**Template Lembar Kerja Individu dan Kelompok**

|  |  |  |
| --- | --- | --- |
| **Nama & NPM** | **Topik:** | **Tanggal:** |
|  |  |  |
| **[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) | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| **Nama & NPM** | **Topik:** | **Tanggal:** |
| **Reksi Hendra Pratama**  **G1A022022**  **Imelda Cyntia**  **G1A022022**  **Evelyn Eunike Aritonang**  **GIA022024** | **Kelas,Objek dan Method** | **15 September 2022** |

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

1. Uraikan permasalahan dan variable

Diketahui dari soal :

Apabila diketahui kelas induk adalah Mahasiswa dan Kelas anak adalah turunan dari mahasiswa maka:

(a)  Analisa atribut, method, dan constructor dari kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa)!

(b)  Evaluasi perbedaan kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa)!

(c)  Rekomendasi atribut, method, dan constructor yang bisa digunakan bersama kelas induk dan kelas anak!

(d)  Desain kode program Java yang berisi atribut, method, dan constructor dari kelas induk Mahasiswa dan kelas anak (turunan dari Mahasiswa)!

1. Rincikan sumber informasi yang relevan (buku / webpage)

### Video penjelasan pembelajaran dapat diakses pada [Chanel Youtube Rumah Ilmu Raflesia https://www.youtube.com/channel/UC8B9rghd3dBiS6OKonLMyIw](file:///C:\Users\HP\Downloads\Chanel%20Youtube%20Rumah%20Ilmu%20Raflesiahttps:\www.youtube.com\channel\UC8B9rghd3dBiS6OKonLMyIw)

### [Video Materi 1 tentang Kelas, Objek, Method – https://www.youtube.com/watch?v=60IdOc8m8Es](file:///C:\Users\HP\Downloads\Video%20Materi%201%20tentang%20Kelas,%20Objek,%20Method%20–%20https:\www.youtube.com\watch%3fv=60IdOc8m8Es )

### [Video Materi 2 tentang – https://www.youtube.com/watch?v=6qULMlcv-eg](file:///C:\Users\HP\Downloads\Video%20Materi%202%20tentang%20–%20https:\www.youtube.com\watch%3fv=6qULMlcv-eg)

1. Uraikan rancangan solusi yang diusulkan (jika ada).

Hasil Salinan kode program sebagai berikut

**public** **class** Mahasiswa {

//deklarasi atribut mahasiswa dalam variabel

String aktivitasmahasiswa, tugas, pengajar, referensi;

String aktivitassiswa, tugas, pengajar, referensi;

//deklarasi constructor

**public** Mahasiswa () {

aktivitasmahasiswa = "BEM dan UKM";

tugas = "Banyak tapi susah";

pengajar = "Dosen";

referensi = "jurnal, artikel dan buku";

**public** Mahasiswa () {

aktivitassiswa = "Ekstrakurikuler dan OSIS";

tugas = "Banyak tapi Mudah";

pengajar = "Guru";

referensi = "buku paket";

System.***out***.println("===============> Constructor <================");

System.***out***.println(" Mahasiswa :"+

"\n kegiatan yang bisa dilakukan : "+ aktivitasmahasiswa +

"\n Tugas yang diberikan : " + tugas +

"\n pengajar : " + pengajar +

"\n referensi : " + referensi);

System.***out***.println("Siswa :"+

"\n kegiatan yang bisa dilakukan : "+ aktivitassiswa +

"\n Tugas yang diberikan : " + tugas +

"\n pengajar : " + pengajar +

"\n referensi : " + referensi);

System.***out***.println("===============> Method <=====================");

}

**void** Ambis(String a) { // method induk spesifik

System.***out***.println("Nugas " + 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 Mahasiswa :");

Mahasiswa objekO = **new** Mahasiswa(); // memanggil objek induk

objekO.Ambis("Bersama-sama"); // memanggil sifat spesifik induk

objekO.sukaMembaca("Majalah"); // memanggil method dengan variabel dapat diubah

System.***out***.println("\n Sifat Siswa :");

Siswa objekA = **new** Siswa(); //memanggil objek anak

objekA.sukaAmbis(9, "Sendirian"); //memanggil sifat spesifik anak yang diturunkan induk

objekA.sukaMembaca("Buku Ensiklopedia"); //memanggil method ke induk yang otomatis diturunkan tanpa deklarasi ulang di anak

} }

**class** Siswa **extends** Mahasiswa {

**void** sukaAmbis(**int** a, String b) {

System.***out***.println("Nonton Jam " + a + " Malam " + b);

