**Template Lembar Kerja Individu dan Kelompok**

|  |  |  |
| --- | --- | --- |
| **Nama & NPM** | **Topik:** | **Tanggal:** |
| **MOZA VIOLETHA SALSABILLA**  **G1F024028** | **Kelas Java** | **18 September 2024** |
| **[Nomor Soal] Identifikasi Masalah:** | | |
| 1. Uraikan permasalahan dan variabel 2. Rincikan sumber informasi yang relevan (buku / webpage) 3. Uraikan rancangan solusi yang diusulkan (jika ada). 4. Analisis susunan solusi, parameter solusi (jika ada). | | |
| **[Nomor Soal] Analisis dan Argumentasi** | | |
| 1. Uraikan rancangan solusi yang diusulkan. 2. Analisis solusi, kaitkan dengan permasalahan. | | |
| **[Nomor Soal] Penyusunan Algoritma dan Kode Program** | | |
| 1. Rancang desain solusi atau algoritma 2. Tuliskan kode program dan luaran 3. Beri komentar pada kode 4. Uraikan luaran yang dihasilkan 5. Screenshot/ Capture potongan kode dan hasil luaran | | |
| **[Nomor Soal] Kesimpulan** | | |
| 1. Analisa 2. Susunlah kesimpulan berdasarkan permasalahan, algoritma, dan kode program! 3. Apakah dasar alasan pengambilan keputusan Anda untuk kasus ini? 4. Evaluasi 5. Apa konsekuensi dari skenario pemprograman ini? 6. Evaluasi input, proses, dan luaran yang dihasilkan! (jika ada) 7. Kreasi 8. Apakah ada pengetahuan baru yang dikembangkan dan konsep baru sebagai usulan solusi? 9. Konstruksikan hubungan antara variabel yang berbeda dengan konsep yang anda ketahui! (jika ada) | | |
|  | | |

Contoh Jawaban:

|  |  |  |
| --- | --- | --- |
| **Nama & NPM** | **Topik:** | **Tanggal:** |
| **Moza Violetha Salsabilla**  **G1F024028** | **Kelas Java** | **18 September 2024** |

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

1. Uraikan permasalahan dan variabel

**Contoh 1:**  
public class Manusia {  // deklarasi kelas  
    //deklarasi atribut Manusia dalam variabel  
    String nama, rambut;  
      
    //deklarasi constructor  
    public Manusia1 (String nama) {  
            System.out.println(" Nama saya : "+ nama +   
            "\n Warna Rambut : " + rambut);      
    }  
      
    //deklarasi method utama  
    public static void main( String[] args) {  
            Manusia1 satu = new Manusia1("Putri", "hitam");  
}  }

**Luaran 1:**  
Exception in thread "main" java.lang.Error: Unresolved compilation problem:   
    The constructor Manusia1(String, String) is undefined  
    at Manusia1.main(Manusia1.java:13)

**Latihan 1:**  
1.1.     Perbaiki pesan kesalahan Contoh 1!  
1.2.     Cermati contoh 1. susun kode menggunakan constructor dengan parameter data pribadi anda!

Jawaban:

Pada kode program masih terdapat beberapa kesalahan yaitu nama constructor yang tidak sesuai dengan nama kelas, parameter constructor yang tidak terinisialisasi, dan objek yang tidak sesuai dengan kelasnya.

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

1. Saya mengusulkan permasalahan ini dapat diatasi dengan cara :
2. Mengubah nama constructor yang awal mulanya Manusia1 menjadi Manusia agar sesuai dengan nama kelas yang telah dibuat.
3. Menambahkan parameter constructor untuk atribut rambut yaitu (String rambut).
4. Mengganti nama objek yang awal mulanya Manusia1 menjadi Manusia agar sesuai dengan nama kelasnya.
5. Alasan solusi ini karena :
6. Dalam Bahasa pemrograman java, constructur harus memiliki nama yang sama dengan kelas tempat objek tersebut dibuat. Apabila nama constructor tidak sesuai dengan nama kelas maka dapat mengakibatkan error saat program mencoba untuk menginisialisasi objek.
7. Dalam Bahasa pemrograman java, apabila pengguna ingin menginisialisasi atribut kelas, maka atribut yang akan diinisialisasi harus dideklarasikan sebagai parameter constructor. Pada kode program tersebut digunakan atribut rambut akan tetapi tidak diberikan nilai, sehingga program akan kebingungan dalam mencetak nilai dari atribut rambut.
8. Karena saat kita ingin membuat objek dalam kelas, maka harus menggunakan nama yang sama dengan kelasnya. Pada kode program telah dideklarasikan bahwa kelas dinamai sebagai Manusia maka dari itu objek nya juga harus dinamai dengan Manusia. Apabila objek dinamai Manusia1 maka program akan error saat runtime karena sistem tidak mengenali objek Manusia1.

