**LAPORAN PRATIKUM KE-3**

**PEMROGRAMAN**

****

**DISUSUN OLEH :**

Nama: Mario Andiko Dwi Saputra

NPM: G1A025015

Kelas: A1

**Nama Asisten Dosen :**

1. Davi Sulaiman (G1A022001)
2. Ade Irawan (G1A022083)
3. Reksi Hendra Pratama (G1A022032)
4. Lio Kusnata (G1A023013)
5. Lantera Meunasah (G1A024105)
6. Ahsan Nurfauzi (G1A024002)
7. Azzahra Faranisa (G1A023010)
8. Evelyn Unike Aritonang (G1A022024)
9. Gusti Revallon Putrada (G1A024081)
10. Julia Mayang Sari (G1A022010)

**Dosen Pengampu :**

1. Funny Farady Coastera, S.Kom, M.T
2. Arie Vatresia, S.T., M.TI., P.h.D..

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS BENGKULU**

**2025**

**LANDASAN TEORI**

Perkembangan teknologi mobile phone membuka peluang besar dalam pengembangan aplikasi edukatif yang dapat diakses secara fleksibel oleh pengguna. Salah satu aspek penting dalam pengembangan aplikasi pembelajaran adalah kemampuan program untuk mengambil keputusan secara logis, sehingga dapat menanggapi berbagai kondisi input dari pengguna dengan tepat. Afrizal (2017) menekankan bahwa dalam rancang bangun aplikasi pembelajaran dasar pemrograman berbasis mobile phone, struktur kontrol percabangan merupakan salah satu komponen fundamental yang harus dipahami dan diterapkan oleh pengembang agar aplikasi dapat berjalan secara dinamis dan interaktif.

Struktur kontrol percabangan memungkinkan program untuk mengeksekusi sekelompok perintah tertentu hanya ketika kondisi tertentu terpenuhi. Pernyataan if adalah bentuk dasar dari percabangan, di mana eksekusi blok kode dilakukan **hanya jika kondisi yang diuji bernilai benar (true)**. Konsep ini sangat berguna dalam membuat aplikasi yang responsif terhadap input pengguna, misalnya dalam aplikasi pembelajaran, ketika pengguna memasukkan jawaban pada soal, program dapat menilai jawaban tersebut dan memberikan feedback yang sesuai. Dengan adanya if, program dapat membedakan antara kondisi yang berbeda dan mengambil langkah yang tepat berdasarkan kondisi tersebut.

Selain pernyataan if, **struktur if-else** digunakan ketika program harus memilih antara dua jalur eksekusi. Jika kondisi dalam if bernilai benar, blok kode if dieksekusi; jika kondisi bernilai salah, blok kode dalam else dijalankan. Penggunaan if-else memperluas kemampuan program dalam menanggapi input yang bervariasi. Sebagai contoh, dalam aplikasi pembelajaran pemrograman, jika pengguna memasukkan jawaban benar, program menampilkan pesan “Selamat, jawaban Anda benar!”; namun jika jawaban salah, program menampilkan pesan “Coba lagi, jawaban salah.” Dengan demikian, if-else memungkinkan interaksi yang lebih informatif dan membuat pengalaman belajar menjadi lebih efektif.

Lebih lanjut, **percabangan bertingkat (if-else if-else)** memungkinkan program untuk mengevaluasi beberapa kondisi secara berurutan, sehingga dapat menangani berbagai kemungkinan dengan lebih kompleks. Dalam aplikasi pembelajaran, percabangan bertingkat dapat digunakan untuk menilai jawaban pengguna dalam beberapa kategori, misalnya jawaban tepat, hampir tepat, atau salah, sehingga feedback yang diberikan dapat lebih spesifik dan membantu proses pembelajaran. Afrizal (2017) menekankan bahwa penggunaan percabangan bertingkat sangat penting untuk meningkatkan kualitas aplikasi, terutama dalam memberikan pengalaman belajar yang mendekati kondisi nyata, di mana banyak faktor harus dipertimbangkan dalam pengambilan keputusan.

Penggunaan percabangan juga berperan penting dalam **membuat alur logika program yang jelas dan mudah dipahami**, baik bagi pengembang maupun pengguna akhir. Dalam konteks pembelajaran pemrograman, penerapan struktur if, if-else, dan if-else if-else membantu mahasiswa memahami konsep logika dasar yang menjadi fondasi dari pemrograman modern. Dengan memahami percabangan, mahasiswa dapat merancang program yang mampu menanggapi berbagai kondisi input, membuat keputusan yang tepat, dan memberikan output yang sesuai.

Afrizal (2017) juga menyoroti pentingnya pemahaman percabangan dalam konteks pengembangan aplikasi mobile phone yang bersifat interaktif. Struktur kontrol percabangan memungkinkan pengembang untuk menambahkan elemen **interaktivitas** dan **responsivitas** pada aplikasi. Misalnya, aplikasi pembelajaran dapat menyesuaikan materi yang ditampilkan berdasarkan jawaban sebelumnya atau kondisi tertentu, sehingga proses pembelajaran menjadi lebih adaptif dan personal bagi setiap pengguna. Hal ini menunjukkan bahwa percabangan bukan hanya sekadar konsep teknis, tetapi juga memiliki peran pedagogis yang signifikan dalam meningkatkan efektivitas pembelajaran berbasis aplikasi.

Secara keseluruhan, pemahaman dan penerapan struktur percabangan seperti if, if-else, dan if-else if-else merupakan landasan penting dalam pengembangan aplikasi pembelajaran dasar pemrograman. Dengan menerapkan konsep ini, pengembang dapat menciptakan aplikasi yang responsif, interaktif, dan memberikan pengalaman belajar yang efektif. Afrizal (2017) menekankan bahwa kemampuan untuk merancang alur logika program melalui percabangan adalah keterampilan dasar yang harus dimiliki oleh setiap mahasiswa atau programmer pemula, karena menjadi fondasi bagi pengembangan aplikasi yang lebih kompleks di masa depan.

Dalam pembelajaran pemrograman, kemampuan siswa untuk memahami logika kondisional menjadi aspek yang sangat penting. Percabangan merupakan salah satu konsep dasar yang memungkinkan program menyesuaikan jalannya eksekusi berdasarkan kondisi tertentu. Pambudi dan Setiawan (2025) menyatakan bahwa pengenalan percabangan melalui praktikum, seperti pembuatan game “Tebak Angka” berbasis Python, memberikan pengalaman belajar yang langsung dan interaktif. Dengan memanfaatkan percabangan, siswa dapat melihat bagaimana setiap keputusan yang mereka buat dalam kode akan memengaruhi hasil program secara nyata.

Percabangan juga memfasilitasi pengembangan kemampuan berpikir analitis. Dalam kegiatan praktikum, siswa belajar untuk mengevaluasi berbagai kemungkinan kondisi dan menentukan aksi yang sesuai untuk setiap kondisi. Proses ini melatih keterampilan berpikir logis dan sistematis, yang tidak hanya berguna dalam pemrograman, tetapi juga dalam pemecahan masalah pada konteks yang lebih luas. Selain itu, interaksi langsung dengan program membuat siswa lebih termotivasi untuk mengeksplorasi kode, mencoba skenario berbeda, dan memahami hubungan sebab-akibat dalam logika program.

Selain aspek teknis, penerapan percabangan dalam praktikum memiliki nilai pedagogis yang signifikan. Dengan menggunakan percabangan, siswa dapat belajar bagaimana merancang alur program yang logis, menilai kondisi, serta memberikan respons yang tepat terhadap input pengguna. Hal ini membantu siswa membangun fondasi logika pemrograman yang kuat, sekaligus menumbuhkan kemampuan berpikir kritis dan kemampuan mengambil keputusan yang terstruktur. Pambudi dan Setiawan (2025) menunjukkan bahwa praktikum berbasis game edukatif dapat menjadikan proses belajar lebih menarik dan partisipatif, karena siswa dapat melihat dampak langsung dari setiap kondisi yang mereka tulis dalam kode.

Selain itu, penerapan percabangan dalam konteks praktikum mendukung pembelajaran yang adaptif dan personal. Dengan percabangan, program dapat merespons input yang berbeda dengan cara yang sesuai, sehingga siswa mendapatkan feedback yang relevan dan spesifik. Hal ini membuat pembelajaran menjadi lebih interaktif dan membantu siswa memahami konsep pemrograman secara lebih mendalam. Integrasi percabangan dalam praktik ini tidak hanya meningkatkan keterampilan teknis, tetapi juga melatih kemampuan berpikir logis, analitis, dan kreatif.

Secara keseluruhan, penerapan percabangan dalam praktikum berbasis Python, seperti game “Tebak Angka”, memberikan kontribusi signifikan terhadap peningkatan literasi teknologi siswa, pengembangan pola pikir komputasional, serta kemampuan problem-solving. Hal ini menegaskan bahwa pemahaman dan penguasaan struktur percabangan merupakan fondasi penting dalam pembelajaran pemrograman dasar, sekaligus mempersiapkan siswa untuk menghadapi tantangan pemrograman yang lebih kompleks di masa mendatang.

Selain meningkatkan pemahaman teknis, penerapan percabangan dalam praktikum juga membantu siswa mengembangkan **strategi pemecahan masalah** yang efektif. Siswa belajar untuk mengantisipasi berbagai kemungkinan yang mungkin muncul dalam alur program dan menyiapkan kondisi yang sesuai agar program dapat berjalan dengan benar. Hal ini menumbuhkan kemampuan berpikir proaktif, di mana siswa tidak hanya menunggu kesalahan terjadi, tetapi juga merancang solusi sejak awal.

Percabangan memungkinkan siswa mengeksplorasi logika bersyarat kompleks dan kombinasi beberapa kondisi. Dengan memahami interaksi kondisi, mereka dapat merancang program lebih fleksibel, adaptif, serta meningkatkan kemampuan berpikir kritis dan sistematis.

**SOAL DAN PEMBAHASAN**

1. Ade merupakan seorang arsitek dan programmer yang mana ia akan membuat program sederhana untuk mengecek apakah sebuah ruangan bisa disebut “ruang kelas”.

Aturan:

Panjang ruangan minimal 5 meter.

Lebar ruangan minimal 4 meter.

Jika kedua syarat dipenuhi, hasilnya “Ruang kelas”. Jika tidak, “Bukan ruang kelas“.

Pernyataan:

Jika sebuah ruangan memiliki panjang = 6 dan lebar = 3, maka program akan mencetak“Ruang kelas”.

Apakah pernyataan ini benar atau salah?

Pembahasan:

jawabannya adalah salah,karena syarat untuk bisa dibilang “Ruang kelas” adalah if (panjang >= 5 && lebar >= 4) sedangkan lebarnya adalah 3, jadi otomatis jawabannya otomatis ke else yaitu “Bukan kelas”

2.Alicia seorang programmer di bidang IoT ia diminta oleh bu arie untuk membuat sistem lampu pintar.

Aturannya:

- Jika sensor mendeteksi orang masuk, lampu menyala.

- Jika tidak ada orang, lampu mati.

