**Template Lembar Kerja Individu dan Kelompok**

|  |  |  |
| --- | --- | --- |
| **Nama & NPM** | **Topik:** | **Tanggal:** |
| **FEBRI G1F024070** | **Kelas, Objek, Method** | **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) | | |
|  | | |

LATIHAN!!

**1.1** Analisa ciri-ciri umum Kelas Manusia yang dapat menjadi  
           a. atribut variabel, dan  
           b. perilaku/ behavior untuk method!

**Analisis Kelas Manusia**

**a. Atribut Variabel:**

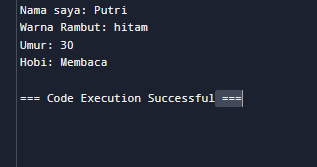
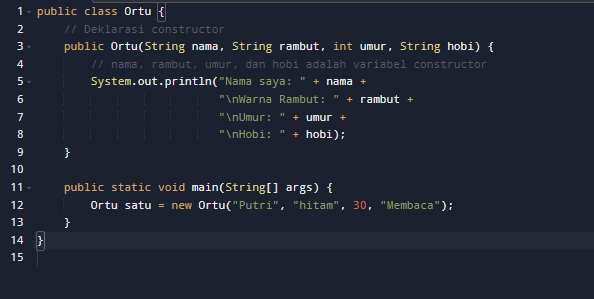
* **nama:** Variabel ini menyimpan nama dari objek Manusia. Ini merupakan atribut yang dapat memberikan identitas pada setiap objek.
* **rambut:** Variabel ini menyimpan informasi mengenai rambut dari objek Manusia. Ini juga merupakan atribut yang memberikan detail lebih lanjut tentang objek.

**b. Perilaku/Behavior untuk Method:**

* Kelas Manusia belum memiliki method lain selain constructor. Namun, kita dapat menambahkan beberapa metode untuk memberikan perilaku:
  + **getNama():** Metode untuk mengembalikan nilai dari atribut nama.
  + **setNama(String nama):** Metode untuk mengubah nilai dari atribut nama.
  + **getRambut():** Metode untuk mengembalikan nilai dari atribut rambut.
  + **setRambut(String rambut):** Metode untuk mengubah nilai dari atribut rambut.
  + **perkenalan():** Metode yang mencetak informasi tentang objek, misalnya "Saya adalah [nama] dengan rambut [rambut]."

Dengan cara ini, kita dapat menambahkan perilaku yang lebih kompleks dan berguna untuk kelas Manusia.

**2.1** Susun kembali kode di contoh 2 dengan menambahkan data ciri-ciri Anda di dalam variabel constructor!



**2.2** Apabila nanti Anda akan memiliki keturunan, analisa sifat (atribut), constructor, dan perilaku positif (behavior) apa yang akan diturunkan?

**Sifat (Atribut) yang akan diturunkan:**

* **nama:** Identitas yang bisa diwariskan ke keturunan.
* **rambut:** Karakteristik fisik, seperti warna rambut.
* **umur:** Meskipun umur tidak bisa diwariskan secara langsung, informasi tentang generasi bisa dimasukkan.
* **hobi:** Minat yang mungkin juga dapat diwariskan atau dipengaruhi oleh lingkungan.

**Constructor untuk Keturunan:**

* Keturunan bisa memiliki constructor yang mencakup atribut dari Ortu serta atribut tambahan yang spesifik untuk keturunan, seperti **sekolah** atau **cita-cita**.

**Perilaku Positif (Behavior) yang akan diturunkan:**

* **perkenalan():** Metode untuk memperkenalkan diri, yang bisa mencetak informasi tentang nama, rambut, umur, dan hobi.
* **hobiKesukaan():** Metode untuk menjelaskan hobi, yang bisa diubah sesuai dengan preferensi keturunan.
* **warisan():** Metode yang menunjukkan atribut apa yang diturunkan dari orang tua ke anak, seperti karakteristik fisik atau minat.

Dengan cara ini, keturunan akan memiliki sifat dan perilaku yang berguna, serta dapat membedakan dirinya dari orang tua.

**3.1** Analisa perbedaan deklarasi constructor, method, dan method utama!

1. **Constructor:**

* Fungsi yang dipanggil saat objek dari kelas dibuat.
* Nama constructor harus sama dengan nama kelas (Manusia dalam hal ini).
* Tidak memiliki tipe pengembalian (return type).
* Digunakan untuk menginisialisasi atribut objek.

2. **Method:**

* Fungsi yang dapat dipanggil pada objek untuk melakukan operasi tertentu.
* Memiliki tipe pengembalian (bisa void atau tipe data lainnya).
* Dapat memiliki parameter dan dapat dipanggil berkali-kali setelah objek dibuat.

