

Python Starter

Циклічні конструкції

Python Starter

План заняття

1. Поняття циклічної конструкції
2. Види циклічних конструкцій
3. Цикл while
4. Ключові слова break і continue
5. Цикл while з гілкою else
6. Цикл for
7. Цикл for з гілкою else
8. Вкладені цикли

Python Starter

Після уроку обов'язково



Повторіть цей урок у відео форматі на [ITVDN.com](http://itvdn.com)

Доступ можна отримати через керівництво вашого навчального центру



Перевірте, як Ви засвоїли цей матеріал на [TestProvider.com](http://testprovider.com)

Python Starter

Тема

Поняття циклічних конструкцій

Python Starter

Основні алгоритмічні конструкції



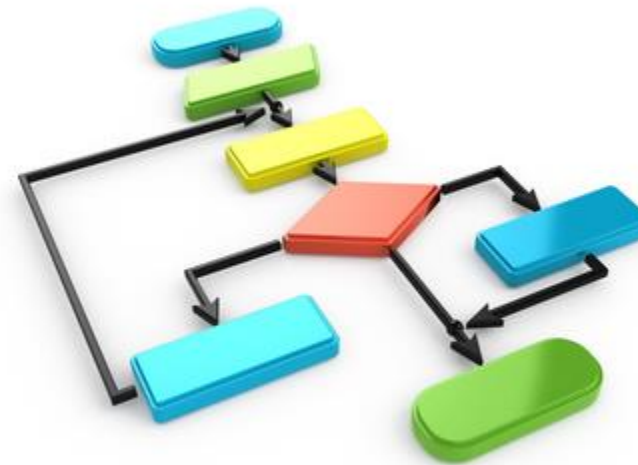
Python Starter

Поняття циклу

Цикл — це керуюча конструкція, призначена для багаторазового виконання набору інструкцій.

Види циклів:

- цикл з передумовою
- цикл з постумовою
- цикл з лічильником
- спільний цикл



Python Starter

Введення в Python

Цикл із передумовою

Python Starter

Цикл `while`

`while` умова:
оператори

Тут:

- *умова* – це логічний вираз,
- *оператори* – це послідовність будь-яких інших команд.

Блок операторів не може бути порожнім. Якщо така необхідність виникає, можна скористатися оператором *pass*, який нічого не робить.

Блок операторів виконується доти, доки умова продовження циклу є істинною. Якщо вона спочатку була хибною, він не виконується жодного разу.

Python Starter

Оператори break та continue

Якщо необхідно перервати виконання циклу достроково, використовується оператор break.

```
while True:
```

```
    response = input('Введіть команду')  
    if response == 'exit':  
        break
```

Якщо необхідно завершити виконання поточної ітерації циклу та перейти до наступної, використовується оператор continue

```
x = 0  
while x < 10:  
    x += 1  
    if x == 5:  
        continue  
    print('Поточне число дорівнює', x)  
    print('(але число 5 ми не виводимо)')
```

Python Starter

Цикл while з гілкою else

```
while умова:  
    блок_операторів_1  
else:  
    блок_операторів_2
```

Оператор while також може мати гілку else (за аналогією з if). На початку кожної ітерації інтерпретатор перевіряє істинність умови виконання циклу, і якщо вона є істинною, то виконує гілку while, інакше виконує гілку else (якщо вона присутня) і завершує виконання циклу, причому це може статися і перед першою ітерацією, якщо умова початково була хибною. Однак, якщо цикл був перерваний оператором break, то гілка else не виконується.

Python Starter

Діапазони

- **Діапазони** – незмінювані послідовності чисел, які задаються початком, кінцем та кроком. Представлені класом `range` (Python 2 – `xrange`; `range` у Python 2 – це функція, яка повертає список).
- Початок за замовчуванням дорівнює нулю, крок – одиниці. Якщо задати нульовий крок, буде викинуто виняток `ValueError`.
- Параметри конструктора повинні бути цілими числами (або екземпляри класу `int`, або будь-який об'єкт з методом `__index__`).
- Елементи діапазону `r` визначаються за формулою $r[i] = \text{start} + \text{step} * i$, де $i \geq 0$ та $r[i] < \text{stop}$ для $\text{step} > 0$ або $r[i] > \text{stop}$ для $\text{step} < 0$.
- Підтримує всі спільні для послідовностей операції, крім повторення, а також, у версіях Python до 3.2, зрізів та негативних індексів.



