Pertemuan ke 7

Fungsi Fungsi

Fungsi adalah suatu blok kode yang digunakan untuk menjalankan tugas pemrograman. Dalam pemrograman, fungsi sering disebut sebagai subprogram. Salah satu keuntungan fungsi adalah dapat digunakan berulang. Ini berarti bahwa fungsi hanya perlu didefinisikan satu kali namun dapat digunakan berulang kali.

Bab ini akan membahas teknik pendefinisian fungsi dalam python, pengiriman parameter kedalam fungsi, penggunaan fungsi dan hal hal lain yang berkaitan dengan fungsi. Fungsi fungsi ini dikelompokan berdasarkan kategorinya. Tujuannya adalah untuk memperkaya wawasan dan meingkatkan kemampuan anda dalam menggunakan bahasa pemrograman Python untuk menangani kasus kasus pemrograman yang sedang anda hadapi.

* Jenis Fungsi

Dalam pemrograman, fungsi dibedakan menjadi dua yaitu

1. Fungsi tanpa nilai balik
2. Fungsi dengan nilai balik

Fungsi tanpa nilai balik adalah fungsi yang tidak menghasilkan nilai. Fungsi tersebut hanya untuk melakukan proses tertentu. Contohnya seperti fungsi print(), Fungsi ini hanya digunakan untuk melakukan pencetakan teks atau bilangan ke layar. Ini berarti bahwa tidak ada nilai yang dihasilkan ketika anda mengeksekusi fungsi tersebut.

Fungsi dengan nilai balik adalah fungsi yang ketika dipanggil akan menghasilkan nilai. Contohnya seperti fungsi sqrt(), Fungsi ini akan menghitung nilai akar kuadrat dari suatu bilangan yang dilewatkan sebagai parameternya.

* Memanggil Fungsi

Ketika suatu fungsi dipanggil, eksekusi program akan meloncat atau mengarah ebagian blok fungsi bersangkutan. Selanjutnya, program akan mengeksekusi satu per satu perintah yang terdapat di dalam fungsi secara beruntun. Contoh pemanggilan fungsi yaitu

1. Untuk Fungsi tanpa nilai balik

cetak = 35  
print("Umur= ", cetak)

hasil program

Umur= 35

1. Untuk Fungsi dengan nilai balik

x = 10  
y = 1  
print("x = ", x)  
print("y = ", y)  
print("hasil (x+y)= ", x+y)

hasil program

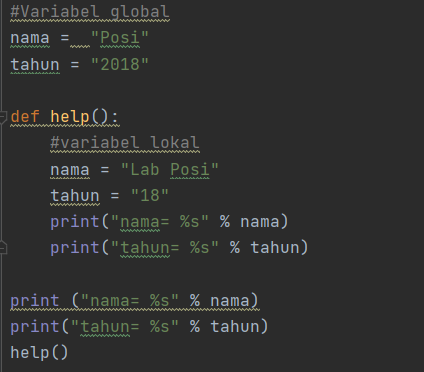
x = 10

y = 1

hasil (x+y)= 11

* Variabel Global dan variabel Lokal

Variabel global adalah variabel yang didefinisikan di luar fungsi sedangkan variable lokal adalah variable yang didefinisikan di dalam fungsi. program python mulai mencari vairabel lokal terlebih dahulu, kalau ada maka itu yang digunakan. Namun jika tidak ada, pencarian terus ke Global, dan Build-in.

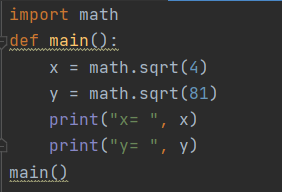


Perhatikanlah variabel nama yang berada di dalam fungsi help() dan diluar fungsi `help().

Variabel nama yang berada di dalam fungsi help() adalah variabel lokal. Jadi, saat kita memanggil fungsi help() maka nilai yang akan tampil adalah nilai yang ada di dalam fungsi help(). Kenapa tidak tampil yang global? Karena Python mulai mencari dari lokal, ke global, dan build-in.

