

Modul 2: Sintaks Dasar Bahasa Pemrograman Python

2.1 Sekilas Pandang tentang Pemrograman Python

Python adalah salah satu bahasa pemrograman yang dirancang untuk memudahkan manusia dalam memberikan instruksi kepada komputer. Jika kita ibaratkan komputer sebagai seorang pekerja yang sangat kuat tetapi hanya dapat memahami bahasa tertentu, maka Python adalah salah satu cara kita untuk "berbicara" kepada komputer tersebut. Namun, yang membedakan Python dari bahasa-bahasa lain adalah kemudahannya untuk dipelajari dan digunakan, bahkan oleh seseorang yang sama sekali tidak memiliki pengalaman sebelumnya dalam dunia komputer.

Salah satu hal utama yang membuat Python begitu populer adalah sintaksisnya yang mirip dengan bahasa manusia. Jika dibandingkan dengan bahasa pemrograman lain seperti C++ atau Java, Python menggunakan lebih sedikit simbol dan struktur kompleks. Misalnya, untuk mencetak teks ke layar, cukup menulis `print("Halo Dunia!")`. Bandingkan ini dengan bahasa lain yang mungkin memerlukan beberapa baris kode untuk melakukan hal yang sama. Kemudahan ini menjadikan Python sebagai pilihan utama bagi pemula, termasuk anak-anak, pelajar, hingga para profesional di berbagai bidang.

Python juga dikenal sebagai bahasa pemrograman "serbaguna" (*general-purpose*), yang artinya dapat digunakan di berbagai bidang dan aplikasi. Di dunia ilmu data, Python digunakan untuk menganalisis data, membuat grafik, dan membangun model prediksi menggunakan pustaka seperti *Pandas*, *Matplotlib*, dan *Scikit-learn*. Di bidang pengembangan web, Python digunakan untuk membangun situs web dengan kerangka kerja seperti Django dan Flask. Bahkan, dalam kecerdasan buatan dan pembelajaran mesin, Python adalah bahasa utama karena mendukung pustaka-pustaka canggih seperti *TensorFlow* dan *PyTorch*.

Dalam dunia pendidikan, Python telah menjadi bahasa pilihan untuk mengajarkan pemrograman kepada pemula. Karena sintaksisnya yang sederhana dan kemampuannya yang luas, Python memungkinkan siswa untuk fokus pada konsep-konsep inti pemrograman tanpa harus terjebak dalam detail teknis yang rumit. Bahkan, banyak universitas dan lembaga pendidikan kini menjadikan Python sebagai bahasa utama dalam kurikulum mereka. Modul ini bertujuan untuk memperkenalkan sintaks dasar Python, mencakup konsep kunci yang membentuk fondasi untuk studi pemrograman lebih lanjut.

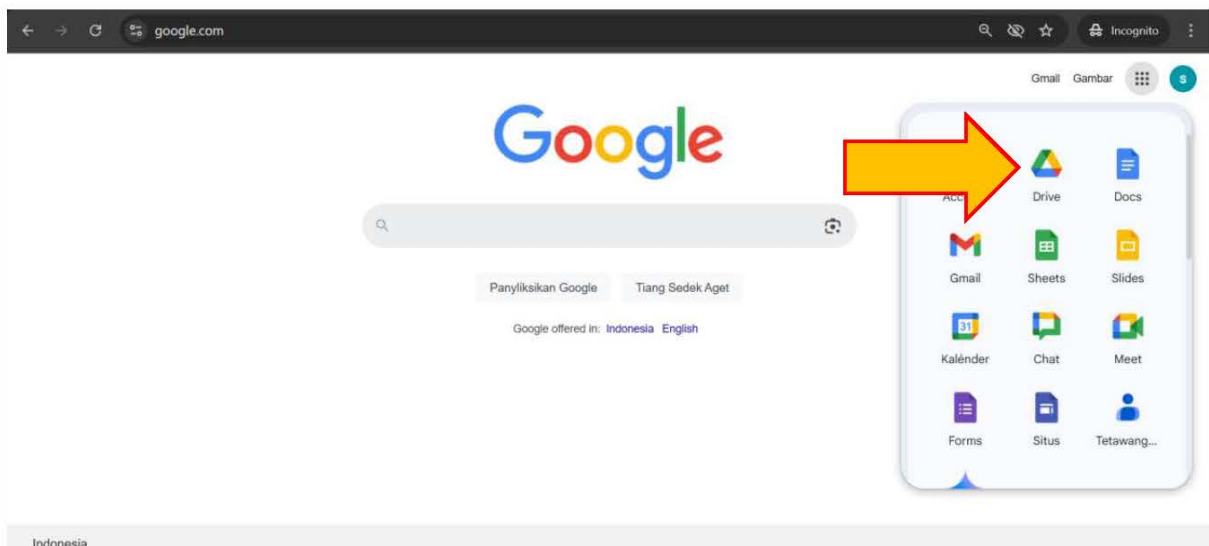
2.2 Mempersiapkan Lingkungan Kompilasi Python

Berikut adalah langkah-langkah untuk mempersiapkan kompilasi Python menggunakan Google Drive yang dihubungkan dengan Google Collaboratory. Panduan ini dirancang untuk orang yang baru mengenal komputer sekalipun, dengan penjelasan runtut serta gambar untuk membantu memahami setiap tahap.

