



Laporan Praktikum Algoritma & Pemrograman

Semester Genap 2025/2026

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.

NIM	71251189
Nama Lengkap	Alicia Luna Santoso
Minggu ke / Materi	02 / Variable, Expression, dan Statements

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
YOGYAKARTA
2026

BAGIAN 1: MATERI MINGGU INI (40%)

Pada bagian ini, tuliskan kembali semua materi yang telah anda pelajari minggu ini. Sesuaikan penjelasan anda dengan urutan materi yang telah diberikan di saat praktikum. Penjelasan anda harus dilengkapi dengan contoh, gambar/ilustrasi, contoh program (source code) dan outputnya. Idealnya sekitar 5-6 halaman.

Link Github : https://github.com/alc357/71251189_alicia.git

VALUES DAN TYPE

Value adalah komponen utama suatu program, dapat berupa huruf atau angka. Contohnya seperti 1,2,'a','b', dan "Hello World". *Value* terdiri dari beberapa tipe seperti tipe *string* yaitu tulisan, tipe *integer* yaitu bilangan bulat, *float* yaitu bilangan pecahan, *character* yaitu huruf, dan *bool* yaitu benar/salah. Interpreter dapat melakukan identifikasi terhadap tipe *string* karena penulisannya menggunakan tanda petik " " (question mark). Untuk memahami values dan type berikut merupakan contoh kode dengan python interaktif.

```
>>> print(3)
3
>>> print(10.357)
10.357
>>> print('A')
A
>>> print('False')
False
```

Gambar 1.1 : values dalam python interaktif

Python menyediakan fungsi built-in untuk melakukan pengecekan tipe data pada value menggunakan fungsi *type()*

```
>>> x = 5
>>> print (x, "tipenya adalah ", type(x))
5 tipenya adalah <class 'int'>
>>> x = 3.5
>>> print (x, "tipenya adalah ", type(x))
3.5 tipenya adalah <class 'float'>
>>> x = "i"
>>> print (x, "tipenya adalah ", type(x))
i tipenya adalah <class 'str'>
```

Gambar 1.2 : menggunakan fungsi *type ()*

Ketika menggunakan bilangan bulat besar, beberapa model penulisan menggunakan tanda koma (,) diantara kelompok tiga digit akan dianggap sebagai bilangan bulat. Berikut contohnya.

```
>>> print(1,000,000)
1 0 0
```

Gambar 1.3 : penggunaan tanda koma dalam tiga digit

Hal ini dapat terjadi karena Python menganggap bahwa 1,000,000 merupakan kiriman parameter sebanyak 3 parameter pada fungsi print yaitu 1, 0, dan 0.

VARIABEL

Python memiliki kemampuan untuk memanipulasi *variable*. *Variable* adalah lokasi memori yang dicadangkan untuk menyimpan nilai. Saat membuat variable, kita memesan ruang dalam memori. *Variable* menyimpan data yang dilakukan selama program dieksekusi dimana isinya dapat diubah oleh operasi-operasi tertentu pada program yang menggunakan *variable*.

```
>>> notifikasi = 'Halo,Selamat Pagi. Waktunya Kuliah'
>>> a = 3
>>> ipk = 3.90
```

Gambar 1.4 : penulisan variabel dalam python

Variabel dapat menyimpan berbagai macam tipe data. Dalam Python, variable bersifat dinamis sehingga tidak perlu dideklarasikan tipenya dan dapat diubah saat program dijalankan. Kita dapat menggunakan perintah print untuk menampilkan variabel.

```
>>> print (ipk)
3.9
```

Gambar 1.5 : penggunaan print untuk memunculkan variabel

NAMA VARIABEL DAN KEYWORDS

Acuan dalam menulis variabel adalah sebagai berikut :

