



PRORAM STUDI TEKNIK INFORMATIKA-D3

FAKULTAS ILMU KOMPUTER

UNIVERSITAS DIAN NUSWANTORO

Modul Praktikum Dasar Pemrograman

Ibnu Utomo WM, M.Kom

Dr. Fikri Budiman, M.Kom

Hanya untuk keperluan pembelajaran di lingkungan Program Studi
Teknik Informatika-D3, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro



PRORAM STUDI TEKNIK INFORMATIKA-D3

FAKULTAS ILMU KOMPUTER

UNIVERSITAS DIAN NUSWANTORO

Modul Praktikum Dasar Pemrograman

Ibnu Utomo WM, M.Kom

Dr. Fikri Budiman, M.Kom

Hanya untuk keperluan pembelajaran di lingkungan Program Studi
Teknik Informatika-D3, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro

Modul Praktikum Dasar Pemrograman

Tim Penyusun Ibnu Utomo WM, M.Kom
 Dr. Fikri Budiman, M.Kom

Pengulas Tim Pengampu

Edisi Pertama

Agustus 2017

Program Studi Teknik Informatika-D3

Fakultas Ilmu Komputer

Universitas Dian Nuswantoro

LEMBAR PENETAPAN

Menetapkan bahwa Modul Praktikum **Dasar Pemrograman** mulai digunakan di semester ganjil Tahun Akademik 2017 / 2018 di lingkungan Program Studi Teknik Informatika-D3, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro.

Semarang, 20 Agustus 2017

Ketua Program Studi TI-D3

Muslih, M.Kom
NPP: 0686.11.1996.082

VISI DAN MISI
PROGRAM STUDI TEKNIK INFORMATIKA-D3

Visi

“Menjadi Pilihan Utama Dalam Pendidikan Vokasi Bidang Teknologi Informasi, dan Mampu Menghasilkan Lulusan Dengan Keahlian Teknologi Web dan Mobile yang Cakap Dalam Bidang *Technopreneurship*”

Misi

- a. Tuntutan dosen profesional yang bermutu dan Ahli dalam bidang kompetensi yang diajarkan, dan mengembangkan riset sesuai dengan kebutuhan keilmuan pada program studi.
- b. Menyelenggarakan pendidikan berbasis teknologi informasi yang profesional dan bermutu yang cakap dalam bidang technopreuner, dan meningkatkan kompetensi mahasiswa dengan sertifikasi uji kompetensi pada bidang web dan mobile programming.
- c. Menyelenggarakan kerjasama dengan berbagai lembaga pihak luar, sehingga dapat mengontrol penyelenggaraan pendidikan yang selalu mutakhir dan dapat diterapkan secara tepat guna.

PROFIL LULUSAN
PROGRAM STUDI TEKNIK INFORMATIKA-D3

Profil	Deskripsi Profile
Programmer	Dapat mengimplemetasikan analisis problem, obyek serta spesifikasi permasalahan hingga tertuang dalam logical flow maupun rancangan model sehingga dapat diselesaikan dengan bahasa pemrograman yang dikuasai khususnya web dan mobil aplikasi dengan penerapan multimedia.
Network Engineer	Dapat Mengelola perangkat-perangkat networking, melakukan monitoring, troubleshooting serta mengaplikasikan keamanan jaringan agar berjalan dengan baik dari sebuah perusahaan maupun instansi.
Database Administrator	Merancang, mengelola serta mengimplementasikan database secara fisik dan mengontrol security, integritas database yang ditanganinya.
Web Developer	Sebagai profesional dari desain sampai dengan implementasi perangkat lunak dengan platform webbased aplication.
Technopreneur	Sebagai pelaku usaha dalam pemanfaatan teknologi yang sedang berkembang pesat khususnya di dunia IT

**TATA TERTIB PENGGUNAAN
LABORATORIUM KOMPUTER
UNIVERSITAS DIAN NUSWANTORO**

1. Pengguna Laboratorium komputer adalah orang yang tercatat resmi sebagai mahasiswa UDINUS, kecuali peserta pelatihan atau semacamnya.
2. Pengguna Laboratorium komputer harus berperilaku sopan dan menggunakan pakaian sopan dan rapi sesuai ketentuan UNIVERSITAS (tidak memakai kaos oblong atau sandal).
3. Mahasiswa wajib mempunyai, menggunakan dan menjaga keamanan user login milik sendiri dengan rutin mengganti password secara periodik.
4. User login mahasiswa terkait dengan quota space drive yang ada di server lab komputer.
5. Tidak diperbolehkan membawa makanan, minuman kedalam laboratorium komputer dan harus membuang sampah pada tempatnya.
6. Semua barang yang ada di lingkungan laboratorium komputer merupakan barang milik Universitas Dian Nuswantoro.
7. Penggunaan perangkat tambahan yang tidak permanen terpasang di laboratorium komputer harus seijin laboran lab yang digunakan.
8. Tidak diperbolehkan merubah konfigurasi, melepas rangkaian, mencorat-coret, dan merusak barang-barang di lingkungan laboratorium komputer.
9. Tidak diperbolehkan membawa keluar barang-barang di lingkungan laboratorium komputer tanpa ijin.
10. Barang-barang yang dibawa dari luar untuk ditempatkan di lab. Komputer harus seijin Ka. Lab untuk didata keperluan dan peruntukannya.
11. Penggunaan lab diluar jam kuliah atau untuk keperluan lain selain perkuliahan harus seijin Ka Laboratorium, yang ketentuannya akan diatur kemudian.
12. Pengguna Lab. Wajib melaporkan kejadian yang membayakan orang lain di lingkungan laboratorium pada pihak yang berwenang (laboran/Satpam).

Semarang, 9 Maret 2015
Ka. UPT. Laboratorium Komputer

Elkaf Rahmawan Pramudya, M. Kom
NPP : 0686.11.1998.147

PERATURAN PRAKTIKUM

1. Mahasiswa wajib membawa modul praktikum.
2. Mahasiswa wajib hadir minimal 75% dari seluruh pertemuan praktikum di lab. Mahasiswa dilarang membuka program aplikasi yang tidak berhubungan dengan praktikum.
3. Durasi kegiatan praktikum adalah 200 menit per pertemuan.
4. Toleransi keterlambatan adalah kurang dari 30 menit.

KATA PENGANTAR

Puji syukur kehadiran Allah SWT dan tak lupa shalawat serta salam senantiasa tercurah kepada Rasulullah Muhammad SAW. Modul Praktikum Dasar Pemrograman sebagai pendukung praktikum telah dapat terselesaikan.

Penyusun berharap bahwa modul ini dapat membantu mahasiswa dalam menyelesaikan mata kuliah serta mendukung terwujudnya capaian pembelajaran yang telah ditentukan.

Penyusun mengucapkan banyak terimakasih kepada semua pihak yang telah membantu penyelesaian modul ini.

Semarang, Agustus 2017

Tim Penyusun

Daftar Isi

Cover.....	1
Cover Dalam	2
Lembar Penetapan.....	3
Visi dan Misi.....	4
Profil Lulusan	5
Tata Tertib Penggunaan Laboratorium Komputer	6
Peraturan Praktikum.....	7
Kata Pengantar	8
Daftar Isi	9
Praktikum 1 Pendahuluan.....	11
I. Struktur Program Java.....	11
II. Mengkompilasi Program Java	11
III. Menggunakan IDE Netbeans.....	12
Praktikum 2 Output, Proses dan Input.....	19
I. Program Java Pertama	19
II. Penggunaan Variabel	19
III. Program yang terdapat proses (perhitungan matematika).....	22
IV. Menginputkan data dari keyboard.....	23
V. Program Konversi Suhu.....	24
VI. Soal Latihan.....	25
Praktikum 3, 4 Kondisi	26
I. Program dengan pernyataan if	26
II. Program dengan pernyataan if...else.....	27
III. Program dengan pernyataan if...else...bertingkat.....	27
IV. Program dengan pernyataan switch...case.....	28
V. Program dengan kondisi majemuk	30
VI. Soal Latihan.....	30
Praktikum 5, 6 Perulangan	32
I. Program dengan pernyataan for	32
II. Program dengan pernyataan break	34
III. Program dengan variabel counter	35
IV. Program dengan pernyataan while.....	35

V.	Program dengan pernyataan do...while.....	36
VI.	Soal Latihan.....	37
	Praktikum 7 Modul / Fungsi.....	38
I.	Membuat fungsi pada Program java.....	38
II.	Membuat fungsi yang mengembalikan suatu nilai	41
III.	Membuat fungsi dengan fungsi rekursif	42
IV.	Soal Latihan.....	43
	Praktikum 8, 9 Array	44
I.	Program dengan Array 1 dimensi.....	44
II.	Program dengan Array 2 dimensi.....	46
III.	Soal Latihan.....	48
	Praktikum 10, 11 Operasi File	49
I.	Program untuk membaca file teks	49
II.	Program untuk menulis file teks	50
III.	Soal Latihan.....	51
	Praktikum 12 Projek Akhir	52



Pendahuluan

Tujuan Pembelajaran

Setelah mempelajari bab ini, diharapkan mampu:

