**UNIT 6**

**Pengantar Bahasa Pemrograman Python**

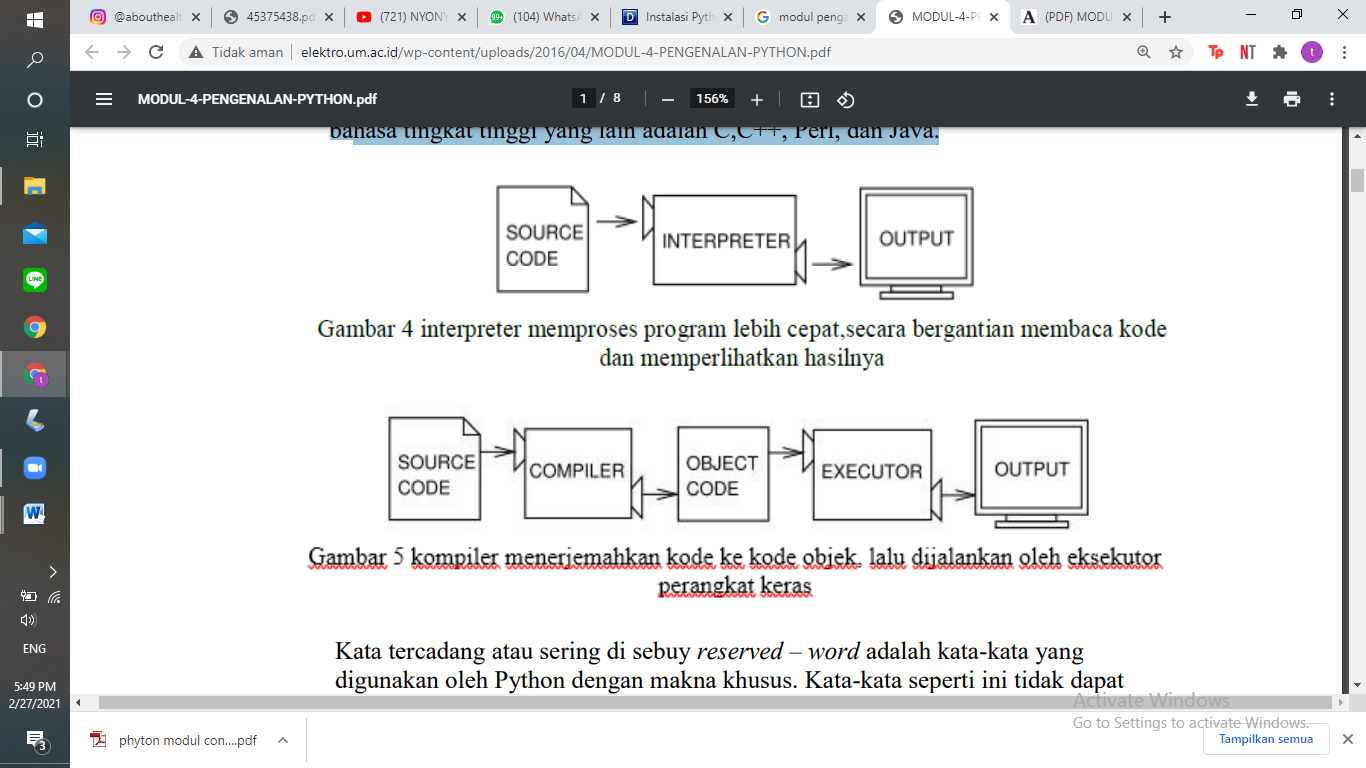
**Tujuan:**

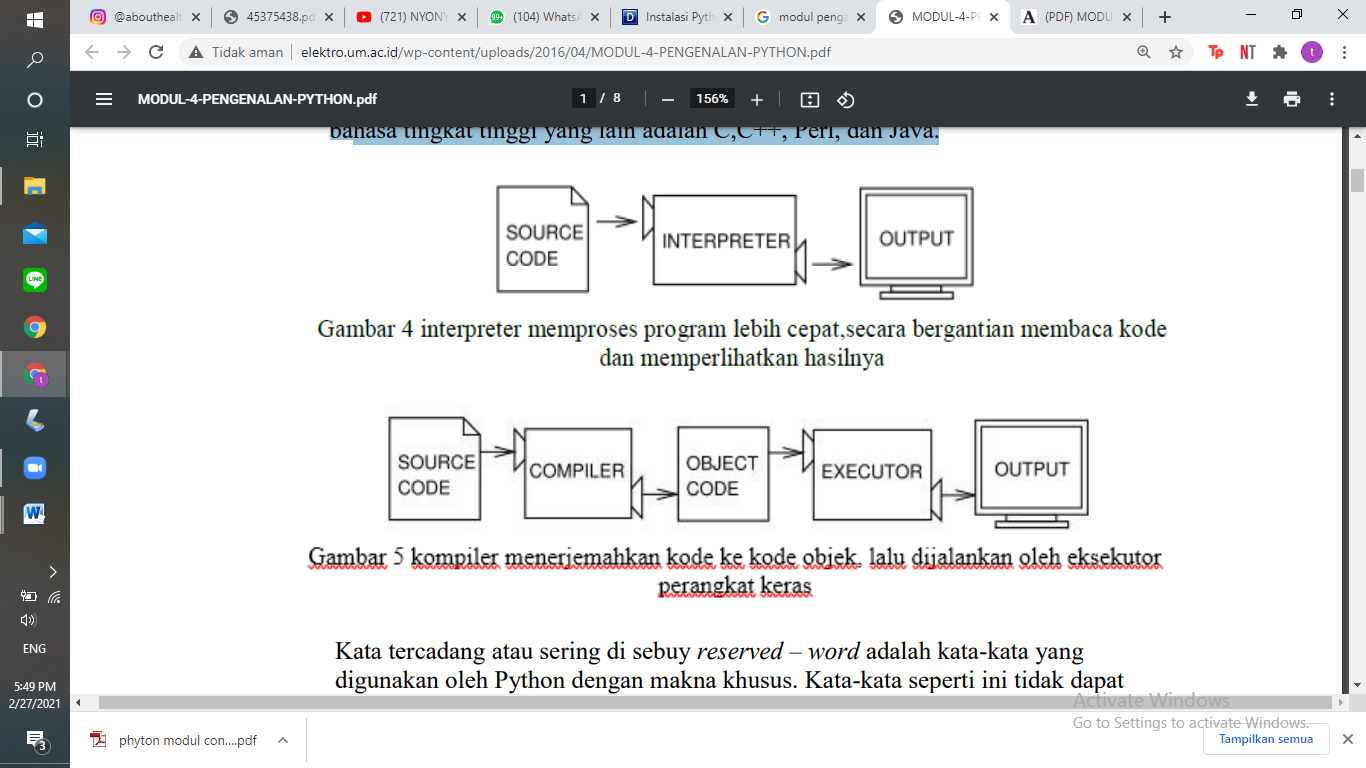
1. Mampu mengerti bahasa pemrograman python dengan baik.
2. Mampu memahami sistematika penulisan bahasa pemograman Pyhton beserta tipe datanya.
3. Mampu memahami bentuk umum algoritma dan implementasinya dalam bahasa python.
4. Mampu membuat program sederhana yang melibatkan input/output, assignment, variable, konstnta, ekspresi, dan tipe data bentukan.

