

Pertemuan 8

Library Fungsi built in



CAPAIAN PEMBELAJARAN

1. Mengimplementasikan program menggunakan library



KEBUTUHAN ALAT/BAHAN/SOFTWARE

1. Sistem Operasi Linux/Window 10
2. Python 3x
3. Teks editor



DASAR TEORI

Python adalah bahasa pemrograman yang powerful. Python dilengkapi dengan banyak fungsi built-in (bawaan) untuk melakukan berbagai hal.

Python 3.6 (versi terbaru python) memiliki 68 fungsi built-in. Tabel berikut menampilkan deskripsi dari semua fungsi tersebut.

Metode	Deskripsi
abs()	Mengembalikan nilai absolut dari suatu bilangan
all()	Mengembalikan True jika semua anggota data iterable bernilai True
any()	Menguji apakah satu atau lebih anggota iterable bernilai True
ascii()	Mengembalikan string yang berisi karakter yang bisa dicetak
bin()	Mengubah integer menjadi string biner
bool()	Mengubah suatu nilai menjadi nilai Boolean
bytearray()	Mengembalikan array dari byte
bytes()	Mengembalikan objek byte immutable
callable()	Menguji apakah sebuah objek bisa dipanggil
chr()	Mengembalikan karakter (string) dari suatu integer
classmethod()	Mengembalikan metode class untuk fungsi yang diberikan

compile()	Mengembalikan kode objek Python
complex()	Menciptakan Bilangan Kompleks
delattr()	Menghapus attribut dari objek
dict()	Menciptakan dictionary
dir()	Mengembalikan atribut objek
divmod()	Mengembalikan tuple dari hasil bagi dan sisa
enumerate()	Mengembalikan objek enumerasi
eval()	Menjalankan kode python dalam program
exec()	Mengeksekusi program secara dinamis
filter()	Menghasilkan iterator dari anggota – anggota yang bernilai True
float()	Mengembalikan bilangan float dari bilangan atau string
format()	Mengembalikan string yang terformat
frozenset()	Mengembalikan objek frozenset immutable
getattr()	Mengembalikan nilai dari atribut objek
globals()	Mengembalikan dictionary table simbol global
hasattr()	Menguji apakah objek memiliki atribut tertentu
hash()	Mengembalikan nilai hash dari suatu objek
help()	Memanggil sistem help built-in
hex()	Mengubah integer menjadi hexadecimal
id()	Mengembalikan identitas objek
input()	Membaca masukan dan mengembalikan string
int()	Mengembalikan integer dari suatu bilangan atau string
isinstance()	Menguji apakah suatu objek merupakan instance dari class
issubclass()	Menguji apakah suatu objek merupakan subclass dari class
iter()	Mengembalikan iterator objek
len()	Mengembalikan panjang objek
list()	Menciptakan list baru
locals()	Mengembalikan dictionary dari tabel simbol lokal
map()	Mengaplikasikan fungsi terhadap list dan mengembalikan list baru
max()	Mengembalikan nilai terbesar
memoryview()	Mengembalikan view memori dari argumen
min()	Mengembalikan nilai terkecil
next()	Mengambil nilai selanjutnya dari iterator
object()	Menciptakan objek
oct()	Mengubah integer menjadi octal
open()	Mengembalikan objek file
ord()	Mengembalikan kode unicode dari karakter
pow()	Mengembalikan pangkat bilangan
print()	Mencetak objek yang diberikan
property()	Mengembalikan atribut properti
range()	Mengembalikan integer yang berurut dari start sampai stop
repr()	Mengembalikan karakter yang bisa dicetak dari suatu objek
reversed()	Mengembalikan iterator sequence yang dibalik
round()	Membulatkan bilangan berkoma

set()	Mengembalikan set
setattr()	Mengeset nilai dari atribut objek
slice()	Menciptakan objek terpotong sesuai range
sorted()	Mengembalikan list yang disortir
staticmethod()	Menciptakan metode static dari suatu fungsi
str()	Mengembalikan string dari objek
sum()	Menambahkan semua anggota dari iterable
super()	Mengacu ke parent class dari objek
tuple()	Menciptakan tuple
type()	Mengembalikan tipe objek
vars()	Mengembalikan atribut __dict__ dari kelas
zip()	Mengembalikan iterator dari tuple
import()	Fungsi lanjut yang dipanggil menggunakan import



PRAKTIK

1. Fungsi abs()

Fungsi abs() mengembalikan nilai absolut dari x

```

1  # fungsi abs()
2  # praktik1.py
3  # nim :.....
4  # nama :.....
5  #*****
6  print("fungsi abs(-10)   : ", abs(-10))
7  print("fungsi abs(20.32) : ", abs(20.32))
8  print("fungsi abs(-1000) : ", abs(-1000))

```

Amati Hasilnya:

2. Fungsi all()

iterable – data iterable (string, list, tuple, dictionary, dan lain-lain), True – apabila semua anggota di dalam iterable bernilai True, False – bila ada anggota dalam iterable bernilai False

```

1  #####
2  # praktik82.py
3  # nim :.....
4  # nama :.....
5  #####
6
7  listAngka = [1, 2, 3, 10]
8  print("all([1, 2, 3]) = ",all(listAngka))
9  listAngka = [1, False, 3]
10 print("all([1, False, 3]) = ",all(listAngka))
11 listTeks = []
12 print("all( [] )=",all(listTeks))
13 # string
14 teks = 'UNIVERSITAS TEKNOLOGI YOGYAKARTA'
15 print("all('UNIVERSITAS TEKNOLOGI YOGYAKARTA') = ",all(teks))
16 teks = '012'
17 print(all(teks))
18 print(all([3,2,1,0,'A']))

