**LAPORAN PRAKTIKUM ALGORITMA DAN PEMROGRAMAN KOMPUTER**

**MODUL 3**

**OPERASI DASAR INPUT DAN OUTPUT**

Laporan ini disusun untuk memenuhi Tugas Mata Kuliah

Praktikum Basis Data

**A logo for a university

Description automatically generated**

Disusun Oleh :

AHSAN MAULANA RIZQI

104062400071

S1 BD-04-B

**PROGRAM STUDI S1 BISNIS DIGITAL**

**TELKOM UNIVERSITY PURWOKERTO**

**2025**

1. **Dasar Teori**
   1. Indentasi

Indentasi mengacu pada spasi atau tab yang digunakan di awal baris kode untuk menunjukkan tingkat penumpukannya dalam blok. Dalam Python, indentasi bukan hanya masalah gaya atau preferensi, tetapi merupakan persyaratan sintaksis. Tidak seperti bahasa pemrograman lain yang menggunakan kurung kurawal atau kata kunci untuk mendefinisikan blok kode.

Aturan Dasar Indentasi: Dalam Python, setiap baris kode dalam satu blok harus diindentasi dengan jumlah spasi atau tab yang sama. Konvensi standar adalah menggunakan empat spasi untuk indentasi. Sebaiknya hindari penggunaan tab untuk indentasi guna mencegah tercampurnya tab dan spasi.

Tingkat Indentasi dan Nesting: Python menggunakan indentasi untuk menunjukkan tingkat nesting dalam sebuah blok. Setiap blok nesting harus diindentasi lebih jauh dari blok induknya. Ini membantu dalam merepresentasikan hierarki blok kode secara visual.

* 1. Komentar

Komentar adalah kode yang ada pada *script* python dan tidak kode tersebut tidak akan dieksekusi. Komentar biasanya digunakan untuk memberikan informasi tambahan terhadap script yang ditulis. Untuk menuliskan komentar diawali dengan menuliskan tanda pagar (**#**).

* 1. Python *Input* dan *Output*

*Input* dan *output* merupakan bagian dalam bahasa pemrograman yang sangat penting. *Input* adalah masukan yang kita berikan kepada program lalu akan muncul outputnya. *Output* merupakan hasil keluaran dari sesuatu yang telah kita input sebelumnya. salah satu fungsi tersebut yaitu fungsi print(), fungsi *print*() pada python digunakan untuk melakukan operasi output sedangkan untuk operasi input pada python menggunakan fungsi *input*().

|  |
| --- |
| ***Catatan: Segala bentuk inputan yang ditambahkan menggunakan fungsi input() tipe datanya akan dianggap sebagai string. Jadi ketika kita inputkan angka maka angka tersebut akan tetap dianggap sebagai string, untuk itu kita perlu melakukan yang namanya casting tipe data atau mengkonversi data yang diinputkan sesuai dengan apa yang kita harapkan*** |

* 1. *Casting* Tipe Data

*Casting* tipe data adalah mengubah tipe data tertentu menjadi tipe data lainnya. Contoh kita mengubah tipe data integer menjadi tipe data float, supaya bisa berubah maka perlu menggunakan casting. Berikut fungsi casting yang bisa digunakan

*Tabel 1. Fungsi-fungsi casting dan kegunaannya*

|  |  |
| --- | --- |
| **Fungsi *Casting*** | **Keteranagan** |
| **int()** | Mengubah menjadi integer |
| **float()** | Mengubah menjadi float |
| **bool()** | Mengubah menjadi Boolean |
| **chr()** | Mengubah menjadi karakter |
| **str()** | Mengubah menjadi string |
| **bin()** | Mengubah menjadi bilangan biner |
| **hex()** | Mengubah menjadi bilangan heksadesimal |
| **oct()** | Mengubah menjadi bilangan oktal |

1. **Pembahasan Tugas Guided**
   1. Indentasi

A black background with white text

AI-generated content may be incorrect.

*Gambar 1. Print angka 5*

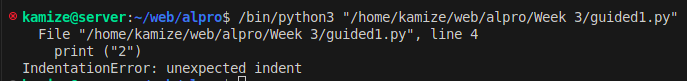
*Gambar 2. Hasil output*

Perintah *print* (“5”)pada gambar 1 di baris 1 akan menghasilkan output gambar 2.