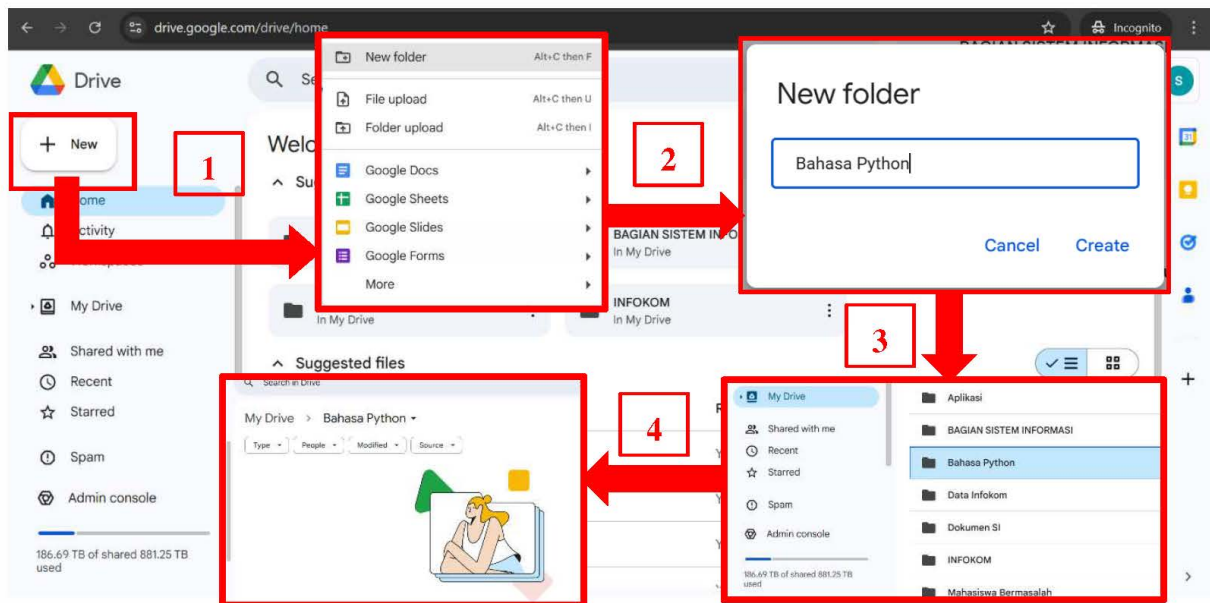
- Login ke akun Google Anda menggunakan email akun email UMM dan webmail UMM dan kata sandi (PIC UMM).



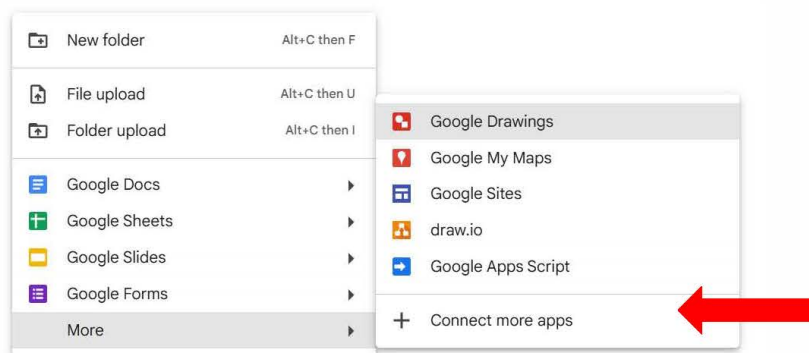
- Jika benar maka akan diarahkan ke halaman dashboard Google, klik icon Google Drive