* Parameter didalam fungsi

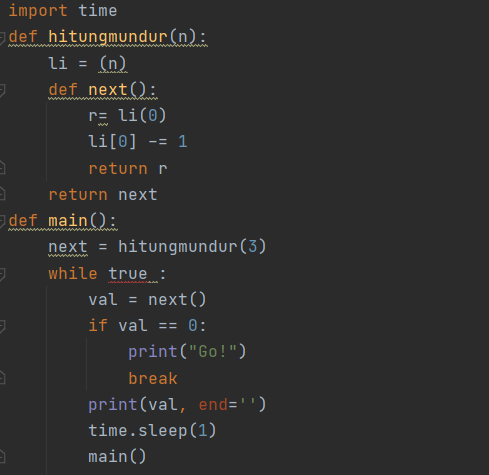
Parameter adalah suatu nilai yang dikirimkan ke dalam fungsi, kemudian akan ikut diproses di dalam badan fungsi. Dengan menggunakan parameter, sutau fungsi dapat memberikan hasil yang dinamis atau berubah ubah setiap fungsi tersebut dipanggil.



Fungsi sqrt() akan memberikan hasil berbeda sesuai dengan nilai parameter yang dilewatkan ke dalam fungsi. Fungsi sqrt() tersimpan di dalam modul *math* dan digunakan untuk menghitung akar kuadrat dari suatu bilangan yang dilewatkan sebagai parameternya.

* Mendefiniskan fungsi didalam fungsi

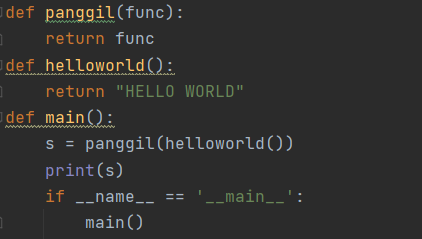
Pada kasus tertentu di dalam pemrograman, ada kalanya kita ingin membuat fungsi yang berada di dalam fungsi lain.



Pada kode diatas, fungsi next() menggunakan variabel li milik fungsi hitungmundur(). Elemen pertama dari *list* li diinisialkan dengan n. Setiap pemanggilan fungsi next(), nilai elemen pertama dari *list* li akan berkurang 1.

* Fungsi yang mengembalikan fungsi

Dalam Python, fungsi adalah objek. Dengan demikian, fungsi dapat digunakan sebagai parameter maupun nilai balik dari fungsi lain.



Contoh diatas menunjukan fungsi yang diperankan sebagai parameter dan nilai kembali.

* Membuat fungsi tanpa nama dengan operator lambda

Untuk fungsi sederehana yang mengembalikan nilai, anda dapat mendefinisikan menggunakan lambda. Fungsi seperti ini sering disebut *lambda function* atau fungsi tanpa nama.



Pada contoh diatas, kita melewatkan *lambda function* ke dalam fungsi sort(). Cara ini digunakan agar elemen elemen di dalam *list* diurutkan berdasarkan abjad tanpa mempedulikan huruf kecil atau besar.

* Rekursi

Suatu fungsi dikatakan sebagai fungsi rekursi jika di dalam badan fungsi tersebut terdapat pemanggilan terhadap dirinya



Contoh kasus yang paling sering digunakan untuk menerapkan konsep rekursi adalah pendefinisian fungsi untuk menghitung nilai faktorial dari suatu bilangan bulat non negatif.

Fungsi *Built In*

Salah satu faktor penentu yang dapat mempercepat proses belajar suatu bahasa pemrograman adalah dengan memahami penggunaan fungsi fungsi yang disediakan oleh bahasa pemrograman tersebut.

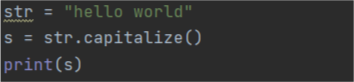
* Fungsi fungsi *String*

Pada subbab ini akan mempelajari dasar dasar penggunaan *string*. Kali ini kita akan membahas lebih jauh tentang penggunaan metode metode yang telah disediakan. Metode adalah fungsi yang didefinisikan di dalam suatu kelas.

