

# [Kelas B+D] Lab 1 Dasar-Dasar Pemrograman 1

## Introduction to IDLE; Variable, Data Types, and Operator

**DEADLINE: 8 September 2017, pukul 15.30 waktu SCell**

### Selamat datang di Lab/Tutorial Dasar-Dasar Pemrograman 1!

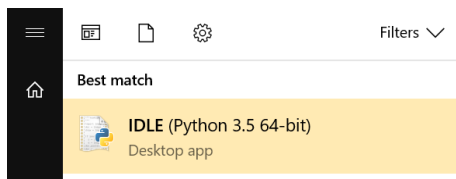
Pertama-tama kami ucapkan selamat kepada kalian semua para peTARUNG yang sudah dianugerahi kesempatan untuk mengejar sarjana di Fasilkom UI. Percayalah, bahwa langkah pertama meraih S.Kom *pasti* bermula dari Dasar-Dasar Pemrograman. Maka dari itu DDP sebaiknya diikuti dengan seksama dan jangan kendor semangatnya! :)

## Introduction to IDLE

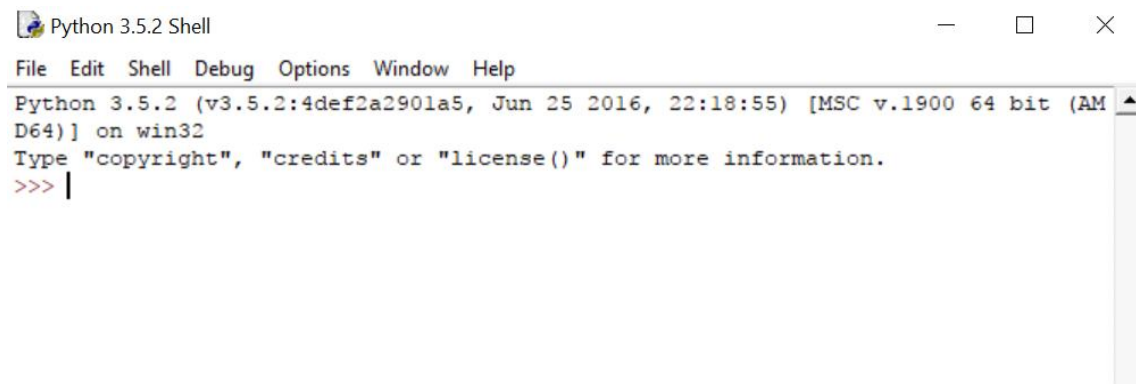
Selamat datang para peTARUNG di Dunia Python! Untuk memulai petualangan kalian selama satu semester ini, kalian tentu memerlukan senjata untuk mampu menaklukkan Dunia Python. Oleh karena itu, kami, para veteran Python, akan memberikan senjata pertama yang akan kalian gunakan untuk mengarungi petualangan ini. Senjata yang akan kalian gunakan untuk petualangan kali ini adalah **Python IDLE**. Untuk menjadi seorang peTARUNG yang hebat, tentunya kalian harus menguasai bagaimana cara menggunakan senjata kalian.

### a. Membuka IDLE (*Interactive Development Learning Environment*)

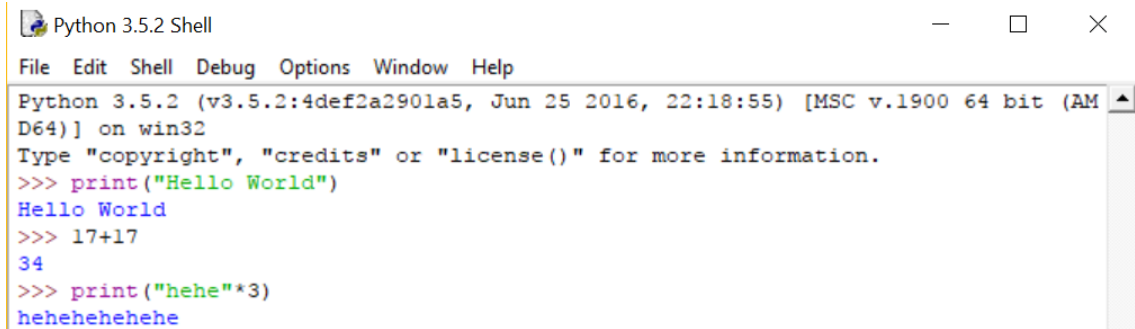
Untuk membuka IDLE, cukup ketik *keyword* "IDLE" pada *search field*, lalu pilih **IDLE Python 3.x 64-bit/32-bit** (Catatan: Pastikan kalian sudah meng-*install* Python 3.x 64-bit/32-bit pada laptop kalian, jika kalian menggunakan laptop).