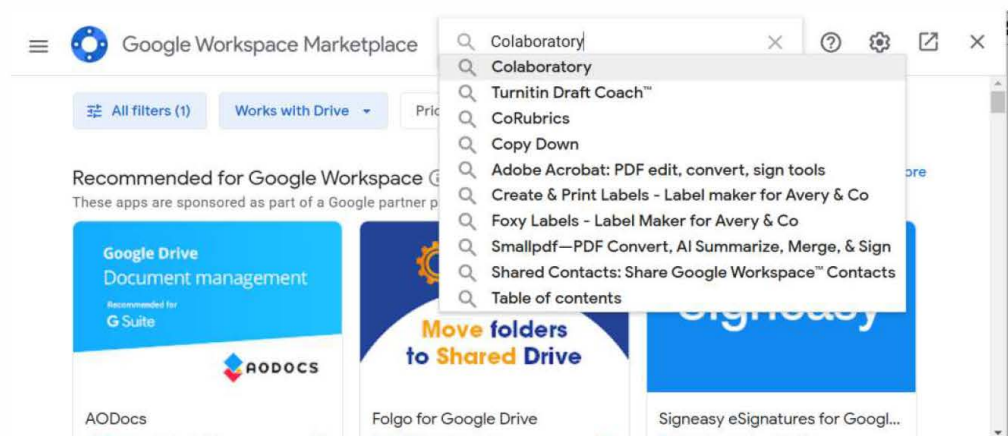
- Setelah masuk, Anda akan melihat ruang penyimpanan untuk menyimpan file Anda. Pilih **New Folder** untuk mengatur file Anda. Ikut step berikut :



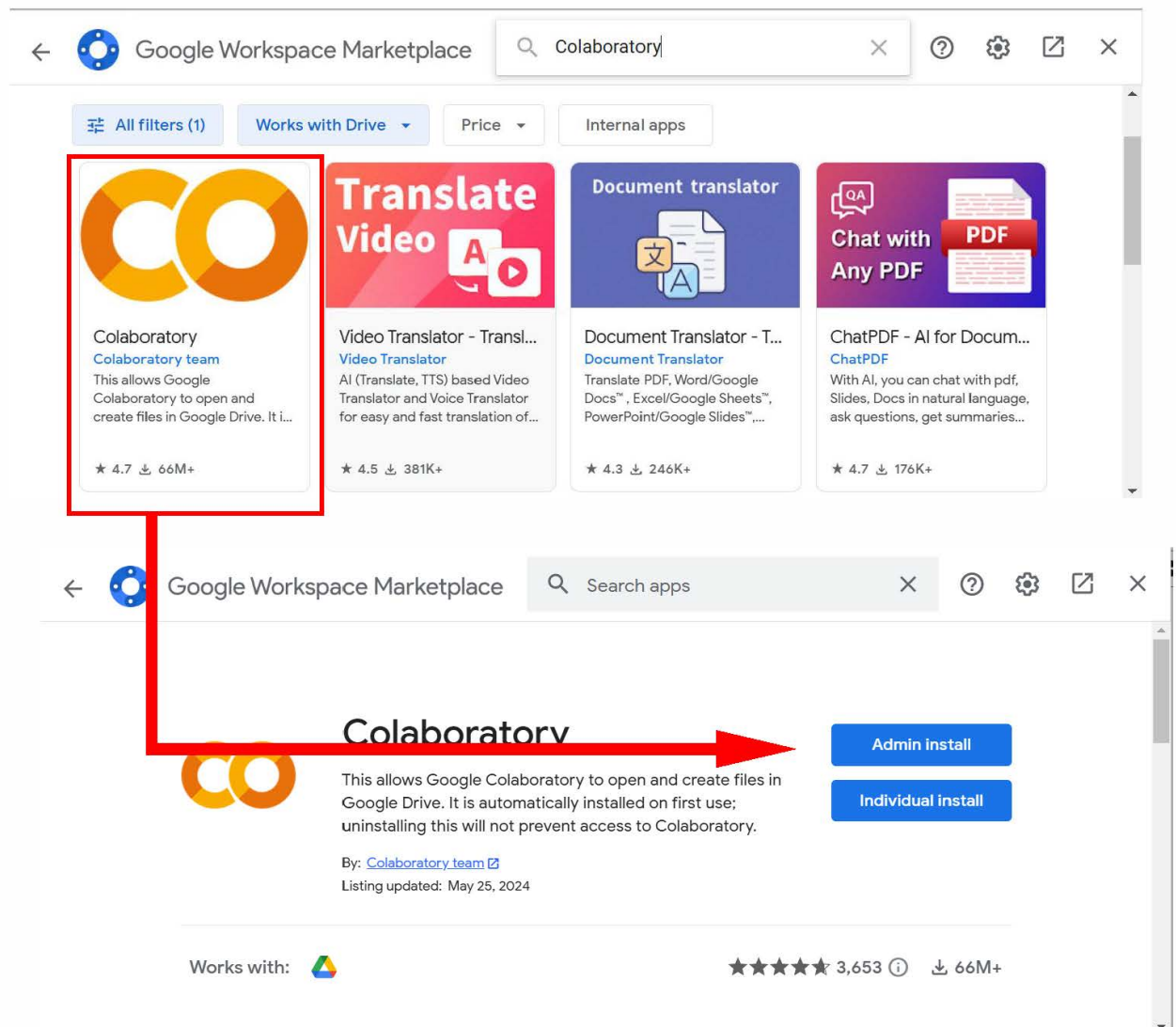
- Di bagian **folder Bahasa Python** bisa di klik kanan, kemudian **+ Connect more apps**