1. Memberikan pengetahuan tentang struktur dasar program Java
2. Memberikan pengetahuan tentang proses mengkompilasi program Java
3. Memberikan pengetahuan tentang penggunaan IDE Netbeans untuk membuat program Java

Pokok Bahasan

- Struktur dasar
- Kompilasi Program

I. Struktur Program Java

Program Java memiliki struktur sebagai berikut :

```
//NamaKelas.java
public class NamaKelas {
    public static void main (String[] args) {
        //kode program
    }
}
```

Yang dicetak tebal adalah *reserved word* (kata tercadang dari bahasa Java yang harus ditulis seperti adanya. Untuk pembuatan program dapat dilakukan menggunakan editor (seperti Notepad) atau menggunakan IDE (seperti Netbeans)

II. Mengkompilasi Program Java

Untuk mengkompilasi program dalam bahasa Java, maka minimal kita harus menginstall software JDK (*Java Development Kit*). Yang paling sederhana adalah dengan menggunakan program **javac.exe** yang merupakan bawaan dari JDK dengan perintah :

Prompt > javac NamaKelas.java(enter)

Yang akan dihasilkan **NamaKelas.class**. Dan untuk menjalankannya dapat digunakan program **java.exe** dengan perintah :

Prompt> java NamaKelas(enter)

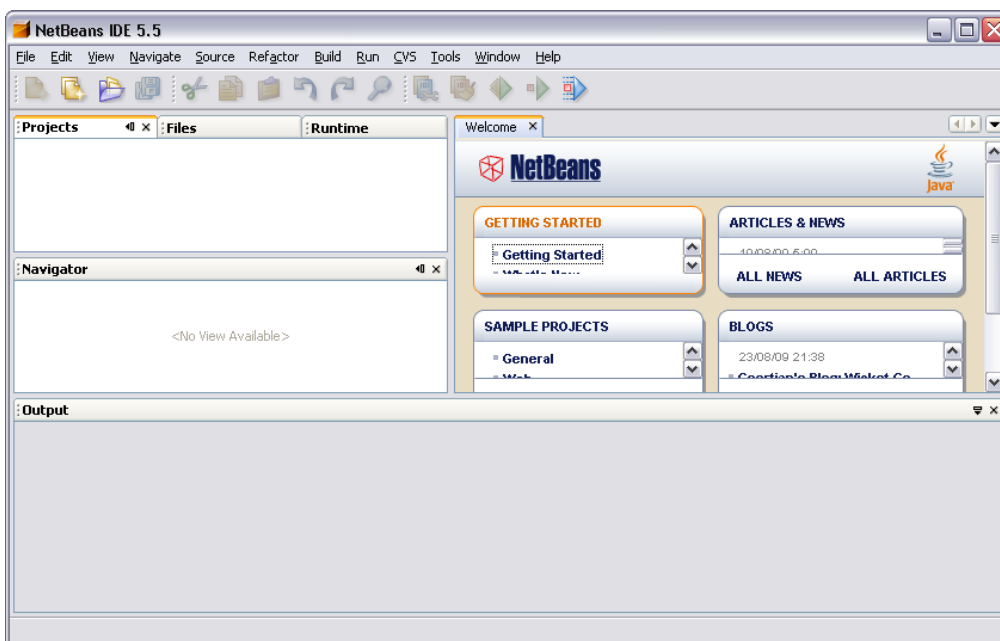
Catatan : Agar program **javac.exe** dan **java.exe** dapat dipanggil dari semua direktori, maka program tersebut harus tersimpan dalam sistem PATH. Caranya adalah dengan klik kanan pada **My Computer**, pilih **properties**. Kemudian pada tab **Advanced**, tekan tombol **Environment Variables**. Pada Panel **System Variables**, pilih Key PATH dan tekan tombol **Edit**. Pada Window Edit System Variables masukkan PATH direktori program tersebut berada. (mis : C:\Program Files\Java\jdk1.6.0_03\bin)

III. Menggunakan IDE Netbeans

Untuk menggunakan IDE NetBeans, maka harus terlebih dahulu dilakukan instalasi software NetBeans (mis : jdk-6u16-nb-6_7_1-windows-ml.exe). Untuk membuat project baru, maka aktifkan NetBeans, dan kemudian pilih menu File-New Project sehingga muncul Window seperti pada gambar 1.

