

# **LAPORAN PRAKTIKUM PEMROGRAMAN GUI**

## **MODUL I VARIABLE, TIPE DATA, DAN OPERATOR**



**Disusun Oleh :**  
Satria Adi Nugraha  
NIM : 19104027

**Dosen**  
Ariq Cahya Wardhana, S.Kom., M.Kom.

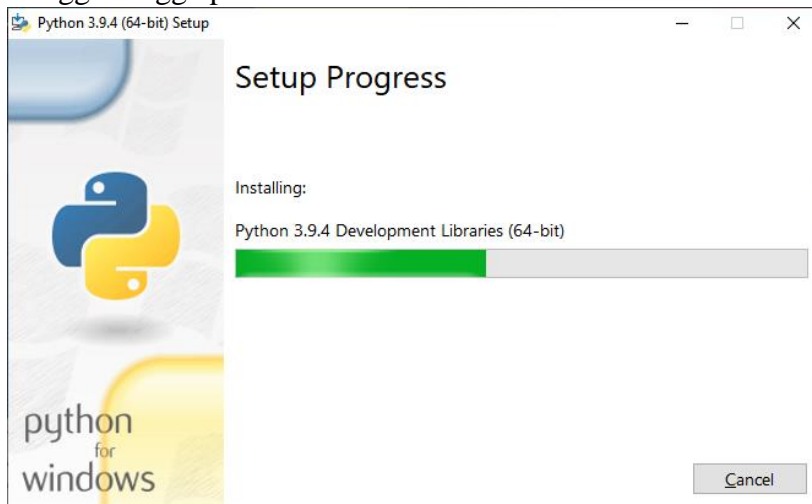
**PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK  
FAKULTAS INFORMATIKA  
INSTITUT TEKNOLOGI TELKOM  
PURWOKERTO  
2020**