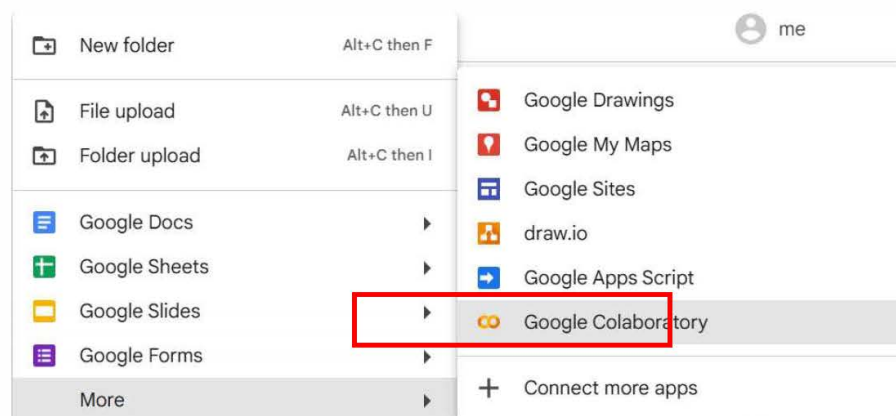
- Di bagian **folder Bahasa Python** bisa di klik kanan, kemudian pilih **+ Connect more apps**, lalu di kolom *search* silahkan masukan keyword '**collaboratory**'




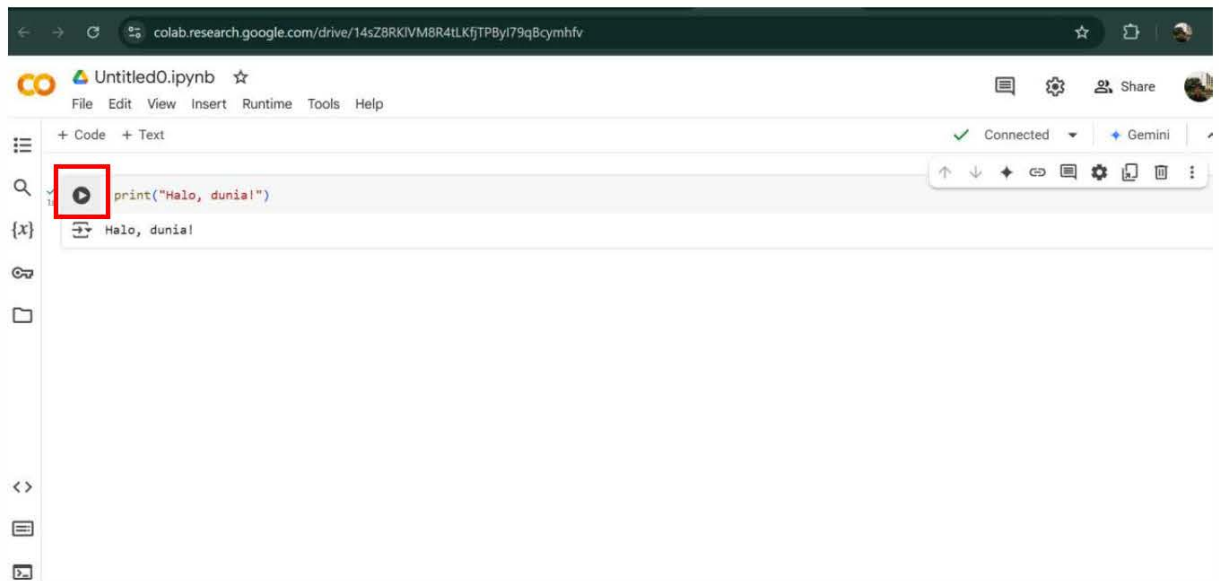
- Pilih aplikasi Collaboratory yang sudah diberi border warna merah silahkan di klik bagian tersebut lalu pilih tombol **Admin Install**.



- Jika langkah tersebut sudah dilakukan, silakan kembali ke **folder Bahasa Python** kemudian ulangi lagi dengan klik kanan lalu pilih **Google Collaboratory**, silahkan dipilih menu itu lalu klik dibagian tersebut.