A screenshot of a computer

AI-generated content may be incorrect.

*Gambar 3. Print anka menggunakan indentasi*

**

*Gambar 4. output error*

Penggunaan indentasi yang salah seperti pada cotoh gambar 3 yang akan menghasilkan output error seperti pada gambar 4. Pesan *syntax error: unexpected indent* yang artinya terjadi error yang berupa kesalahan dalam pemberian indentasi.

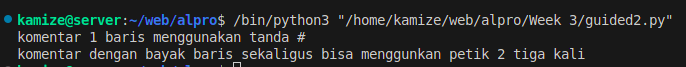
* 1. *Comment*

**A screenshot of a computer

AI-generated content may be incorrect.**

*Gambar 5. Contoh pemggunaan komentar*

Baris kode yang diawali dengan tanda pagar (#) menunjukan komentar untuk satu baris. Sedangkan baris yang diawali dan diakhiri dengan tanda perik dua (“““ … ”””) tiga kali menunjukan komentar dengan banya baris.



*Gambar 6. Output komentar*

kode program pada gamber 5 akan menghasilkan output pada gambar 6. Yang dimana komenter tidak di eksekusi dan hanya dijadikan keterangan pada *source code*.

* 1. *Output*

**A computer screen shot of a program

AI-generated content may be incorrect.**

*Gambar 7. Penggunaan fungsi print()*

Kode di gambar 7 adalah latihan penggunaan fungsi `*print*()` dalam Python untuk menampilkan output ke layar dengan berbagai format. Fungsi `*print*()` memiliki beberapa parameter opsional seperti `*sep*` (separator) dan `*end*`, yang digunakan untuk mengatur cara elemen-elemen dalam output dipisahkan dan bagaimana baris output diakhiri.

A screen shot of a computer

AI-generated content may be incorrect.

*Gambar 8. fungsi print() angka dan teks*

Pada gambar 8, kode menampilkan angka dan teks sederhana:

Baris pertama mencetak tiga angka (11, 22, 33) yang secara default dipisahkan oleh spasi karena parameter `sep` tidak diubah. Baris kedua mencetak string "Belajar membuat komputer dalam python".

Selanjutnya, kode menggunakan string interpolasi (f-string) untuk mencetak variabel `prodi` dan `kampus`:

A screen shot of a computer program

AI-generated content may be incorrect.

*Gambar 9. fungsi print() untuk variabel*

F-string memungkinkan penggabungan nilai variabel ke dalam string dengan format yang lebih rapi. Outputnya adalah "program studi : S1 Bisnis Digital,Tel-U Purwokerto".

Bagian berikutnya menunjukkan penggunaan parameter `sep` dan `end`:

A computer code with text

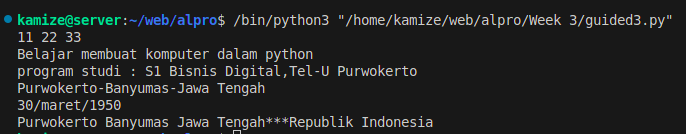
AI-generated content may be incorrect.

*Gambar 10. fungsi print() parameter “sep” dan “end”*

Parameter `sep` menentukan karakter yang digunakan untuk memisahkan elemen dalam output. Pada baris pertama, elemen dipisahkan dengan tanda hubung (`-`), menghasilkan "Purwokerto-Banyumas-Jawa Tengah". Pada baris kedua, elemen dipisahkan dengan garis miring (`/`), menghasilkan "30/maret/1950".

Terakhir, kode menggunakan parameter `end` untuk mengatur karakter yang mengakhiri baris output:

Parameter `end` menggantikan karakter akhir default (`\n`, baris baru). Pada baris pertama, output diakhiri dengan "\*\*\*", sehingga "Purwokerto-Banyumas-JawaTengah\*\*\*" dicetak tanpa baris baru. Baris kedua mengakhiri output dengan spasi, sehingga "Republik " dicetak diikuti oleh "Indonesia" pada baris yang sama.