# INSTALASI PYTHON

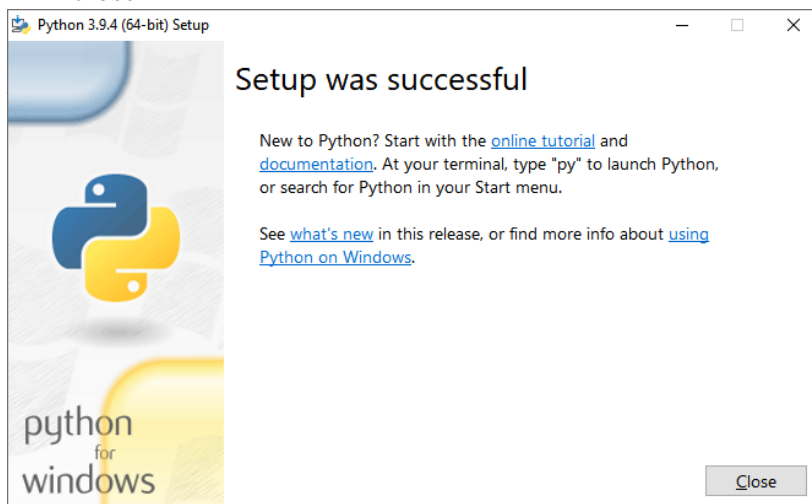
## 1. Klik next



## 2. Tunggu hingga proses selesai

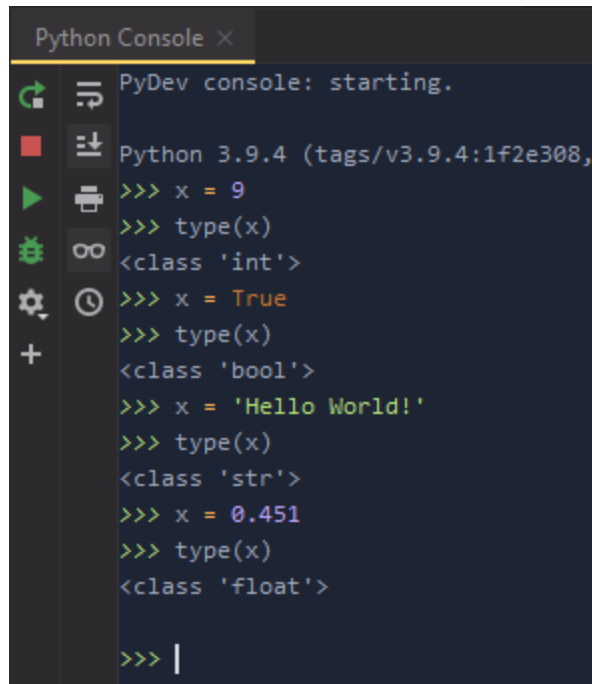


## 3. Klik close



## PYTHON BASICS

### VARIABLES AND DATA TYPES

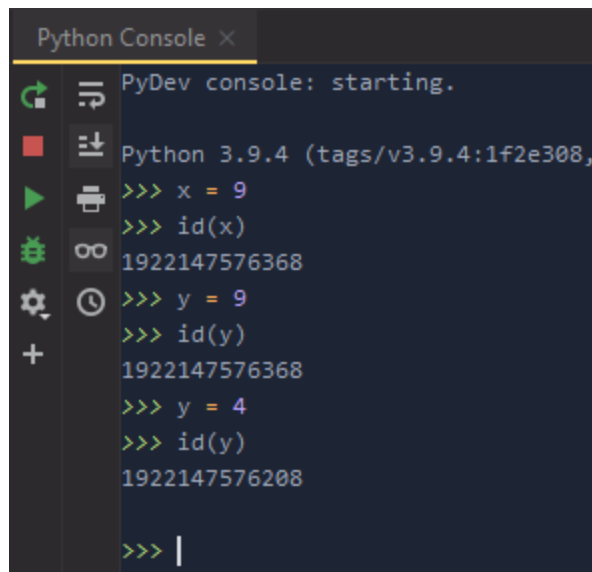


```
Python Console x
PyDev console: starting.
Python 3.9.4 (tags/v3.9.4:1f2e308,
>>> x = 9
>>> type(x)
<class 'int'>
>>> x = True
>>> type(x)
<class 'bool'>
>>> x = 'Hello World!'
>>> type(x)
<class 'str'>
>>> x = 0.451
>>> type(x)
<class 'float'>
>>> |
```

Empat tipe data utama yang ada di Python adalah:

- Integer (bilangan bulat)
- Float (bilangan desimal)
- String (karakter)
- Boolean (benar/salah)

### VARIABLE IDS

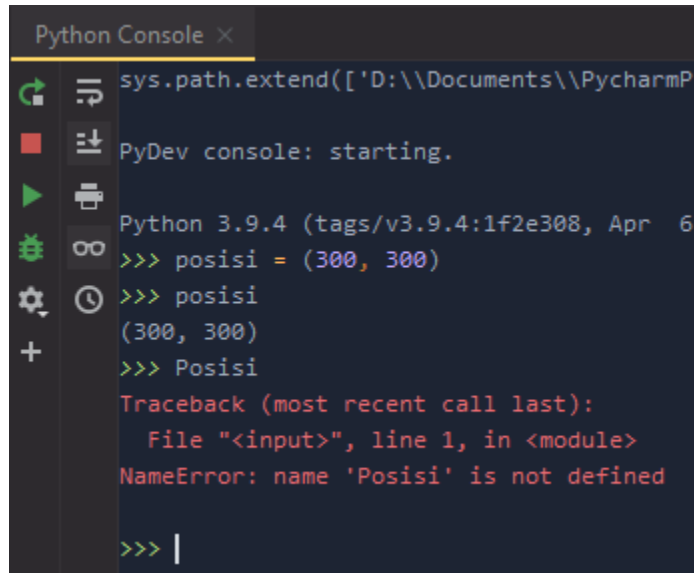


```
Python Console x
PyDev console: starting.
Python 3.9.4 (tags/v3.9.4:1f2e308,
>>> x = 9
>>> id(x)
1922147576368
>>> y = 9
>>> id(y)
1922147576368
>>> y = 4
>>> id(y)
1922147576208
>>> |
```

Python menyimpan nilai-nilai variable di memory di PC kita. Block memory yang digunakan Python tentunya memiliki ID. Kita dapat mengecek ID yang digunakan

Python untuk menyetorkan nilai sebuah variable dengan statement di atas. Jika diperhatikan, berdasarkan gambar di atas, dua variable berbeda yang memiliki nilai yang sama memiliki ID yang sama pula. Variable yang sama jika nilainya berbeda, maka IDnya juga berbeda.

