|  |  |  |
| --- | --- | --- |
| **Nama & NPM** | **Topik:** | **Tanggal:** |
| **Annas Zam Zam**  **G1F022013** | **Operator** | **09 September 2022** |

**1.Operator Aritmatika: +,-,\*,/**

|  |
| --- |
| **Identifikasi Masalah:**  1.1. Rekomendasikan perbaikan kode agar program Contoh 1 dapat berjalan! 1.2. Tambahkan baris untuk menampilkan perhitungan dengan operator ( -, \*, /, %) pada Contoh 1!  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));  //menampilkan hasil penjumlahan }   }  **Luaran:**         Exception in thread "main" java.lang.Error: Unresolved compilation problems:      Syntax error on token ""a + b = "", AssignmentOperator expected after this token     The left-hand side of an assignment must be a variable  Program diatas sebelumnya belum bisa dijalankan di karenakan belum begitu betul perlu beberapa penambahan variable tipe data seperti float c = 20;  Selanjutnya yaitu penambahan di system.out.println yaitu : (“a + b = “ + ( a + b ));  (“a - b = “ + ( a - b ));  (“a \* b = “ + ( a \* b ));  (“a / b = “ + ( c / b ));  (“a % b = “ + ( a % b )); |
| 1. Uraikan permasalahan dan variable   menampilkan perhitungan dengan operator  +,-,\*,/,+% seperti latihan 2.1   1. Rincikan sumber informasi yang relevan (jika ada)   Informasi yang saya simpulkan harus memasukan tipe data yang benar dan variable jika tida ingnin mendapatkan kesalahan dalam menjalankan program   1. Uraikan rancangan solusi yang diusulkan (jika ada).   Solusi yang saya usulkan adalah memperbaiki variable yang kurang lengkap supaya bisa berjalan dengan benar   1. Analisis susunan solusi, parameter solusi (jika ada).   Fungsi dari float c = 20; adalah untuk mendpatkan hasil bagi dalam variable   |  | | --- | | **Penyusunan Algoritma dan Kode Program** | | 1. Rancang desain solusi 2. Susunan algoritma (jika ada)   -Memasukkan kode program  -Analisa kesalahan yang terjadi  -Perbaiki kode program  -menambahkan tipe data dan variable  -lalu jalankan program   1. Analisa prinsip pemprograman   Meperhatikan seditail mungkin kesalahan yang terjadi dalam pembuatan kode program supaya tidak terjadi kesalahan dalam menjalankan program   1. Tuliskan kode program dan luaran 2. Beri komentar pada kode   //\* menambahkan aritmatika untuk menjalankan kode program   1. Analisa sintaks, semantik, dan alur logika pemprograman   -Memasukankode kode aritmatika yang benar dan jelas  -di pastikan bahwa mendapatkan luaran yang di inginkan  -jika menambahkan aritmatika seperti system.out.println (“a + b = “ + ( a + b )); akan mendapatkan luaran a + = 23  -jika menambahkan aritmatika seperti system.out.println (“a - b = “ + ( a - b ));  akan mendapatkan luaran a – b = 17  -jika menambahkan aritmatika seperti system.out.println (“a \* b = “ + ( a \* b ));  akan mendapatkan luaran a \* b = 60  -jika menambahkan aritmatika seperti system.out.println (“a / b = “ + ( c / b ));  akan mendapatkan luaran a / b = 6.6666665  -jika menambahkan aritmatika seperti system.out.println (“a % b = “ + ( a % b ));  akan mendapatkan luaran a % b = 2   1. Uraikan luaran yang dihasilkan   Luaran yang dihasilkan adalah hasil dari kode aritmatika yang di masukkan dan diproses berbentuk anggka dan huruf   1. Screenshot/ Capture potongan kode dan hasil luaran     Kode program yang di masukkan    Luaran yang dihasilkan  **Kesimpulan**  1 ) Analisa  a.mengamati kode program dengan teliti kenapa bisa terjadi kesalahan lalu memperbaiki kode program dengan kode yang sebenarnya ,membuat susunan algoritma suapaya berurutan,dan memasukan kode aritmatika untuk menjalankan kode program dan mendapatkan luaran yang diinginkan  b.dasar dari pengambilan keputusan ini yaitu untuk dapat menjalankan perintah yang di kerjakan dan mendapatkan luaran yang benar  **2.Operator Penugasan: =**  **Identifikasi Masalah:**  2.1.  Tambahkan baris Contoh 2 untuk menampilkan perhitungan dengan operator ( -=, \*=,  /=, %=)! 2.2.  Berikan argumentasi tentang perbedaan luaran dan waktu eksekusi Contoh 1 dan Contoh 2!  