|  |  |  |
| --- | --- | --- |
| **Nama & NPM** | **Topik:** | **Tanggal:** |
| **Muhammad Ariq Hafidl Fatanah  G1F024014** | **Operator Java** | **14 September 2024** |

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

1. Uraikan permasalahan dan variabel

Tuliskan kembali soal:

public class OperatorAritmatika{   
   public static void main(String[] args)  {  
     // deklarasi nilai  
        int a = 20, b = 3;  
     //operator aritmatika   
        System.out.println("a: " +a);   
        System.out.println("b: " +b);   
        System.out.println("a + b = "  + (a + b));   
}   }

**Luaran:**

a: 20  
b: 3  
a + b = 23

Diketahui dari soal : variabel int a = 20 dan b = 3, dimana operator aritmatika yang digunakan dalam program adalag penjumlahan (a+b) menghasilkan luaran 23 karena (20+3).

**Latihan 1**

Susun kode Java untuk perhitungan dengan ekspresi (2\*3 + 6 / 2 - 4). Simpulkan urutan prioritas operator yang dijalankan ekspresi tersebut!

**[No.1] Analisis dan Argumentasi**

1. Saya mengusulkan permasalahan ini dapat diatasi dengan cara

* Mendeklarasikan nilai int a,b,c,d.
* Memulai operator aritmatika (2\*3 + 6 / 2 - 4). Dimana :

1. a= 2
2. b= 3
3. c= 4
4. d= 6
5. Alasan solusi ini karena setiap operasi aritmatika harus mendeklarasikan dahulu varibel dan nilainya bertujuan untuk menghasilkan operasi aritmatika yang benar
6. Perbaikan kode program dengan cara:

Deklarasikan variabel a, b, c, dan d sebagai integer dengan nilai:

a = 2

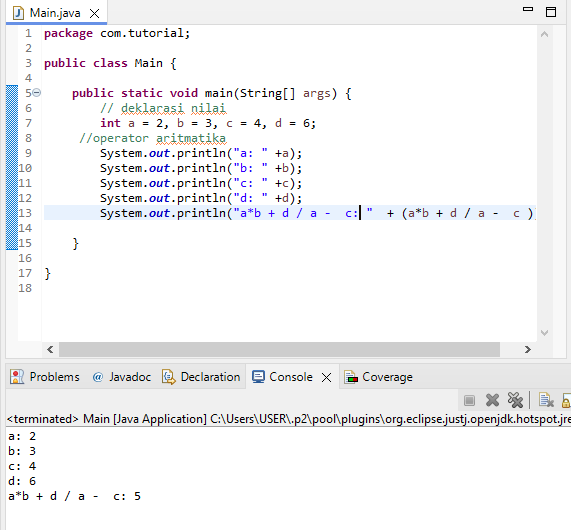
b = 3

c = 4

d = 6

**[No.1 ] Penyusunan Algoritma dan Kode Program**

1. Algoritma Operaot Aritamatika
2. Mulai
3. Deklarasikan variabel a, b, c, dan d sebagai integer
4. Cetak nilai a, b, c, dan d.
5. Hitung hasil dari operasi aritmatika
6. Cetak hasil dari perhitungan operasi tersebut.
7. Selesai
8. Kode program dan luaran



1. Screenshot/ Capture potongan kode dan hasil luaran

Beri komentar pada kode yang di Screenshot

1. Analisa luaran yang dihasilkan:

Program di atas adalah sebuah program pengoperasian Aritmatika dimana Luaran sudah sesuai dengan program yang disusun. Tipe data yang ditampilkan telah sesuai dengan kebutuhan dan permintaan data.

**[No.1] Kesimpulan   
(PILIH SALAH SATU ANDA INGIN MEMBAHAS DENGAN CARA ANALISA/ EVALUASI / KREASI)**

1. **Analisa**
2. Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!
3. Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Jawaban Analisa:

1. Pada program itu saya menggunakan variable int a,b,c,d karena saya akan menginput nilai dari keempat variable tersembut kedalam operasi Aritmatika, nantinya variabel tersebut akan disubstitusikan kedalam nilai yang sudah di deklarasikan. Kesimpulan dari prioritas urutan perhitungan sama halnya dengan operasi Aritmatika pada umumnya yaitu,\* dan / yang didahulukan pada perhitungan dan dilanjutkan dengan + dan -.
2. Alasan saya mengambil keputusan program tersebut adalah didasari dari ketentuan pengoperasian aritmatika itu sendiri

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

1. Uraikan permasalahan dan variabel

Tuliskan kembali soal:

public class OperatorPenugasan {  
    public static void main(String[] args) {  
      // deklarasi nilai  
       int a = 20, b = 3;  
       //operator penugasan   
        b += a;  
        System.out.println("Penambahan : " + b);  
    }  
}

**Luaran:**  
Penambahan : 23

**Latihan 2.**  
Susun kode untuk menampilkan perhitungan dengan operator ( -=, \*=,  /=, %=)! Simpulkan hasilnya!

**[No.2] Analisis dan Argumentasi**

1. Saya mengusulkan permasalahan ini dapat diatasi dengan cara memahami operasi operator penugasan dan bagaimana setiap operasi mengubah nilai variabel.
2. Alasan solusi ini karena penggunaan operator penugasan seperti +=, -=, \*=, /=, dan %= memungkinkan kode menjadi lebih ringkas dan efisien

**[No.2 ] Penyusunan Algoritma dan Kode Program**

1. Algoritma Operasi Operator Penugasan:
2. Mulai

1. Deklarasikan dua variabel a dan b dengan nilai awal:

a = 20

b = 3

1. Penambahan:

Tambahkan nilai a ke b dengan operator +=.

