

 alifazahra729 / Workshop--python- Public[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)[master](#) Workshop--python- / teori / minggu-02 /

alifazahra000 README.md minggu-02 ...

3 days ago [History](#)

README.MD

3 days ago

README.MD



Ringkasan :

- Bab 4 Selain pernyataan while yang baru saja diperkenalkan, Python menggunakan pernyataan kontrol aliran biasa yang dikenal dari bahasa lain, dengan beberapa perubahan. Bisa ada nol atau lebih bagian elif, dan bagian lain bersifat opsional. Kata kunci 'elif' merupakan kependekan dari 'else if', dan berguna untuk menghindari indentasi yang berlebihan. Daripada selalu mengulangi deret aritmatika angka, atau memberi pengguna kemampuan untuk menentukan langkah iterasi dan kondisi penghentian, pernyataan for Python mengulang item dari urutan apa pun, dalam urutan kemunculannya dalam urutan. Hal yang aneh terjadi jika Anda hanya mencetak range: range range

Kami mengatakan objek seperti itu dapat diubah, yaitu cocok sebagai target untuk fungsi dan konstruksi yang mengharapkan sesuatu dari mana mereka dapat memperoleh item berturut-turut hingga persediaan habis. Pernyataan break, seperti di C, keluar dari loop for atau while terlampir terdalam. Pernyataan pass tidak melakukan apa-apa. Contoh: while True: pass # Busy-wait for keyboard interrupt. Pola pertama memiliki dua literal, dan dapat dianggap sebagai perluasan dari pola literal yang ditunjukkan di atas.

Cara yang disarankan untuk membaca pola adalah dengan melihatnya sebagai bentuk lanjutan dari apa yang akan Anda letakkan di sebelah kiri tugas, untuk memahami variabel mana yang akan disetel ke apa. Hanya nama mandiri yang diberikan oleh pernyataan kecocokan.

Tidak ada

Pernyataan kembali kembali dengan nilai dari suatu fungsi. Hasil pernyataan. Tambahkan panggilan metode hasil objek daftar. Metode adalah fungsi yang 'milik' objek dan diberi nama obj.

Methodname, di mana obj adalah object, dan methodname adalah nama method yang ditentukan oleh tipe objek. Dimungkinkan juga untuk mendefinisikan fungsi dengan sejumlah variabel argumen. Ada tiga bentuk yang bisa digabungkan. Ini menciptakan fungsi yang dapat dipanggil dengan lebih sedikit argumen daripada yang diizinkan. Ini dapat digabungkan dengan parameter formal dari bentuk *nama yang menerima tuple yang berisi argumen posisi di luar daftar parameter formal. Secara default, argumen dapat diteruskan ke fungsi Python baik dengan posisi atau secara eksplisit dengan kata kunci.

-- Posisi hanya di mana / dan * adalah opsional. Jika / dan * tidak ada dalam definisi fungsi, argumen dapat diteruskan ke fungsi berdasarkan posisi atau kata kunci. Melihat ini sedikit lebih detail, dimungkinkan untuk menandai parameter tertentu sebagai hanya posisi. Parameter khusus posisi ditempatkan sebelum /. Jika tidak ada / dalam definisi fungsi, tidak ada parameter hanya posisi. Untuk menandai parameter sebagai hanya kata kunci, yang menunjukkan bahwa parameter harus diteruskan oleh argumen kata kunci, beri tanda * pada daftar argumen tepat sebelum parameter khusus kata kunci pertama. Dengan kata lain, nama parameter hanya posisi dapat digunakan dalam **kwds tanpa ambiguitas. Sebagai pedoman

Gunakan hanya posisi jika Anda ingin nama parameter tidak tersedia untuk pengguna. Ini berguna ketika nama parameter tidak memiliki arti sebenarnya, jika Anda ingin menerapkan urutan argumen saat fungsi dipanggil atau jika Anda perlu mengambil beberapa parameter posisi dan kata kunci arbitrer.

Terakhir, opsi yang paling jarang digunakan adalah menentukan bahwa suatu fungsi dapat dipanggil dengan jumlah argumen yang berubah-ubah. Argumen ini akan dibungkus dalam tuple. Sebelum jumlah argumen variabel, nol atau lebih argumen normal dapat terjadi.

Bergabunglah dengan concat 'earth/mars/venus' concat 'earth'.

Situasi sebaliknya terjadi ketika argumen sudah ada dalam daftar atau tuple tetapi perlu dibongkar untuk pemanggilan fungsi yang memerlukan argumen posisi terpisah. Misalnya, fungsi rentang bawaan mengharapkan argumen start dan stop yang terpisah. Fungsi anonim kecil dapat dibuat dengan kata kunci lambda. Fungsi Lambda dapat digunakan di mana pun objek fungsi diperlukan. Berikut adalah beberapa konvensi tentang konten dan pemformatan string dokumentasi. Parser Python tidak menghapus indentasi dari literal string multi-baris di Python, jadi alat yang memproses dokumentasi harus menghapus indentasi jika diinginkan. Baris tidak kosong pertama setelah baris pertama string menentukan jumlah lekukan untuk seluruh string dokumentasi. Spasi "setara" dengan lekukan ini kemudian dihapus dari awal semua baris string.

Garis yang indentasinya lebih kecil tidak boleh muncul, tetapi jika muncul, semua spasi putih di depannya harus dihilangkan. Kesetaraan spasi harus diuji setelah perluasan tab.

Anotasi fungsi adalah informasi metadata yang sepenuhnya opsional tentang jenis yang digunakan oleh fungsi yang ditentukan pengguna. Anotasi parameter ditentukan oleh tanda titik dua setelah nama parameter, diikuti dengan ekspresi yang mengevaluasi nilai anotasi.

Sekarang Anda akan menulis potongan Python yang lebih panjang dan lebih kompleks, inilah saat yang tepat untuk berbicara tentang gaya pengkodean. Sebagian besar bahasa dapat ditulis dengan gaya yang berbeda; beberapa lebih mudah dibaca daripada yang lain. Memudahkan orang lain untuk membaca kode Anda selalu merupakan ide yang bagus, dan mengadopsi gaya pengkodean yang bagus sangat membantu untuk itu. Tab menimbulkan kebingungan, dan sebaiknya ditinggalkan. Bungkus garis sehingga tidak melebihi 79 karakter. Ini membantu pengguna dengan tampilan kecil dan memungkinkan beberapa file kode berdampingan pada tampilan yang lebih besar. Gunakan spasi di sekitar operator dan setelah koma, tetapi tidak langsung di dalam konstruksi tanda kurung: $a = f + g$. Beri nama kelas dan fungsi Anda secara konsisten; konvensinya adalah menggunakan UpperCamelCase untuk kelas dan huruf kecil_dengan_garis bawah untuk fungsi dan metode. Selalu gunakan self sebagai nama untuk argumen metode pertama.

Give feedback