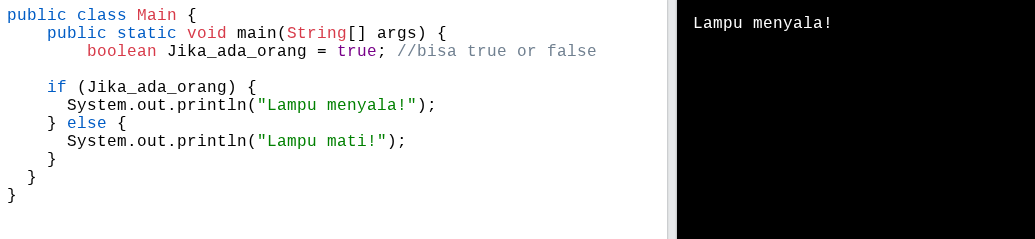
a. Buatlah kode program sesuai cerita di atas.

b. Jalankan program jika adaOrang = true.

c. Jalankan program jika adaOrang = false

Pembahasan:

a.**Printscreen Source Code**



G1A025015 G1A025015

**Source Code**

public class Main {

public static void main(String[] args) {

boolean Jika\_ada\_orang = true; //bisa true or false

if (Jika\_ada\_orang) {

System.out.println("Lampu menyala!");

} else {

System.out.println("Lampu mati!");

}

}

}

**Penjelasan Source Code**

"public class Main {"

-> Mendefinisikan class utama bernama Main. Nama class harus sama dengan nama file Java.

"public static void main(String[] args) {"

-> Titik awal program dijalankan. Semua program Java dieksekusi dari method main.

"boolean Jika\_ada\_orang = true;"

-> Mendeklarasikan variabel bertipe boolean. Bisa bernilai true (ada orang) atau false (tidak ada orang).

"if (Jika\_ada\_orang) {"

-> Mengecek kondisi. Jika variabel Jika\_ada\_orang bernilai true, blok ini dijalankan.

"System.out.println(\"Lampu menyala!\");"

-> Menampilkan pesan "Lampu menyala!" di layar jika kondisi true.

"} else { ... System.out.println(\"Lampu mati!\"); }"

-> Jika kondisi false, maka mengeksekusi blok else dan menampilkan "Lampu mati!".

"}"

-> Menutup method main.

"}"

-> Menutup class Main.

b.**Printscreen Source Code**



G1A025015 G1A025015

**Printscreen Output**

****

G1A025015 G1A025015

c.**Printscreen Source Code**



G1A025015 G1A025015

**Printscreen Output**



G1A025015 G1A025015

3. Bencoolen mall memiliki aturan tarif parkir yang berbeda sesuai waktu kendaraan masuk. Sistem parkir otomatis harus mengikuti ketentuan berikut:

- Jika kendaraan masuk sebelum jam 12 siang, maka sistem akan menampilkan “Pagi, tarif Rp5000”.

- Jika kendaraan masuk antara jam 12 siang hingga sebelum jam 18 sore, maka sistem akan

menampilkan “Siang, tarif Rp 7000”.

- Jika kendaraan masuk jam 18 sore atau lebih, maka sistem akan menampilkan “Malam, tarif Rp 10000”.

Cahaya merupakan programmer yang dipilih oleh pihak Bencoolen Mall, ia diminta untuk membuat program berdasarkan aturan tersebut.

Pertanyaan:

a. Buatlah kode program menggunakan else if untuk menerapkan aturan parkir mall tersebut.

b. Jalankan program dengan input jam = 10 (kendaraan masuk pagi hari).

c. Jalankan program dengan input jam = 20 (kendaraan masuk malam hari).

Pembahasan:

**Printscreen Source Code**

**Source Code**

public class Main {

public static void main(String[] args) {

int Tarif\_parkir = 1; //keterangan jam = pagi, siang, malam

if (Tarif\_parkir < 12) {

System.out.println("siang");

System.out.println("Rp.5000");

} else if (Tarif\_parkir > 12 && Tarif\_parkir < 18) {

System.out.println("sore");

System.out.println("Rp.7.000");

} else if (Tarif\_parkir > 18) {

System.out.println("malam");

System.out.println("Rp.10.000.");

}

}

}

**Penjelasan Source Code**

public class Main

→ mendefinisikan class utama bernama Main.

public static void main(String[] args)

→ titik awal program dijalankan.

int Tarif\_parkir = 1

→ variabel yang menyimpan jam parkir.

if (Tarif\_parkir < 12)

→ jika jam parkir pagi/siang, tampilkan “siang” + Rp.5000.

else if (Tarif\_parkir > 12 && Tarif\_parkir < 18)

→ jika jam parkir sore, tampilkan “sore” + Rp.7.000.

else if (Tarif\_parkir > 18) → jika jam parkir malam, tampilkan “malam” + Rp.10.000.

4. Rumah sakit Tiara Sella memiliki sebuah lift dengan sistem otomatis untuk mengecek jumlah orang di dalamnya. Sistem tersebut bekerja dengan aturan:

Jika jumlah orang ≤ 8, maka layar menampilkan: “Lift bisa jalan”.

Jika jumlah orang > 8, maka layar menampilkan: “Terlalu penuh, tunggu lift berikutnya”.

Mario diminta membuat program sederhana untuk menguji aturan tersebut.

Pertanyaan:

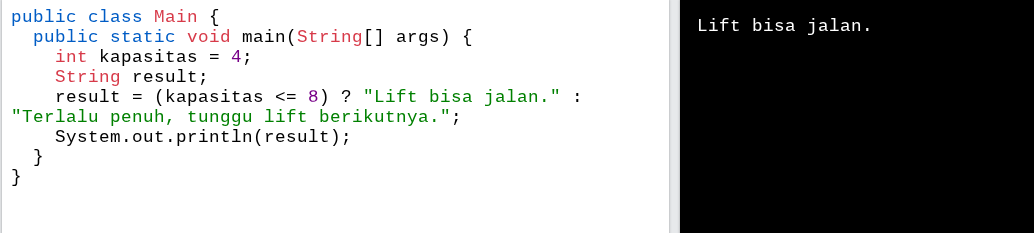
a. Buatlah kode program menggunakan ternary operator untuk aturan lift tersebut.

b. Uji program dengan jumlahOrang = 6 (bayangkan kondisi lift pagi hari saat belum terlalu ramai).

c. Uji program dengan jumlahOrang = 10 (bayangkan kondisi lift siang hari saat penuh sesak).

Pembahasan:

**Printscreen Source Code**



G1A025015 G1A025015

**Source Code**

public class Main {

public static void main(String[] args) {

int kapasitas = 4;

String result;

result = (kapasitas <= 8) ? "lift bisa jalan." : "Terlalu penuh,tunggu lift selanjutnya.";

System.out.println(result);

}

}

**Penjelasan Source Code**

public class Main

→ mendefinisikan class utama bernama Main.

public static void main(String[] args)

→ titik awal program dijalankan.

int Tarif\_parkir = 1

→ variabel yang menyimpan jam parkir.

if (Tarif\_parkir < 12)

→ jika jam parkir pagi/siang, tampilkan “siang” + Rp.5.000.

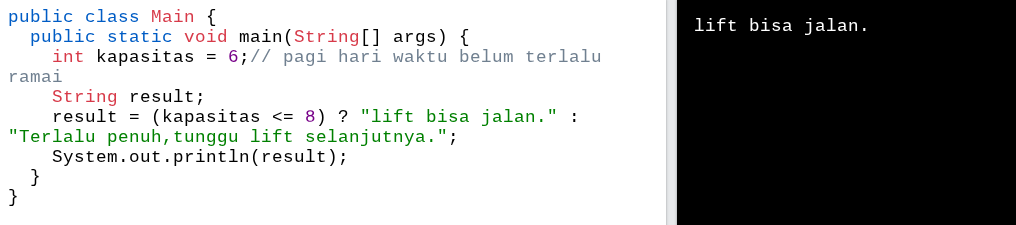
else if (Tarif\_parkir > 12 && Tarif\_parkir < 18)

→ jika jam parkir sore, tampilkan “sore” + Rp.7.000.

else if (Tarif\_parkir > 18)

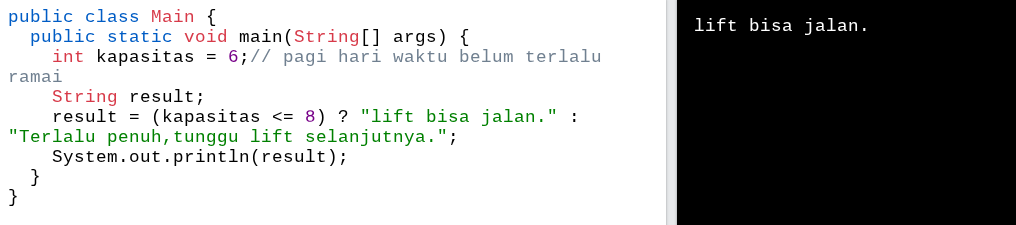
→ jika jam parkir malam, tampilkan “malam” + Rp.10.000.

b.**Printscreen Source Code**



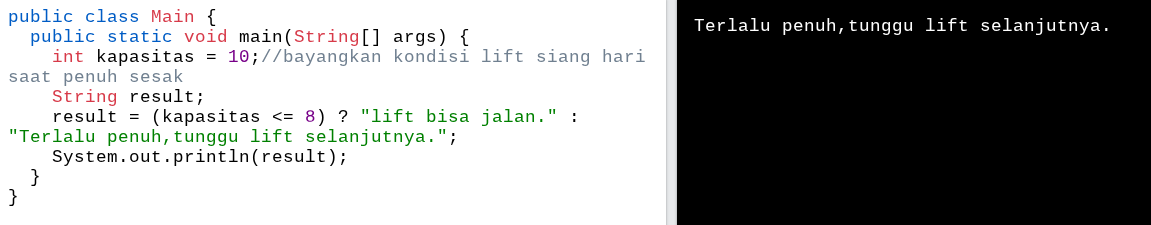
G1A025015 G1A025015

**Printscreen Output**

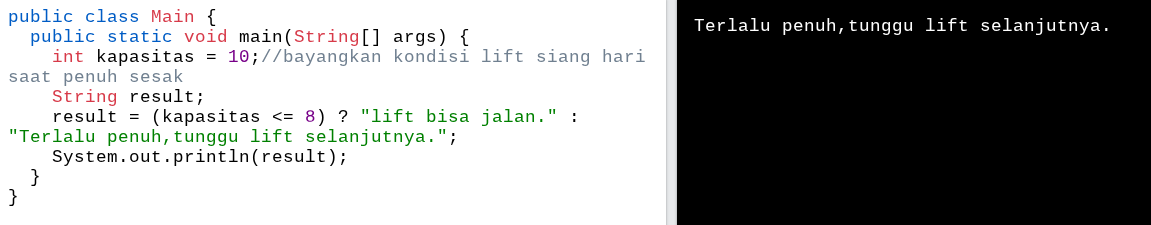


G1A025015 G1A025015

c.**Printscreen Source Code**

****

**Printscreen Output**

****

G1A025015 G1A025015

5.Rifqi dan Zora adalah mahasiswa yang ingin mengakses koleksi e-book di perpustakaan digital kampus.

Namun, sistem memiliki aturan akses sebagai berikut:

Jika mahasiswa sudah terdaftar sebagai anggota, maka dicek lagi:

Jika ia sudah membayar iuran, tampilkan: “Akses e-book dibuka”.

