|  |  |
| --- | --- |
| A picture containing text, clipart  Description automatically generated | **Laporan Praktikum Algoritma dan Pemrograman**  Semester Genap 2023/2024 |

|  |  |
| --- | --- |
| **NIM** | **<71231038>** |
| **Nama Lengkap** | **<AMELIA AGUSTIN>** |
| **Minggu ke / Materi** | **02 / 2. Variable, Expression dan Statements** |

**SAYA MENYATAKAN BAHWA LAPORAN PRAKTIKUM INI SAYA BUAT DENGAN USAHA SENDIRI TANPA MENGGUNAKAN BANTUAN ORANG LAIN. SEMUA MATERI YANG SAYA AMBIL DARI SUMBER LAIN SUDAH SAYA CANTUMKAN SUMBERNYA DAN TELAH SAYA TULIS ULANG DENGAN BAHASA SAYA SENDIRI.**

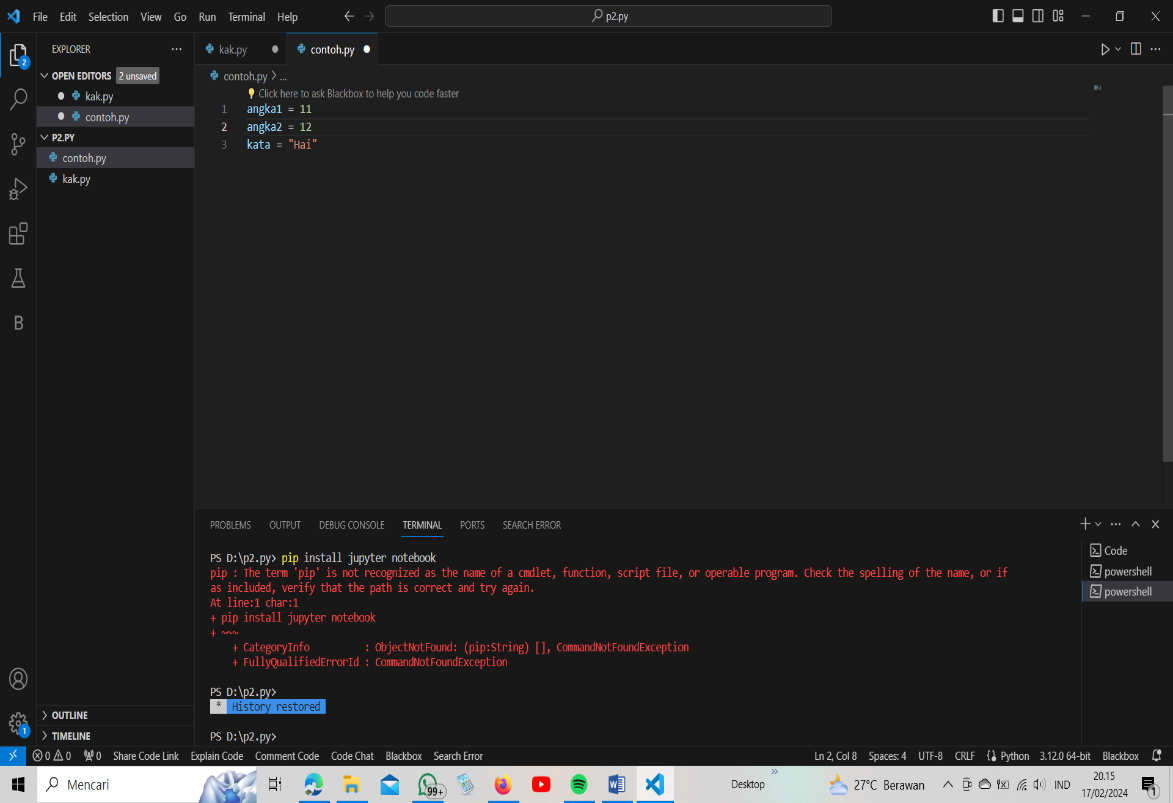
**SAYA SANGGUP MENERIMA SANKSI JIKA MELAKUKAN KEGIATAN PLAGIASI, TERMASUK SANKSI TIDAK LULUS MATA KULIAH INI.**

**PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS KRISTEN DUTA WACANA  
YOGYAKARTA  
2024**

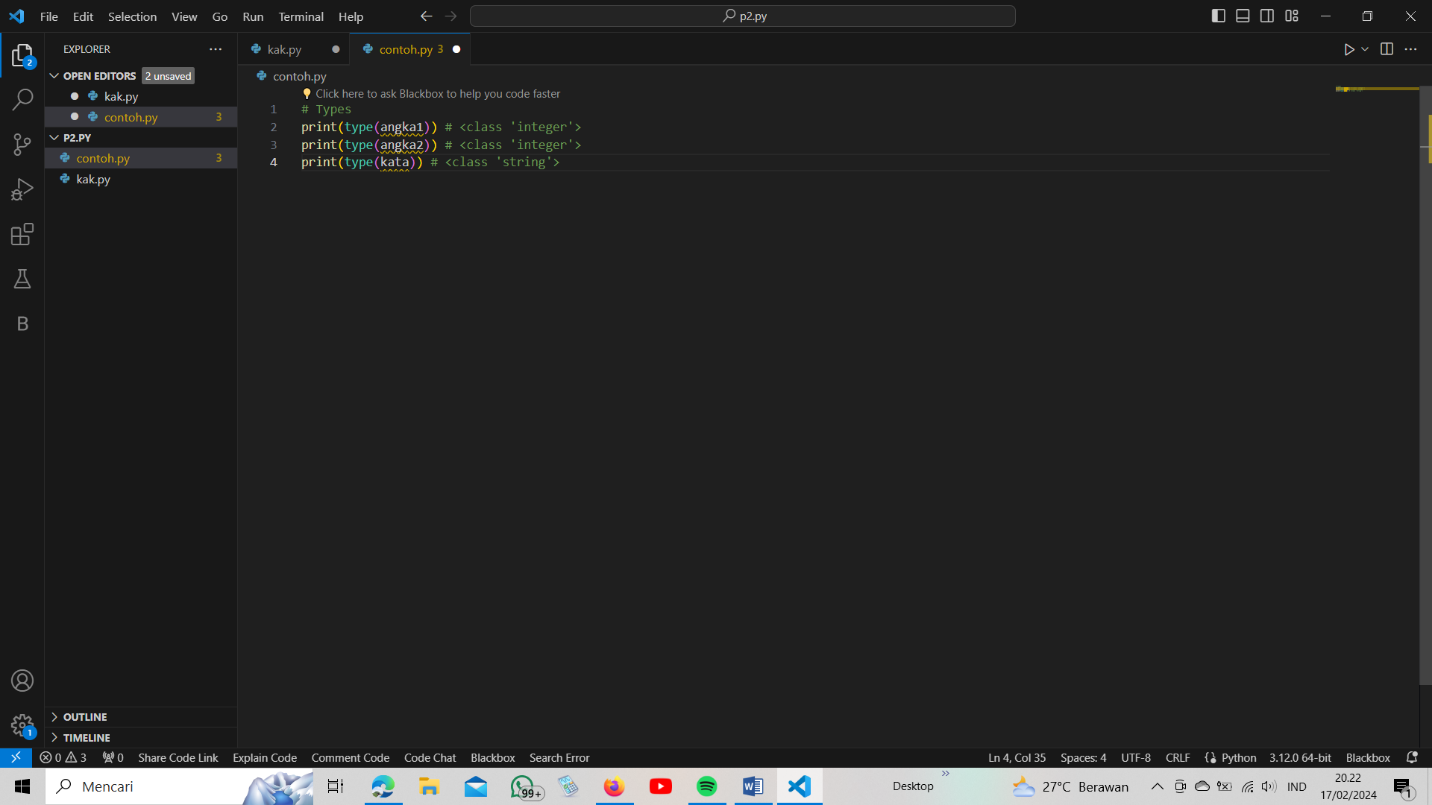
# BAGIAN 2: 2. Variable, Expression dan Statements

**2.3.1 Values dan type**

Values and type merupakan dua konsep yang dimana sangat berbeda namun memiliki hubungan yang erat atau saling keterkaitan satu sama lain dalam representasi data di program komputer. Namun keduanya berbeda, values merepresentasikan data itu sendiri sedangkan types mengatur data tersebut. Bisa di perjelas bahwa values adalah data, seperti angka, teks, atau objek dalam program komputer, namun ada dua tipe yaitu antara integer dan string misalnya adalah jika 5 untuk nilai integer sedangkan “Hello World” untuk sebuah string.



Sedangkan untuk type sendiri merupakan setiap value memiliki sebuah type yang menentukan bagaimana ia bisa digunakan dan dioperasikan. Type menentukan operasi apa saja yang valid untuk dilakukan pada suatu value. Misalnya kita bisa menjumlahkan dua integer tapi tidak bisa menjumlahkan integer dengan string. Beberapa contoh umum types adalah number, boolean, string, array, object, dll.



Selain value berkerja di python menggunakan string ada beberapa seperti seperti integer (bilangan bulat), float(bilangan pecahan), character (huruf), atau bool (benar/salah). Dengan begitu python menyediakan untuk melakukan pegecekan tipe data pada value dengan menggunakan fungsi type ()

# string

data = "Ini string"

print(data)

print(type(data))

# integer

angka = 10

print(angka)

print(type(angka))

# float

desimal = 9.5

print(desimal)

print(type(desimal))

## MATERI 1

**2.3.2 Variabel**

Variabel merupakan wilayah penyimpanan yang dicadangkan untuk menyimpan nilai-nilai, yang dapat diartikan bahwa jjika membuat sebuah variabel akan terjadi memesan beberapa ruang di penyimpanan dan varibael akan menyimpan data yang dilakukan selama program tersebut dijalankan, dimana variabel tersebut dapat diubah oleh operasi-operasi tertentu pada program dengan menggunakan variabel.