Cetak hasil penambahan b += a.

1. Pengurangan:

Kurangkan nilai a dari b dengan operator -=.

Cetak hasil pengurangan b -= a.

1. Perkalian:

Kalikan nilai b dengan a menggunakan operator \*=.

Cetak hasil perkalian b \*= a.

1. Pembagian:

Bagi nilai b dengan a menggunakan operator /=.

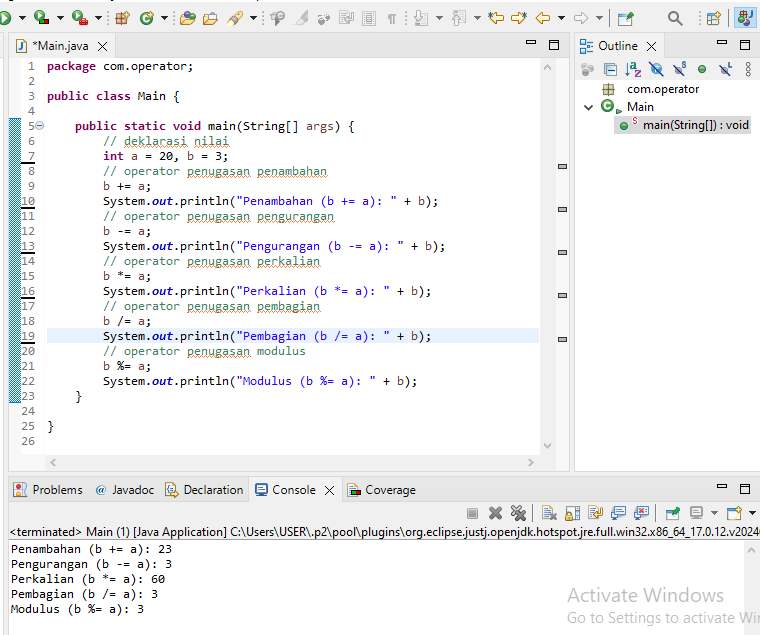
Cetak hasil pembagian b /= a.

1. Modulus:

Hitung sisa pembagian b dengan a menggunakan operator %=.

Cetak hasil modulus b %= a.

1. Selesai
2. Kode program dan luaran



1. Screenshot/ Capture potongan kode dan hasil luaran

Beri komentar pada kode yang di Screenshot

1. Analisa luaran yang dihasilkan:

Luaran yang dihasilkan sesuai dengan program yang sudah dibuat, dimana nilai dari variable dipengaruhi oleh operasi penugasan sebelumnya, jadi jika ingin mereset nilai variable yang ada harus ditambahkan b=3;, agar nilai varibel tidak berubah mengikuti hasil perhitungan operasi penugasan sebelumnya. Akan tetapi saya tidak mereset nilai variable b tersebut sehingga menghasilkan luaran seperti gambar diatas.

**[No.2] Kesimpulan   
(PILIH SALAH SATU ANDA INGIN MEMBAHAS DENGAN CARA ANALISA/ EVALUASI / KREASI)**

1. **Analisa**
2. Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!
3. Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

**Jawaban Analisa:**

Program tersebut sudah saya analisa bahwa setiap operasi penugasan yang telah dilakukan akan merubah nilai dari variable di awal, contohnya pada saat operasi penugasan penambahan (+=) nilai variable yang diambil dari operasi tersebut dalah nilai a = 20, b=3

Dimana akan menghasilkan nilai = 23. Hasil dari operasi penugasan penjumlahan tersebut berpengaruh pada operasi penugasan selanjutnya yaitu operasi penugasan pengurangan

(-=) dimana nantinya nilai b yang diambil bukan lagi nilai dari variable b=3 melainkan nilai

dari hasil operasi penugasan sebelumnya yaitu = 23. Namun jika ingin mengambil nilai

variable b=3, kita dapat mereset nilai variable b dengan cara *b=3;* . Maka nilai variabel b akan

tereset kembali.

