**LAPORAN PRAKTIKUM ALGORITMA DAN PEMROGRAMAN KOMPUTER**

**MODUL 5**

**SELECTION**

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. Percabangan

Percabangan adalah cara untuk menjalankan kode yang berbeda tergantung pada apakah kondisi yang diberikan bernilai benar (*True*) atau salah (*False*). Bentuk paling sederhana dari percabangan ini adalah dengan menggunakan if untuk memeriksa kondisi.

* 1. *if statement*

*if statement* digunakan ketika ada satu pilihan keputusan. Percabangan ini digunakan untuk menjalankan program jika kondisinya bernilai *True* dan jika kondisi bernilai *False* maka *if statement* tidak akan dijalankan.

* 1. *if else statement*

*If else statement* adalah percabangan program yang memiliki dua kondisi. Jika kondisi pertama terpenuhi, maka blok kode program di dalam *if statement* akan dieksekusi. Jika kondisi pertama tidak terpenuhi, maka blok kode program di dalam *else statement* akan dieksekusi.

* 1. *if elif statement*

*If elif statement* adalah percabangan program yang memiliki lebih dari dua kondisi. Jika kondisi pertama terpenuhi, maka blok kode program di dalam *if statement* akan dieksekusi. Jika kondisi pertama tidak terpenuhi, maka kondisi kedua akan dicek. Jika kondisi kedua terpenuhi, maka blok kode program di dalam *if elif statement* akan dieksekusi. Jika kondisi kedua tidak terpenuhi, maka *statement* tidak akan ada yang dieksekusi.

* 1. *if elif else statement*

*If elif else statement* adalah percabangan program yang memiliki lebih dari dua kondisi. Jika kondisi pertama terpenuhi, maka blok kode program di dalam *if statement* akan dieksekusi. Jika kondisi pertama tidak terpenuhi, maka kondisi kedua akan dicek. Jika kondisi kedua terpenuhi, maka blok kode program di dalam *elif statement* akan dieksekusi. Jika kondisi kedua tidak terpenuhi, maka blok kode program di dalam *else statement* akan dieksekusi.

* 1. *Nested if statement*

*Nested if statement* adalah cara untuk memasukkan satu atau lebih pernyataan *if-else* di dalam pernyataan *if-else* lainnya. Dalam *nested conditional statement*, kita dapat mengevaluasi beberapa kondisi dalam beberapa tingkat, tergantung pada kebutuhan program. Pernyataan *if-else* yang ditempatkan di dalam *if-else* lainnya disebut sebagai blok bersarang (*nested block*).

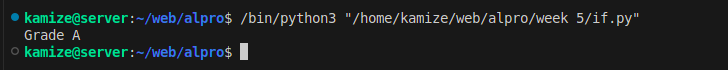
1. **Pembahasan Tugas Guided**
   1. Latihan *if statement*

A screenshot of a computer

AI-generated content may be incorrect.

*Gambar 1. Contoh code if statement*

Pada gambar 1 varibel nilai memiliki nilai 91 dan if nilai >= 90 sebagai *statement*. Jika *statement* pada variabel nilai lebih dari sama dengan 90 bernilai *True*, maka *if statement* akan dieksekusi dan teks Grade A di tampikan ke layar. Jika tidak maka *statement* tersebut tidak dieksekusi dan tidak ditampikan ke layar.

**

*Gambar 2. hasil output pada latihan if statement*

Pada gamber 2 menghasilkan *output* Grade A karena variabel nilai yang memiliki nilai 91 dan variabel nilai tersebut lebih dari sama dengan 90, maka *if statement* dieksekusi yang menghasilkan tulisan Grade A yang ditampilkan ke layar.

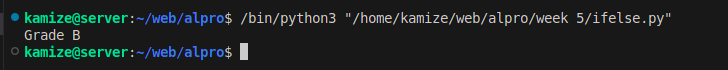
* 1. Latihan *if else statement*

A screenshot of a computer code

AI-generated content may be incorrect.

*Gambar 3. Contoh code if else statement*

Pada gambar 3 varibel nilai memiliki nilai 71 dan if nilai >= 90 sebagai *statement*. Jika *statement* pada variabel nilai lebih dari sama dengan 90 bernilai *True*, maka *if statement* akan dieksekusi dan teks Grade A di tampikan ke layar. Jika tidak maka *else statement* akan dieksekusi dan teks Grade B ditampikan ke layar.