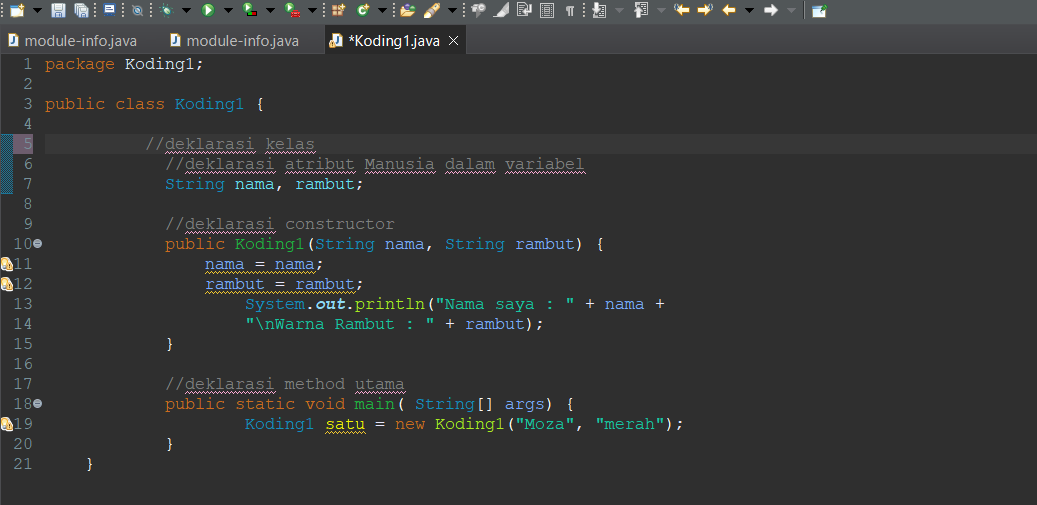
**[No.1 ] Penyusunan Algoritma dan Kode Program**

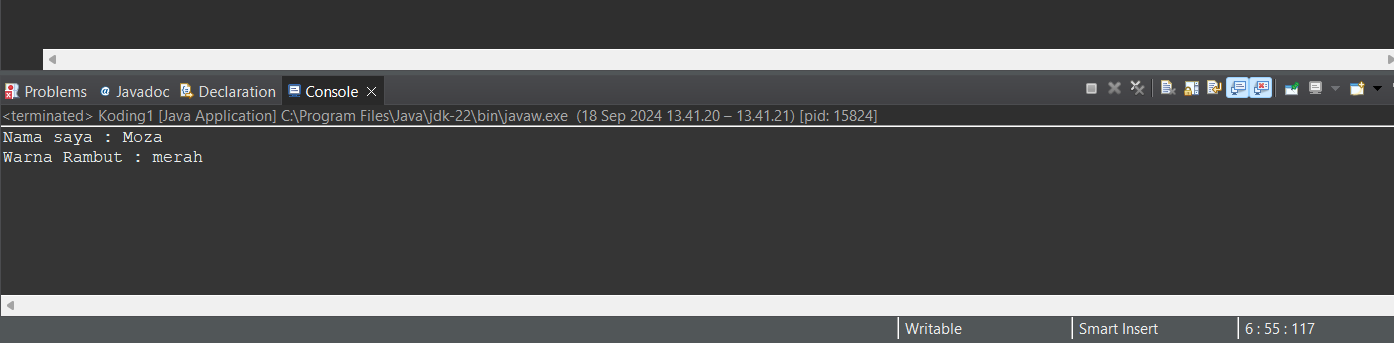
1. Algoritma

Algoritma adalah langkah-langkah penyelesaian masalah.

1. Mulai.
2. Membuat kelas.
3. Mendeklarasikan atribut.
4. Mendeklarasikan method main.
5. Menampilkan luaran.
6. Selesai.
7. Kode program dan luaran
8. 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. Pada kode program atribut yang digunakan adalah tipe data string untuk atribut nama dan rambut dan di luaran telah ditampilkan nama dan warna rambut.

**[No.1] Kesimpulan   
1)**  **Analisa**

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

Jawaban Analisa:

1. Pada program itu saya memperbaiki beberapa kode program yang salah berupa mengubah nama contructor, menambahkan parameter untuk atribut rambut, dan mengubah nama objek.Dapat disimpulkan bahwa dalam membuat suatu kelas dalam Bahasa pemrograman java tidak dapat dilakukan secara semena-mena karena terdapat bermacam-macam aturan yang harus dipatuhi agar tidak terjadi error saat runtime.
2. Dasar pengambilan Keputusan saya dalam hal ini adalah materi mengenai class, method, dan objek.

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

1. Uraikan permasalahan dan variabel

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

public class Ortu {  
    //deklarasi constructor  (variabel constructor)  
    public ortu   {    
        //nama dan rambut adalah variabel constructor  
        System.out.println(" Nama saya : "+ nama +   
        "\n Warna Rambut : " + rambut);      
}  
    public static void main (String[] args) {  
        Ortu satu = new Ortu("Putri", "hitam");  
}   }

**Luaran 2:**  
Exception in thread "main" java.lang.Error: Unresolved compilation problem:   
    The constructor Ortu(String, String) is undefined

    at Ortu.main(Ortu.java:9)

**Latihan 2:**  
2.1.  Evaluasi penyebab kesalahan dan perbaiki kode tersebut!  
2.2.  Apabila nanti Anda akan memiliki keturunan, analisa sifat (atribut) dan constructor sebagai Ortu apa yang akan diturunkan (gunakan data karakter pribadi anda) ?  
2.3.  Rancanglah kode program untuk sifat (atribut) dan constructor overloaded dari Latihan 2.2!

