

While-loopar

Programmering 1

2024/25

Outline

Upprepa

if-satsen

while

Loop i en loop

Speciella kommandon

Pass

Continue

Break

Övningar

Blad 1

Blad 2

Upprepa

If-satsen

Vi har tidigare kollat på `if`-satser: Om något, gör något. Typ:

```
1 tal = input("Skriv ett tal: ")
2 tal = int(tal)
3
4 if tal > 10:
5     print("Ditt tal är skitstort!")
```

Det vi ska kolla på idag är: "Så länge något, gör något"

Upprepa

Så länge något

Tänk att vi har kodsnutten:

```
1 tal = input("Skriv ett heltal: ")
2 tal = int(tal)
3
4 if tal != 0:
5     dubbel = tal*2
6     print(dubbel, "är dubbelt så mycket som", tal)
```

Och så vill vi att programmet ska fortsätta att fråga tills vi är nöjda (och skriver 0).

Upprepa

Så länge något

```
1 tal = input("Skriv ett heltal: ")
2 tal = int(tal) # Konverterar till heltal
3
4 while tal != 0:
5     dubbel = tal*2 # Dubblar talet
6     print(dubbel, "är dubbelt så mycket som", tal)
7     tal = input("Skriv ett heltal: ")
8     tal = int(tal)
```

Upprepa

while

Vi kan också ha en kod som ska vi redan innan vet hur många gånger den ska upprepa. Då behöver vi en räknare, eller en *iterator*. Det vanligaste är att man döper den variabeln till `i`.

```
1 i = 0
2
3 while i < 10:
4     print(i)
5     i += 1 # Jätteviktigt
```

När man gör en `while`-loop är det viktigt att man uppdaterar någon av variablerna i villkoret. Annars fastnar man i en oändlig loop.

Upprepa

Antal iterationer

Vi kan låta användaren bestämma antalet iterationer:

```
1 antal = input("Hur många tal vill du ha? ")
2 antal = int(antal)
3 i = 0
4
5 while i < antal:
6     print(2**i)
```

Loopa i en loop

```
1 i = 1
2 while i < 11:
3     j = 1
4     while j < 11:
5         print(i, j, i*j, end=" | ") # !!
6         j += 1
7     i += 1
8     print()
```

Vad händer på rad 5?

Outline

Upprepa

- if-satsen

- while

- Loop i en loop

Speciella kommandon

- Pass

- Continue

- Break

Övningar

- Blad 1

- Blad 2

Speciella kommandon

Det finns tre speciella kommandon som är kopplade till loopar i Python:

1. `pass`
2. `continue`
3. `break`

Speciella kommandon

Pass

Kommandot `pass` används mest under tiden man bygger sitt program och vill testköra utan att behöva göra klart alla delar.

```
1 i = 0
2 while i < 10:
3     if i%2==0:
4         pass
5     i += 1
```

Här vill vi att något särskilt ska hända när `i` är delbart med två, men vi har inte börjat skriva den delen än. Då kan vi använda `pass` direkt efter `if`-satsen för att förhindra ett `error`.

Speciella kommandon

Continue

Kommandot `continue` kan användas för att få loopen att fortsätta med nästa iteration direkt. Detta är vanligare att använda i `for`-loopar som kommer senare.

```
1 tal = int(input("Skriv: "))
2 while tal != -1:
3     if tal == 0:
4         tal = int(input("Skriv inte 0: "))
5         continue
6     print(1/tal)
7     tal = int(input(tal))
```

Eftersom vi inte får dela med noll så behöver vi ta hand om det fallet. Här hade det också funnits andra lösningar.

Speciella kommandon

Break

Kommandot `break` kan användas för att avsluta loopen under en iteration.

```
1 tal = int(input("Skriv: "))
2 while tal != -1:
3     if tal == 0:
4         break
5     print(1/tal)
6     tal = int(input(tal))
```

Helst ska man försöka undvika att använda `break` för att avsluta en loop.

Outline

Upprepa

- if-satsen

- while

- Loop i en loop

Speciella kommandon

- Pass

- Continue

- Break

Övningar

- Blad 1

- Blad 2

Övningar

Blad 1

1. Skriv en kod som skriver ut alla tal från 0 till 100
2. Skriv en kod som skriver ut talen 0.1, 0.2, 0.3 ... 1.0
3. Skriv en kod som skriver ut varannat tal från 10 till -10.
4. Skriv ut ettan till tolvans gångertabell.
5. Skriv en bit kod som skriver ut Fibonacci-serien upp till det hundra talet (1, 1, 2, 3, 5, 8, 13...)
6. Skriv en loop som skriver ut A–Z (tips kolla upp `chr()`)
7. Skriv ett program som frågar efter ett lösenord. Om användaren skriver **admin** ska det svara **Lösenord accepterat** annars ska det fråga efter lösenordet på nytt.

Övningar

Blad 2

Skriv ett program som frågar efter:

- ▶ Lånebelopp
- ▶ Räntesats
- ▶ Amortering/år

Och som sen skriver ut:

- ▶ Tid för att betala tillbaka lånet
- ▶ Totalt betalad ränta

Ex:

```
1 Hur mycket ska du låna? 2500000
2 Vad är räntesatsen (i %)? 2
3 Hur mycket ska du amortera? 60000
4 Det kommer att ta dig 42 år att betala tillbaka lånet
5 Du kommer att ha betalat 1066800 kr i ränta
```