**LAPORAN HASIL PRAKTIKUM**

**ALGORITMA DAN STRUKTUR DATA**

**JOBSHEET 2**



NAMA : Handino Asa Galih r

NIM : 244107020237

KELAS : 1-E

PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLINEMA

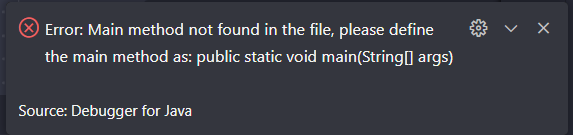
2024

Percobaan 1 : Deklarasi Class, Atribut dan Method

Kode Program

|  |
| --- |
| public class Mahasiswa12 {      // Deklarasi atribut      String nama;      String nim;      String kelas;  *double* ipk;    *void* tampilkanInformasi() {          System.out.println("Nama : " + nama);          System.out.println("Nim : " + nim);          System.out.println("Kelas : " + kelas);          System.out.println("Ipk : " + ipk);      }    *void* ubahKelas(String *kelasBaru*) {          kelas = *kelasBaru*;      }    *void* updateIpk(*double* *ipkBaru*) {          ipk = *ipkBaru*;      }        String nilaiKinerja() {          if (ipk >= 3.5) {              return "Kinerja sangat baik";          } else if (ipk >= 3.0) {              return "Kinerja baik";          } else if (ipk >= 2.0) {              return "Kinerja cukup";          } else {              return "Kinerja kurang";          }      }  } |

Verifikasi Hasil Percobaan



Pertanyaan!

1. a. Atribut (Properties/Data)

* Atribut adalah karakteristik atau sifat dari objek tersebut.
* Contoh pada class percobaan diatas yaitu nama, nim, kelas, ipk

b.Method (Behavior/Function)

* Fungsi Method yaitu Menggambarkan perilaku atau kemampuan dari objek.
* Contoh pada percobaan diatas yaitu tampilkanInformasi(), ubahKelas(), updateIpk(), nilaiKinerja()

2. Pada praktikum percobaan diatas terdapat 4 atribut yaitu :

- nama (String)

- nim (String)

- kelas (String)

- Ipk (Double)

3. Pada percobaan diatas terdapat 4 Method yaitu :

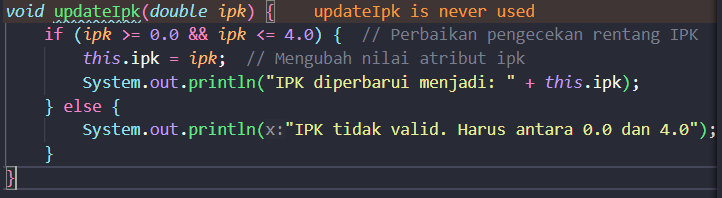
- tampilkanInformasi()

- ubahKelas()

- updateIpk()

- nilaiKinerja()

4. Modifikasi isi method tersebut sehingga IPK yang dimasukkan valid yaitu terlebih dahulu dilakukan pengecekan apakah IPK yang dimasukkan di dalam rentang 0.0 sampai dengan 4.0 .



5. Cara kerja Method nilaiKinerja() untuk mengevaluasi kinerja mahasiswa yaitu :

1. dengan menambahkan validasi menggunakan if-else, menggunakan Kondisi ipk >= 0,0 && ipk <= 4.0 untuk memastikan nilai yang di inputkan tetap berada dala rentang yang valid. Menggunakan operator && (AND) untuk memastikan kedua kondisi tersebut terpenuhi.

2.jika nilai ipk valid maka akan diperbarui

3. jika nilai ipk tidak valid maka akan menampilkan pesan error dan nilai ipk tidak berubah

6. Push and Commit ke Github

Percobaan 2 : Instansi Object, serta Mengakses Atribut dan Method

Kode Program

|  |
| --- |
| import java.util.Scanner;  public class MahasiswaMain12 {      public static *void* main(String[] *args*) {          Scanner sc = **new** Scanner(System.in);            Mahasiswa12 mhs1 = **new** Mahasiswa12();            mhs1.nama = "Muhammad Ali Farhan";          mhs1.nim = "2241720171";          mhs1.kelas = "SI 2J";          mhs1.ipk = 3.55;            mhs1.tampilkanInformasi();          mhs1.ubahKelas("SI 2K");          mhs1.updateIpk(3.60);          mhs1.tampilkanInformasi(); |

Hasil Run



Pertanyaan!

