

DE24- Programmering 40 Yhp

Operatorer och if-satser

Logiska beslut

- Om bussen är mer än 5 minuter försenad ta tunnelbanan annars ta bussen.
- Om det finns mindre än 10 exemplar kvar av en vara beställ hem nya varor.
- Om det regnar ta med paraplyet annars lämna det hemma.
- Om en faktura inte är betalad efter 30 dagar skicka påminnelse.

Jämförelseoperatorer

Comparison operators are used to compare two values:

Operator	Name	Example
==	Equal	<code>x == y</code>
!=	Not equal	<code>x != y</code>
>	Greater than	<code>x > y</code>
<	Less than	<code>x < y</code>
>=	Greater than or equal to	<code>x >= y</code>
<=	Less than or equal to	<code>x <= y</code>

Används ju med if eller while . Men matematiskt kan man

`2 == 2`

`1 == 2`

`2 == 2.`

IF-satser

- För att kunna skapa logiska villkor för hur koden körs används IF-satser.
- För att kunna skapa villkor i en IF sats används olika logiska operatorer.
- Exempel:

```
ålder = 18
```

```
if ålder >= 18:
```

```
    print("Grattis! Du har nu rätt att rösta.")
```

Kontrollsatser - if/elif/else

- Exempel:

```
x = 42
```

```
if x > 0:
```

```
    print("x is positive!")
```

```
elif x < 0:
```

```
    print("x is negative!")
```

```
else:
```

```
    print("x is zero!")
```

```
print("Done.")
```

Logiska uttryck

- Ett logiskt uttryck är vilket uttryck som helst som utvärderas som antingen sant ('True') eller falskt ('False').
- Det kan vara så enkelt som en enda jämförelse eller en kombination av flera villkor.

Logiskaoperationer

Logical operators are used to combine conditional statements:

Operator	Description	Example
and	Returns True if both statements are true	<code>x < 5 and x < 10</code>
or	Returns True if one of the statements is true	<code>x < 5 or x < 4</code>
not	Reverse the result, returns False if the result is true	<code>not(x < 5 and x < 10)</code>

Multiple conditions

- Multiple condition refererar till situationer där man behöver testa flera villkor samtidigt i en if-sats. För att hantera sådana fall kan man använda logiska operatorer som and (och) och or (eller).

Recap

- Givet ett logiskt uttryck så kan det vara så att vi vill köra en viss kod-snutt eller en annan kod-snutt baserat på utfallet av det logiska uttrycket!
- Sådana önskade val åstadkoms med hjälp av vägvalsfunktionerna if, elif och else i Python, ett annat ord är kontrollsatser.
- Dessa funktioner exekverar sin indenterade kod endast om något logiskt uttryck är sant (True).
- **NOTERA!** En if-sats börjar alltid med if-funktionen. Efter det så kan vi välja att ha 0 eller hur många elif vi vill. Därefter kan vi välja att antingen ha en else eller ingen else. Detta då else alltid exekverar sin kod om alla föreliggande vägvalsfunktioner resulterade i falska logiska uttryck.
- **NOTERA!** Endast en av vägvalen körs i en if-sats.

if-satser

```
if uttryck1:  
    print('Vi kör kod som finns här om uttryck1  
    utvärderas till (True)')  
  
elif uttryck2:  
    print('Annars kör vi koden här om uttryck2  
    utvärderas till (True)')  
  
elif uttryck3:  
    print('Annars kör vi koden här om uttryck3  
    utvärderas till (True)')  
  
else:  
    print('Om alla ovanstående uttryck utvärderades till  
(False) så kommer koden här att köras')
```