Jika ia belum membayar iuran, tampilkan: “Harap bayar iuran dulu”.

Jika mahasiswa bukan anggota, tampilkan: “Silakan daftar anggota dahulu”.

Mahasiswa tersebut kemudian meminta bantuan temannya yang belajar pemrograman untuk membuat simulasi aturan ini.

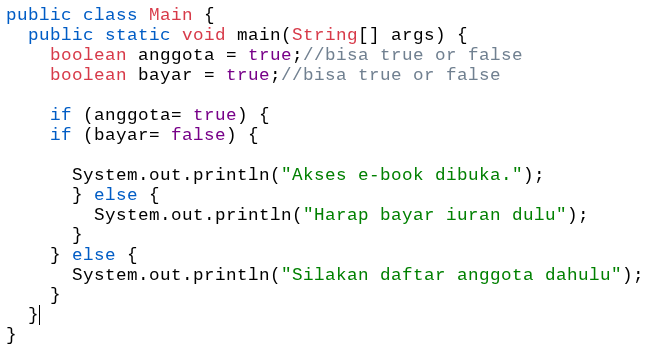
Pertanyaan:

a. Buatlah kode program menggunakan nested if untuk aturan akses e-book di atas.

b. Uji program dengan kondisi: anggota = true, bayar = true.

c. Uji program dengan kondisi: anggota = true, bayar = false.

a.**Printscreen Source Code**



G1A025015 G1A025015

**Source Code**

public class Main {

public static void main(String[] args) {

boolean anggota = true;//bisa true or false

boolean bayar = true;//bisa true or false

if (anggota= true) {

if (bayar= false) {

System.out.println("Akses e-book dibuka.");

} else {

System.out.println("Harap bayar iuran dulu");

}

} else {

System.out.println("Silakan daftar anggota dahulu");

}

}

}

**Source Code**

public class Main

→ mendefinisikan class utama bernama Main.

public static void main(String[] args)

→ titik awal program dijalankan.

boolean anggota = true;

→ variabel boolean untuk menyimpan status anggota (true atau false).

boolean bayar = true;

→ variabel boolean untuk menyimpan status pembayaran iuran (true atau false).

if (anggota)

→ cek apakah mahasiswa sudah terdaftar sebagai anggota.

if (bayar)

→ nested if: jika mahasiswa anggota, cek apakah sudah membayar iuran.

System.out.println("Akses e-book dibuka.");

→ tampilkan pesan jika mahasiswa sudah anggota dan membayar iuran.

else

→ jika mahasiswa anggota tapi belum membayar iuran.

System.out.println("Harap bayar iuran dulu");

→ tampilkan pesan agar mahasiswa membayar iuran terlebih dahulu.

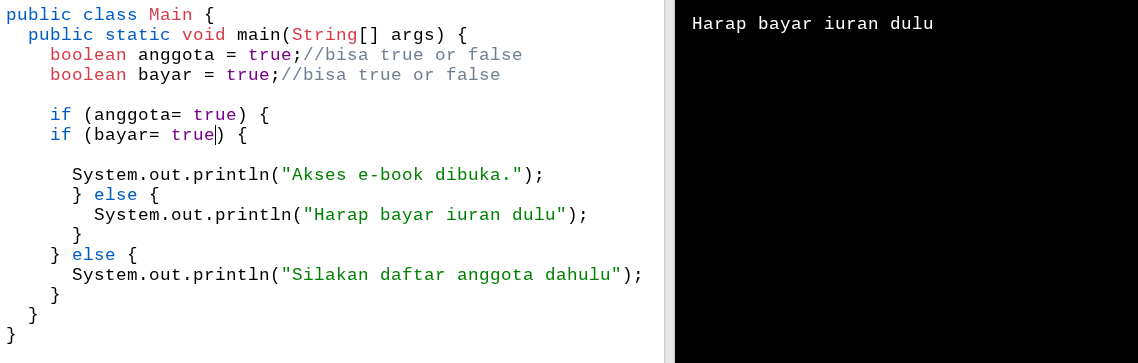
else

→ jika mahasiswa bukan anggota.

System.out.println("Silakan daftar anggota dahulu");

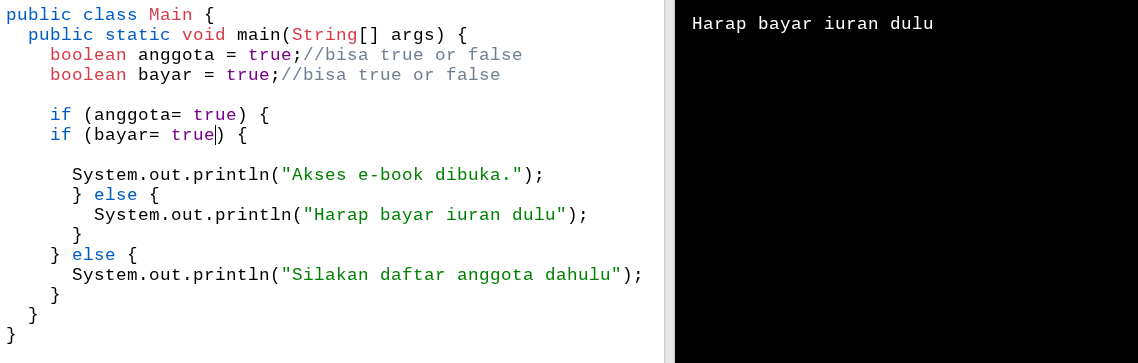
→ tampilkan pesan agar mahasiswa mendaftar sebagai anggota.

b.**Printscreen Source Code**



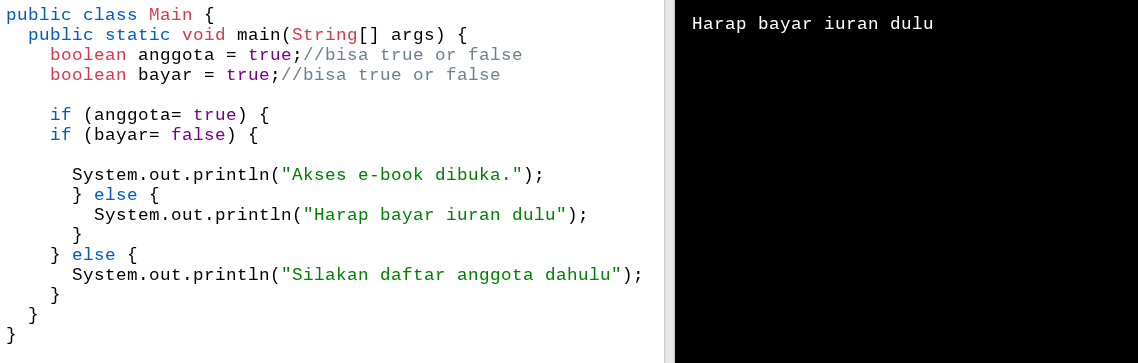
G1A025015 G1A025015

**Printscreen Output**



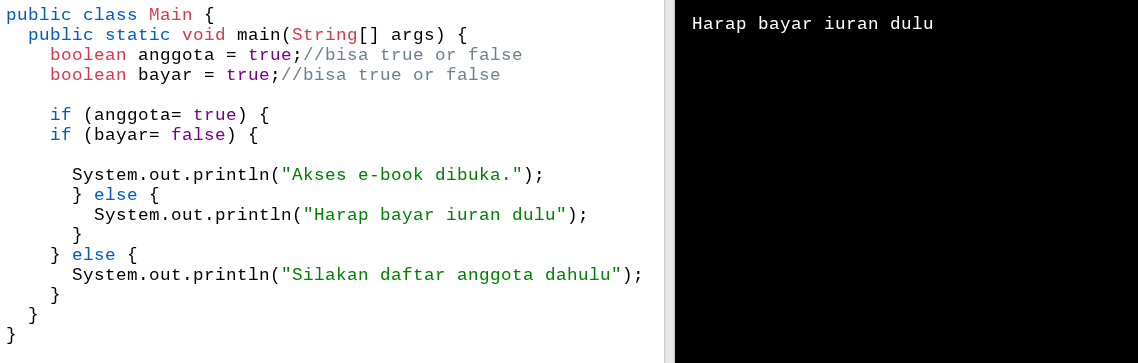
G1A025015 G1A025015

c.**Printscreen Source Code**

****

G1A025015 G1A025015

**Printscreen Output**

****

G1A025015 G1A025015

6. Sebuah drone harus memilih arah berdasarkan kondisi:

Jika angin datang dari utara (drone terbang ke selatan).

Jika angin datang dari barat (drone terbang ke timur).

Selain itu (drone hover atau diam di tempat).

Lengkapi kode berikut agar sesuai aturan:

Maka jawablah soal berikut

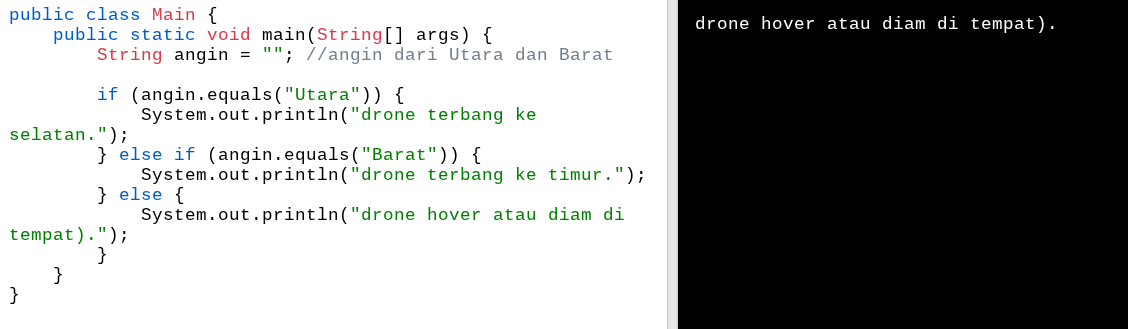
a. Lengkapi bagian yang kosong.

b. Uji dengan angin = “barat”.

c. Uji dengan angin = “selatan”.