1. Jadi baris kode program yang digunakan untuk proses instansiasi adalah sebagai berikut :



Objek yang dihasillkan adalah objek dari kelas Mahasiswa12. dan, mhs1 digunakan untuk menyimpan data, seperti menampilkan informasi, mengubah kelas, dan memperbarui IPK.

2. 1. Untuk mengakses Atribut ada dua cara yaitu :

- jika atribut public, maka langsung diakses dengan object.atribut

- jika atribut private, maka dapat menggunakan getter dan setter untuk mengaksesnya.

2. Adapun untuk mengakses Method yaitu :

- jika method public, dapat memanggil melalui objek menggunakan object.method()

- Method dengan parameter dipanggil dengan memberikan nilai pada parameter yang diperlukan.

3. Hasil Output pemanggilan method pertama dan kedua berbeda karena pada pemanggilan kedua, ada perubahan yang terjadi pada data yang dimiliki objekl mhs1, sebelum method tampilkanInformasi() dipanggil lagi.

4. Push dan Commit ke Github

Percobaan 3 : Membuat Konstruktor

Kode Program

|  |
| --- |
| import java.util.Scanner;  public class MahasiswaMain12 {      public static *void* main(String[] *args*) {          Scanner sc = **new** Scanner(System.in);            Mahasiswa12 mhs1 = **new** Mahasiswa12();            mhs1.nama = "Muhammad Ali Farhan";          mhs1.nim = "2241720171";          mhs1.kelas = "SI 2J";          mhs1.ipk = 3.55;            mhs1.tampilkanInformasi();          mhs1.ubahKelas("SI 2K");          mhs1.updateIpk(3.60);          mhs1.tampilkanInformasi();            Mahasiswa12 mhs2 = **new** Mahasiswa12 ("Anissa Nabila", "2141720106", 3.25, "TI 2L");          mhs2.updateIpk(3.30);          mhs2.tampilkanInformasi();          sc.close();      }  } |

Hasil Run



Pertanyaan!

1. Pada class Mahasiswa di Percobaan 3, tunjukkan baris kode program yang digunakan untuk mendeklarasikan konstruktor berparameter!

|  |
| --- |
| public Mahasiswa12(String nama, String nim, double ipk, String kelas) {  this.nama = nama;  this.nim = nim;  this.ipk = ipk;  this.kelas = kelas;  } |

2. – Membuat objek mhs dari class Mahasiswa12

- Mengirim Nilai ke Konstruktor, yang menerima 4 parameter : nama, nim, ipk, kelas.

3. jika konstruktor pada class Mahasiswa12 dihapus yang terjadi adalah error. maka java tidak akan menyediakan konstruktor otomatis. Maka seharusnya menggunakan konstruktor berparameter saat membuat objek sari class tersebut

4. Tidak, methoddi dalam kelas Mahasiswa tidak perlu diakses secara berurutan setelah melakukan instansiasi objek. method di dalam kelas dapat diakses dengan terpisah atau dalam urutan apapun selama tidak ada ketergantungan antara method tersebut.

5. Kode Program

|  |
| --- |
| import java.util.Scanner;  public class MahasiswaMain12 {      public static *void* main(String[] *args*) {          Scanner sc = **new** Scanner(System.in);            Mahasiswa12 mhs1 = **new** Mahasiswa12();            mhs1.nama = "Muhammad Ali Farhan";          mhs1.nim = "2241720171";          mhs1.kelas = "SI 2J";          mhs1.ipk = 3.55;            mhs1.tampilkanInformasi();          mhs1.ubahKelas("SI 2K");          mhs1.updateIpk(3.60);          mhs1.tampilkanInformasi();            Mahasiswa12 mhs2 = **new** Mahasiswa12 ("Anissa Nabila", "2141720106", 3.25, "TI 2L");          mhs2.updateIpk(3.30);          mhs2.tampilkanInformasi();          Mahasiswa12 mhsHandino = **new** Mahasiswa12 ("Handino Asa Galih R", "244107020237", 3.45, "TI 1E");          mhsHandino.tampilkanInformasi();          sc.close();      }  } |

Latihan !