1. Nama variabel boleh diawali dengan huruf atau garis bawah (_). Contoh : harga, _harga;
2. Karakter selanjutnya dapat berupa huruf, garis bawah, atau angka;
3. Karakter pada nama variabel bersifat sensitif (*case sensitive*). Artinya huruf besar dan kecil dibedakan. Contohnya variabel Uang dan uang akan menjadi variabel yang berbeda;
4. Nama variabel tidak boleh menggunakan kata kunci yang ada dalam python. Berikut daftarnya

and	continue	except	global	lamda	pass	while
as	def	false	if	none	raise	width
assert	del	finally	import	nonlocaly	return	yield
break	elif	for	in	not	true	async
class	else	from	is	or	try	Wait

Berikut merupakan contoh penggunaan variabel dalam Bahasa Python :

```
#proses memasukan data dalam variabel
nama = "Alicia Luna Santoso"

#proses mencetak variabel
print(nama)

#nilai dan tipe data dalam variabel dapat diubah
umur = 18
print(umur)
type (umur)
umur = "delapan belas" #nilai setelah diubah
print(umur) #mencetak nilai umur
type (umur) #mengecek tipe data umur
namaDepan = "Alicia"
namaBelakang = "Santoso"
hobi = "Main Musik"
print("Biodata\n", nama, "\n", umur, "\n", hobi)
```

Gambar 1.6 : penggunaan variabel dalam python

STATEMENTS

Statements adalah bagian code interpreter Python yang dapat dieksekusi. Misal pada statements print, dapat berupa expression statements dan assignment. Saat menggunakan python mode interaktif, interpreter secara langsung melakukan eksekusi dan menampilkan hasilnya. Namun jika menggunakan script mode, script biasanya berisi statements yang saling berhubungan secara sekuensial.

```
1 print (3)
2 x = 5
3 print (x)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\prak_alpro\praktikum alpro - 71251189\Praktikum-Alpro---71251189> &thont.exe" "c:/prak_alpro/praktikum alpro - 71251189/Praktikum-Alpro---71251189"
3
5
PS C:\prak_alpro\praktikum alpro - 71251189\Praktikum-Alpro---71251189>

Gambar 1.7 : Contoh statement print dan outputnya

OPERATOR DAN OPERAND

Operator adalah symbol yang digunakan untuk melakukan operasi aritmatika maupun logika. Sedangkan nilai yang dioperasikan disebut operand. Misalnya 3 + 5 maka tanda + adalah operator penjumlahan sedangkan 3 dan 5 adalah operand. Operator aritmatika dalam Python yaitu :

- (+) sebagai operator penjumlahan, contoh penulisannya yaitu $x + y$
- (-) sebagai operator pengurangan, contoh penulisannya yaitu $x - y$
- (*) sebagai operator perkalian, contoh penulisannya yaitu $x * y$

- (/) sebagai operator pembagian, contoh penulisannya yaitu x / y
- (**) sebagai operator pemangkatan, contoh penulisannya yaitu $x ** y$

```
>>> 20 + 52
72
>>> hour = 20
>>> print (hour-3)
17
>>> minute = 60
>>> print (hour*2+minute)
100
>>> print (minute/60)
1.0
>>> 3**2
9
>>> (3+5)*(9-2)
56
>>>
```

Gambar 1.8 : Penggunaan operator dalam Python

EXPRESSION

Expression adalah representasi dari nilai dan dapat terdiri dari gabungan antara *values*, *variable*, dan *operator*. *Values* dapat dianggap sebagai *expression* dan variabel. Secara umum, semua dapat disebut sebagai *expression*.

