|  |  |  |
| --- | --- | --- |
| **Nama & NPM** | **Topik:** | **Tanggal:** |
| **WINDA DWI SALISTIANNA**  **G1F024004** | **TIPE DATA** | **27 AGUSTUS 2024** |
| ***Unit 1: Deklarasi dasar Java*** | | |
| Salin dan tempel potongan kode ini ke Eclipse atau Jdoodle.  public class KelasKu {   private static void main(String[] args) {     System.out.println("Halo Mahasiswa UNIB) }   }  Luaran: Exception in thread "main" java.lang.Error: Unresolved compilation problem:    String literal is not properly closed by a double-quote    Syntax error, insert ";" to complete BlockStatements  **Latihan 1:**  1.1  Evaluasi penyebab kesalahan terjadi dan perbaiki agar program dapat berjalan! 1.2. Ubah teks yang ditampilkan program menjadi nama lengkap Anda. 1.3. Tambahkan baris System.out.println(“”}; untuk diisi dengan data alamat, dan jenis kelamin.  **JAWAB :**  ***Sebuah gambar berisi teks, cuplikan layar, software, komputer  Deskripsi dibuat secara otomatisSebuah gambar berisi teks, cuplikan layar, software, komputer  Deskripsi dibuat secara otomatis1.1***   ***Evaluasi penyebab kesalahan terjadi dan perbaiki agar program dapat berjalan!***   * ***Kode Eror***   ***Sebuah gambar berisi teks, cuplikan layar, software, komputer  Deskripsi dibuat secara otomatis***  1.1 ada beberapa kesalahan pada kode yang menyebabkan program tidak dapat berjalan:   1. **Kesalahan penulisan tanda kutip:** Pada baris System.out.println("Halo Mahasiswa UNIB), tanda kutip penutup (") setelah teks "Halo Mahasiswa UNIB" tidak ada, sehingga menyebabkan kesalahan sintaks. 2. **Nama metode main:** Metode main harus bersifat publik (public), agar JVM dapat menjalankannya. Selain itu, kurung kurawal penutup pada metode main tidak ditutup dengan benar. 3. **Kurung kurawal penutup kelas:** Kurung kurawal penutup untuk kelas KelasKu tidak ditutup dengan benar.  * ***Perbaikan Data***   Kode:  package windaa;  public class kelasku {  public static void main(String[] args) {    System.***out***.println("Halo Mahasiswa UNIB");  } }    Penjelasan:  Penjelasan perbaikan:   * Menambahkan tanda kutip penutup pada System.out.println("Halo Mahasiswa UNIB");. * Mengubah private static void main menjadi public static void main agar metode main dapat diakses oleh JVM. * Menutup kurung kurawal pada metode main dan kelas KelasKu dengan benar.   Dengan perbaikan ini, program akan berjalan dan mencetak "Halo Mahasiswa UNIB" ke layar.  ***1.2 Ubah teks yang ditampilkan program menjadi nama lengkap And*a.**  Kode :  package windaa;  public class kelasku {  public static void main(String[] args) {    System.***out***.println("WINDA DWI SALISTIANNA");  } }  Penjelasan kode:  Kode di atas mendefinisikan sebuah kelas bernama `KelasKu` yang berisi metode `main`. Metode `main` ini adalah titik masuk utama untuk menjalankan program Java. Pada metode `main`, terdapat satu perintah untuk mencetak teks menggunakan `System.out.println()`. Teks yang akan dicetak adalah nama lengkap Anda, "WINDA DWI SALISTIANNA". Ketika program ini dijalankan, konsol akan menampilkan nama tersebut. Kelas dan metode telah ditulis dengan struktur yang benar, memungkinkan program untuk dieksekusi tanpa kesalahan.  Sebuah gambar berisi teks, cuplikan layar, Font, software  Deskripsi dibuat secara otomatisOutput kode :  ***1.3. Tambahkan baris System.out.println(“”}; untuk diisi dengan data alamat, dan jenis kelamin.***  Kode :  package windaa;  public class latihan1 {  public static void main(String[] args) {  System.