**[No. 3] Identifikasi Masalah:**

1. Uraikan permasalahan dan variabel

Tuliskan kembali soal:

public class OperatorRelasional {  
    public static void main(String[] args) {  
        int nilaiA = 12;  
        int nilaiB = 4;  
        boolean hasil;

        System.out.println(" A = " + nilaiA + "\n B = " + nilaiB);  
        // apakah A lebih besar dari B?  
        hasil = nilaiA > nilaiB;  
        System.out.println("\n Hasil A > B = "+ hasil);

        // apakah A lebih kecil dari B?  
        hasil = nilaiA < nilaiB;  
        System.out.println("\n Hasil A < B = "+ hasil);

        // apakah A lebih besar samadengan B?  
        hasil = nilaiA >= nilaiB;  
        System.out.println("\n Hasil A >= B = "+ hasil);

        // apakah A lebih kecil samadengan B?  
        hasil = nilaiA <= nilaiB;  
        System.out.println("\n Hasil A <= B = "+ hasil);

        // apakah nilai A sama dengan B?  
        hasil = nilaiA == nilaiB;  
        System.out.println("\n Hasil A == B = "+ hasil);

        // apakah nilai A tidak samadengan B?  
        hasil = nilaiA != nilaiB;  
        System.out.println("\n Hasil A != B = "+ hasil);  
    }  
}

**Latihan 3**

Susunlah perintah kode dengan operator relasional (<, >, <=, >=, =, ==, !=) untuk nilai a dan b yang menghasilkan luaran TRUE!

**[No.3] Analisis dan Argumentasi**

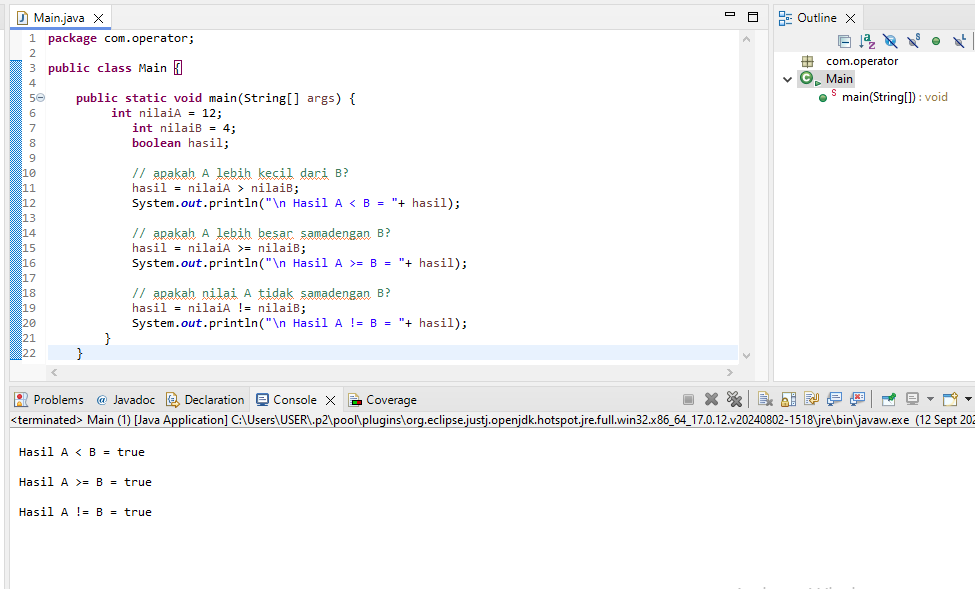
1. Saya mengusulkan permasalahan ini dapat diatasi dengan cara menentukan operator relasional mana yang dapat menghasilkan luaran TRUE
2. Alasan solusi ini karena untuk **membandingkan** dua nilai dan membuat keputusan berdasarkan perbandingan tersebut.

**[No.3 ] Penyusunan Algoritma dan Kode Program**

1. Algoritma

Algoritma Operasi Relasional:

1. Mulai
2. Inisialisasi nilai nilaiA dengan 12 dan nilaiB dengan 4
3. Tampilkan nilai nilaiA dan nilaiB
4. Bandingkan nilai menggunakan operator relasional (cari yang manghasilkan luaran TRUE)
5. Selesai
6. Kode program dan luaran



a) Screenshot/ Capture potongan kode dan hasil luaran

Beri komentar pada kode yang di Screenshot

b) Analisa luaran yang dihasilkan:

Luaran sudah sesuai dengan program yang disusun. Hasil dari operasi relasional sesuai dengan program yang saya tulis dan menghasilkan nilai TRUE

**(PILIH SALAH SATU ANDA INGIN MEMBAHAS DENGAN CARA ANALISA/ EVALUASI / KREASI)**

1. **Analisa**

a) Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!

b) Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Jawaban Analisa:

Kode ini menunjukkan bagaimana berbagai operator relasional digunakan untuk

membandingkan dua nilai (nilaiA dan nilaiB):

* Operator > memeriksa apakah nilaiA lebih besar dari nilaiB.
* Operator < memeriksa apakah nilaiA lebih kecil dari nilaiB.
* Operator >= memeriksa apakah nilaiA lebih besar atau sama dengan nilaiB.
* Operator <= memeriksa apakah nilaiA lebih kecil atau sama dengan nilaiB.
* Operator == memeriksa apakah nilaiA sama dengan nilaiB.
* Operator != memeriksa apakah nilaiA tidak sama dengan nilaiB.

