|  |  |  |
| --- | --- | --- |
| **Nama & NPM** | **Topik:** | **Tanggal:** |
| **Meida Dinafani G1F024058** | **IF dan SWITCH** | **25 September 2024** |

**[No. 1] Identifikasi Masalah:**

**Contoh 1:** Salin dan tempel kode program berikut ke Eclipse.

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");  
}    }  
}

**Luaran  Contoh 1:**  
Masukkan Angka Anda : 8  
Nilai Bukan Sepuluh

**Contoh 2:** Salin dan tempel kode program berikut ke Eclipse.

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");  
    }  
    }  
}

**Luaran Contoh 2:**  
Masukkan Angka Tugas Anda : 70  
Masukkan Angka Quiz Anda : 70  
Masukkan Angka UTS Anda : 70  
Anda TIDAK mendapatkan nilai A

**Latihan 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).

Jawaban :

Pada program tersebut menggunakan operator modulus %, program disini menghitung sisa bagi nilai dengan 2. Jika hasilnya 0, maka angka tersebut genap, jika hasilnya 1, maka angka tersebut ganjil. Dalam program ini saya memasukkan angka 8 dengan hasilnya Bilangan genap dan Nilai bukan Sepuluh.

* 1. 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.

Jawaban :

Pada program tersebut menambahkan satu kondisi IF Bersarang yaitu nilai UTS. Program meminta input dari pengguna untuk nilai Tugas, Quiz, dan UTS. Jika kondisi terpenuhi, siswa mendapatkan nilai A. Jika salah satu kondisi tidak terpenuhi, maka siswa tidak 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)

Jawaban :

Ketiga kondisi IF tersebut dapat diringkas menjadi satu kondisi. Pilihan yang paling tepat adalah kondisi d.    IF (nilaiU >= 80 && nilaiT >= 80 && nilaiQ >= 80) karena memastikan bahwa semua syarat untuk mendapatkan nilai A terpenuhi. Dengan menggunakan kondisi ini, dapat menghindari situasi di mana mendapatkan nilai A meskipun jika ada salah satu nilai di bawah 80.

1.4. Uraikan gambar diagram flowchart dari Latihan 1.2!

Masukkan Angka Tugas Anda

Masukkan Angka Quiz Anda

Masukkan Angka UTS Anda

Apakah nilaiT >= 80?

**True**

Apakah nilaiQ >= 80?

**True**

Apakah nilaiU >= 80?

