

# TUTOR ML #1

## Install Python

Pada Linux dan Mac, silakan panggil perintah berikut di konsol atau command prompt:

```
1. python --version
```

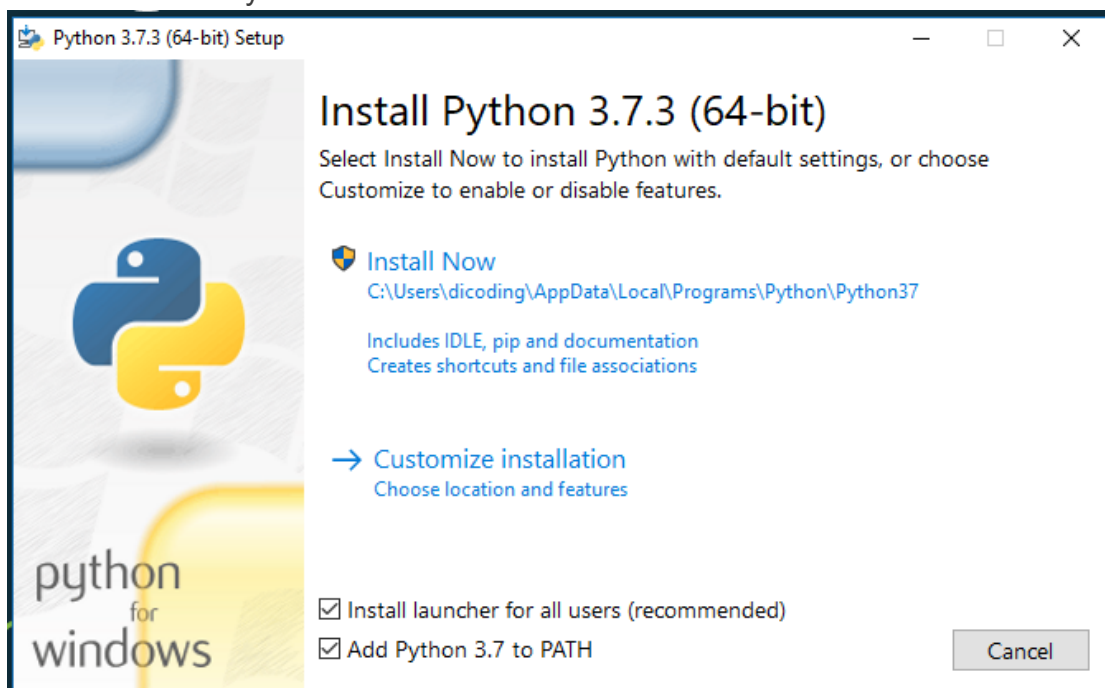
atau

```
1. python3 --version
```

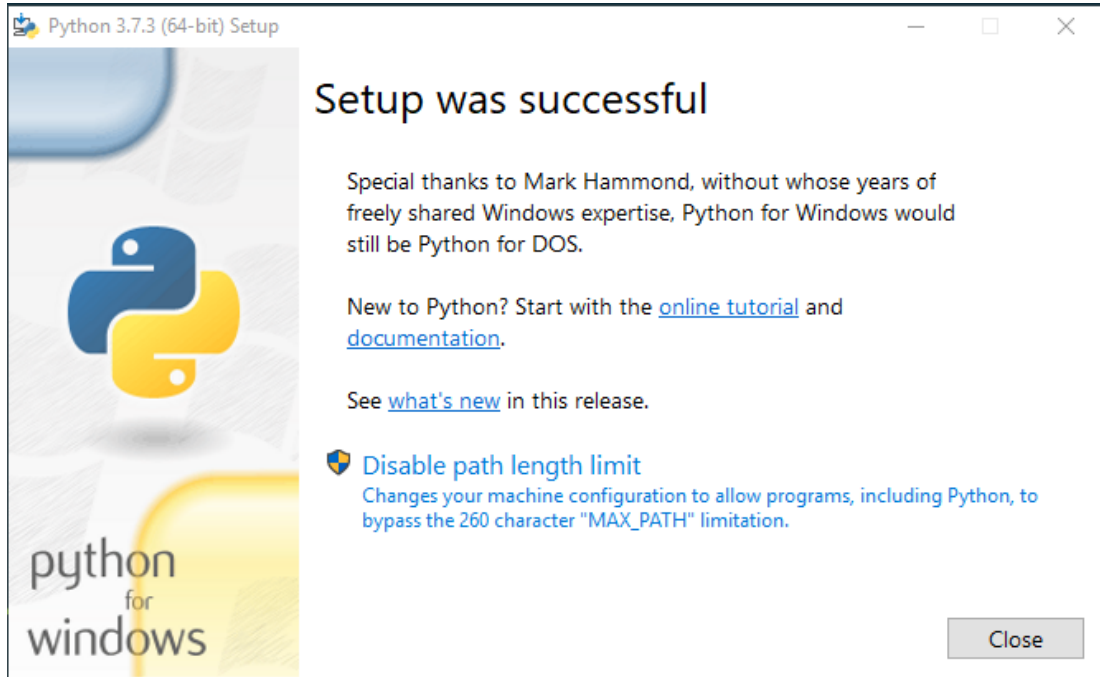
Python versi terbaru dapat diakses di <https://www.python.org/downloads/>

