

Template Lembar Kerja Individu

Nama & NPM	Topik:	Tanggal:
Nafiisah Calista Hersa Agustine_G1F024077	IF dan SWITCH Java	23 September 2024

[Nomor 1] Identifikasi Masalah:

- 1) Uraikan permasalahan dan variable

Contoh 1

```
import java.util.Scanner; //memanggil impor package yang membaca masukan pengguna

public class PercabanganIf {
    public static void main(String[] args) {
        Scanner masuk = new Scanner(System.in); // membaca teks yang dimasukkan pengguna
        System.out.print("Masukkan Angka Anda : "); //pengguna memasukkan data
        int nilai = masuk.nextByte(); //menyimpan masukan pengguna ke tipe data

        if (nilai == 10) { //percabangan yang memeriksa kondisi
            System.out.println("Sepuluh"); //baris kode yang dieksekusi bila benar
        }
        else { //baris kode yang dieksekusi bila kondisi tidak terpenuhi dan salah
            System.out.println("Nilai Bukan Sepuluh");
        }
    }
}
```

Contoh 2

```
import java.util.Scanner;

public class IfBersarang {
    public static void main(String[] args) {
        Scanner varT = new Scanner(System.in);
        System.out.print("Masukkan Angka Tugas Anda : ");
        int nilaiT = varT.nextByte();

        Scanner varQ = new Scanner(System.in);
        System.out.print("Masukkan Angka Quiz Anda : ");
        int nilaiQ = varQ.nextByte();

        if (nilaiU >= 80) {
            if (nilaiT >= 80) {
                System.out.println("Anda mendapatkan nilai A");
            }
        }
        else{
            System.out.println("Anda TIDAK mendapatkan nilai A");
        }
    }
}
```

1.1. Bilangan genap merupakan bilangan yang habis dibagi 2. Bilangan ganjil adalah bilangan yang tidak habis dibagi 2. Analisa kode program yang tepat untuk menghitung masukan pengguna termasuk bilangan genap atau bilangan ganjil (lihat Contoh 1)?

(Petunjuk: hitung = nilai % 2 jika hitung = 0 maka bilangan genap, jika hitung = 1 maka bilangan ganjil)

```
import java.util.Scanner;
public class PercabanganIf {
    public static void main(String[] args) {
        Scanner masuk = new Scanner(System.in);
        System.out.print("Masukkan Angka Anda : ");
        int nilai = masuk.nextByte();
        if (nilai % 2 == 0) {
            System.out.println("Bilangan genap");
        }
    }
}
```

```

else {
    System.out.println("Bilangan ganjil");
} }
}

```

Luaran :

Masukkan Angka Anda : 9

Bilangan ganjil

1.2. Cermati contoh 2, analisa kondisi pada IF bersarang!

Tambahkan satu kondisi IF dengan satu nilai input Quiz (nilaiQ). Jika nilai UTS, Tugas, dan Quiz lebih besar sama dengan 80 maka siswa mendapat nilai A.

```
import java.util.Scanner;
```

```

public class IfBersarang {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Masukkan Angka Tugas Anda: ");
        int nilaiT = scanner.nextInt();

        System.out.print("Masukkan Angka Quiz Anda: ");
        int nilaiQ = scanner.nextInt();

        System.out.print("Masukkan Angka Ujian Anda: ");
        int nilaiU = scanner.nextInt();

        if (nilaiU >= 90 && nilaiT >= 85 && nilaiQ >= 95) {
            System.out.println("Anda mendapatkan nilai A");
        } else {
            System.out.println("Anda TIDAK mendapatkan nilai A");
        }
    }
}

```

Luaran :