Lalu akan muncul window IDLE seperti di bawah ini:



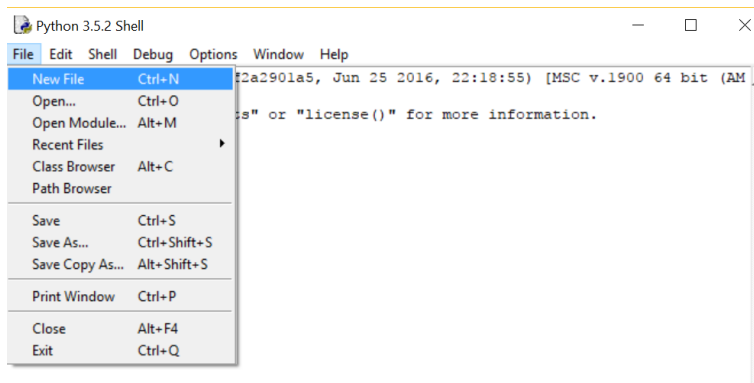
Kalian dapat menuliskan perintah pemrograman di samping tanda ">>>" seperti pada contoh berikut ini:



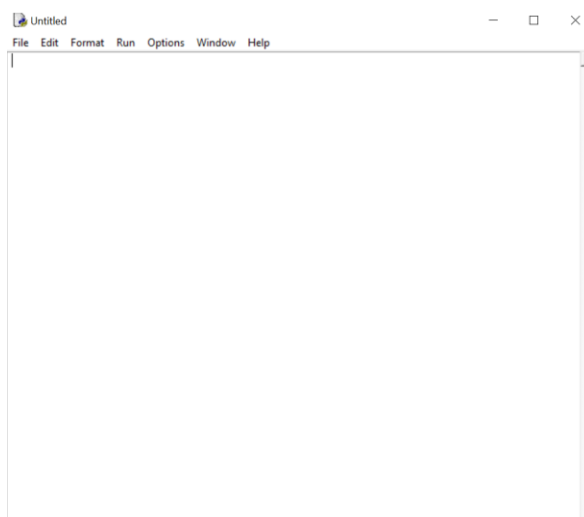
```
Python 3.5.2 Shell
File Edit Shell Debug Options Window Help
Python 3.5.2 (v3.5.2:4def2a2901a5, Jun 25 2016, 22:18:55) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> print("Hello World")
Hello World
>>> 17+17
34
>>> print("hehe"*3)
hehehehehehe
```

## b. Membuat *file* Python

Untuk membuat *file* Python, kalian dapat meng-klik **File -> New File** atau menggunakan *shortcut* **Ctrl + N**.



Lalu akan muncul window seperti di bawah ini:



Disini, kalian bisa mengetikkan barisan kode yang kalian inginkan. **Perbedaannya adalah program yang kalian ketikkan tidak bisa dijalankan secara langsung hanya dengan menekan "Enter".** Bagaimana cara menjalankan program yang telah kalian buat akan kami jelaskan di bagian berikutnya.

Coba kalian masukkan potongan kode program di bawah ini:

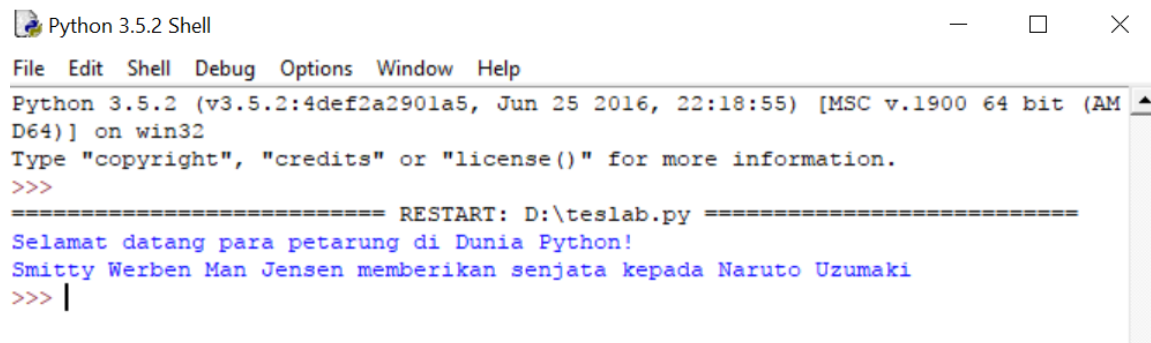
```
print("Selamat datang para petarung di Dunia Python!")
nama_veteran = "Smitty Werben Man Jensen"
nama_petarung = "Naruto Uzumaki"
print(nama_veteran + " memberikan senjata kepada " +
nama_petarung)
```

#### c. Menyimpan *file* Python

Untuk menjalankan program yang telah kalian ketikkan di atas, kalian perlu menyimpan file tersebut terlebih dahulu dengan cara **File -> Save As** atau kalian dapat menggunakan *shortcut* **Ctrl + Shift + S**, beri nama file sesuai dengan keinginan kalian, lalu klik **Save**.

