|  |  |  |
| --- | --- | --- |
| **Nama & NPM** | **Topik:** | **Tanggal:** |
| **Zaira ayu wandira**  **G1F024055** | **Pengenalan Java dan Tipe Data** | 1. **Agustus 2024** |
| **[No.1] Identifikasi Masalah:** | | |
| * 1. Evaluasi penyebab kesalahan terjadi dan perbaiki agar program dapat berjalan   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. | | |
| **[No.1l] Analisis dan Argumentasi** | | |
| * 1. Saya mengusulkan permasalahan ini dapat diatasi dengan cara Evaluasi penyebab kesalahan terjadi   **Kesalahan:**   1. Tanda kutip ganda pada System.out.println("Halo Mahasiswa UNIB) tidak ditutup dengan benar. Ini menyebabkan kesalahan kompilasi. 2. Modifier akses private seharusnya tidak digunakan pada metode main. Seharusnya menggunakan public static void main(String[] args). 3. Pada akhir pernyataan tidak menggunakan titik koma(;),maka akan muncul kesalahan kompilasi   Perbaikan kode program dengan cara  Perbaikan;  1.menambah kutip penutup  2. menggunakan public static void main(String[] args)  3.mengunakan titik koma (;)  1.2 untuk mengubah teks yang ditampilkan program maka mengganti Halo Mahasiswa UNIB menjadi nama lengkap yaitu Zaira Ayu Wandira  1.3 kita Tambahkan baris System.out.println(“”}; yang petama diisi data alamat dan yang kedua kita Tambahkan baris System.out.println(“”}; lagi dan diisi dengan jenis kelamin | | |
| **[No.1] Penyusunan Algoritma dan Kode Program** | | |
| * 1. Evaluasi penyebab kesalahan terjadi dan perbaiki agar program dapat berjalan   Algoritma:  1).mulai  2).berikan nama class dan file  3). tambahkan method main  4).tambah kan system.out.println  5).menambahkan kata yang akan dideklarasikan  6).run  7).selesai    Luaran sudah sesuai dengan program yang disusun   * 1. Ubah teks yang ditampilkan program menjadi nama lengkap Anda   Algoritma:  1).mulai  2).berikan nama class dan file  3). tambahkan method main  4).tambah kan system.out.println  5).menambahkan nama lengkap  6).run  7).selesai    Luaran sudah sesuai dengan program yang disusun   * 1. Tambahkan baris System.out.println(“”}; untuk diisi dengan data alamat, dan jenis kelamin.   Algoritma:  1).mulai  2).berikan nama class dan file  3). tambahkan method main  4).tambah kan system.out.println  5).menambahkan kata yang akan dideklarasikan  6).run    Luaran sudah sesuai dengan program yang disusun | | |
| **[No.1] Kesimpulan** | | |
| Pada program itu saya menggunakan bentuk kelas public karena modifier akses private seharusnya tidak digunakan. Perbaikan program dengan menambahkan mengunakan titik koma (;) dan menambahkan kutip penutup .jika tidak digunakan pada program java dapat menyebabkan kesalahan kompilasi. | | |
| **[No. 2] Identifikasi Masalah:** | | |
| 2.1. Rekomendasikan tipe data yang tepat dari data Contoh 2.  2.2.  Simpulkan karakteristik penggunaan setiap tipe data! | | |
| **[No.2] Analisis dan Argumentasi** | | |
| * 1. Saya mengusulkan permasalahan ini dapat diatasi dengan cara mengevaluasi masing-masing data dari Contoh 2 dan rekomendasikan tipe data yang sesuai:  1. **5** Tipe data: int Penjelasan: Angka bulat seperti 5 umumnya disimpan sebagai int karena merupakan tipe data primitif yang paling umum digunakan untuk bilangan bulat. 2. **‘L’** Tipe data: char Penjelasan: Karakter tunggal seperti 'L' disimpan dalam tipe data char, yang dirancang untuk menyimpan satu karakter Unicode. 3. **“mobil”** Tipe data: String Penjelasan: Rangkaian karakter atau teks seperti "mobil" disimpan dalam tipe data String, yang merupakan tipe data non-primitif di Java. 4. **5.0** Tipe data: double Penjelasan: Angka desimal seperti 5.0 disimpan dalam tipe data double, yang digunakan untuk angka dengan presisi ganda. 5. **5.0f** Tipe data: float Penjelasan: Angka desimal dengan akhiran f (atau F) seperti 5.0f disimpan dalam tipe data float, yang digunakan untuk angka dengan presisi tunggal. 