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

* 1. Pada kode program tersebut masih terdapat beberapa kesalahan yaitu :
  2. Nama constructor tidak sama dengan nama kelas.
  3. Parameter constructor nama dan rambut tidak dideklarasikan.
  4. Variabel nama dan rambut tidak dideklarasikan sebagai atribut.

Saya mengusulkan permasalahan ini dapat diatasi dengan cara

1. Dalam Bahasa pemrograman java, constructur harus memiliki nama yang sama dengan kelas tempat objek tersebut dibuat. Apabila nama constructor tidak sesuai dengan nama kelas maka dapat mengakibatkan error saat program mencoba untuk menginisialisasi objek.
2. Dalam Bahasa pemrograman java, apabila pengguna ingin menginisialisasi atribut kelas, maka atribut yang akan diinisialisasi harus dideklarasikan sebagai parameter constructor. Pada kode program tersebut digunakan atribut nama dan rambut akan tetapi tidak diberikan nilai, sehingga program akan kebingungan dalam mencetak nilai dari atribut rambut dan nama.
3. Menambahkan variabel nama dan rambut sebagai atribut dalam kelas tersebut.
   1. Analisa sifat dari atribut dan constructor yang akan diturunkan yaitu :
4. Nama, nama belakang biasanya diwariskan oleh orang tua.
5. Rambut, warna rambut juga dapat diturunkan dari genetic orang tua.
6. Tinggi, tinggi seorang anak biasanya diwariskan dari gen orang tuanya.
7. Hobi, hobi seorang anak biasanya akan terinspirasi dari hobi orang tuanya.

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

* + 1. Algoritma

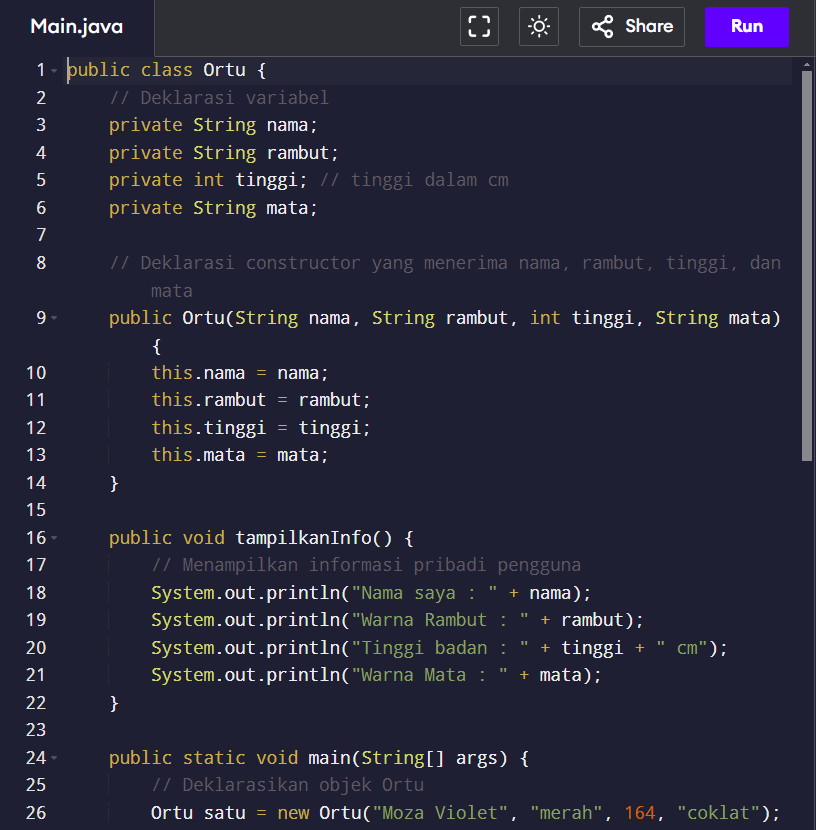
Algoritma adalah langkah-langkah penyelesaian masalah.