1. Metode *Capitalize* ()

Bentuk umum penggunaan

*str.capitalize().*



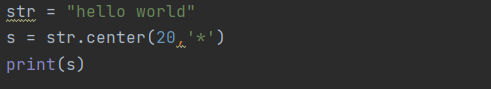
Metode ini digunakan untuk mengubah huruf pertama menjadi huruf kapital.

Pada pemrograman diatas, huruf dari kata pertama akan berubah menjadi huruf kapital (huruf besar).

1. Metode *Center* ()

Bentuk umum penggunaan

*center(width, fillchar)*

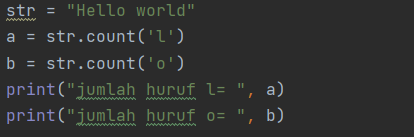


Parameter *width* adalah lebar *String* yang diinginkan, sedangkan *fillchar* adalah karakter yang akan mengisi ruang kanan dan kiri.

1. Metode *Count* ()

Bentuk umum penggunaan

*Str.count(substr, start=0, end=len(str))*

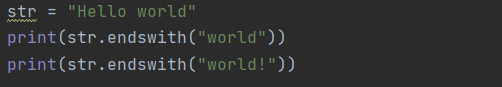


Metode ini akan menghitung berapa kali substr muncul di dalam stirng str, mulai dari indeks ke – start sampai end.

1. Metode *endswith* ()

Bentuk umum penggunaan

*Str.endswith(substr, start=0, end=len(str))*

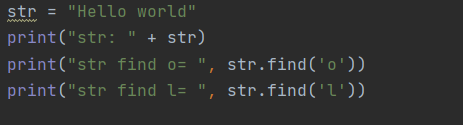


Metode ini berguna untuk memeriksa apakah string str dari indeks ke start sampai ke end diakhiri oleh substr atau tidak. Jika benar maka akan mengembalikan nilai True. Namun jika salah, akan mengembalikan nilai False.

1. Metode *Find* ()

Bentuk umum penggunaan

*Str.find(substr, start=0, end=len(str))*

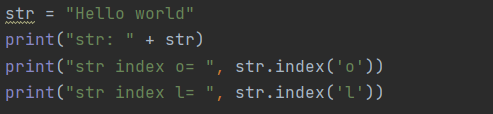
**

Metode ini digunakan untuk mencari substr di dalam stirng str. Jika ditemukan, metode akan mengembalikan nilai indeks atau posisi dari substirng bersangkutan. Jika tidak ditemukan, nilai yang akan dikembalikan adalah -1.

1. Metode *index* ()

Bentuk umum penggunaan

*Str.index(substr, start=0, end=len(str))*

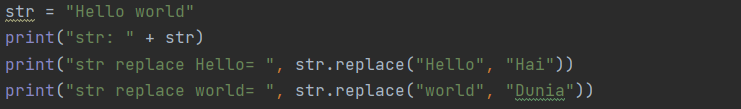
**

Metode ini fungsinya sama dengan *find* (), akan tetapi *index* () akan membangkitkan eksepsi (*error*) jika *substring* substr yang dicari tidak ditemukan

1. Metode *replace* ()

Bentuk umum penggunaan

*Str.replace(old, new[, max])*



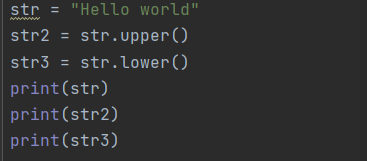
Metode ini digunakan untuk mengganti *substring* old menjadi new. Pada contoh diatas, kata “Hello” digantikan dengan kata “Hai”.

1. Metode *Upper* () dan *Lower* ()

Bentuk umum penggunaan

*str.upper* ()

*str.lower* ()



Metode ini digunakan untuk mengubah huruf dalam string str menjadi huruf besar maupun kecil. Metode upper () digunakan untuk huruf besar sedangkan metode lower digunakan untuk huruf kecil.

