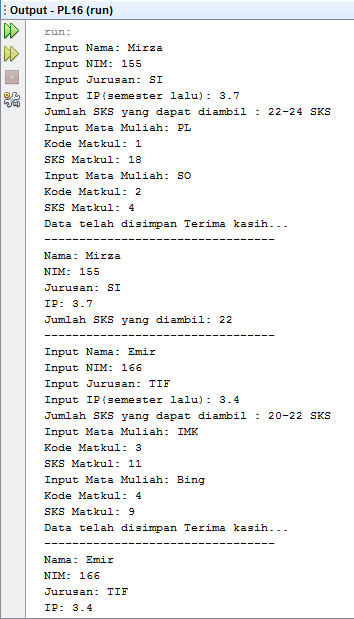
1. **DEFINISI MASALAH**
2. Buatlah sebuah sistem sederhana yang menyerupai Sistem Informasi Akademik Mahasiswa (SIAM), dengan ketentuan user menginputkan Nama, Nim, IP serta jurusan. Selain itu mahasiswa juga dapat memasukkan kode Mata kuliah, Nama Mata kuliah dan jumlah sks mata kuliah tersebut. Jumlah sks yang di ambil harus sesuai dengan IP yang di dapat pada semester lalu. Buat skenario dengan banyak mahasiswa minimal 3 orang.
3. **SOURCE CODE**

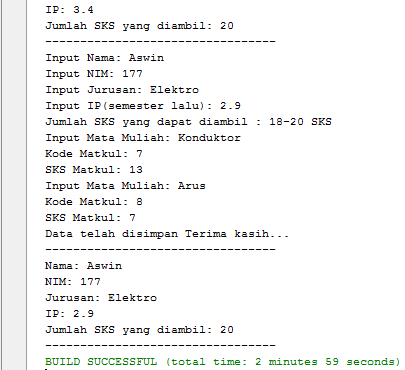
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113 | import java.util.Scanner;  class Mahasiswa {  String nama, nim, jur;  String matkul[] = new String[5];  String kode[] = new String[5];  int sksmatkul[] = new int[5];  int totalsks=0;  double ip;  int min, max;  public void rangeSKS() {  if (ip > 3.5) {  min = 22;  max = 24;  } else if (ip < 3.0) {  min = 18;  max = 20;  } else {  min = 20;  max = 22;  }  }  public void cetakData() {  System.out.println("---------------------------------");  System.out.println("Nama: " + nama);  System.out.println("NIM: " + nim);  System.out.println("Jurusan: " + jur);  System.out.println("IP: " + ip);  System.out.println("Jumlah SKS yang diambil: " + totalsks);  System.out.println("---------------------------------");  }  }  public class ClassDanObject {  public static void main(String[] args) {  Scanner in = new Scanner(System.in);  Mahasiswa m1 = new Mahasiswa();  System.out.print("Input Nama: ");  m1.nama = in.next();  System.out.print("Input NIM: ");  m1.nim = in.next();  System.out.print("Input Jurusan: ");  m1.jur = in.next();  System.out.print("Input IP(semester lalu): ");  m1.ip = in.nextDouble();  m1.rangeSKS();  System.out.println("Jumlah SKS yang dapat diambil : " + m1.min + "-" + m1.max + " SKS");  for (int i = 0; i < 5; i++) {  System.out.print("Input Mata Muliah: ");  m1.matkul[i] = in.next();  System.out.print("Kode Matkul: ");  m1.kode[i] = in.next();  System.out.print("SKS Matkul: ");  m1.sksmatkul[i] = in.nextInt();  m1.totalsks = m1.totalsks + m1.sksmatkul[i];  if (m1.totalsks >= m1.min && m1.totalsks <= m1.max) {  i=5;  }  }  System.out.println("Data telah disimpan Terima kasih...");  m1.cetakData();    Mahasiswa m2 = new Mahasiswa();  System.out.print("Input Nama: ");  m2.nama = in.next();  System.out.print("Input NIM: ");  m2.nim = in.next();  System.out.print("Input Jurusan: ");  m2.jur = in.next();  System.out.print("Input IP(semester lalu): ");  m2.ip = in.nextDouble();  m2.rangeSKS();  System.out.println("Jumlah SKS yang dapat diambil : " + m2.min + "-" + m2.max + " SKS");  for (int i = 0; i < 5; i++) {  System.out.print("Input Mata Muliah: ");  m2.matkul[i] = in.next();  System.out.print("Kode Matkul: ");  m2.kode[i] = in.next();  System.out.print("SKS Matkul: ");  m2.sksmatkul[i] = in.nextInt();  m2.totalsks = m2.totalsks + m2.sksmatkul[i];  if (m2.totalsks >= m2.min && m2.totalsks <= m2.max) {  i=5;  }  }  System.out.println("Data telah disimpan Terima kasih...");  m2.cetakData();    Mahasiswa m3 = new Mahasiswa();  System.out.print("Input Nama: ");  m3.nama = in.next();  System.out.print("Input NIM: ");  m3.nim = in.next();  System.out.print("Input Jurusan: ");  m3.jur = in.next();  System.out.print("Input IP(semester lalu): ");  m3.ip = in.nextDouble();  m3.rangeSKS();  System.out.println("Jumlah SKS yang dapat diambil : " + m3.min + "-" + m3.max + " SKS");  for (int i = 0; i < 5; i++) {  System.out.print("Input Mata Muliah: ");  m3.matkul[i] = in.next();  System.out.print("Kode Matkul: ");  m3.kode[i] = in.next();  System.out.print("SKS Matkul: ");  m3.sksmatkul[i] = in.nextInt();  m3.totalsks = m3.totalsks + m3.sksmatkul[i];  if (m3.totalsks >= m3.min && m3.totalsks <= m3.max) {  i=5;  }  }  System.out.println("Data telah disimpan Terima kasih...");  m3.cetakData();  }  } |

