>

Aritmetiske operatorer



Hva skjer her?

```
1  from math import sqrt
2  a = sqrt(2)
3  b = 2**(1/2)
4
5  print(a == b)
```

Aritmetiske operatorer



Hva skjer her?

True printes. Hvorfor trenger man da *sqrt* funksjonen?

```
1  from math import sqrt
2  a = sqrt(2)
3  b = 2**(1/2)
4
5  print(a == b)
```