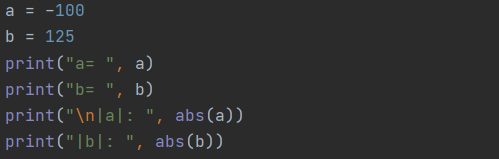
Fungsi fungsi Numerik

Selain string, banyak dijumpai dalam kasus kasus nyata di dunia pemrograman adalah numerik. Fungsi fungsi numerik digunakan untuk melakukan operasi operasi tertentu yang terhadap suatu bilangan.

1. Fungsi abs ()

Bentuk umum penggunaan

Abs(x)

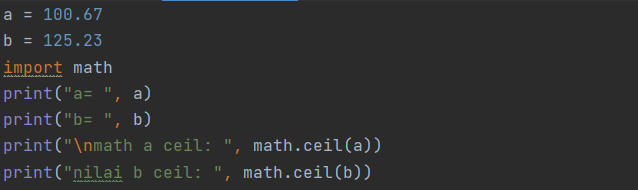


Fungsi ini akan mengembalikan nilai absolut (Harga mutlak) dari suatu billangan yang dilewatkan.

1. Fungsi ceil()

Bentuk umum penggunaan

Math.ceil()

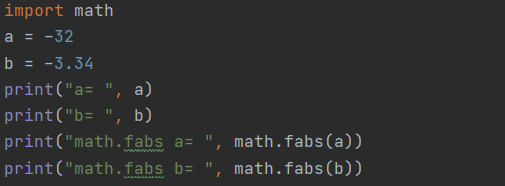


Fungsi ini digunakan untuk melakukan pembulatan bilangan riil ke bilangan bulat diatasnya.

1. Fungsi fabs ()

Bentuk umum penggunaan

Math.fabs()

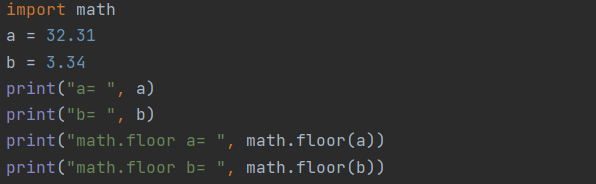


Fungsi ini akan mengembalikan nilai absolut dari suatu bilnagan, tujuannya hampir sama dengan fungsi abs().

1. Fungsi floor ()

Bentuk umum penggunaan

Math.floor ()



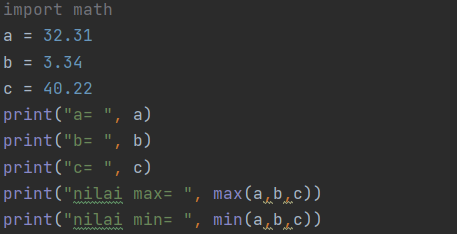
Fungsi ini digunakan untuk pembulatan bilangan riil ke bilangan bulat dibawahnya. Fungsi ini berbeda dengan fungsi ceil(), karena fungsi ini melakukan pembulatan kebawah.

1. Fungsi min() dan max()

Bentuk umum penggunaan

Min()

Max()

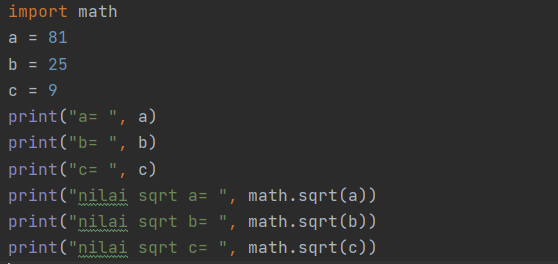


Fungsi ini digunakan untuk mencari nilai maximum maupun minimum dari suatu data.