Kode dan algoritma ini memungkinkan untuk mengevaluasi bagaimana dua nilai berhubungan dan memfasilitasi pengambilan keputusan atau logika program berdasarkan hasil perbandingan tersebut.

**[No. 4] Identifikasi Masalah:**

1) Uraikan permasalahan dan variabel

Tuliskan kembali soal:

public class operator {  
    public static void main(String[] args) {  
        // deklarasi nilai  
              int a = 5;   
                
              System.out.println("a: " +a);   
              System.out.println("b: " + (a++));      
}    }

**Luaran:**  
a: 5  
b: 5

Latihan 4.

4.1. Susunlah kode program untuk menghasilkan luaran nilai a = 5 dan b = 6 dengan pre/post increment dan pre/post decrement.

4.2. Simpulkan hasil perbandingan Anda (pre/post increment, pre/post decrement)!

**[No.4] Analisis dan Argumentasi**

1. Saya mengusulkan permasalahan ini dapat diatasi dengan cara mengetahui cara operasi Operator Increment dan Decrement: ++, -- dalam prosesnya, sehingga kita dapa menulikan program sesuai tanpa terjadi error.
2. Alasan solusi ini karena operasi Operator Increment dan Decrement: ++, -- adalah operasi yang memerlukan konsentrasi yang cukup agar nilai dari varibael sesuai dengan yang kita inginkan.

**[No.4 ] Penyusunan Algoritma dan Kode Program**

1. Algoritma

Algoritma adalah langkah-langkah penyelesaian masalah.  
Algoritma Operator Increment dan Decrement: ++, -- :

1. Mulai

2. Post-Increment

* Deklarasi variabel a dengan nilai 5
* Tampilkan nilai dari a
* Tampilkan nilai dari a++ (post-increment: gunakan nilai a lalu tambahkan 1 ke a)
* Tampilkan nilai a setelah post-increment (nilai a sekarang sudah ditambah 1)

3. Pre-Increment

* Deklarasi variabel b dengan nilai 5
* Tampilkan nilai dari b
* Tampilkan nilai dari ++b (pre-increment: tambahkan 1 ke b lalu gunakan nilai b)
* Tampilkan nilai b setelah pre-increment (nilai b sekarang sudah ditambah 1)

4. Post-Decrement

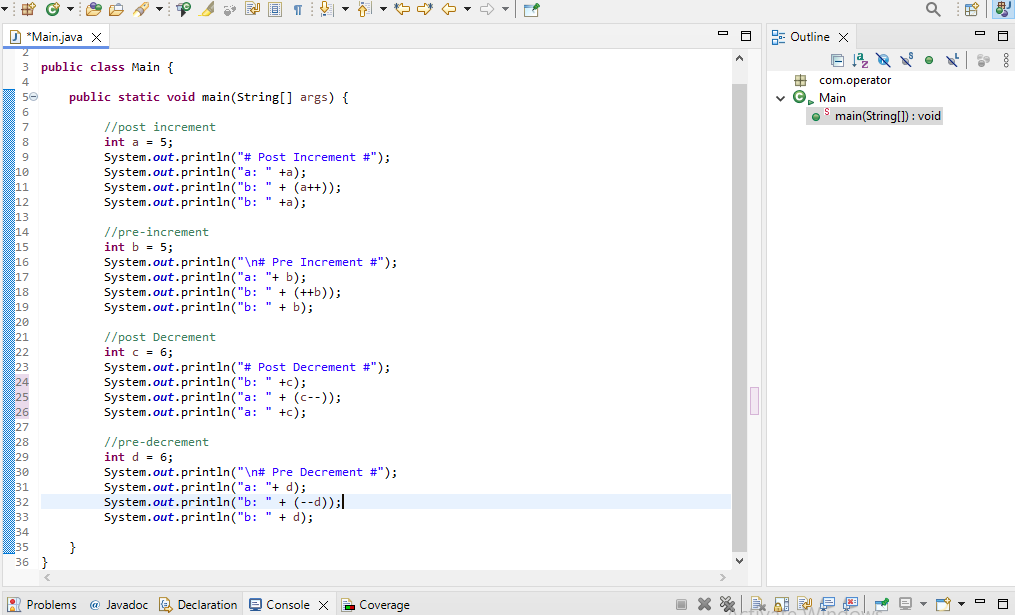
* Deklarasi variabel c dengan nilai 6
* Tampilkan nilai dari c
* Tampilkan nilai dari c-- (post-decrement: gunakan nilai c lalu kurangi 1 dari c)
* Tampilkan nilai c setelah post-decrement (nilai c sekarang sudah dikurangi 1)

5. Pre-Decrement

* Deklarasi variabel d dengan nilai 6
* Tampilkan nilai dari d
* Tampilkan nilai dari --d (pre-decrement: kurangi 1 dari d lalu gunakan nilai d)
* Tampilkan nilai d setelah pre-decrement (nilai d sekarang sudah dikurangi 1)

