

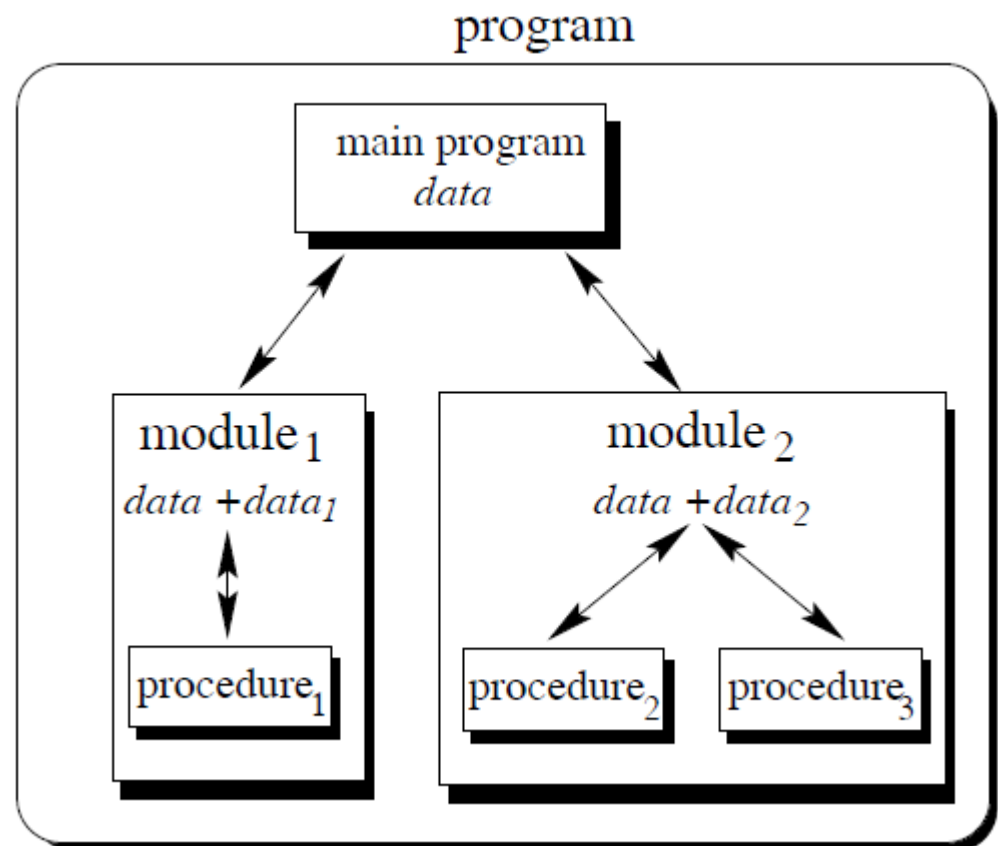
Python Fundamental

Ch.8 : Modul

Instruktur : Ahmad Rio Adriansyah
Nurul Fikri Komputer

Pemrograman Modular

- Teknik desain software yang menekankan pada pemisahan fungsionalitas program ke dalam bagian (modul) independent.
- Tiap modul mengandung fungsionalitas tertentu



Contoh

- Mau membuat software untuk menampilkan peta wilayah yang berkaitan dengan
 - Visualisasi peta
 - Kalkulasi jarak
 - Konversi derajat ke kilometer dan sebaliknya
 - Database lokasi daerah tertentu
- Jika dibuat dalam sebuah file akan *overwhelming* bagi programmer maupun penggunaanya.
- Fungsi fungsi yang mirip dikelompokkan dan bisa dimasukkan dalam sebuah modul yang sama

Modul Python

- Sebuah modul adalah sebuah file yang berisi definisi-definisi dan statemen-statementen python
- Nama file adalah nama modul ditambah ekstensi '.py'
- Python menyediakan banyak modul yang dapat digunakan (baik yang bawaan atau perlu diinstall dulu)
- User (programmer) juga dapat membuat modul sendiri

Modul Python

- Sebelum definisi dan statemen pada modul yang dimaksud dapat digunakan, harus dipanggil terlebih dahulu dengan perintah import

```
>>> import nama_modul
```

- Perintah **import** akan mengarahkan python untuk mencari dan menemukan file **nama_modul.py**, mengeksekusi, memasukkan seluruh objek, dan mendefinisikannya ke dalam namespace modul baru (`__name__`). Sementara itu, objek modul `nama_modul` dimasukkan ke dalam current namespace (`__main__`)
- Objek pada modul tersebut dipanggil dengan atribut reference (tanda titik setelah nama modul)

Modul Python

- Atau cara lainnya jika ingin import tanpa atribut reference dapat menggunakan

```
>>> from nama_modul import nama_fungsi
```

Modul Buatan Sendiri

- Buat sebuah file dengan nama modulku.py yang berisi :

```
def cetak(item):
```

```
    print (item)
```

```
def kali(x,y):
```

```
    print (x*y)
```

```
a = 10
```

```
b = 'sepuluh'
```

Modul Buatan Sendiri

- Buka python shell di directory yang sama dengan file **modulku.py** (atau ganti working directorynya ke folder tersebut)

```
>>> from os import getcwd, chdir, listdir
```

```
>>> getcwd()
```

```
>>> chdir('path/ke/folder')
```

```
>>> getcwd()
```

```
>>> listdir()
```


Modul Buatan Sendiri

- Import modul modulkmu yang tadi sudah dibuat

```
>>> from modulkmu import *
```

```
>>> cetak('ini dipanggil dari program lain')
```

```
>>> kali(2,10)
```

```
>>> print(a)
```