public class OperatorPenugasan {     public static void main(String[] args) {       // deklarasi nilai        int a = 20, b = 3;        //operator penugasan          b += a;  //melakukan perhitungan penjumlahan         System.out.println("Penambahan : " + b);  // menampilkan hasil perhitungan penjumlahan     } }  **Luaran:** Penambahan : 23  Kode program di atas belum benar di karenakan masih terjadi kesalahan dalam mejalankan kode program  Melakukan perbaikan seperti :  System.out.println (“penambahan : “ + b);  System.out.println (“pegurangan : “ + b);  System.out.println (“perkalian : “ + b);  System.out.println (“pembagian : “ + b);  System.out.println (“sisa : “ + b);  1). Uraikan permasalahan dan variable  Menampilkan perhitungan dengan operator ( -=, \*=,  /=, %=)!  2). Rincikan sumber informasi yang relevan (jika ada)  Informasi yang saya berikan jika ingin menampilkan perhitungn maka perlu menambhakan operator penugasan  3). Uraikan rancangan solusi yang diusulkan (jika ada).  Solusi dari saya menambahkan penugasan seusai dengan perintah 2.1  4).Analisis susunan solusi, parameter solusi (jika ada)  Dengan menambhakan operator (-=,\*=,/=,%=) supaya mendapatkan perhitungan yang diingnkan  **Penyusunan algoritma dan kode pemrograman**   1. Rancang desain solusi Susunan algoritma (jika ada   -Mengidentifikasi masalah  -Memasukan kode program  -Menambahkan penugasan kode program  -Mendapatakn hasil pemrograman   1. Uraikan luaran yang dihasilkan   Luaran yang di hasilkan berupa penjumlahan dari penugasan yang telah di masukan  2.1.  Tambahkan baris Contoh 2 untuk menampilkan perhitungan dengan operator ( -=, \*=,  /=, %=)!   1. Screenshot/ Capture potongan kode dan hasil luaran     Hasil kode program yang ditambahakn penugasan berserta hasil dari pemrograman  2.2.  Berikan argumentasi tentang perbedaan luaran dan waktu eksekusi Contoh 1 dan Contoh 2!   1. Screenshot/ Capture potongan kode dan hasil luaran   Contoh 1      Kode program sebelum di rubah menghasilkan luaran seperti diatas  Contoh 2      Kode program yang sudah dirubah dan menghasilkan luaran seperti diatas  Kesimpulan perubahan yang terjadi  Tentu saja banyak sekali perubahan yang terjadi pada kedua kode program tersebut yang pertama kode program contoh 1 hanya menghasilkan luaran penambahan = 23 saja karena belum di tambahkan penugasan  Yang kedua dua kode program contoh 2 menghasilkan luaran lebih dari contoh kode program pertama seperti penambhan = 2,pengurangan = 3,perkalian = 60,pembagian = 3,sisa 3  Untuk kesimpulannya contoh 2 lah kode program yang benar untuk membuat kode program penugasan dengan hasil yang di ingginkan  **3.Operasi rasional: ++,--**  **Identifikasi masalah:**  3.1.  Ubahlah nilai A = 4 dan B = 4 pada Contoh 3. Simpulkan perubahan yang terjadi!  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);     } }  **Luaran:**  A = 12  B = 4   Hasil A > B = true  Hasil A < B = false  Hasil A >= B = true  Hasil A <= B = false  Hasil A == B = false  Hasil A != B = true  Dalam variable ini tidak begitu banyak kesalahan yang terjadi namun ada satu kesalahan dalam pemasukan nilai yaitu mengubah int nilaiA = 12; menjadi int nilaiA = 4;  1). Uraikan permasalahan dan variable  Permasalahan yang terjadi mengubah nilai A = 4 dan B = 4  **Penyusunan algoritma dan kode program**  1).Rancang desain solusi Susunan algoritma (jika ada)  -Memasukkan kode program  -Analisa kesalahan yang terjadi  -Perbaiki kode program  -mengubah nilai menjadi nilai yang sebenarnya  -lalu jalankan program   1. Screenshot/ Capture potongan kode dan hasil luaran       Kode luaran yang telah dirubah   * 1. Kesimpulan   Pada tampilan kode program yang dibuat banyak sekali perubahan yang  Terjadi seperti : A = 12 setelah diubah dan mendapat luaran a = 4  Hasil A < B = true Setelah diubah mendapat luaran  Hasil A > B = false  Hasil A <= B = false setelah diubah mendapat luaran  Hasil A <= B = true  Hasil A == B = false setelah diubah mendapat luaran  Hasil A == B = true  Hasil A != B = true setelah diubah mendapat luaran  Hasil A != B = false  Jadi dapat di simpulkan bahawa jika mengubah nilai A = 4 dan B = 4 pada variable maka akan menghasilkan luaran yang berbeda dari semula menjadi luaran yang baru | |  | |