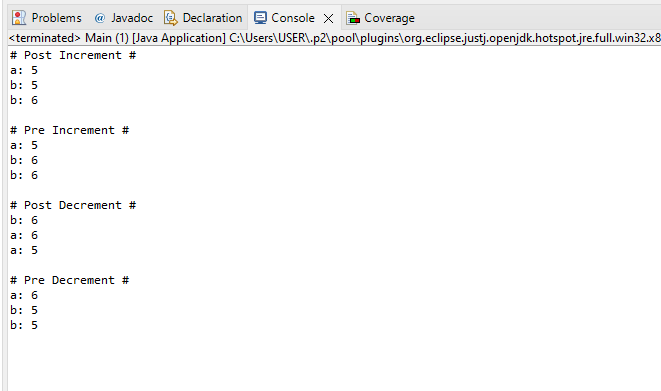
6. Selesai

1. Kode program dan luaran



1. Screenshot/ Capture potongan kode dan hasil luaran

Beri komentar pada kode yang di Screenshot

1. Analisa luaran yang dihasilkan :  
   

Luaran sudah sesuai dengan program yang disusun. Menghasilkan nilai sesuai dengan operasi yang dilakukan dan sudah sesuai dengan perintah yang diberikan.

**[No 4] Kesimpulan   
(PILIH SALAH SATU ANDA INGIN MEMBAHAS DENGAN CARA ANALISA/ EVALUASI / KREASI)**

1. **Analisa**
2. Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!
3. Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Jawaban Analisa:

Tujuan dari kode program adalah untuk mendemonstrasikan penggunaan operator pre-increment, post-increment, pre-decrement, dan post-decrement dalam Java. Permasalahan yang harus dipecahkan adalah memahami bagaimana setiap operator mempengaruhi nilai variabel sebelum dan setelah operasi dilakukan, serta bagaimana hasilnya ditampilkan. Dengan memahami perbedaan dan efek dari pre dan post increment/decrement, kita dapat mengendalikan bagaimana nilai variabel digunakan dan dimodifikasi dalam program dengan lebih efektif.

**[No. 5] Identifikasi Masalah:**

1) Uraikan permasalahan dan variabel

Tuliskan kembali soal:

public class operator {  
    public static void main(String[] args) {  
        // deklarasi nilai  
        boolean a = true;  
        boolean b = false;  
          
        System.out.println("Hasil logika (a && b) : " + (a && b));  
}    }

**Luaran:**  
Hasil logika (a && b) : false

**Latihan 5**  
Susun kode program dengan mengubah nilai a dan b untuk menghasilkan luaran true dengan operator && dan  operator | |.  Beri kesimpulan!

**[No.5] Analisis dan Argumentasi**

1. Saya mengusulkan permasalahan ini dapat diatasi dengan cara mengetahui konsep operator tersebut. Untuk menghasilkan keluaran true dengan operator && (AND) dan operator || (OR), perlu memahami bagaimana kedua operator ini bekerja:

* Operator && (AND): Mengembalikan nilai true hanya jika kedua operand bernilai true.
* Operator || (OR): Mengembalikan nilai true jika salah satu atau kedua operand bernilai true.

1. Alasan solusi ini karena operasi ini dapat mengubah nilai Boolean dengan cara mengetahui nilai dari variable yang ada.

**[No.5 ] Penyusunan Algoritma dan Kode Program**

Algoritma

Algoritma adalah langkah-langkah penyelesaian masalah.  
Algoritma untuk program yang menggunakan operator logika && dan ||:

1. Mulai program
2. Deklarasikan dua variabel boolean, a dan b

* Atur a = true
* Atur b = true

1. Gunakan operator && untuk mengevaluasi apakah kedua variabel a dan b bernilai true

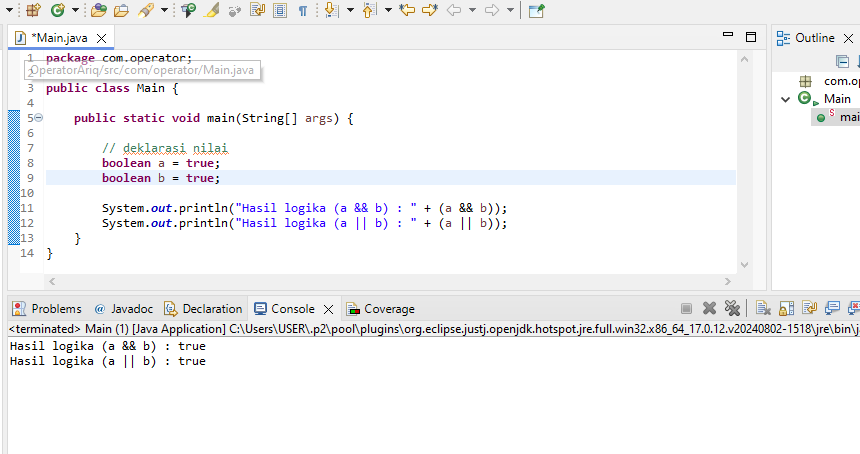
* Jika a AND b bernilai true, cetak hasil true
* Jika tidak, cetak hasil false

1. Ubah nilai variabel b menjadi false
2. Gunakan operator || untuk mengevaluasi apakah salah satu dari variabel a atau b bernilai true

* Jika a OR b bernilai true, cetak hasil true
* Jika tidak, cetak hasil false

1. Akhiri program

2. Kode program dan luaran



1. Screenshot/ Capture potongan kode dan hasil luaran

Beri komentar pada kode yang di Screenshot

1. Analisa luaran yang dihasilkan :

Luaran sudah sesuai dengan program yang disusun. Operator && (AND) hanya menghasilkan true jika kedua operand bernilai true. Jika salah satu atau keduanya false, hasilnya false. Operator || (OR) menghasilkan true jika salah satu atau kedua operand bernilai true. Hanya jika keduanya false, hasilnya false.