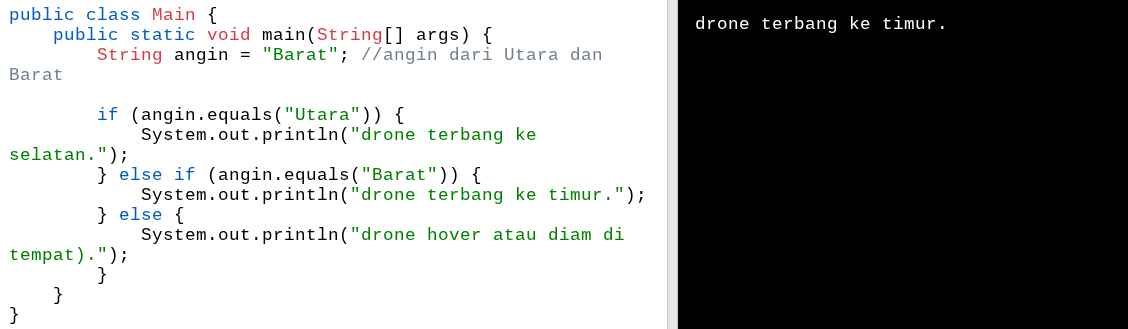
Pembahasan:

a.**Printscreen Source Code**



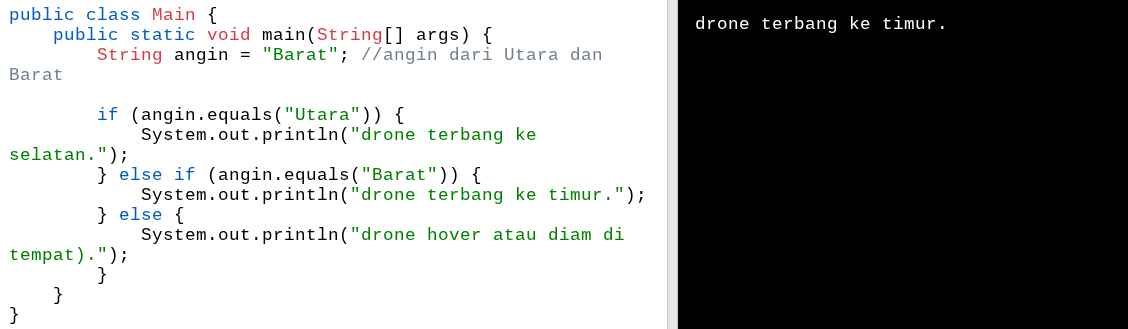
G1A025015 G1A025015

b.**Printscreen Source Code**



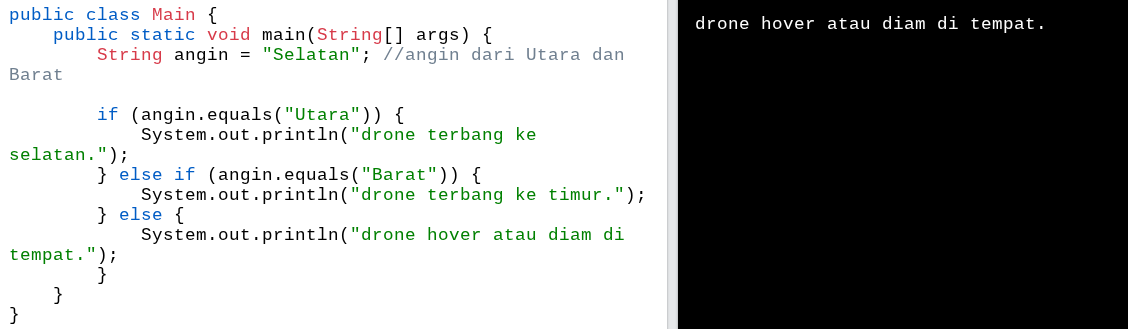
G1A025015 G1A025015

**Printscreen Output**



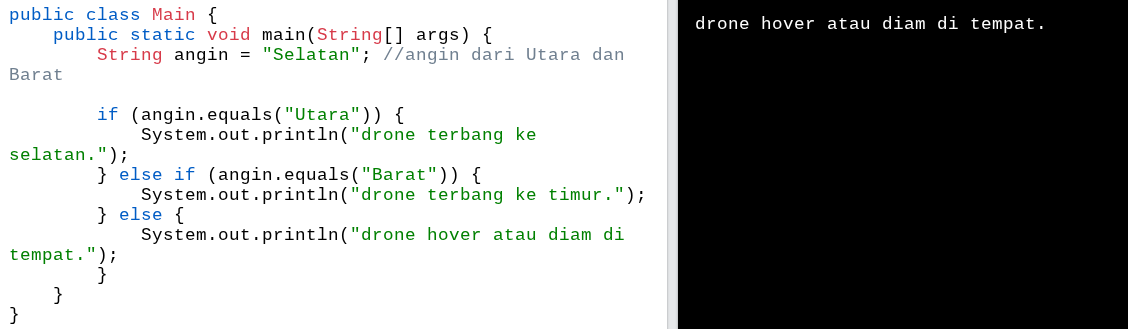
G1A025015 G1A025015

c.**Printscreen Source Code**

****

G1A025015 G1A025015

**Printscreen Output**

****

G1A025015 G1A025015

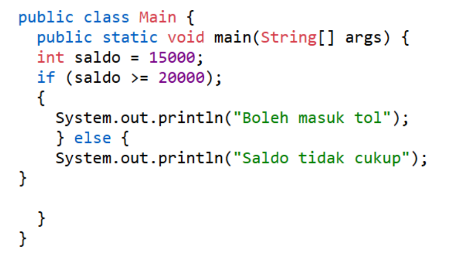
7. Melzaqi menulis program untuk mengecek apakah mobil boleh masuk jalan tol.

Aturannya:

Jika saldo e-Toll ≥ 20.000 → boleh masuk.

Jika tidak, tidak boleh masuk.

Kode Melzaqi:



G1A025015 G1A025015

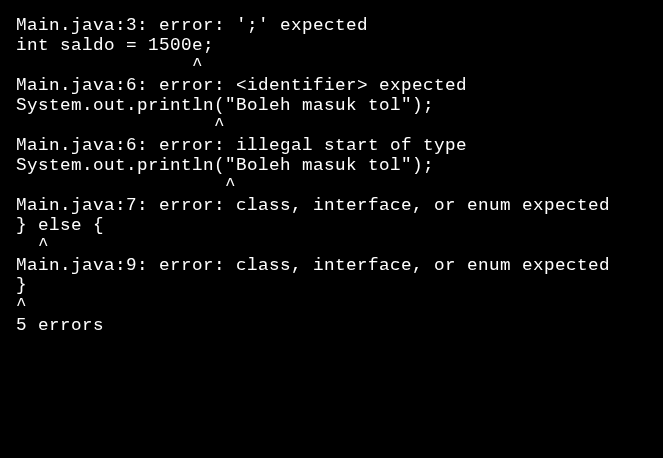
Maka,

a. Apa output yang didapatkan jika ada kesalahan tunjukkan dimana kesalahan pada kode program tersebut.

b. Perbaiki kode program tersebut.

Pembahasan:

**Printscreen Output**



G1A025015 G1A025015

masalah:

1.1500e bukan angka yang valid untuk tipe int

2.Method main ditutup di baris ke-4 (}), jadi semua statement setelahnya (if dan System.out.println) tidak berada di dalam method.

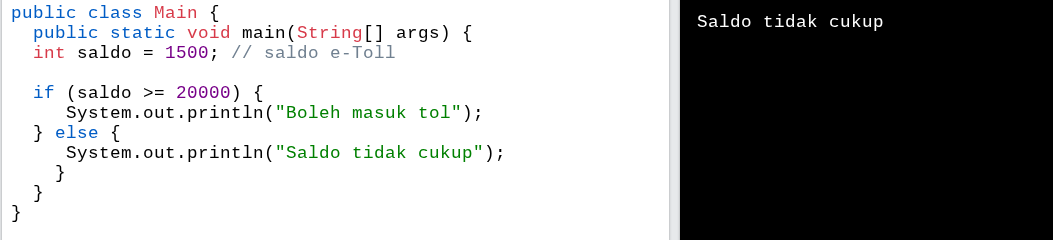
3.println berada di tempat yang salah → bukan bagian dari method → error.

4.else muncul di luar if, karena {} sebelumnya sudah ditutup.

5.System.out.println("Saldo tidak cukup");

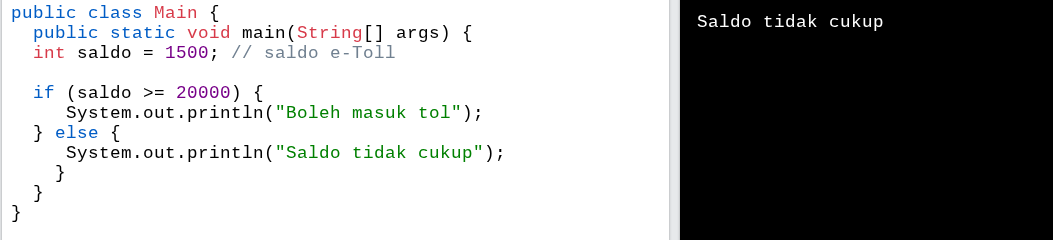
Masih berada di luar method/class, sehingga compiler menganggapnya bukan bagian dari program → error class, interface, or enum expected.

b.**Printscreen Source Code**



G1A025015 G1A025015

**Printscreen Output**



G1A025015 G1A025015

**KESIMPULAN DAN SARAN**

KESIMPULAN  
 Berdasarkan hasil praktikum dan kajian teori, dapat disimpulkan bahwa struktur percabangan seperti if, if-else, dan if-else if-else memiliki peran sangat penting dalam pemrograman dasar. Percabangan memungkinkan program menyesuaikan jalannya eksekusi berdasarkan kondisi tertentu, memberikan respons yang tepat terhadap input pengguna, serta membuat program menjadi lebih interaktif, adaptif, dan mudah dipahami. Dalam kegiatan praktikum, baik melalui pembuatan game berbasis Python maupun pengembangan aplikasi pembelajaran berbasis mobile phone, percabangan membantu siswa memahami logika kondisional, mengevaluasi kondisi yang berbeda, serta menentukan aksi yang tepat untuk setiap skenario. Dengan demikian, siswa dapat melihat secara langsung dampak dari setiap keputusan logis yang mereka buat, sehingga pemahaman terhadap alur program menjadi lebih nyata dan konkret.

Selain aspek teknis, penerapan percabangan juga mendorong pengembangan kemampuan berpikir analitis, logis, dan sistematis. Siswa belajar memecah masalah menjadi beberapa kondisi, menganalisis berbagai kemungkinan, dan mengambil keputusan yang sesuai untuk masing-masing kondisi. Proses ini melatih pola pikir komputasional, kemampuan problem-solving, serta keterampilan berpikir kritis yang berguna tidak hanya dalam pemrograman, tetapi juga dalam pemecahan masalah pada konteks lain. Praktikum berbasis game atau aplikasi edukatif juga meningkatkan motivasi siswa untuk bereksperimen dengan kode, mencoba berbagai skenario, dan memahami hubungan sebab-akibat dalam logika program, sehingga pengalaman belajar menjadi lebih interaktif, menyenangkan, dan mendalam.

SARAN  
 Disarankan agar praktikum pemrograman selalu menekankan latihan yang melibatkan percabangan dan skenario nyata, sehingga siswa memperoleh pengalaman langsung dalam merancang program yang interaktif, adaptif, dan efektif. Praktikum sebaiknya dilengkapi dengan diskusi, evaluasi hasil, serta refleksi terhadap kesalahan, agar siswa dapat memahami penyebab kesalahan dan belajar memperbaikinya. Pengajar juga dianjurkan mendorong siswa mencoba berbagai kombinasi kondisi, percabangan bertingkat, serta simulasi yang relevan, sehingga kemampuan berpikir logis, analitis, kreatif, dan sistematis berkembang secara optimal. Dengan pendekatan ini, siswa tidak hanya menguasai keterampilan teknis menulis kode, tetapi juga mampu mengaplikasikan logika pemrograman secara adaptif dan kreatif dalam berbagai situasi pembelajaran, sekaligus membangun dasar yang kuat untuk memahami konsep pemrograman yang lebih kompleks di masa depan.