3. **Method Utama (main method):**

* Titik masuk utama untuk program Java.
* Harus memiliki signature tertentu (public static void main(String[] args)).
* Digunakan untuk menjalankan program dan dapat membuat objek dari kelas lain.

**3.2** Tentukan kapan Anda perlu menggunakan constructor dan method?

A. **Constructor:**

* Digunakan ketika Anda perlu menginisialisasi atribut objek saat objek pertama kali dibuat.
* Berguna untuk menetapkan nilai default atau untuk mempersiapkan objek sebelum digunakan.

B. **Method:**

* Digunakan untuk melakukan operasi atau tindakan tertentu setelah objek dibuat.
* Berguna untuk memperbarui status objek, melakukan perhitungan, atau menampilkan informasi.

**3.3** Uraikan perbedaan berikut:  
        a) constructor overloading dan overriding  
        b) method overloading, dan method overriding  
        c) method yang mengembalikan nilai dan method tidak mengembalikan nilai

* **Method Overloading:**
  + Memungkinkan dua atau lebih metode dengan nama yang sama dalam satu kelas, tetapi dengan parameter yang berbeda (jumlah atau tipe).
  + Contoh: Metode sukaNonton(String film) dan sukaNonton(String film, int tahun).
* **Method Overriding:**
  + Memungkinkan subclass untuk memberikan implementasi spesifik untuk metode yang sudah ada di superclass.
  + Contoh: Jika kelas Manusia memiliki metode perkenalan(), kelas Anak yang merupakan subclass dapat mengoverride metode ini untuk memberikan implementasi yang berbeda.

**c) Method yang Mengembalikan Nilai dan Method Tidak Mengembalikan Nilai:**

* **Method yang Mengembalikan Nilai:**
  + Memiliki tipe pengembalian yang ditentukan (misalnya, int, String).
  + Harus menggunakan kata kunci return untuk mengembalikan nilai.
  + Contoh: int hitungLuas() { return 5 \* 10; }.
* **Method Tidak Mengembalikan Nilai:**
  + Tipe pengembalian dideklarasikan sebagai void.
  + Tidak menggunakan kata kunci return untuk mengembalikan nilai (walaupun return; bisa digunakan untuk keluar dari method).
  + Contoh: void tampilkanInfo() { System.out.println("Info"); }.

Dengan pemahaman ini, Anda dapat mengorganisir dan merancang kelas serta metode dengan lebih efektif dalam pemrograman Java.

**4.1** Bandingkan method yang dimiliki class Anak extends Ortu dengan method di class Ortu!

A. **Method di Kelas Ortu:**

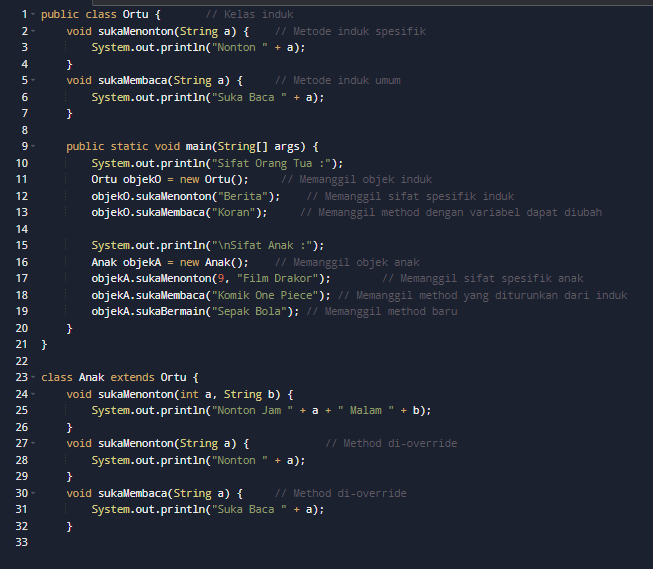
* void sukaMenonton(String a): Metode spesifik yang mencetak aktivitas menonton.
* void sukaMembaca(String a): Metode umum yang bisa diwarisi dan diubah oleh kelas anak.

B. **Method di Kelas Anak:**

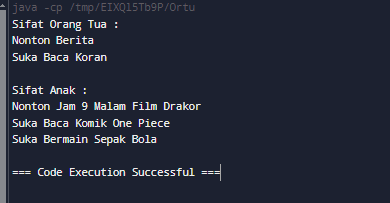
* void sukaMenonton(int a, String b): Metode spesifik yang di-overload, menerima dua parameter dan mencetak informasi tentang menonton dengan jam.
* void sukaMenonton(String a): Metode ini di-override untuk memberikan implementasi yang sama dengan Ortu, tetapi dapat tetap digunakan.
* void sukaMembaca(String a): Metode ini di-override untuk memberikan implementasi khusus untuk kelas Anak.