- Akhirnya kita sudah masuk **workspace Google Colaboratory**, di sini kita bisa running sintaks yang sudah ditulis menggunakan sintaks python. Sebagai contoh Tulis kode Python Anda di sel kosong. Lalu ketika perintah berikut `print("Halo, dunia!")`  di sebelah kiri atas untuk menjalankan kode. Hasilnya akan muncul tepat di bawah sel kode.



2.3 Sintaks Dasar Python

a. Sintaks Python

Sintaks Python mengacu pada seperangkat aturan yang menentukan bagaimana program Python ditulis dan dipahami. Di bawah ini adalah contoh sederhana dari program Python:

```
# Python Code
# Ini adalah program Python sederhana yang mencetak pesan sapaan.
print("Halo, Dunia")
```

Poin Penting

- Komentar dimulai dengan # dan diabaikan oleh interpreter Python.
- Fungsi `print()` mengeluarkan teks ke konsol.

b. Variable

Dalam kehidupan sehari-hari, kita sering menyimpan sesuatu di dalam wadah, seperti kotak untuk menyimpan mainan atau kantong untuk menyimpan uang. Di dalam Pemrograman Python, kita bisa menggunakan **variabel** sebagai "wadah" untuk menyimpan data. Contoh nya sebagai berikut :

```
# Menyatakan variable
nama = "Alice" # Variabel string
usia = 21      # Variabel integer
tinggi = 5.6   # Variabel float
```

Poin Kunci:

- Nama variabel harus deskriptif dan mengikuti konvensi penamaan (misalnya, `tinggi_badan`), serta tidak boleh ada spasi
- Python bersifat dinamis terketik, yang berarti Anda tidak perlu menyatakan tipe secara eksplisit.

c. Tipe Data

Python memiliki beberapa tipe data bawaan, termasuk:

- **Integer:** Angka bulat (misalnya, 5, -3)
- **Float:** Angka desimal (misalnya, 3.14, -0.001)
- **String:** Teks (misalnya, "Halo", 'Python')
- **Boolean:** Nilai True atau False (True, False)

Contoh penggunaan berbagai tipe data:

```
# Contoh tipe data yang berbeda
nilai_integer = 10
nilai_float = 10.5
nilai_string = "Halo, Python!"
nilai_boolean = True
```

d. Operator

Operator digunakan untuk melakukan operasi pada variabel dan nilai. Jenis operator yang paling umum di Python adalah:

- **Operator Aritmatika:** +, -, *, /, % (modulus)

- **Operator Perbandingan:** `==`, `!=`, `<`, `>`, `<=`, `>=`
- **Operator Logika:** `and`, `or`, `not`

Contoh Operator Aritmatika:

```
# Melakukan operasi aritmatika
a = 10
b = 5
penjumlahan = a + b # Penjumlahan
selisih = a - b      # Pengurangan
perkalian = a * b    # Perkalian
pembagian = a / b    # Pembagian
modulus = a % b      # Modulus (siswa hasil bagi)
```

e. Input dan Output

Python menyediakan fungsi bawaan untuk operasi input dan output. Fungsi `input()` digunakan untuk mengambil input dari pengguna, sementara fungsi `print()` mengeluarkan data ke konsol.

Contoh Input dan Output:

```
# Mengambil input dari pengguna dan menampilkannya
nama_pengguna = input("Masukkan nama Anda: ") # Input dari pengguna
print("Halo, " + nama_pengguna + "!")          # Output ke konsol
```

2.4 Latihan Sintaks Dasar Python

Berikut beberapa latihan untuk mempraktikkan konsep yang dibahas dalam modul 2 ini. Setiap latihan diikuti dengan solusinya dan penjelasan untuk mempermudah pemahaman peserta pelatihan.

Latihan 1: Deklarasi Variabel, Input, Output, dan Aritmatika Dasar (Sistem ATM)

Tugas: Seseorang memiliki saldo awal di rekening sebesar Rp100.000. Pada suatu waktu, orang tersebut melakukan setoran tambahan sebesar Rp200.000 ke rekeningnya. Buat program untuk menghitung jumlah saldo orang tersebut setelah ditambah setoran dengan jumlah tertentu.

Solusi:

```
# Menginisialisasi saldo akun
saldo_akun = 100000 # Saldo awal

# inputan dari user untuk jumlah setoran
jumlah_setoran = int(input("Masukkan jumlah setoran: "))

# Operasi setoran
saldo_akun = saldo_akun + jumlah_setoran # Perbarui saldo
print("Saldo baru Anda adalah:", saldo_akun) # Cetak hasil
```

Penjelasan:

Program ini bertujuan untuk menghitung saldo baru pada akun setelah dilakukan setoran. Pertama-tama kita mendeklarasikan variabel untuk jumlah saldo awal: `saldo_akun`. Variabel `saldo_akun` menyimpan nilai saldo awal akun yang dimiliki, dalam hal ini sebesar 100.000. Sedangkan variabel `jumlah_setoran` akan menyimpan nilai uang yang akan disetor ke dalam akun, yang didapatkan dari masukan pengguna.