```
17
x
x+17

>>> 1 + 1
2
>>> 3 + 2
5
```

Gambar 1.9 : Contoh penggunaan expression

Ketika menggunakan expression dalam mode interaktif, interpreter akan melakukan evaluasi dan menampilkan hasilnya.

URUTAN OPERASI

Urutan operasi berlaku bila ada lebih dari satu operator dalam expression. Urutan bergantung pada prioritas. Untuk operasi matematika, Python mengikuti konvensi matematika. Urutan operasi sering disingkat dengan **PEMDAS** yaitu :

- Paranthese (tanda kurung) : adalah prioritas tertinggi dan digunakan untuk mendahulukan expression dalam urutan yang sesuai. Contoh $3*(3-1)$ maka $(3-1)$ dikerjakan terlebih dahulu baru dikalikan dengan 3.
- Exponetiton (eksponensial/pemangkatan) : merupakan urutan prioritas berikutnya. Contohnya $2**1+1$ maka 2 dipangkatkan 1 terlebih dahulu baru ditambah 1 bukan 2 dipangkatkan dengan hasil $1+1$

- Multiplication and Division (perkalian dan pembagian) : memiliki prioritas yang sama tetapi lebih tinggi dari penjumlahan dan pengurangan.
- Operators : operator yang memiliki prioritas yang sama dibaca dari kiri ke kanan.

Jika terjadi keraguan, maka letakkan tanda kurung di dalam ekspresi sehingga memastikan komposisi sesuai dengan yang diinginkan.

OPERATOR MODULUS DAN STRING

Modulus

Yaitu sisa hasil bagi dari 2 bilangan. Operator ini hanya berlaku untuk tipe integer dan dalam python dilambangkan dengan tanda persen (%). Biasanya digunakan untuk memeriksa apakah satu angka dapat dibagi dengan suatu angka lain maupun untuk mengekstrak digit paling kanan dari suatu angka.

```
>>> 9%2
1
```

Gambar 1.10 : Penggunaan operator modulus

String

Operator + jika digunakan dalam string maka artinya adalah penggabungan antar string.

```
>>> satu = 10
>>> dua = 15
>>> print (satu + dua)
25
>>> satu = "10"
>>> dua = "15"
>>> print (satu + dua)
1015
```

Gambar 1.11 : Penggunaan operator + dalam tipe data string

Operator * berfungsi untuk melakukan perkalian antar konten string dan integer

```
>>> satu = 'Halo'
>>> dua = 5
>>> print (satu*dua)
HaloHaloHaloHaloHalo
```

Gambar 1.12 : Penggunaan operator * dalam tipe data string

MENANGANI INPUT DARI PENGGUNA

Program memiliki alur kerja Input – Proses – Output. Input adalah data yang dimasukkan supaya program bisa berjalan. Proses adalah langkah yang dilakukan program dalam memecahkan masalah. Sedangkan Output merupakan hasil yang didapat.

Python dapat menangani input dari pengguna (dalam konteks teks yang dimasukan pengguna) dengan keyboard dalam program yang disebut input. Ketika fungsi dipanggil, program akan berhenti menunggu pengguna mengetik sesuatu. Ketika enter ditekan, program akan berlanjut dan input akan mengembalikan apa yang diketik oleh pengguna sebagai string.

Sebelum mendapatkan input pengguna, ada baiknya untuk mencetak prompt atau pengantar yang memberitahu pengguna apa yang harus mereka input. String itu dapat diteruskan ke input untuk ditampilkan kepada pengguna sebelum berhenti untuk melakukan input.

```
>>> nama = input('siapa nama teman disebelahmu? ')
siapa nama teman disebelahmu? Cheila Zefanya W.
>>> print (nama)
Cheila Zefanya W.
```

Gambar 1.13 : Penggunaan fungsi input dalam Python

Ketika mengharapkan pengguna mengetik bilangan bulat, dapat dilakukan dengan mengkonversi nilai kembali ke int menggunakan fungsi int():

```
Berapa porsi makanmu? 3
>>> porsiMakan = int(input('Berapa porsi makanmu? '))
Berapa porsi makanmu? yang kapan?
Traceback (most recent call last):
  File "<python-input-8>", line 1, in <module>
    porsiMakan = int(input('Berapa porsi makanmu? '))
ValueError: invalid literal for int() with base 10: 'yang kapan?'
```

Gambar 1.14 : Penggunaan integer pada input.

Akan terjadi error jika pengguna menginput selain angka.

KOMENTAR

Tanda pagar (#) digunakan untuk memberi komentar pada python. Komentar tidak akan diproses oleh interpreter Python dan hanya digunakan untuk memudahkan programmer dalam memahami maksud kode.