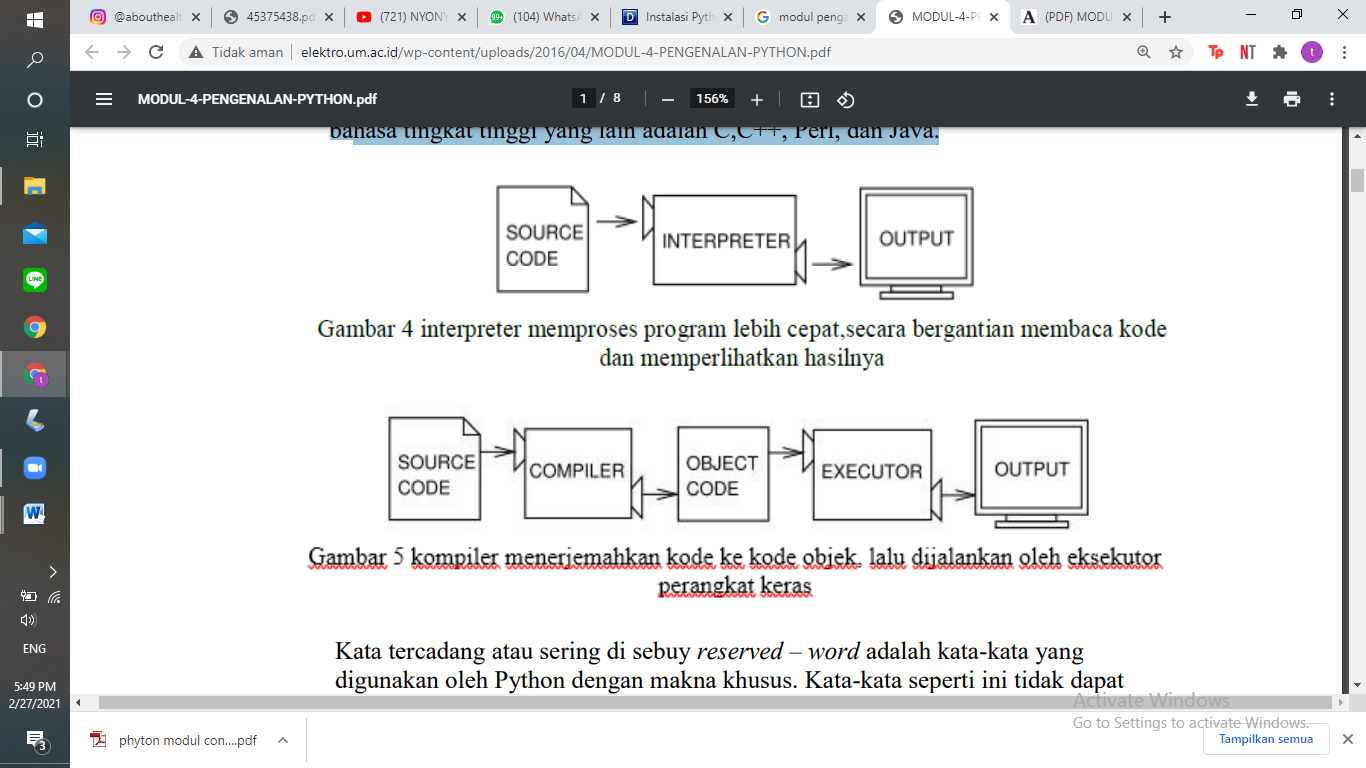
**Dasar Teori:**

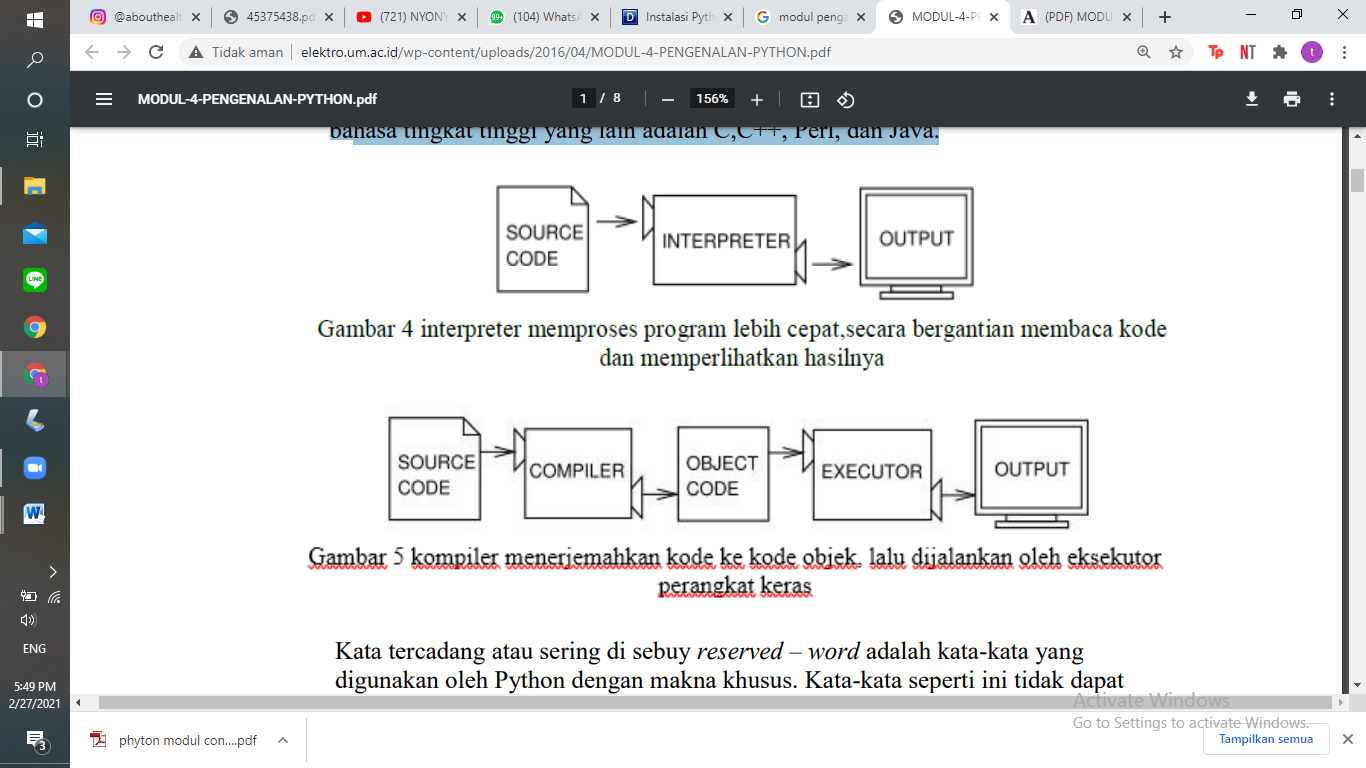
***Bahasa Pemrograman Python***

Python merupakan bahasa pemrograman dinamis yang mendukung pemrograman berbasis objek. File python selalu diakhiri dengan *.py* yang mengindikasikan bahwa file tersebut adalah *python program*.









File python selalu diakhiri dengan *.py* yang mengindikasikan bahwa file tersebut adalah *python program*. Editor akan menjalankan file tersebut melalui *python interpreter*, yang mana akan membaca program dan menentukan arti dari tiap kata pada program tersebut.

**Fitur fitur pada Pyhton**

Beberapa **fitur** yang dimiliki Python adalah :

1. Memiliki kepustakaan yang luas; dalam distribusi Python telah disediakan modul-modul siap pakai untuk berbagai keperluan.
2. Memiliki tata bahasa yang jernih dan mudah dipelajari.
3. Memiliki aturan *layout* kode sumber yang memudahkan pengecekan, pembacaankembali dan penulisan ulang kode sumber. berorientasi obyek.
4. Memiliki sistem pengelolaan memori otomatis (garbage collection, seperti java)modular, mudah dikembangkan dengan menciptakan modul-modul baru; modul-modul tersebut dapat dibangun dengan bahasa Python maupun C/C++.
5. Memiliki fasilitas pengumpulan sampah otomatis, seperti halnya pada bahasapemrograman Java, python memiliki fasilitas pengaturan penggunaan ingatankomputer sehingga para pemrogram tidak perlu melakukan pengaturan ingatankomputer secara langsung.