Dalam kasus ini, kedua luaran menghasilkan true karena syarat untuk operator AND dan OR dipenuhi.

**[No 5] Kesimpulan   
(PILIH SALAH SATU ANDA INGIN MEMBAHAS DENGAN CARA ANALISA/ EVALUASI / KREASI)**

1. **Analisa**
2. Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!
3. Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Jawaban Analisa:

Tujuan dari kode program adalah untuk memahami bagaimana kedua operator ini bekerja dalam menghasilkan keluaran TRUE atau FALSE. Permasalahan melibatkan penggunaan operator logika AND (&&) dan OR (||) untuk menentukan hasil evaluasi logika berdasarkan nilai boolean dari dua variabel (a dan b). Dengan menggunakan kombinasi nilai boolean yang berbeda pada variabel a dan b, kita dapat mengendalikan keluaran berdasarkan aturan operator logika. Dasar keputusan ini sepenuhnya berdasarkan pada sifat matematis dari operator logika yang digunakan dalam pemrograman.

**[No. 6] Identifikasi Masalah:**

1) Uraikan permasalahan dan variabel

Tuliskan kembali soal:

public class OperatorKondisi{  
   public static void main( String[] args ){  
      String status = "";  
      int nilai = 80;   
      status = (nilai > 60)?"Lulus":"Gagal";  
      System.out.println( status );  
}    }

**Luaran:**  
Lulus

**Latihan 6**  
Susun kode program! Dengan informasi berikut:  
Diketahui nama variabel Jam = 12  
Apabila jam < 12 maka tampil "Selamat Pagi", apabila jam > 12 maka tampil "Selamat Malam".

**[No.6] Analisis dan Argumentasi**

1. Saya mengusulkan permasalahan ini dapat diatasi dengan cara mengetahui konsep operator tersebut. Ternary operator memungkinkan kita untuk mengekspresikan pilihan dengan cara yang lebih deklaratif, yang berarti kita bisa langsung melihat kondisi dan pilihan yang terlibat dalam satu baris, tanpa harus menyebar di beberapa baris kode.
2. Alasan solusi ini karena operasi ini membuat kode lebih ringkas dan mudah dibaca ketika berhadapan dengan kondisi sederhana. Namun, jika logikanya lebih rumit, maka menggunakan if-else bisa menjadi pilihan yang lebih baik. \

**[No.6 ] Penyusunan Algoritma dan Kode Program**

Algoritma

Algoritma adalah langkah-langkah penyelesaian masalah.  
Algoritma untuk program yang menggunakan operator ternary

1. Mulai Program.
2. Inisialisasi variabel:

* Buat variabel status dengan tipe data String dan set awalnya ke nilai kosong ("").
* Buat variabel jam dengan tipe data int dan beri nilai awal 8.

1. Pengecekan Kondisi Pertama:

* Jika nilai jam kurang dari 12:

Set variabel status menjadi "Selamat Pagi".

* Jika nilai jam tidak kurang dari 12:

Set variabel status menjadi "Selamat Malam".

* Tampilkan hasil dari variabel status ke layar.

1. Pengecekan Kondisi Kedua:

* Jika nilai jam lebih besar dari 12:

Set variabel status menjadi "Selamat Pagi".

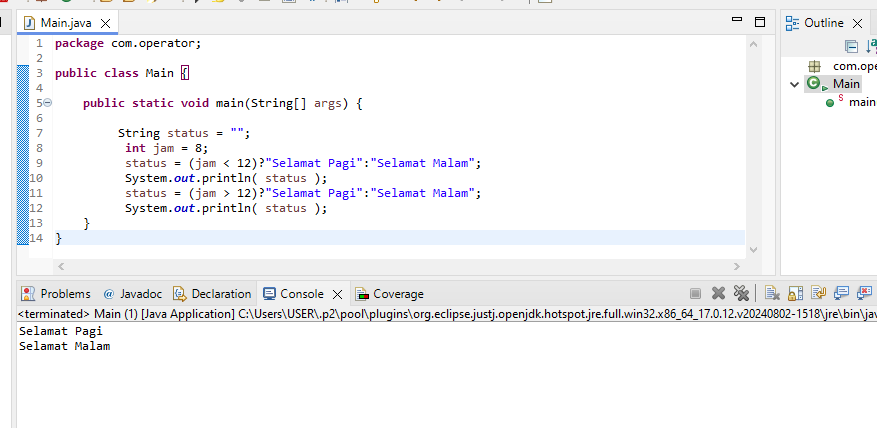
* Jika nilai jam tidak lebih besar dari 12:

Set variabel status menjadi "Selamat Malam".