*# variabel dengan tipe data string*

nama = "John Doe"

*# variabel dengan tipe data integer*

umur = 25

*# variabel lain dengan tipe data integer*

harga = 15000

Potongan program di atas merupakan contoh dari penggunaan variabel. Untuk contoh pertama merupakan variabel nama yang berisi string, untuk contoh kedua merupakan variabel umur yang berisikan integer, dan untuk contoh yang terakhir ini merupakan variabel lain yang berisikan harga.

Di dalam variabel sendiri dapat menyimpan berbagai macam tipe data. Dalam bahasa program python, variabel memiliki sifat yang dinamis yang bisa diartikan variabel tersebut dapat berubah-ubah nilianya saat program dijalankan, artinya nilai variabel tersebut dapat diperbarui sewaktu-waktu berdasarkan logika progam yang ada. Sesuai dengan penyimpanan data yang berpotensi berubah variabel dinamis cocok digunakan untuk menampung jenis data yang memiliki kemungina nilainya berbeda atau berubah-ubah dari waktu-waktu.

**2.3.3 Nama Variabel dan Keywords**

Variabel memiliki nama atau identitas unik yang digunakan untuk mengakses dan memanipulasi data yang disimpan di dalamnya. Dalam menentukan nama sebuah variabel, ada aturan main yang harus di ikuti, diantaranya:

1. Nama variabel tidak boleh ssama dengan kata kunci(keywords) yang sudah ditentukan oleh bahasa pemograman yang digunakan. Kata kunci merupakan kata yang dimana memiliki fungsi dan makna spesifik dalam sintaks bahasa pemograman.

2. Panjang nama variabel dapat lebih sari satu kata, misal( first\_name, last\_name, current\_balance.

3. Karakter prtama dari nama variabel harus berupa huruf atau underscore, tidak boleh diawali dengan angka.

4. Nama variabel sebisa mungkin harus memperesentasikan data apa yang disimpan di dalamnya agar memudahkan pemahanan program.

Nama variabel harus unik, tidak ambigu, dan mengikuti aturan penamaan baku bahasa pemograman yang digunakan agar terhindar dari konflik kata.

Python sendiri memiliki 35 keyword yang tidak boleh digunakan untuk memberi nama variabel.

**and del from None True as elif global nonlocaly try**

**assert else if not while break except import or width**

**class False in pass yield continue finally is raise async**

**def for lamda return wait**

Berikut ini merupakan contoh pemograman bahasa python:

1. Deklarasi Variabel

Dimana dalam pyrhon ini cukup menetapkan nilai ke variabel tersebt tanpa harus mendeklarasikan tipe datanya secara ambigu, contoh:

# variabel bertipe integer

umur = 25

# variabel bertipe string

nama = "John Doe"

2. Inisialisasi Variabel

Dalam python ini langsung bisa menganalisis nilai ke dalam variabel saat pertama kali deklarasi, contoh:

# langsung inisialisasi saat deklarasi

gaji = 15000000

alamat = "Jln. Anggrek No. 75"

3. Menimpa Nilai Variabel

Nilai variabel dapat ditimpa dengan cara assign ulang variabel tersebut. Contoh:

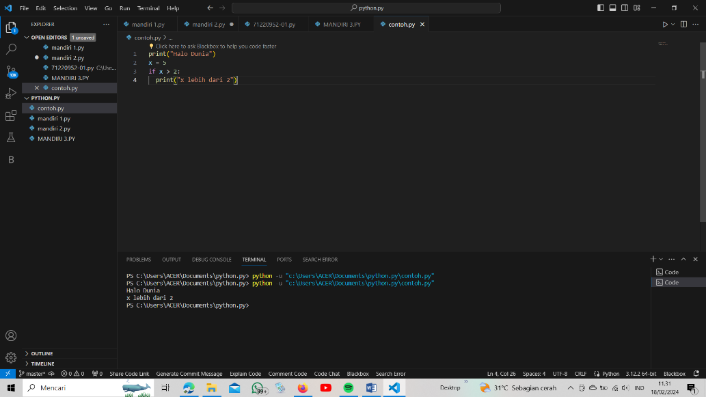
umur = 20 # nilai awal umur adalah 20

umur = 25 # lalu diubah menjadi 25

## MATERI 2

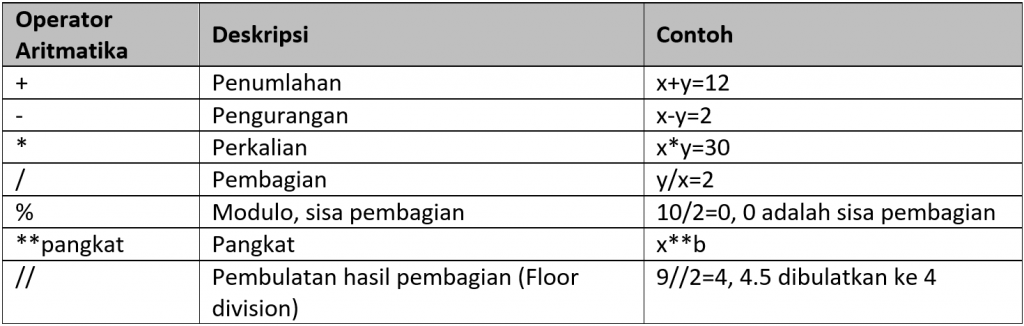
**2.3.4 Statements**

Statement dalam bahasa python merupakan baris per baris perintah program yang dijalankan secara berurutan untuk melakukan tugas-tugas tertentu yang diinginkan oleh program. Setiap baris statement menjalankan satu tugas spesifik dan hasilnya digunakan untuk menjalankan statement berikutnya demo mencapai tujuan akhir program secara menyeluruh. Dengan adanya mekanisme dijalankan statement per statement secara berurutan dan tersturktur inilah yang memungkinkan program python bekerja sessuai fungsional yang diinginkan. Statement yang satu akan menjalankan terlebih dahulu tugasnya, kemudian output atau hasil dijalankan statement tersebut dapat dimanfaatkan statement lain sebagai input untuk melanjutkan proses menjalankan berikutnya hingga program python berjalan dengan akurat dan sukses menyeluruh. Jadi sederhananya, statement ini bertanggung jawab untuk menjalankan atau mengeksekusi bari demi baris intruski program untuk mencapai yang diinginkan.

Berikut merupakan contoh dari satement dan outputnya :

**2.3.5 Operator dan Operand**

Operator sendiri merupakan simbol-simbol yang tertentu dalam pemograman yang digunakan untuk melakukan manipulasi, pebandingan, perhitungan matematika, atau evaluasi logika terhadap suatu nilai, variabel, atau ekspresi. Operator mewakili semacam aksi atau proses yang akan diterapkan pada satu atau beberapa operand.

Sedangkan operand itu sendiri merupakan nilai yang merujuk pada variabel, atau ekspresi yang menjadi “target” dari operator. Operand akan dioperasikan oleh operator. Tanpa operand, operator tidak dapat melakukan tugasnya. Pada python ini operator menggunakan jenis aritmatika, operator aritmatika merupakan operator yang digunakan untuk melakukan operasi matematika, seperti penjumlahan, pengurangan, perkalian, pembagian, dan sebagainya. Berikut tabel menunjukkan jenis operator aritmatika:

Berikut merupakan contoh operator aritmatika dalam bahasa python:

# Penjumlahan

angka1 = 10

angka2 = 20

jumlah = angka1 + angka2

print(jumlah) # Output: 30

# Pengurangan

angka1 = 15

angka2 = 10

kurang = angka1 - angka2

print(kurang)

# Output: 5

# Perkalian

angka1 = 5

angka2 = 4

kali = angka1 \* angka2

print(kali) # Output: 20

# Pembagian

angka1 = 20

angka2 = 2

bagi = angka1 / angka2

print(bagi)

# Output: 10.0

# Sisa bagi (modulus)

angka1 = 29

angka2 = 3

mod = angka1 % angka2

print(mod) # Output: 2

# Pangkat (exponent)

pangkat1 = 2 \*\* 4 # 2 pangkat 4

print(pangkat1) # Output: 16

pangkat2 = 5 \*\* 2 # 5 pangkat 2

print(pangkat2) # Output: 25

# Pembagian bulat ke bawah

pembulatan\_bawah = 10 // 3 # pembagian bulat hasilnya 3

print(pembulatan\_bawah) # Output: 3

## MATERI 3

## **2.3.6 Expressions**

Expressions atau ekspresi pada dasarnya adalah kombinasi satu atau lebih nilai konstanta(literal), variabel, operator, dan panggilan fungsi yang dievaluasi oleh interpreter bahasa pemograman menjadi satu nilai tunggal (single value). Ekspresi sendiri merupakan building block (blok pembangun) paling mendasar dalam bahasa pemograman. Tnapa ekspresi, kode program tidak dapat berbuat banyak hal.

Ekspresi dapat berdiri sendiri layaknya satu statement, atau digabungkan dengan statement lain sebagai bagian dari statement yang lebih besar dan kompleks seperti pada percabangan/perulangan. Nilai hasil evaluasi (result) dari ekspresi dapat digunakan untuk berbagai keperluan seperti melakukan penugasan nilai (assignment), mencetak keluaran, maupun pengambilan keputusan (decision making). Berikut ini adalah contoh program python pada expressions:

# Aritmatic Expressions

nilai1 = 5 \* (10 + 3) # evaluasi urutan operasi dalam ()

nilai2 = 15 / 3 # pembagian desimal

total = nilai1 + 10 # penjumlahan

print(total)

# String Expressions

nama\_depan = "Budi"

nama\_belakang = "Susanto"

nama\_lengkap = nama\_depan + " " + nama\_belakang

print(nama\_lengkap)

**2.3.7 Urutan Operasi**

Saat interpreter python mengevaluasi suatu expressions yang memiliki beberapa operator didalamnya, terdapat aturan tetang operator yang harus dijalankan terlebih dahulu. Aturan inilah yang disebut urutan operasi atau presendence. Memahami urutan operasi sangat penting agar logika program sesuai dengan yang diharapkan.