**4.Operator increment dan decrement: ++,--**

**Identifikasi masalah:**

4.1. Berikan saran operasi apa yang diperlukan (pre/post increment, pre/post decrement) agar Contoh 4 menghasilkan nilai a = 5 dan b = 6?   
4.2. Simpulkan hasil eksperimen Anda!

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

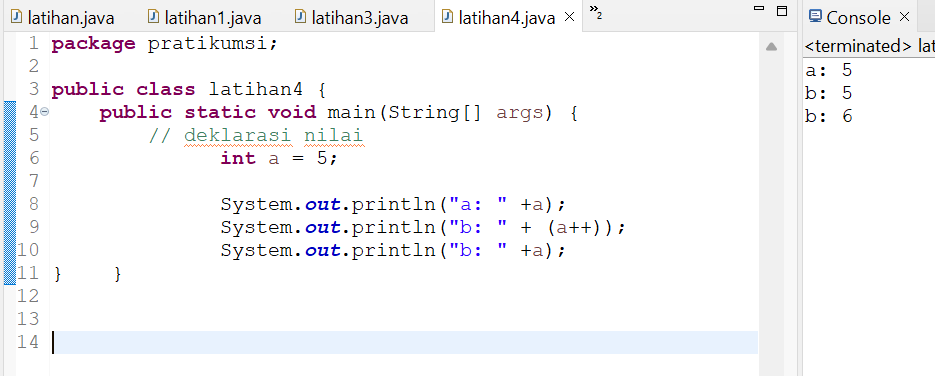
ada beberapa penambahan yang akan di lakukan pada variable diatas

4.1 saran operasi yang diperlukan

penambahan pada System.***out***.println("b: " +a); untuk dapat menghasilkan:

a = 5 dan b = 6

Screenshot/ Capture potongan kode dan hasil luaran



kode pemrograman yang telah dirubah menghasilkan data seperti diatas

4.2.kesimpulan

Dapat saya simpulkan bahwa menambahkan System.***out***.println("b: " +a); maka akan menghasilkan kode luaran yang diinginkan seperti diatas

1. **Operator logika: &&,||,!**

**Identifikasi masalah**

5.1.  Rekomendasikan berapa nilai a dan b apabila ingin menghasilkan luaran true dengan operator && dan  operator | | ?    
5.2.  Berikan kesimpulan dari latihan 5.1.