* Tampilkan hasil dari variabel status ke layar.

1. Selesai.

2. Kode program dan luaran



1. Screenshot/ Capture potongan kode dan hasil luaran

Beri komentar pada kode yang di Screenshot

1. Analisa luaran yang dihasilkan :

Luaran sudah sesuai dengan program yang disusun. Penggunaan ternary operator dapat membuat kode terlihat lebih ringkas dan jelas. Ternary operator ideal untuk kondisi yang singkat dan langsung.. Program pertama kali menampilkan "Selamat Pagi" karena jam 8 adalah pagi hari. rogram kedua kali menampilkan "Selamat Malam" karena kondisi jam > 12 tidak terpenuhi, sehingga masuk ke blok else, yaitu "Selamat Malam".

**[No 6] Kesimpulan   
(PILIH SALAH SATU ANDA INGIN MEMBAHAS DENGAN CARA ANALISA/ EVALUASI / KREASI)**

**a) Analisa**

1. Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!
2. Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Jawaban Analisa:

Tujuan dari kode program adalah harus memeriksa nilai variabel jam untuk menentukan apakah akan menampilkan pesan "Selamat Pagi" atau "Selamat Malam", berdasarkan kondisi apakah jam lebih kecil atau lebih besar dari 12. Kode program telah berhasil menyelesaikan permasalahan yang diminta, yaitu menampilkan pesan yang sesuai berdasarkan nilai jam. Hasil output menampilkan "Selamat Pagi" saat jam kurang dari 12 dan "Selamat Malam" saat jam tidak lebih besar dari 12. Operator ternary digunakan karena kasus kali ini sederhana, yaitu hanya memerlukan pengecekan satu kondisi dan melakukan satu tindakan, yang membuat penggunaan ternary operator lebih efisien daripada penggunaan if-else biasa.

**[No. 7] Identifikasi Masalah:**

1) Uraikan permasalahan dan variabel

Tuliskan kembali soal:

public class operatorBitwise {  
    public static void main(String[] args) {  
        int a = 10;  
        int b = 7;  
        int hasil;  
            
        hasil = a & b;  
        System.out.println("Hasil dari a & b : " + hasil );    
            
        hasil = a | b;  
        System.out.println("Hasil dari a | b : " + hasil );    
            
        hasil = a ^ b;  
        System.out.println("Hasil dari a ^ b : " + hasil );     
}   }

**Luaran:**  
Hasil dari a & b : 6  
Hasil dari a | b : 7  
Hasil dari a ^ b : 1

**Latihan 7**  
Susun kode tambahan dari Contoh 7 untuk melakukan perhitungan dengan operator (>>, <<). Hubungkan hasil luaran dengan perhitungan manual bilangan biner!

**[No.7] Analisis dan Argumentasi**

1. Saya mengusulkan permasalahan ini dapat diatasi dengan cara mengetahui konsep dasar-dasar Bitwise dan Shift Operators. Untuk:

* AND (&), Menghasilkan 1 hanya jika kedua bit pada posisi yang sama adalah 1.
* OR (|): Menghasilkan 1 jika salah satu bit pada posisi yang sama adalah 1.
* XOR (^): Menghasilkan 1 jika bit pada posisi yang sama berbeda.
* Left Shift (<<): Menggeser bit ke kiri. Setiap geseran ke kiri setara dengan perkalian dengan 2.
* Right Shift (>>): Menggeser bit ke kanan. Setiap geseran ke kanan setara dengan pembagian dengan 2 (pembulatan ke bawah).

1. Alasan solusi ini karena operasi ini dapa digunakan dalam kompresi data, enkripsi, dan manipulasi bit untuk penghematan ruang dan efisiensi. Shift operators (<<, >>) sering digunakan untuk operasi perkalian dan pembagian dengan pangkat dua, yang lebih cepat dibandingkan operasi aritmatika konvensional.

**[No.7 ] Penyusunan Algoritma dan Kode Program**

Algoritma

Algoritma adalah langkah-langkah penyelesaian masalah.  
Algoritma untuk program yang menggunakan operator ternary

1. Mulai Program.
2. Inisialisasi Variabel:

* Deklarasikan variabel a dengan nilai 10 (dalam biner: 1010).
* Deklarasikan variabel b dengan nilai 7 (dalam biner: 0111).
* Deklarasikan variabel hasil yang akan digunakan untuk menyimpan hasil operasi.