Masukkan Angka Tugas Anda: 85

Masukkan Angka Quiz Anda: 95

Masukkan Angka Ujian Anda: 90

Anda mendapatkan nilai A

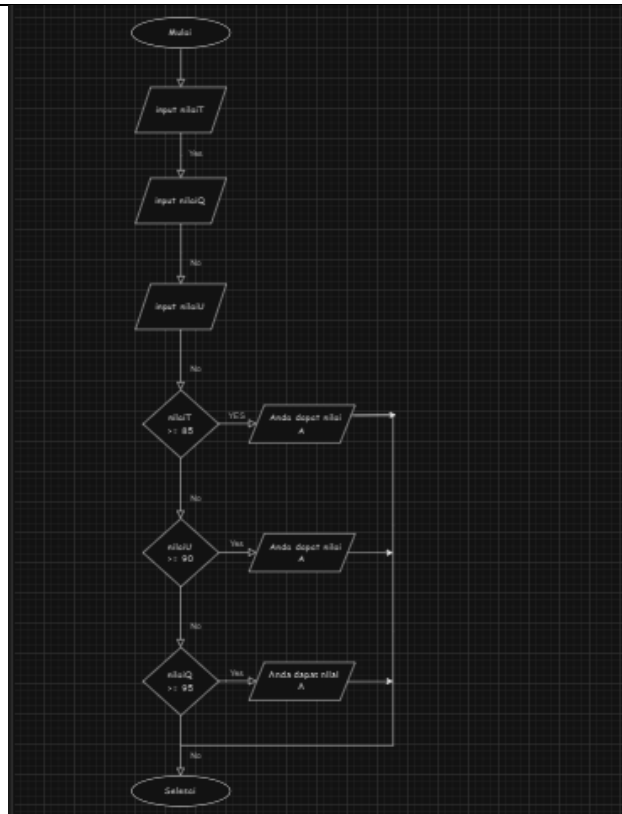
1.3. Apakah ketiga kondisi IF pada Contoh 1.2. dapat diringkas menjadi satu kondisi?

Periksa satu kondisi mana yang paling tepat menggantikan ketiga kondisi itu!

- a. IF (nilaiU >= 80 || nilaiT >= 80 || nilaiQ >= 80)
- b. IF (nilaiU >= 80 || nilaiT >= 80 && nilaiQ >= 80)
- c. IF (nilaiU >= 80 && nilaiT >= 80 || nilaiQ >= 80)
- d. IF (nilaiU >= 80 && nilaiT >= 80 && nilaiQ >= 80)

Kondisi yang tepat menggantikannya adalah D

1.4. Uraikan gambar diagram flowchart dari Latihan 1.2!



2) Rincikan sumber informasi yang relevan (buku / webpage)

Video Materi 1 tentang IF – <https://www.youtube.com/watch?v=G0dfdAFa9iM>

Video Materi 2 tentang SWITCH – <https://www.youtube.com/watch?v=RB4nz4xkisM>

[Nomor 1] Analisis dan Argumentasi

Contoh 1

1) Uraikan rancangan solusi yang diusulkan.

Program ini dirancang untuk menentukan apakah bilangan yang dimasukkan pengguna adalah genap atau ganjil dengan menggunakan operator modulus.

2) Analisis solusi, kaitkan dengan permasalahan.

Solusi ini efektif dan efisien, namun perlu ditambahkan validasi input untuk menangani kesalahan jika pengguna memasukkan nilai yang tidak valid.

Contoh 2

1) Uraikan rancangan solusi yang diusulkan.

Program ini dirancang untuk menilai apakah pengguna mendapatkan nilai A berdasarkan nilai tugas, kuis, dan ujian yang dimasukkan.

2) Analisis solusi, kaitkan dengan permasalahan.

Solusi ini efektif, namun perlu ditambahkan validasi input dan penanganan kesalahan untuk memastikan pengguna tidak memasukkan nilai yang tidak valid.

[Nomor 1] Penyusunan Algoritma dan Kode Program

1) Rancang desain solusi atau algoritma

Contoh 1

- Mulai program
- Inisialisai scanner
- Minta input pengguna
- Validasi input
- Baca input
- Cek bilangan genap atau ganjil
- Tutup scanner
- Selesai program

Contoh 2

- Mulai program
 - Inisialisasi scanner
 - Minta input nilai tugas
 - Minta input nilai quiz
 - Minta input nilai ujian
 - Cek kelayakan untuk nilai A
 - Tutup scanner
 - Selesai program
- 2) Tuliskan kode program dan luaran
- a) Beri komentar pada kode

Contoh 1

Program ini menggunakan kelas scanner untuk meminta pengguna memasukkan angka, kemudian memeriksa apakah angka tersebut genap atau ganjil dengan menggunakan operator modulus. Jika angka genap, program menampilkan “bilangan genap”, dan jika tidak, menampilkan “bilangan ganjil”. Setelah itu, objek scanner ditutup untuk membebaskan sumber daya.