#### d. Menjalankan program

Untuk menjalankan program melalui IDLE, kalian dapat mengklik **Run -> Run Module** atau kalian dapat menggunakan *shortcut* **F5**. Potongan kode di atas akan menghasilkan *output* sebagai berikut:



```
Python 3.5.2 Shell
File Edit Shell Debug Options Window Help
Python 3.5.2 (v3.5.2:4def2a2901a5, Jun 25 2016, 22:18:55) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:\teslab.py =====
Selamat datang para petarung di Dunia Python!
Smitty Werben Man Jensen memberikan senjata kepada Naruto Uzumaki
>>> |
```

Pada Python, ada 2 *method* dasar untuk mencetak output dan menerima input dari user. Kedua *method* tersebut adalah **print()** dan **input()**.

**SELAMAT!** Kalian telah menyelesaikan tahap pengenalan senjata kalian, Python IDLE! Selanjutnya, kalian akan mempelajari jutsu pertama dalam Python, yaitu **Variable-no-Jutsu**, **Data-Type-no-Jutsu**, dan **Operator-no-Jutsu**.

## Variable, Data Types, and Operator

## 1. Variable

Variabel bisa dianggap sebagai alokasi memori di dalam komputer yang digunakan untuk menyimpan suatu nilai. Nilai tersebut dapat berupa bilangan, karakter, dan lain sebagainya yang akan dibahas selanjutnya di bagian tipe data.

Contoh pembuatan dan penggunaan variabel:

```
>>> nilai_kuis_terakhir = 70 # Membuat variabel bernilai 70 tipe data integer
>>> print(nilai_kuis_terakhir) # Cetak nilai variabel tersebut
70
>>> nilai_kuis_terakhir = 100 # Ubah nilai variabel tersebut
>>> print(nilai_kuis_terakhir) # Cetak nilai variabel tersebut
100
>>> nilai_kuis_terakhir = 'A' # Variabel di Python bersifat dinamis sehingga tipe datanya bisa berubah-ubah sesuai nilai yang dia
>>> print(nilai_kuis_terakhir) # Cetak nilai variabel tersebut
A
```

Untuk membuat suatu variabel, kita harus mengerti aturan penamaan variabel. Salah satu aturan penamaan variabel yang paling penting adalah **variabel harus berawal dengan huruf alfabet atau underscore '\_'**. Variabel tidak boleh berawal dengan karakter selain itu, seperti angka! Variabel di Python juga bersifat **case sensitive** yang artinya huruf besar dan huruf kecil berpengaruh (contoh: variabel Nama dan nama dianggap berbeda).

Contoh penamaan variabel yang **valid**: nama\_123, umur, tinggi, \_role

Contoh penamaan variabel yang **tidak valid**: 123\_nama, .umur, #nama

Penamaan variabel juga sebaiknya sejasas mungkin, agar kode kalian mudah dibaca dan mempermudah proses *debugging* (proses menemukan kesalahan dalam kode kalian). Untuk gaya penulisan variabel dengan dua kata atau lebih, kalian dapat menggunakan salah satu dari beberapa cara berikut ini:

1. *Snake case*. Contoh: nama\_depan
2. *Camel case*. Contoh: namaDepan
3. *Pascal case*. Contoh: NamaDepan

Pada bahasa Python lebih disarankan menggunakan penamaan **snake case**. Sedangkan pada bahasa lain seperti Java penamaan *camel case* lebih sering digunakan untuk nama variabel dan penamaan *pascal case* untuk nama *class*. Untuk *best practice style guide* Python bisa dibaca lebih lanjut di:

<https://www.python.org/dev/peps/pep-0008/>

## 2. Data Types

Tipe data (*data type*) adalah jenis dari suatu nilai (*value*), contohnya seperti *integer* (bilangan bulat) atau *string* (karakter) ataupun *float* (bilangan real).