***out***.println("Nama :winda dwi salistianna");  System.***out***.println("Alamat :Muara Bangka Hulu");  System.***out***.println("Jenis Kelamin :Perempuan");  }  }  Penjelasan kode :  Kode tersebut mendefinisikan sebuah kelas bernama `KelasKu` dengan metode `main` yang merupakan titik awal eksekusi program. Di dalam metode `main`, terdapat tiga perintah `System.out.println()` yang digunakan untuk mencetak informasi pribadi Anda ke layar konsol. Baris pertama mencetak nama lengkap Anda, "winda dwi salistianna," baris kedua mencetak alamat Anda, "Muara Bangka Hulu," dan baris ketiga mencetak jenis kelamin Anda, "Perempuan." Saat program dijalankan, ketiga informasi ini akan ditampilkan berurutan di layar.  Output : | | |
| ***Unit 2: Variabel dan tipe data*** | | |
| Java memiliki 8 tipe data primitif dan non-primitif (baca slide atau video materi).  **Contoh 2:**Apabila diketahui data berikut   1. 5 2. ‘L’ 3. “mobil” 4. 5.0 5. 5.0f 6. -5   **Latihan 2:** 2.1. Rekomendasikan tipe data yang tepat dari data Contoh 2. 2.2.  Simpulkan karakteristik penggunaan setiap tipe data!  **JAWAB :**  ***2.1. Rekomendasikan tipe data yang tepat dari data Contoh 2.***  package windaa;  public class latihan2 {  public static void main(String[] args) {    int a = 5;  System.***out***.println(a);  int a2 = 44;  System.***out***.println(a2);  float b = 12.345679f;  System.***out***.println(b);  double c = 12.34567891d;  System.***out***.println(c);  byte d = 0;  System.***out***.println(d);  }  }Untuk menentukan tipe data yang tepat berdasarkan contoh data yang diberikan, berikut adalah analisisnya:  1. 5  Tipe data yang tepat: Integer  Alasan: Ini adalah bilangan bulat tanpa desimal, sehingga tipe datanya adalah `int`.  2. ‘L’  Tipe data yang tepat: Character  Alasan: Karakter tunggal, sehingga tipe datanya adalah `char`.  3. “mobil”  Tipe data yang tepat: String  Alasan: Ini adalah teks yang terdiri dari lebih dari satu karakter, sehingga tipe datanya adalah `String`.  4. 5.0  Tipe data yang tepat: Double  Alasan: Ini adalah bilangan desimal, sehingga tipe datanya adalah `double`.  5. 5.0f  Tipe data yang tepat: Float  Alasan: Ini adalah bilangan desimal dengan akhiran `f` yang menandakan bahwa tipe datanya adalah `float`.  6. -5  Tipe data yang tepat: Integer  Alasan: Ini adalah bilangan bulat negatif, sehingga tipe datanya adalah `int`.  Jadi, rekomendasi tipe data yang tepat untuk setiap contoh adalah sebagai berikut:  - 5: `int`  - ‘L’: `char`  - “mobil”: `String`  - 5.0: `double`  - 5.0f: `float`  - -5: `int`  ***2.2.  Simpulkan karakteristik penggunaan setiap tipe data!***  1. Integer (`int`)  - Karakteristik: Digunakan untuk menyimpan bilangan bulat (baik positif maupun negatif) tanpa komponen desimal.  - Contoh Penggunaan: Nilai-nilai seperti jumlah, hitungan, atau indeks dalam perulangan.  2. Character (`char`)  - Karakteristik:Digunakan untuk menyimpan satu karakter tunggal, seperti huruf, angka, atau simbol.  - Contoh Penggunaan: Menyimpan huruf individu, seperti inisial, atau elemen dari string.  3. String (`String`)  - Karakteristik: Digunakan untuk menyimpan teks yang terdiri dari satu atau lebih karakter. Merupakan kumpulan dari karakter.  - Contoh Penggunaan: Menyimpan kalimat, kata, nama, atau data teks lainnya.  4. Double (`double`)  - Karakteristik:Digunakan untuk menyimpan bilangan desimal dengan presisi ganda, yang membutuhkan lebih banyak memori dibanding `float`. Cocok untuk perhitungan numerik yang memerlukan presisi tinggi.  - Contoh Penggunaan: Operasi aritmatika yang memerlukan akurasi tinggi, seperti perhitungan ilmiah atau keuangan.  5. Float (`float`)  - Karakteristik: Digunakan untuk menyimpan bilangan desimal dengan presisi tunggal. Lebih hemat memori dibanding `double`, tetapi dengan akurasi yang lebih rendah.  - Contoh Penggunaan: Perhitungan numerik yang tidak memerlukan presisi tinggi, seperti grafik atau game.  **Contoh 3**: Salin dan tempel potongan kode ini ke Eclipse atau Jdoodle  public class TipeData {  public static void main(String args[]) { int a = 55555555555; byte b = 4444444444; float c = 12.345678910f; double d = 12.345678910d; char e = 'abc';  System.out.println(a); System.out.println(b); System.out.println(c); System.out.println(d); System.out.println(e); }    }  Luaran:  /TipeData.java:3: error: integer number too large   int a = 55555555555;  /TipeData.java:4: error: integer number too large    byte b = 4444444444;  /TipeData.java:7: error: unclosed character literal    char e = abc'; 3 errors  **Latihan 3:**  3.1. Evaluasi penyebab kesalahan pada Contoh 3! 3.2. Rekomendasikan tipe data yang sesuai untuk data tersebut!  **JAWAB :**  ***3.1. Evaluasi penyebab kesalahan pada Contoh 3!***   * **Sebuah gambar berisi teks, cuplikan layar, Font    Deskripsi dibuat secara otomatisKode Error**   Penjelasan kode :  Kode di atas memiliki beberapa kesalahan, seperti penggunaan tipe data `int` untuk angka besar yang melebihi kapasitasnya, penggunaan tipe data `byte` untuk angka yang terlalu besar, penggunaan `float` dengan presisi yang lebih tinggi dari yang didukung, serta penggunaan tipe data `char` untuk menyimpan lebih dari satu karakter. Untuk memperbaikinya, variabel `a` harus diubah menjadi tipe `long`, variabel `b` menjadi `int` atau `long`, sementara `c` sebaiknya tetap `float` tetapi dengan pemahaman tentang batasan presisinya, dan `e` harus diubah menjadi tipe `String` untuk menyimpan lebih dari satu karakter.  ***3.2. Rekomendasikan tipe data yang sesuai untuk data tersebut!***  package windaa;  public class latihan5 {  public static void main(String[] args) {  long a = 55555555555L; // Menggunakan tipe data long untuk angka besar  long b = 4444444444L; // Menggunakan tipe data long untuk angka besar  float c = 12.345678910f; // Menggunakan float dengan presisi terbatas  double d = 12.345678910d; // Menggunakan double untuk presisi lebih tinggi  String e = "abc"; // Menggunakan String untuk lebih dari satu karakter  System.***out***.println(a);  System.***out***.println(b);  System.***out***.println(c);  System.***out***.println(d);  System.***out***.println(e);  }  } | | |
| ***Unit 3 Konversi Tipe Data*** | | |
| **Contoh 4:** Salin dan tempel potongan kode ini ke Eclipse atau Jdoodle  public class KonversiTipeData {    public static void main(String args[]) {    byte x;    int a = 270;    double b = 128.128;    System.out.println("int dikonversi ke byte");    x = (byte) a;    System.out.println("a dan x " + a + " " + x);    System.out.println("double dikonversi ke int");    a = (int) b;    System.out.println("b dan a " + b + " " + a);    System.out.println("double dikonversi ke byte");    x = (byte)b;    System.out.println("b dan x " + b + " " + x); }      }  Luaran:  int dikonversi ke byte  a dan x 270 14  double dikonversi ke int  b dan a 128.128 128  double dikonversi ke byte  b dan x 128.128 -128  **Latihan 4:** 4.1. Rekomendasikan konversi tipe data pada Latihan 2 ke bentuk tipe data lain yang kompatibel. 