Contoh 2

Program java ini meminta pengguna untuk memasukkan nilai tugas, kuis, dan ujian, lalu mengecek apakah nilai ujian ≥ 90 , nilai tugas ≥ 85 , dan nilai quiz ≥ 95 untuk menentukan apakah pengguna mendapatkan nilai A atau tidak.

- b) Uraikan luaran yang dihasilkan

Contoh 1

Masukkan Angka Anda : 9

Bilangan ganjil

Contoh 2

Masukkan Angka Tugas Anda: 85

Masukkan Angka Quiz Anda: 95

Masukkan Angka Ujian Anda: 90

Anda mendapatkan nilai A

- c) Screenshot/ Capture potongan kode dan hasil luaran

Contoh 1



```
1 import java.util.Scanner;
2
3 public class Percobaan1 {
4     public static void main(String[] args) {
5         Scanner masuk = new Scanner(System.in);
6         System.out.print("Masukkan Angka Anda : ");
7         int nilai = masuk.nextInt();
8
9         if (nilai % 2 == 0) {
10             System.out.println("Bilangan genap");
11         } else {
12             System.out.println("Bilangan ganjil");
13         }
14     }
15 }
```

Input/Output

Language Version: JDK 21.0.0 Interactive

Input Arguments

Output Generated Files

Masukkan Angka Anda : 9
Bilangan ganjil

Compiled and executed in 7.452 secs

Contoh 2



```
1 import java.util.Scanner;
2
3 public class IFBersyarat {
4     public static void main(String[] args) {
5         Scanner scanner = new Scanner(System.in);
6
7         System.out.print("Masukkan Angka Tugas Anda: ");
8         int nilaiT = scanner.nextInt();
9
10        System.out.print("Masukkan Angka Quiz Anda: ");
11        int nilaiQ = scanner.nextInt();
12
13        System.out.print("Masukkan Angka Ujian Anda: ");
14        int nilaiU = scanner.nextInt();
15
16        if (nilaiU >= 90 && nilaiT >= 85 && nilaiQ >= 95) {
17            System.out.println("Anda mendapatkan nilai A");
18        } else {
19            System.out.println("Anda tidak mendapatkan nilai A");
20        }
21    }
22 }
23 }
```

Input/Output

Language Version: JDK 21.0.0 Interactive

Input Arguments

Output Generated Files

Masukkan Angka Tugas Anda: 85
Masukkan Angka Quiz Anda: 95
Masukkan Angka Ujian Anda: 90
Anda mendapatkan nilai A

Compiled and executed in 106.474 secs

[Nomor 1] Kesimpulan
<div data-bbox="229 228 959 302" data-label="List-Group"><ol style="list-style-type: none">1) Evaluasi<ol style="list-style-type: none">a) Apa konsekuensi dari skenario pemrograman ini?</div> <div data-bbox="181 302 308 336" data-label="Section-Header">Contoh 1</div> <div data-bbox="181 336 1356 481" data-label="Text"><p>Konsekuensi dari skenario pemrograman ini mencakup batasan pada input tipe byte yang dapat menyebabkan kesalahan jika angka terlalu besar, kesederhanaan dan interaktivitas yang baik untuk pemula, serta perlunya validasi input untuk mencegah kesalahan runtime, dengan potensi untuk pengembangan lebih lanjut.</p></div> <div data-bbox="181 515 308 551" data-label="Section-Header">Contoh 2</div> <div data-bbox="181 551 1382 694" data-label="Text"><p>Konsekuensi dari skenario pemrograman ini mencakup ketergantungan pada input pengguna yang dapat memengaruhi akurasi hasil, kriteria penilaian yang ketat, dan keterbatasan dalam memberikan umpan balik, serta perlunya validasi input dan potensi untuk pengembangan lebih lanjut.</p></div>

Template Lembar Kerja Individu

Nama & NPM	Topik:	Tanggal:
Nafiisah Calista Hersa Agustine_G1F024077	IF dan SWITCH Java	24 September 2024

[Nomor 2] Identifikasi Masalah:

- 1) Uraikan permasalahan dan variable

Contoh 3

```
import java.util.Scanner;
public class SwitchBersarang {
    public static void main(String[] args) {
        Scanner masukData = new Scanner(System.in);
        // mengambil input
        System.out.print("Pilih A atau B : ");
        char data = masukData.next().charAt(0);
        switch(data) {
            case 'A':
                System.out.print("Anda sudah rajin belajar");
                break; // baris 1
            case 'B':
                System.out.print(" Anda perlu kurangi main game");
                break; // baris 2
            default:
                System.out.print(" Pilihan anda diluar A atau B ");
        }
    }
}
```

Contoh 4

```
import java.util.Scanner;
public class SwitchBersarang {
    public static void main(String[] args) {
        byte bulan;
        int tahun = 2022;
        int jumlahHari = 0;
        System.out.print("Masukkan data bulan (dalam angka): ");
        Scanner masukData = new Scanner(System.in);
        bulan = masukData.nextByte();

        switch (bulan) {
            case 1: case 3: //baris 1
                jumlahHari = 31;
                break;
            case 4: //baris 2
                jumlahHari = 30;
                break;
            case 2:
                if (tahun % 4 == 0)
                    jumlahHari = 29;
                else
                    jumlahHari = 28;
                break;
            default:
                System.out.println("Maaf bulan hanya sampai 12.");
                break;
        }
        System.out.println("Jumlah hari = " + jumlahHari);
    }
}
```

- 2.1. Cermati kode pada Contoh 3.

Hapuslah kode **break**; pada //baris 1, eksekusi kembali.

```
import java.util.Scanner;
public class SwitchBersarang {
```

```

public static void main(String[] args) {
    Scanner masukData = new Scanner(System.in);
    // mengambil input
    System.out.print("Pilih A atau B : ");
    char data = masukData.next().charAt(0);
    switch(data) {

        case 'B':
            System.out.print(" Anda perlu kurangi main game");
            break; // baris 2
        default:
            System.out.print(" Pilihan anda diluar A atau B ");
    } }

```

Kemudian hapuslah kode `break`; pada //baris 2, eksekusi kembali.

```

import java.util.Scanner;
public class SwitchBersarang {
    public static void main(String[] args) {
        Scanner masukData = new Scanner(System.in);
        // mengambil input
        System.out.print("Pilih A atau B : ");
        char data = masukData.next().charAt(0);
        switch(data) {
            default:
                System.out.print(" Pilihan anda diluar A atau B ");
        } } }

```

Analisis perbedaan hasil luaran ketika kode menggunakan `break`, ketika kode `break` baris 1 dihapus, dan ketika kode `break` baris 2 dihapus!

Hasil luaran ketika menggunakan `break` adalah memastikan program memberikan respon yang tepat untuk input A atau B tanpa melanjutkan ke opsi lain, dan juga memberikan pesan jika input tidak valid. Ketika kode `break` baris 1 di hapus yang terjadi tanpa `break` di case A, program menampilkan pesan untuk A dan langsung melanjutkan ke pesan untuk B, sehingga menghasilkan output yang membingungkan. Ketika kode `break` baris ke 2 yang dihapus yang terjadi adalah tanpa `break` di case B, program menampilkan pesan untuk A, kemudian melanjutkan ke pesan untuk B dan pesan kesalahan, sehingga output menjadi membingungkan.

Analisa kegunaan baris kode dengan `break` dan kata kunci `default`!

`Break` digunakan untuk menghentikan eksekusi didalam switch, mencegah output yang membingungkan dengan memastikan hanya satu blok yang dijalankan, sementara `default` menangani input yang tidak valid, memberikan umpan balik yang jelas dan meningkatkan keandalan program.

2.2. Cermati kode pada Contoh 4 yang menampilkan jumlah hari sesuai dengan bulannya. Namun kode tersebut baru sampai bulan ke-4. Tambahkan sampai bulan ke-12 pada baris ke-1 dan baris ke-2.

```

import java.util.Scanner;
public class SwitchBersarang {
    public static void main(String[] args) {
        byte bulan;
        int tahun = 2022;
        int jumlahHari = 0;
        System.out.print("Masukkan data bulan (dalam angka): ");
        Scanner masukData = new Scanner(System.in);
        bulan = masukData.nextByte();

        switch (bulan) {

```

```

case 1: case 3: case 5: case 7: case 9: case 11:
    jumlahHari = 31;
    break;
case 4: case 6: case 8: case 10: case 12:
    jumlahHari = 30;
    break;
case 2:
    if (tahun % 4 == 0)
        jumlahHari = 29;
    else
        jumlahHari = 28;
    break;
default:
    System.out.println("Maaf bulan hanya sampai 12.");
    break;
}
System.out.println("Jumlah hari = " + jumlahHari);
}
}

```

2.3. Cermati permasalahan yang dipecahkan pada Contoh 3.

Apakah masalah ini bisa diubah menjadi perintah **IF**?