**True**

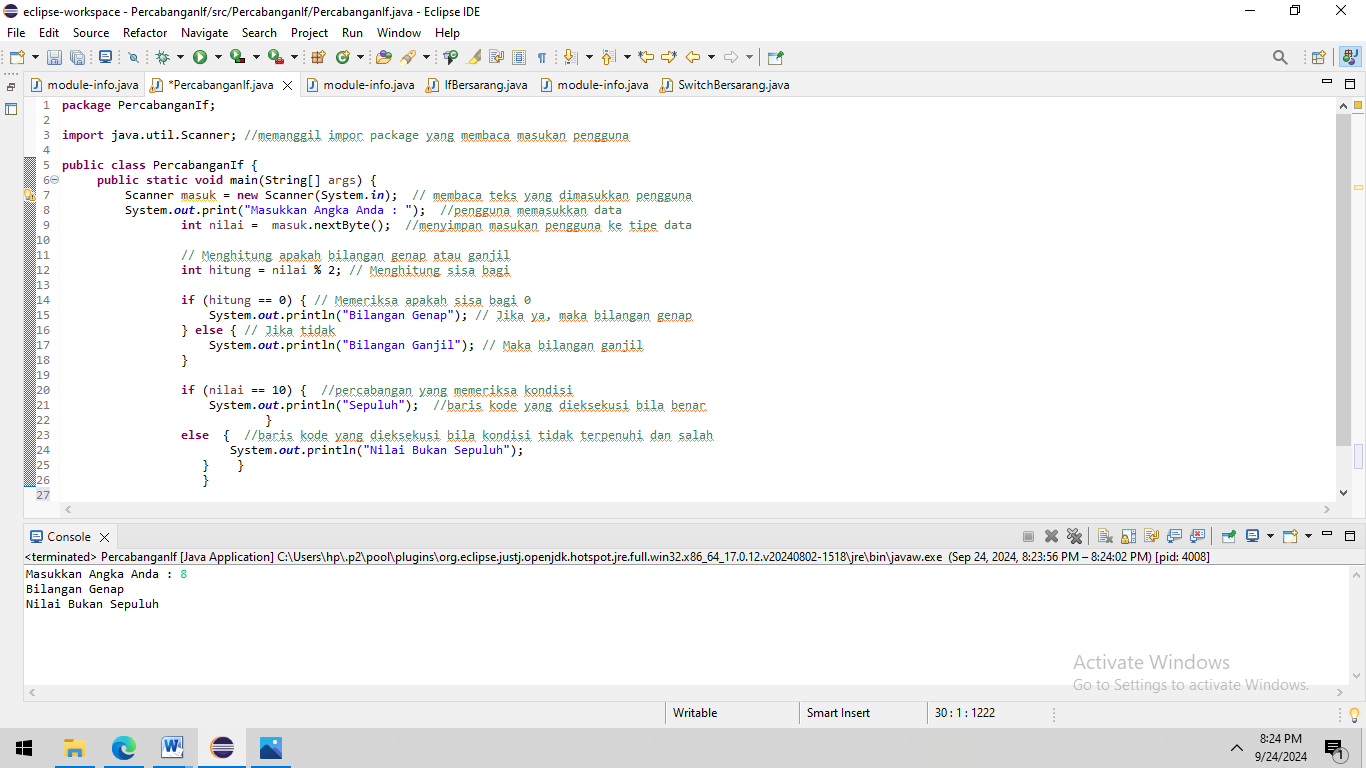
Anda Mendaoatkan Nilai A

**[No1.1] Analisis dan Argumentasi**

1. Saya mengusulkan permasalahan ini dapat diatasi dengan cara menyusun Kode program untuk menghitung apakah masukkan pengguna termasuk ke bilangan genap atau ganjil pada kode program.
2. Alasan solusi ini karena untuk mengidentifikasi apakah pengguna termasuk ke bilangan genap atau ganjil.
3. Perbaikan kode program dengan cara menambahkan penggunaan operasi modulus (hitung = nilai % 2 jika hitung = 0 maka bilangan genap, jika hitung = 1 maka bilangan ganjil). Untuk mengetahui apakah termasuk bilangan genap atau ganjil pada kode program.

**[No1.1 ] Penyusunan Algoritma dan Kode Program**

1. Algoritma
2. Mulai
3. Inisialisasi Scanner
4. Menampilkan pesan
5. Membaca input
6. Menghitung sisa bagi
7. Mengecek genap atau ganjil
8. Mengecek apakah sama dengan 10
9. Menutup objek Scanner
10. Selesai
11. Kode program dan luaran
12. Screenshot Kode program, hasil luaran, dan Komentar



1. Analisa luaran yang dihasilkan.

Luaran sudah sesuai dengan program yang disusun. Lalu saya memasukan angka pada output yaitu 8 dan keluar hasil bahwa angka 8 ini adalah bilangan genap dan Nilai bukan sepuluh.