Pertama, expressions dalam tanda kurung akan selalu dievaluasi terlebih dahulu sebelum opeartor di luarnya. Lalu, operator perpangkatan akan dievaluasi sebelum perkalian dan pembagian, begitu juga perkalian dan pembagian akan mendahului penjumlahan dan pengurangan. Operator perbandingan dievaluasi setelah operasi matematika, dan operator logika paling akhir. Secara horizontal, operasi yang setara akan dieksekusi dari kiri ke kanan urutannya. Dengan memahami presedence atau urutan ini, kita bisa membuat ekspresi matematika dan logika sesuai keinginan, entah itu untuk melakukan kalkulasi, membandingkan nilai, membuat keputusan, dan kebutuhan program lainnya. Tanda kurung juga bisa digunakan secara manual untuk mengatur ulang priorita operator jika diperlukan. PEMDAS merupakan singkatan yang digunakan untuk mengingat urutan operasi dalam matematika dan pemograman

P : Parentheses ()

E : Exponents (Perpangkatan)

MD : Multiplication and Division (Perkalian dan pembagian)

AS : Addtion and Subtraction (Penjumlahan dan Pengurangan)

1. Operasi di dalam tanda kurung lebih dahulu dievaluasi. Misalnya ( 5 + 3 ) akan dihitung menjadi 8 terlebih dahulu.

2. Operasi perpangkatan dievaluasi berikutnya. Contoh 3^4 akan dihitung menjadi 81.

3. Operasi perkalian dan pembagian dieksekusi berurutan dari kiri ke kanan. Misalnya pada 4 x 5 / 2 = 10.

4. Terakhir operasi penjumlahan dan pengurangan juga dievaluasi dari kiri ke kanan. Seperti pada 2 + 4 – 3 = 3.

Jadi tentang PEMDAS ini penting agar dapat membuat ekspresi matematika maupun logika pemograman dengan urutan kalkulasi yang tepat dan akurat. Tanda kurung bisa “mengoceh” evaluator agar menghitung bagian dalam kurung terlebih dahulu.

**2.3.8 Operator Modulus dan String**

Operator modulus (%) merupakan sisa hasil yang dimana dibagi dari bilangan pertama dengan bilangan kedua. Operator ini hanya berlaku pada tipe data integer.

Berikut merupakan contoh penggunaan operator modulus % pada tipe data integer di Python:

# Modulus positif

print(10 % 3) # 1

print(15 % 4) # 3

# Modulus negatif

print(-10 % 3) # -1

print(-15 % 4) # -3

# Modulus 0

print(10 % 5) # 0

Berikut merupakan cara kerja:

1. Jika bilangan positif, modulus menghasilkan sisa baginya

2. Jika bilangan negatif, modulus juga negatif

3. Jika habis dibagi, modulusnya adalah 0

Jadi operator modulus (%) adalah operator aritmatika yang sangat berguna dalam banyak situasi yang melibatkan bilangan bulat.

**String**

Operator + ketika bekerja dengan string tidak berarti penjumlahan secara matematika, melainkan penggabungan antar string. Contoh:

>>> first = '20'

>>> second = '30'

>>> print(first + second)

2030

Operator \* juga bekerja dengan string dengan melakukan perkalian antara content string dan integer

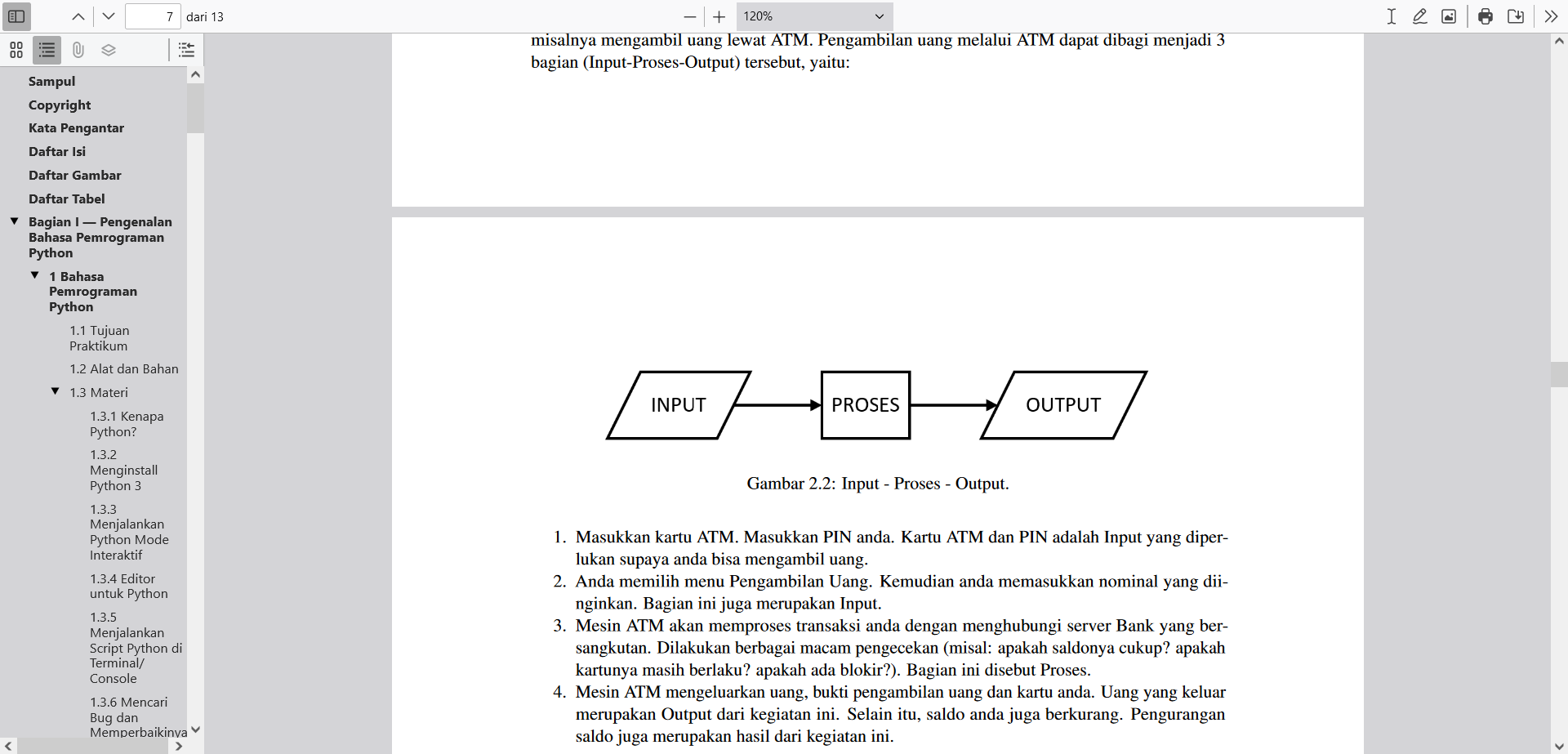
>>> first = 'Test '