*Gambar 4. hasil output pada latihan if else statement*

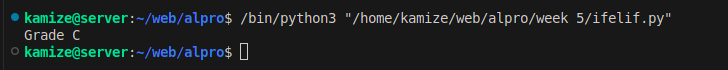
Pada gamber 4 menghasilkan *output* Grade B karena variabel nilai yang memiliki nilai 71 dan variabel nilai tersebut tidak lebih dari sama dengan 90, maka *if statement* tidak dieksekusi. Tetapi *else statement* akan dieksekusi dan teks Grade B yang ditampilkan ke layar.

* 1. Latihan *if elif statement*

*A screenshot of a computer code

AI-generated content may be incorrect. Gambar 5. Contoh code if elif statement*

Pada gambar 5 varibel nilai memiliki nilai 61. Jika *statement* pertama dengan kondisi variabel nilai lebih dari sama dengan 90 divalidasi dan menghasilkan nilai *True*, maka *if statement* pertama akan dieksekusi dan teks Grade A ditampikan ke layar. Jika *statement* pertama *False*, maka *statement* kedua divalidasi dengan kondisi variabel nilai lebih dari sama dengan 70 dan menghasilkan nilai *True*, maka *if statement* kedua akan dieksekusi dan teks Grade B ditampikan ke layar. Jika *statement* kedua *False*, maka *statement* ketiga divalidasi dengan kondisi variabel nilai lebih dari sama dengan 60 dan menghasilkan nilai *True*, maka *if statement* ketiga akan dieksekusi dan teks Grade C di tampikan ke layar. Jika tidak ada *statement* yang bernilai *True* maka *statement* tidak akan dieksekusi dan tidak ditampikan ke layar.

**

*Gambar 6. hasil output pada latihan if elif statement*

Pada gamber 6 menghasilkan *output* Grade C karena variabel nilai yang memiliki nilai 61 dan variabel nilai tersebut memenuhi *statement* ketiga, maka *if statement* ketiga dieksekusi dan teks Grade C akan ditampilkan ke layar.

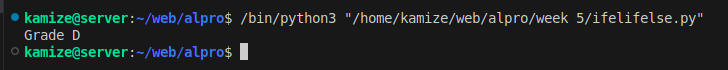
* 1. Latihan *if elif else statement*

*A screen shot of a computer code

AI-generated content may be incorrect.*

*Gambar 7. Contoh code if elif else statement*

Pada gambar 7 varibel nilai memiliki nilai 51. Jika *statement* pertama dengan kondisi variabel nilai lebih dari sama dengan 90 divalidasi dan menghasilkan nilai *True*, maka *if statement* pertama akan dieksekusi dan teks Grade A ditampikan ke layar. Jika *statement* pertama *False*, maka *statement* kedua divalidasi dengan kondisi variabel nilai lebih dari sama dengan 70 dan menghasilkan nilai *True*. Maka *if statement* kedua akan dieksekusi dan teks Grade B ditampikan ke layar. Jika *statement* kedua *False*, maka *statement* ketiga divalidasi dengan kondisi variabel nilai lebih dari sama dengan 60 dan menghasilkan nilai *True*, maka *if statement* ketiga akan dieksekusi dan teks Grade C di tampikan ke layar. Jika tidak ada *statement* yang bernilai *True* maka *else statement* akan dieksekusi dan teks Grade D ditampikan ke layar.

**

*Gambar 8. hasil output pada latihan if elif else statement*

Pada gamber 8 menghasilkan *output* Grade D karena variabel nilai yang memiliki nilai 51 dan variabel nilai tersebut tidak memenuhi *statement* yang ada, maka *if statement* tidak dieksekusi. Tetapi *else statement* akan dieksekusi dan teks Grade D yang ditampilkan ke layar.

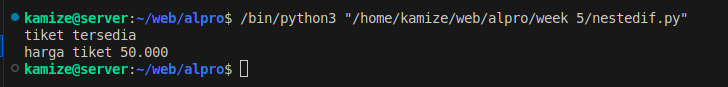
* 1. Latihan *Nested if statement*

*A screenshot of a computer program

AI-generated content may be incorrect.*

*Gambar 9. Contoh code nested if statement*

Pada gambar 1 varibel umur memiliki nilai 38. Jika *statement* pertama dengan kondisi variabel umur lebih dari sama dengan 11 divalidasi dan menghasilkan nilai *True*, maka *if statement* pertama akan dieksekusi. Yang menghasilkan *output* teks tiket tersedia yang ditampikan ke layar dan *statement*. *statement* lain yang terdapat pada *statement* pertama memeiliki kondisi jika umur kurang dari sama dengan 20 atau umur lebih dari sama dengan 60, maka menghasikan *output* teks harga tiket 35.000, jika tidak maka akan menghasilkan *output* teks harga tiket 50.000. jika *statement* pertama tidak terpenuhi maka *else statement* akan dieksekusi yang menghasilkan *output* teks tiket tidak tersedia.