Dalam bahasa Python beberapa tipe data yang biasa digunakan adalah sebagai berikut:

1. Integer, contoh: 10, 100
2. String, contoh: "Kirigaya Kazuto", "Yuuki Asuna"
3. Float, contoh: 0.25, 10.5, 99.75
4. Boolean, hanya dapat bernilai dua kemungkinan yaitu True atau False
5. List, contoh: [1, 2, 3, 4, 5]

6. Dictionary, contoh: {"umur": 17, "nama": "Asuna"}

Selain itu, kalian dapat menggunakan fungsi **type()** untuk mengetahui jenis tipe data dari suatu nilai. Contoh penggunaannya adalah sebagai berikut:

```
>>> angka = 1000
>>> type(angka)
<class 'int'>
>>> type('kalimat')
<class 'str'>
```

Memahami tipe data dalam Python sangat penting karena tipe data mendefinisikan operasi-operasi yang bisa dilakukan terhadap nilai tersebut.

**Perlu diketahui bahwa di Python, tipe data nilai suatu variabel dapat berubah-ubah (*dynamic*), jadi kalian harus berhati-hati.**

### 3. Operator

Operator adalah sebuah simbol yang digunakan untuk mengubah nilai operan. Ada banyak operator yang dapat kita gunakan, salah satunya adalah operator aritmatika seperti berikut:

#### Arithmetic Operators

Pertambahan (+)	x = 3 + 2 # x bernilai 5
Pengurangan (-)	x = 3 - 2 # x bernilai 1
Perkalian (*)	x = 3 * 2 # x bernilai 6
Pembagian (/)	x = 3 / 2 # x bernilai 1.5
Modulus / Sisa Pembagian (%)	x = 3 % 2 # x bernilai 1
Pemangkatan (**)	x = 3 ** 2 # x bernilai 9
Pembagian Integer (//)	x = 3 // 2 # x bernilai 1

Meskipun operator-operator tersebut adalah operator aritmatika yang biasanya hanya bisa dioperasikan untuk tipe data bilangan (*integer / float*), ada beberapa operator yang bisa digunakan untuk tipe data yang lain pula, contohnya adalah *string*.

```
>>> hehe = 'hehe'
>>> haha = 'haha'
>>> print(hehe + haha + '.') # Penjumlahan string dengan string
hehehaha.
>>> print(hehe * 5) # Perkalian string dengan angka, string 'hehe'
akan direpetisi 5 kali
hehehehehehehehehehe
```

Masih terdapat banyak operator lain seperti **logical operators** (*and, or, >, <, ==*) dan **bitwise operators** (*<<, >>, &, |, ~, ^*) yang kalian bisa pelajari sendiri (atau sudah pelajari di kelas) namun tidak akan dibahas di tutorial kali ini.

#### 4. Type Casting (Mengubah tipe data)

Dalam bahasa Python, sangat penting untuk mengetahui cara mengubah tipe data dari sebuah nilai ke tipe data yang lain.

Khususnya, karena dalam bahasa Python untuk menerima input dari user menggunakan fungsi **input()** yang pasti mengembalikan nilai bertipe data **string**. Sedangkan, terkadang kita memerlukan input berupa bilangan bulat ataupun bilangan real.

Perlu diketahui bahwa sangat penting untuk mengoperasikan nilai apabila tipe data nya sudah sama. Ekpresi seperti '1' + 1 tidak akan menghasilkan 2 namun akan menghasilkan *error*. Oleh karena itu, kita perlu mengetahui cara untuk mengubah suatu tipe data menjadi tipe data yang lain.

Beberapa cara untuk mengubah tipe data:

- **int** (*sesuatu*) akan menghasilkan *sesuatu* bertipe data integer
- **float** (*sesuatu*) akan menghasilkan *sesuatu* bertipe data floating number
- **str** (*sesuatu*) akan menghasilkan *sesuatu* bertipe data string

Catatan: *sesuatu* di sini bisa merupakan variabel atau suatu nilai

Saat mengubah tipe data, kita harus berhati-hati karena operasi pengubahan tipe data bisa menghasilkan suatu *exception*.

---

## Python Turtle Library

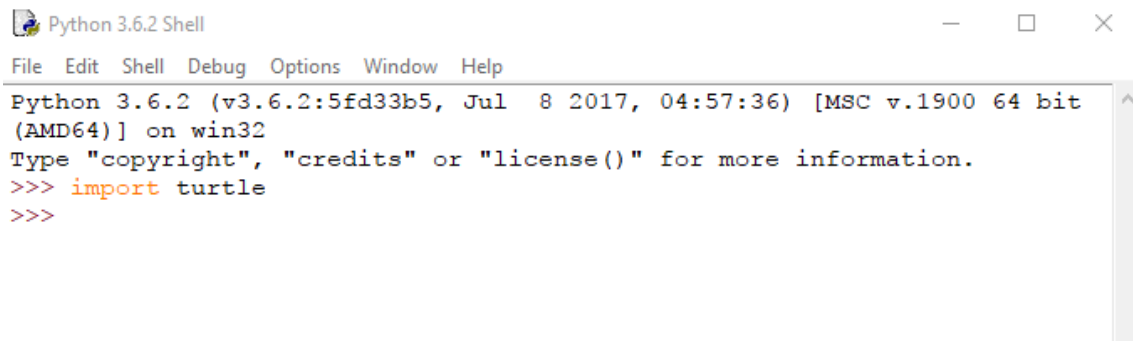
Pusing dengan materi-materi di atas? Tenang. Belajar Python itu menyenangkan, mari kita belajar menggambar dengan Turtle dalam Python!

