

For lykkjur

# For lykkju kynning

Seinast sýndum við ykkur hvernig á að gera while lykkjur. Eitt dæmi sýndi annað veldi talna á bilinu 1-10

```
i = 1 # Base value
while i < 11:
    print(i**2, end=" ")
    i += 1 # Increment
print() # Newline
print("While loop has finished")
```

```
1 4 9 16 25 36 49 64 81 100
While loop has finished
```

For lykkjur eru mjög svipaðar while lykkjum og hægt er að skrifa sömu virkni svona:

```
for i in range(1, 11):
    print(i**2, end=" ")
print() # Newline
print("For loop has finished")
```

```
1 4 9 16 25 36 49 64 81 100
For loop has finished
```

# Range(a,b,c)

**Range** er hlutur í python sem samsvarar talnabili. Range getur tekið inn 3 breytur, range(a,b,c).

**Fyrsta breytan(a)** segir til um hvar bilið eigið að byrja.

**Seinni breytan(b)** segir til um hvar eigið að stoppa.

**Þriðja breytan(c)** segir til um stærð skref við hverja ítrun

Ef range tekur aðeins einni breytu, þá er byrjunarskref(a) 0 og ítrunarskref(c) 1.

Ef range tekur tvær breytur, er ítrunarskref(c) 1.

```
range(10)          # 0,1,2,3, ... , 9
range(0, 10, 1)    # 0,1,2,3, ... , 9
range(5, 8)        # 5,6,7
range(5, 8, 1)     # 5,6,7
range(1, 11, 2)    # 1,3,5,7,9
```

Bilið í range er  $a \leq i < b$ . Þar sem  $i += c$  í hverri ítrun

Takið eftir því að hann b(endapunkturinn) er ekki á bilinu.

# Range(a,b,c) – áframh.

Hægt er að geyma range innan í breytu, fínt ef þú vilt endurnota sama range aftur

```
my_range = range(10)
print(type(my_range))
```

```
<class 'range'>
```

```
for i in my_range:
    print(i, end=" ")
print()
for i in my_range:
    print(i**2, end=" ")
```

```
0 1 2 3 4 5 6 7 8 9
0 1 4 9 16 25 36 49 64 81
```

Segjum að þú þyrftir að breyta bilinu á þessum tveimur lykkjum. Ef þú ert með breytuna `my_range`, þarftu bara að breyta henni og kemst þar með framhjá því að breyta range í hverju einasta for lykkju

# For i in Range()

Þegar við skrifum **For i in range(a,b,c)**. Erum við að búa til lykkju, með breytu i sem byrjar sem a. Hún mun stoppa þegar i verður b og i stækkar um c við hverja ítrun.

Það er alltaf hægt að skrifa while lykkju sem hefur sömu virkni og for lykkja. For lykkja er einfaldlega styttri og auðveldari að lesa

```
for i in range(a,b,c):  
    print(i)
```

Hefur sömu virkni og

```
i = a  
while i < b:  
    print(i)  
    i += c
```

# For lykkju dæmi með mismunandi range



```
for i in range(10):  
    print(i, end=" ")  
print() # newline  
  
for i in range(5, 15):  
    print(i, end=" ")  
print() # newline  
  
for i in range(3, 31, 3):  
    print(i, end=" ")
```

```
0 1 2 3 4 5 6 7 8 9  
5 6 7 8 9 10 11 12 13 14  
3 6 9 12 15 18 21 24 27 30
```

# Range í öfugri röð

```
for i in range(0, 10, 1):  
    print(i, end=" ")
```

0 1 2 3 4 5 6 7 8 9

Hér er einföld for lykkja sem sýnir tölur á bilinu 0-9. En hvað ef við viljum skila útkomu í öfugri röð? Hægt er að setja mínus tölu sem ítrunartölu en þá þarf að hafa í huga að inntök `range(a,b,c)` þurfa að vera í takti við hvað við viljum

```
for i in range(9, -1, -1):  
    print(i, end=" ")
```

9 8 7 6 5 4 3 2 1 0

Við byrjum á 9, í hverri ítrun athugar for lykkjan hvort  $i < -1$ , og minnkar svo  $i$  um 1. Þegar  $i$  verður -1 mun  $-1 < -1$  ekki standast og þá hættir lykkjan.

While lykkja með sömu virkni lítur svona út:

```
i = 9  
while i > -1:  
    print(i, end=" ")  
    i -= 1
```

# Break og continue

**Break** og **continue** virka eins og með while lykkju.

**Break** stoppar keyrslu á lykkjunni

**Continue** hoppar yfir á næstu ítrun. Í for lykkjum mun i hækka um ítrunarstærð

```
for i in range(10):  
    if i == 5:  
        break  
    print(i, end=" ")
```

0 1 2 3 4

```
for i in range(10):  
    if i % 2 == 1:  
        continue # Continue if the number is odd, i+= 1  
    print(i, end=" ")
```

0 2 4 6 8



# Innri for lykkja

Eins og með while lykkjur er hægt að vera með innri for lykkjur innan í annari forlykkju.

```
1 for i in range(2):  
2     for j in range(2):  
3         for k in range(2):  
4             print(i + j + k, end=" ")
```

```
0 1 1 2 1 2 2 3
```

Í dæminu fyrir ofan mun lína 1 keyra, hún setur  $i=0$ . Næst keyrir lína 2, hún setur  $j = 0$ . Næst keyrir lína 3. Hún setur  $k=0$ .

Forritið klárar fyrst að keyra innstu lykkjuna, svo næst innstu og svo ystu lykkuna. Breyturnar breytast svona:

$i = 0, j = 0, k = 0$

$i = 0, j = 0, k = 1$

$i = 0, j = 1, k = 0$

$i = 0, j = 1, k = 1$

$i = 1, j = 0, k = 0$

$i = 1, j = 0, k = 1$

$i = 1, j = 1, k = 0$

$i = 1, j = 1, k = 1$