```
1 #ini komentar 1
2 print('Halo semua, ini test') #kasi komen dulu
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\prak_alpro\praktikum_alpro - 71251189\Praktikum-Alpro--
thon.exe" "c:/prak_alpro/praktikum_alpro - 71251189/Praktikum
Halo semua, ini test

Gambar 1.15 : Penggunaan pagar sebagai komentar dalam python

Sumber : Modul 02–Variable Expression dan Statement ([02-Variable Expression, dan Statement](#))

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.

Link Github : https://github.com/alc357/71251189_alicia.git

SOAL 1

Source Code :

```
#tinggi badan harus dalam meter
```

```
tinggi_badan = float (input("Berapa tinggi badanmu dalam meter? "))
```

```
#BMI harus dalam kg/m^2
```

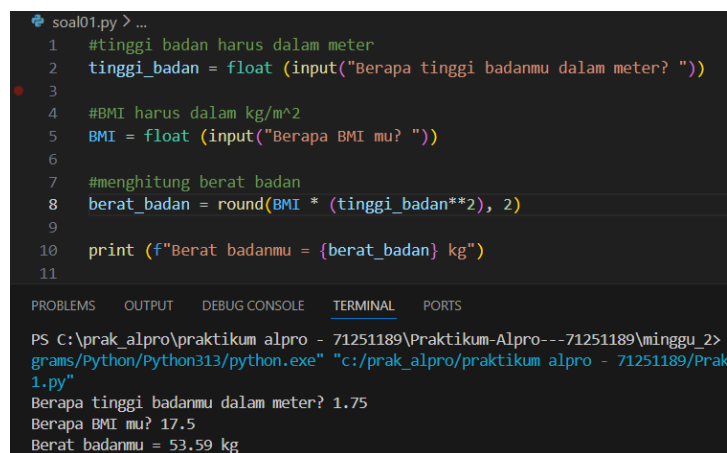
```
BMI = float (input("Berapa BMI mu? "))
```

```
#menghitung berat badan
```

```
berat_badan = round(BMI * (tinggi_badan**2), 2)
```

```
print (f"Berat badanmu = {berat_badan} kg")
```

Output :



```
soal01.py > ...
1  #tinggi badan harus dalam meter
2  tinggi_badan = float (input("Berapa tinggi badanmu dalam meter? "))
3
4  #BMI harus dalam kg/m^2
5  BMI = float (input("Berapa BMI mu? "))
6
7  #menghitung berat badan
8  berat_badan = round(BMI * (tinggi_badan**2), 2)
9
10 print (f"Berat badanmu = {berat_badan} kg")
11
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\prak_alpro\praktikum alpro - 71251189\Praktikum-Alpro---71251189\minggu_2>
grams/Python/Python313/python.exe" "c:/prak_alpro/praktikum alpro - 71251189/Prak
1.py"
Berapa tinggi badanmu dalam meter? 1.75
Berapa BMI mu? 17.5
Berat badanmu = 53.59 kg
```

Gambar 2.1 : Kode program soal no.1 menggunakan Bahasa python pada Vs Code

Langkah – langkah :