### Menggambar Dengan Turtle

Turtle adalah sebuah module built-in yang disediakan oleh Python yang digunakan untuk menggambar. Ibaratnya adalah seperti ada seekor kura-kura yang ditemplei dengan pulpen di bawahnya, lalu kita beri instruksi apa yang akan dilakukan si kura-kura itu.

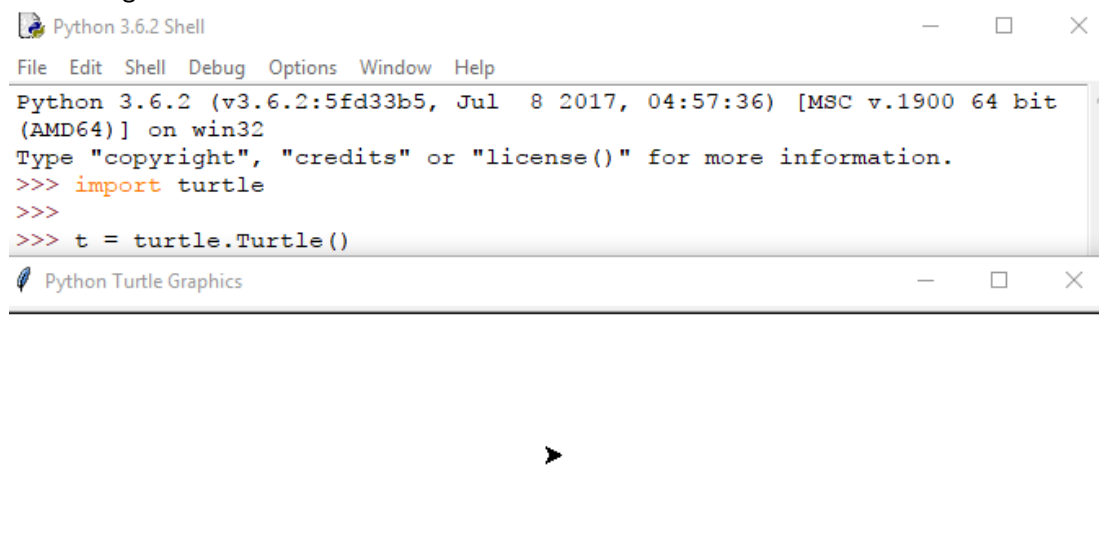
#### a. Import Module Turtle

Untuk bisa mengakses fitur dari module turtle ini, pertama-tama kita harus meng-*import* modulnya dengan cara memasukkan keyword **import** **turtle**



```
Python 3.6.2 Shell
File Edit Shell Debug Options Window Help
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:57:36) [MSC v.1900 64 bit
(AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> import turtle
>>>
```