Berikut langkah-langkah meng-update atau menginstal Python pada sistem operasi Windows:

1. Unduh dari <https://www.python.org/downloads/windows/>, pilih Windows X86-64 (64 Bit) atau Windows X86 (32 Bit). Pemilihan 32 dan 64 bit ini akan juga mempengaruhi jumlah memori yang dapat digunakan. Umumnya, komputer saat ini mendukung 64 bit.
2. Untuk Anda yang memiliki koneksi internet relatif baik, Anda dapat menggunakan web installer yang relatif kecil, namun akan melakukan pengunduhan di belakang layar. Jika Anda menggunakan komputer lain untuk mengunduh installer, silakan unduh executable installer (+- 25 Mb).
3. Lakukan instalasi, pastikan mencentang Add Python 3.7 to PATH untuk menambahkan Python dalam Environment Variables.



| Catatan: sebaiknya klik disable path length limit untuk mengatasi problem path yang melebihi 255 karakter (nama folder yang terlalu panjang).



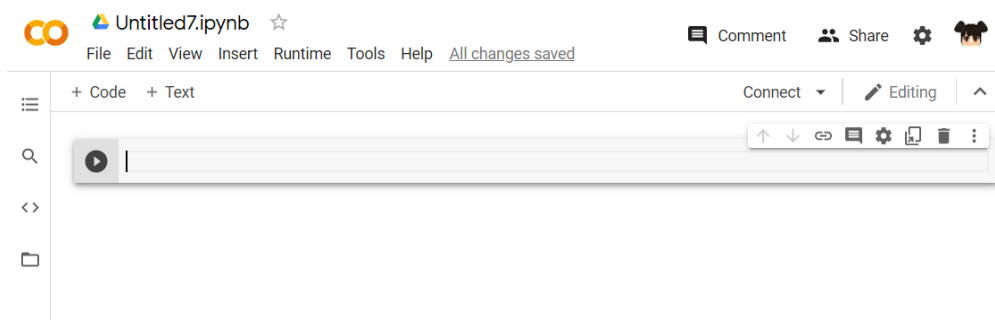
## Link Penting

Apabila Anda mengalami kesulitan, silakan merujuk ke halaman <https://docs.python.org/id/3.8/installing/index.html> atau gunakan mesin pencarian untuk mencari solusi apabila Anda mengalami kendala dalam instalasi.