**[No1.1] Kesimpulan   
(PILIH SALAH SATU ANDA INGIN MEMBAHAS DENGAN CARA ANALISA/ EVALUASI / KREASI)**

1. **Analisa**

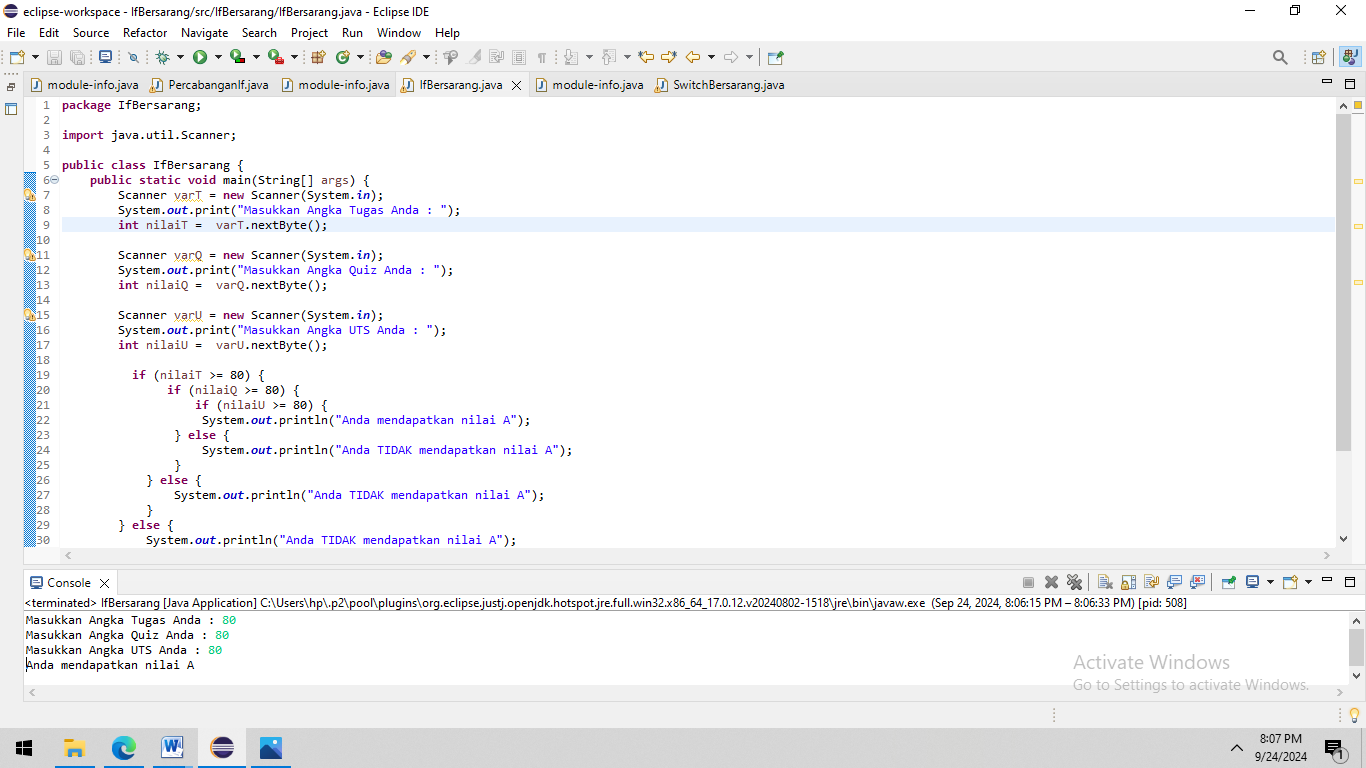
Pada program itu saya menggunakan bentuk penggunaan operasi modulus dengan menghitung : (hitung = nilai % 2 jika hitung = 0 maka bilangan genap, jika hitung = 1 maka bilangan ganjil). Kode program ini bertujuan untuk menentukan apakah sebuah angka yang dimasukkan oleh pengguna adalah genap atau ganjil, serta memeriksa apakah angka tersebut sama dengan Sepuluh.

**[No1.2] Analisis dan Argumentasi**

1. Saya mengusulkan permasalahan ini dapat diatasi dengan cara menyusun Kode program untuk menambahkan satu kondisi IF dengan satu nilai input yaitu nilai UTS (nilaiU).
2. Alasan solusi ini karena untuk mengevaluasi nilai berdasarkan tiga komponen : Tugas, Quiz,dan UTS .
3. Perbaikan kode program dengan cara menambahkan,kondisi yang memeriksa nilai Tugas, Quiz, dan UTS. Jika semua nilai tersebut lebih besar atau sama dengan 80, maka akan mendapatkan nilai A.