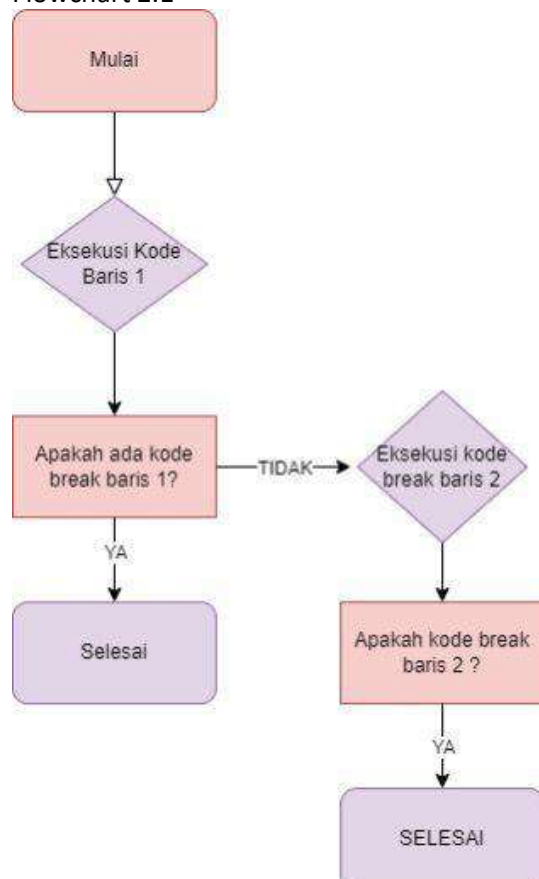
Jika bisa, rincikan analisa Anda!

Bandingkan masalah yang dapat diselesaikan percabangan dengan **IF** atau **SWITCH** !

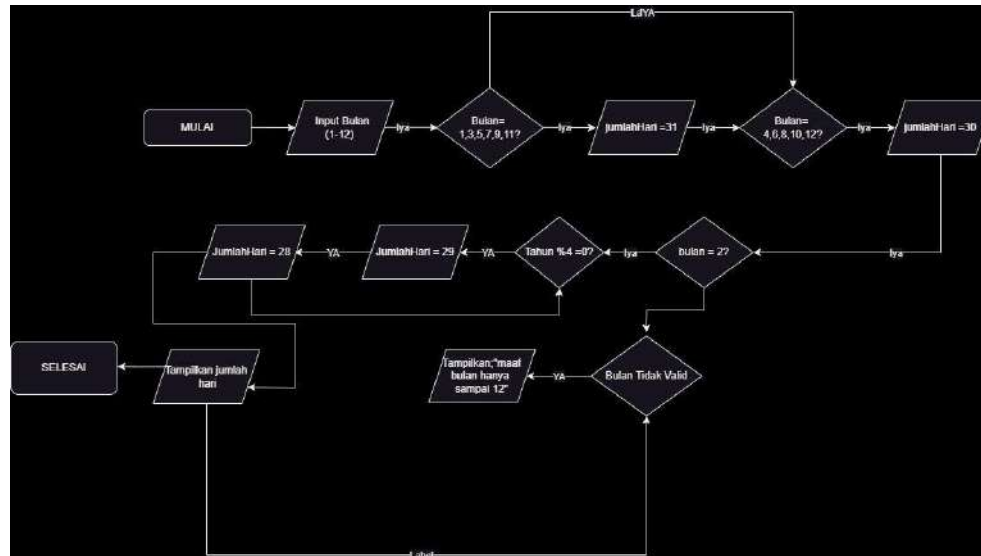
Masalah ini dapat diubah menjadi perintah if, yang lebih fleksibel untuk kondisi kompleks, sementara switch lebih jelas dan terstruktur untuk banyak nilai tetap. Pilihan antara if dan switch tergantung pada konteks switch ideal untuk opsi tetap, sedangkan if lebih baik untuk logika dinamis.