## ONLINE COMPILER

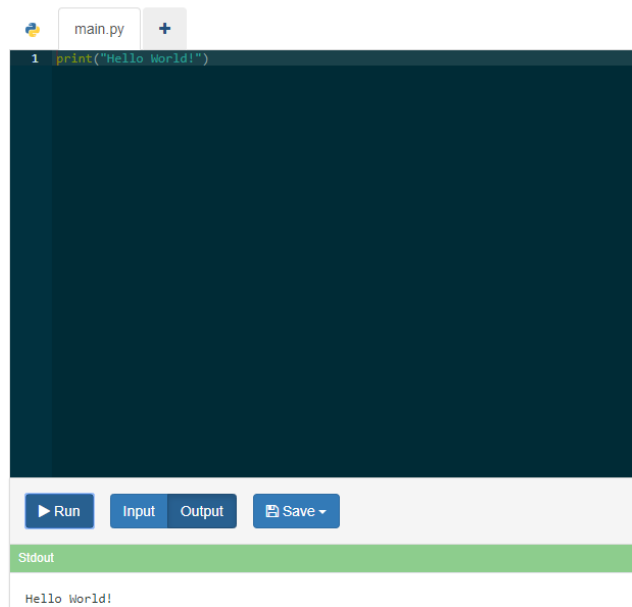
Google Colab

Silahkan mengakses google colab pada link <https://colab.research.google.com/>



Glottio

## Untitled



## Repl.it



## Apa saja yang dapat Anda lakukan pada python interactive ini?

- **InteractiveExample: Calculator**

```
yoza@beaver:~$ python
Python 3.7.3 (default, Mar 26 2019, 00:55:50)
[GCC 7.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 3+3
6
>>> 3*3
9
>>> 3**3
27
>>>
```

Anda dapat menggunakan python pada sesi interaktif ini menjadi kalkulator untuk perhitungan matematika.

- **InteractiveExample: Print**

```
yoza@beaver:~$ python
Python 3.7.3 (default, Mar 26 2019, 00:55:50)
[GCC 7.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello, World!")
Hello, World!
>>>
```

- **InteractiveExample: import library dan mendapatkan Current working directory**

```
yoza@beaver:~$ python
Python 3.7.3 (default, Mar 26 2019, 00:55:50)
[GCC 7.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import os
>>> os.getcwd()
'/home/yoza'
>>>
```

## TIPE DATA

### Numbers

Tipe numerik pada Python dibagi menjadi 3:

- Int
- Float
- Complex

Cobalah bermain-main dengan contoh berikut:

1. `a = 10`
2. `print(a, "bertipe", type(a))`
3. `b = 1.7`
4. `print(a, "bertipe", type(b))`
5. `c = 1+3j`
6. `print(c, " Bertipe bilangan kompleks? ", isinstance(1+3j,complex))`

Output seharusnya:

```
10 bertipe <class 'int'>
1.7 bertipe <class 'float'>
(1+2j) Bertipe bilangan kompleks? True
```

Integer tidak dibatasi oleh angka atau panjang tertentu, namun dibatasi oleh memori yang tersedia. Sehingga Anda tidak perlu menggunakan variabel yang menampung big number misalnya long long (C/C++), biginteger, atau sejenisnya. Contoh kode untuk menunjukkan bahwa Python tidak membatasi output integer adalah pencarian bilangan ke-10.000 pada deret fibonacci (catatan: bilangan ke-10.000 pada deret fibonacci memiliki panjang 2.090 digit) sebagai berikut:

```
1. x=[0]*10005;           #inisialisasi array 0 sebanyak 10005; x[0]=0
2. x[1]=1;                #x[1]=1
3.
4. for j in range(2,10001):
5.     x[j]=x[j-1]+x[j-2]  # Fibonacci
6. print(x[10000])
```

Output:

```
1. 3364476487643178326662161200510754331030214846068006390656476997468008144216666236
8155595513633734025582065332680836159373734790483865268263040892463056431887354544
3695598274916066020998841839338646527313000888302692356736131351175792974378544137
5213052050434770160226475831890652789085515436615958298727968298751063120057542878
3453215515103870818298969791613127856265033195487140214287532698187962046936097879
9003509623022910263681314931952756302278376284415403605844025721143349611800230912
0828704608892396232883546150577658327125254609359112820392528539343462090424524892
9403901706233888991085841065183173360437470737908552631764325733993712871937587746
8974799263058370657428301616374089691784263786242128352581128205163702980893320999
0570792006436742620238978311147005407499845925036063356093388383192338678305613643
5351892133279732908133732642652633989763922723407882928177953580570993691049175470
8089318410561463223382174656373212482263830921032977016480547262438423748624114530
9381220656491403275108664339451751216152654536133311131404243685480510676584349352
3836959653428071768775328348234345557366719731392746273629108210679280784718035329
131176778924659089938635459327894523776744061922403376386740040213303432974969020
2832814593341882681768389307200363479562311710310129195316979460763273758925353077
2552375943788434504067715555779056450443016640119462580972216729758615026968443146
9520346149322911059706762432685159928347098912847067408620085871350162603120719031
7208609408129832158107728207635318662461127824553720853236530577595643007251774431
5051539600905168603220349163222640885248852433158051534849622434848299380905070483
4824493274537326245677558790891871908036620580095947431500524025327097469953187707
2437682590741993963226598414749819360928522394503970716544315642132815768890805878
3183404917434556270520223564846495196112460268313970975069382648706613264507665074
6115126775227486215986425307112984411826226610571635150692600298617049454250474913
7811515413994155067125627119713325276363193960690289565028826860836224108205056243
0701794976171121233066073310059947366875
```