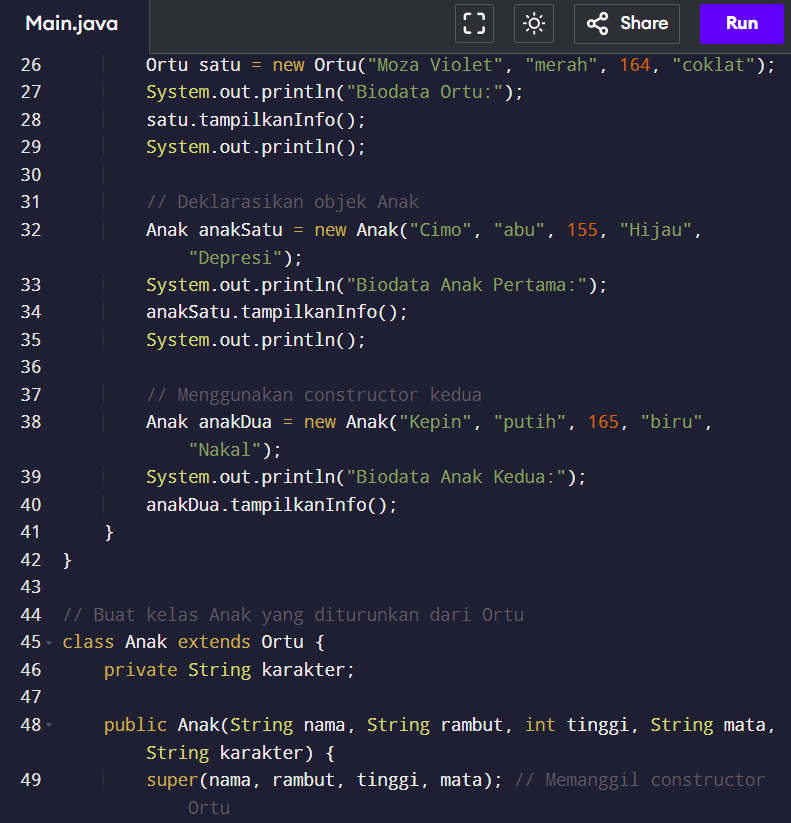
* + - 1. Mulai.
      2. Membuat kelas.

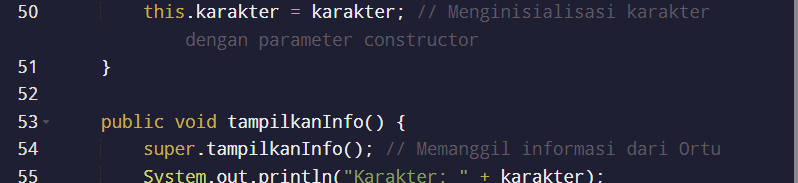
(c) Mendeklarasikan atribut.

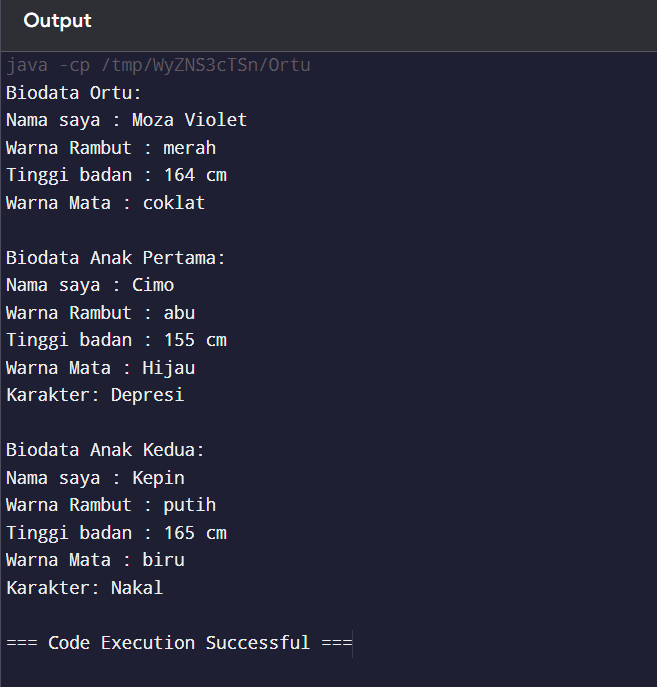
1. Mendeklarasikan method main.
2. Menampilkan luaran ke layar.
3. Selesai.
   * 1. Kode program dan luaran
        + 1. Screenshot/ Capture potongan kode dan hasil luaran

Beri komentar pada kode yang di Screenshot









* + - * 1. Analisa luaran yang dihasilkan

Luaran sudah sesuai dengan program yang disusun. Pada kode program atribut yang digunakan adalah tipe data string untuk atribut nama, rambut, mata, karakter dan tipe data integer untuk atribut tinggi.

**[No.2] Kesimpulan   
1)**  **Analisa**

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

Jawaban Analisa:

1. Pada program itu saya memperbaiki beberapa kode program yang salah berupa menyamakan nama constructor dengan nama kelas, mendeklarasikan parameter constructor dari nama,rambut,tinggi,dan hobi, mendeklarasikan parameter dari nama, rambut, tinggi, dan warna mata. Sama seperti soal sebelumnya bahwasannya Kesimpulan dalam program ini adalah harus sangat teliti dalam pembuatannya karena sangatlah sensitive terhadap kesalahan sekecil apapun.
2. Dasar pengambilan Keputusan saya dalam hal ini adalah materi mengenai class, method, dan objek.