Setelah itu, kita melakukan operasi aritmatika untuk menghitung saldo baru. Dalam hal ini, kita menambahkan `saldo_akun` dengan `jumlah_setoran`, yang berarti saldo yang ada di akun akan bertambah sebesar jumlah setoran. Hasil penjumlahan ini kemudian disimpan kembali ke dalam variabel `saldo_akun`, yang berarti saldo akun telah diperbarui dengan jumlah uang yang baru. Terakhir, program mencetak hasilnya ke layar menggunakan perintah `print()`. Fungsi `print()` ini akan menampilkan kalimat "Saldo baru Anda adalah:" diikuti dengan nilai saldo akun yang telah diperbarui.

Secara keseluruhan, program ini adalah contoh sederhana yang menunjukkan bagaimana kita dapat memanipulasi data di dalam program (dalam hal ini saldo akun) dan menampilkan hasilnya kepada pengguna dengan menggunakan konsep dasar seperti variabel, operasi penjumlahan, input dengan fungsi `input()`, dan output dengan fungsi `print()`. Program ini mudah dipahami karena menghubungkan logika pemrograman dengan situasi dunia nyata yang familiar, seperti mengelola saldo dalam rekening.

Latihan 2: Deklarasi Variabel, Input, Output, dan Aritmatika Dasar (Transaksi Peminjaman Buku)

Tugas: Di sebuah perpustakaan, terdapat sebuah buku berjudul "*Pemrograman Python untuk Pemula*". Buku ini memiliki jumlah stok sebanyak 10 buku. Pada suatu hari, ada 3 buku yang dipinjam oleh mahasiswa. Buat informasi berapa stok buku yang tersedia, jumlah buku yang dipinjam oleh mahasiswa dan sisa stok.

Solusi:

```
# Inisialisasi variabel
judul_buku = "Pemrograman Python untuk Pemula"
stok_buku = 10

#mendapatkan jumlah buku yang dipinjam dari input pengguna
jml_pinjam = int(input("Berapa jumlah buku yang sedang dipinjam: "))

# Perhitungan sisa stok
sisa_stok = stok_buku - jml_pinjam

# Output hasil
print("Judul Buku:", judul_buku)
print("Stok Awal:", stok_buku)
print("Jumlah Buku Dipinjam:", jml_pinjam)
print("Sisa Stok Buku:", sisa_stok)
```

Penjelasan :

Program Python ini adalah contoh sederhana untuk menghitung sisa stok buku di perpustakaan setelah beberapa buku dipinjam. Program dimulai dengan mendeklarasikan tiga variabel utama: `judul_buku`, `stok_buku`, dan `jml_pinjam`. Variabel `judul_buku` berisi teks yang menunjukkan judul buku yang ada di perpustakaan, yaitu "*Pemrograman Python untuk Pemula*". Variabel `stok_buku` menyimpan jumlah total buku yang tersedia di perpustakaan, yaitu 10 buku. Sedangkan `jml_pinjam` menyimpan informasi tentang berapa banyak buku yang telah dipinjam, yang dalam hal ini didapatkan dari masukan pengguna. Selanjutnya, program menghitung sisa stok buku yang tersedia setelah peminjaman. Hal ini dilakukan dengan mengurangi nilai `stok_buku` (10) dengan

nilai dari variabel `jml_pinjam` yang nilainya didapat dari masukan pengguna. Hasilnya akan disimpan di variabel `sisastok`.

Terakhir, program menampilkan informasi yang relevan ke layar menggunakan fungsi `print()`. Fungsi ini digunakan untuk menampilkan judul buku, stok buku yang tersedia, jumlah buku yang dipinjam, dan sisa stok buku yang ada. Secara keseluruhan, program ini menunjukkan bagaimana cara sederhana untuk melakukan perhitungan dan menampilkan hasilnya dalam konteks yang sangat mudah dipahami, seperti pengelolaan stok buku di perpustakaan. Program ini juga memperkenalkan konsep dasar pemrograman seperti penggunaan variabel, operasi aritmatika (pengurangan), dan fungsi `print()` untuk output.