Untuk bisa menggunakan turtle, kita harus membuat *instance variable* dari modul turtle dengan cara sebagai berikut:



```
Python 3.6.2 Shell
File Edit Shell Debug Options Window Help
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:57:36) [MSC v.1900 64 bit
(AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> import turtle
>>>
>>> t = turtle.Turtle()

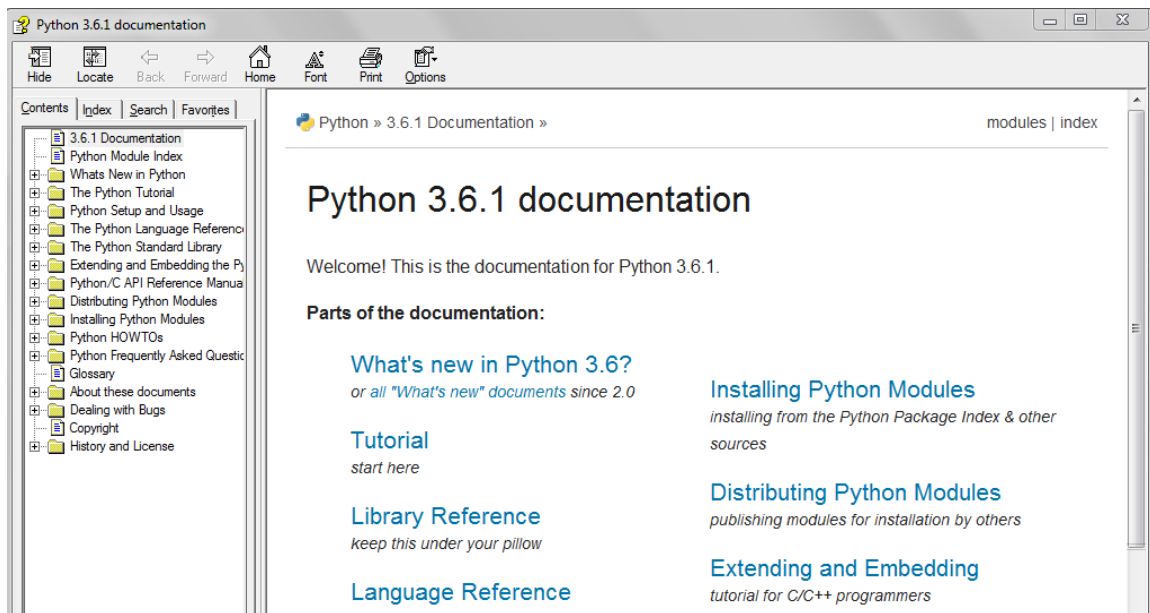
Python Turtle Graphics
```

Maka akan muncul **Python Turtle Graphics** yang akan berfungsi sebagai **Canvas** yang digunakan sebagai tempat menggambar.

#### b. Fungsi-fungsi di Module Turtle

Ada beberapa fungsi yang bisa kalian gunakan untuk mengoperasikan turtle yang telah kalian buat, beberapa diantaranya adalah:

- I. **pendown ()**  
Fungsi ini digunakan untuk mengaktifkan pena, ibaratnya pena tersebut dijatuhkan mengenai kanvas sehingga bisa dipakai untuk menggambar
- II. **penup ()**  
Fungsi ini digunakan untuk menon-aktifkan pena, sehingga kalian bisa memindahkan turtle ke koordinat lain tanpa meninggalkan coretan/jejak di kanvas
- III. **forward(x)**  
Fungsi ini digunakan untuk menggerakkan turtle ke depan sejauh x pixel dan meninggalkan jejak berupa garis **jika** pena diaktifkan (**pendown ()**).
- IV. **left(x) & right(x)**



Fungsi ini digunakan untuk mengubah haluan dari turtle itu sejauh x derajat ke kiri (**left(x)**) atau ke kanan (**right(x)**).

#### V. **color("color")**

Fungsi color adalah untuk mengubah warna garis yang ditarik pada canvas. Ada belasan pilihan warna default seperti "red", "blue", "yellow", "green", "brown", "violet", "purple", "white", "black", "gray". Silahkan tebak sisa-sisa warna yang tersedia :)

**Coba kalian praktekan penggunaan fungsi-fungsi diatas :)**

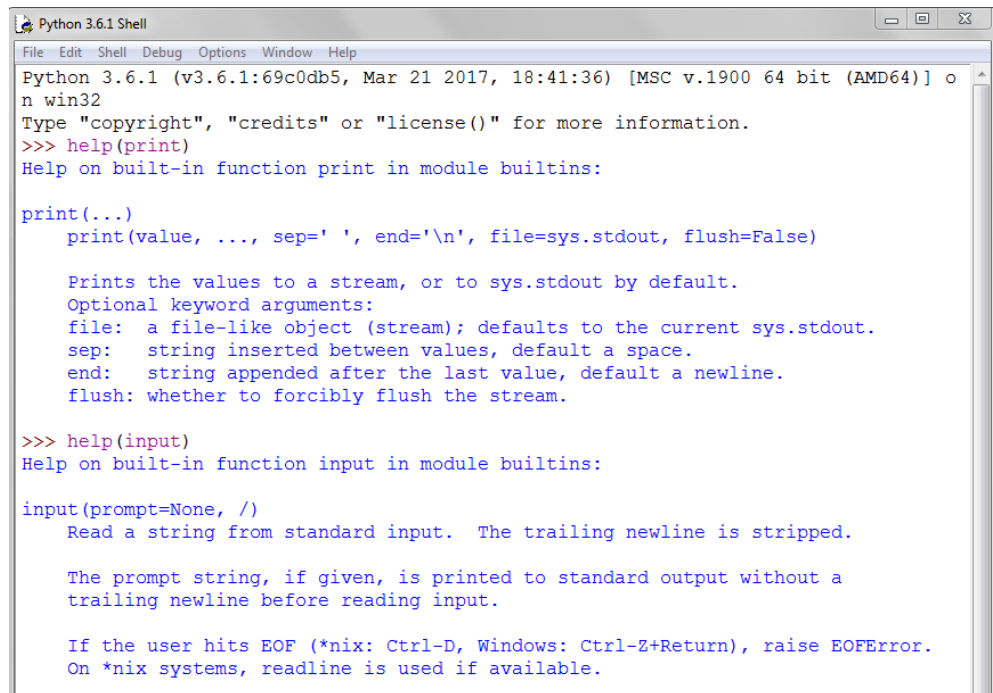
Masih banyak fungsi yang dapat digunakan dalam turtle. Kalian dapat melihat fungsi-fungsi tersebut dengan mengetik `help(turtle)` atau dari dokumentasi Python (Python Docs).