2.4. Buatlah dokumentasi gambar flowchart dari Latihan 2.1. dan Latihan 2.2!

Flowchart 2.1



Flowchart 2.2



2) Rincikan sumber informasi yang relevan (buku / webpage)

Video Materi 1 tentang IF – <https://www.youtube.com/watch?v=G0dfdAFa9iM>

Video Materi 2 tentang SWITCH – <https://www.youtube.com/watch?v=RB4nz4xkisM>

[Nomor 2] Analisis dan Argumentasi

Contoh 1

1) Uraikan rancangan solusi yang diusulkan.

Program meminta pengguna untuk memilih antara A atau B dan menggunakan struktur switch untuk memproses input. Jika pengguna memilih B program mencetak pesan tentang mengurangi bermain game. Jika input tidak valid, program memberikan pesan bahwa pilihan di luar A atau B.

2) Analisis solusi, kaitkan dengan permasalahan.

Solusi ini efektif dalam memberikan umpan balik atas input pengguna, tetapi hanya menangani pilihan B secara spesifik, sedangkan A tidak didefinisikan. Kurangnya validasi lebih lanjut untuk input lain dapat membingungkan pengguna. Meskipun menggunakan switch adalah pilihan yang baik, penanganan kondisi tambahan dan validasi input yang lebih baik dapat meningkatkan kejelasan dan fungsionalitas program.

Contoh 2

1) Uraikan rancangan solusi yang diusulkan.

Program menghitung jumlah hari dalam bulan tertentu berdasarkan input pengguna. Pengguna diminta memasukkan nomor bulan, dan program menggunakan switch untuk menentukan jumlah hari. Bulan dengan 31 hari diatur ke 31, bulan dengan 30 hari diatur ke 30, dan untuk Februari, program memeriksa apakah tahun kabisat untuk menentukan jumlah hari. Jika input bulan tidak valid, program memberikan pesan kesalahan. Akhirnya, jumlah hari dicetak sebagai output.

2) Analisis solusi, kaitkan dengan permasalahan.

Solusi ini efektif dalam menghitung jumlah hari berdasarkan input, dengan penanganan yang baik untuk bulan dan tahun kabisat. Namun, tahun yang digunakan di-hardcore menjadi 2022, membatasi fleksibilitas, dan program tidak menangani input yang tidak valid dengan baik, berpotensi menyebabkan kesalahan. Secara keseluruhan, meskipun fungsional, program dapat ditingkatkan dengan memungkinkan input tahun dinamis dan menambah validasi input untuk keandalan yang lebih baik.

[Nomor 2] Penyusunan Algoritma dan Kode Program

1) Rancang desain solusi atau algoritma

Contoh 1

- Inisialisasi scanner
- Mint input pengguna
- Pengolahan input dengan switch
- Akhiri program

Contoh 2

- Inisialisasi variable
- Minta input pengguna
- Pengolahan input dengan switch
- Tampilkan hasil
- Akhiri program

- 1) Tuliskan kode program dan luaran
 - a) Beri komentar pada kode

Contoh 1

program ini meminta pengguna untuk memilih antara A atau B lalu menggunakan struktur switch untuk memproses input, jika pengguna memilih B, program menampilkan pesan tentang mengurangi bermain game, dan jika input tidak valid, program memberikan pesan kesalahan. Versi kedua dari program hanya menangani input yang tidak valid tanpa penanganan khusus untuk pilihan yang benar.

Contoh 2

Program ini meminta pengguna untuk memasukkan bulan dalam angka, lalu menggunakan struktur switch untuk menentukan jumlah hari dalam bulan tersebut, termasuk penanganan khusus untuk bulan februari berdasarkan tahun kabisat. Jika bulan tidak valid, program memberikan pesan kesalahan, dan akhirnya menampilkan jumlah hari yang sesuai.