1. Langkah pertama, buka aplikasi yang ingin kamu gunakan untuk menjalankan python. Salah satunya dengan menggunakan aplikasi visual studio code. Buat file baru dengan format judul.py;
2. Setelah python sudah dapat dijalankan, kita akan mulai membuat kode pemrograman. Pertama-tama buatlah variabel baru yang fungsinya untuk menampung data tinggi badan yang diinput oleh pengguna. Saya menggunakan variabel dengan nama tinggi_badan dan karena tinggi badan yang diinput oleh pengguna harus dalam satuan meter maka jenis data yang digunakan adalah float. Float memungkinkan pengguna menginput angka pecahan dibanding dengan integer yang hanya dapat menginput angka berupa bilangan bulat;
3. Kemudian, buat variabel kedua yang berfungsi untuk menampung data BMI yang berasal dari input pengguna. Saya menggunakan variabel BMI. Karena angka BMI tidak selalu bilangan bulat maka saya menggunakan jenis data float;
4. Selanjutnya, buatlah variabel baru yang berisi expression untuk menghitung berat badan. Expression berisi perhitungan BMI dikalikan tinggi badan pangkat dua. Penulisan biasanya dapat dituliskan berat_badan = BMI x tinggi_badan. Sedangkan dalam python dapat ditulis berat_badan = round(BMI * (tinggi_badan**2), 2). Statement round berfungsi untuk membulatkan bilangan. Penulisannya yaitu round(expression, banyak digit dibelakang koma)
5. Untuk memunculkan hasil perhitungan, kita dapat menggunakan statement print dengan variabel berat_badan. Penulisannya yaitu print (f"Berat badanmu = {berat_badan} kg"). Huruf f berfungsi untuk penformatan supaya kita dapat memasukkan variabel yang ingin kita print yaitu berat_badan dengan menggunakan {} di antara kalimat yang ingin kita print.
6. Output dapat dilihat pada gambar 2.1 . Contohnya Ketika kita menginput tinggi badan 1.75 dan BMI 17.5 hasil perhitungan berat badan adalah 53.59

SOAL 2

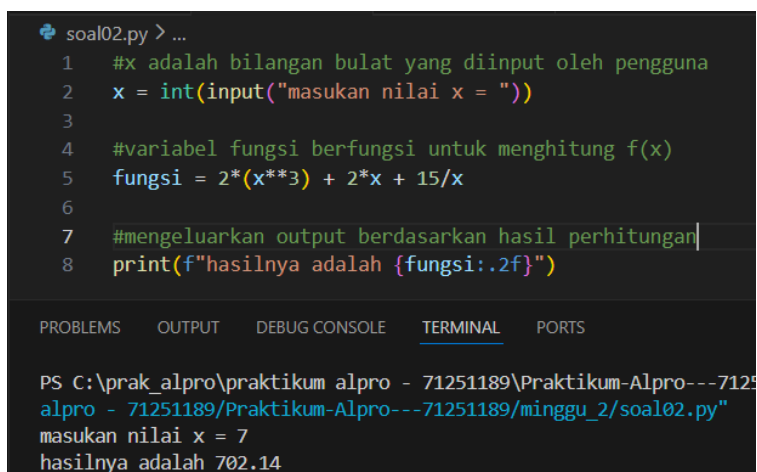
Source Code :

```
# x adalah bilangan bulat yang diinput oleh pengguna
x = int(input("masukan nilai x = "))

# variabel fungsi berfungsi untuk menghitung f(x)
fungsi = 2*(x**3) + 2*x + 15/x

# mengeluarkan output berdasarkan hasil perhitungan
print(f"hasilnya adalah {fungsi:.2f}")
```

Output :



The screenshot shows the Visual Studio Code editor with a file named `soal02.py`. The code is as follows:

```
1 #x adalah bilangan bulat yang diinput oleh pengguna
2 x = int(input("masukan nilai x = "))
3
4 #variabel fungsi berfungsi untuk menghitung f(x)
5 fungsi = 2*(x**3) + 2*x + 15/x
6
7 #mengeluarkan output berdasarkan hasil perhitungan
8 print(f"hasilnya adalah {fungsi:.2f}")
```

Below the code editor, the **TERMINAL** tab is active, showing the command prompt output:

```
PS C:\prak_alpro\praktikum_alpro - 71251189\Praktikum-Alpro---71251189\minggu_2\soal02.py
masukan nilai x = 7
hasilnya adalah 702.14
```

Gambar 2.2 : Kode program soal nomor 2 pada Visual Studio Code

Langkah – langkah :

1. Langkah pertama, buka aplikasi yang ingin kamu gunakan untuk menjalankan python. Salah satunya dengan menggunakan aplikasi visual studio code. Buat file baru dengan format judul.py;
2. Setelah python sudah dapat dijalankan, kita akan mulai membuat kode pemrograman. Pertama-tama buatlah variabel baru yang fungsinya untuk menampung data bilangan buat

yang akan diinput oleh pengguna. Saya menggunakan variabel dengan nama x dan karena bilangan yang diinput oleh pengguna harus berupa bilangan bulat maka jenis data yang digunakan adalah integer.