**

*Gambar 10. Hasil output pada latihan nested if statement*

Pada gamber 10 menghasilkan *output* teks tiket tersedia dan harga tiket 50.000 karena variabel umur yang memiliki nilai 38 dan variabel umur tersebut memenuhi *statement* pertama, maka menghasilkan *output* teks tiket tersedia. Akan tetapi nested *if statement*nya tidak terpenuhi, maka akan menghasilkan *output* teks harga tiket 50.000 yang ditampilkan ke layar.

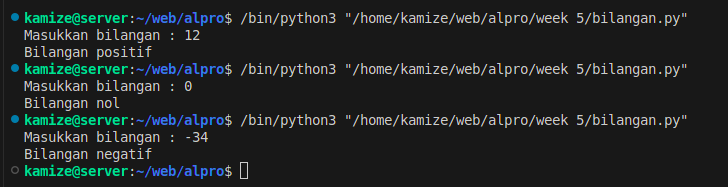
* 1. Latihan *if elif else statement* dengan *input*

*A screen shot of a computer

AI-generated content may be incorrect.*

*Gambar 11. Contoh code if elif else statement dengan input*

Pada gambar 11 variabel bilangan sebagai nilai input dari pengguna. Jika *statement* pertama yang memiliki kondisi bilangan kurang dari 0 divalidasi dan menghasilkan nilai *True*, maka *statement* pertama akan dieksekusi dan menghasilkan *output* teks berupa bilangan negatif. Jika *statement* pertama menghasilkan nilai *False*, maka *statement* kedua yang memiliki kondisi bilangan sama dengan 0 divalidasi dan menghasilkan nilai *True*. *Statement* kedua akan dieksekusi yang menghasilkan ouput teks berupa bilangan nol. Jika *if statement* tidak ada yang terpenuhi maka *else statement* akan dieksekusi yang menghasilkan *output* teks berupa bilangan positif.

**

*Gamabar 12. Hasil output pada latihan if elif else statement dengan input*

Pada gambar 12 menghasilkan *output* teks bilangan negatif jika pengguna memasukan bilangan kurang dari 0, contohnya -34. Jika pengguna memasukan bilangan 0, maka akan menghasilkan *output* teks bilangan nol. Jika pengguna tidak memasukan sesusi *statement* pertama atau *statement* kedua maka akan menghasilkan *output* teks bilangan positif, contohnya 12.

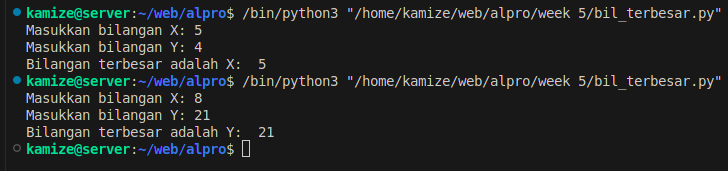
* 1. Latihan *if else statement* dengan operator

*A screenshot of a computer code

AI-generated content may be incorrect.*

*Gambar 13. Contoh code if else statement dengan operator*

Pada gambar 13 variabel x dan y digunakan untuk menyimpan nilai input dari pengguna. Jika *statement* yang memiliki kondisi x lebih dari y divalidasi dan menghasilkan nilai *True*, maka *statement*nya akan dieksekusi yang menghsailkan *output* teks bilangan terbesar adalah x : (nilai variabel x). Jika *if statement* tidak terpenuhi maka *else statement* akan dieksekusi yang menghasilkan *output* bilangan terbesar adalah y : (nilai variabel y).