**DAFTAR PUSTAKA**

Afrizal, A. S. (2017). Rancang Bangun Aplikasi Pembelajaran Dasar Pemrograman Berbasis Mobile Phone. *Jurnal TIPS: Jurnal Teknologi Informasi dan Komputer Politeknik Sekayu*, *6*(1), 1-19.

Pambudi, S., & Setiawan, A. (2025). Pelatihan Pemrograman Dasar Python:: Meningkatkan Literasi Teknologi Siswa melalui Pembuatan Game Tebak Angka di SMAN 1 Pamotan. *Indonesian Research Journal on Education*, *5*(1), 7-12.

Sagala, A. A. H., & Yahfizham, Y. (2024). Analisis Pengenalan Konsep Algoritma Pemrograman Matematika Pada Kehidupan Sehari Hari. *Morfologi: Jurnal Ilmu Pendidikan, Bahasa, Sastra dan Budaya*, *2*(1), 01-16.

**LANDASAN TEORI**

Perkembangan teknologi mobile phone membuka peluang besar dalam pengembangan aplikasi edukatif yang dapat diakses secara fleksibel oleh pengguna. Salah satu aspek penting dalam pengembangan aplikasi pembelajaran adalah kemampuan program untuk mengambil keputusan secara logis, sehingga dapat menanggapi berbagai kondisi input dari pengguna dengan tepat. Afrizal (2017) menekankan bahwa dalam rancang bangun aplikasi pembelajaran dasar pemrograman berbasis mobile phone, struktur kontrol percabangan merupakan salah satu komponen fundamental yang harus dipahami dan diterapkan oleh pengembang agar aplikasi dapat berjalan secara dinamis dan interaktif.

Struktur kontrol percabangan memungkinkan program untuk mengeksekusi sekelompok perintah tertentu hanya ketika kondisi tertentu terpenuhi. Pernyataan if adalah bentuk dasar dari percabangan, di mana eksekusi blok kode dilakukan **hanya jika kondisi yang diuji bernilai benar (true)**. Konsep ini sangat berguna dalam membuat aplikasi yang responsif terhadap input pengguna, misalnya dalam aplikasi pembelajaran, ketika pengguna memasukkan jawaban pada soal, program dapat menilai jawaban tersebut dan memberikan feedback yang sesuai. Dengan adanya if, program dapat membedakan antara kondisi yang berbeda dan mengambil langkah yang tepat berdasarkan kondisi tersebut.

Selain pernyataan if, **struktur if-else** digunakan ketika program harus memilih antara dua jalur eksekusi. Jika kondisi dalam if bernilai benar, blok kode if dieksekusi; jika kondisi bernilai salah, blok kode dalam else dijalankan. Penggunaan if-else memperluas kemampuan program dalam menanggapi input yang bervariasi. Sebagai contoh, dalam aplikasi pembelajaran pemrograman, jika pengguna memasukkan jawaban benar, program menampilkan pesan “Selamat, jawaban Anda benar!”; namun jika jawaban salah, program menampilkan pesan “Coba lagi, jawaban salah.” Dengan demikian, if-else memungkinkan interaksi yang lebih informatif dan membuat pengalaman belajar menjadi lebih efektif.

Lebih lanjut, **percabangan bertingkat (if-else if-else)** memungkinkan program untuk mengevaluasi beberapa kondisi secara berurutan, sehingga dapat menangani berbagai kemungkinan dengan lebih kompleks. Dalam aplikasi pembelajaran, percabangan bertingkat dapat digunakan untuk menilai jawaban pengguna dalam beberapa kategori, misalnya jawaban tepat, hampir tepat, atau salah, sehingga feedback yang diberikan dapat lebih spesifik dan membantu proses pembelajaran. Afrizal (2017) menekankan bahwa penggunaan percabangan bertingkat sangat penting untuk meningkatkan kualitas aplikasi, terutama dalam memberikan pengalaman belajar yang mendekati kondisi nyata, di mana banyak faktor harus dipertimbangkan dalam pengambilan keputusan.

Penggunaan percabangan juga berperan penting dalam **membuat alur logika program yang jelas dan mudah dipahami**, baik bagi pengembang maupun pengguna akhir. Dalam konteks pembelajaran pemrograman, penerapan struktur if, if-else, dan if-else if-else membantu mahasiswa memahami konsep logika dasar yang menjadi fondasi dari pemrograman modern. Dengan memahami percabangan, mahasiswa dapat merancang program yang mampu menanggapi berbagai kondisi input, membuat keputusan yang tepat, dan memberikan output yang sesuai.

Afrizal (2017) juga menyoroti pentingnya pemahaman percabangan dalam konteks pengembangan aplikasi mobile phone yang bersifat interaktif. Struktur kontrol percabangan memungkinkan pengembang untuk menambahkan elemen **interaktivitas** dan **responsivitas** pada aplikasi. Misalnya, aplikasi pembelajaran dapat menyesuaikan materi yang ditampilkan berdasarkan jawaban sebelumnya atau kondisi tertentu, sehingga proses pembelajaran menjadi lebih adaptif dan personal bagi setiap pengguna. Hal ini menunjukkan bahwa percabangan bukan hanya sekadar konsep teknis, tetapi juga memiliki peran pedagogis yang signifikan dalam meningkatkan efektivitas pembelajaran berbasis aplikasi.

Secara keseluruhan, pemahaman dan penerapan struktur percabangan seperti if, if-else, dan if-else if-else merupakan landasan penting dalam pengembangan aplikasi pembelajaran dasar pemrograman. Dengan menerapkan konsep ini, pengembang dapat menciptakan aplikasi yang responsif, interaktif, dan memberikan pengalaman belajar yang efektif. Afrizal (2017) menekankan bahwa kemampuan untuk merancang alur logika program melalui percabangan adalah keterampilan dasar yang harus dimiliki oleh setiap mahasiswa atau programmer pemula, karena menjadi fondasi bagi pengembangan aplikasi yang lebih kompleks di masa depan.

Dalam pembelajaran pemrograman, kemampuan siswa untuk memahami logika kondisional menjadi aspek yang sangat penting. Percabangan merupakan salah satu konsep dasar yang memungkinkan program menyesuaikan jalannya eksekusi berdasarkan kondisi tertentu. Pambudi dan Setiawan (2025) menyatakan bahwa pengenalan percabangan melalui praktikum, seperti pembuatan game “Tebak Angka” berbasis Python, memberikan pengalaman belajar yang langsung dan interaktif. Dengan memanfaatkan percabangan, siswa dapat melihat bagaimana setiap keputusan yang mereka buat dalam kode akan memengaruhi hasil program secara nyata.

Percabangan juga memfasilitasi pengembangan kemampuan berpikir analitis. Dalam kegiatan praktikum, siswa belajar untuk mengevaluasi berbagai kemungkinan kondisi dan menentukan aksi yang sesuai untuk setiap kondisi. Proses ini melatih keterampilan berpikir logis dan sistematis, yang tidak hanya berguna dalam pemrograman, tetapi juga dalam pemecahan masalah pada konteks yang lebih luas. Selain itu, interaksi langsung dengan program membuat siswa lebih termotivasi untuk mengeksplorasi kode, mencoba skenario berbeda, dan memahami hubungan sebab-akibat dalam logika program.

Selain aspek teknis, penerapan percabangan dalam praktikum memiliki nilai pedagogis yang signifikan. Dengan menggunakan percabangan, siswa dapat belajar bagaimana merancang alur program yang logis, menilai kondisi, serta memberikan respons yang tepat terhadap input pengguna. Hal ini membantu siswa membangun fondasi logika pemrograman yang kuat, sekaligus menumbuhkan kemampuan berpikir kritis dan kemampuan mengambil keputusan yang terstruktur. Pambudi dan Setiawan (2025) menunjukkan bahwa praktikum berbasis game edukatif dapat menjadikan proses belajar lebih menarik dan partisipatif, karena siswa dapat melihat dampak langsung dari setiap kondisi yang mereka tulis dalam kode.

Selain itu, penerapan percabangan dalam konteks praktikum mendukung pembelajaran yang adaptif dan personal. Dengan percabangan, program dapat merespons input yang berbeda dengan cara yang sesuai, sehingga siswa mendapatkan feedback yang relevan dan spesifik. Hal ini membuat pembelajaran menjadi lebih interaktif dan membantu siswa memahami konsep pemrograman secara lebih mendalam. Integrasi percabangan dalam praktik ini tidak hanya meningkatkan keterampilan teknis, tetapi juga melatih kemampuan berpikir logis, analitis, dan kreatif.

Secara keseluruhan, penerapan percabangan dalam praktikum berbasis Python, seperti game “Tebak Angka”, memberikan kontribusi signifikan terhadap peningkatan literasi teknologi siswa, pengembangan pola pikir komputasional, serta kemampuan problem-solving. Hal ini menegaskan bahwa pemahaman dan penguasaan struktur percabangan merupakan fondasi penting dalam pembelajaran pemrograman dasar, sekaligus mempersiapkan siswa untuk menghadapi tantangan pemrograman yang lebih kompleks di masa mendatang.

Selain meningkatkan pemahaman teknis, penerapan percabangan dalam praktikum juga membantu siswa mengembangkan **strategi pemecahan masalah** yang efektif. Siswa belajar untuk mengantisipasi berbagai kemungkinan yang mungkin muncul dalam alur program dan menyiapkan kondisi yang sesuai agar program dapat berjalan dengan benar. Hal ini menumbuhkan kemampuan berpikir proaktif, di mana siswa tidak hanya menunggu kesalahan terjadi, tetapi juga merancang solusi sejak awal.

Percabangan memungkinkan siswa mengeksplorasi logika bersyarat kompleks dan kombinasi beberapa kondisi. Dengan memahami interaksi kondisi, mereka dapat merancang program lebih fleksibel, adaptif, serta meningkatkan kemampuan berpikir kritis dan sistematis.

**SOAL DAN PEMBAHASAN**

1. Ade merupakan seorang arsitek dan programmer yang mana ia akan membuat program sederhana untuk mengecek apakah sebuah ruangan bisa disebut “ruang kelas”.

Aturan:

Panjang ruangan minimal 5 meter.

Lebar ruangan minimal 4 meter.

Jika kedua syarat dipenuhi, hasilnya “Ruang kelas”. Jika tidak, “Bukan ruang kelas“.

Pernyataan:

Jika sebuah ruangan memiliki panjang = 6 dan lebar = 3, maka program akan mencetak“Ruang kelas”.

Apakah pernyataan ini benar atau salah?

Pembahasan:

jawabannya adalah salah,karena syarat untuk bisa dibilang “Ruang kelas” adalah if (panjang >= 5 && lebar >= 4) sedangkan lebarnya adalah 3, jadi otomatis jawabannya otomatis ke else yaitu “Bukan kelas”

2.Alicia seorang programmer di bidang IoT ia diminta oleh bu arie untuk membuat sistem lampu pintar.

Aturannya:

- Jika sensor mendeteksi orang masuk, lampu menyala.

- Jika tidak ada orang, lampu mati.