3. Selanjutnya buat variabel yang berisi expression untuk menghitung fungsi $2x^3 + 2x + \frac{15}{x}$. Variabel yang saya buat adalah fungsi dengan penulisan yaitu `2*(x**3) + 2*x + 15/x`. Tidak seperti penulisan biasa dimana $2x$ secara otomatis kita pahami sebagai 2 dikali x, interpreter tidak dapat memahami. Maka dari itu, kita harus memberikan tanda kali antara 2 dan x. Selain itu, supaya urutannya benar maka `x**3` harus dikurung supaya x dipangkatkan 3 terlebih dahulu baru dikalikan dengan 2 bukan `2*x` baru dipangkatkan 3.
4. Untuk memunculkan hasil perhitungan, kita dapat menggunakan statement print dengan variabel fungsi. Penulisananya yaitu `print(f"hasilnya adalah {fungsi:.2f}")`. Huruf f berfungsi untuk penformatan supaya kita dapat memasukan variabel yang ingin kita print yaitu fungsi dengan menggunakan `{}` diantara kalimat yang ingin kita print. Sedangkan `:2f` berfungsi untuk membuat hasil perhitungan variabel fungsi menjadi 2 angka di belakang koma saja.
5. Output dapat dilihat pada gambar 2.2. Contohnya Ketika kita menginput `x = 7` maka hasil perhitungan fungsi adalah 702.14.

SOAL 3

Source Code :

```
# membuat variabel untuk menyimpan input pengguna tentang gaji perjam dan jam kerja
gaji_per_jam = int(input("Berapa Gaji yang diinginkan Budi tiap jamnya? "))
jumlah_jam_kerja = int(input("Berapa jam kerja yang diinginkan Budi tiap minggu? "))
jumlah_minggu = 5

# menghitung total gaji Budi selama 5 minggu bekerja dan setelah dipotong pajak
total_gaji = round((gaji_per_jam * jumlah_jam_kerja*jumlah_minggu),2)
setelah_potong_pajak = round(total_gaji - (total_gaji * (14/100)), 2)
```

```
# menghitung uang belanja dan alat tulis
uang_belanja = round(setelah_potong_pajak * (10/100), 2)
uang_alat_tulis = round(setelah_potong_pajak * (1/100), 2)
total_belanja = uang_belanja + uang_alat_tulis

# menghitung uang setelah belanja
uang_setelah_belanja = setelah_potong_pajak - total_belanja

# menghitung uang sedekah (diterima anak yatim dan kaum dhuafa)
uang_sedekah = round(uang_setelah_belanja * (25/100), 2)
untuk_yatim = round(uang_sedekah * (30/100), 2)
untuk_kaum_dhuafa = round((uang_sedekah-untuk_yatim), 2)

# memberikan output
print(f"Pendapatan Budi selama libur musim panas sebelum melakukan pembayaran pajak
      adalah Rp {total_gaji}")
print(f"Pendapatan Budi selama libur musim panas setelah melakukan pembayaran pajak adalah
      Rp {setelah_potong_pajak}")
print(f"Total uang yang Budi habiskan untuk membeli pakaian dan aksesoris adalah Rp
      {uang_belanja}")
print(f"Total uang yang Budi habiskan untuk membeli alat tulis adalah Rp {uang_alat_tulis}")
print(f"Uang yang Budi sedekahkan adalah Rp {uang_sedekah}")
print(f"Uang sedekah Budi yang diterima anak yatim adalah Rp {untuk_yatim}")
print(f"Uang sedekah Budi yang diterima kaum dhuafa adalah Rp {untuk_kaum_dhuafa}")
```