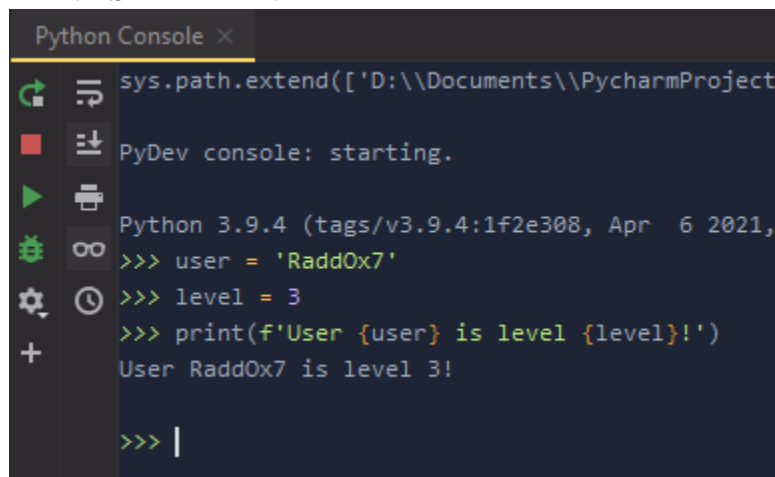
## PYTHON IS CASE SENSITIVE



```
Python Console X
sys.path.extend(['D:\\Documents\\PycharmPr
PyDev console: starting.
Python 3.9.4 (tags/v3.9.4:1f2e308, Apr 6
>>> posisi = (300, 300)
>>> posisi
(300, 300)
>>> Posisi
Traceback (most recent call last):
  File "<input>", line 1, in <module>
NameError: name 'Posisi' is not defined
>>> |
```

Di Python, penggunaan huruf kecil dan huruf kapital **SANGAT** berpengaruh, karena keduanya dianggap karakter yang berbeda.

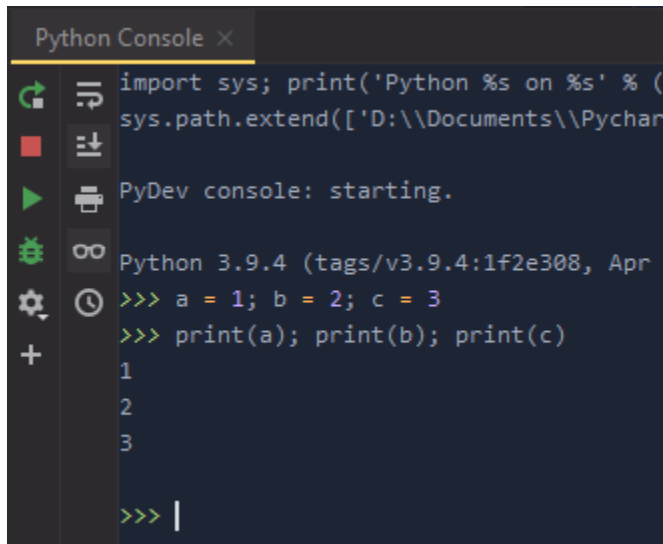
## PRINT STATEMENT



```
Python Console X
sys.path.extend(['D:\\Documents\\PycharmProject
PyDev console: starting.
Python 3.9.4 (tags/v3.9.4:1f2e308, Apr 6 2021,
>>> user = 'RaddOx7'
>>> level = 3
>>> print(f'User {user} is level {level}!')
User RaddOx7 is level 3!
>>> |
```

Statement print di Python bisa menggunakan single quote (') maupun double quote ("). Jika anda ingin mengoutputkan variable atau statement di antara string, anda bisa menuliskan huruf 'f' sebelum tanda quote, lalu menggunakan bracket {} dan meletakan variable atau statement yang ingin anda print di dalamnya.