Python Starter

Введення в Python

Цикл із лічильником

Python Starter

Цикл for

Цикл із лічильником (цикл **for**) – це цикл, в якому змінна - лічильник ітерацій змінює своє значення від початкового до кінцевого з певним кроком.

for змінна **in range**(кінцеве_значення):
оператори

або

for змінна **in range**(початкове_значення, кінцеве_значення):
оператори

або

for змінна **in range**(початкове_значення, кінцеве_значення, крок):
оператори

Початкове значення входить у діапазон, кінцеве – ні.

Якщо змінна не використовується в коді, але все одно потрібна, відсутня потреба у вигадуванні імені для непотрібної змінної.

for _ **in range**(початкове_значення, кінцеве_значення, крок):
оператори

Python Starter

Обхід рядків

Для обходу елементів рядків використовується цикл for:

```
for element in string:  
    # ...  
    # виконання операцій із element  
    # ...  
    pass
```

Python Starter

Цикл for з гілкою else

В циклах for та while опціонально може використовуватися блок **else**.

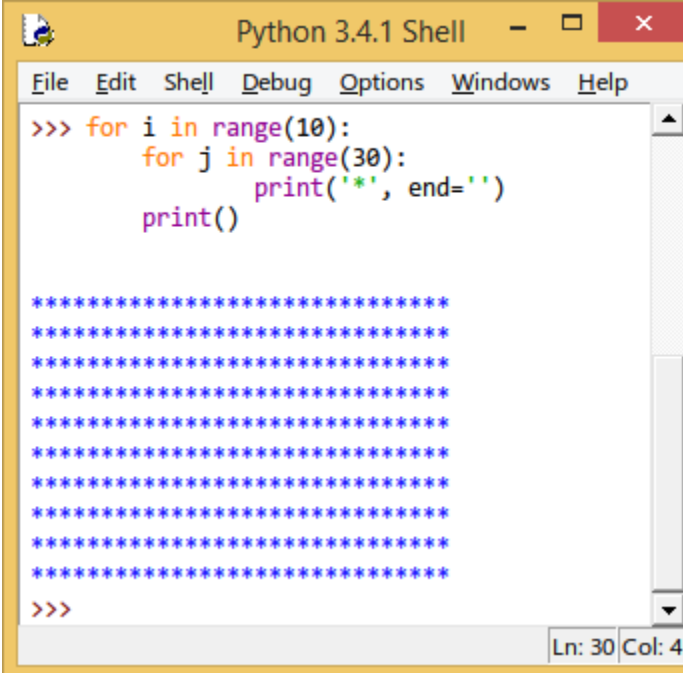
Реалізація без гілки else:	Реалізація з гілкою else:
<pre>for n in range(1, 10): for x in range(2, n): if n % x == 0: print(n, 'equals', x, '*', n / x) break</pre>	<pre>for n in range(1, 10): for x in range(2, n): if n % x == 0: print(n, 'equals', x, '*', n / x) break else: # loop fell through without finding a factor print(n, 'is a prime number')</pre>

Python Starter

Вкладені цикли

Вкладені цикли – це цикли, що знаходяться всередині інших циклів. Цикл, вкладений у тіло іншого, називається внутрішнім циклом. Цикл, у тіло якого вкладено інший цикл, називається зовнішнім.

```
for i in range(10):  
    for j in range(30):  
        print('*', end='')  
    print()
```



```
Python 3.4.1 Shell  
File Edit Shell Debug Options Windows Help  
>>> for i in range(10):  
    for j in range(30):  
        print('*', end='')  
    print()  
  
*****  
*****  
*****  
*****  
*****  
*****  
*****  
*****  
*****  
*****  
>>>  
Ln: 30 Col: 4
```


Дивіться наші уроки у відео форматі

ITVDN.com




Перегляньте цей урок у відео форматі на освітньому порталі [ITVDN.com](http://itvdn.com) для закріплення пройденого матеріалу.

Усі курси записані сертифікованими тренерами, які працюють у навчальному центрі CyberBionic Systematics



Перевірка знань

TestProvider.com



TestProvider – це online сервіс перевірки знань з інформаційних технологій. За його допомогою Ви можете оцінити Ваш рівень та виявити слабкі місця. Він буде корисним як у процесі вивчення технології, так і для загальної оцінки знань IT спеціаліста.

Після кожного уроку проходите тестування для перевірки знань на TestProvider.com

Успішне проходження фінального тестування дозволить Вам отримати відповідний Сертифікат.



Python Starter

Q&A

Python Starter

Дякую за увагу!

Інформаційний відеосервіс для розробників програмного забезпечення