#### c. **Documentation**

Ada kalanya kita kebingungan mengenai kegunaan fungsi X, Y, dsb. Python sendiri sudah menyediakan dokumentasi mengenai segala sesuatu mengenai Python, dokumentasinya bisa kalian akses dengan cara klik **Help -> Python Docs**.

Atau kalian juga bisa menggunakan fungsi **help()** dimana paramaternya merupakan fungsi/objek yang ingin kalian tanyakan.



A screenshot of a Python 3.6.1 Shell window. The window has a menu bar with 'File', 'Edit', 'Shell', 'Debug', 'Options', 'Window', and 'Help'. The title bar says 'Python 3.6.1 Shell'. The main text area shows the following content:

```
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 18:41:36) [MSC v.1900 64 bit (AMD64)] o
n win32
Type "copyright", "credits" or "license()" for more information.
>>> help(print)
Help on built-in function print in module builtins:

print(...)
    print(value, ..., sep=' ', end='\n', file=sys.stdout, flush=False)

    Prints the values to a stream, or to sys.stdout by default.
    Optional keyword arguments:
    file: a file-like object (stream); defaults to the current sys.stdout.
    sep: string inserted between values, default a space.
    end: string appended after the last value, default a newline.
    flush: whether to forcibly flush the stream.

>>> help(input)
Help on built-in function input in module builtins:

input(prompt=None, /)
    Read a string from standard input. The trailing newline is stripped.

    The prompt string, if given, is printed to standard output without a
    trailing newline before reading input.

    If the user hits EOF (*nix: Ctrl-D, Windows: Ctrl-Z+Return), raise EOFError.
    On *nix systems, readline is used if available.
```

Atau kalian bisa searching di *google/stack-overflow like anybody else ever* :)

**Setelah kalian telah mempelajari jutsu pertama kalian, mari kita mulai beberapa perTARUNgan agar kalian semakin mahir di Dunia Python!**

## Soal Tutorial

### Pertarungan #1 Turtle

Benny dan Ibnu merupakan mahasiswa baru Fasilkom UI, setelah mempelajari cara penggunaan IDLE, cara membuat program sederhana, dan cara menggambar dengan Turtle, kini Benny dan Ibnu dapat menulis program untuk membuat bentuk segitiga dengan turtle.

**Contoh program:**

```
import turtle

# Instansiasi objek turtle
t = turtle.Turtle()

# Mengaktifkan pena
t.pendown()

# Memakai tinta biru
t.color('blue')

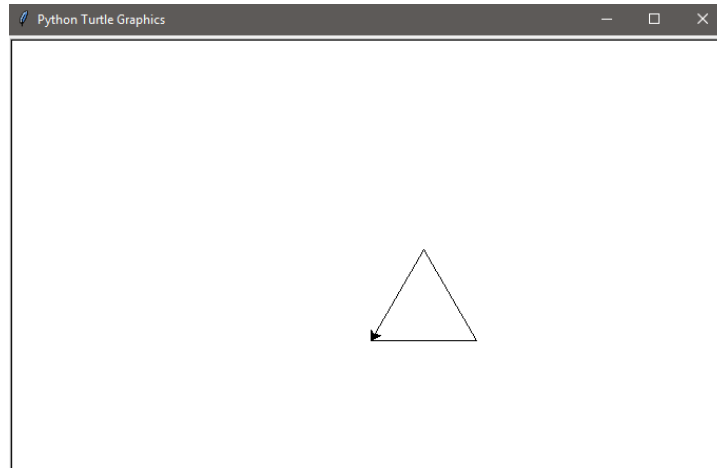
# Bergerak maju sejauh 100 satuan
t.forward(100)

# Memutar ke arah kiri sebesar 120 derajat
t.left(120)

t.forward(100)
t.left(120)
t.forward(100)

# Menon-aktifkan pena
t.penup()

# Menutup turtle setelah di-click
turtle.exitonclick()
```