**Kelebihan Kekurangan Pada Pyhton**

Beberapa kelebihan bahasa Python antara lain :

1. Tidak ada tahapan kompilasi dan penyambungan (link) sehingga kecepatan perubahan pada masa pembuatan system aplikasi meningkat.
2. Tidak ada deklarasi tipe sehingga program menjadi lebih sederhana, singkat, dan fleksible.
3. Manajemen memori otomatis yaitu kumpulan sampah memori sehingga dapat menghindari pencatatan kode
4. Tipe data dan operasi tingkat tinggi yaitu kecepatan pembuatan system aplikasimenggunakan tipe objek yang telah ada
5. Pemrograman berorientasi objek
6. Pelekatan dan perluasan dalam C
7. Pengenalan Python7
8. Terdapat kelas, modul, eksepsi sehingga terdapat dukungan pemrograman skala besarsecara modular
9. Pemuatan dinamis modul C sehingga ekstensi menjadi sederhana dan berkas biner yangkecil
10. Pemuatan kembali secara dinamis modul phyton seperti memodifikasi aplikasi tanpa menghentikannya
11. Model objek universal kelas Satu
12. Konstruksi pada saat aplikasi berjalan
13. Interaktif, dinamis dan alamiah
14. Akses hingga informasi interpreter
15. Portabilitas secara luas seperti pemrograman antar platform tanpa ports
16. Kompilasi untuk portable kode byte sehingga kecepatan eksekusi bertambah dan melindungi kode sumber
17. Antarmuka terpasang untuk pelayanan keluar seperti perangkat Bantu system, GUI,persistence, database, dll