## SEMICOLON



```
Python Console X
import sys; print('Python %s on %s' % (
sys.path.extend(['D:\\Documents\\Pychar

PyDev console: starting.

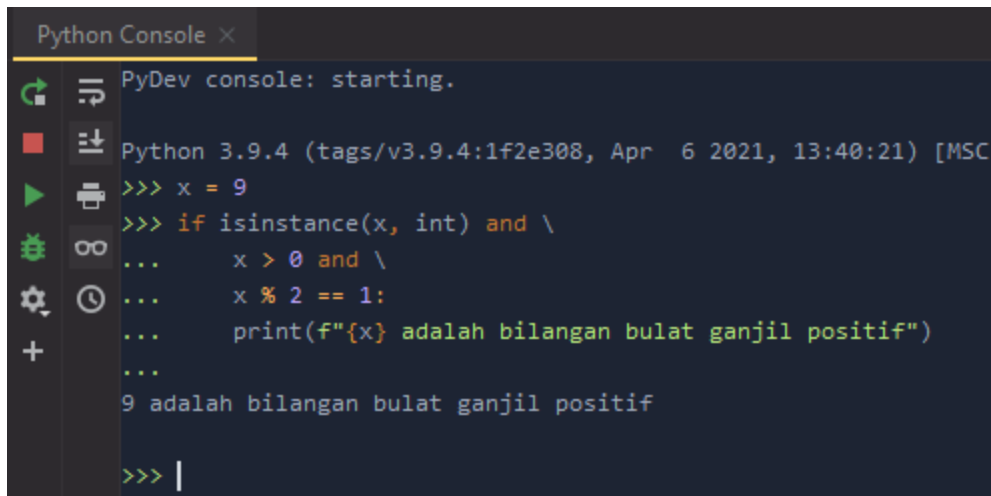
Python 3.9.4 (tags/v3.9.4:1f2e308, Apr

>>> a = 1; b = 2; c = 3
>>> print(a); print(b); print(c)
1
2
3

>>> |
```

Pada Python setiap kode program yang dituliskan tidak harus diakhiri dengan sebuah statement (biasanya tanda titik koma / semicolon) seperti pada Java dan C. Titik koma pada python hanya diberikan jika ingin menulis dua atau lebih statement pada satu baris yang sama.

## BACKSLASH



```
Python Console X
PyDev console: starting.

Python 3.9.4 (tags/v3.9.4:1f2e308, Apr 6 2021, 13:40:21) [MSC

>>> x = 9
>>> if isinstance(x, int) and \
...     x > 0 and \
...     x % 2 == 1:
...     print(f"{x} adalah bilangan bulat ganjil positif")
...
9 adalah bilangan bulat ganjil positif

>>> |
```

Kita dapat menggunakan backslash (\) untuk menuliskan satu statement yang sama pada baris berbeda. Biasanya anda hanya ingin melakukan ini jika anda memiliki statement yang sangat panjang dan demi menjaga kerapihan kode anda memutuskan untuk meneruskan statement di baris selanjutnya.

```
Python Console X
PyDev console: starting.
Python 3.9.4 (tags/v3.9.4:1f2e308, Apr 6 2021, 13:40:21)
>>> print("Pemrograman GUI " +
...       " dengan Python dan PyQt")
Pemrograman GUI  dengan Python dan PyQt
>>> data = [
...     100,
...     200,
...     300
... ]
>>> kamus = {
...     'one': 'satu',
...     'two': 'dua',
...     'three': 'tiga'
... }
>>> data
[100, 200, 300]
>>> kamus
{'one': 'satu', 'two': 'dua', 'three': 'tiga'}
>>> |
```