}

**void** sukaAmbis(String a) { // method induk spesifik

System.***out***.println("Ambis " + a);

}

**void** sukaMembaca(String a) { // method induk umum bisa diubah anak

System.***out***.println("Suka Baca " + a);

}

}

Dengan hasil luaran

Sifat Mahasiswa :

===============> Constructor <================

Mahasiswa :

kegiatan yang bisa dilakukan : BEM dan UKM

Tugas yang diberikan : Banyak tapi susah

pengajar : Dosen

referensi : jurnal, artikel dan buku

===============> Method <=====================

Nugas Bersama-sama

Suka Baca Majalah

Sifat Siswa :

===============> Constructor <================

Mahasiswa :

kegiatan yang bisa dilakukan : BEM dan UKM

Tugas yang diberikan : Banyak tapi susah

pengajar : Dosen

referensi : jurnal, artikel dan buku

===============> Method <=====================

Nonton Jam 9 Malam Sendirian

Suka Baca Buku Ensiklopedia

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

Kami mengusulkan permasalahan ini dapat diatasi dengan cara Menambahkan Atribut Variabel dan membuat constructor dan method. Constructor Merupakan method yang namanya sama dengan nama class dimana method itu berada. Constructor digunakan untuk menginisialisasi nilai awal (membenikan nilai detault) pada atribut-atribut yang dimiliki oleh suatu object saat object pertama kali dibuat.

Method adalah kegiatan/aksi/perilaku yang dilakukan oleh objek. Manusia terdiri dari beragam objek dengan karakteristik serta sifat yang berbeda. Setiap objek memilki perilaku yang berbeda disimpan sebagai method seperti makan,berbicara,berpikir dan berjalan. Umumnya method berbentuk kata kerja karena perilaku yang dilakukan oleh objek.

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

1. Algoritma

Algoritma adalah langkah-langkah penyelesaian masalah.  
Membuat java project

1. Membuat Package
2. Membuat class
3. Membuat deklarasi atribut mahasiswa dalam variable

**public** **class** Mahasiswa {

//deklarasi atribut mahasiswa dalam variabel

String aktivitasmahasiswa, tugas, pengajar, referensi;

String aktivitassiswa, tugas, pengajar, referensi;

1. Deklarasikan Constructor Mahasiswa

//deklarasi constructor

**public** Mahasiswa () {

aktivitasmahasiswa = "BEM dan UKM";

tugas = "Banyak tapi susah";

pengajar = "Dosen";

referensi = "jurnal, artikel dan buku";

1. Deklarasikan Constructor Siswa

**public** Mahasiswa () {

aktivitassiswa = "Ekstrakurikuler dan OSIS";

tugas = "Banyak tapi Mudah";

pengajar = "Guru";

referensi = "buku paket";

1. Deklarasikan Method

**void** Ambis(String a) { // method induk spesifik

System.***out***.println("Nugas " + 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 Mahasiswa :");

Mahasiswa objekO = **new** Mahasiswa(); // memanggil objek induk

objekO.Ambis("Bersama-sama"); // memanggil sifat spesifik induk

objekO.sukaMembaca("Majalah"); // memanggil method dengan variabel dapat diubah

System.***out***.println("\n Sifat Siswa :");

Siswa objekA = **new** Siswa(); //memanggil objek anak

objekA.sukaAmbis(9, "Sendirian"); //memanggil sifat spesifik anak yang diturunkan induk

objekA.sukaMembaca("Buku Ensiklopedia"); //memanggil method ke induk yang otomatis diturunkan tanpa deklarasi ulang di anak

} }

1. Mendapatkan Luaran

Sifat Mahasiswa :

===============> Constructor <================

Mahasiswa :

kegiatan yang bisa dilakukan : BEM dan UKM

Tugas yang diberikan : Banyak tapi susah

pengajar : Dosen

referensi : jurnal, artikel dan buku

===============> Method <=====================

Nugas Bersama-sama

Suka Baca Majalah

Sifat Siswa :