**

*Gambar 11. Hasil Output*

* 1. *Input*

**A screen shot of a computer program

AI-generated content may be incorrect.**

*Gambar 12. Fungsi input() dan print()*

Kode gambar 12 adalah contoh sederhana dari penggunaan fungsi bawaan Python, yaitu *input*() dan *print*(), untuk menerima masukan dari pengguna dan menampilkan hasilnya ke layar.

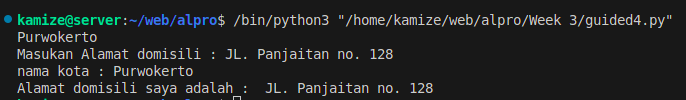
kota = *input*(): Fungsi input() dipanggil tanpa argumen, sehingga tidak ada pesan prompt yang ditampilkan. Program akan menunggu pengguna memasukkan teks, yang kemudian disimpan dalam variabel kota.

alamat = *input*("Masukan Alamat domisili : ") : Fungsi *input*() dipanggil dengan argumen *string* "Masukan Alamat domisili : ". *String* ini ditampilkan sebagai prompt untuk memberi tahu pengguna apa yang harus dimasukkan. Masukan pengguna kemudian disimpan dalam variabel alamat.

Fungsi *input*() selalu mengembalikan data dalam bentuk string, sehingga masukan pengguna akan disimpan sebagai teks, terlepas dari apa yang dimasukkan.

*print*(f"nama kota : {kota}") : Fungsi *print*() digunakan untuk mencetak teks ke layar. Dalam hal ini, *f-string* (`f"..."`) digunakan untuk menyisipkan nilai variabel kota ke dalam string. Misalnya, jika pengguna memasukkan "Purwokerto" untuk kota, outputnya akan menjadi nama kota : Purwokerto.

*print*("Alamat domisili saya adalah : ", alamat) : Fungsi *print*() mencetak teks statis "Alamat domisili saya adalah : " diikuti oleh nilai variabel alamat. Fungsi ini secara default memisahkan argumen dengan spasi (sep=" ").

**

*Gambar 13. Hasil Output*

* 1. *Casting Type Data*

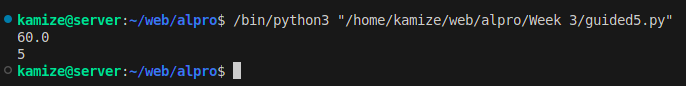
Kode gambar 14 menunjukkan penggunaan *casting* (konversi tipe data) dalam Python, serta perbedaan hasil operasi sebelum dan sesudah casting. Berikut adalah penjelasan rinci:

A screenshot of a computer code

AI-generated content may be incorrect.

*Gambar 14. Casting float dan integer*

*float*(60) : Fungsi *float*() mengonversi bilangan bulat 60 menjadi bilangan desimal (*floating-point*). Outputnya adalah 60.0. *int*(5.8967): Fungsi *int*() mengonversi bilangan desimal 5.8967 menjadi bilangan bulat dengan membuang bagian desimalnya. Outputnya adalah 5.

**

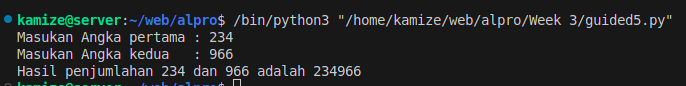
*Gambar 15. Hasil Output*

*A screen shot of a computer code

AI-generated content may be incorrect.*

*Gambar 16. Sebelum casting integer*

Fungsi *input*() menerima masukan dari pengguna dalam bentuk *string*. Variabel angka1 dan angka2 menyimpan nilai string yang dimasukkan. Operasi angka1 + angka2 adalah penggabungan string (*concatenation*), bukan penjumlahan numerik. Misalnya, jika pengguna memasukkan 234 untuk angka1 dan 966 untuk angka2, hasilnya adalah *string* 234966. Fungsi *print*() kemudian menampilkan hasil penggabungan string tersebut.

**

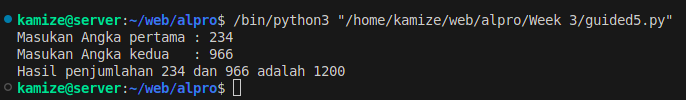
*Gambar 17. Hasil Output*

A screen shot of a computer code

AI-generated content may be incorrect.