### Batasan akurasi variabel bertipe float

Python melakukan pemotongan pada digit ke 16 pada variabel float. Float atau bilangan pecahan dibatasi akurasinya pada 15 desimal. Yang membedakan Integer dan Float adalah titik (decimal points). Misalnya dalam penulisan angka 1 jenisnya Integer, tapi jika dituliskan sebagai 1.0 artinya berjenis Float atau pecahan.

```
1. b = 0.1234567890123456789
2. print(b)
```

Output:

```
0.12345678901234568
```

Contoh jika berupa integer:

```
1. a = 1234567890123456789
2. print(a)
```

Output:

```
1234567890123456789
```

Karena Python banyak digunakan juga oleh matematikawan, tipe bilangan di Python juga mendukung bilangan imajiner dan bilangan kompleks. Nilai bilangan kompleks (complex) dituliskan dalam formulasi  $x + yj$ , yakni bagian  $x$  adalah bilangan real dan  $y$  adalah bilangan imajiner. Contohnya adalah sebagai berikut:

```
1. c = 1+5j
2. print(c)
```

Output:

```
(1+5j)
```

## Strings

String adalah urutan dari karakter unicode yang dideklarasikan dengan petik tunggal atau ganda. String >1baris dapat ditandai dengan tiga petik tunggal atau ganda `'''` atau `"""`.

```
1. s = "Ini adalah string baris tunggal"
1. s = '''Ini adalah string
2. yang memiliki baris pertama
3. dan selanjutnya baris kedua'''
```

## Bool/Boolean

Tipe data bool atau Boolean merupakan turunan dari bilangan bulat (integer atau int) yang hanya punya dua nilai konstanta: True dan False.

### *Nilai Boolean*

Nilai konstanta False dan True merepresentasikan nilai kebenaran (truth values), meskipun ada nilai-nilai lain yang juga dianggap benar atau salah. Di dalam konteks angka, misalnya digunakan sebagai argumen dari operator matematika aritmatika, kedua nilai ini berlaku seperti halnya bilangan bulat 0 dan 1, sesuai False dan True.

Ada fungsi bawaan bool() yang dapat mengubah nilai menjadi nilai Boolean, apabila nilai tersebut dapat direpresentasikan sebagai nilai kebenaran (truth values).

Nilai kebenaran adalah sebuah nilai yang dapat diuji sebagai benar atau salah, untuk digunakan di sintaksis kondisi if atau while atau sebagai operan dari operasi Boolean.

Berikut adalah objek bawaan yang didefinisikan bernilai salah dalam pengujian nilai kebenaran:

- Konstanta yang sudah didefinisikan bernilai salah: None dan False.
- Angka nol dari semua tipe numeric: 0, 0.0, 0j, Decimal(0), Fraction(0, 1).
- Urutan (sequence) dan koleksi (collection) yang kosong: "", {}, set(), range(0).

Untuk objek yang didefinisikan sendiri, representasi nilai Boolean akan bergantung dari definisi metode (method) khusus bernama \_\_bool\_\_(self). Jika metode ini mengembalikan True maka interpretasi nilai dari objeknya akan True, demikian juga sebaliknya.

### *Operasi Boolean*

Operasi dan fungsi bawaan yang memiliki hasil Boolean akan selalu mengembalikan 0 atau False untuk yang bernilai salah, serta 1 atau True untuk yang bernilai benar, kecuali dinyatakan berbeda dalam dokumentasi.

Operasi untuk tipe Boolean akan dijelaskan lebih lanjut di modul Operator, Operands, dan Expressions.