Kode & Output :

```
soal03.py > ...
1 # membuat variabel untuk menyimpan input pengguna tentang gaji perjam dan jam kerja
2 gaji_per_jam = int(input("Berapa Gaji yang diinginkan Budi tiap jamnya? "))
3 jumlah_jam_kerja = int(input("Berapa jam kerja yang diinginkan Budi tiap minggu? "))
4 jumlah_minggu = 5
5
6 # menghitung total gaji Budi selama 5 minggu bekerja dan setelah dipotong pajak
7 total_gaji = round((gaji_per_jam * jumlah_jam_kerja * jumlah_minggu), 2)
8 setelah_potong_pajak = round(total_gaji - (total_gaji * (14/100)), 2)
9
10 # menghitung uang belanja dan alat tulis
11 uang_belanja = round(setelah_potong_pajak * (10/100), 2)
12 uang_alat_tulis = round(setelah_potong_pajak * (1/100), 2)
13 total_belanja = uang_belanja + uang_alat_tulis
14
15 # menghitung uang setelah belanja
16 uang_setelah_belanja = setelah_potong_pajak - total_belanja
17
18 # menghitung uang sedekah (diterima anak yatim dan kaum dhuafa)
19 uang_sedekah = round(uang_setelah_belanja * (25/100), 2)
20 untuk_yatim = round(uang_sedekah * (30/100), 2)
21 untuk_kaum_dhuafa = round((uang_sedekah - untuk_yatim), 2)
22
23 # memberikan output
24 print(f"Pendapatan Budi selama libur musim panas sebelum melakukan pembayaran pajak adalah Rp {total_gaji}")
25 print(f"Pendapatan Budi selama libur musim panas setelah melakukan pembayaran pajak adalah Rp {setelah_potong_pajak}")
26 print(f"Total uang yang Budi habiskan untuk membeli pakaian dan aksesoris adalah Rp {uang_belanja}")
27 print(f"Total uang yang Budi habiskan untuk membeli alat tulis adalah Rp {uang_alat_tulis}")
28 print(f"Uang yang Budi sedekahkan adalah Rp {uang_sedekah}")
29 print(f"Uang sedekah Budi yang diterima anak yatim adalah Rp {untuk_yatim}")
30 print(f"Uang sedekah Budi yang diterima kaum dhuafa adalah Rp {untuk_kaum_dhuafa}")
```

Gambar 2.3 : Kode pemrograman soal nomor 3 pada Visual Studio Code

```
PS C:\prak_alpro\praktikum alpro - 71251189\Praktikum-Alpro---71251189\minggu_2> & "C:/Users/alc 3
alpro - 71251189/Praktikum-Alpro---71251189\minggu_2/soal03.py"
Berapa Gaji yang diinginkan Budi tiap jamnya? 30000
Berapa jam kerja yang diinginkan Budi tiap minggu? 56
Pendapatan Budi selama libur musim panas sebelum melakukan pembayaran pajak adalah Rp 8400000
Pendapatan Budi selama libur musim panas setelah melakukan pembayaran pajak adalah Rp 7224000.0
Total uang yang Budi habiskan untuk membeli pakaian dan aksesoris adalah Rp 722400.0
Total uang yang Budi habiskan untuk membeli alat tulis adalah Rp 72240.0
Uang yang Budi sedekahkan adalah Rp 1607340.0
Uang sedekah Budi yang diterima anak yatim adalah Rp 482202.0
Uang sedekah Budi yang diterima kaum dhuafa adalah Rp 1125138.0
PS C:\prak_alpro\praktikum alpro - 71251189\Praktikum-Alpro---71251189\minggu_2> |
```

Gambar 2.4 : Output pemrograman soal nomor 4 pada Visual Studio Code

Langkah – langkah :