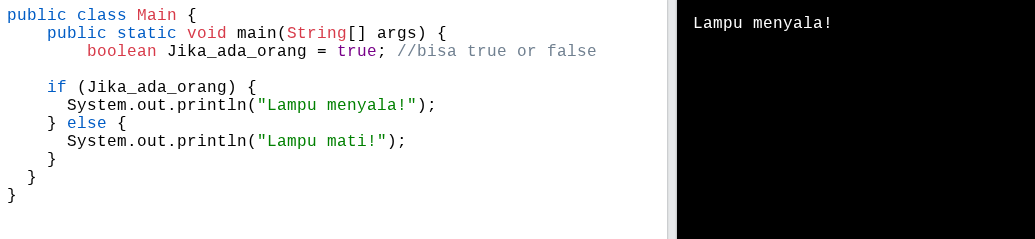
a. Buatlah kode program sesuai cerita di atas.

b. Jalankan program jika adaOrang = true.

c. Jalankan program jika adaOrang = false

Pembahasan:

a.**Printscreen Source Code**



**Source Code**

public class Main {

public static void main(String[] args) {

boolean Jika\_ada\_orang = true; //bisa true or false

if (Jika\_ada\_orang) {

System.out.println("Lampu menyala!");

} else {

System.out.println("Lampu mati!");

}

}

}

**Penjelasan Source Code**

"public class Main {"

-> Mendefinisikan class utama bernama Main. Nama class harus sama dengan nama file Java.

"public static void main(String[] args) {"

-> Titik awal program dijalankan. Semua program Java dieksekusi dari method main.

"boolean Jika\_ada\_orang = true;"

-> Mendeklarasikan variabel bertipe boolean. Bisa bernilai true (ada orang) atau false (tidak ada orang).

"if (Jika\_ada\_orang) {"

-> Mengecek kondisi. Jika variabel Jika\_ada\_orang bernilai true, blok ini dijalankan.

"System.out.println(\"Lampu menyala!\");"

-> Menampilkan pesan "Lampu menyala!" di layar jika kondisi true.

"} else { ... System.out.println(\"Lampu mati!\"); }"

-> Jika kondisi false, maka mengeksekusi blok else dan menampilkan "Lampu mati!".

"}"

-> Menutup method main.

"}"

-> Menutup class Main.

b.**Printscreen Source Code**



**Printscreen Output**

****

c.**Printscreen Source Code**



**Printscreen Output**



3. Bencoolen mall memiliki aturan tarif parkir yang berbeda sesuai waktu kendaraan masuk. Sistem parkir otomatis harus mengikuti ketentuan berikut:

- Jika kendaraan masuk sebelum jam 12 siang, maka sistem akan menampilkan “Pagi, tarif Rp5000”.

- Jika kendaraan masuk antara jam 12 siang hingga sebelum jam 18 sore, maka sistem akan

menampilkan “Siang, tarif Rp 7000”.

- Jika kendaraan masuk jam 18 sore atau lebih, maka sistem akan menampilkan “Malam, tarif Rp 10000”.

Cahaya merupakan programmer yang dipilih oleh pihak Bencoolen Mall, ia diminta untuk membuat program berdasarkan aturan tersebut.

Pertanyaan:

a. Buatlah kode program menggunakan else if untuk menerapkan aturan parkir mall tersebut.

b. Jalankan program dengan input jam = 10 (kendaraan masuk pagi hari).

c. Jalankan program dengan input jam = 20 (kendaraan masuk malam hari).

Pembahasan:

**Printscreen Source Code**

**Source Code**

public class Main {

public static void main(String[] args) {

int Tarif\_parkir = 1; //keterangan jam = pagi, siang, malam

if (Tarif\_parkir < 12) {

System.out.println("siang");

System.out.println("Rp.5000");

} else if (Tarif\_parkir > 12 && Tarif\_parkir < 18) {

System.out.println("sore");

System.out.println("Rp.7.000");

} else if (Tarif\_parkir > 18) {

System.out.println("malam");

System.out.println("Rp.10.000.");

}

}

}

**Penjelasan Source Code**

public class Main

→ mendefinisikan class utama bernama Main.

public static void main(String[] args)

→ titik awal program dijalankan.

int Tarif\_parkir = 1

→ variabel yang menyimpan jam parkir.

if (Tarif\_parkir < 12)

→ jika jam parkir pagi/siang, tampilkan “siang” + Rp.5000.

else if (Tarif\_parkir > 12 && Tarif\_parkir < 18)

→ jika jam parkir sore, tampilkan “sore” + Rp.7.000.

else if (Tarif\_parkir > 18) → jika jam parkir malam, tampilkan “malam” + Rp.10.000.

4. Rumah sakit Tiara Sella memiliki sebuah lift dengan sistem otomatis untuk mengecek jumlah orang di dalamnya. Sistem tersebut bekerja dengan aturan:

Jika jumlah orang ≤ 8, maka layar menampilkan: “Lift bisa jalan”.

Jika jumlah orang > 8, maka layar menampilkan: “Terlalu penuh, tunggu lift berikutnya”.

Mario diminta membuat program sederhana untuk menguji aturan tersebut.

Pertanyaan:

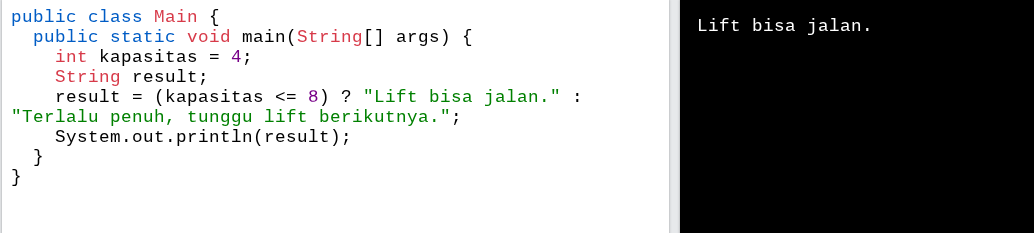
a. Buatlah kode program menggunakan ternary operator untuk aturan lift tersebut.

b. Uji program dengan jumlahOrang = 6 (bayangkan kondisi lift pagi hari saat belum terlalu ramai).

c. Uji program dengan jumlahOrang = 10 (bayangkan kondisi lift siang hari saat penuh sesak).

Pembahasan:

**Printscreen Source Code**



**Source Code**

public class Main {

public static void main(String[] args) {

int kapasitas = 4;

String result;

result = (kapasitas <= 8) ? "lift bisa jalan." : "Terlalu penuh,tunggu lift selanjutnya.";

System.out.println(result);

}

}

**Penjelasan Source Code**

public class Main

→ mendefinisikan class utama bernama Main.

public static void main(String[] args)

→ titik awal program dijalankan.

int Tarif\_parkir = 1

→ variabel yang menyimpan jam parkir.

if (Tarif\_parkir < 12)

→ jika jam parkir pagi/siang, tampilkan “siang” + Rp.5.000.

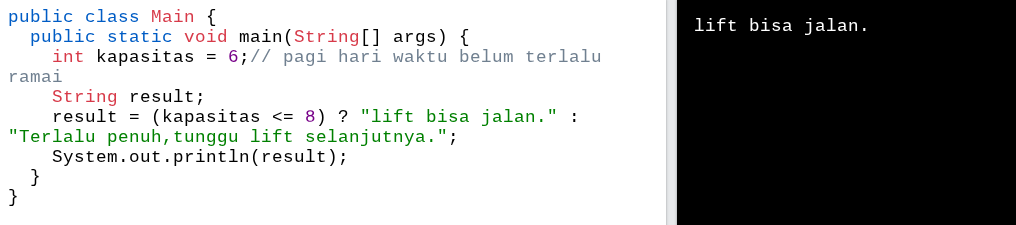
else if (Tarif\_parkir > 12 && Tarif\_parkir < 18)

→ jika jam parkir sore, tampilkan “sore” + Rp.7.000.

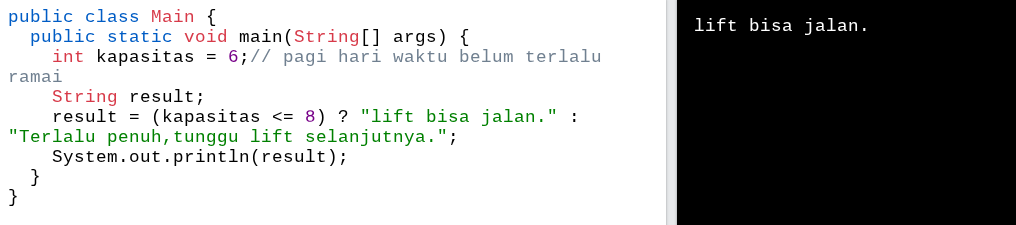
else if (Tarif\_parkir > 18)

→ jika jam parkir malam, tampilkan “malam” + Rp.10.000.

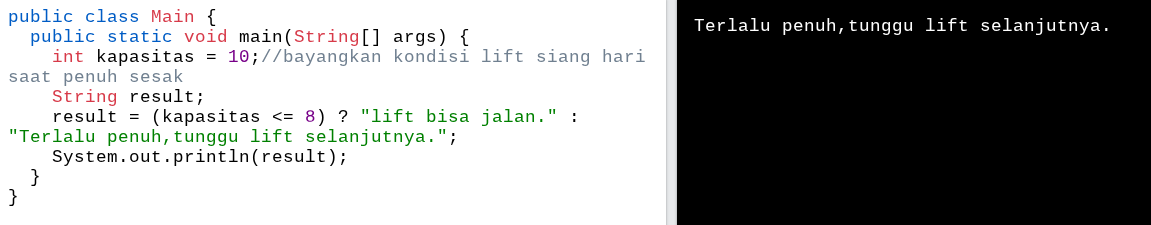
b.**Printscreen Source Code**



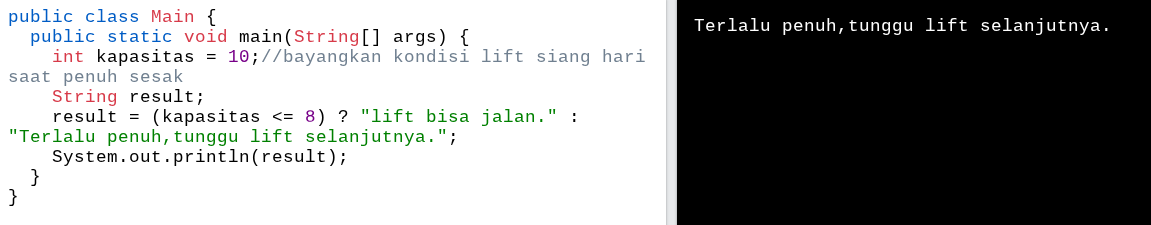
**Printscreen Output**



c.**Printscreen Source Code**

****

**Printscreen Output**

****

5.Rifqi dan Zora adalah mahasiswa yang ingin mengakses koleksi e-book di perpustakaan digital kampus.

Namun, sistem memiliki aturan akses sebagai berikut:

Jika mahasiswa sudah terdaftar sebagai anggota, maka dicek lagi:

Jika ia sudah membayar iuran, tampilkan: “Akses e-book dibuka”.

Jika ia belum membayar iuran, tampilkan: “Harap bayar iuran dulu”.

Jika mahasiswa bukan anggota, tampilkan: “Silakan daftar anggota dahulu”.

Mahasiswa tersebut kemudian meminta bantuan temannya yang belajar pemrograman untuk membuat simulasi aturan ini.