1. Operasi Bitwise:

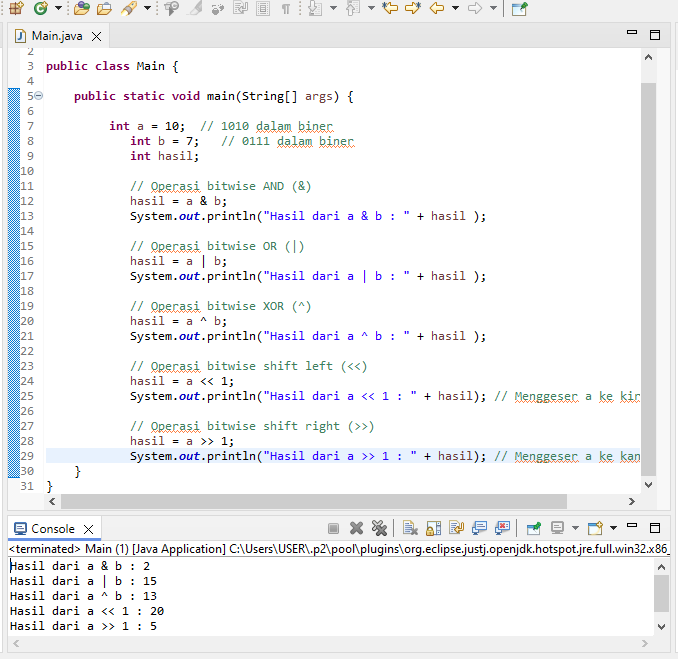
* Bitwise AND (&):
* Hitung a & b.
* Operasi 1010 & 0111 menghasilkan 0010 (nilai dalam desimal: 2).
* Tampilkan hasil: "Hasil dari a & b : 2".
* Bitwise OR (|):
* Hitung a | b.
* Operasi 1010 | 0111 menghasilkan 1111 (nilai dalam desimal: 15).
* Tampilkan hasil: "Hasil dari a | b : 15".
* Bitwise XOR (^):
* Hitung a ^ b.
* Operasi 1010 ^ 0111 menghasilkan 1101 (nilai dalam desimal: 13).
* Tampilkan hasil: "Hasil dari a ^ b : 13".

1. Operasi Bitwise Shift:

* Shift Kiri (<<):
* Hitung a << 1.
* Geser bit 1010 ke kiri satu posisi menghasilkan 10100 (nilai dalam desimal: 20).
* Tampilkan hasil: "Hasil dari a << 1 : 20".
* Shift Kanan (>>):
* Hitung a >> 1.
* Geser bit 1010 ke kanan satu posisi menghasilkan 0101 (nilai dalam desimal: 5).
* Tampilkan hasil: "Hasil dari a >> 1 : 5".

1. Selesai.

2. Kode program dan luaran



1. Screenshot/ Capture potongan kode dan hasil luaran

Beri komentar pada kode yang di Screenshot

1. Analisa luaran yang dihasilkan :

Luaran sudah sesuai dengan program yang disusun. Kode program menghasilkan hasil operasi bitwise dan shift dengan benar: operasi AND antara 10 (1010 dalam biner) dan 7 (0111 dalam biner) menghasilkan 2 (0010 dalam biner), operasi OR menghasilkan 15 (1111 dalam biner), dan XOR menghasilkan 13 (1101 dalam biner). Untuk operasi shift, geseran kiri 10 sebanyak satu bit menghasilkan 20 (10100 dalam biner), sementara geseran kanan 10 sebanyak satu bit menghasilkan 5 (0101 dalam biner). Hasil tersebut menunjukkan bagaimana operasi bitwise dan shift memanipulasi bit individu dan memberikan hasil yang efisien untuk pengolahan data biner.

**[No 7] Kesimpulan   
(PILIH SALAH SATU ANDA INGIN MEMBAHAS DENGAN CARA ANALISA/ EVALUASI / KREASI)**

**a) Analisa**

1. Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program!
2. Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini?

Jawaban Analisa:

Dapat disimpulkan bahwa operasi bitwise dan shift sangat efektif dalam manipulasi data pada tingkat bit. Kode program berhasil mengilustrasikan penggunaan operator bitwise AND, OR, XOR, serta operator shift kiri dan kanan dengan hasil yang sesuai dengan perhitungan manual. Hasil dari operasi bitwise memberikan nilai yang sesuai dengan aturan bitwise untuk setiap jenis operasi, sedangkan operasi shift menunjukkan perubahan pada nilai ketika bit digeser ke kiri atau kanan.

Keputusan untuk menggunakan operator bitwise dan shift didasarkan pada kebutuhan untuk efisiensi dalam pengolahan data serta kontrol langsung pada level bit. Operator ini memungkinkan manipulasi data dengan cepat dan tepat

**Refleksi**

Dari diri saya pribadi sudah sedikit memahami materi untuk operasi penggunanaan bebearpa operator yang saya lakukan. Pembelajaran mengenai berbagai operator dalam pemrograman termasuk operator aritmatika, penugasan, relasional, increment dan decrement, boolean logika, kondisional (ternary), serta bitwise menunjukkan betapa pentingnya pemahaman mendalam tentang setiap jenis operator untuk menyusun dan mengelola kode program yang efektif dan efisien. Operator aritmatika menyediakan dasar untuk perhitungan matematika, sedangkan operator penugasan memungkinkan kita untuk menyimpan dan mengupdate nilai variabel dengan cara yang singkat dan efisien.