*Gambar18. Setelah casting integer*

Fungsi *int*() digunakan untuk mengonversi masukan *string* dari *input*() menjadi bilangan bulat (integer). Dengan demikian, variabel angka1 dan angka2 sekarang menyimpan nilai numerik. Operasi angka1 + angka2 adalah penjumlahan numerik. Misalnya, jika pengguna memasukkan 234 untuk angka1 dan 966 untuk angka2, hasilnya adalah bilangan bulat 1200. Fungsi *print*() menampilkan hasil penjumlahan numerik tersebut.

**

*Gambar 19. Hasil Output*

1. **Pembahasan Tugas Unguided**
   1. Tuliskan kode di dalamnya sehingga hasil yang didapatkan seperti berikut :

A blue arrow with stars

AI-generated content may be incorrect.

A computer screen shot of a code

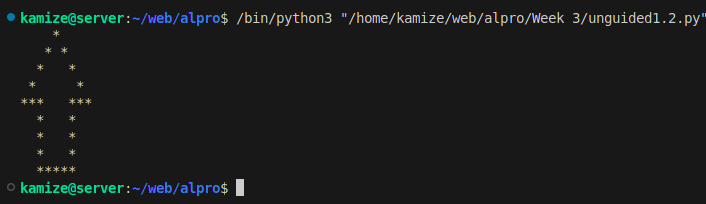
AI-generated content may be incorrect.

*Gambar 19. Code isialisasi unguided 1*

Inisalisasi variabel sisi dengan nilai 9 sebagai panjang sisi segitiga dan spasi yang dihitung dari setengah nilai sisi untuk menentukan nilai spasi awal sebelum simbol \* pada setiap baris segitiga.

*For count in range* (1,sisi +1 ,2) untuk menghasilkan *looping* dari 1 sampai variabel sisi+1 dengan kenaikan 2 yang akan menghasilkan angka ganjil. Jika variabel *count* sama dengan 1 maka cetak \* dengan spasi awal sebanyak variabel spasi. Jika variabel *count* sama dengan variabel sisi maka jalankan proses variabel sisi dan count dibagi dengan 3 untuk menentukan jumlah \* dan spasi untuk membuat baris terakhir segitiga. Jika kondisi sebelumnya tidak ada yang terpenuhi maka akan mencetak dua simbol \* dengan spasi diataranya. Jumlah spasi sebanyak variabel count – 2. Variabel spasi akan dikurangi 1 disetiap barisnya untuk menciptakan pola runcing.

*For count in range* (sisi) untuk menghasilkan *looping* dari 1 sampai variabel sisi dengan output spasi sejumlah setengah dari variabel sisi dan 2 simbol \* yang diataran keduanya terdapat spasi sebanyak setengah dari variabel sisi. Baris terakhir akan dicetak penuh.

**

*Gambar 20. hasil output unguided 1*

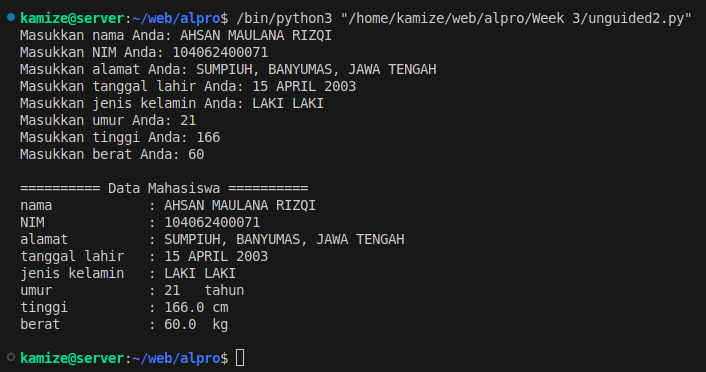
* 1. Gunakan fungsi *input*() untuk memasukkan biodata dan gunakan casting tipe data yang benar. Data dari biodata sebagai berikut :
     + Nama
     + NIM
     + Alamat
     + Tanggal Lahir
     + Jenis Kelamin
     + Umur
     + Berat Badan
     + Tinggi Badan
     + No. HP

*A screen shot of a computer program

AI-generated content may be incorrect.*

*Gambar 21. input biodata unguided 2*

Inisialisasi variabel input menggunakan casting *str*(), *int*(), dan *float*(). *Casting str*() untuk variabel nama, Alamat, tanggal\_lahir, dan jenis\_kelamin. *Casting int*() untuk variabel NIM dan umur. *Casting float*() untuk variabel tinggi dan berat. Variabel *display* sebagai template untuk menampilkan data mahsiswa sesuai *input user*.