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

1. Uraikan permasalahan dan variabel

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

public class Manusia {  
    //deklarasi atribut Manusia dalam variabel  
    String nama, rambut;  
      
    //deklarasi constructor  
    public Manusia1(String nama, String rambut) {  
            System.out.println(" Nama saya : "+ nama +   
            "\n Warna Rambut : " + rambut);      
    }  
      
    //deklarasi method  
    void sukaNonton  {  
        System.out.println(" Hobi Menonton : " + film);  
    }  
      
    int sukaNonton {  
        episode\*durasi;  
    }  
              
    //deklarasi method utama  
    public static void main( String[] args) {  
            Manusia satu = new Manusia("Putri", "hitam");  
            satu.sukaNonton("Drakor");  
            int jumlahJam = satu.sukaNonton(2, 2);  
            System.out.println("Jam nonton = " +jumlahJam + " jam");  
}  }

**Luaran 3:**  
Exception in thread "main" java.lang.Error: Unresolved compilation problems:   
    The method sukaNonton(String) is undefined for the type Manusia1  
    The method sukaNonton(int, int) is undefined for the type Manusia1  
    at Manusia1.main(Manusia1.java:23)

**Latihan 3:**  
3.1.  Evaluasi penyebab kesalahan dan perbaiki kode tersebut!  
3.2.  Berdasarkan Latihan 2.2. Anda sudah punya kode program untuk atribut dan constructor sebagai Ortu.  
        Kembangkanlah kode program untuk method dari Ortu dengan data perilaku pribadi Anda yang menggunakan:  
        a) method overloading,  
        b) method dengan return value  
        c) method tanpa return value

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

1. Pada kode program tersebut masih terdapat beberapa kesalahan yaitu :
2. Nama pada constructor tidak sesuai dengan nama kelas.
3. Atribut suka nonton tidak memiliki parameter yang diperlukan.
4. Cara penulisan atribut suka nonton kedua salah karena yang seharusnya menghitung jam nonton.
5. Tidak mendeklarasikan variabel film, durasi, dan episode.
6. Permasalahan ini dapat diatasi dengan cara:
7. Mengubah nama dari constructor menjadi Manusia agar sesuai dengan nama kelas.
8. Menambahkan parameter constructor untuk method sukanonton agar dapat diisi dengan nama film.
9. Mendeklarasikan sukanonton dengan format (int episode, int durasi).
10. Menghapus variabel yang tidak dideklarasikan agar tidak mengganggu jalannya run.

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

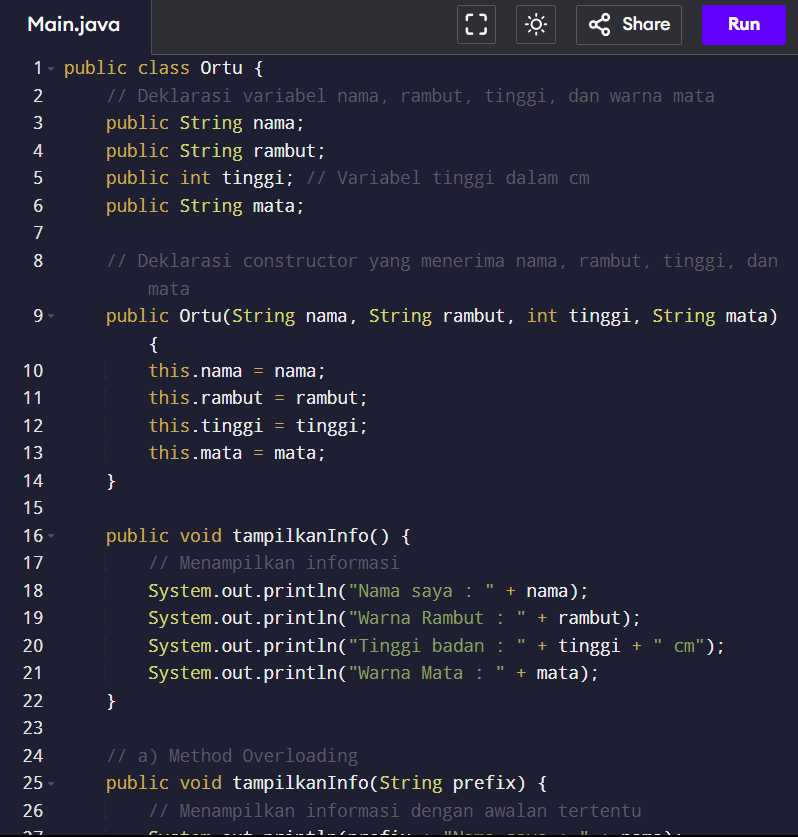
1. Algoritma

Algoritma adalah langkah-langkah penyelesaian masalah.

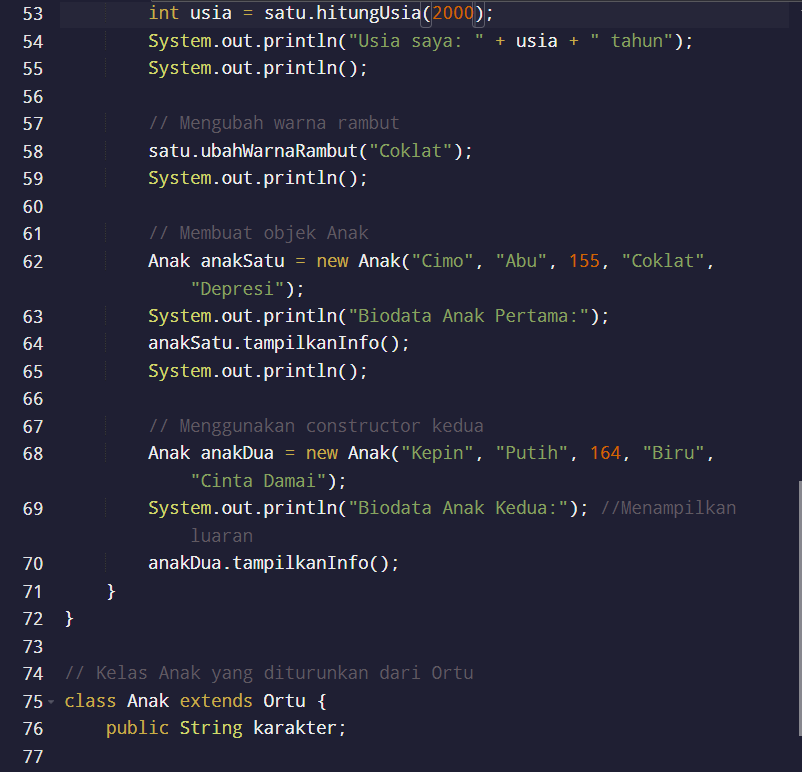
a)Mulai.