*Gambar 14. Hasil output pada latihan if else statement*

Pada gambar 14 menunjukan beberapa contoh *output* jika pengguna memasukan nilai variabel x dan y. Yang pertama pengguna memasukan nilai variabel x dengan nilai 5 dan varibel y dengan nilai 4 karena pengguna memasukan nilai variabel x lebih dari nilai variaberl y. Maka *if statement* akan dieksekusi yang menghasilkan nilai *output* teks bilangan terbesar adalah x : 5. Yang kedua pengguna memasukan nilai variabel x dengan nilai 8 dan variabel y dengan nilai 21 karena pengguna memasukan nilai varabel x tidak lebih dari variabel y. Maka *if statement* tidak dieksekusi tetapi *else statement* akan dieksekusi yang menghasilkan *output* teks bilangan terbesar adalah y : 21.

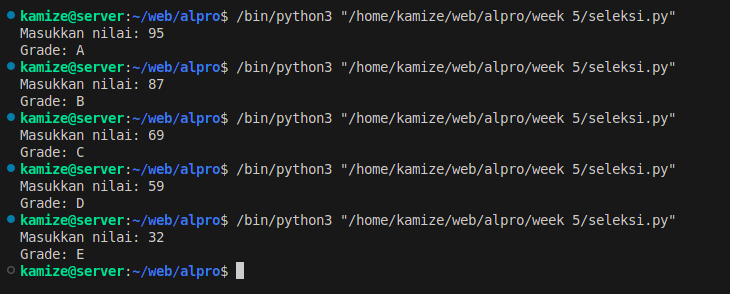
1. **Pembahasan Tugas Unguided**
   1. Terjemahkan *flowchart* berikut ke dalam kode program python simpan dengan nama seleksi.py:

*A screenshot of a computer

AI-generated content may be incorrect.*

*Gambar 15. Jawaban tugas unguided 1*

Pada gambar 15 varibel nilai digunakan untuk menyimpan *value* input dari pengguna. Jika *statement* pertama divalidasi dengan kondisi variabel nilai lebih dari sama dengan 90 divalidasi dan menghasilkan nilai *True*, maka *if statement* pertama akan dieksekusi yang menghasilkan variavel grade memiliki *value* A. Jika *statement* pertama *False*, maka *statement* kedua divalidasi dengan kondisi variabel nilai lebih dari sama dengan 70 and variabel nilai kurang dari 90 dan menghasilkan nilai *True*. Maka *if statement* kedua akan dieksekusi yang menghasilkan variabel grade mimiliki *value* B. Jika *statement* kedua *False*, maka *statement* ketiga divalidasi dengan kondisi variabel nilai lebih dari sama dengan 60 and variabel nilai kurang dari 70 dan menghasilkan nilai *True*. Maka *if statement* ketiga akan dieksekusi yang menghasilkan variabel grade mimiliki *value* C. Jika *statement* ketiga *False*, maka *statement* keempat divalidasi dengan kondisi variabel nilai lebih dari sama dengan 50 and variabel nilai kurang dari 60 dan menghasilkan nilai *True*. Maka *if statement* keempat akan dieksekusi yang menghasilkan variabel grade mimiliki *value* D. Jika tidak ada *statement* yang bernilai *True* maka *else statement* akan dieksekusi yang menghasilkan variabel grade memiliki *value* E. Hasil *output* *string* ditampilkan kelayar berupa teks Grade : (*value* dari variabel grade).