>>> second = 3

>>> print(first \* second)

Test Test Test

**2.3.9 Menangani Input dari Pengguna**

Menerima input dari pengguna sangatlah penting agar program python yang dibuat bisa berinteraksi dan menerima data dari user. Untuk menangkap inputan pengguna, python menyedikan fungsi input() bawaan. Fungsi input() menerima sebuah prompt string sebagai parameter masukan dan menunggu user memasukkan data. Fungsi mengembalikkan data yang dimasukkan oleh user sebagai string. Menyimpan string bisa input user tersebut ke dalam variabel untuk diproses lebih lanjut. Kadang juga perlu mengubah tipe data inputan mejadi tipe lain seperti integer atau float. Untuk itu digunakan casting tipe data dengan fungsi int(), float() atau yang lain, passing input() ke parameter fungsi tersebut. Dengan fungsi input dan csting, user dapat memasukkan berbagai jenis data program python seperti string, angka dan lainnya

Berikut merupakan langkah-langkah dalam pemesanan makanan onine

1. penggunaan membukaan menu “makanan” pada aplikasi pemesanan makanan.

2. pengguna disajikan dengan berbagai pilihan kategori makanan seperti bakso, pecel lele, soto, nasi goreng dan lainnya.

3. pengguna memilih salah satu kategori minuman yang diinginkan, misalnya bakso.

4. kemudian pengguna diperlihatkan dengan ragam pilihan variasi yang tersedia, seperti bakso beranak, bakso telur, baksu urat dan lainnya.

5. setelah memutuskan variasi bakso, pengguna juga dapat menentukan rasa kepedesan yang di inginkan.

6. jika pengguna ingin menambahkan topping atau mengurangi topping, dapat menggunakan kolom catatan tambahan.

7. Setelah selesai mengcustom bakso yang diinginkan, pengguna menekan tombol “ tambah ke keranjang” agar minuman tersebut masuk ke dalam pesanan.

8. pengguna dapat menggulangi langkah 2 sampai 7 untuk menambahkan menu lainnya ke dalam keranjang belanja.

Python punya fitur bawaan yang memiliki nama input (). Fungsinya merupakan mendapatkan data yang dimasukkan lewat keyboard.

Cara memakainya seperti, tinggal memanggil fungsi input (), lalu program python akan berhenti dahulu, bisa ketuk sesuatu di keyboard. Setelah itu tekan enter, program akan jalan kembali. Input () bawa data yang sudah diketik, data nya berupa text alias string. Input () sangat berguna untuk membuat program python yang interaktif. Misalnya kalkulator sederhana, tebak-tebakan, formulir, dan program lain yang butuh input.

nama = input("Siapa nama kamu? ")

print("Halo", nama)

Sebelum menggunakan input (), kita harus memberikan petunjuk terlebih dahulu untuk user. Petunjuknya berupa kalimat yang memberikan tahukan data apa yang harus di input.

nama = input("Silahkan masukkan nama Anda")

dengan adanya petunjuknya, user akan menjadi paham untuk melakukan hal yang selanjutnya. Inputnya akan pastinya sesuai dengan yang diminta program.

nama = input("Siapa nama kamu? \\n")

jika memberikan \n di akhir prompt, berarti memberikan perintah ganti baris.