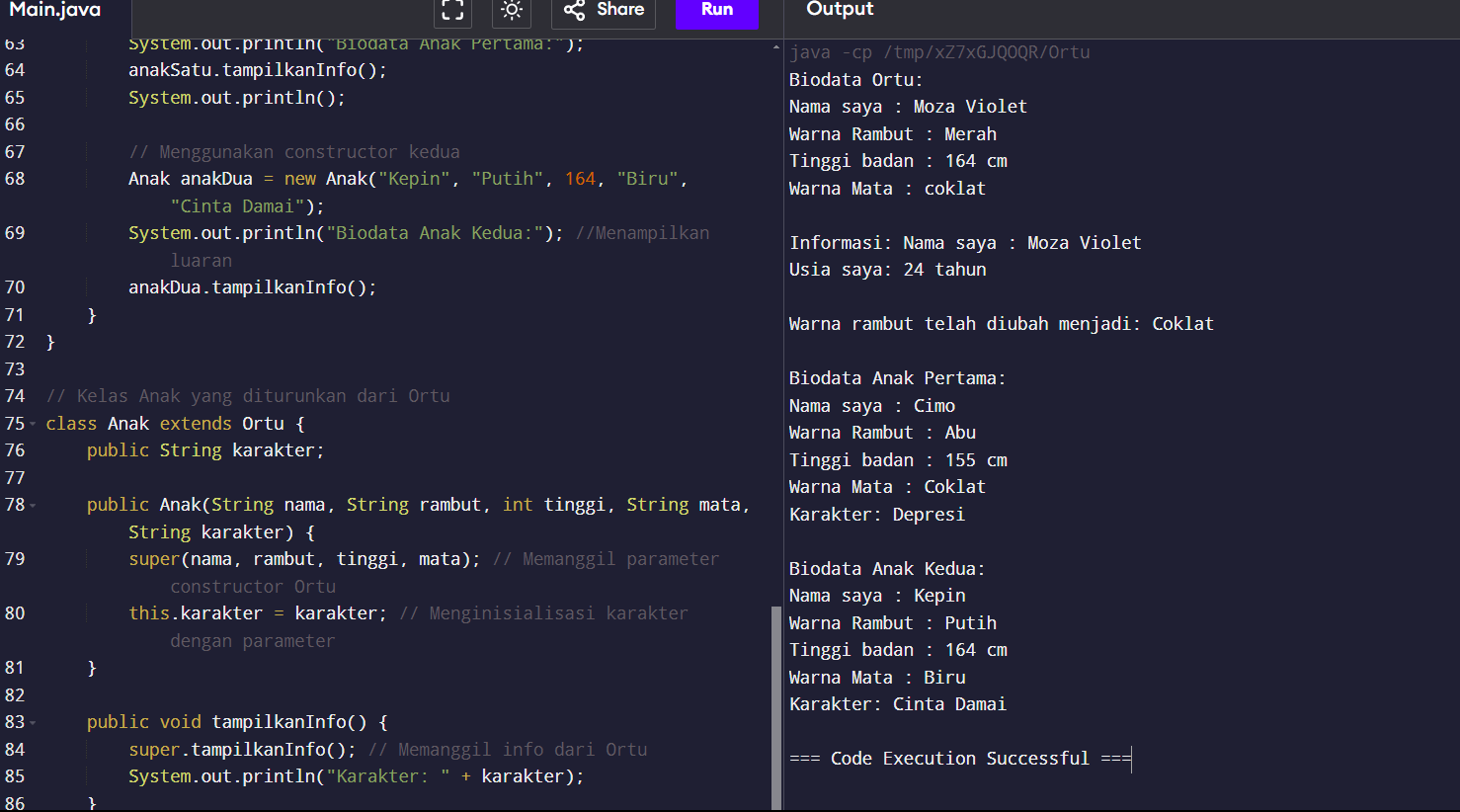
1. Mendeklarasikan kelas ortu.
2. Mendeklarasikan main method.
3. Mendeklarasikan kelas anak.
4. Buat objek anak.
5. Menampilkan luaran.
6. Selesai.
7. Kode program dan luaran
8. Screenshot/ Capture potongan kode dan hasil luaran

Beri komentar pada kode yang di Screenshot









1. Analisa luaran yang dihasilkan

Luaran sudah sesuai dengan program yang disusun. Pada kode program atribut yang digunakan adalah tipe data string untuk atribut nama, rambut, mata, karakter dan tipe data integer untuk atribut tinggi.