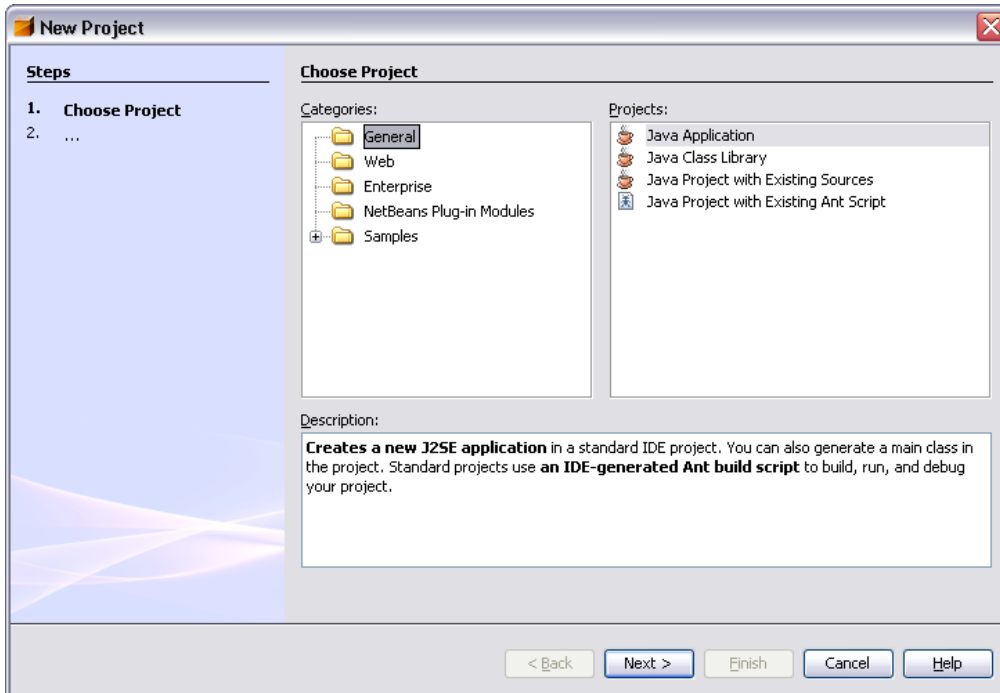
Langkah – langkah praktikum

1. Pilih Start → Program → NetBeands
2. Anda akan peroleh tampilan awal sebagai berikut :

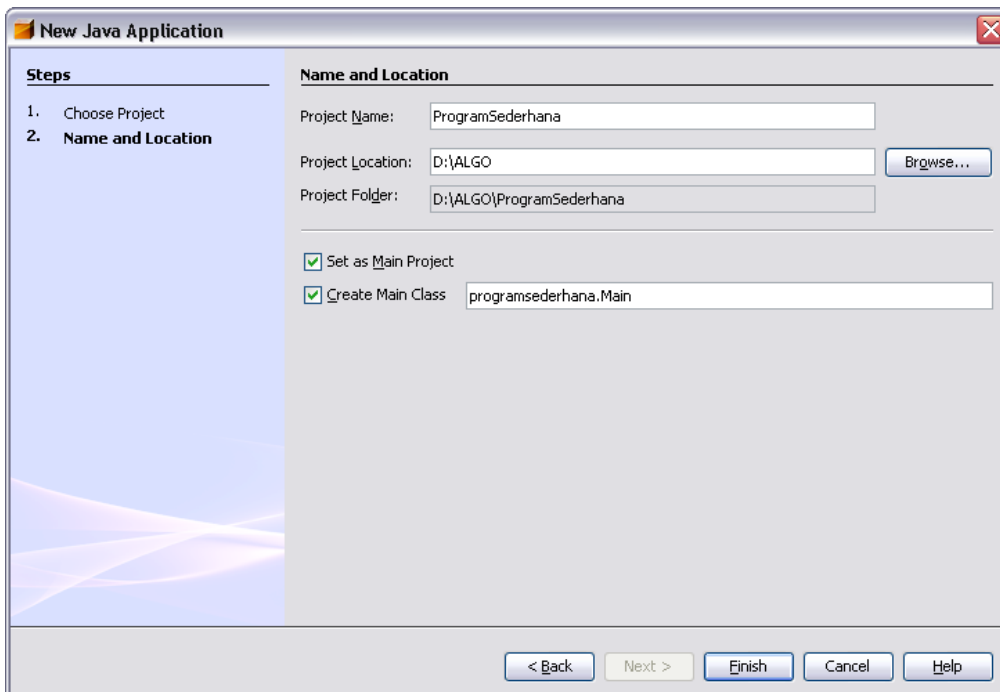


3. Pilih menu File

- Pilih sub menu New Project. Akan muncul layar sebagai berikut

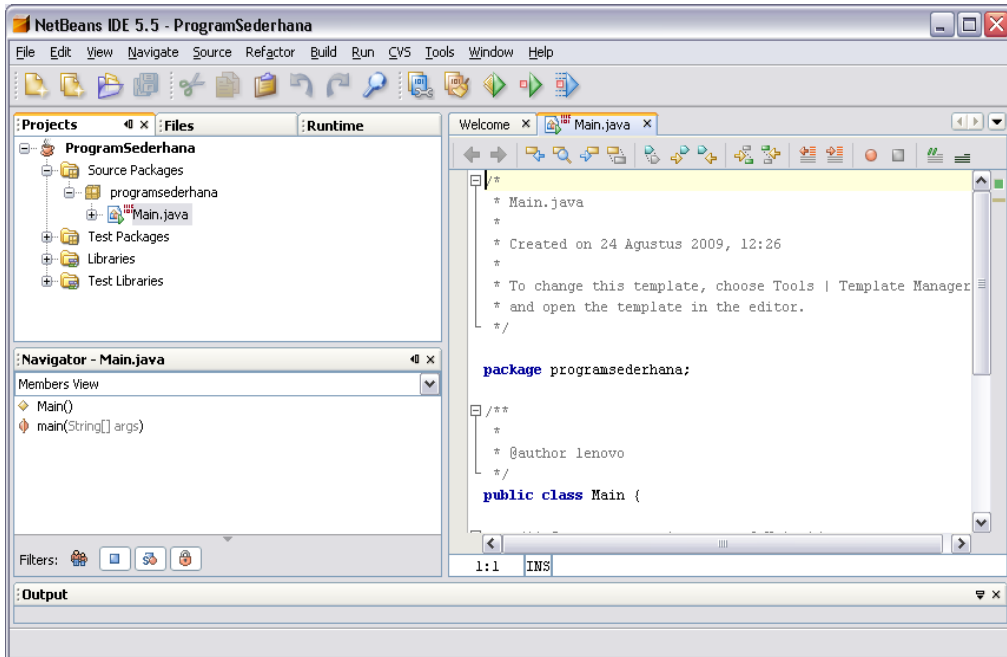


- Pilih General, pada pilihan Categories
- Pilih Java Application pada pilihan Project
- Klik Next >
- Akan muncul layar sebagai berikut

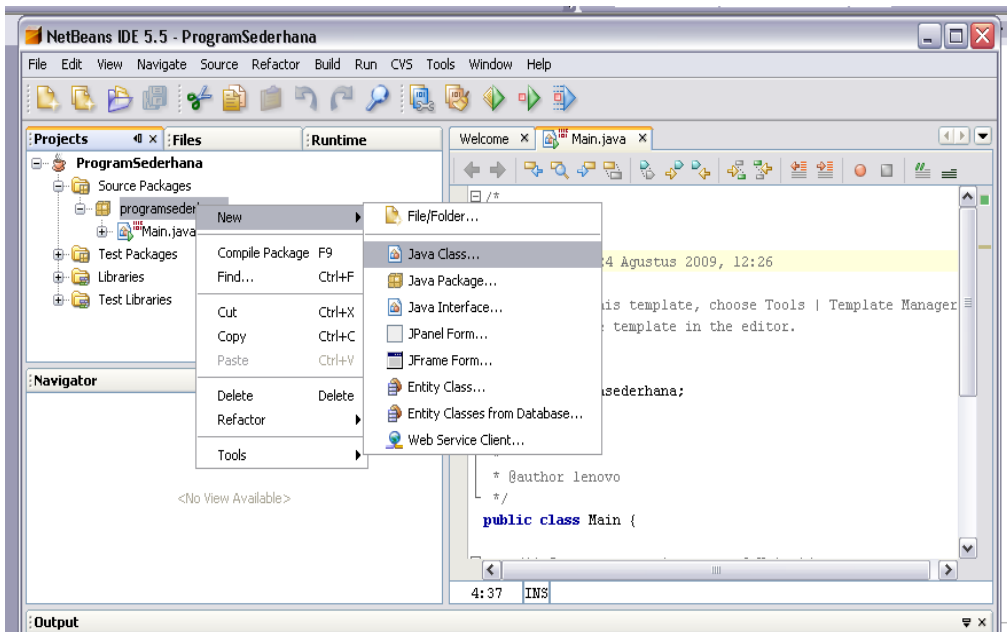


- Tulis nama project pada isian Project Name

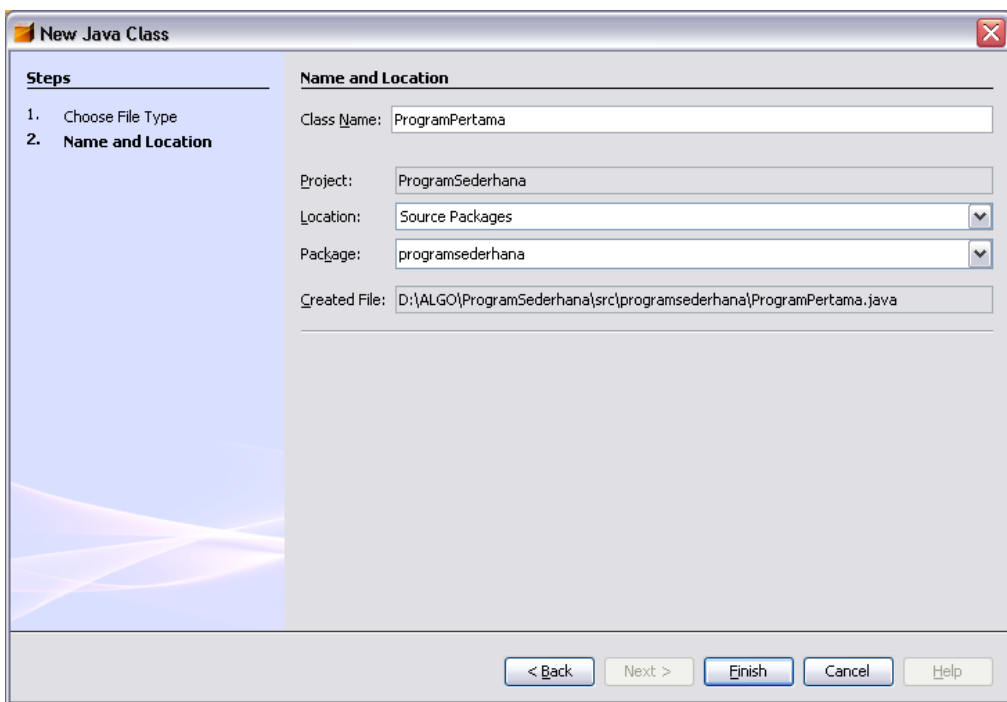
10. Pilih lokasi penyimpanan pada isian Project Location. Anda bisa menuliskan lokasinya atau menggunakan tombol Browse
11. Centang pilihan Set as Main Project, jika Anda berharap ini akan menjadi project utama Anda.
12. Centang pilihan Create Main Class jika Anda akan langsung membuat kelas main. Atau Anda bisa mengganti nama kelasnya(bukan kelas main)
13. Kemudian klik finish. Anda akan menjumpai tampilan sebagai berikut:



14. Setelah tampilan ini, Anda bisa membuat program dalam kelas Main atau Anda bisa membuat kelas sendiri.
15. Jika anda akan membuat kelas dari tampilan ini, maka pilih lokasi yang akan anda gunakan untuk meletakkan kelas Anda, bisa di **Source Package** atau progranmsederhana (dalam contoh ini). Pilih New→Java Class



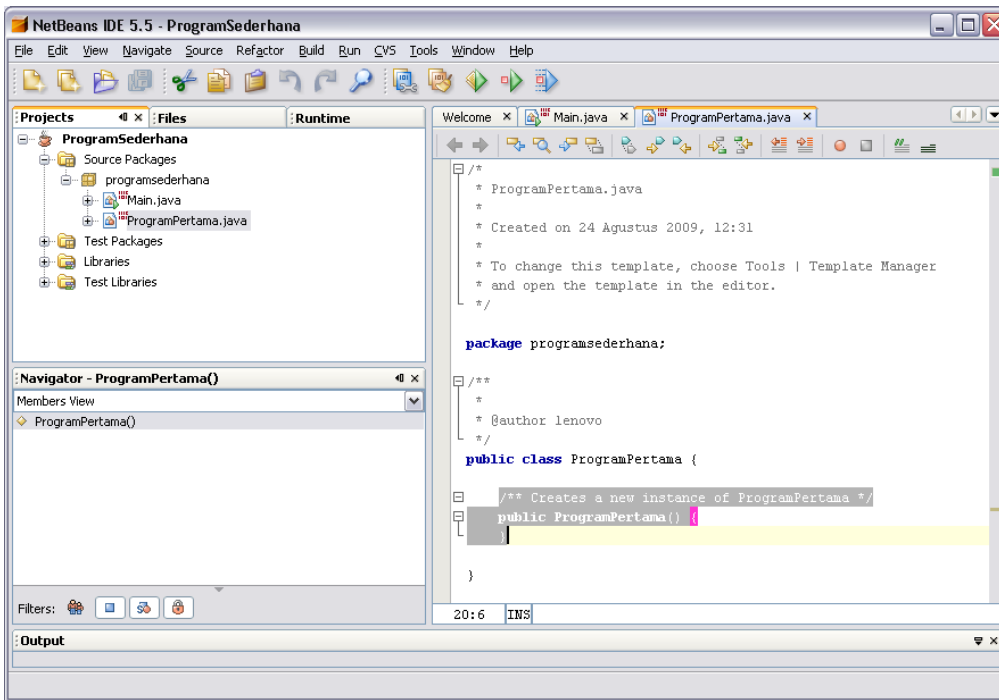
16. Akan muncul tampilan sebagai berikut:



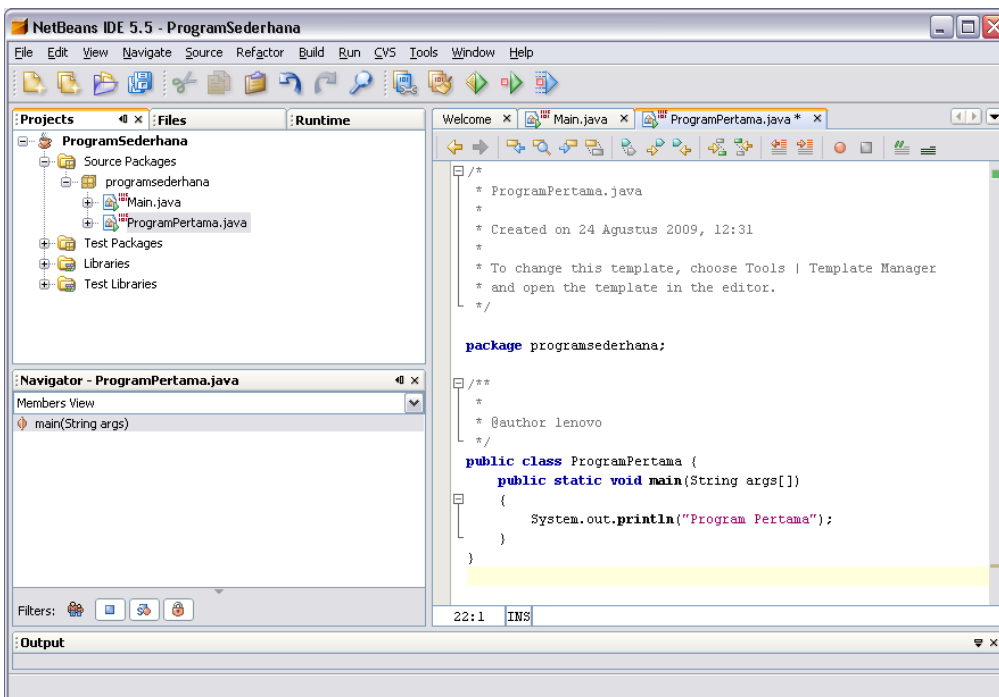
17. Kemudian tuliskan nama kelasnya. Dalam contoh ini diberi nama ProgramPertama

18. Klik Finish

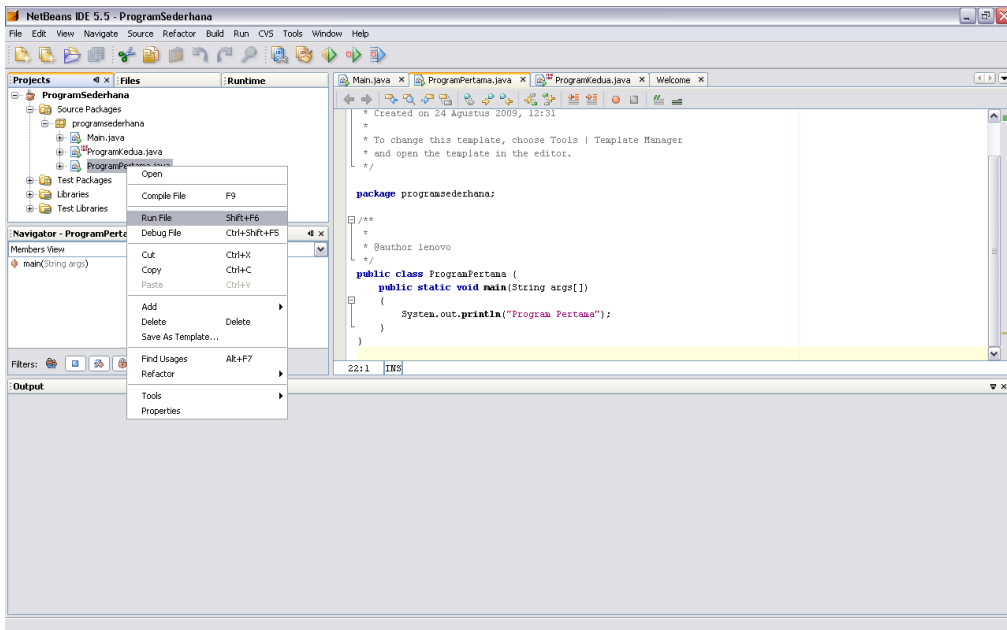
19. Anda akan menjumpai tampilan seperti berikut:



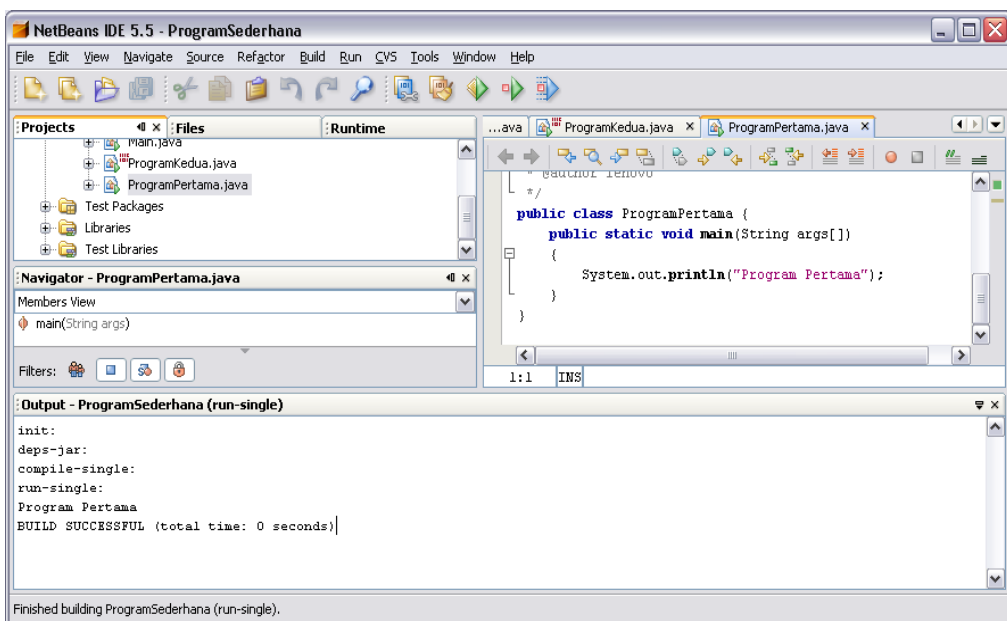
20. Untuk program pertama ini, hapus bagian yang di blok. Ganti menjadi sebagai berikut:



21. Setelah itu jalankan dengan cara klik kanan pada tab **Projects** pada nama kelasnya, dalam hal ini ProgramPertama. Tampilannya adalah sebagai berikut:



22. pilih run Akan terjadi proses kompilasi. Perhatikan bagian tab Output. Jika berhasil, tidak ada kesalahan akan muncul tampilan sebagai berikut :



Dengan demikian anda berhasil membuat dan menjalankan program java dengan menggunakan NetBeans.

1. LATIHAN

- a) Buatlah project baru dengan nama Daspro
- b) Buatlah kelas baru dengan nama Latihan1
- c) Ketikkan program berikut :

```
public class Latihan1
{
    public static void main(String args[])
    {
        // ini adalah komentar
        System.out.println("Halooo semua...");
        System.out.println("Selamat belajar java");
    }
}
```

Hasil Output

Halooo semua...

Selamat belajar java

- d) Ubah bagian program berikut
- ```
System.out.println("Halooo semua...");
System.out.println("Selamat belajar java");
```

### Menjadi

```
System.out.print("Halooo semua...");
System.out.print("Selamat belajar java");
```

- e) Dari hasil output apa yang dapat Anda simpulkan?
- f) Buatlah kelas baru lagi dengan nama Latihan2
- g) Buatlah program untuk menampilkan output seperti berikut :

### Output

Kami sedang belajar java

Jangan diganggu...!