Jika ingin menerima input angka dari user, bisa mengubah ke tipe data integer()

umur = int(input("Berapa umur kamu? \\n"))

inputan umur yang tadinya tipe integer, bukan string. Program ini bisa dioperasikan dalam kalkulasi angka.

**2.4 Komentar**

Tanda pagar ( # ) digunakan untuk menandai komentar di python. Komentar tidak akan diproses oleh interpreter python. Komentar hanya berguna untuk programmer untuk memudahkan dalam memahami arti dari kode.

# Ini adalah komentar satu baris

# BAGIAN 2: LATIHAN MANDIRI (60%)

Pada bagian ini anda menuliskan jawaban dari soal-soal Latihan Mandiri yang ada di modul praktikum. Jawaban anda harus disertai dengan source code, penjelasan dan screenshot output.

## SOAL 1

tinggi\_badan = float(input("Masukkan tinggi badan (dalam m): "))

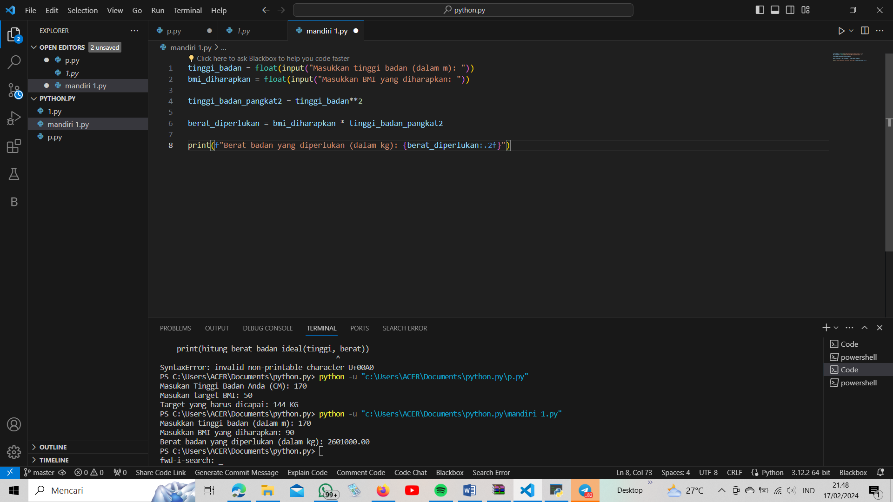
bmi\_diharapkan = float(input("Masukkan BMI yang diharapkan: "))

tinggi\_badan\_pangkat2 = tinggi\_badan\*\*2

berat\_diperlukan = bmi\_diharapkan \* tinggi\_badan\_pangkat2

print(f"Berat badan yang diperlukan (dalam kg): {berat\_diperlukan:.2f}")

Program ini dengan menjabarkan beberapa variabel kosong yang akan digunakan, seperti tinggi\_badan, BMI\_diharapkan, tinggi\_badan\_pangkat2 dan berat\_diperlukan. Kemudian diminta untuk memasukkan data tinggi badan dalam satuan meter ke dalam variabel tinggi\_badan untuk selanjutnya diubah ke tipr data float agar bisa dilakukan operasi matematika. Selanjutnya program ini meminta agara untuk memasukan angka BMI yang diinginkan dan menyimpannya ke dalam variabel BMI\_diharapkan, juga di ubah ke tipe float. Lalu program ini akan menghtiung tinggi badan pangkat2 dan meyimpannya ke variabel tinggi\_badan\_pangkat2 untuk digunkana dalam rumus BMI merupakan BB = BMI x Tinggi ^2 Kemudian nilai BMI yang diinput akan dikalikan dengan nilai tinggi\_badan\_pangkat2 untuk menghasilkan berat badan ideal yang akan disimpan ke variabel berat\_dipelukan. Lalu terakhir program akan mencetak nilai berat\_diperlukan dengan pesan keterangan dan membulatkan ke dua angka di belakang koma agar rapi.



## SOAL 2

# Program menghitung nilai fungsi f(x)

# Fungsi

def f(x):

    return 2\*x\*\*3 + 2\*x + 15

# Input nilai x

x = int(input("Masukkan nilai x: "))