6. **-5** Tipe data: int Penjelasan: Angka bulat negatif seperti -5 juga disimpan sebagai int.   2.2 Berikut adalah karakteristik dari setiap tipe data di Java:  **Tipe Data Primitif:**   1. **byte**    * Rentang: -128 hingga 127    * Ukuran: 1 byte    * Penggunaan: Memori kecil untuk bilangan bulat yang tidak terlalu besar 2. **short**    * Rentang: -32,768 hingga 32,767    * Ukuran: 2 byte    * Penggunaan: Bilangan bulat yang sedikit lebih besar daripada byte 3. **int**    * Rentang: -2^31 hingga 2^31-1 (sekitar -2.1 miliar hingga 2.1 miliar)    * Ukuran: 4 byte    * Penggunaan: Bilangan bulat umum 4. **long**    * Rentang: -2^63 hingga 2^63-1 (sekitar -9.2 quintillion hingga 9.2 quintillion)    * Ukuran: 8 byte    * Penggunaan: Bilangan bulat yang sangat besar 5. **float**    * Rentang: Sekitar ±3.40282347E+38F    * Ukuran: 4 byte    * Penggunaan: Angka desimal dengan presisi tunggal 6. **double**    * Rentang: Sekitar ±1.79769313486231570E+308    * Ukuran: 8 byte    * Penggunaan: Angka desimal dengan presisi ganda 7. **char**    * Rentang: 0 hingga 65,535 (karakter Unicode)    * Ukuran: 2 byte    * Penggunaan: Menyimpan satu karakter 8. **boolean**    * Nilai: true atau false    * Ukuran: Tidak ditentukan secara spesifik, biasanya 1 bit secara konseptual    * Penggunaan: Menyimpan nilai kebenaran   **Tipe Data Non-Primitif:**   1. **String**    * Tipe: Objek (kelas)    * Penggunaan: Menyimpan urutan karakter dan menyediakan berbagai metode untuk manipulasi teks | | |
| **[No.2 ] Penyusunan Algoritma dan Kode Program** | | |
| Algoritma:  1).mulai  2).berikan nama class dan file  3). tambahkan method main  4).deklarasikan variable  5). Print variable  6).run  7.)selesai    Luaran sudah sesuai dengan program yang disusun | | |
| **[No.2] Kesimpulan** | | |
| dari data yang diberikan,saya mengevaluasi bahwa tipe data int lebih baik digunakan untuk bentuk data bilangan bulat, tipe data char lebih baik digunakan untuk bentuk data Karakter tunggal, tipe data string lebih baik digunakan untuk bentuk data Rangkaian karakter atau teks, tipe data double lebih baik digunakan untuk bentuk data angka desimal, dan tipe data float lebih baik digunakan untuk bentuk data angka desimal dengan akhiran f (atau F) | | |
| **[No. 3] Identifikasi Masalah:** | | |
| 3.1. Evaluasi penyebab kesalahan pada Contoh 3 3.2. Rekomendasikan tipe data yang sesuai untuk data tersebut | | |
| **[No.3l] Analisis dan Argumentasi** | | |
| Saya mengusulkan permasalahan ini dapat diatasi dengan cara mengevaluasi Penyebab Kesalahan  **Evaluasi Penyebab Kesalahan**   1. **int a = 55555555555;**    * Kesalahan:nilai angka integer terlalu panjang(melebihi rentang)    * Penjelasan: Tipe data int memiliki rentang dari -2,147,483,648 hingga 2,147,483,647. Nilai 55555555555 melebihi rentang ini. Untuk menyimpan nilai yang lebih besar, Anda harus menggunakan tipe data long. 2. **byte b = 4444444444;**    * Kesalahan: nilai angka byte terlalu panjang(melebihi rentang)    * Penjelasan: Tipe data byte memiliki rentang dari -128 hingga 127. Nilai 4444444444 jauh melebihi rentang ini. Untuk nilai yang lebih besar, Anda bisa menggunakan tipe data int atau long. 3. **char e = 'abc';**    * Kesalahan: lebih dari satu karakter    * Penjelasan: Tipe data char hanya dapat menyimpan satu karakter. 'abc' adalah lebih dari satu karakter, jadi ini adalah kesalahan. Untuk menyimpan string dari beberapa karakter, gunakan tipe data String.   3.2 Berdasarkan kesalahan dan tipe data yang tersedia, berikut adalah tipe data yang sesuai untuk data tersebut   1. **int a = 55555555555;**    * Rekomendasi Tipe Data: long    * Penjelasan: long memiliki rentang yang lebih besar dibandingkan int dan dapat menyimpan nilai besar seperti 55555555555. Anda harus menambahkan huruf L di akhir angka untuk menunjukkan bahwa itu adalah long. Contoh: long a = 55555555555L;. 