Latihan 3: Deklarasi Variabel dan Aritmatika Dasar (Transaksi Pemesanan Kereta Api)

Tugas: Sebuah kereta api menawarkan tiket perjalanan dengan harga Rp 150.000 per orang. Dalam satu perjalanan, terdapat 3 penumpang yang membeli tiket. Buat program digunakan untuk menghitung total harga tiket yang harus dibayar oleh semua penumpang.

Solusi:

```
# Inisialisasi variabel
harga_tiket = 150000 # Harga tiket per orang
jumlah_penumpang = 3 # Jumlah penumpang
total_harga = harga_tiket * jumlah_penumpang # Total harga tiket

# Output hasil
print("Harga Tiket per Orang: Rp", harga_tiket)
print("Jumlah Penumpang:", jumlah_penumpang)
print("Total Harga Tiket: Rp", total_harga)
```

Penjelasan:

Kode tersebut adalah contoh sederhana dari sebuah program Bahasa Pemrograman Python yang digunakan untuk menghitung total biaya tiket perjalanan berdasarkan harga tiket per orang dan jumlah penumpang. Program ini dimulai dengan mendefinisikan dua variabel, yaitu `harga_tiket` dan `jumlah_penumpang`. Variabel `harga_tiket` menyimpan informasi tentang harga tiket untuk satu orang, yaitu Rp 150.000. Sementara itu, variabel `jumlah_penumpang` menyimpan jumlah total penumpang yang akan melakukan perjalanan, dalam hal ini sebanyak 3 orang.

Langkah selanjutnya adalah menghitung `total_harga` tiket dengan menggunakan operasi perkalian. Nilai `harga_tiket` dikalikan dengan `jumlah_penumpang` untuk

mendapatkan jumlah total yang harus dibayar oleh semua penumpang. Hasil dari perhitungan ini disimpan dalam variabel `total_harga`.

Setelah perhitungan selesai, program menampilkan hasilnya ke layar menggunakan fungsi `print()`. Fungsi ini akan mencetak tiga informasi utama: harga tiket per orang, jumlah penumpang, dan total harga tiket. Dengan menjalankan program ini, pengguna akan melihat informasi yang sudah diformat dengan jelas dan mudah dipahami.

Latihan 4: Deklarasi Variabel, Input, Output, dan Aritmatika Dasar (Perencanaan Anggaran Pribadi)

Tugas: Seseorang memiliki pendapatan bulanan sebesar Rp 5.000.000. Dalam satu bulan, individu tersebut mengeluarkan uang untuk berbagai keperluan:

- Rp 1.500.000 untuk makanan,
- Rp 2.000.000 untuk biaya sewa tempat tinggal,
- Rp 500.000 untuk transportasi.

Hitung berapa pendapatan, total pengeluaran dan sisa anggaran dalam satu bulan

Solusi:

```
# Inisialisasi variabel
pendapatan = float(input("Masukkan pendapatan bulanan Anda: "))
pengeluaran_makan = float(input("Masukkan pengeluaran makanan Anda: "))
pengeluaran_sewa = float(input("Masukkan pengeluaran sewa Anda: "))
pengeluaran_transport = float(input("Masukkan pengeluaran transport Anda: "))

# Perhitungan anggaran
total_pengeluaran = pengeluaran_makan + pengeluaran_sewa +
pengeluaran_transport
sisa_anggaran = pendapatan - total_pengeluaran

# Output hasil
print("Pendapatan Bulanan: Rp", pendapatan)
print("Total Pengeluaran: Rp", total_pengeluaran)
print("Sisa Anggaran: Rp", sisa_anggaran)
```

Penjelasan :

Source code tersebut merupakan contoh program sederhana menggunakan pemrograman Python yang dirancang untuk menghitung sisa anggaran bulanan seseorang

setelah memperhitungkan pendapatan dan berbagai pengeluaran. Program ini dimulai dengan mendefinisikan beberapa variabel yang mewakili data penting dalam perhitungan anggaran.

Pertama, variabel `pendapatan` digunakan untuk menyimpan jumlah pendapatan bulanan. Selanjutnya, tiga variabel lain, yaitu `pengeluaran_makan`, `pengeluaran_sewa`, dan `pengeluaran_transport`, masing-masing digunakan untuk mencatat pengeluaran pada kategori makanan, sewa tempat tinggal, dan transportasi. Dalam hal ini, nominal pengeluaran didapatkan dari masukan pengguna.