Beberapa kekurangan bahasa Python antara lain :

1. Beberapa penugasan terdapat diluar dari jangkauan python, seperti bahasa pemrograman dinamis lainnya, python tidak secepat atau efisien sebagai statis, tidak seperti bahasa pemrograman kompilasi seperti bahasa C.
2. Disebabkan python merupakan interpreter, python bukan merupakan perangkat bantu terbaik untuk pengantar komponen performa kritis.
3. Python tidak dapat digunakan sebagai dasar bahasa pemrograman implementasi untuk beberapa komponen, tetapi dapat bekerja dengan baik sebagai bagian depan skrip antarmuka untuk mereka.
4. Python memberikan efisiensi dan fleksibilitas tradeoff by dengan tidak memberikannya secara menyeluruh.

**Identitas Python**

Bahasa pemograman Python adalah bahasa pemograman yang mudah dibaca dan terstruktur, hal ini karena digunakannya sistem indentasi. Yaitu memisahkan blok - blok program dengan susunan indentasi. Jadi untuk memasukan sub - sub program dalam suatu blok, sub - sub program tersebut diletakkan satu atau lebih spasi dari kolom suatu blok program.

Python memiliki sedikit perbedaan pada cara penulisan program dengan bahasa pemrograman yang lain seperti C/Java. Kalau pada C/Java menggunakan tanda kurung sebagai pemisah blok program, di Python kita hanya menggunakan spasi sebagai pemisah blok program yang biasa disebut sebagai Indentasi. Karena Python menjalankan perintah secara berurutan, maka kita harus pintar menyusun perintah agar mendapatkan hasil seperti yang diinginkan.

Python mempunyai 28 kata kunci yang mendefinisikan aturan - aturan dan struktur bahasa, dan mereka tidak dapat digunakan sebagai nama variabel.

**Variabel dan Konstanta**

*Variabel*

Sama seperti bahasa c, variabel pada python merupakan tempat untuk menyimpan suatu nilai yang mana nilai tersebut bisa berubah-ubah sejalan dengan program. Pada bahasa python, variabel bisa langsung digunakan tanpa harus dideklarasikan terlebih dahulu.

>>> a = "belajar Python"

>>> b = 5

>>> phi = 3.14

*Konstanta*

Konstanta adalah tipe dari variable yang nilainya tidak dapat diubah. Pada Python, konstanta biasanya di deklarasikan dan ditetapkan pada sebuah modul. Modul adalah file baru yang berisi variabel, fungsi, *dll.*yang dapat diimport kedalam file utama. Didalam modul, konstanta ditulis dengan huruf kapital dan underscore untuk memisahkan kata.

Contoh Mendeklarasikan dan Menetapkan Nilai Pada Konstanta:

|  |  |
| --- | --- |
| Membuat file dengan nama **konstanta.py** | PHI = 3.14  GRAVITASI = 9.8 |
| Membuat sebuah file dengan nama **main.py** | **import** konstanta  print(konstanta.PHI)  print(konstanta.GRAFITASI) |
| Maka Output | 3.14  9.8 |

Untuk penamaan dari variabel dan konstanta harus mengikuti beberapa aturan berikut:

* Nama dari suatu variabel hanya bisa terdiri dari huruf, angka dan garis bawah (underscore)
* Bisa diawali dengan huruf ataupun garis bawah tapi tidak diperbolehkan menggunakan awalan angka. Sebagai contoh, variabel bisa dinamai dengan nama\_1 tapi tidak dengan 1\_nama.
* Hindari menggunakan keyword atau sebuah fungsi sebagai nama dari suatu variabel.
* Nama dari suatu variabel harus simpel tapi bersifat deskriptif
* Hati-hati ketika menggunakan huruf kecil dari l dan huruf kapital dari O karena bisa disalah artikan sebagai angka 1 dan 0
* Statemen yang tidak boleh dijadikan nama variabel adalah keywords pada Python.