**[No1.2 ] Penyusunan Algoritma dan Kode Program**

1. Algoritma
2. Mulai
3. Deklarasi Variabel
4. Input Nilai
5. Menggunakan kondisi
6. Selesai
7. Kode program dan luaran
8. Screenshot Kode program, hasil luaran, dan Komentar



1. Analisa luaran yang dihasilkan.

Luaran sudah sesuai dengan program yang disusun. Lalu saya memasukan angka pada Nilai Tugas, Quiz, dan UTS dan keluar hasil bahwa mendapatkan nilai A

**[No1.2] Kesimpulan   
(PILIH SALAH SATU ANDA INGIN MEMBAHAS DENGAN CARA ANALISA/ EVALUASI / KREASI)**

1. **Analisa**

Pada program itu saya menggunakan bentuk if Bersarang untuk memeriksa beberapa kondisi. Dengan cara ini, dapat dengan mudah menentukan apakah berhak mendapatkan nilai A berdasarkan Nilai Tugas, Quiz, dan UTS. Jika semua nilai 80 atau lebih, maka akan mendapatkan nilai A.

**[No. 2] Identifikasi Masalah:**

**Contoh 3:**Salin dan tempel kode program berikut ke Eclipse.

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 ");   
}      }     }

**Luaran Contoh 3:**

Pilih A atau B :  A  
Anda sudah rajin belajar

**Contoh 4:**Salin dan tempel kode program berikut ke Eclipse.

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);  
    }  
}

**Luaran Contoh 4:**  
Masukkan data bulan (dalam angka): 7  
Jumlah hari = 31

**Latihan 2:**

2.1.    Cermati kode pada Contoh 3.   
          Hapuslah kode break; pada //baris 1, eksekusi kembali.   
          Kemudian hapuslah kode break; pada //baris 2, eksekusi kembali.  
          Analisis perbedaan hasil luaran ketika kode menggunakan break, ketika kode break baris 1 dihapus, dan ketika kode break baris 2 dihapus!   
          Analisa kegunaan baris kode dengan break dan kata kunci default!

Jawaban :

1. Menghapus break pada baris 1

Jika menghapus break; setelah case 'A':, hasilnya akan menjadi:

Jika pengguna memilih A, program akan mencetak "Anda sudah rajin belajar" **dan** kemudian melanjutkan ke case 'B':, sehingga program juga akan mencetak "Anda perlu kurangi main game".

1. Menghapus break pada baris 2

Jika Anda menghapus break; setelah case 'B':, hasilnya akan menjadi:

Jika pengguna memilih B, program akan mencetak "Anda perlu kurangi main game" dan kemudian melanjutkan ke default, yang akan mencetak "Pilihan anda diluar A atau B".

1. Analisis perbedaan hasil

* Dengan break: Hasil sesuai dengan yang diharapkan; hanya satu pernyataan yang dieksekusi berdasarkan input.
* Tanpa break pada baris 1: Program terus mengeksekusi pernyataan berikutnya setelah case 'A', sehingga dua pesan dicetak untuk input A.
* Tanpa break pada baris 2: Program melanjutkan ke default, mencetak dua pesan untuk input B.

1. Kegunaan break dan default

Break : digunakan untuk mencegah fall-through dalam pernyataan switch. Tanpa break, setelah mengeksekusi satu kasus, control akan terus bergerak ke kasus berikutnya. Hal ini berguna ketika kita ingin memastikan hanya satu blok kode yang dieksekusi berdasarkan input.

Default : digunakan untuk menyediakan jalur untuk menangani kasus yang tidak ditangani oleh kasus yang sudah ditentukan. Ini adalah cara untuk memberikan umpan balik kepada pengguna jika input tidak sesuai dengan ekspektasi (misalnya, bukan 'A' atau 'B').

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.

Jawaban :

Pada Kode program tersebut untuk melengkapi kode sehingga menampilkan jumlah hari untuk semua bulan hingga bula ke-12 dengan menambahkan case untuk bulan yang tersisa di bagian switch. Dengan cara ini, program sekarang dapat menangani semua bulan dari bulan ke-1 hingga 12 dan menampilkan jumlah hari yang benar.

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 !