Setelah semua variabel diinisialisasi, program melanjutkan dengan menghitung total pengeluaran. Proses ini dilakukan dengan menjumlahkan semua pengeluaran yang telah dicatat sebelumnya. Hasil penjumlahan ini disimpan dalam variabel `total_pengeluaran`. Langkah berikutnya adalah menghitung sisa anggaran, yaitu dengan mengurangi total pengeluaran dari pendapatan. Hasil perhitungan ini disimpan dalam variabel `sisa_anggaran`. Terakhir, program menampilkan hasil perhitungan ke layar menggunakan fungsi `print()`. Informasi yang ditampilkan meliputi pendapatan bulanan, total pengeluaran, dan sisa anggaran. Ketika program dijalankan, hasil outputnya adalah sebagai berikut:

Program ini dirancang untuk memberikan gambaran sederhana tentang bagaimana pengelolaan keuangan pribadi dapat diimplementasikan dalam sebuah program komputer. Dengan pendekatan yang runtut, kode ini menunjukkan konsep dasar pemrograman seperti penggunaan variabel, operasi aritmatika, dan pencetakan hasil ke layar, sehingga sangat cocok untuk dipelajari oleh pemula di semua jurusan.

2.5 Tugas 10 soal pemrograman python meliputi : operator, variable dan input/output

Ketentuan Tugas : setiap mahasiswa diminta mengerjakan 2 soal: 1 soal sesuai dengan NIM-nya, dan 1 soal bebas memilih.

Tugas 1 : Perhitungan Sederhana

Deskripsi:

Buat program untuk menghitung jumlah dua angka yang dimasukkan oleh pengguna.

Contoh Input/Output:

```
Masukkan angka pertama: 5
Masukkan angka kedua: 7
Hasil: 12
```

Tugas 2: Operasi Matematika**Deskripsi:**

Tulis program yang menerima dua angka dari pengguna, lalu menampilkan hasil operasi penjumlahan, pengurangan, perkalian, dan pembagian.

Contoh Input/Output:

```
Masukkan angka pertama: 8
Masukkan angka kedua: 4
Hasil penjumlahan: 12
Hasil pengurangan: 4
Hasil perkalian: 32
Hasil pembagian: 2.0
```

Tugas 3: Luas Persegi Panjang**Deskripsi:**

Buat program untuk menghitung luas persegi panjang. Pengguna harus memasukkan panjang dan lebar.

Contoh Input/Output:

```
Masukkan panjang: 10
Masukkan lebar: 5
Luas persegi panjang: 50
```

Tugas 4: Menukar Nilai Variabel**Deskripsi:**

Tulis program untuk menukar nilai dua variabel yang dimasukkan oleh pengguna.

Contoh Input/Output:

```
Masukkan nilai A: 5
Masukkan nilai B: 10
Setelah ditukar:
Nilai A: 10
```

Nilai B: 5

Tugas 5: Menampilkan Daftar Barang

Deskripsi:

Buat program untuk menampilkan daftar barang.

Contoh Input/Output:

Masukkan nama barang pertama: Sabun

Masukkan nama barang kedua: Shampoo

Masukkan nama barang ketiga: Pasta Gigi

Daftar barang yang Anda masukkan: ['Sabun', 'Shampoo', 'Pasta Gigi']

Tugas 6: Menghitung Umur

Deskripsi:

Buat program untuk menghitung umur berdasarkan tahun lahir yang dimasukkan pengguna.

Contoh Input/Output:

Masukkan tahun lahir: 2000

Umur Anda: 23

Tugas 7: Mengubah Jam ke Detik

Deskripsi:

Buat program yang menerima input waktu dalam jam, lalu mengubahnya menjadi detik.

Contoh Input/Output:

Masukkan waktu dalam jam: 2

Waktu dalam detik: 7200

Tugas 8: Kalkulator Diskon

Deskripsi:

Buat program untuk menghitung harga akhir sebuah barang setelah diskon.

Contoh Input/Output:

Masukkan harga awal: 100000

Masukkan diskon (%): 20

Harga setelah diskon: 80000

Tugas 9: Menghitung Nilai Rata-Rata

Deskripsi:

Tulis program untuk menghitung rata-rata dari tiga angka yang dimasukkan pengguna.

Contoh Input/Output:

```
Masukkan angka pertama: 10
Masukkan angka kedua: 20
Masukkan angka ketiga: 30
Rata-rata: 20.0
```

Tugas 10: Mengonversi Suhu

Deskripsi:

Buat program untuk mengonversi suhu dari Celcius ke Fahrenheit.

Contoh Input/Output:

```
Masukkan suhu dalam Celcius: 25
Suhu dalam Fahrenheit: 77.0
```

Kesimpulan

Dalam modul ini, kita telah membahas dasar-dasar sintaks Python, termasuk deklarasi variabel, tipe data, operator, dan operasi input/output. Dengan berlatih pada latihan yang diberikan, mahasiswa dapat memperkuat pemahaman mereka tentang konsep-konsep fundamental ini. Menguasai dasar-dasar ini akan menjadi fondasi untuk topik pemrograman yang lebih lanjut.