## Output, Proses dan Input

### Tujuan Pembelajaran

Setelah mempelajari bab ini, diharapkan mampu:

- Mengetahui statement output dengan print dan println
- Mengetahui penggunaan variabel
- Mengetahui proses pada program Java
- Mengetahui statement input

### Pokok Bahasan

- Mengetahui statement output dengan print dan println
- Mengetahui penggunaan variabel
- Mengetahui proses pada program Java
- Mengetahui statement input

### I. Program Java Pertama

Tulis program berikut dan beri simpan sebagai file dengan nama **Prak\_101.java**, kemudian kompilasi dan jalankan.

```
01 public class Prak_101 {
02 public static void main (String[] args) {
03 System.out.println("Hello");
04 System.out.println("Selamat membuat program Java");
05 }
06 }
```

### II. Penggunaan Variabel

Secara umum, elemen-elemen dasar pemrograman Java terdiri dari :

#### 1. Himpunan Karakter

Himpunan karakter terdiri dari huruf, digit maupun simbol-simbol lainnya (termasuk spasi, karakter kontrol).

Contoh :

Huruf : A, a, B, b, C, c .....

Digit : 0, 1, 2, 3, 4, 5 .....

Simbol dan lainnya : \_ - + \* dan sebagainya

## 2. Pengenal (identifier)

Pengenal atau identifier adalah suatu nama yang bisa dipakai dalam pemrograman untuk menyatakan :

- variabel
- konstanta bernama
- tipe data
- fungsi
- label
- obyek

Contoh :

```
moMhs;
```

```
no_Mhs;
```

## 3. Kata Kunci

Pengenal sistem yang mempunyai makna khusus bagi kompilernya. Kegunaan dari golongan ini tidak dapat diubah.

Contoh :

```
case, char, const, do, else, for, return, void, while, dan lain-lain
```

## 4. Tipe Data Primitif

Bahasa Pemrograman Java adalah bahasa pemrograman yang selalu menggunakan tipe data untuk setiap variabelnya. Itu berarti bahwa semua variabel harus dideklarasikan terlebih dahulu sebelum mereka digunakan.

Misalnya

```
int nilai = 1;
```

- **byte**: Tipe data byte adalah 8-bit integer bertanda *two's complement*. Tipe ini mempunyai nilai minimum -128 dan nilai maksimumnya adalah 127.
- **short**: Tipe data short merupakan integer 16 bit *two's complement* yang mempunyai nilai minimum -32,768 dan nilai maksimumnya adalah 32,767.
- **int**: Tipe data int adalah integer 32 bit *two's complement*. Dia mempunyai nilai minimum -2,147,483,648 sedangkan nilai maksimumnya adalah 2,147,483,647 (inclusive).
- **long**: Tipe data long adalah integer 64 bit *two's complement*. Nilai minimumnya adalah -9,223,372,036,854,775,808 sedangkan nilai maksimumnya adalah 9,223,372,036,854,775,807

(inclusive). Gunakan tipe data ini pada saat anda memerlukan jangkauan nilai yang lebih besar daripada yang bisa disajikan oleh int.

- **float**: Tipe data float merupakan *single-precision 32-bit IEEE 754 floating point*.
- **double**: Tipe data double adalah *double-precision 64-bit IEEE 754 floating point*.

Tabel 1.1. Batas nilai floating point

| Parameter | float | double |
|-----------|-------|--------|
| N         | 24    | 53     |
| K         | 8     | 11     |
| $E_{max}$ | +127  | +1023  |
| $E_{min}$ | -126  | -1022  |

- **boolean**: Tipe data boolean hanya mempunyai dua nilai yang mungkin, yaitu true dan false. Gunakan tipe data ini untuk flag-flag sederhana untuk menjejak/menelusuri kondisi true atau false. Tipe data ini merepresentasikan satu bit informasi, tetapi ukurannya tidak didefinisikan dengan tepat.
- **char**: Tipe data char adalah karakter Unicode 16 bit. Tipe data ini mempunyai nilai minimum '\u0000' (atau 0) dan nilai maksimum '\uffff' (atau 65,535).

## 5. Variabel dan Konstanta

Variabel digunakan dalam program untuk menyimpam suatu nilai, dan nilai yang ada padanya dapat diubah selama eksekusi berlangsung.

Konstanta adalah nilai yang tetap.

Contoh :

```
float jumlah;
jumlah = 10;
```

## 6. Konstanta bernama

Hal ini dapat dilakukan dengan menggunakan kata kunci const.

Contoh :

```
const float PHI = 3.14;
```

## NILAI DEFAULT

Tidak selalu perlu untuk menentukan suatu nilai ketika sebuah field dideklarasikan. Field yang dideklarasikan tetapi tidak diinisialisasi akan diset ke default yang ada oleh kompiler. Secara umum, default ini akan bernilai *null* atau *zero* tergantung pada tipe datanya. Tabel 1.2 merangkum nilai default untuk tipe-tipe data di atas.

Tabel 1.2. Daftar nilai default untuk tipe data tertentu

| Tipe data           | Nilai Default (untuk field) |
|---------------------|-----------------------------|
| byte                | 0                           |
| short               | 0                           |
| int                 | 0                           |
| long                | 0L                          |
| float               | 0.0f                        |
| double              | 0.0d                        |
| char                | '\u0000'                    |
| String (atau obyek) | Null                        |
| boolean             | False                       |

Program berikut akan mendeklarasikan suatu variabel untuk menampung suatu nilai (bertipe integer), dan menampilkannya ke layar. Tulis, dan jalankan program berikut (**Prak\_102.java**).

```

01 public class Prak_102 {
02 public static void main (String[] args) {
03 int var_a, var_b;
04 var_a = 10;
05 var_b = 500;
06 System.out.println("Variabel yang terdapat dalam program :");
07 System.out.println("var_a = " + var_a);
08 System.out.println("var_b = " + var_b);
09 }
10 }

```

### III. Program yang terdapat proses (perhitungan matematika)

Program berikut adalah program untuk menjumlahkan 2 buah bilangan. Tulis dan jalankan programnya. (**Prak\_103.java**)

```

01 public class Prak_103 {
02 public static void main (String[] args) {
03 int bil_1, bil_2, hasil;
04 bil_1 = 10;
05 bil_2 = 500;
06 hasil = bil_1 + bil_2;
07 System.out.print("Hasil Penjumlahan kedua bilangan :");
08 System.out.println(hasil);
09 }
10 }

```

Cobalah ubah nilai dari var bil\_1 menjadi 10.5 dan bil\_2 menjadi 500.5, apa yang terjadi ketika program dikompilasi. Untuk menyimpan bilangan pecahan gunakan tipe data **float** atau **double**.

Program berikut adalah program untuk menghitung luas suatu persegi panjang. Tulis dan jalankan programnya. (**Prak\_104.java**)

```
01 public class Prak_104 {
02 public static void main (String[] args) {
03 float panjang = 12.5f;
04 float lebar = 5.5f;
05 float luas = panjang * lebar;
06 System.out.println("Luas persegi panjang tersebut =" + luas);
07 }
08 }
```

#### IV. Menginputkan data dari keyboard

Program berikut akan menerima masukan dari keyboard, dan menyimpannya ke dalam suatu variabel, dan kemudian menampilkannya ke layar. Untuk menggunakan masukan keyboard, digunakan *library* Scanner, dari paket java.util.

##### 1. Komponen masukan

Untuk memasukkan sebuah nilai ke variabel yang sudah didefinisikan digunakan kelas Scanner.

##### a. Input data bertipe Integer

Untuk menginputkan data dengan tipe integer digunakan method `nextInt` di dalam kelas Scanner.

##### b. Input data bertipe String

Untuk menginputkan data dengan tipe integer digunakan method `nextInt` di dalam kelas Scanner.

##### c. Input tipe yang lain

Untuk input data dari keyboard dengan tipe yang lain dan disediakan oleh kelas Scanner adalah sebagai berikut

- `nextBoolean` : input tipe boolean (true atau false saja)
- `nextShort` : input tipe short integer
- `nextLong` : input tipe long integer
- `nextFloat` : input tipe float
- `nextDouble` : input tipe double



Tulis dan jalankan program berikut. (**Prak\_105.java**)

```
01 import java.util.Scanner;
02 public class Prak_105 {
03 public static void main (String[] args) {
04 Scanner masukan = new Scanner(System.in);
05 int var_a, var_b;
06 System.out.print("Masukkan nilai var var_a :");
07 var_a = masukan.nextInt();
08 System.out.print("Masukkan nilai var var_b :");
09 var_b = masukan.nextInt();
10 System.out.println();
11 System.out.println("Variabel yang terdapat dalam program :");
12 System.out.println("var_a = " + var_a);
13 System.out.println("var_b = " + var_b);
14 }
15 }
```

#### V. Program Konversi Suhu

Program berikut adalah program untuk mengkonversi suhu Celcius yang diinputkan oleh pengguna ke dalam suhu Reamur yang dihasilkan oleh program. Tulis dan jalankan program berikut. (**Prak\_106.java**)

```
01 import java.util.Scanner;
02 public class Prak_106 {
03 public static void main (String[] args) {
04 Scanner masukan = new Scanner(System.in);
05 float celcius, reamur;
06 System.out.print("Masukkan nilai suhu (celcius) : ");
07 celcius = masukan.nextFloat();
08 reamur = 0.8f * celcius;
09 System.out.println();
10 System.out.print("nilai suhu reamur dari input : ");
11 System.out.println(reamur);
12 }
13 }
```