**Tipe Data**

|  |  |  |
| --- | --- | --- |
| Tipe Data | Fungsi | Format |
| str() | Untuk konversi type data ke String | %d |
| int() | Untuk konversi type data ke Interger | %f |
| float() | Untuk konversi type data ke Float | %s |

*Number*

Tipe data Number merepresentasikan nilai-nilai berupa angka. Python menggolongkan beberapa tipe data umum seperti integer (bilangan bulat) dan floating-point (bilangan desimal) ke dalam tipe data number.

*String*

Selain angka, python juga mampu melakukan manipulasi string, yang dapat diekspresikan dengan beberapa cara. Penulisan nilai string pada python menggunakan tandapetik satu ( ' ) atau tanda petik dua (“).

Operator String literal juga dapat menggabungkan beberapa baris dalam berbagai cara. Dengan menggunakan operator ( \n\ ) di akhir kalimat untuk menyambung kalimat selanjutnya yang berada di baris selanjutnya

Penulisan string untuk multiple line juga dapat dilakukan dengan menggunakan tanda petik dua atau satu sebanyak 3 kali, ( “ “ “ atau ' ' ' ). Untuk menggabungkan dua buah string atau lebih dapat dilakukan dengan dua cara. Pertama, dengan menulis langsung dua buah string yang diapit dengan tanda kutip atau dengan penggunaan operator tambah (+).

|  |
| --- |
| >>>"Belajar" "python"  'Belajarpython'  >>> 'Pulang' + 'Pergi'  'PulangPergi' |

Sebuah string, setiap karakternya dapat diindex, seperti pengindexan pada bahasa C.Karakter pertama pada sebuah string berindex 0, karakter ke-dua berindex 1 dan seterusnya.

|  |
| --- |
| >>> kata = "Veteran"  >>> kata[0]  'V'  >>> kata[4:7]  'eran'  >>> kata[:6]  'Vetera'  >>> kata[5:]  'ran' |

Python merupakan bahasa pemrograman yang fleksibel, python memiliki beberapa bentuk dalam melakukan proses print, yakni:

|  |
| --- |
| angka = 5  #Menggunakan format seperti C  print("1 | Nilai dari angka adalah : %d" %(angka))  #Menggunakan concatenation  #Karena proses print dalam bentuk string  #maka fungsi str() digunakan untuk mengubah integer menjadi string  print("2 | Nilai dari angka adalah : " + str(angka)) |

#Menggunakan pemisahan koma

print("3 | Nilai dari angka adalah :" ,angka)

#Menggunakan bentuk format()

print("4 | Nilai dari angka adalah : {}".format(angka))

#Menggunakan f-string

print(f"5 | Nilai dari angka adalah : {angka}")

OUTPUT :

>>

1 | Nilai dari angka adalah : 5

2 | Nilai dari angka adalah : 5

3 | Nilai dari angka adalah : 5

4 | Nilai dari angka adalah : 5

5 | Nilai dari angka adalah : 5

Selain itu tipe data variabel string memiliki beberapa fungsi khusus, yang digunakan untuk mengubah dan memanipulasi teks, seperti dibawah berikut:

|  |  |
| --- | --- |
| Nama | Keterangan |
| upper() | Mengonversi huruf kecil dalam bentuk string ke huruf besar. |
| lower() | Hapus item pada list sesuai dengan nilai yang didefinisikan |
| isalpha() | Mengembalikan true jika string memiliki minimal 1 karakter dan semua karakter adalah abjad dan false sebaliknya. |
| isnumeric () | Mengembalikan true jika string unicode hanya berisi karakter numerik dan false sebaliknya. |
| strip() | Mengubah kata psfs suatu kalimat menjadi *list* |