- b) Uraikan luaran yang dihasilkan

Contoh 1

Pilih A atau B : A

Pilihan anda diluar A atau B

Contoh 2

Masukkan data bulan (dalam angka): 12

Jumlah hari = 30

- c) Screenshot/ Capture potongan kode dan hasil luaran

Contoh 1



The screenshot shows an online Java compiler interface. On the left, the source code is displayed in a dark-themed editor. It is a Java class named 'SwitchBerserang' with a 'main' method. The code uses a 'Scanner' to read input from the user, checks if it's 'A' or 'B', and prints a message if the input is invalid. On the right, the 'Input/Output' panel shows the execution results. The 'Language Version' is set to 'JDK 21.0.0' and 'Interactive' is checked. The 'Output' tab shows the program's execution: 'Pilih A atau B : A' followed by 'Pilihan anda diluar A atau B'. At the bottom, it states 'Compiled and executed in 8.81 sec(s)'.

```
1 import java.util.Scanner;
2
3 public class SwitchBerserang {
4     public static void main(String[] args) {
5         Scanner masukData = new Scanner(System.in);
6         // enum dari input
7         System.out.print("Pilih A atau B : ");
8         char data = masukData.next().charAt(0);
9         switch(data) {
10
11             default:
12                 System.out.print(" Pilihan anda diluar A atau B ");
13         }
14     }
15 }
```

Input/Output

Language Version: JDK 21.0.0 Interactive

Input Arguments

Output Generated Files

Pilih A atau B : A
Pilihan anda diluar A atau B

Compiled and executed in 8.81 sec(s)

Contoh 2

The screenshot shows an online Java IDE with a dark theme. The editor on the left contains a Java program named `SwitchBersarang`. The program uses a `switch` statement to determine the number of days in a month based on the month number and the year. The output window on the right shows the program's execution, where it prompts for a month and year, and then displays the result: "Masukkan data bulan (dalam angka): 12" and "Jumlah hari = 30". The IDE interface includes a top bar with "New Project", "Online Java Compiler IDE", and buttons for "Execute" and "Perl". The right sidebar has tabs for "Input/Output", "Generated Files", and "Console".

```
1 import java.util.Scanner;
2
3 public class SwitchBersarang {
4
5     public static void main(String[] args) {
6         byte bulan;
7         int tahun = 2022;
8         int jumlahHari = 0;
9         System.out.print("Masukkan data bulan (dalam angka): ");
10        Scanner scanner = new Scanner(System.in);
11        bulan = scanner.nextByte();
12
13        switch (bulan) {
14            case 1: case 3: case 5: case 7: case 9: case 11:
15                jumlahHari = 31;
16                break;
17            case 4: case 6: case 8: case 10: case 12:
18                jumlahHari = 30;
19                break;
20            case 2:
21                if (tahun % 4 == 0)
22                    jumlahHari = 29;
23                else
24                    jumlahHari = 28;
25                break;
26            default:
27                System.out.println("Maaf bulan hanya sampai 12.");
28                break;
29        }
30        System.out.println("Jumlah hari = " + jumlahHari);
31    }
32 }
```

Input/Output

Language Version: JDK 21.0.0 ☒ Interactive

Input Arguments

Output Generated Files

Masukkan data bulan (dalam angka): 12
Jumlah hari = 30

Compiled and executed in 21.362 sec(s)

[Nomor 2] Kesimpulan

1) Evaluasi

Contoh 1

a) Apa konsekuensi dari skenario pemrograman ini?

Scenario pemrograman ini memiliki beberapa konsekuensi, termasuk keterbatasan respons untuk pilihan A, kurangnya validasi input yang dapat menyebabkan kebingungan bagi pengguna, serta pengalaman pengguna yang buruk akibat umpan balik yang tidak memadai. Selain itu, struktur kode yang mirip dan tidak terorganisir dapat menyulitkan pemeliharaan, meskipun ada peluang untuk meningkatkan program dengan penanganan input yang lebih baik.

Contoh 2

a) Apa konsekuensi dari skenario pemrograman ini?

Scenario pemrograman ini memiliki beberapa konsekuensi, termasuk keterbatasan pada tahun yang di hardcode, kurangnya validasi input yang dapat menyebabkan kesalahan, serta pesan kesalahan yang tidak informatif. Pengalaman pengguna bias menjadi negative karena ketidaksengajaan mengenai logika bulan dan tahun kabisat, dan kode yang tidak fleksibel menyulitkan pemeliharaan serta ekspansi di masa depan.