## **VI. Soal Latihan**

### **Soal 1.1 (Soal\_101.java)**

Buat program yang meminta inputan jari-jari, yang kemudian dihitung dan ditampilkan luas dan keliling lingkaran tersebut. (mis : input jari-jari = 5.5)

### **Soal 1.2 (Soal\_102.java)**

Buat program yang meminta inputan suatu nilai rupiah, yang kemudian dihitung nilai US Dollar dan Euro dari nilai rupiah tersebut. (mis : input rupiah = 1000000.00)



## Kondisi

### Tujuan Pembelajaran

Setelah mempelajari bab ini, diharapkan mampu:

- If
- If...Else...
- If...Else...bertingkat
- Switch...case

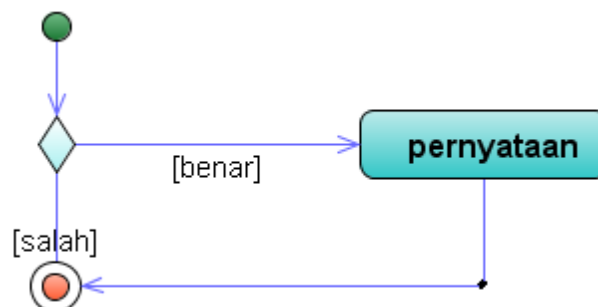
### Pokok Bahasan

- If
- If...Else...
- If...Else...bertingkat
- Switch...case

#### I. Program dengan pernyataan If

Pernyataan seleksi dengan IF akan mempunyai beberapa bentuk. Bentuk yang pertama adalah IF dengan satu pilihan. Bentuk umumnya adalah sebagai berikut.

```
if (kondisi)
 pernyataan;
```



Gambar Pernyataan IF

Keterangan :

- Kondisi digunakan untuk menentukan pengambilan keputusan. Jika kondisi bernilai benar, maka pernyataan dikerjakan
- Pernyataan, berisi perintah-perintah dan akan dijalankan jika kondisi bernilai benar. Pernyataan disini bisa berupa pernyataan tunggal maupun majemuk.

Program di bawah adalah program untuk menentukan suatu kelulusan mahasiswa terhadap suatu mata kuliah. Program tersebut menggunakan pernyataan If untuk penentuan keputusannya. Tulis dan jalankan program berikut (**Prak\_201.java**).

```
01 import java.util.Scanner;
02 public class Prak_201 {
03 public static void main (String[] args) {
04 Scanner masukan = new Scanner(System.in);
05 int nilai;
06 System.out.print("Masukkan nilai akhir mata kuliahnya :");
07 nilai = masukan.nextInt();
08 if (nilai < 55)
09 System.out.println("Mahasiwatersebut tidak lulus");
10 }
11 }
```

Jika diinputkan nilai < 55, maka akan muncul keterangan mahasiswa tersebut tidak lulus, tetapi jika nilai > atau = 55, maka program akan berhenti tanpa menampilkan suatu pesan.

## II. Program dengan pernyataan If..Else..

Sempurnakan program di atas dengan menambahkan pernyataan sebagaimana berikut. (**Prak\_202.java**).

```
else
 System.out.println("Mahasiwatersebut lulus");
```

## III. Program dengan pernyataan If..Else.. bertingkat

Program di bawah adalah program untuk mengkonversi nilai angka ke nilai huruf, yang dilakukan dengan menggunakan pernyataan if..else.. bertingkat. Tulis dan jalankan program tersebut (**Prak\_203.java**).

```
01 import java.util.Scanner;
02 public class Prak_203 {
03 public static void main (String[] args) {
04 Scanner masukan = new Scanner(System.in);
05 System.out.print("Masukkan nilai mata kuliahnya :");
06 int nilai = masukan.nextInt();
07 if (nilai >= 80)
08 System.out.println("NilainyaA");
09 else if (nilai >= 70)
10 System.out.println("NilainyaB");
11 else if (nilai >= 55)
12 System.out.println("NilainyaC");
13 else if (nilai >= 40)
14 System.out.println("NilainyaD");
15 else
16 System.out.println("Nilainya E");
17 }
18 }
```

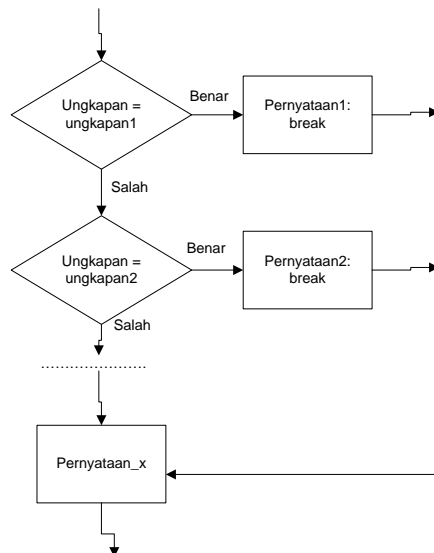
#### IV. Program dengan pernyataan Switch..Case..

Pernyataan switch adalah pernyataan yang digunakan untuk menjalankann salah satu pernyataan dari beberapa kemungkinan pernyataan, berdasarkan nilai dari sebuah ungkapan dan nilai penyeleksi. Setiap ungkapan diungkapkan dengan sebuah nilai integral konstan, seperti sebuah nilai dengan tipe byte, short, int atau char.

Bentuknya :

```
switch (ungkapan)
{
 case ungkapan1:
 pernyataan1;
 break;
 case ungkapan2:
 pernyataan2;
 break;

 default:
 pernyataan_x;
}
```



Keterangan :

- ungkapan1, ungkapan2 dan seterusnya dilakukan secara berurutan dimulai dari yang pertama, sekiranya cocok pernyataan yang mengikuti **case** dijalankan.
- **break** ditemukan dari eksekusi pernyataan **switch** berakhir
- **default** hanya akan dijalankan jika ungkapan pada bagian case tidak ada yang cocok.

Program berikut akan menampilkan tulisan sesuai dengan input yang diberikan, dengan batasan input 1 sampai 3. Tulis dan jalankan program berikut (**Prak\_204.java**).

```

01 import java.util.Scanner;
02 public class Prak_204 {
03 public static void main (String[] args) {
04 Scanner masukan = new Scanner(System.in);
05 System.out.print("Masukkan angka 1 - 3 : ");
06 int bil = masukan.nextInt();
07 switch (bil) {
08 case 1 : System.out.println("Satu");break;
09 case 2 : System.out.println("Dua");break;
10 case 3 : System.out.println("Tiga");}
11 }
12 }

```

Coba tambahkan pernyataan di bawah, yang diletakkan setelah pernyataan case 3, kemudian inputkan bilangan lebih besar dari 4. (**Prak\_205.java**)

```
default : System.out.println("Bilangandi luar range");
```

## V. Program dengan kondisi majemuk

Program berikut adalah program untuk menyeleksi mahasiswa baru lewat jalur SPMB, dimana mahasiswa dinyatakan diterima bila nilai matematikanya di atas 80 dan nilai fisiknya di atas 70. Tulis dan jalankan program di bawah ini (**Prak\_206.java**).

```
01 import java.util.Scanner;
02 public class Prak_206 {
03 public static void main (String[] args) {
04 Scanner masukan = new Scanner(System.in);
05 System.out.print("Masukkan nilai Matematika-nya :");
06 int matematika = masukan.nextInt();
07 System.out.print("Masukkan nilai Fisika-nya :");
08 int fisika = masukan.nextInt();
09 if ((matematika > 80) && (fisika > 70))
10 System.out.println("Siswatersebut DITERIMA");
11 else
12 System.out.println("Siswatersebut TIDAK DITERIMA");
13 }
14 }
```

## VI. Soal latihan

### Soal 2.1 (Soal\_201.java)

Buat program untuk menentukan kriteria kegemukan dengan menggunakan Indeks Massa Tubuh (IMT), yang dihitung berdasarkan rumus :

$$\text{IMT} = b / t^2 \quad \text{dengan } b : \text{berat badan (kg) dan } t : \text{tinggi badan (m)}$$

Kriteria penentuan berat berdasarkan IMT adalah sebagai berikut :

| Nilai IMT                   | Kriteria             |
|-----------------------------|----------------------|
| $\text{IMT} \leq 18.5$      | Kurus                |
| $18.5 < \text{IMT} \leq 25$ | Normal               |
| $25 < \text{IMT} \leq 30$   | Gemuk                |
| $\text{IMT} > 30$           | Kegemukan (Obesitas) |

### Soal 2.2 (Soal\_202.java)

Buat program untuk menentukan gaji seorang karyawan berdasarkan waktu/jam kerjanya, dengan aturan bahwa batasan jam kerja normal adalah 50 sampai 60 jam/minggu dengan upah Rp.5000/jam. Jika lebih dari batas tersebut, maka kelebihan perjamnya dibayar Rp 7500/jam. Tetapi jika jam kerjanya kurang dari 50 jam/minggu, maka dikenakan potongan Rp.2500/jam nya. (Input : jam kerja/minggu, dan output : total gaji)

**Soal 2.3 (Soal\_203.java)**

Buat program untuk menentukan kuadran dari suatu titik koordinat. Nilai inputnya adalah nilai X dan Y suatu titik, dan outputnya adalah nomor kuadran dari koordinat tersebut.