Jawaban :

Program tersebut meminta pengguna untuk memilih antara 'A' atau 'B' dan memberikan respons berdasarkan pilihan tersebut. Dalam hal ini, bisa menggunakan pernyataan if untuk menangani logika yang sama.

Rincian Analisa

1. Fungsi yang sama: Baik switch maupun if akan memberikan hasil yang sama berdasarkan input pengguna
2. Keterbacaan: Struktur switch sering kali lebih mudah dibaca ketika ada banyak kondisi, terutama jika kita membandingkan nilai yang sama.
3. Kinerja: Dari segi kinerja, untuk jumlah kasus yang kecil, perbedaan tidak signifikan. Namun, untuk banyak kasus switch bisa lebih efisien karena cara implementasinya di belakang layar.

2.4.    Buatlah dokumentasi gambar flowchart dari Latihan 2.1.!

Tanya: "Pilih A atau B:"

Switch Input

Kasus B 'B'

Kasus A 'A'

Cetak: "Anda perlu kurangi main game"

Cetak: "Anda sudah rajin belajar"

Kasus Default (Bukan A atau B)

Cetak: "Pilihan anda diluar A atau B"

2.4.    Buatlah dokumentasi gambar flowchart dari Latihan 2.2!

Input Tahun (2022)

Input Bulan (1-12)

Apakah Bulan valid (1-12) ?

True

Apakah jumlah hari valid ?