1. **PEMBAHASAN**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  22  23  24  25  26  27  28  29  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  58  59  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  84  85  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  110  111  112  113 | Pemanggilan fungsi scanner yang dimiliki oleh java  Pembuatan class dengan nama Mahasiswa  Inisialisasi variabel instance nama, nim, jur dengan tipe data String  Inisialisasi variabel matkul dengan tipe data String array dengan panjang array 5  Inisialisasi variabel kode dengan tipe data String array dengan panjang array 5  Inisialisasi variabel sksmatkul dengan tipe data integer array dengan panjang array 5  Insialisasi variabel totalsks dengan tipe data int dan nilai diberi nilai awal 0  Inisialisasi variabel ip dengan tipe data double  Inisialisasi variabel min, max dengan tipe data integer  Pembuatan method rangeSKS() tanpa mengembalikan nilai untuk menetukan batas pengambilan sks sesuai ip  Proses seleksi kondisi jika nilai ip > 3.5 maka  Inisialisasi nilai sks min = 22 dan  Inisialisasi nilai sks max = 24  Lain jika nilai ip < 3.0 maka  Inisialisasi nilai sks min = 18 dan  Inisialisasi nilai sks max = 20  Lainnya maka  Insialisasi nilai sks min = 20 dan  Insialisasi nilai sks max = 22  Pembuatan method cetakData() tanpa mengembalikan nilai untuk mencetak data mahasiswa yang sudah mengisi rencana studi  Proses mencetak garis pembatas  Proses mencetak nama mahasiswa  Proses mencetak nim mahasiswa  Proses mencetak jurusan mahasiswa  Proses mencetak nilai ip yang didapat mahasiswa  Proses mencetak total sks yang diambil oleh mahasiswa  Proses mencetak garis pembatas  Pembuatan class dengan nama ClassDanObject {  Inisialisasi main method sebagai program utama untuk dieksekusi  Pembuatan objek Scanner in untuk menginputkan masukkan user  Instansiasi objek m1 dari class Mahasiswa  Proses mencetak input user untuk nama mahasiswa  Inisialisasi input nama mahasiwa yang disimpan pada objek m1.nama bertipe String  Proses mencetak input user untuk nim mahasiswa  Inisialisasi input nim mahasiwa yang disimpan pada objek m1.nim bertipe String  Proses mencetak input user untuk jurusan mahasiswa  Inisialisasi input jurusan mahasiwa yang disimpan pada objek m1.jur bertipe String  Proses mencetak input user untuk nilai ip mahasiswa  Inisialisasi input ip mahasiwa yang disimpan pada objek m1.ip bertipe double  Pemanggilan method rangeSKS() untuk objek m1 untuk menetukan batas sks yang bisa diambil oleh masing-masing mahasiswa  Proses mencetak output banyaknya sks yang dapat diambil mahasiswa dalam satu semester sesuai ip  Proses perulangan i sebanyak 4 kali dengan menaikkan nilai variabel i  Proses mencetak input nama mata kuliah yang ingin diambil mahasiswa bertipe data String  Inisialisasi nama mata kuliah yang akan disimpan pada array matkul berurutan mulai dari indeks ke 0 pada objek m1  Proses mencetak input kode mata kuliah yang ingin diambil mahasiswa bertipe data String  Inisialisasi kode mata kuliah yang akan disimpan pada array kode berurutan mulai dari indeks ke 0 pada objek m1  Proses mencetak input jumlah sks yang ingin diambil dari matkul tersebut  Inisialisasi jumlah sks matkul yang akan disimpan pada array sksmatkul bertipe data integer pada objek m1  Proses perhitungan total sks sementara yang telah diambil mahasiswa  Proses seleksi kondisi jika total sks yg telah diambil telah memenuhi batas sks maksimum dan minimum yang telah disediakan maka  Inisialisasi nilai i = 5 agar program berhenti berjalan.  