**[No.3] Kesimpulan   
1)**  **Analisa**

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

Jawaban Analisa:

1. Pada program itu saya memperbaiki beberapa kode program yang salah berupa Mengubah nama dari constructor menjadi Manusia agar sesuai dengan nama kelas, menambahkan parameter constructor untuk method sukanonton agar dapat diisi dengan nama film, mendeklarasikan sukanonton dengan format (int episode, int durasi), menghapus variabel yang tidak dideklarasikan agar tidak mengganggu jalannya run.Pada program tersebut digunakan dua kelas yaitu Ortu dan Anak yangmana Ortu sebagai kelas dan Anak sebagai subclass. Kelas ortu memiliki 4 atribut yang akan diwariskan ke Anak yaitu nama, rambut, tinggi, dan warna mata. Agar sang Anak dapat mewarisi atribut Ortu maka harus melakukan pemanggilan terhadap constructor.
2. Dasar pengambilan Keputusan saya dalam hal ini adalah materi mengenai class, method, dan objek.

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

1. Uraikan permasalahan dan variabel

**Contoh 4:**Salin dan tempel kode program berikut ke JDoodle. Kemudian catat waktu eksekusinya.

public class Ortu {       // membuat kelas induk  
    void sukaMenonton(String a) {    // method induk spesifik  
      System.out.println("Nonton " + a);  
    }  
    void sukaMembaca(String a) {     // method induk umum bisa diubah anak  
      System.out.println("Suka Baca " + a);  
    }

public static void main(String [] args) {  
    System.out.println("Sifat Orang Tua :");   
    Ortu objekO = new Ortu();     // memanggil objek induk  
    objekO.sukaMenonton("Berita");    // memanggil sifat spesifik induk  
    objekO.sukaMembaca("Koran");     // memanggil method dengan variabel dapat diubah  
      
    System.out.println("\n Sifat Anak :");      
    Anak objekA = new Anak();    //memanggil objek anak  
    objekA.sukaMenonton(9, "Film Drakor");        //memanggil sifat spesifik anak yang diturunkan induk  
    objekA.sukaMembaca("Komik One Piece"); //memanggil method ke induk yang otomatis diturunkan tanpa deklarasi ulang di anak  
}    }

class Anak extends Ortu {  
  void sukaMenonton(int a, String b) {  
        System.out.println("Nonton Jam " + a + " Malam " + b);  
  }      
  void sukaMenonton(String a) {            // method induk spesifik  
        System.out.println("Nonton " + a);  
  }  
  void sukaMembaca(String a) {     // method induk umum bisa diubah anak  
        System.out.println("Suka Baca " + a);  
  }  
public static void main(String [] args) {  
    System.out.println("Sifat Orang Tua :");   
    Ortu objekO = new Ortu();     // memanggil objek induk  
    objekO.sukaMenonton("Berita");        // memanggil sifat spesifik induk  
    objekO.sukaMembaca("Koran");     // memanggil method dengan variabel dapat diubah  
              
    System.out.println("\n Sifat Anak :");      
    Anak objekA = new Anak();    //memanggil objek anak  
    objekA.sukaMenonton(9, "Film Drakor");        //memanggil sifat spesifik anak yang diturunkan induk  
    objekA.sukaMembaca("Komik One Piece"); //memanggil method ke induk yang otomatis diturunkan tanpa deklarasi ulang di anak  
}    }

**Luaran 4:**  
Sifat Orang Tua :  
Nonton Berita  
Suka Baca Koran

Sifat Anak :  
Nonton Jam 9 Malam Film Drakor  
Suka Baca Komik One Piece

**Latihan 4:**  
4.1. Evaluasi method yang dimiliki Contoh 4  pada class Anak extends Ortu dengan method di class Ortu. Simpulkan hasil evaluasi Anda agar method ini menjadi efisien!  
4.2. Setelah dirunning di JDoodle, catat waktu eksekusinya.   
       Susun kembali kode program yang dapat mengefisienkan waktu eksekusi!

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

1. Pada method dalam kelas Ortu terdapat dua method yaitu sukaMenonton dan sukaMembaca. Method sukaMenonton diinisialisasi menggunakan tipe data string dan tidak dapat diubah oleh Anak, Walaupun begitu, dapat diturunkan ke subclass anak. Sedangkan, method suka membaca bersifat open dan dapat diubah oleh Anak(tidak dipengaruhi kelas Ortu).
2. Kesimpulan mengenai hasil evaluasi pada kode program ini adalah bahwa pada kode program masi ada yang Namanya redudansi atau pengulangan. Subclass anak mengulang kembali method sukaMenonton yang telah dideklarasikan di kelas ortu. Seharusnya, method suka menonton dapat dipanggil saja, tidak perlu dideklarasikan ulang. Program juga menambahkan method baru yaitu sukaMenonton yang tidak ada pada kelas Ortu.