True

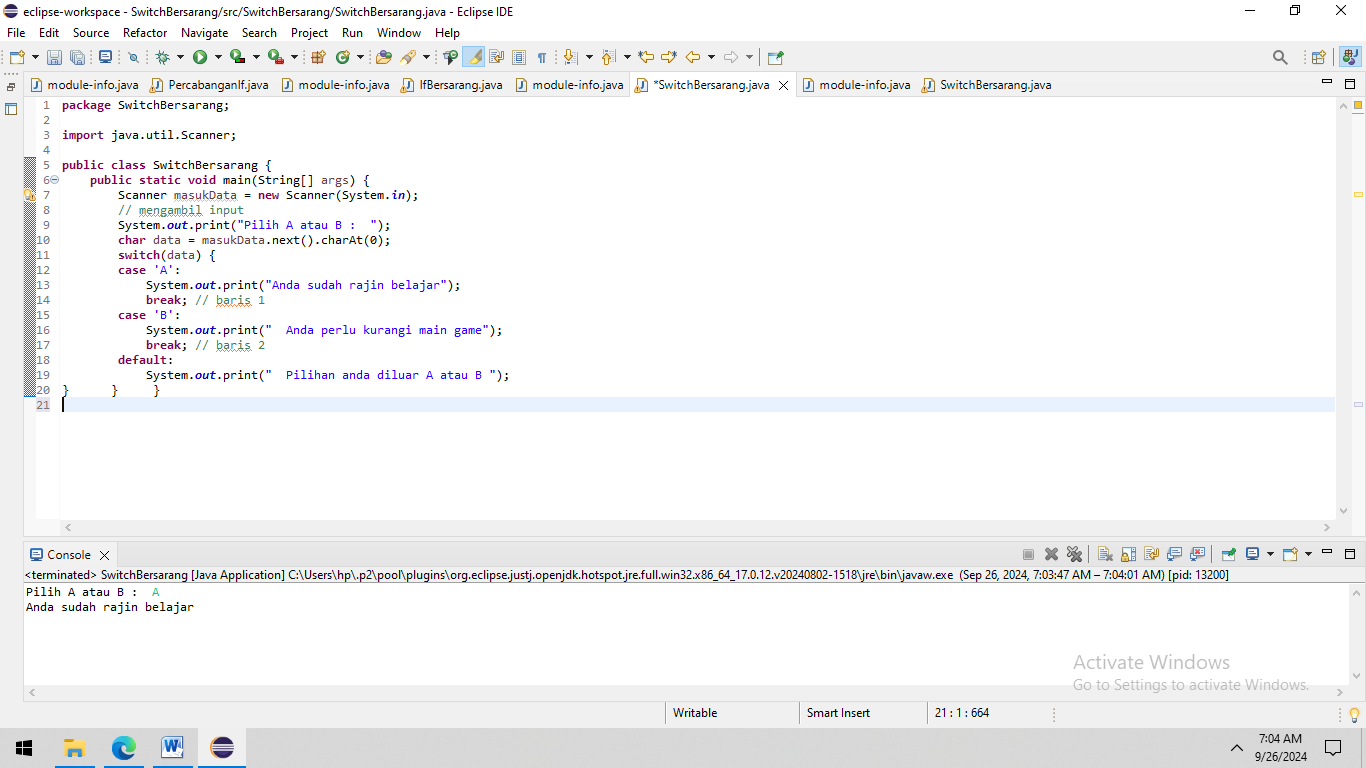
Tampilkan Jumlah Hari di Bulan

**[No2.1] Analisis dan Argumentasi**

1. Saya mengusulkan permasalahan ini dapat diatasi dengan cara menyusun Kode program untuk menghapus break pada baris yang sesuai untuk melihat bagaimana perubahan ini memengaruhi hasil keluaran.
2. Alasan solusi ini karena untuk melihat bagaimana perubahan ini memengaruhi hasil keluaran.
3. Perbaikan kode program dengan cara membuat Kode program untuk menghapus break pada baris yang sesuai

**[No2.1 ] Penyusunan Algoritma dan Kode Program**

1. Algoritma
2. Mulai
3. Inisialisai
4. Menampilkan pesan
5. Ambil Input pengguna
6. Selesai
7. Kode program dan luaran
8. Screenshot Kode program, hasil luaran, dan Komentar



1. Analisa luaran yang dihasilkan.

Luaran sudah sesuai dengan program yang disusun. Dan ketika memasukan pilihan A dan keluar hasil bahwa Anda sudah rajin belajar.

**[No2.1] Kesimpulan   
(PILIH SALAH SATU ANDA INGIN MEMBAHAS DENGAN CARA ANALISA/ EVALUASI / KREASI)**

1. **Analisa**

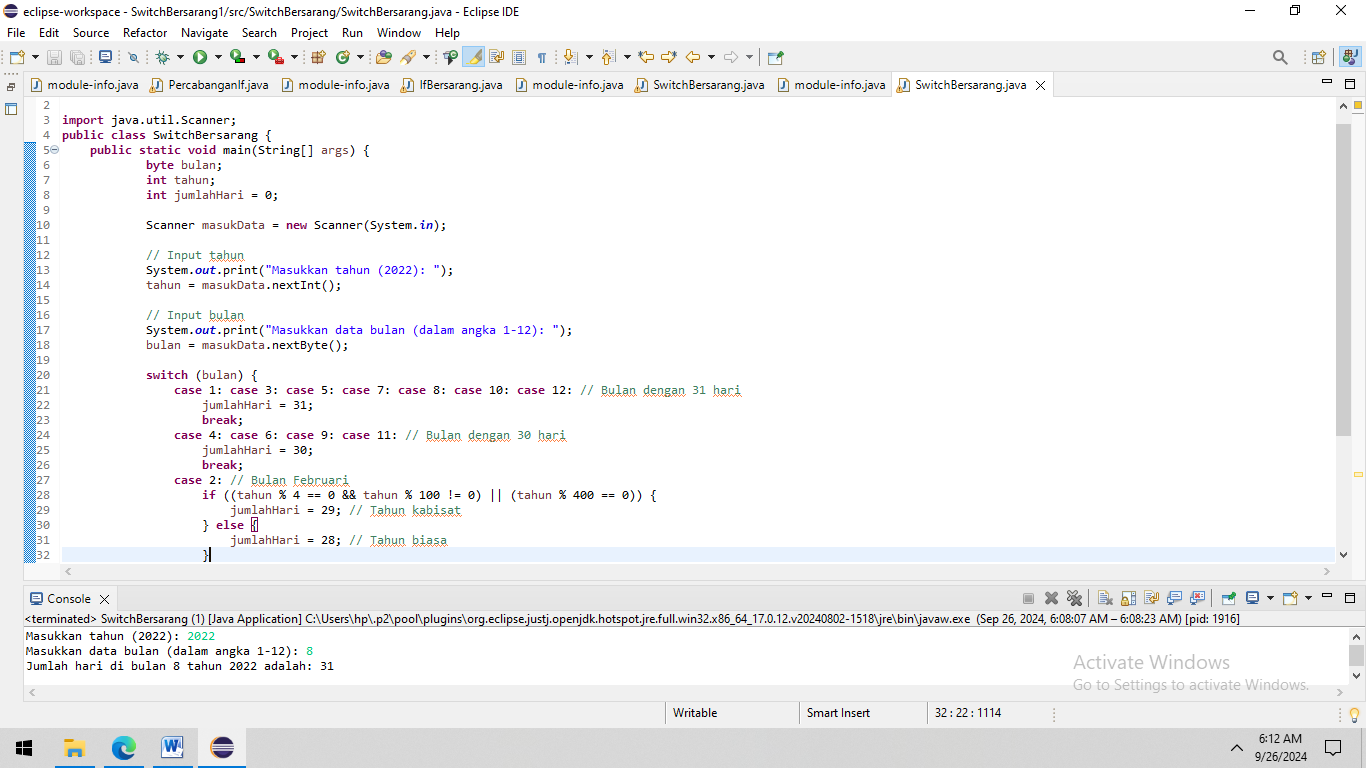
Pada program itu saya menggunakan bentuk Switch untuk menangani pilihan pengguna antara opsi 'A' dan 'B'. Program ini meminta pengguna untuk memasukkan pilihan dan memberikan umpan balik berdasarkan input yang diterima, dengan memeriksa input dan memastikan hanya satu karakter yang diterima, program dapat mencegah kesalahan input dari pengguna.