2. **byte b = 4444444444;**    * Rekomendasi Tipe Data: long    * Penjelasan: Nilai 4444444444 jauh melebihi rentang byte dan juga melebihi int, sehingga Anda harus menggunakan long. Contoh: long b = 4444444444L;. 3. **float c = 12.345678910f;**    * Rekomendasi Tipe Data: double    * Penjelasan: Tipe data float digunakan untuk angka desimal dengan presisi tunggal. Namun, angka 12.345678910 memiliki lebih banyak digit desimal daripada yang dapat diwakili dengan float. Jika Anda memerlukan presisi lebih tinggi, Anda harus menggunakan double. nilai 12.345678910 akan disimpan dengan lebih akurat dalam tipe double dibandingkan dengan float. 4. **double d = 12.345678910d;**    * Rekomendasi Tipe Data: double    * Penjelasan: double adalah tipe data yang tepat untuk angka desimal dengan presisi ganda. Tidak ada masalah dengan tipe data ini. 5. **char e = 'abc';**    * Rekomendasi Tipe Data: String    * Penjelasan: Untuk menyimpan lebih dari satu karakter, Anda harus menggunakan tipe data String daripada char. Contoh: String e = "abc";. | | |
| **[No.3] Penyusunan Algoritma dan Kode Program** | | |
| Algoritma:  1).mulai  2).berikan nama class dan file  3). tambahkan method main  4).membuat variable  5)mendeklarasikan varibel  6)run  7.selesai    Luaran sudah sesuai dengan program yang disusun.  Tipe data yang ditampilkan telah sesuai dengan kebutuhan dan permintaan data. | | |
| **[No.3] Kesimpulan** | | |
| Dari kesalahan dalam pemakaian tipe data ini saya mengetahui jika tipe data memiliki rentangnya masing masing seperti int memiliki rentang dari -2,147,483,648 hingga 2,147,483,647 dan byte memiliki rentang dari -128 hingga 127. Tipe data char hanya dapat menyimpan satu karakter | | |
| **[No. 4] Identifikasi Masalah:**  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 | | |
| **[No.4] Analisis dan Argumentasi** | | |
| 4.1. pada pembahasan ini kami merekomendasikan konversi sebagai berikut:  Pada data 1  KONVERSI DARI TIPE DATA BYTE KE TIPE DATA INTEGER  Pada data 2  KONVERSI DARI TIPE DATA CHAR KE TIPE DATA STRING  Pada data 3  KONVERSI DARI TIPE DATA STRING KE TIPE DATA OBJECT  Pada data 4  KONVERSI DARI TIPE DATA DOUBLE KE TIPE DATA BYTE  Pada data 5  KONVERSI DARI TIPE DATA FLOAT KE TIPE DATA INTEGER  Pada data 6  KONVERSI DARI TIPE DATA BYTE KE TIPE DATA INTEGER  4.2.  alasan kami memilih jenis konversi tipe data tersebut  yaitu:   * Pada data 1   Konversi ke tipe data integer  **Karna rentang Lebih Luas** & Mencegah Overflow   * Pada data 2   Konversi ke tipe data data string  Kemudahan Pengolahan Teks   * Pada data 3   Konversi ke tipe data object  Karna tipe data object memungkinkan pengolahan data dalam konteks yang lebih umum dan fleksibel.   * Pada data 4   Konversi ke tipe data byte  Jika data yang diwakili oleh double sebenarnya tidak memerlukan presisi tinggi dan berada dalam rentang byte, mengkonversi ke byte dapat menyederhanakan pengolahan atau analisis.   * Pada data 5   Konversi ke tipe data integer  **Karna rentang Lebih Luas** & Mencegah Overflow   * Pada data 6   Konversi ke tipe data integer  **Karna rentang Lebih Luas** & Mencegah Overflow | | |
| **[No.4] Penyusunan Algoritma dan Kode Program** | | |
| Algorima**:**  1).mulai  2).berikan nama class dan file  3). tambahkan method main  4).membuat variable konvensi  5)mendeklarasikan varibel konvensi  6)run  7.selesai  **Codingan:**    **Luaran:**    Luaran sudah sesuai dengan program yang disusun | | |
| **[No.4] Kesimpulan** | | |
| Dari pembahasan no 4 ini kita mengetahui jika setiap data memiliki tipe data nya sendiri tapi kita dapat mengkorversi kan ke bentuk tipe data lain yang kompatibel | | |
| **Refleksi:**  **Dari pembelajaran ini saya mendapat banyak pengetahuan baru seperti belajar dasar tentang Bahasa pemrograman java,menyesuaikan data dengan tipe datanya,dan mengkonversikan data.pada latihan ini saya juga belajar dalam memecahkan masalah,misalnya pada saya menghadapi codingan eror dan mencari solusi untuk memperbaikinya** | | |
|  | | |