Proses mencetak output kesimpulan program mengenai data mahasiswa yang telah berhasil disimpan  Pemanggilan method cetakData untuk menampilkan seluruh data mahasiwa pada objek m1    Instansiasi objek m2 dari class Mahasiswa  Proses mencetak input user untuk nama mahasiswa  Inisialisasi input nama mahasiwa yang disimpan pada objek m2.nama bertipe String  Proses mencetak input user untuk nim mahasiswa  Inisialisasi input nim mahasiwa yang disimpan pada objek m2.nim bertipe String  Proses mencetak input user untuk jurusan mahasiswa  Inisialisasi input jurusan mahasiwa yang disimpan pada objek m2.jur bertipe String  Proses mencetak input user untuk nilai ip mahasiswa  Inisialisasi input ip mahasiwa yang disimpan pada objek m2.ip bertipe double  Pemanggilan method rangeSKS() untuk objek m2 untuk menetukan batas sks yang bisa diambil oleh mahasiswa  Proses mencetak output banyaknya sks yang dapat diambil mahasiswa dalam satu semester sesuai ip  Proses perulangan i sebanyak 4 kali dengan menaikkan nilai variabel i  Proses mencetak input nama mata kuliah yang ingin diambil mahasiswa bertipe data String  Inisialisasi nama mata kuliah yang akan disimpan pada array matkul berurutan mulai dari indeks ke 0 pada objek m2  Proses mencetak input kode mata kuliah yang ingin diambil mahasiswa bertipe String  Inisialisasi kode mata kuliah yang akan disimpan pada array kode berurutan mulai dari indeks ke 0 pada objek m2  Proses mencetak input jumlah sks yang ingin diambil dari matkul tersebut  Inisialisasi jumlah sks matkul yang akan disimpan pada array sksmatkul bertipe data integer pada objek m2  Proses perhitungan total sks sementara yang telah diambil mahasiswa  Proses seleksi kondisi jika total sks yg telah diambil telah memenuhi batas sks maksimum dan minimum yang telah disediakan maka  Inisialisasi nilai i = 5 agar program berhenti berjalan.  Proses mencetak output kesimpulan program mengenai data mahasiswa yang telah berhasil disimpan  Pemanggilan method cetakData untuk menampilkan seluruh data mahasiwa pada objek m2    Instansiasi objek m3 dari class Mahasiswa  Proses mencetak input user untuk nama mahasiswa  Inisialisasi input nama mahasiwa yang disimpan pada objek m3.nama bertipe String  Proses mencetak input user untuk nim mahasiswa  Inisialisasi input nim mahasiwa yang disimpan pada objek m3.nim bertipe String  Proses mencetak input user untuk jurusan mahasiswa  Inisialisasi input jurusan mahasiwa yang disimpan pada objek m3.jur bertipe String  Proses mencetak input user untuk nilai ip mahasiswa  Inisialisasi input ip mahasiwa yang disimpan pada objek m3.ip bertipe double  Pemanggilan method rangeSKS() untuk objek m3 untuk menetukan batas sks yang bisa diambil oleh mahasiswa  Proses mencetak output banyaknya sks yang dapat diambil mahasiswa dalam satu semester sesuai ip  Proses perulangan i sebanyak 4 kali dengan menaikkan nilai variabel i  Proses mencetak input nama mata kuliah yang ingin diambil mahasiswa bertipe data String  Inisialisasi nama mata kuliah yang akan disimpan pada array matkul berurutan mulai dari indeks ke 0 pada objek m3  Proses mencetak input kode mata kuliah yang ingin diambil mahasiswa bertipe String  Inisialisasi kode mata kuliah yang akan disimpan pada array kode berurutan mulai dari indeks ke 0 pada objek m3  Proses mencetak input jumlah sks yang ingin diambil dari matkul tersebut  Inisialisasi jumlah sks matkul yang akan disimpan pada array sksmatkul bertipe data integer pada objek m3  Proses perhitungan total sks sementara yang telah diambil mahasiswa  Proses seleksi kondisi jika total sks yg telah diambil telah memenuhi batas sks maksimum dan minimum yang telah disediakan maka  Inisialisasi nilai i = 5 agar program berhenti berjalan.  Proses mencetak output kesimpulan program mengenai data mahasiswa yang telah berhasil disimpan  Pemanggilan input cetakData untuk menampilkan seluruh data mahasiwa pada objek m3  End main method  End main class |