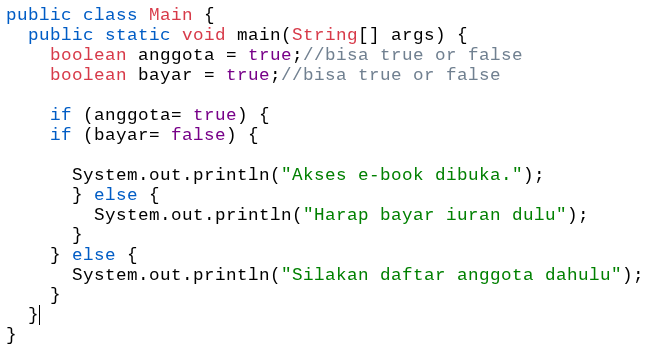
Pertanyaan:

a. Buatlah kode program menggunakan nested if untuk aturan akses e-book di atas.

b. Uji program dengan kondisi: anggota = true, bayar = true.

c. Uji program dengan kondisi: anggota = true, bayar = false.

a.**Printscreen Source Code**



**Source Code**

public class Main {

public static void main(String[] args) {

boolean anggota = true;//bisa true or false

boolean bayar = true;//bisa true or false

if (anggota= true) {

if (bayar= false) {

System.out.println("Akses e-book dibuka.");

} else {

System.out.println("Harap bayar iuran dulu");

}

} else {

System.out.println("Silakan daftar anggota dahulu");

}

}

}

**Source Code**

public class Main

→ mendefinisikan class utama bernama Main.

public static void main(String[] args)

→ titik awal program dijalankan.

boolean anggota = true;

→ variabel boolean untuk menyimpan status anggota (true atau false).

boolean bayar = true;

→ variabel boolean untuk menyimpan status pembayaran iuran (true atau false).

if (anggota)

→ cek apakah mahasiswa sudah terdaftar sebagai anggota.

if (bayar)

→ nested if: jika mahasiswa anggota, cek apakah sudah membayar iuran.

System.out.println("Akses e-book dibuka.");

→ tampilkan pesan jika mahasiswa sudah anggota dan membayar iuran.

else

→ jika mahasiswa anggota tapi belum membayar iuran.

System.out.println("Harap bayar iuran dulu");

→ tampilkan pesan agar mahasiswa membayar iuran terlebih dahulu.

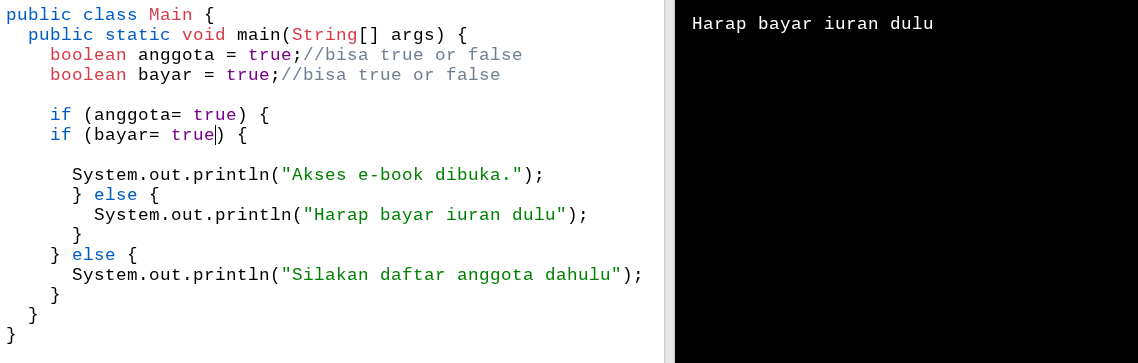
else

→ jika mahasiswa bukan anggota.

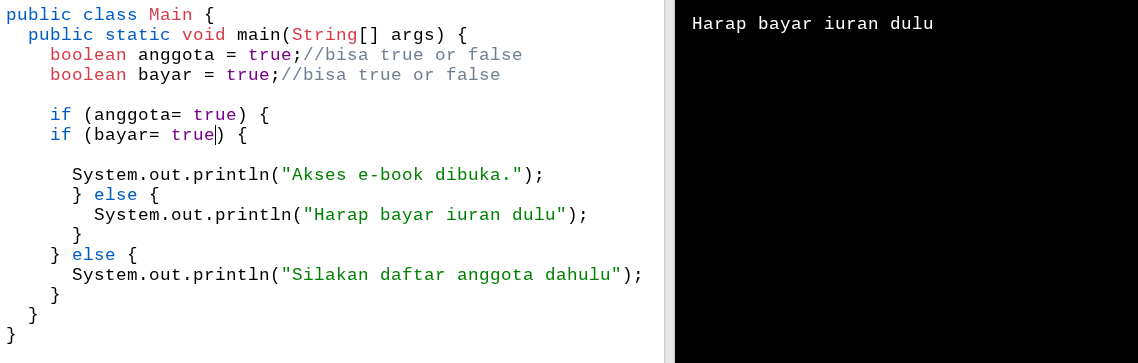
System.out.println("Silakan daftar anggota dahulu");

→ tampilkan pesan agar mahasiswa mendaftar sebagai anggota.

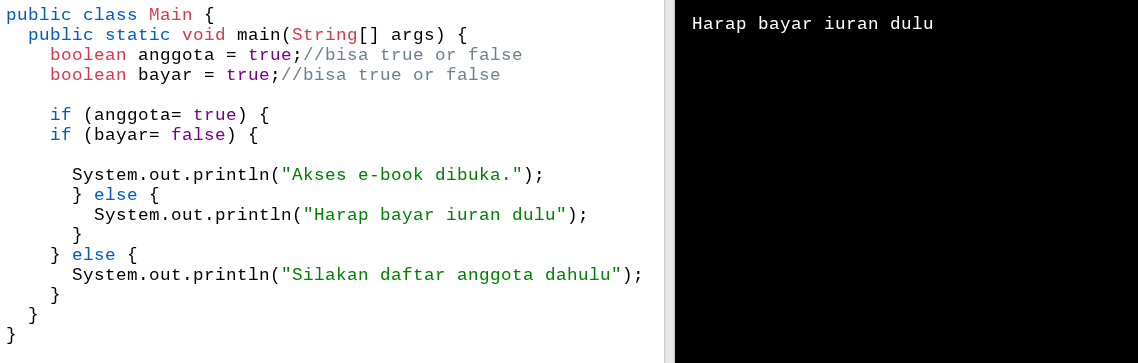
b.**Printscreen Source Code**



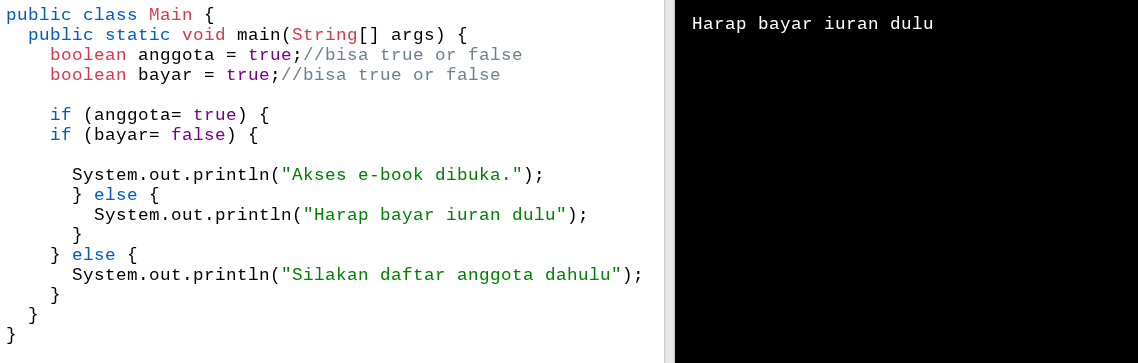
**Printscreen Output**



c.**Printscreen Source Code**

****

**Printscreen Output**

****

6. Sebuah drone harus memilih arah berdasarkan kondisi:

Jika angin datang dari utara (drone terbang ke selatan).

Jika angin datang dari barat (drone terbang ke timur).

Selain itu (drone hover atau diam di tempat).

Lengkapi kode berikut agar sesuai aturan:

Maka jawablah soal berikut

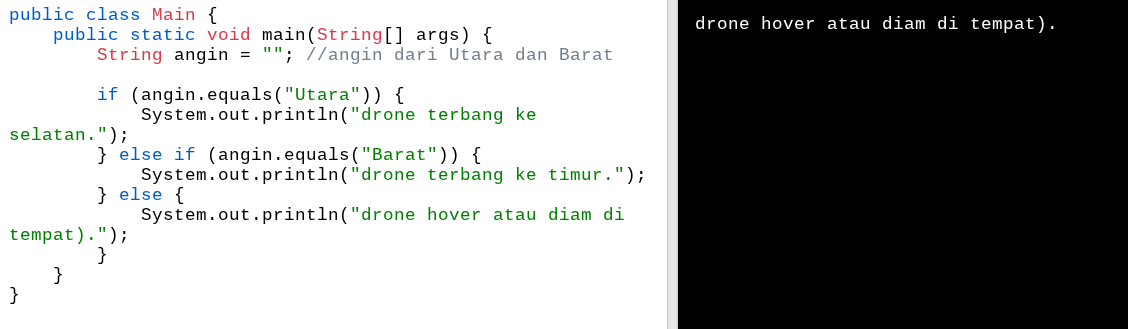
a. Lengkapi bagian yang kosong.

b. Uji dengan angin = “barat”.

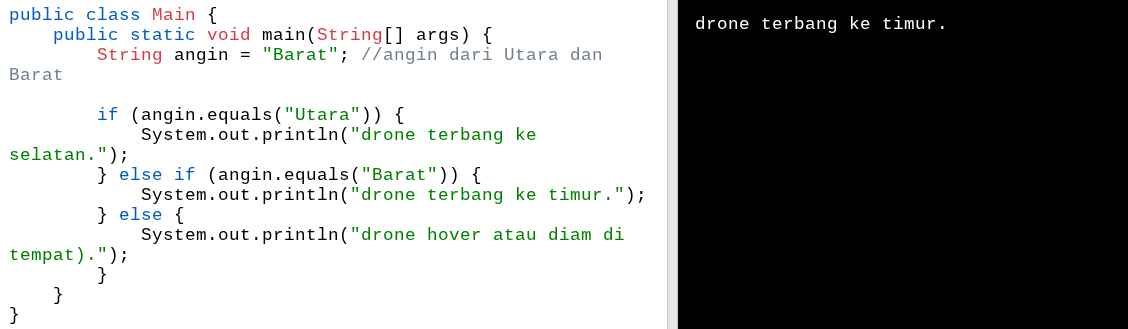
c. Uji dengan angin = “selatan”.