**[No2.2] Analisis dan Argumentasi**

1. Saya mengusulkan permasalahan ini dapat diatasi dengan cara menyusun Kode program dengan menambahkan hingga bulan ke-12 .
2. Alasan solusi ini karena agar dapat menangani semua bulan dari bulan ke-1 hingga 12 dan menampilkan jumlah hari yang benar.
3. Perbaikan kode program dengan cara membuat Kode program dengan menambahkan case untuk bulan yang tersisa di bagian switch dengan modifikasi ini, program sekarang dapat menangani semua bulan dari 1 hingga 12.

**[No2.2 ] Penyusunan Algoritma dan Kode Program**

1. Algoritma
2. Mulai
3. Deklarasi Variabel
4. Input Tahun
5. Input Bulan
6. Validasi Bulan
7. Menghitung jumlah hari
8. Selesai
9. Kode program dan luaran
10. Screenshot Kode program, hasil luaran, dan Komentar



1. Analisa luaran yang dihasilkan.

Luaran sudah sesuai dengan program yang disusun. Dan menampilkan data bulan dan jumlah hari yang benar ketika ditampilkan.

**[No2.2] Kesimpulan   
(PILIH SALAH SATU ANDA INGIN MEMBAHAS DENGAN CARA ANALISA/ EVALUASI / KREASI)**

1. **Analisa**

Pada program ini menghitung dan menampilkan jumlah hari dalam bulan tertentu untuk tahun yang diberikan sehingga menampilkan jumlah yang benar. Ini mencakup penanganan tahun kabisat untuk bulan Februari, memastikan akurasi dalam hasil. Pada program ini menggunakan struktur kontrol seperti switch untuk membuat logika program mudah dibaca dan dipahami.

**Refleksi :**

Pengalaman belajar yang didapat yakni mendapatkan pelajaran dan pemahaman mengenai pemrograman yakni IF dan SWITCH yang ada di java meskipun ada yang masih belum dipahami. Pengetahuan yang di dapatkan yakni dapat mengetahui bagaimana cara kerja yang ada pada IF dan SWITCH . Serta mendapat pengetahuan juga mengenai Coding . Tantangan yang saya dihadapi yakni agak kesusahan dalam melakukan coding dan terkadang terjadi error dan agak kesusahan juga dalam pembuatan Flowchart.