1. Langkah pertama, buka aplikasi yang ingin kamu gunakan untuk menjalankan python. Salah satunya dengan menggunakan aplikasi visual studio code. Buat file baru dengan format judul.py

2. Langkah pertama, buka aplikasi yang ingin kamu gunakan untuk menjalankan python. Salah satunya dengan menggunakan aplikasi visual studio code. Buat file baru dengan format judul.py;
3. Ketika python sudah dapat dijalankan, kita akan membuat kode pemrograman. Pertama-tama kita buat variabel-variabel yang berfungsi untuk menyimpan data-data yang diperlukan berdasarkan inputan dari pengguna. Pada soal ini, data yang diperlukan yaitu gaji per jam yang diinginkan Budi (`gaji_per_jam`), jumlah jam kerja yang diinginkan Budi dalam seminggu (`jumlah_jam_kerja`), dan kita buat juga variabel yang berisi jumlah minggu Budi bekerja (`jumlah_minggu`) dimana dikatakan di soal Budi bekerja selama 5 minggu maka langsung kita tulis `jumlah_minggu = 5`.
4. Selanjutnya, kita hitung jumlah gaji yang didapatkan Budi dengan membuat variabel yang berisi expression gaji per jam yang diinginkan Budi dikali dengan jumlah jam kerja yang diinginkan Budi kemudian dikali lagi dengan jumlah minggu yaitu 5. Penulisan dalam program yaitu `total_gaji = round((gaji_per_jam * jumlah_jam_kerja * jumlah_minggu), 2)`.
5. Setelah memperoleh jumlah gaji, kita hitung jumlah uang yang dimiliki Budi setelah dipotong pajak dengan mengurangi jumlah gaji dengan jumlah gaji dikali dengan 14 % . Penulisannya yaitu `setelah_potong_pajak = round(total_gaji - (total_gaji * (14/100)), 2)`
6. Kemudian, kita hitung uang yang dikeluarkan Budi untuk membeli baju dan aksesoris dengan mengalikan uang setelah dipotong pajak dengan 20% penulisannya yaitu `uang_belanja = round(setelah_potong_pajak * (10/100), 2)`
7. Kita hitung juga uang yang dikeluarkan Budi untuk membeli alat tulis dengan mengalikan uang setelah dipotong pajak dengan 1% penulisannya yaitu `uang_alat_tulis = round(setelah_potong_pajak * (1/100), 2)`
8. Kemudian kita hitung total uang yang digunakan Budi untuk belanja dengan menjumlahkan uang belanja dengan uang alat tulis. Penulisannya yaitu `total_belanja = uang_belanja + uang_alat_tulis`.
9. Kita cari uang Budi setelah Budi setelah belanja dengan mengurangi uang setelah dipotong pajak dengan total belanja. Penulisannya yaitu `uang_setelah_belanja = setelah_potong_pajak - total_belanja`

10. Selanjutnya, kita cari jumlah yang digunakan Budi untuk melakukan sedekah dengan mengalikan uang setelah belanja dengan 25%. Penulisannya yaitu $\text{uang_sedekah} = \text{round}(\text{uang_setelah_belanja} * (25/100), 2)$
11. Kemudian, kita akan mencari jumlah yang akan diterima oleh anak yatim dari uang sedekah yang diberikan Budi dengan mengalikan uang sedekah dengan 30%. Penulisannya adalah $\text{untuk_yatim} = \text{round}(\text{uang_sedekah} * (30/100), 2)$. Sedangkan uang untuk kaum dhuafa dapat dihitung dengan mengurangi uang sedekah dengan uang yang diberikan kepada anak yatim. Penulisannya adalah $\text{untuk_kaum_dhuafa} = \text{round}((\text{uang_sedekah} - \text{untuk_yatim}), 2)$.
12. Penggunaan round pada setiap expression digunakan supaya bilangan yang dihasilkan dapat dibulatkan menjadi 2 angka di belakang koma.
13. Kemudian print semua hasil yang ingin ditampilkan.