

Sammenligningsoperatorer



Sammenligningsoperatorer og betingede hendelser er kanskje det viktigste med datamaskiner. Det er nødvendig for å gjøre en datamaskin *Turing-complete*.

Sammenligningsoperatorer



Boolske variabler er essensielle for sammenligningsoperatorer. En boolsk variabel kan kun være to verdier, *True* eller *False*.

Sammenligningsoperatorer

Sammenligningsoperatorer og logiske operatorer.

Operator	Eksempel	Resultat
<	2 < 2	False
>	2 > 2	False
<=	2 <= 2	True
>=	2 >= 2	True
==	2 == 2	True
!=	2 != 2	False
and	True and False	False
or	True or False	True
not	not False	True
in	1 in [3, "hei", 1]	True
not in	"hade" not in [3, "hei", 1]	True

Betingelser



if, *elif* og *else* brukes for å kontrollere programflyten. Man kan dermed aktivere en utvalgt del av koden ved å sjekke boolske verdier.

Betingelser



if, *elif* og *else* brukes for å kontrollere programflyten. Man kan dermed aktivere en utvalgt del av koden ved å sjekke boolske verdier.

```
num = 1
if num == 0:
    print("num er 0")
elif num == 1:
    print("num er 1")
else:
    print("num er noe annet")
```

Betingelser



if, *elif* og *else* brukes for å kontrollere programflyten. Man kan kjede så mange *elif*-er etter hverandre.

```
num = 1
if num == 0:
    print("num er 0")
elif num == 1:
    print("num er 1")
elif num == 2:
    print("num er 2")
elif num == 3:
    print("num er 3")
else:
    print("num er noe annet")
```

Betingelser



if, *elif* og *else* brukes for å kontrollere programflyten.

Man kan kjede så mange *elif*-er etter hverandre.

Viktig: Kun en av kode-snuttene vil kjøre mellom en *if* og *else*!

```
num = 1
if num == 0:
    print("num er 0")
elif num == 1:
    print("num er 1")
elif num == 2:
    print("num er 2")
elif num == 3:
    print("num er 3")
else:
    print("num er noe annet")
```

Betingelser

if, *elif* og *else* brukes for å kontrollere programflyten.

Man kan kjede så mange *elif*-er etter hverandre.

Viktig: Kun en av kode-snuttene vil kjøre mellom en *if* og *else*!

Her er det nye *if*-er, og dermed kan flere aktiveres samtidig.

```
num = 1
if num < 0:
    print("num mindre enn 0")
if num < 1:
    print("num mindre enn 1")
if num < 2:
    print("num mindre enn 2")
if num < 3:
    print("num mindre enn 3")
else:
    print("num er noe annet")
```