## List

List adalah jenis kumpulan data terurut (ordered sequence), dan merupakan salah satu variabel yang sering digunakan pada Python. Serupa, namun tak sama dengan array pada bahasa pemrograman lainnya. Bedanya, elemen List pada Python tidak harus memiliki tipe data yang sama. Mendeklarasikan List cukup mudah dengan kurung siku dan elemen yang dipisahkan dengan koma.

Setiap data di dalamnya dapat diakses dengan indeks yang dimulai dari 0.

```
1. a = [1, 2.2, 'python']
```

Python mengenal slicing operator `[]` yang dapat melakukan ekstraksi sebuah item atau beberapa item yang berada dalam range tertentu pada tipe data urutan (sequences), misalnya list, string dan tuple. Beberapa tipe urutan juga mendukung "extended slicing" dengan parameter ketiga berupa "step".

- **x[0]** artinya mengambil elemen paling awal, dengan index 0 dari List x.
- **x[5]** artinya mengambil elemen dengan index 5 dari List x.
- **x[-1]** artinya mengambil elemen dengan index paling belakang ke-1 dari List x.
- **x[3:5]** artinya membuat list dari anggota elemen List x dengan index 3 hingga sebelum index 5 (tidak termasuk elemen dengan index 5, dalam hal ini hanya index 3-4).
- **x[:5]** artinya membuat list dari anggota elemen List x paling awal hingga sebelum index 5 (tidak termasuk elemen dengan index 5, dalam hal ini hanya index 0-4).
- **x[-3:]** artinya membuat list dari anggota elemen List x mulai index ke-3 dari belakang hingga paling belakang.
- **x[1:7:2]** artinya membuat list dari anggota elemen List x dengan index 1 hingga sebelum index 7, dengan "step" 2 (dalam hal ini hanya index 1, 3, 5).

```
1. x = [5,10,15,20,25,30,35,40]
2. print(x[5])
3. print(x[-1])
4. print(x[3:5])
5. print(x[:5])
6. print(x[-3:])
7. print(x[1:7:2])
```

Output:

```
30
40
[20, 25]
[5, 10, 15, 20, 25]
[30, 35, 40]
[10, 20, 30]
```

Elemen pada list dapat diubah atau ditambahkan. Misalnya untuk melakukan perubahan kemudian penambahan:

```
1. x = [1,2,3]
2. x[2]=4
3. print (x)
```

Output:

```
[1, 2, 4]
```



```
1. x = [1,2,3]
2. x[2]=4
3. x.append(5)
4. print(x)
```

Output:

```
[1, 2, 4, 5]
```

**Untuk menghapus item pada list, gunakan fungsi del.** Ingat bahwa Indeks Python dimulai dari 0:

```
1. binatang = ['kucing', 'rusa', 'badak', 'gajah']
2. del binatang[2]
3. print(binatang)
```

Output:

```
['kucing', 'rusa', 'gajah']
```

Coba tambahkan kembali:

```
1. del bintang [2]
2. print(binatang)
```

Pada baris terbawah kode di atas, maka output akan menjadi:

Output:

```
['kucing', 'rusa']
```

## Tuple

Tuple adalah jenis dari list yang tidak dapat diubah elemennya. Umumnya tuple digunakan untuk data yang bersifat sekali tulis, dan dapat dieksekusi lebih cepat. Tuple didefinisikan dengan kurung dan elemen yang dipisahkan dengan koma.

```
1. t = (5, 'program', 1+3j)
```

Seperti list, kita dapat melakukan slicing, namun pada tuple kita tidak dapat melakukan perubahan:

```
1. t = (5, 'program', 1+3j)
2. print(t[1])
3. print(t[0:3])
4. print(t[0]=10)
```

Output:

```
'program'  
(5, 'program', (1+3j))  
Traceback (most recent call last):  
File "<stdin>", line 1, in <module>  
TypeError: 'tuple' object does not support item assignment
```

Referensi yang dapat dipelajari:

- <https://automatetheboringstuff.com/2e/chapter6/>
- [https://docs.python.org/3/reference/lexical\\_analysis.html#literals](https://docs.python.org/3/reference/lexical_analysis.html#literals)
- <https://python-reference.readthedocs.io/en/latest/docs/str/>
- <https://docs.python.org/id/3.8/library/string.html#format-specification-mini-language>