## Perulangan

### Tujuan Pembelajaran

Setelah mempelajari bab ini, diharapkan mampu:

- for
- break
- counter
- while
- do...while

### Pokok Bahasan

- for
- break
- counter
- while

#### I. Program dengan pernyataan for

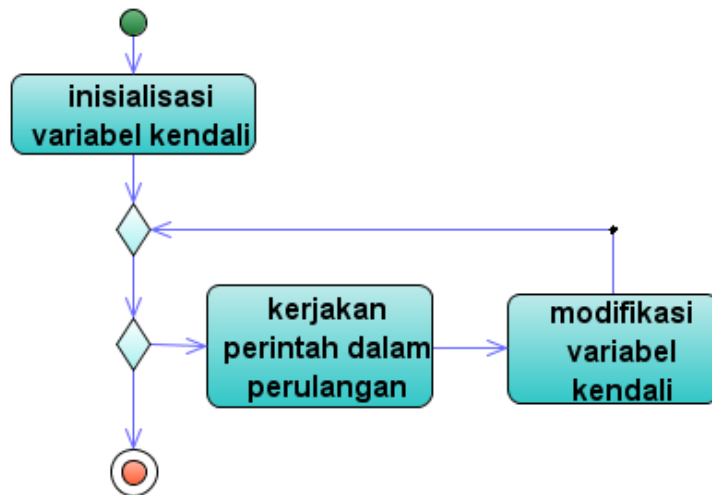
pernyataan for juga digunakan untuk mengerjakan pernyataan atau sekelompok pernyataan secara berulang.

Bentuknya :

```
for (ungkapan1;ungkapan2;ungkapan3)
 Pernyataan;
```

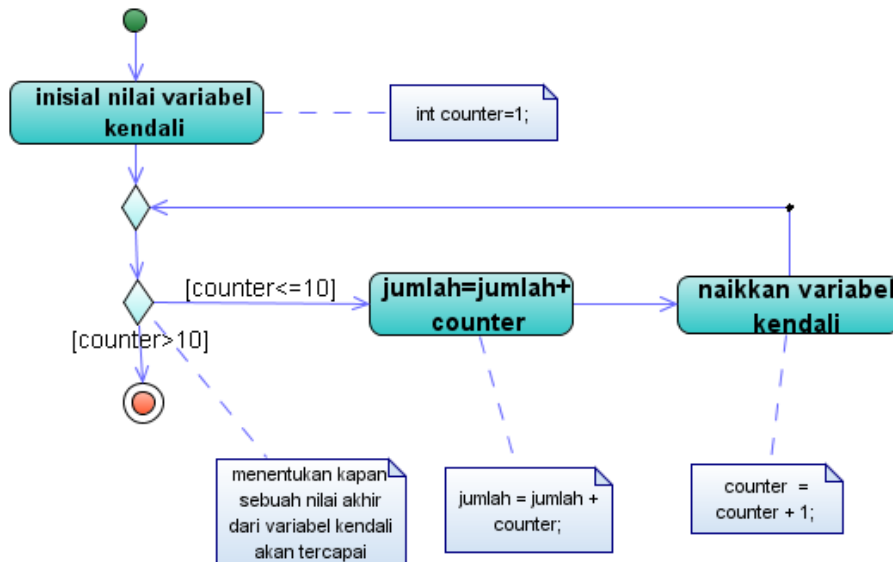
Keterangan :

- ungkapan1 merupakan pernyataan inisialisasi
- ungkapan2 sebagai kondisi yang menentukan pengulangan terhadap pernyataan atau tidak
- ungkapan3 digunakan sebagai pengatur variabel yang digunakan didalam ungkapan1



Gambar : Perulangan dengan FOR

Contoh perulangan dengan for.



Gambar : Contoh perulangan dengan FOR

Program di bawah ini akan menampilkan tulisan "Hello World!!" sebanyak 5 kali. Tulis dan jalankan program tersebut (**Prak\_301.java**).

```

01 public class Prak_301 {
02 public static void main (String[] args) {
03 for(int i=1; i<5; i++)
04 System.out.println("Hello World!!");
05 }
06 }

```

Program di bawah ini akan menampilkan bilangan genap mulai dari 2 sampai bilangan yang diinputkan oleh pengguna. Tulis dan jalankan program berikut. (**Prak\_302.java**)

```
01 import java.util.Scanner;
02 public class Prak_302 {
03 public static void main (String[] args) {
04 Scanner masukan = new Scanner(System.in);
05 System.out.println("Masukkan batas bilangannya : ");
06 int batas = masukan.nextInt();
07 System.out.println("Bilangan genap dari 2 sampai bil tsb : ");
08 for(int i=2; i<=batas; i=i+2)
09 System.out.print(i+ " ");
10 }
11 }
```

## II. Program dengan pernyataan break

Pernyataan break adalah pernyataan untuk menghentikan perulangan, sehingga akan keluar dari perulangan tersebut walaupun proses perulangan belum berakhir. Berikut ini adalah program untuk menentukan apakah suatu bilangan itu termasuk bilangan prima atau tidak. Tulis dan jalankan program dibawah ini (**Prak\_303.java**).

```
01 import java.util.Scanner;
02 public class Prak_303 {
03 public static void main (String[] args) {
04 Scanner masukan = new Scanner(System.in);
05 System.out.println("Masukkan sebuah bilangan : ");
06 int bil = masukan.nextInt();
07 boolean prima = true;
08 for(int i=2; i<bil; i++){
09 if ((bil % i) == 0) {
10 prima = false; break;}
11 }
12 if (prima)
13 System.out.print(bil + " adalah bilangan PRIMA");
14 else
15 System.out.print(bil + " adalah BUKAN bilangan PRIMA");
16 }
17 }
```

### III. Program dengan variabel counter

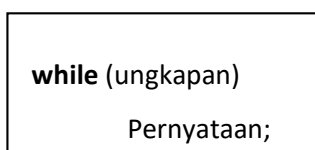
Variabel counter adalah suatu variabel yang menyimpan hasil operasi secara kontinyu (mis : hasil = hasil + 5 atau ditulis hasil += 5). Berikut adalah program untuk menjumlahkan bilangan sampai dengan bilangan yang diinputkan oleh pengguna. Tulis dan jalankan program tersebut (**Prak\_304.java**).

```
01 import java.util.Scanner;
02 public class Prak_304 {
03 public static void main (String[] args) {
04 Scanner masukan = new Scanner(System.in);
05 System.out.println("Masukkan batas bilangannya : ");
06 int batas = masukan.nextInt();
07 int hasil = 0;
08 for(int i=1; i<=batas; i++)
09 hasil += i;
10 System.out.println("Total jumlahnya adalah : " + hasil);
11 }
```

### IV. Program dengan pernyataan while

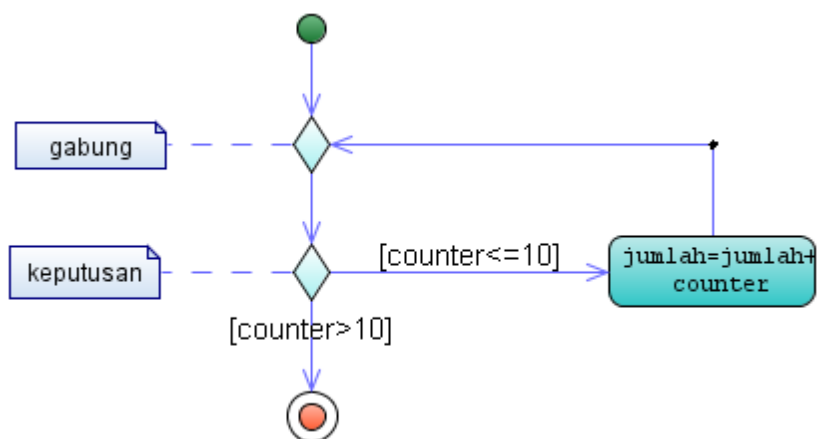
Pernyataan ini berguna untuk memproses suatu pernyataan atau beberapa pernyataan beberapa kali. Selama ungkapan bernilai benar, pernyataan akan selalu dikerjakan.

Bentuknya :



Keterangan :

- bagian pernyataan akan dieksekusi selama ungkapan dalam **while** bernilai benar.
- Pengujian terhadap ungkapan pada **while** dilakukan sebelum bagian pernyataan.
- Kemungkinan pernyataan pada **while** tidak dijalankan sama sekali, jika ketemu kondisi yang pertama kali bernilai salah.



Catatan :

Pernyataan perulangan dengan while akan selalu dikerjakan jika ungkapan selalu benar. Oleh karena itu, kita harus membuat kondisi suatu saat ungkapan bernilai salah agar perulangan berakhir.