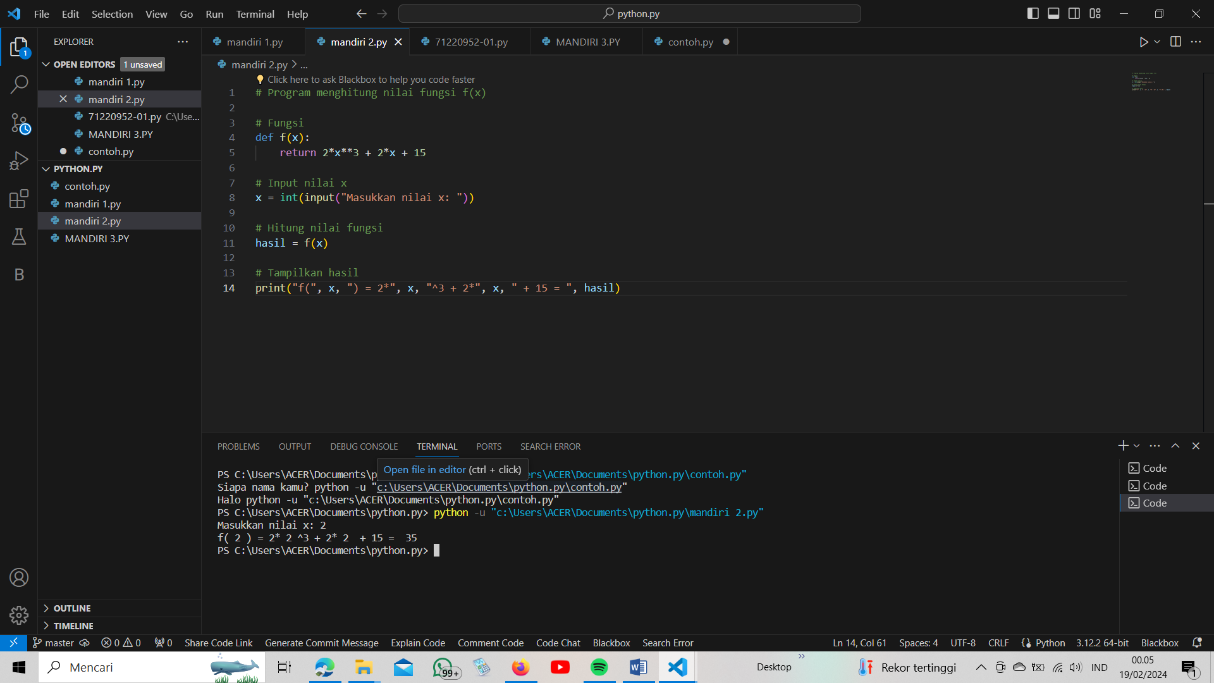
# Hitung nilai fungsi

hasil = f(x)

# Tampilkan hasil

print("f(", x, ") = 2\*", x, "^3 + 2\*", x, " + 15 = ", hasil)

jadi hal yang pertama yang di lakukan merupakan memasukan rumus yang ada, lalu membuat program yang bisa menghitung f(x) nya, tampilkan tulisan “masukkan nilai x:” , lalu ketik angka misalnya “2”, program simpan dengan angka x yang diinput, program akan dihitung 2 x 2 pangkat 3 (jawab 16), tambahkan dengan 2 x 2 (jawab 4 ), tambahkan 15 , totalnya (16 + 4 + 15 = 35 ) jadi program hitungnya adalah ambil input x lalu hitung 2 \*x \*x\*x tambah 2 \* x tambah 15 lalu cetak hasil totalnya.



## SOAL 3

# Input

gaji\_per\_jam = int(input("Masukkan gaji per jam: "))

total\_jam\_per\_minggu = int(input("Masukkan total jam kerja per minggu: "))

# Hitung pendapatan kotor (sebelum pajak)

pendapatan\_kotor = gaji\_per\_jam \* total\_jam\_per\_minggu \* 5

# Total pajak (diasumsikan 14%)

pajak = pendapatan\_kotor \* 0.14

# Pendapatan setelah pajak

pendapatan\_bersih = pendapatan\_kotor - pajak

# Persentase untuk beli baju dan aksesoris (10%)

persen\_baju\_aksesoris = 0.10

# Uang untuk baju dan aksesoris

baju\_aksesoris = pendapatan\_bersih \* persen\_baju\_aksesoris

# Persentase untuk alat tulis (1%)

persen\_alat\_tulis = 0.01

# Uang untuk alat tulis

alat\_tulis = pendapatan\_bersih \* persen\_alat\_tulis

# Sisa uang

sisa\_uang = pendapatan\_bersih - baju\_aksesoris - alat\_tulis

# Uang untuk disedekahkan (25% dari sisa uang)

sedekah = sisa\_uang \* 0.25

# 30% sedekah untuk anak yatim

anak\_yatim = sedekah \* 0.3

# Sisanya untuk kaum dhuafa

kaum\_dhuafa = sedekah - anak\_yatim

# Output

print(pendapatan\_kotor)

print(pendapatan\_bersih)

print(baju\_aksesoris)

print(alat\_tulis)

print(sedekah)

print(anak\_yatim)

print(kaum\_dhuafa)

Input yang dimina adalah gaji perjam budi dan jam kerja per minggu, untuk menghitung semua presentase pengeluaran uang budi selama 5 minggu kerja. Masukan input rumusnya lalu hal pertama adalah kalikan jam kerja per minggu dengan 5 minggu, dapat total jam kerja, dari situ bisa menghitung gaji kotor (sebelum diptong pajak), lalu menghitung uang pajaknya kemudian dikurangi, Budi menggunakan 10% dari uang sisa untuk beli baju dan aksesoris, 1 % lagi untuk beli alat tulis, 25% dari uang sisanya untuk disedekahkan, 3% sedekah untuk anak yatim, sisanya kaum dhuafa. Setelah itu outputnya masukkan gaji per jam nya. Jadi program ini berurutan hitung pendapatan kotor, pendapatan bersih, berapa yang dibelanjakan dan disedekahkan, lalu tampilkan masing-masing hasil perhitungan dari pengeluaran uang budi.