- Fungsi dan nilai yang ada pada modul tersebut dapat diakses dari program utama yang sedang kita jalankan

Modul Buatan Sendiri

- Agar modul yang kita buat tidak hanya bisa digunakan untuk diimport tapi juga bisa dijalankan, kita bisa tambahkan baris kode berikut untuk main programnya

```
if __name__ == "__main__":  
    expression
```

- Ekspresi di bawah kode tersebut hanya akan dijalankan jika filenya dipanggil, tapi tidak akan dijalankan jika filenya diimport sebagai modul

Modul Bawaan Python

- Kita akan mencoba eksplorasi beberapa modul berikut :

os, sys

string, math

datetime

urllib / urllib2 / requests

beautiful soup

OS, Sys

- Digunakan untuk mengakses sistem
- OS telah ditunjukkan pada bagian sebelumnya. Di bawah ini akan ditunjukkan beberapa objek pada modul sys

```
>>> import sys
```

```
>>> sys.version
```

```
>>> sys.ps1 #primary prompt
```

```
>>> sys.ps2 #secondary prompt
```

String

- Modul untuk memanipulasi string

```
>>> import string
```

```
>>> string.digits
```

```
>>> string.lowercase
```

```
>>> string.uppercase
```

```
>>> string.split('Nurul Fikri Computer')
```

```
>>> string.center('Selamat Datang',100)
```

```
>>> string.count('halo apa kabar?', 'a', 4)
```

Math

- Modul untuk operasi-operasi matematis

```
>>> import math
```

```
>>> math.pow(2,3)
```

```
>>> math.sqrt(25)
```

```
>>> math.log10(1000)
```

```
>>> math.log(10)
```

```
>>> math.fmod(17,3)
```

Datetime

- Modul datetime adalah bawaan python
- Digunakan untuk hal-hal yang berkaitan dengan waktu (tahun, bulan, tanggal, jam, menit, detik)

```
>>> import datetime
```

```
>>> datetime.datetime.now()
```

```
>>> print('sekarang tahun %d'  
%datetime.datetime.now().year)
```

Datetime

- Kita coba untuk mengukur waktu eksekusi kode python, bandingkan antara waktu rekursif dan non rekursif fungsi fibonacci

```
>>> start = datetime.datetime.now()  
#letakkan sebelum proses dimulai
```

```
>>> fin = datetime.datetime.now()  
#letakkan setelah proses selesai
```

```
>>> delta = fin-start
```

```
>>> print(delta.__str__())
```


Datetime

- Selisih dari datetime bertipe datetime.timedelta (bisa dicek menggunakan fungsi type())
- Sebaliknya, kita bisa menambahkan objek bertipe datetime.timedelta ke datetime.datetime
- Contoh : misalnya kita ingin mengetahui tanggal berapa 100 hari setelah hari ini?

```
>>> seratus = datetime.datetime.now() +  
datetime.timedelta(days=100)
```

```
>>> print(seratus.__str__())
```

Datetime

- Kita coba import object date dari modul datetime agar dapat digunakan tanpa atribut reference

```
>>> from datetime import date
```

```
>>> hari_ini = date.today()
```

```
>>> print(hari_ini.weekday())
```

```
#akan memunculkan hari apa dalam integer
```

```
#0 = senin, 6 = ahad
```

Latihan

- Buat fungsi yang dapat :
 - Menentukan hari kelahiran jika diberikan tanggalnya
 - Mengetahui pada umur berapa hari ulang tahunnya akan jatuh di hari yang sama dengan hari lahirnya

Urllib dan Beautiful Soup

- Kedua modul ini akan digunakan bersamaan
- **Urllib** modul bawaan python (ada variannya **urllib2**, **urllib3**, atau **requests** yang harus diinstall dahulu) digunakan untuk berkomunikasi dengan internet melalui protokol http
- Direkomendasikan menggunakan urllib3 atau requests. Contoh akan menggunakan **requests**.
- **Beautiful Soup** adalah modul yang digunakan untuk mengolah hasil respon dari http

Repository dan Package Manager

- Modul yang umum disimpan dalam repository
 - Salah satunya adalah PyPI (Python Package Index)
- Untuk mengatur modul, menambahkan, atau menghapus, digunakan package manager
 - Diantaranya pip, easy_install, yum, conda
 - Pip adalah package manager standard python
- Menginstall modul baru dari repository menggunakan pip
`$ pip install nama_modul`

Requests dan Beautiful Soup

- Install modul tersebut menggunakan pip atau package manager lain

```
$ pip install requests
```

```
$ pip install bs4
```

- Masuk ke python shell

```
$ python
```

```
>>>
```

Http Request

```
>>> import requests  
>>> url = 'http://big.go.id'  
>>> req = requests.get(url)  
>>> req.status_code  
>>> req.headers  
>>> req.encoding  
>>> req.text
```

Beautiful Soup

```
>>> from bs4 import BeautifulSoup as BS  
>>> laman = req.text  
>>> soup = BS(laman)  
>>> print(soup)  
>>> print(soup.prettify())
```


Beautiful Soup

- Mencari bagian laman yang menggunakan tag tertentu (misal tag script)

```
>>> soup.find('script')
```

- Mencari semua tag tertentu di laman tersebut

```
>>> soup.find_all('script')
```

Latihan

- Pada html, link disimpan dalam tag a
- Coba buka sebuah laman web yang sering dikunjungi
- Tunjukkan semua link yang ada pada laman tersebut mengarah ke mana saja