**4.2** Ubahlah Contoh 4 dengan menambahkan objek anak dengan method yang berbeda!

Mari kita tambahkan metode baru di kelas Anak, misalnya void sukaBermain(String permainan):

O

Output:



Dengan penambahan ini, kelas Anak kini memiliki metode baru yang memungkinkan anak untuk menyatakan hobi atau aktivitas baru yang tidak ada di kelas Ortu.

Contoh Jawaban:

|  |  |  |
| --- | --- | --- |
| **Nama & NPM** | **Topik:** | **Tanggal:** |
| **Putri  G1A000001** | **Tipe Data** | **26 Agustus 2022** |

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

1. Uraikan permasalahan dan variabel

Contoh:

Tuliskan kembali soal:

Pada soal masih ada pesan kesalahan \_\_\_\_\_\_\_\_\_\_\_\_

Atau  
Diketahui dari soal : variabel \_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

1. Saya mengusulkan permasalahan ini dapat diatasi dengan cara \_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. Alasan solusi ini karena \_\_\_\_\_\_\_\_\_\_\_\_\_\_
3. Perbaikan kode program dengan cara \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

1. Algoritma

Algoritma adalah langkah-langkah penyelesaian masalah.  
Misalkan algoritma memasak mi instan:

1. Masak air
2. Buka bungkus
3. Masukkan mie
4. Masukkan bumbu
5. Hasilnya mie matang, taruh di piring
6. Mie siap disantap.
7. Kode program dan luaran



1. Screenshot/ Capture potongan kode dan hasil luaran

Beri komentar pada kode yang di Screenshot

1. Analisa luaran yang dihasilkan   
   Contoh:

Luaran sudah sesuai dengan program yang disusun.   
Tipe data yang ditampilkan telah sesuai dengan kebutuhan dan permintaan data.  
  
(Tuliskan penjelasan dari program yang dibuat, apakah kode dan luaran sudah benar?)

**[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?

Contoh jawaban Analisa:

Pada program itu saya menggunakan bentuk kelas public karena \_\_\_\_\_\_\_\_\_\_  
 Perbaikan program dengan menambahkan \_\_\_\_\_\_\_\_\_\_\_\_ karena struktur java   
 mengharuskan \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_   
 (penjelasan analisa mengulangi kembali materi yang sudah diberikan)  
 (penjelasan mengkaitkan dengan materi yang ada)

1. **Evaluasi**
2. Apa konsekuensi/dampak dari kode program yang dibuat?
3. Evaluasi input program, proses perhitungan, dan luaran yang dihasilkan! (jika ada)

Contoh jawaban Evaluasi:

Pada program itu saya mengkonversi bentuk kelas public karena \_\_\_\_\_\_\_\_\_\_  
 Setelah dikonversi, saya mengevaluasi bahwa tipe data \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ lebih baik   
 digunakan untuk bentuk data seperti \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(penjelasan evaluasi mengulangi kembali materi yang sudah diberikan dan   
mengetahui kekurangan dari materi hasil eksperimen   
(misal tipe data \_\_\_ ternyata tidak dapat dipakai untuk \_\_\_\_\_\_ karena \_\_\_\_\_ )

1. **Kreasi**
2. Apakah ada pengetahuan baru yang dikembangkan dan konsep baru sebagai usulan solusi?
3. Susunlah hubungan antara variabel yang berbeda dengan konsep yang anda ketahui! (jika ada)

Contoh jawaban Kreasi:

Pada program itu saya mengkonversi bentuk kelas public karena \_\_\_\_\_\_\_\_\_\_  
 Setelah dikonversi, saya mengevaluasi bahwa tipe data \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ lebih baik   
 digunakan untuk bentuk data seperti \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Saya telah mencoba mengubah menjadi kelas private dan protected, ternyata menghasilkan \_\_\_\_\_\_\_\_\_\_

Berarti kelas private dan protected mempengaruhi \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_   
(sampaikan temuan Anda yang baru diketahui, eksperimen baru diluar materi yang diberikan)

(penjelasan kreasi mengulangi kembali materi yang sudah diberikan dan

menambahkan pengetahuan baru dari pengalaman dari hasil eksperimen )

Lanjutkan ke soal nomor 2 – 3 – … – dan seterusnya

**Refleksi**

(Tuliskan singkat tentang pengalaman belajar, pemaknaan pengetahuan yang baru, tantangan yang dihadapi pada minggu tersebut. Ringkasan singkat dari semua soal, bukan per soal)