1. **SCREENSHOT PROGRAM**

****

****

1. **PRAKTIKUM**
2. Apakah yang disebut dengan variabel instance dan lokal variabel? Jelaskan perbedaanya!

>Variabel instance adalah variabel yang di deklarasikan di dalam kelas tetapi diluar method, konstruktor maupun blok statement.

Lokal variabel adalah variabel yang di deklarasikan di dalam method, konstruktor ataupun blok statement.

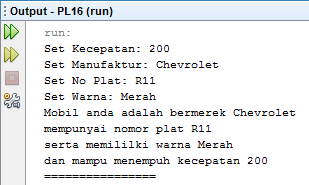
Perbedaan antara variabel instance dan lokal variabel yaitu Lokal variable tidak mempunyai default value sedangkan variabel instance punya nilai awal. Lalu kita tidak bisa memberikan access modifier pada local variable pada lokal variable sedangkan Instance variabel dapat menggunakan access modifier seperti private, public, protected, dan default.

1. Lakukan percobaan diatas dan benahi jika menemukan kesalahan!

>Program berjalan dengan baik dan tidak ada kesalahan

1. Rubah kode pada mainMobil diatas menjadi proses meminta masukan dari user dan buat menjadi interaktif!

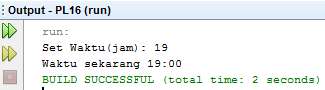
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16 | import java.util.Scanner;  public class MainMobil {  public static void main(String[] args) {  Scanner in = new Scanner(System.in);  //instan objek bernama m1  Mobil m1 = new Mobil();  System.out.print("Set Kecepatan: ");  m1.setKecepatan(in.nextInt());  System.out.print("Set Manufaktur: ");  m1.setManufaktur(in.next());  System.out.print("Set No Plat: ");  m1.setNoPlat(in.next());  System.out.print("Set Warna: ");  m1.setWarna(in.next());  m1.displayMessage();  System.out.println("================"); |



1. Tambahkan method pada class mobil bernama setWaktu yang berparameter double, yang kemudian disimpan pada variabel waktu! (Ketetuannya adalah user harus menginputkan dalam satuan jam)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | public class Mobil {  private double waktu;    public void setWaktu(double waktu){  this.waktu=waktu;  System.out.printf("Waktu sekarang %.0f%s",waktu,":00");  } |