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

1. Algoritma

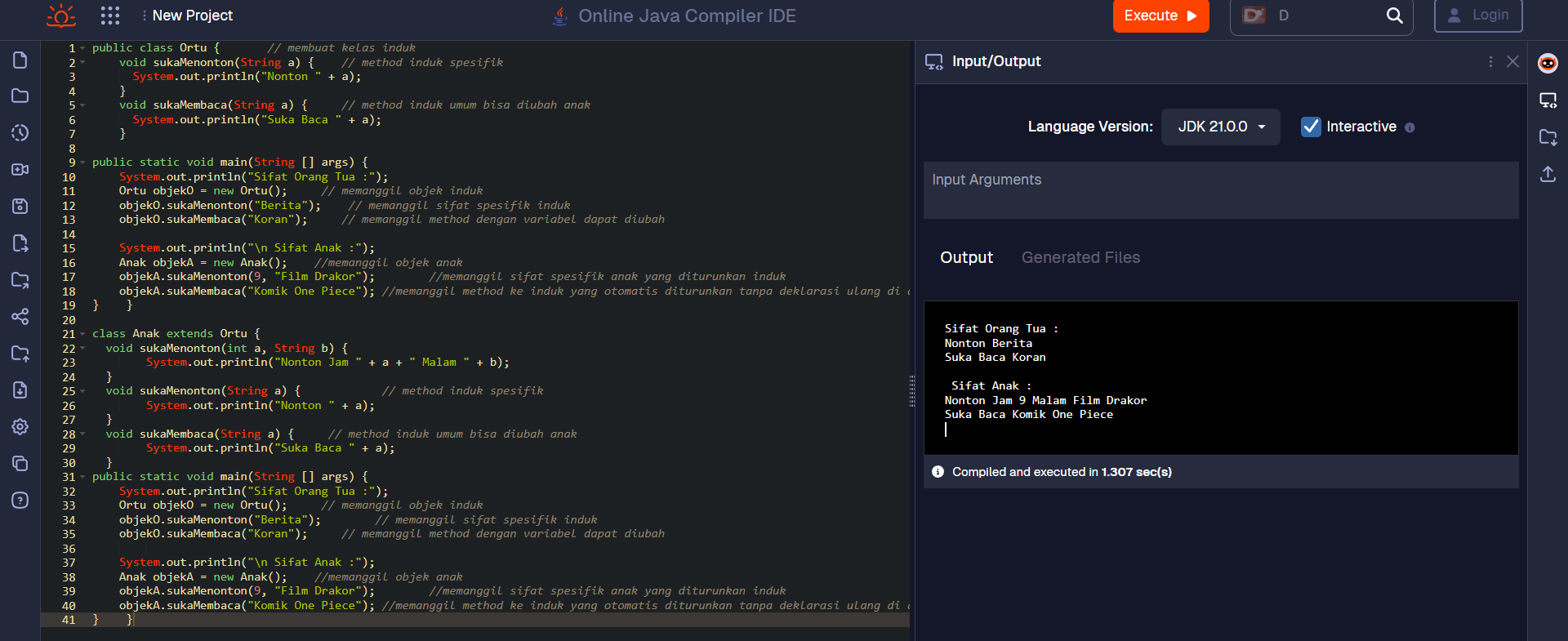
Algoritma adalah langkah-langkah penyelesaian masalah.

a)Mulai.

b)Mendeklarasikan kelas ortu.

1. Mendeklarasikan main method pada kelas ortu
2. Mendeklarasikan kelas anak.
3. Buat objek anak dan deklarasikan main method.
4. Menampilkan luaran.
5. Selesai.
6. Kode program dan luaran
7. Screenshot/ Capture potongan kode dan hasil luaran

Beri komentar pada kode yang di Screenshot



Kode program awal



Kode program setelah diperbaiki

1. Analisa luaran yang dihasilkan

Luaran sudah sesuai dengan program yang disusun. Kode program sebelum diperbaiki membutuhkan waktu 1.307 sec untuk dijalankan. Sedangkan, setelah diperbaiki waktu yang dibutuhkan untuk runtime adalah 1.278 sec. Memang waktu yang diperlukan lebih singkat, akan tetapi untuk kode program yang telah diperbaiki output yang keluar tidak se-spesifik seperti kode program awal.

**[No.4] Kesimpulan   
1)**  **Analisa**

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

Jawaban Analisa:

1. Pada program tersebut diminta untuk mengefisienkan waktu saat runtime, untuk dapat melakukan hal itu maka kita harus menghapus beberapa elemen pada program yang tidak terlalu penting dan berulang-ulang(redudansi). Seperti halnya pendeklarasian atribut sukaNonton yang seharusnya dipanggil saja dari kelas Ortu, tidak perlu ditulis berulang kali.
2. Dasar pengambilan Keputusan saya dalam hal ini adalah materi mengenai class, method, dan objek.

**Refleksi**

Pada pembelajaran kali ini, saya belajar mengenai konsep kelas dan method dalam Bahasa pemrograman java, terutama dalam hal pewarisan method. Dalam beberapa tugas yang diberikan kali ini dapat memahami bahwasannya method dari kelas dapat diwariskan ke subclass dibawahnya. Fungsi override juga memungkinkan kita untuk dapat menjalankan kode program dengan lebih cepat.