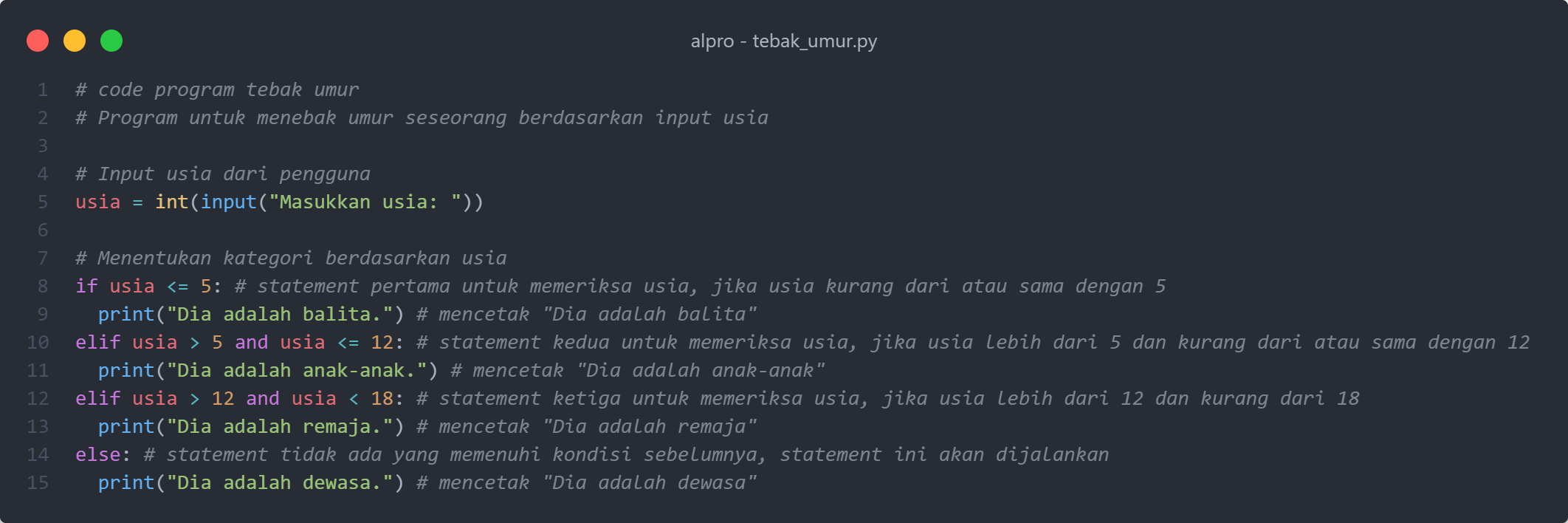
**

*Gambar 16. Hasil output dari jawaban tugas unguided 1*

Pada gambar 16 menunjukan beberapa contoh *output* jika pengguna memasukan *value* ke variabel nilai:

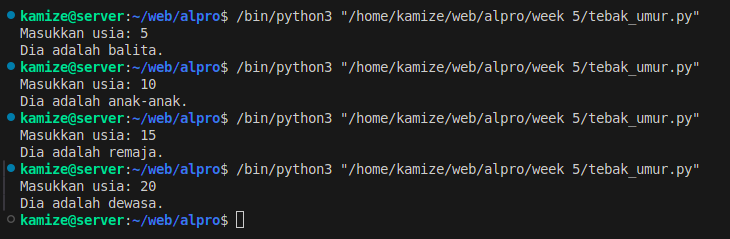
1. Jika pengguna memasukan *value* 95 ke variabel nilai. Maka *if statement* pertama akan dieksekusi karena *value* variabel nilai memenuhi kondisi pertama yang menghasilkan *output* *string* Grade : A.
2. Jika pengguna memasukan *value* 87 ke variabel nilai. Maka *if statement* kedua akan dieksekusi karena *value* variabel nilai memenuhi kondisi kedua yang menghasilkan *output* *string* Grade : B.
3. Jika pengguna memasukan *value* 69 ke variabel nilai. Maka *if statement* ketiga akan dieksekusi karena *value* variabel nilai memenuhi kondisi ketiga yang menghasilkan *output* *string* Grade : C.
4. Jika pengguna memasukan *value* 59 ke variabel nilai. Maka *if statement* keempat akan dieksekusi karena *value* variabel nilai memenuhi kondisi pertama yang menghasilkan *output* *string* Grade : D.
5. Jika pengguna memasukan *value* 32 ke variabel nilai. Maka *if statement* tidak akan dieksekusi karena *value* variabel usia tidak memenuhi kondisi yang ada, tetapi *else statement* akan dieksekusi yang menghasilkan *output* *string* Grade : E.
   1. Terjemahkan skenario berikut ke dalam kode program python, simpan dengan nama tebak\_umur.py:

* Jika usia 5 tahun ke bawah (usia<=5), maka dia adalah balita
* Jika usia di atas 5 tahun ke atas sampai 12 tahun (usia >5 dan usia <=12), maka dia adalah anak-anak
* Jika usia di atas 12 tahun dan usia di bawah 18 tahun (usia >12 dan usia < 18) maka dia adalah remaja
* Jika usia di atas 18 tahun maka dia adalah dewasa