Berikut merupakan program dari setiap fungsi diatas yang diawali dengan deklarasi multi line string petik tiga (‘’’), program dibawah bertujuan untuk mendeteksi dan mengubah teks yang telah di deklarasikan, yakni:

1. Mendeteksi apakah seluruh teks tersusun atas huruf alphabet
2. Mendeteksi apakah seluruh teks tersusun bilangan numerik
3. Mengubah seluruh teks menjadi huruf kecil
4. Mengubah seluruh teks menjadi huruf besar
5. Mengubah seluruh teks menjadi list

# mendeklarasikan string multi line dengan petik 3

teks = '''

Saya adalah mahasiswa Teknik Elektro angkatan 2021

dengan NIM 2110314069

'''

# Mengecek apakah seluruh teks adalah abjad

# JIka benar akan mengembalikan nilai True

print(f"Apakah Teks berbentuk abjad semua ? : {teks.isalpha()}")

# Mengecek apakah seluruh teks adalah angka

# JIka benar akan mengembalikan nilai True

print(f"Apakah Teks berbentuk angka semua ? : {teks.isnumeric()}")

#mengubah teks kedalam huruf kecil semua

print(f"\nMengubah teks menjadi huruf kecil : {teks.lower()}")

#mengubah teks kedalam huruf kapital semua

print(f"Mengubah teks menjadi huruf kapital : {teks.upper()}")

#Mengubah teks ke dalam bentuk list

print(f"Mengubah kedalam bentuk list : \n{teks.split()}")

OUTPUT

>> Apakah Teks berbentuk abjad semua ? : False

>> Apakah Teks berbentuk angka semua ? : False

>>

Mengubah teks menjadi huruf kecil :

saya adalah mahasiswa teknik elektro angkatan 2021

Mengubah teks menjadi huruf kapital :

SAYA ADALAH MAHASISWA TEKNIK ELEKTRO ANGKATAN 2021

DENGAN NIM 2110314069

#Mengubah teks ke dalam bentuk list

print(f"Mengubah kedalam bentuk list : \n{teks.split()}")

['Saya', 'adalah', 'mahasiswa', 'Teknik', 'Elektro', 'angkatan', '2021', 'dengan', 'NIM', '2110314069']

Mengubah kedalam bentuk list :

dengan nim 2110314069

***Format Penulisan Integer:***

|  |  |
| --- | --- |
| Format | Keterangan |
| %f | Akan menampilkan 6 digit angka dibelakang koma |
| %xf | Akan menampilkan sebanyak x digit dengan spasi sebagai pemisah |
| %0xf | Akan menampilkan angka sebanyak x digit dengan 0 sebagai pemisah |
| %.xf | Akan menampilkan x digit dibelakang koma |
| \*catatan x diisi dengan angka yang diinginkan | |

***Operator***

*Operator Matematika dan Operator Penugasan*

|  |  |
| --- | --- |
| Operator | Keterangan |
| \* | Perkalian |
| / | Pembagian |
| % | Modulo |
| + | Penjumlahan |
| - | Pengurangan |

|  |  |  |
| --- | --- | --- |
| Operator | Contoh | Sama dengan |
| \*= | x \*= 50 | x = x \* 50 |
| /= | x /= 50 | x = x / 50 |
| %= | x %= 50 | x = x % 50 |
| += | x += 50 | x = x + 50 |
| -= | x -= 50 | x = x - 50 |

*Operaator Perbandingan*

|  |  |  |  |
| --- | --- | --- | --- |
| Operator | Keterangan | Operator | Keterangan |
| >= | Lebih besar atau sama dengan | < | Lebih kecil |
| <= | Lebih kecil atau sama dengan | > | Lebih besar |
| != | Tidak sama dengan | == | Sama dengan |

**Contoh Program:**

***Contoh 1: Mengukur Luas Lingkaran***

# Mengukur luas lingkaran

pi = 3.14159

jari\_jari = input("Masukkan nilai jari-jari : ")

# Mengubah jari2 ke dalam float

jari\_jari = float(jari\_jari)

luas\_lingkaran = pi \* jari\_jari \* jari\_jari

print(f"Luas dari lingakaran adalah : {luas\_lingkaran}")

OUTPUT:

>> Masukkan nilai jari-jari : 7

>> Luas dari lingakaran adalah : 153.93791