===============> Constructor <================

Mahasiswa :

kegiatan yang bisa dilakukan : BEM dan UKM

Tugas yang diberikan : Banyak tapi susah

pengajar : Dosen

referensi : jurnal, artikel dan buku

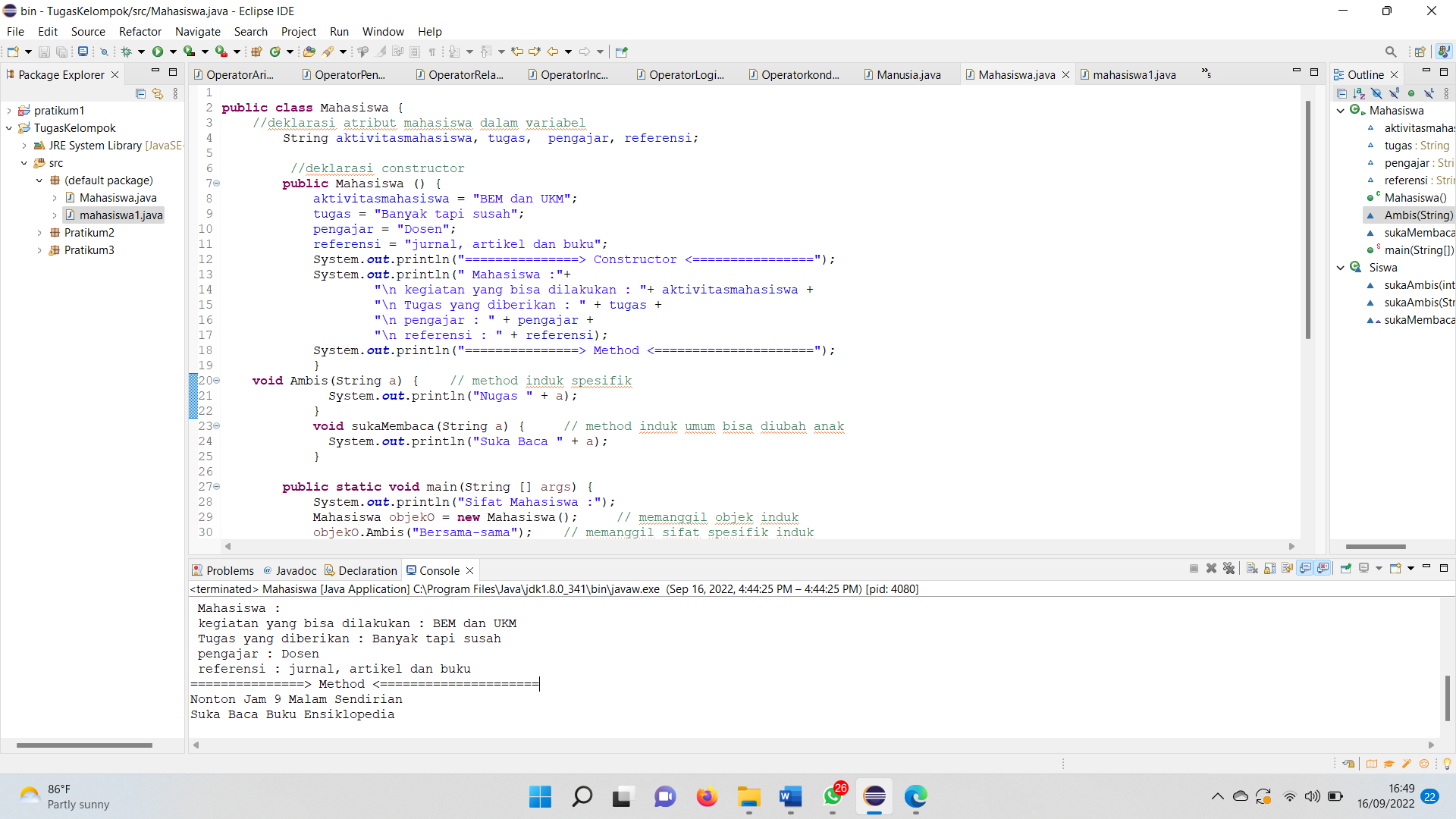
===============> Method <=====================