1. Fungsi sqrt()

Bentuk umum penggunaan

Math.sqrt()

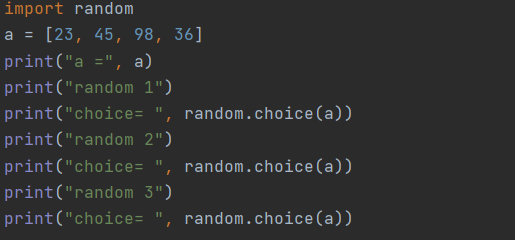


Fungsi ini digunakan untuk menghitung nilai akar kuadrat dari nilai nilai paramter.

1. Fungsi choice()

Bentuk umum penggunaan

Random.choice()



Fungsi ini digunakan untuk mengambil nilai acak dari suatu objek, list, maupun string. Setiap proses pemanggilan fungsi akan mengembalikan niali yang berbeda. Fungsi ini tersimpan dalam modul random.

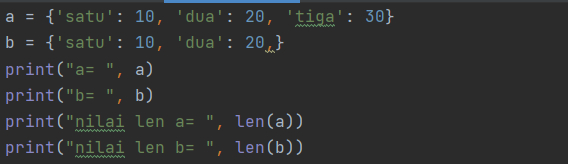
Fungsi fungsi *Dictionary*

Python menyediakan fungsi fungsi yang dapat diterapkan untuk melakukan operasi operasi terhadap suatu *dictionary.*

1. Fungsi len()

Bentuk umum penggunaan

Len()

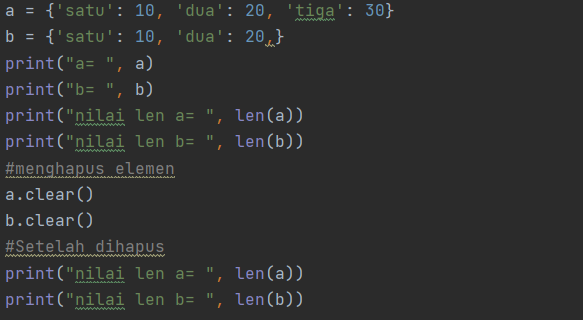


Fungsi ini digunakan untuk mengembalikan jumlah elemen di dalam *dictionary*

1. Metode clear()

Bentuk umum penggunaan

dict.clear()

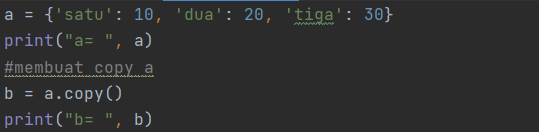


Metode ini digunakan untuk mengosongkan atau menghapus semua elemen didalam *dictionary*

1. Metode copy()

Bentuk umum penggunaan

dict.copy()



Fungsi ini digunakan untuk membuat objek baru yang merupakan salinan dari *dictionary*

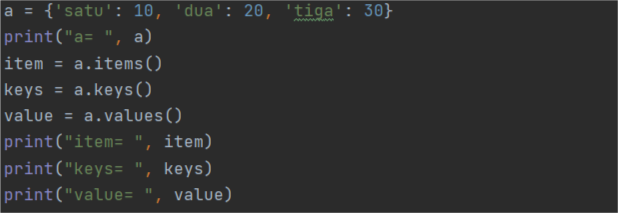
1. Metode items(), keys(), dan values()

Bentuk umum penggunaan

dict.items()

dict.keys()

dict.values()



Fungsi items digunakan untuk mengembalikan daftar pasangan kunci dan niali yang terdapat pada suatu *dictionary*. Metode keys digunakan untuk mendapatkan daftar kucninya saja, sedangkan metode values digunakan untuk mendapatkan daftar nilainya saja.

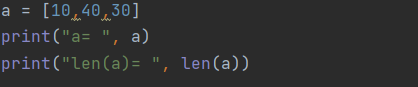
Fungsi fungsi list

Setelah *dictionary,* pada bagian ini kita akan membahas tentang fungsi fungsi yang dapat diterapkan pada objek list.

1. Fungsi len()

Bentuk umum penggunaan

Len()



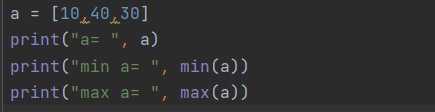
Fungsi len() akan mengembalikan jumlah atau banyaknya elemen yang terdapat list.