```

Amati Hasilnya:

3. Fungsi any(),

Fungsi any() mengembalikan True jika ada anggota iterable yang bernilai True. Bila tidak ada satupun, maka any() mengembalikan False.

```

1  #####
2  # praktik83.py
3  # nim :.....
4  # nama :.....
5  #####
6
7  listAngka = [1, 2, 3, 10]
8  print("any([1, 2, 3]) = ",any(listAngka))
9  listAngka = [1, False, 3]
10 print("any([1, False, 3]) = ",any(listAngka))
11 listTeks = []
12 print("any( [] )=",any(listTeks))
13 # string
14 teks = 'UNIVERSITAS TEKNOLOGI YOGYAKARTA'
15 print("all('UNIVERSITAS TEKNOLOGI YOGYAKARTA') = ",any(teks))
16 teks = '012'
17 print(any(teks))
18 print(any(['',3,2,1,0,'B']))

```

Amati Hasilnya:

4. Fungsi bin(),

Fungsi bin() mengubah integer dan mengembalikan nilai dalam bentuk string biner.

```
1  #####
2  # praktik84.py
3  # nim :.....
4  # nama :.....
5  #####
6  angka=int(input("Angka Bilangan bulat = "))
7  print("Bilangan ",angka," ke bilangan biner = ",bin(angka))
8
```

Amati Hasilnya:

5. Fungsi chr():

fungsi chr() mengembalikan karakter unicode dari bilangan integer (mengubah integer menjadi string)

```
1  #####
2  # praktik85.py
3  # nim :.....
4  # nama :.....
5  #####
6  print("chr(65)  = ",chr(65))
7  print("chr(97)  = ",chr(97))
8  print("chr(1239) = ",chr(1239))
9  for x in range(65,123):
10 |   print(chr(x), end = " ")
11
12  for x in range(65,123):
13 |   print("chr(",x,")=",chr(x))
```

Amati Hasilnya:

6. Fungsi dict(),

Fungsi dict() berfungsi untuk membuat dictionary.

```

1  #####
2  # praktik85.py
3  # nim :.....
4  # nama :.....
5  #####
6  angka = dict(x=5, y=0)
7  print('angka =', angka)
8
9  # dict menggunakan iterable
10 angka1 = dict([('x', 5), ('y', -5)])
11 print('angka1 =', angka1)
12
13 # dict menggunakan mapping
14 angka2 = dict({0: {'x':4, 'y':5}, 1: {'x':6, 'y':100}})
15 print('angka2 =', angka2)

```

Amati Hasilnya:

7. Fungsi enumerate() :

enumerate() mengembalikan objek enumerate yaitu objek iterable yang tiap itemnya berpasangan dengan indeks.

```

1
2  #####
3  # praktik87.py
4  # nim :.....
5  # nama :.....
6  #####
7  hari = ['Senin', 'Selasa', 'Rabu', 'Kamis', 'Jumat', 'Sabtu', 'Minggu']
8  urutHari = enumerate(hari)
9  print(type(urutHari))
10 print(list(urutHari))
11
12 for x in urutHari:
13     | print(x)
14
15 for i, hari in enumerate(hari, 1) :
16     | print (i, hari)

```

Amati Hasilnya:

8. Fungsi eval() :

fungsi `eval()` berfungsi untuk memarsing (menguraikan) string ekspresi yang dilewatkan ke dalamnya, dan menjalankannya sebagai ekspresi Python murni.

```
1  #*****
2  # praktik88.py
3  # nim :.....
4  # nama :.....
5  #*****
6  from math import *
7
8  def membuatFungsi():
9      # ekspresi
10     expr = input("Masukkan contoh : x*(x+1)*(x+2) : ")
11     # nilai variabel x
12     x = int(input("Jika nilai x : "))
13     # menggunakan eval
14     y = eval(expr)
15     # cetak hasil
16     print("y = {}".format(y))
17
18 if __name__ == "__main__":
19     membuatFungsi()
```

Amati Hasilnya:

9. Fungsi `pow()`:

Fungsi `pow()` berfungsi untuk mendapatkan nilai pangkat dari suatu bilangan.

sintaks : `pow(x, y[, z])`

x – bilangan yang akan dipangkatkan

y – pangkat

z (opsional) – modulus

```

1  #*****
2  # praktik89.py
3  # nim :.....
4  # nama :.....
5  #*****
6  # 2 pangkat 2
7  print(pow(2, 2))
8  # -2 pangkat 2
9  print(pow(-2, 2))
10 # 2 pangkat -2
11 print(pow(2, -2))
12 # 25 pangkat 0.5
13 print(pow(9, 0.5))
14 # (7**2) % 5
15 print(pow(7, 2, 5 ))

```

Amati Hasilnya:



LATIHAN

Buatlah program dengan penerapan kasus bebas, untuk menggunakan fungsi berikut:

1. fungsi slice()
2. fungsi filter()
3. Fungsi callable()
4. Fungsi round()
5. Fungsi divmod()



REFERENSI

<https://www.pythonindo.com/fungsi-sum/>