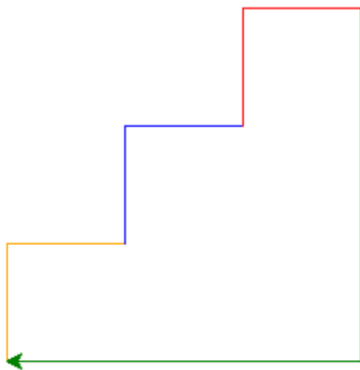
Benny, sebagai sesama mahasiswa baru, tidak ingin kalah dengan Ibnu. Ia pun berniat membuat bentuk yang lebih rumit dari Ibnu, yaitu bentuk **anak tangga**. Sebagai mahasiswa Fasilkom UI yang baik hati dan tidak sombong, bantulah keinginan Benny!

- Buatlah program seperti contoh program di atas, tetapi sekarang dengan bentuk **tangga dengan tiga anak tangga yang tertutup!**
- Coba mintalah **input** dari user untuk digunakan sebagai **panjang** dari sisi tangga tersebut.
- Aturan pewarnaan tangga disesuaikan dengan contoh output yang dilampirkan.

Contoh Masukan:

```
>>> Masukkan panjang setiap anak tangga: 75
```

Contoh Keluaran:



Format penamaan file: **lab01\_[kelas]\_[kode asdos]\_[nama kalian]\_[npm]\_pertarungan1.py**

## Pertarungan #2

### Tipe Data

Benny merupakan seorang anak yang rajin, selalu ingin tahu dan senang membantu. Suatu ketika, Kakak dari Benny yaitu Hamish, akan melangsungkan pernikahan. Hamish memiliki pekerjaan sebagai aktor yang dibayar sebesar Rp.500.000 per harinya. Karena biaya resepsi pernikahan mahal, tentunya akan membutuhkan waktu yang lama untuk dapat memenuhinya. Mengetahui Anda sedang belajar Dasar-Dasar Pemrograman, Benny memiliki ide untuk membantu kakaknya dengan meminta Anda membuat sebuah program untuk memprediksi lama waktu yang dibutuhkan untuk dapat memenuhi biaya pernikahan tersebut.

**Asumsikan:**

**1 tahun = 12 bulan**

**1 bulan = 4 minggu**

**1 minggu = 7 hari**

**Format Masukan**

Masukan berupa biaya pernikahan. Biaya pernikahan **dijamin habis dibagi Rp. 500.000 dan merupakan bilangan non-negatif.**

**Format Keluaran**

Keluaran berupa hasil prediksi waktu untuk dapat memenuhi biaya pernikahan dan **harus dalam format:** Anda harus bekerja selama  $x$  tahun,  $y$  bulan,  $z$  minggu,  $a$  hari untuk memenuhi biaya pernikahan.

dengan  $x, y, z, a \in \mathbb{Z}^+$  (elemen bilangan bulat non-negatif) yang merupakan hasil prediksi waktu tahun, bulan, minggu, dan hari secara urut serta:

- $x \geq 0$ ,
- $0 \leq y < 12$ ,
- $0 \leq z < 4$ ,
- $0 \leq a < 7$

**Contoh Masukan 1** (Masukan adalah bagian yang di-*bold*)

Masukkan biaya pernikahan: Rp **964000000**

**Hint:** `print()` akan selalu membuat cursor ke newline (baris baru), supaya cursor tidak ke baris baru, gunakan: `print(sesuatu, end="")`

**Contoh Keluaran 1**

Anda harus bekerja selama 5 tahun 8 bulan 3 minggu 3 hari untuk memenuhi biaya pernikahan.

#### Contoh Masukan 2

Masukkan biaya pernikahan: Rp 168000000

#### Contoh Keluaran 2

Anda harus bekerja selama 0 tahun 12 bulan 0 minggu 0 hari untuk memenuhi biaya pernikahan. (SALAH)

Anda harus bekerja selama 1 tahun 0 bulan 0 minggu 0 hari untuk memenuhi biaya pernikahan. (BENAR)

Format penamaan file: lab01\_[kelas]\_[kode asdos]\_[nama kalian]\_[npm]\_pertarungan2.py

Kedua file digabungkan dalam satu file .zip dengan format penamaan file:

lab01\_[kelas]\_[kode asdos]\_[nama kalian]\_[npm].zip

lalu dikumpulkan ke form submisi yang ada di SCell.

SELAMAT MENERJAKAN DAN HAPPY CODING 🐼