1. Fungsi min() dan max()

Bentuk umum penggunaan

Min()

Max()

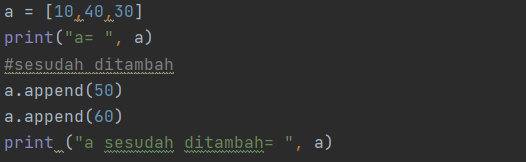


Fungsi ini digunakan untuk menentukan nilai minimal maupun maksimal didalam *list*

1. Metode append()

Bentuk umum penggunaan

List.append()

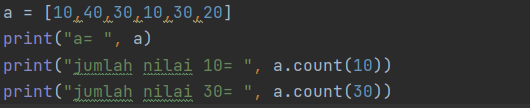


Metode ini digunakan untuk menambahkan objek kedalam list pada posisi terakhir

1. Metode count()

Bentuk umum penggunaan

List.count()

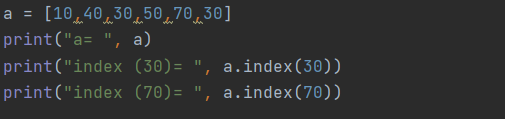


Metode ini berguna untuk menghitung jumlah atau banyaknya objek yang ada didalam list objek.

1. Metode index()

Bentuk umum penggunaan

List.index()

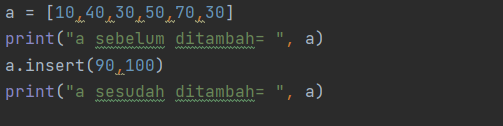


Metode ini digunakan untuk mengetahui indeks dari suatu objek dalam list objek. Jika objek muncul beberapa kali, maka yang kaan dikembalikan adalah indeks terendah.

1. Metode insert()

Bentuk umum penggunaan

List.insert()

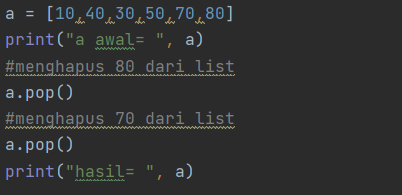


Metode ini digunakan untuk memasukan objek kedalam list objek. Indeks dalam list dimulai dari awal (0).

1. Metode pop()

Bentuk umum penggunaan

List.pop()

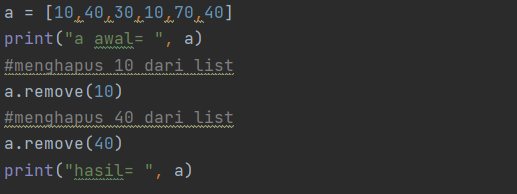


Metode ini digunakan untuk menghapus objek yang berada pada list. Objek yang terhapus dimulai dari indeks paling akhir.

1. Metode remove()

Bentuk umum penggunaan

List.remove()

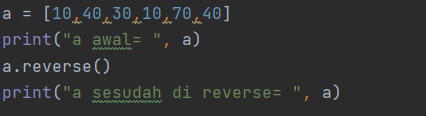


Metode ini digunakan untuk menghapus objek yang berada pada lisr objek. Jika objek lebih dari satu, maka yang akan dihapus adalah yang terletak pada indeks terendah

1. Metode reverse()

Bentuk umum penggunaan

List.reverse()

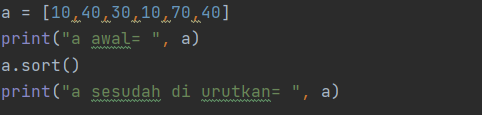


Metode ini digunakan untuk membalik urutan elemn di dalam list.

1. Metode sort()

Bentuk umum penggunaan

List.sort()



Metode ini digunakan untuk mengurutkan elemen di dalam list objek.

Jika ingin mengurutkan elemen list tanpa mengubah urutan aslinya, gunakan fungsi sorted().