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | public class MainMobil {  public static void main(String[] args) {  Scanner in = new Scanner(System.in);  //instan objek bernama m1  Mobil m1 = new Mobil();  System.out.print("Set Waktu(jam): ");  m1.setWaktu(in.nextDouble()); |



1. Tambahkan method bernama rubahSekon mempunyai parameter bertipe double dan hanya dapat dipanggil pada class mobil. Method ini memiliki fungsi untuk merubah masukan user yaitu jam menjadi sekon. Method tersebut di panggil pada method setWaktu dengan nilai parameter adalah nilai dari variabel parameter method setWaktu!

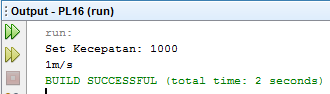
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | public class Mobil {  private double waktu;  public void setWaktu(double jam){  this.waktu=jam;  rubahSekon(jam);  System.out.printf("Detik sekarang %.0f",waktu);  }  private double rubahSekon(double jam){  return waktu=3600\*jam;  } |

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | public class MainMobil {  public static void main(String[] args) {  Scanner in = new Scanner(System.in);  //instan objek bernama m1  Mobil m1 = new Mobil();  System.out.print("Set Waktu(jam): ");  m1.setWaktu(in.nextDouble()); |

1. Tambahkan method pada class mobil dan hanya dapat dipanggil pada class mobil bernama rubahKecepatan yang mempunyai fungsi untuk merubah format kecepatan yang awalnya km/h menjadi m/s. Dipanggil di method setKecepatan!

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | public class Mobil {  private int kecepatan;  public void setKecepatan(int i) {  rubahKecepatan(i);  System.out.println(kecepatan+"m/s");  }  private int rubahKecepatan(int i){  return kecepatan =i/1000;  } |

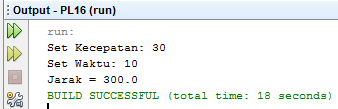
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | public class MainMobil {  public static void main(String[] args) {  Scanner in = new Scanner(System.in);  //instan objek bernama m1  Mobil m1 = new Mobil();  System.out.print("Set Kecepatan: ");  m1.setKecepatan(in.nextInt()); |



1. Tambahkan method pada class mobil bernama hitungJarak yang mempunyai aksi untuk menghitung jarak yang dapat di tempuh oleh mobil dengan rumus jarak = kecepatan \* waktu!

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | public class Mobil {  private int kecepatan;  private double waktu;    public void hitungJarak(){  double jarak = kecepatan\*waktu;  System.out.println("Jarak = "+jarak);  }  public void setKecepatan(int i) {  kecepatan = i;  }  public void setWaktu(double jam){  this.waktu=jam;  } |

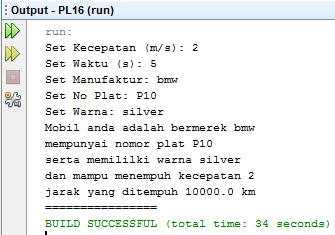
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | import java.util.Scanner;  public class MainMobil {  public static void main(String[] args) {  Scanner in = new Scanner(System.in);  //instan objek bernama m1  Mobil m1 = new Mobil();  System.out.print("Set Kecepatan: ");  m1.setKecepatan(in.nextInt());  System.out.print("Set Waktu (jam): ");  m1.setWaktu(in.nextDouble());  m1.hitungJarak(); |



1. Tambahkan informasi jarak yang dapat ditempuh pada method displayMessage kemudian rubah satuannya yang awalnya m (meter) menjadi km (kilometer)!

