



# Python

---

For and while are a family

# There are two duplicate structures in the Python language

---

Initial value  
*while* Conditional:  
statement  
Conditions change

*for* Control variables *in*  
Conditional range:  
statement



Print from 1 to 10

Exercise



# Print from 1 to 10

---

By **while**

```
i = 1
while i < 11:
    print(i)
    i = i + 1
```

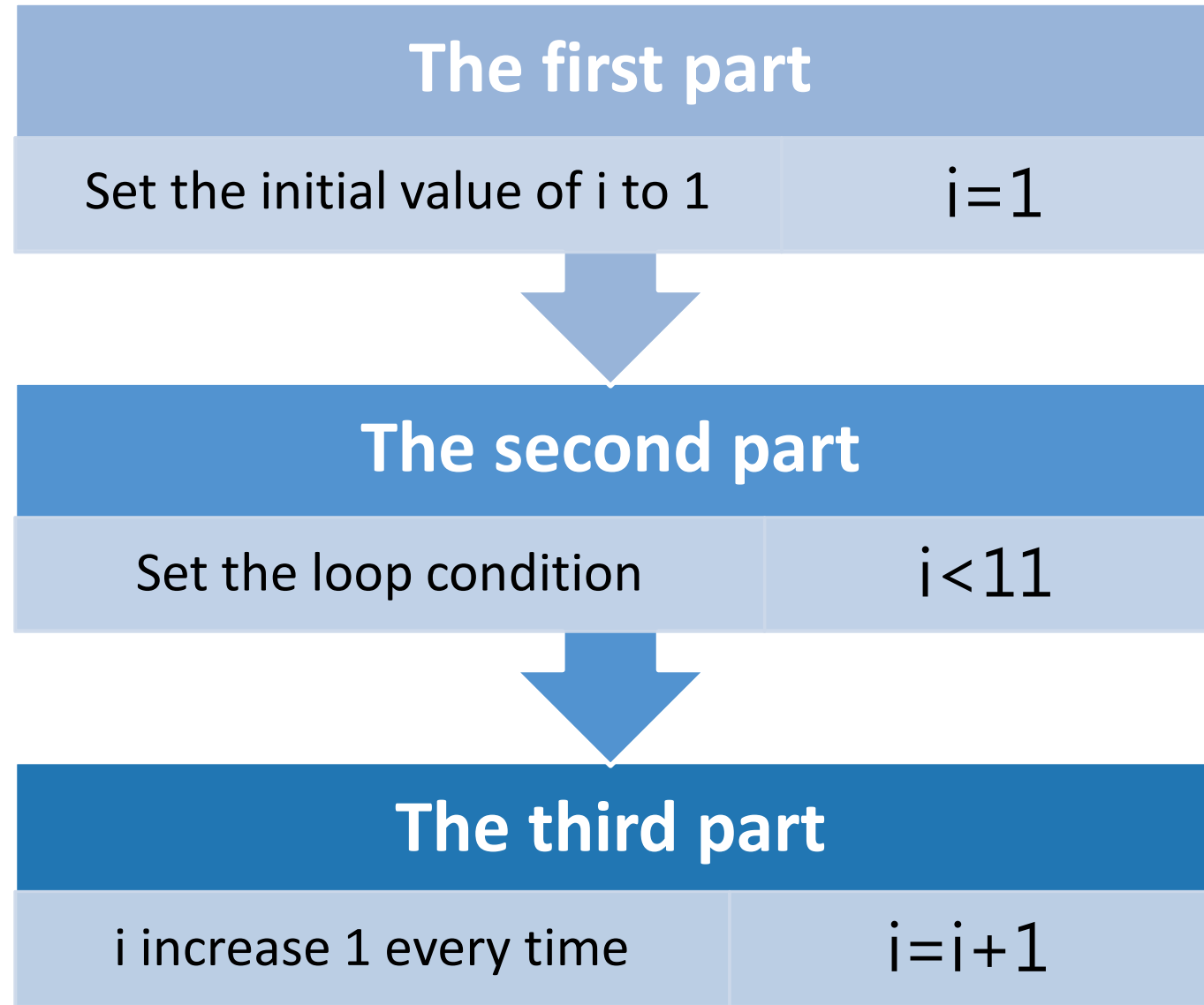
By **for**

```
for i in range(1, 11, 1):
    print(i)
```



# While loops :

Divided into 3 parts. Controlling the while loop to executed from 1 to 10.