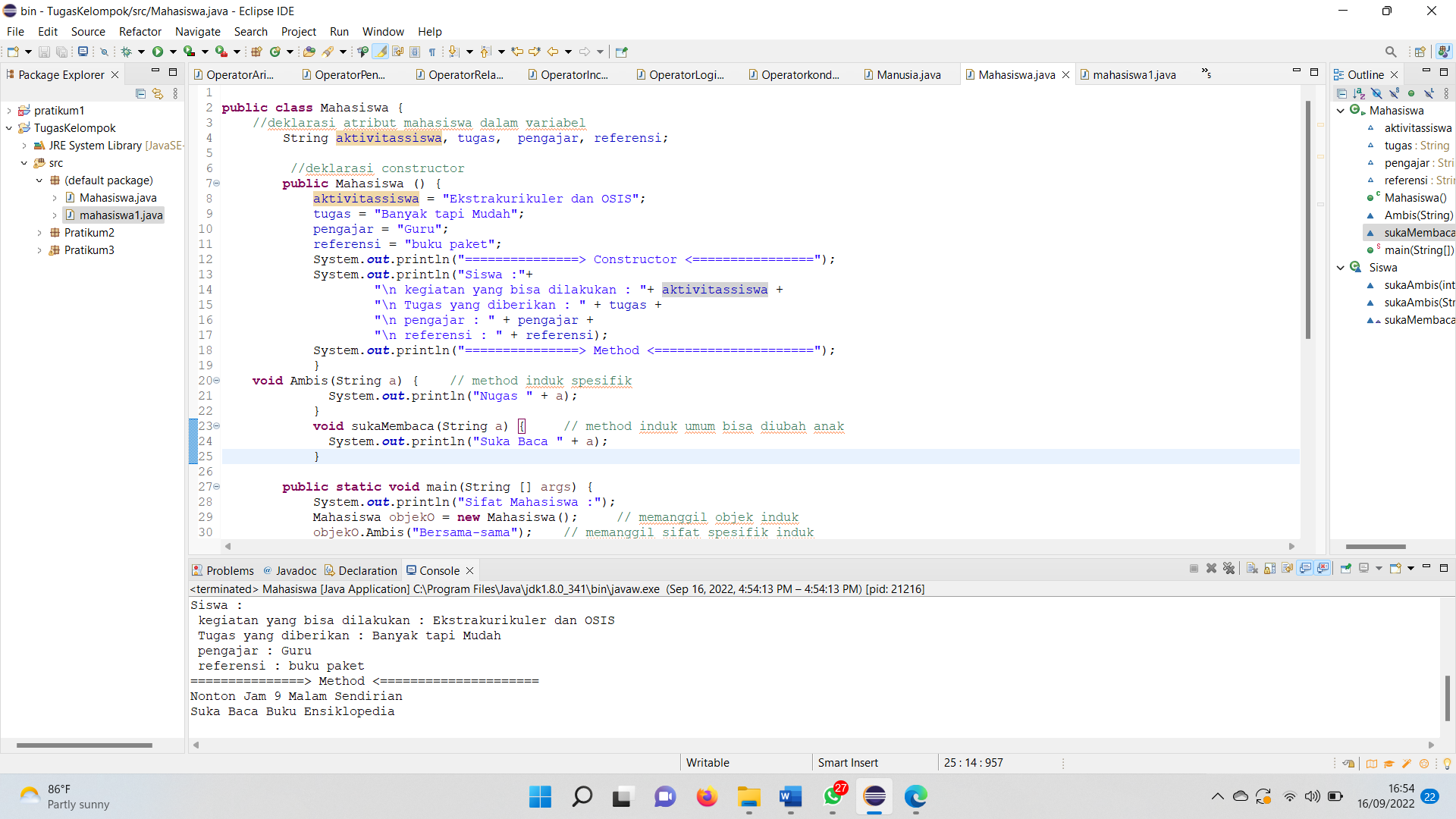
Nonton Jam 9 Malam Sendirian

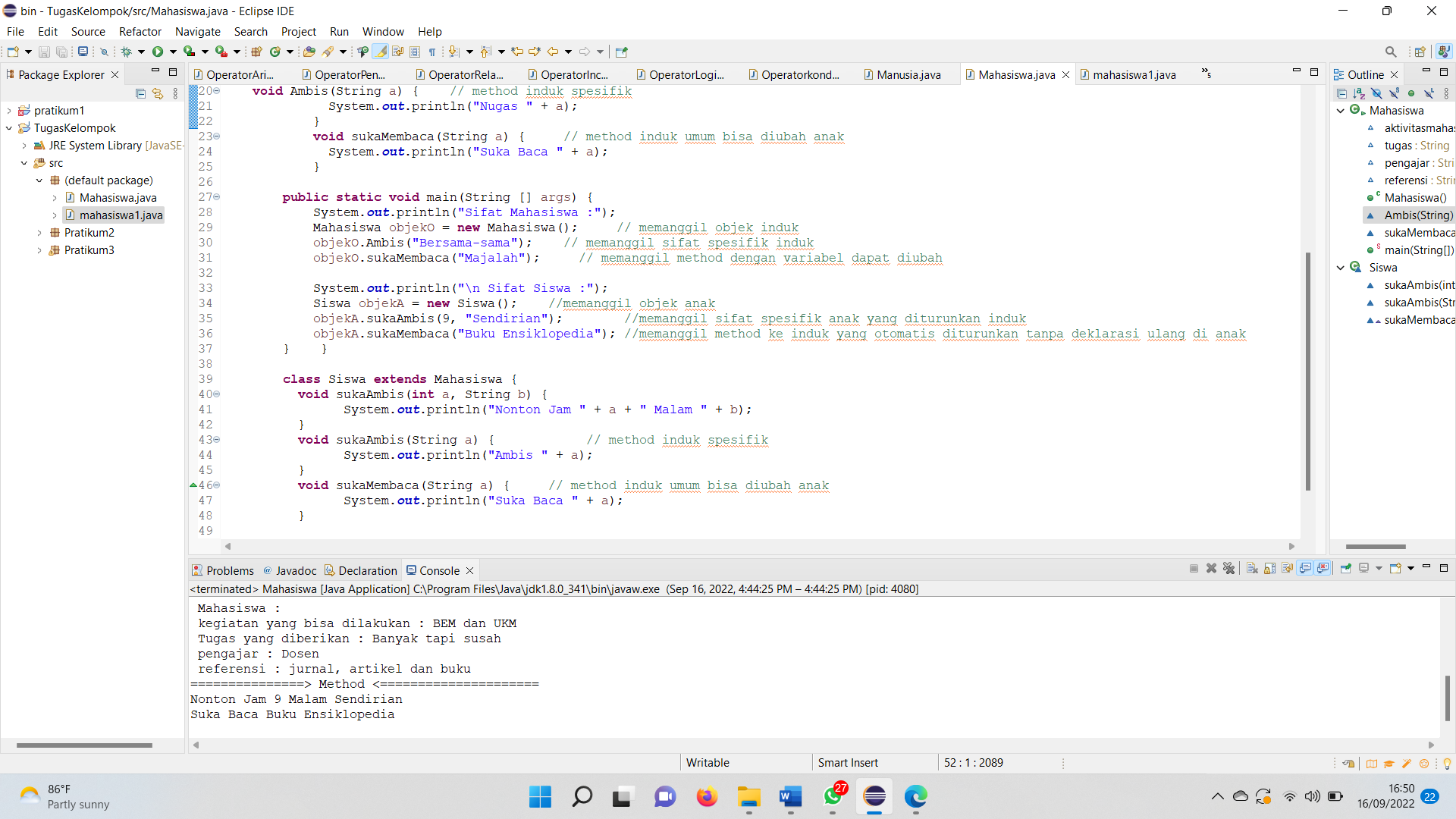
Suka Baca Buku Ensiklopedia

1. Kode Program dan Luaran
2. 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.   
Tipe data yang ditampilkan telah sesuai dengan kebutuhan dan permintaan data.

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

Pada program ini kami menggunakan induk kelas mahasiswa yang turunan dari kelasnya ialah siswa. Alasan kami mengambil kelas turunan siswa adalah karena mahasiswa memiliki sifat atau behavior yang dapat diturunkan ke siswa. Kami juga menambahkan Atribut Variabel dan membuat constructor dan method. Constructor Merupakan method yang namanya sama dengan nama class dimana method itu berada. Constructor digunakan untuk menginisialisasi nilai awal (membenikan nilai detault) pada atribut-atribut yang dimiliki oleh suatu object saat object pertama kali dibuat. Method adalah kegiatan/aksi/perilaku yang dilakukan oleh objek. Manusia terdiri dari beragam objek dengan karakteristik serta sifat yang berbeda. Setiap objek memilki perilaku yang berbeda disimpan sebagai method seperti makan,berbicara,berpikir dan berjalan. Umumnya method berbentuk kata kerja karena perilaku yang dilakukan oleh objek. Didalam pembuatan kelas, objek, dan method dalam java, perlu diperhatikan setiap aturannya untuk menghindari terjadinya error saat ingin melihat hasil atau luaran.

**Refleksi**

Pada saat kelompok kami mengerjakan tugas ini, kami banyak mempelajari hal baru tentang materi class, object, dan method. Kami juga dapat mengetahui bahwa kelas itu dapat diturunkan dengan mewarisi sifat dari induk kelasnya. Kami sempat mengalami kesulitan dalam membuat kode program untuk deklarasi constructor akan tetapi kami mencoba mencari letak dari kesalahannya dan memperbaikinya. Materi minggu ini cukup menantang bagi, dikarenakan kami dituntut untuk aktif mencari tahu tentang apa saja yang diminta didalam soal.