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

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

Ada beberapa perbaikan yang harus dilakuka untuk menjalankan kode program

**Penyusunan algoritma dan kode pemrograman**

1. Rancang desain solusi Susunan algoritma (jika ada

-Mengidentifikasi masalah

-Memasukan kode program

-Menambahkan nilai true pada bolean

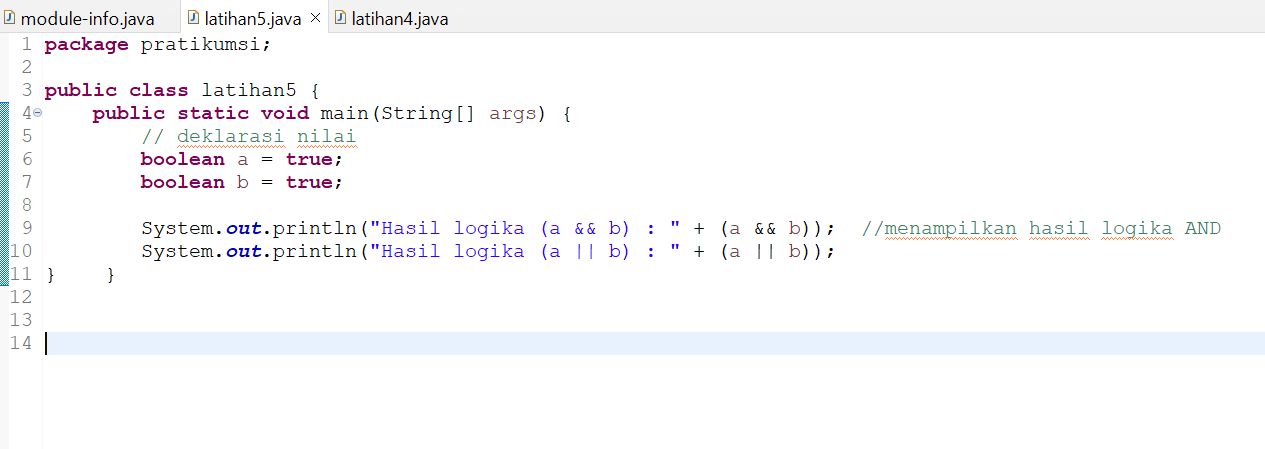
-Mendapatakn hasil pemrograman

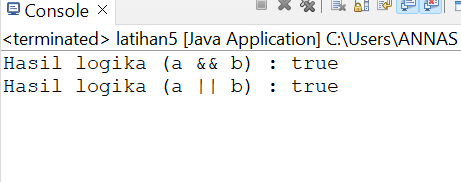
* 1. berapa nilai a dan b apabila ingin menghasilkan luaran true dengan operator && dan  operator | | ?

yaitu : System.***out***.println("Hasil logika (a || b) : " + (a || b));

dan juga mengganti bolean b menjadi = true;

Screenshot/ Capture potongan kode dan hasil luaran





Kode hasil dari penambambahan nilai untuk menghasilkan nilai true dengan operator &&,||

* 1. kesimpulan

jadi kesimpulannya adalah jika ingin mendapatkan nilai true maka perlu mengganti nilai dari bolean menjadi true bukan false dengan menggunakan :

system.out.println.(“Hasil logika ( a || b )); untuk dapat menghasilkan luaran true

1. **operator kondisional ternary : ? :**

**Identifikasi masalah**

6.1 Rekomendasikan apa bentuk tanda operator agar nilai = 60 memenuhi untuk Lulus !

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

Jika ingin mendapatkan nilai 60 bisa lulus maka ada nilai yang harus di ubah

**Penyusunan algoritma dan kode pemrograman**

1. Rancang desain solusi Susunan algoritma (jika ada

-Mengidentifikasi masalah

-Mengubah kode program

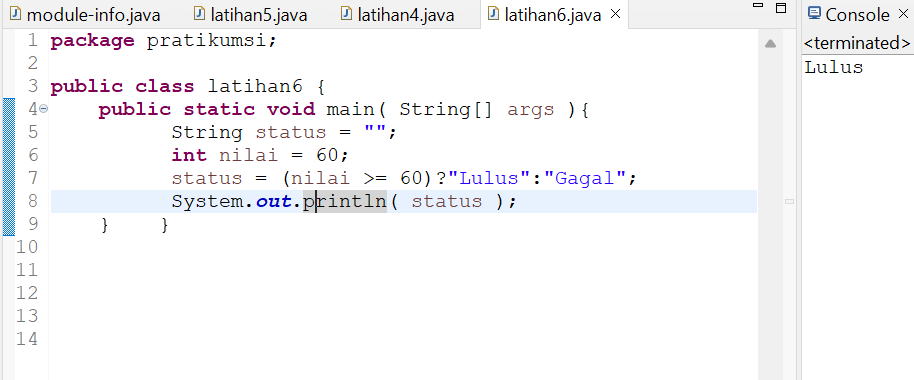
-Menambahkan nilai yang sebenarnya agas bisa lulus

