

Lecture_12_For_loop_while_and_manipulation

15 September 2024 14:10



Lecture-12_ For and while loops.ipynb - Colab

General Format

```

while <test>: # Loop test
    <statements1> # Loop body
else: # Optional else
    <statements2> # Run if didn't exit loop with break

```

while —: True
False

```

while True
    print('Type Ctrl-C to stop me!')

```

```

x = 'spam'
while x: # While x is not empty
    print(x, end=' ')
    x = x[1:] # Strip first character off x

```



0 1 2 3 4 5 6

```

spam pam am m

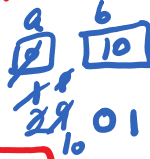
```

spam pam am m

```

a=0; b=10
while a < b: # One way to code counter loops
    print(a, end=' ')
    a += 1 # Or, a = a + 1

```



0 1 2 3 ... 9

```

for <target> in <object>: # Assign object items to target
    <statements> # Repeated loop body: use target
else:
    <statements> # If we didn't hit a 'break'

```

```

for x in ["spam", "eggs", "ham"]:
    print(x, end=' ')

```

```

spam eggs ham

```

```

sum = 0
for x in [1, 2, 3, 4]:
    sum = sum + x

```

```

sum
10

```

```

prod = 1
for item in [1, 2, 3, 4]: prod *= item

```

```

prod
24

```

break: Jumps out of the closest enclosing loop (past the entire loop statement) continue: Jumps to the top of the closest enclosing loop (to the loop's header line)

```

while <test1>:
    <statements1>
    if <test2>:
        break # Exit loop now, skip else
    if <test3>:
        continue # Go to top of loop now, to test1
else:
    <statements2> # Run if we didn't hit a 'break'

```

```

x= 10
while x:
    x = x-1 # Or, x -= 1
    if x % 2 != 0: continue # Odd? -- skip print
    print(x, end=' ')

```

```

8 6 4 2 0

```

```

while True:
    name = input('Enter name:')

```

20/09/2024, 06:47

Lecture-12: For and while loops.ipynb - Colab

```
if name == 'stop': break
age = input('Enter age: ')
print('Hello', name, '=>', int(age) ** 2)
```

```
↵ Enter name:Sadbhawna
Enter age: 29
Hello Sadbhawna => 841
Enter name:Sad
Enter age: 30
Hello Sad => 900
Enter name:stop
```

Start coding or [generate](#) with AI.