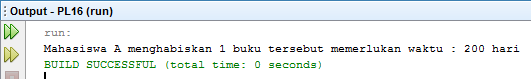
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35 | public class Mobil {  private String noPlat;  private String warna;  private String manufaktur;  private int kecepatan;  private double waktu;  public double jarak;  public void hitungJarak(){  jarak = kecepatan\*waktu;  jarak = jarak\*1000;  }  public void setKecepatan(int i) {  kecepatan = i;  }  public void setWaktu(double jam){  this.waktu=jam;  }  public void setNoPlat(String s) {  noPlat = s;  }  public void setWarna(String s) {  warna = s;  }  public void setManufaktur(String s) {  manufaktur = s;  }  public void displayMessage() {  System.out.println("Mobil anda adalah bermerek " + manufaktur);  System.out.println("mempunyai nomor plat " + noPlat);  System.out.println("serta memililki warna " + warna);  System.out.println("dan mampu menempuh kecepatan " + kecepatan);  System.out.println("jarak yang ditempuh "+ jarak + " km");  }  } |

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18 | public class MainMobil {  public static void main(String[] args) {  Scanner in = new Scanner(System.in);  //instan objek bernama m1  Mobil m1 = new Mobil();  System.out.print("Set Kecepatan (m/s): ");  m1.setKecepatan(in.nextInt());  System.out.print("Set Waktu (s): ");  m1.setWaktu(in.nextDouble());  m1.hitungJarak();  System.out.print("Set Manufaktur: ");  m1.setManufaktur(in.next());  System.out.print("Set No Plat: ");  m1.setNoPlat(in.next());  System.out.print("Set Warna: ");  m1.setWarna(in.next());  m1.displayMessage();  System.out.println("================"); |



1. Mahasiswa A ingin menulis pada sebuah buku tulis yang ingin dia miliki, isi lembar buku tersebut adalah 50 lembar. Setiap harinya ia menulis sebanyak 100 kata perhari yang cukup untuk 1/2 halaman buku. Buatlah rumus untuk menghitung berapa lama ia menghabiskan 1 buku tersebut serta identifikasilah objek, dan karakteristiknya kemudian implementasikan dalam bentuk class.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21 | class buku{  double hlmbuku;  int kataperhari = 100;  double hlmperhari = 0.5;  int waktu;  public void setBooks(double x) {  hlmbuku = x \* 2;  waktu = (int) (hlmbuku / hlmperhari);  }  public void getBooks() {  System.out.println("Mahasiswa A menghabiskan 1 buku tersebut memerlukan waktu : " + waktu + " hari");  }  }  public class Bukutulis {  public static void main(String[] args) {  buku bukutulis1 = new buku();  bukutulis1.setBooks(50);  bukutulis1.getBooks();  }  } |



1. **KESIMPULAN**

OOP (Object Oriented Programming) adalah suatu metode pemrograman yang berorientasi kepada objek. Tujuan dari OOP diciptakan adalah untuk mempermudah pengembangan program dengan cara mengikuti model yang telah ada di kehidupan sehari-hari. Jadi setiap bagian dari suatu permasalahan adalah objek. Objek itu sendiri merupakan gabungan dari beberapa objek yang lebih kecil lagi.

Untuk menginstansiasi atau membuat sebuah object, kita bisa menggunakan perintah new dengan sebuah nama class yang akan dibuat sebagai instance dari class tersebut.

Nama\_kelas nama\_object = new nama\_kelas (parameter);

Date hari = new Date();

hari adalah object reference dari class Date yang akan digunakan untuk mengakses

class Date. Sedangkan operator new adalah operator yang akan menghasilkan hari sebagai reference ke instance dari class Date().

Class

Class adalah penggambaran sebuah objek dan juga tentang fungsi objek itu sendiri. Di dalam class dideklarasikan variable dan method yang dimiliki objek. Proses pembuatan obyek dari sebuah class disebut Instantiation. Class seperti cetakan kue, dimana kue yg dihasilkan dari cetakan kue itu adalah object.

Attribute

Atribut adalah variable yang mengitari class, dengan nilai datanya bisa ditentukan di object. Variable digunakan untuk menyimpan nilai yang nantinya akan digunakan pada program.

Behaviour/Method

Method atau behaviour adalah suatu operasi berupa fungsi-fungsi yang dapat dikerjakan oleh suatu objek. Method didefinisikan pada class, akan tetapi dipanggil melalui object. Method sangat menentukan perilaku objek.

Object

Object adalah instance dari class, hasil bentuk nyata dari sebuah class.