*Gambar 22. hasil output unguided 2*

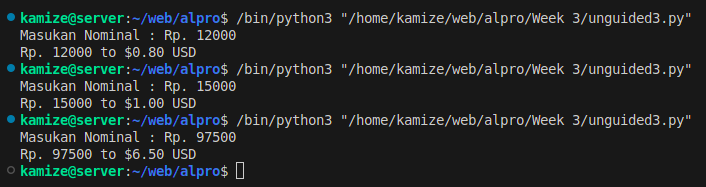
* 1. Diketahui kurs rupiah dalam dollar sebagai berikut: 1 dollar = 15000 rupiah Buatlah sebuah program konversi dari rupiah menjadi dollar, gunakan fungsi *input*() dan casting.

A screen shot of a computer code

AI-generated content may be incorrect.

*Gambar 23. konversi dolar ke rupiah unguided 3*

Inisialisasi variabel rupiah dengan fungsi input yang di casting int(). Variabel kurs dengan nilai 15000. Variabel dolar untuk menyimpan rumus dasar konversi rupiah ke dolar yaitu variabel rupiah dibagi dengan variabel kurs. Hasil outputnya pada gambar 24.

**

*Gambar 24. hasil output unguided 3*

1. **Ringkasan**

Laporan ini menjelaskan tentang konsep-konsep dasar bahasa pemrograman Python, di dalam mudul ini menjelaskan tentang indentasi, komentar, input-output, dan konversi tipe data. Indentasi digunakan untuk tanda pada blok kode yang harus digunakan dengan benar agar program yang dijalankan tidak mengalami pesan *syntax error: unexpected indent*. Komentar dalam bahasa pemprogaman Python ditulis menggunakan simbol/tanda pagar untuk satu baris sedangkan tanda kutip tiga kali (“““ … ”””) untuk komentar lebih dari satu baris, yang biasanya digunakan sebagai dokumentasi kode.

Dalam hal input dan output, Bahasa pemprogaman Python menggunakan fungsi *print*() untuk menampilkan hasil output ke terminal atau layer, sedangkan *input*() untuk menerima masukan pengguna dari terminal. Fungsi *input*() selalu memperoleh data yang berupa tipe data *string*, sehingga jika memerlukan tipe data selain *string*, maka harus dikonversi menggunakan fungsi *casting* seperti int() untuk tipe data integer, *float*() untuk tipe data *float*, atau *bool*() untuk tipe data *boolean*.

Laporan ini juga berisi tugas praktik, seperti penggunaan indentasi, komentar, dan fungsi *print*() serta *input*(). Salah satu contoh latihan adalah penggunaan parameter *sep* dan *end* dalam fungsi *print*() untuk mengatur pemisah elemen *output* dan cara mengakhiri baris. Selain itu, dicontohkan pula penggunaan *casting* untuk mengubah *input string* menjadi angka, agar dapat digunakan dalam operasi aritmatika.

Dalam bagian tugas mandiri (unguided), diberikan studi kasus seperti membuat segitiga dengan simbol \*, memasukkan biodata dengan *casting* tipe data yang benar, serta konversi nilai mata uang dari Rupiah ke Dollar menggunakan fungsi *input*() dan operasi matematika sederhana.

Dalam laporan ini, dapat disimpulkan bahwa pentingnya memehami indentasi, komentar, input-output, dan konversi tipe data dalam bahasa pemrograman Python. Menggunakan aturan ini dengan benar akan membantu dalam menulis kode yang lebih rapi, mudah dibaca, serta menghindari kesalahan yang dapat menghambat jalannya program.