4.2. Simpulkan alasan jenis konversi tipe data tersebut!  **JAWAB:**  ***4.1. Rekomendasikan konversi tipe data pada Latihan 2 ke bentuk tipe data lain yang kompatibel.***   * ***Kode Tidak Terjadi Error*** * ***Output Kode***   Untuk merekomendasikan konversi tipe data pada Latihan 2 ke bentuk tipe data lain yang kompatibel, mari kita lihat tipe data yang diberikan dan konversinya yang relevan:  1. 5: `int`  - Konversi ke `double`: 5 dapat dikonversi menjadi 5.0.  - Konversi ke `float`: 5 dapat dikonversi menjadi 5.0f.  - Konversi ke `String`: 5 dapat dikonversi menjadi "5".  - Konversi ke `char`: 5 dapat dikonversi menjadi karakter dengan kode ASCII 5, tetapi ini mungkin tidak selalu sesuai dengan penggunaan umum.  2. ‘L’: `char`  - Konversi ke `String`: 'L' dapat dikonversi menjadi "L".  - Konversi ke `int`: 'L' dapat dikonversi menjadi kode ASCII atau Unicode-nya, yang adalah 76.  - Konversi ke `double`: Kode ASCII 76 dapat dikonversi menjadi 76.0.  - Konversi ke `float`: Kode ASCII 76 dapat dikonversi menjadi 76.0f.  3. “mobil”: `String`  - Konversi ke `int`: "mobil" tidak dapat dikonversi langsung ke `int`.  - Konversi ke `double`: "mobil" tidak dapat dikonversi langsung ke `double`.  - Konversi ke `float`: "mobil" tidak dapat dikonversi langsung ke `float`.  - Konversi ke `char`: Konversi ini tidak langsung, tetapi Anda bisa mengakses karakter individu dari string "mobil".  4. 5.0: `double`  - Konversi ke `int`: 5.0 dapat dikonversi menjadi 5.  - Konversi ke `float`: 5.0 dapat dikonversi menjadi 5.0f.  - Konversi ke `String`: 5.0 dapat dikonversi menjadi "5.0".  - Konversi ke `char`: 5.0 tidak dapat dikonversi langsung ke `char`.  5. 5.0f: `float`  - Konversi ke `int`: 5.0f dapat dikonversi menjadi 5.  - Konversi ke `double`: 5.0f dapat dikonversi menjadi 5.0.  - Konversi ke `String`: 5.0f dapat dikonversi menjadi "5.0f".  - Konversi ke `char`: 5.0f tidak dapat dikonversi langsung ke `char`.  Jadi, tipe data mana yang ingin dikonversikan bergantung pada kebutuhan spesifik konversi dan konteks penggunaannya.  ***4.2. Simpulkan alasan jenis konversi tipe data tersebut!***  Konversi tipe data dilakukan untuk memastikan kompatibilitas dan kelancaran operasi antar berbagai tipe data dalam pemrograman. Misalnya, mengonversi `int` ke `double` atau `float` memungkinkan perhitungan yang melibatkan angka desimal tanpa kehilangan presisi, sedangkan konversi dari `char` ke `int` memberikan akses ke nilai numerik dari karakter. Sebaliknya, konversi dari `String` ke tipe numerik tidak langsung karena `String` berisi karakter yang tidak memiliki representasi numerik intrinsik. Konversi ini memungkinkan adaptasi data sesuai kebutuhan operasi dan meningkatkan fleksibilitas dalam pengolahan informasi di dalam program. | | |
| ***Refleksi***  Selama minggu ini, pengalaman belajar saya menekankan pentingnya memahami dan menerapkan konsep-konsep dasar dalam pemrograman, seperti pengelolaan aksesibilitas kelas dan optimasi kode. Saya mendapatkan pemahaman yang lebih mendalam tentang bagaimana pilihan desain, seperti penggunaan kelas public, private, dan protected, mempengaruhi keamanan dan modularitas program. Tantangan terbesar yang saya hadapi adalah menemukan keseimbangan antara fleksibilitas dan keamanan, serta mengoptimalkan kinerja program tanpa mengorbankan kejelasan kode. Pengalaman ini mengajarkan saya bahwa eksplorasi dan eksperimen sangat penting dalam menemukan solusi yang efektif. | | |