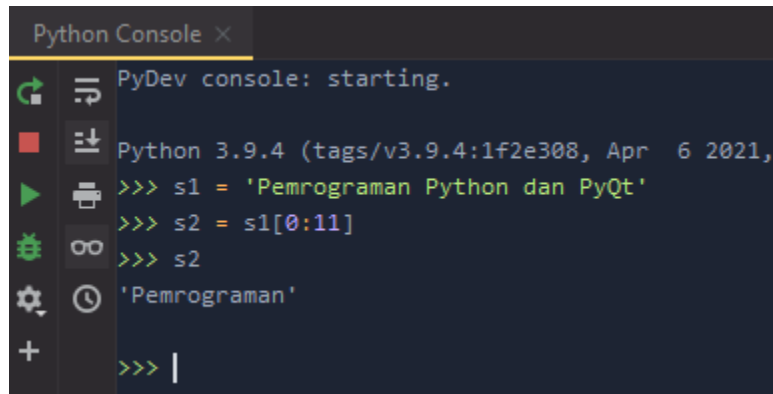
Tetapi tanda backslash tidak diperlukan jika kita menulis perintah kode dalam bentuk array atau kode yang terdapat diantara tanda (...), [...] atau {...}.

## STRING COMPARISON

```
Python Console X
PyDev console: starting.
Python 3.9.4 (tags/v3.9.4:1f2e308,
>>> s1 = 'python'
>>> s2 = 'PYTHON'
>>> s1 == s2
False
>>> s1 != s2
True
>>> s1 < s2
False
>>> |
```

Untuk membandingkan kesamaan string Python menggunakan operator ==. Sedangkan untuk membandingkan id objek string menggunakan is. Selain kedua operator tersebut, Python juga dapat menggunakan operator lainnya untuk membandingkan tipe data string.

## SUB-STRING

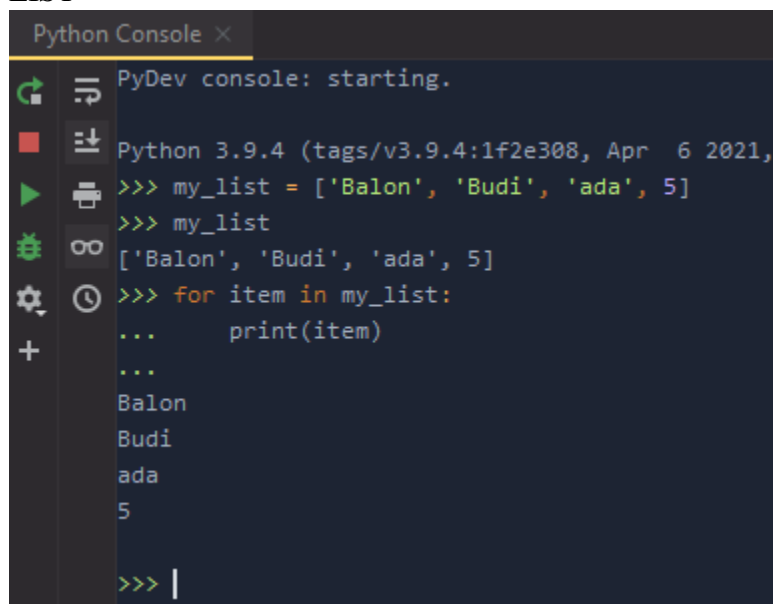
A screenshot of a Python console window titled "Python Console x". The console shows the following code and output:

```
PyDev console: starting.  
Python 3.9.4 (tags/v3.9.4:1f2e308, Apr 6 2021,  
>>> s1 = 'Pemrograman Python dan PyQt'  
>>> s2 = s1[0:11]  
>>> s2  
'Pemrograman'  
>>> |
```

Substring di dalam string dapat diekstrak dengan menggunakan operator slice (:) dengan menyertakan indeks awal dan akhir sebagai penanda. Kode tersebut mengambil substring dari variable s mulai dari indeks ke 0 sampai indeks ke 11.

## COLLECTIONS

### LIST

A screenshot of a Python console window titled "Python Console x". The console shows the following code and output:

```
PyDev console: starting.  
Python 3.9.4 (tags/v3.9.4:1f2e308, Apr 6 2021,  
>>> my_list = ['Balon', 'Budi', 'ada', 5]  
>>> my_list  
['Balon', 'Budi', 'ada', 5]  
>>> for item in my_list:  
...     print(item)  
...  
Balon  
Budi  
ada  
5  
>>> |
```

List adalah salah satu tipe data collection di Python. Sekilas, list ini mirip dengan array. Untuk melihat apa saja item-item yang ada di dalam list, anda bisa menggunakan statement for.