-Mendapatakn hasil pemrograman

* 1. tanda operator agar dapat nilai 60 memenuhi untuk lulus

yaitu mengganti : int nilai = 80; menjadi int nilai = 60; agar mendapatkan luaran lulus dengan nilai 60 dan juga menambahkan tanda => sesudah (nilai >= 60)?"Lulus":"Gagal";

Screenshot/ Capture potongan kode dan hasil luaran



Kode program yang sudah di perbaiki bisa menghasilkan luaran lulus

**7.operator bitwise: &,|,~,<<,>>,,>>>**

**Identifikasi masalah**

Evaluasi penyebab hasil ~a = -11 ? Buktikan jawaban Anda dalam perhitungan biner!

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 );    
            
        hasil = ~a;  
        System.out.println("Hasil dari ~a : " + hasil );    
            
        hasil = a >> 1;  
        System.out.println("Hasil dari a >> 1 : " + hasil );    
            
        hasil = b << 2;  
        System.out.println("Hasil dari b << 2 : " + hasil );  
}   }

**Luaran:**   
Hasil dari a & b : 6  
Hasil dari a | b : 7  
Hasil dari a ^ b : 1  
Hasil dari ~a : -11  
Hasil dari a >> 1 : 3  
Hasil dari b << 2 : 28

Kode program perhitungan biner

Penyusunan algoritma dan kode pemrograman

1. Rancang desain solusi Susunan algoritma (jika ada

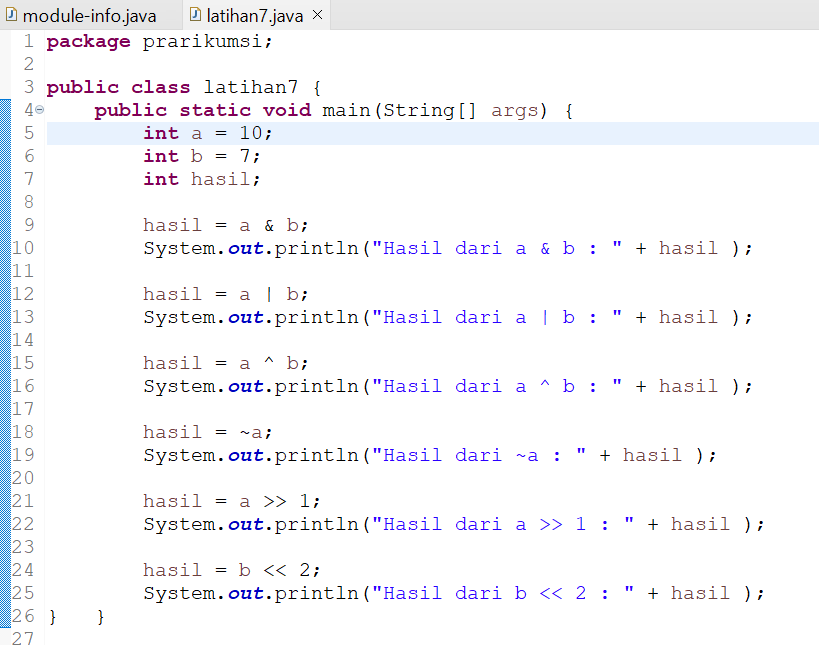
-Mengidentifikasi masalah

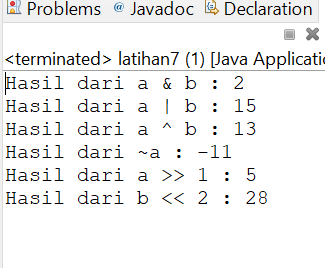
-Menghitung biner

-evaluasi

-bukti jawaban dalam perhitungan biner

Screenshot/ Capture potongan kode dan hasil luaran





Evaluasi

1. Evaluasi input, proses, dan luaran yang dihasilkan! (jika ada)

Di dalam variable a = 10; di situ di hitung dari 0 sampai 10 maka 0 di hitung 1 selanjutnya, 1 di hitung 2 sampai anggka 10 karena itu NOT (~) maka kebalikan dari anggka tersebut jika positif angka tersebut menjadi negatif.