Pembahasan:

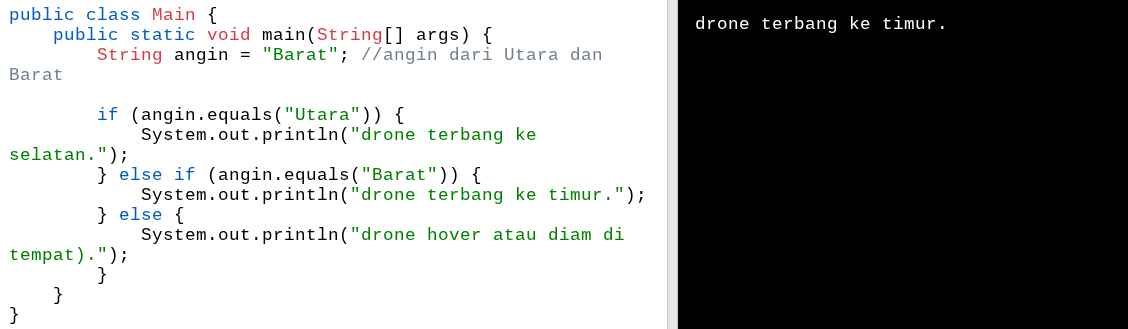
a.**Printscreen Source Code**



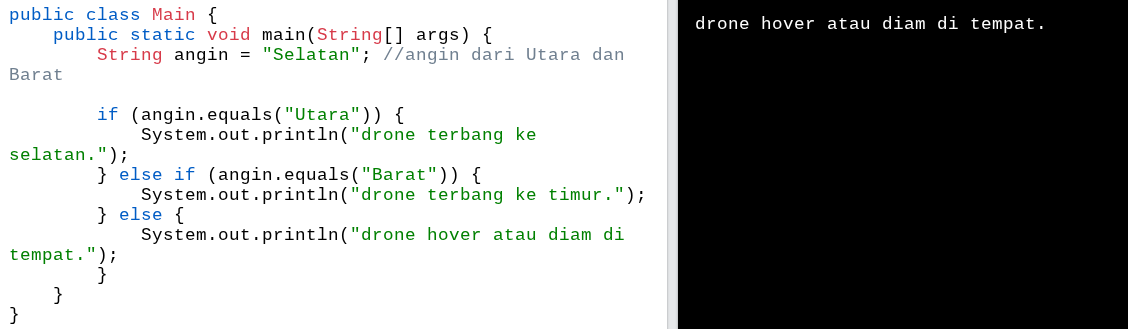
b.**Printscreen Source Code**



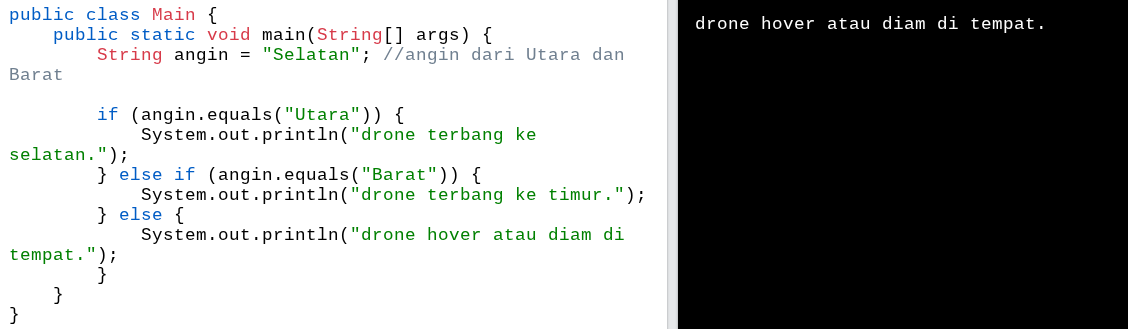
**Printscreen Output**



c.**Printscreen Source Code**

****

**Printscreen Output**

****

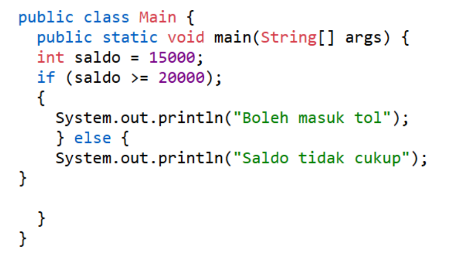
7. Melzaqi menulis program untuk mengecek apakah mobil boleh masuk jalan tol.

Aturannya:

Jika saldo e-Toll ≥ 20.000 → boleh masuk.

Jika tidak, tidak boleh masuk.

Kode Melzaqi:



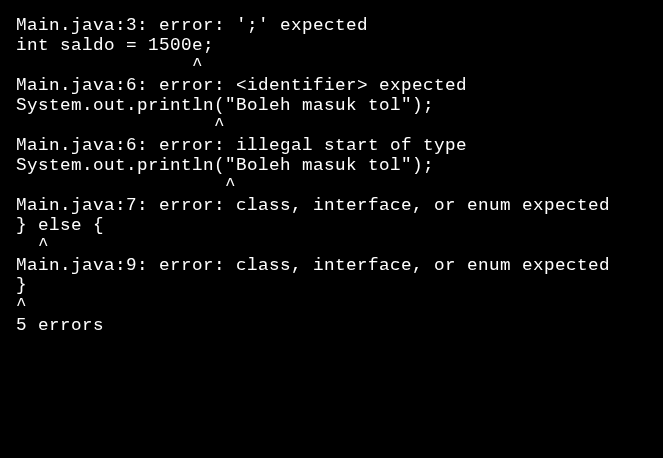
Maka,

a. Apa output yang didapatkan jika ada kesalahan tunjukkan dimana kesalahan pada kode program tersebut.

b. Perbaiki kode program tersebut.

Pembahasan:

**Printscreen Output**



masalah:

1.1500e bukan angka yang valid untuk tipe int

2.Method main ditutup di baris ke-4 (}), jadi semua statement setelahnya (if dan System.out.println) tidak berada di dalam method.

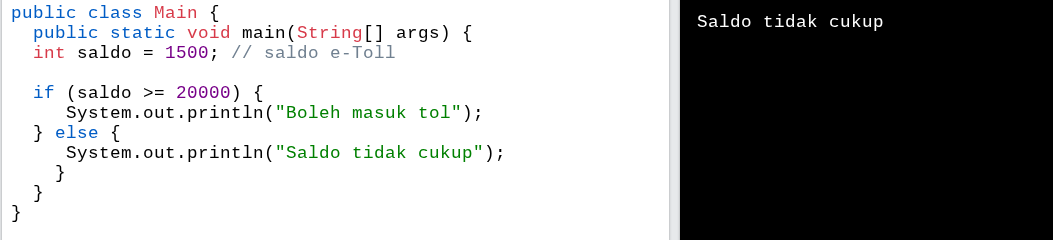
3.println berada di tempat yang salah → bukan bagian dari method → error.

4.else muncul di luar if, karena {} sebelumnya sudah ditutup.

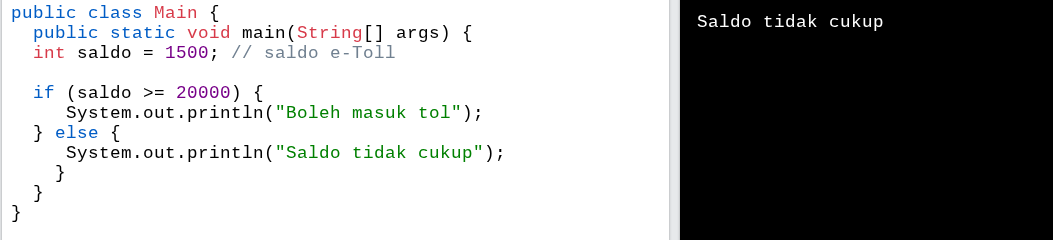
5.System.out.println("Saldo tidak cukup");

Masih berada di luar method/class, sehingga compiler menganggapnya bukan bagian dari program → error class, interface, or enum expected.

b.**Printscreen Source Code**



**Printscreen Output**



**KESIMPULAN DAN SARAN**

KESIMPULAN  
 Berdasarkan hasil praktikum dan kajian teori, dapat disimpulkan bahwa struktur percabangan seperti if, if-else, dan if-else if-else memiliki peran sangat penting dalam pemrograman dasar. Percabangan memungkinkan program menyesuaikan jalannya eksekusi berdasarkan kondisi tertentu, memberikan respons yang tepat terhadap input pengguna, serta membuat program menjadi lebih interaktif, adaptif, dan mudah dipahami. Dalam kegiatan praktikum, baik melalui pembuatan game berbasis Python maupun pengembangan aplikasi pembelajaran berbasis mobile phone, percabangan membantu siswa memahami logika kondisional, mengevaluasi kondisi yang berbeda, serta menentukan aksi yang tepat untuk setiap skenario. Dengan demikian, siswa dapat melihat secara langsung dampak dari setiap keputusan logis yang mereka buat, sehingga pemahaman terhadap alur program menjadi lebih nyata dan konkret.

Selain aspek teknis, penerapan percabangan juga mendorong pengembangan kemampuan berpikir analitis, logis, dan sistematis. Siswa belajar memecah masalah menjadi beberapa kondisi, menganalisis berbagai kemungkinan, dan mengambil keputusan yang sesuai untuk masing-masing kondisi. Proses ini melatih pola pikir komputasional, kemampuan problem-solving, serta keterampilan berpikir kritis yang berguna tidak hanya dalam pemrograman, tetapi juga dalam pemecahan masalah pada konteks lain. Praktikum berbasis game atau aplikasi edukatif juga meningkatkan motivasi siswa untuk bereksperimen dengan kode, mencoba berbagai skenario, dan memahami hubungan sebab-akibat dalam logika program, sehingga pengalaman belajar menjadi lebih interaktif, menyenangkan, dan mendalam.

SARAN  
 Disarankan agar praktikum pemrograman selalu menekankan latihan yang melibatkan percabangan dan skenario nyata, sehingga siswa memperoleh pengalaman langsung dalam merancang program yang interaktif, adaptif, dan efektif. Praktikum sebaiknya dilengkapi dengan diskusi, evaluasi hasil, serta refleksi terhadap kesalahan, agar siswa dapat memahami penyebab kesalahan dan belajar memperbaikinya. Pengajar juga dianjurkan mendorong siswa mencoba berbagai kombinasi kondisi, percabangan bertingkat, serta simulasi yang relevan, sehingga kemampuan berpikir logis, analitis, kreatif, dan sistematis berkembang secara optimal. Dengan pendekatan ini, siswa tidak hanya menguasai keterampilan teknis menulis kode, tetapi juga mampu mengaplikasikan logika pemrograman secara adaptif dan kreatif dalam berbagai situasi pembelajaran, sekaligus membangun dasar yang kuat untuk memahami konsep pemrograman yang lebih kompleks di masa depan.

**DAFTAR PUSTAKA**

Afrizal, A. S. (2017). Rancang Bangun Aplikasi Pembelajaran Dasar Pemrograman Berbasis Mobile Phone. *Jurnal TIPS: Jurnal Teknologi Informasi dan Komputer Politeknik Sekayu*, *6*(1), 1-19.

Pambudi, S., & Setiawan, A. (2025). Pelatihan Pemrograman Dasar Python:: Meningkatkan Literasi Teknologi Siswa melalui Pembuatan Game Tebak Angka di SMAN 1 Pamotan. *Indonesian Research Journal on Education*, *5*(1), 7-12.

Sagala, A. A. H., & Yahfizham, Y. (2024). Analisis Pengenalan Konsep Algoritma Pemrograman Matematika Pada Kehidupan Sehari Hari. *Morfologi: Jurnal Ilmu Pendidikan, Bahasa, Sastra dan Budaya*, *2*(1), 01-16.