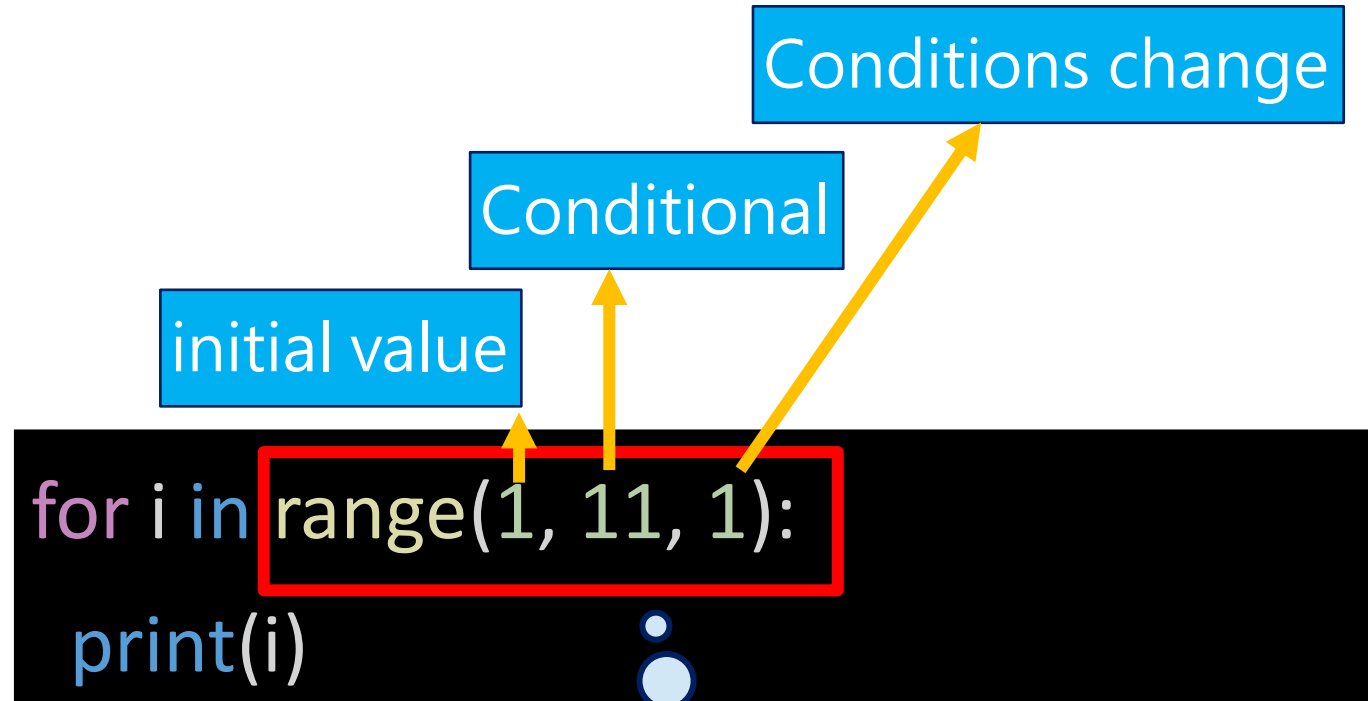


Reduce the three parts of while to one line.



# For loop :

Combine **the initial value, conditional and conditions change** into a conditional range.



range() can be consider as an arithmetic series

- Leading term => initial value
- Last term => conditional
- tolerance => conditions change



# Is **for** better than **while**?

---

Much more  
concise

Suitable for loop's  
counting, such as  
repeat 100 times  
or repeat n times.

But not every  
conditions is  
better than using  
while.





for is **much more concise**  
Three parts.

Different situations have different circumstances. There is no absolutely good or bad between while and for.

for is **more readable**  
while usually needs go with an internal narrative to confirm the loop content.

while has **more application range** than for  
You don't know when is the end, but you only know the end condition, does not apply for.

while is **more dangerous** than for  
When the conditional setting of the while is incorrect. May cause unexpected infinite loops.



Print the even  
number from 1  
to 10

Exercise



# Print the even number from 1 to 10

---

By **while**

```
i = 2
while i < 101:
    print(i)
    i = i + 2
```

By **for**

```
for i in range(2, 101, 2):
    print(i)
```