*Gambar 17. Jawaban tugas unguided 2*

Pada gambar 17 varibel usia digunakan untuk menyimpan *value* dari input pengguna. Jika *statement* pertama dengan kondisi variabel usia kurang dari sama dengan 5 divalidasi dan menghasilkan nilai *True*, maka *if statement* pertama akan dieksekusi yang menghasilkan *output* *string* Dia adalah balita. Jika *statement* pertama *False*, maka *statement* kedua divalidasi dengan kondisi variabel umur lebih dari 5 and kurang dari sama dengan 12 dan menghasilkan nilai *True*. Maka *if statement* kedua akan dieksekusi yang menghasilkan *output* *string* Dia adalah anak-anak. Jika *statement* kedua *False*, maka *statement* ketiga divalidasi dengan kondisi variabel umur lebih dari 12 and kurang dari sama dengan 18 dan menghasilkan nilai *True*. Maka *if statement* kedua akan dieksekusi yang menghasilkan *output* *string* Dia adalah remaja. Jika tidak ada *statement* yang bernilai *True* maka *else statement* akan dieksekusi yang menghasilkan *output* *string* Dia adalah dewasa.

**

*Gambar 18. Hasil output dari jawaban tugas unguided 2*

Pada gambar 18 menunjukan beberapa contoh *output* jika pengguna memasukan *value* ke variabel nilai:

1. Jika pengguna memasukan *value* 4 ke variabel usia. Maka *if statement* pertama akan dieksekusi karena *value* variabel usia memenuhi kondisi pertama yang menghasilkan *output* *string* dia adalah balita.
2. Jika pengguna memasukan *value* 11 ke variabel usia. Maka *if statement* kedua akan dieksekusi karena *value* variabel usia memenuhi kondisi kedua yang menghasilkan *output* *string* dia adalah anak-anak.
3. Jika pengguna memasukan *value* 16 ke variabel usia. Maka *if statement* ketiga akan dieksekusi karena *value* variabel usia memenuhi kondisi ketiga yang menghasilkan *output* *string* dia adalah remaja.
4. Jika pengguna memasukan *value* 21 ke variabel usia. Maka *if statement* tidak akan dieksekusi karena *value* variabel usia tidak memenuhi kondisi yang ada, tetapi *else statement* akan dieksekusi yang menghasilkan *output* *string* Dia adalah remaja.
5. **Ringkasan**

Modul ini membahas dasar-dasar python tentang *selection*. *Selection* adalah suatu opsi dimana terdapat kondisi tertentu yang harus dipenuhi oleh program untuk menjalankan suatu perintah. Terdapat beberapa jenis selection didalam bahasa python seperti *if, if-else, if-else, if-elif-else*. *Selection* jiga dapat digunakan untuk membuat selection dengan kondisi lebih kompleks dengan menggunakan nested *statement* atau pernyataan bersarang.

*If statement* adalah *selection* yang dieksekusi jika *statement* memiliki kondisi *True*, tetapi jika *statement* memiliki kondisi *False* *statement* itu tidak akan di eksekusi. if-*else statement* adalah selection yang dieksekusi jika kondisi *True* dan jika kondisi *False* maka *else statement* akan dieksekusi. If-el*if statement* adalah *selection* jika kondisi pertama *True* maka kondisi pertama akan dieksekusi, jika kondisi pertama *False* maka kondisi kedua dengan kondisi *True* akan dikesekusi dan seterusnya, jika tidak ada ada kondisi yang terpenuhi atau *False* maka selection tidak akan dieksekusi. If-elif-else adalah *selection* if-elif yang tidak ada kondisi yang terpenuhi atau *False* maka *else statement* akan dieksekusi.

Laporan ini juga berisi tugas praktik yang mencakup implementasi selection pada bahasa python. Salah satu contoh latihan adalah pengguaan selection untuk mengetahu grade berdasarkan besaran *value* nilai dan ketersedian tiket beserta harganya sesuai dengan *value* variabel umur. Selain itu terdapat juga tugas mandiri untuk memahami lebih dalam tentang pengguaan selection dengan cara diberi studi kasus yang dikerjakan dengan mandiri. Salah satu studi kasusnya berupa menentukan grade nilai berdasarkan *value* variabel nilai yang dimasukkan oleh pengguna dan menebak apakah usia pengguna termasuk katergori balita, anak-anak, remaja, atau dewasa berdasarkan input usia oleh pengguna.

Dari laporan ini disimpulkan bahwa pemahaman konsep selection dalam Python sangat penting. Dengan menguasainya programmer dapat menuliskan kode sesuai dengan kondisi tertentu. Sehingga kode yang ditulis akan menghasilkan *output* sesuai degan kondisi yang telah ditentukan, lebih fleksibel, dan lebih mudah dipahami.