Pernyataan perulangan dengan while, umumnya digunakan untuk perulangan yang belum dapat dipastikan jumlah perulangannya. Berikut contoh program untuk menjumlahkan suatu bilangan sampai input yang dimasukkan adalah bilangan 0. Tulis dan jalankan program berikut (**Prak\_305.java**).

```
01 import java.util.Scanner;
02 public class Prak_305 {
03 public static void main (String[] args) {
04 Scanner masukan = new Scanner(System.in);
05 int n=0, total=0, bil=9;
06 while (bil!=0){
07 n++;
08 System.out.println("Masukkan bilangan ke- "+n+" : ");
09 bil = masukan.nextInt();
10 total += bil;
11 }
12 System.out.print("Total jumlah "+(n-1)+" bilangan tsb : ");
13 System.out.println(total);
14 }
15 }
```

## V. Program dengan pernyataan do..while

Seperti halnya perulangan dengan while, perulangan dengan do ... while ini juga digunakan untuk mengerjakan sebuah atau sekelompok pernyataan berulang-ulang. Bedanya dengan while adalah pernyataan do ... while akan mengecek kondisi di belakang, sementara while cek kondisi ada di depan.

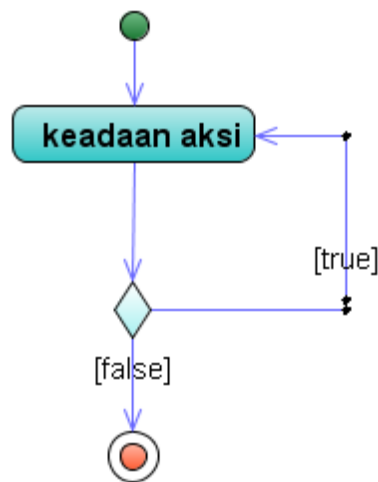
Bentuknya :

```
do
{
 pernyataan1;
 pernyataan2;

 pernyataan_N;
}
while (ungkapan)
```

Keterangan :

- Bagian pernyataan1 hingga pernyataanN dijalankan secara berulang sampai ungkapan bernilai salah.
- Pengujian ungkapan dilakukan setelah bagian pernyataan, maka pada pernyataan **do ... while** minimal akan dijalankan sekali, karena begitu masuk ke blok perulangan, tidak ada cek kondisi tetapi langsung mengerjakan pernyataan.



Buatlah program di atas dengan menggunakan pernyataan do..while(**Prak\_306.java**).

## VI. Soal latihan

### Soal 3.1 (**Soal\_301.java**)

Buat program untuk menghitung angsuran hutang, dimana diinputkan besar hutang/pinjaman, lamanya angsuran (bulan), dan besarnya bunga perbulan. Bunga dihitung dari hutang/pinjaman yang tersisa. Tampilan daftar pembayaran mulai dari pembayaran pertama sampai terakhir (lunas).

### Soal 3.2 (**Soal\_302.java**)

Buat program untuk mengkonversi bilangan bulat ke bilangan romawi dengan input bilangan bulat dengan range 1- 5000.

### Soal 3.3 (**Soal\_303.java**)

Buat program untuk mencetak bilangan prima mulai dari bilangan prima pertama (2) sampai dengan bilangan prima terakhir, yang banyaknya bilangan primanya ditentukan oleh pengguna.



## Modul / Fungsi

### Tujuan Pembelajaran

Setelah mempelajari bab ini, diharapkan mampu:

- Fungsi
- Fungsi yang mengembalikan suatu nilai
- Fungsi rekursif

### Pokok Bahasan

- Fungsi
- Fungsi yang mengembalikan suatu nilai
- Fungsi rekursif

#### I. Membuat fungsi pada Program Java

Method (atau dalam beberapa bahasa pemrograman sering disebut fungsi atau prosedur) adalah sub program yang membiarkan seorang programmer untuk membagi program dengan membagi masalah ke dalam beberapa sub masalah yang bisa diselesaikan secara modular. Dengan cara demikian, maka pembuatan program bisa lebih dimanajemen.

Contoh :

1. `public class FungsiParameter`
2. `{`
3. `public static int jumlah(int a){`
4. `return a;`
5. `}`
6. `public static void main(String args[]){`
7. `System.out.println("Hasil pemanggilan metode ");`
8. `System.out.println(jumlah(5));`
9. `}`
10. `}`

## Hasil Output

Hasil pemanggilan method jumlah

5

Press any key to continue . . .

Parameter pada baris kedua disebut sebagai parameter formal, dan pada baris ke 8 disebut parameter aktual.

Ada 2 buah parameter yaitu

- parameter formal adalah parameter yang tertulis dalam definisi method
- Parameter aktual parameter yang berada pada inputan langsung pada saat penggunaan method tersebut.

Parameter bisa lebih dari satu dengan dipisahkan tanda koma,. Yang perlu diperhatikan pada saat pemanggilan method adalah jumlah, urutan dan tipe parameter aktual harus sesuai dengan jumlah urutan dan tipe parameter formal.

Pemberian Variabel Dalam Method

Ada dua tipe data variable passing pada method, yaitu *pass-by-value* dan *pass-by-reference*.

### ***Pass-by-value***

Ketika *pass-by-value* terjadi, method membuat sebuah salinan dari nilai variable yang dikirimkan ke method. Walaupun demikian method tidak dapat secara langsung memodifikasi nilai variable pengirimnya meskipun parameter salinannya sudah dimodifikasi nilainya di dalam method.

### ***Pass-by-reference***

Ketika sebuah *pass-by-reference* terjadi, alamat memori dari nilai pada sebuah variable dilewatkan pada saat pemanggilan method. Ini tidak seperti pada *pass-by-value*, method dapat memodifikasi variable asli dengan menggunakan alamat memori tersebut, meskipun berbeda nama variable yang digunakan dalam method dengan variable aslinya, kedua variable ini menunjukkan lokasi dari data yang sama.



Tulis dan jalankan program berikut. (Prak\_401.java)

```
01 import java.util.Scanner;
02 public class Prak_401 {
03 public static void main (String[] args) {
04 Scanner masukan = new Scanner(System.in);
05 int pilihan=9;
06 do {
07 System.out.println("\nMENU\n");
08 System.out.println("1. Menghitung Luas Persegi Panjang");
09 System.out.println("2. Menghitung Luas Lingkaran");
10 System.out.println("0. Keluar");
11 System.out.print("Masukkan Pilihan Anda : ");
12 pilihan = masukan.nextInt();
13 switch(pilihan){
14 case 1 : luasPersegiPanjang();break;
15 case 2 : luasLingkaran();break;
16 }
17 } while(pilihan != 0);
18 }
19
20 private static void luasPersegiPanjang(){
21 Scanner masukan = new Scanner(System.in);
22 float panjang, lebar, luas;
23 System.out.print("Masukkan nilai panjang : ");
24 panjang = masukan.nextFloat();
25 System.out.print("Masukkan nilai lebar : ");
26 lebar = masukan.nextFloat();
27 luas = panjang * lebar;
28 System.out.println("Luas Persegi Panjang : " + luas);
29 }
30
31 private static void luasLingkaran(){
32 Scanner masukan = new Scanner(System.in);
33 float jari2, luas;
34 System.out.print("Masukkan nilai jari-jari : ");
35 jari2 = masukan.nextFloat();
36 luas = 3.14f * jari2 * jari2;
37 System.out.println("Luas Lingkaran : " + luas);
38 }
39 }
```

## II. Membuat fungsi yang mengembalikan suatu nilai

Fungsi yang mengembalikan suatu nilai adalah fungsi yang ketika kembali ke program utamanya disertai dengan membawa suatu nilai. Berikut adalah program untuk menjumlahkan 2 buah bilangan yang di dalamnya menggunakan fungsi. Tulis dan jalankan program berikut (**Prak\_402.java**).

```
01 import java.util.Scanner;
02 public class Prak_402 {
03 public static void main (String[] args) {
04 Scanner masukan = new Scanner(System.in);
05 int bil1, bil2, bil3;
06 System.out.println("Program Menjumlahkan 2 bilangan");
07 System.out.print("Masukkan bilangan 1 : ");
08 bil1 = masukan.nextInt();
09 System.out.print("Masukkan bilangan 2 : ");
10 bil2 = masukan.nextInt();
11 System.out.print("Masukkan bilangan 3 : ");
12 bil3 = masukan.nextInt();
13 System.out.println();
14 System.out.println("Jumlah bil1 + bil2 : "+(jumlah(bil1,bil2)));
15 System.out.println("Jumlah bil2 + bil3 : "+(jumlah(bil2,bil3)));
16 System.out.println("Jumlah bil1 + bil3 : "+(jumlah(bil1,bil3)));
17 }
18
19 private static int jumlah(int a, int b){
20 int hasil = a + b;
21 return hasil;
22 }
23 }
```

Ubahlah program di atas sehingga menjadi program untuk mencari nilai terbesar dan terkecil dari 3 buah bilangan. (**Prak\_403.java**)

### III. Membuat program dengan fungsi rekursif

Suatu fungsi rekursif adalah suatu fungsi yang memanggil dirinya sendiri. Berikut ini adalah contoh program rekursif untuk menghitung nilai faktorial suatu bilangan. Tulis dan jalankan program berikut.

(Prak\_404.java).

```
01 import java.util.Scanner;
02 public class Prak_404 {
03 public static void main (String[] args) {
04 Scanner masukan = new Scanner(System.in);
05 int bil, hasil;
06 System.out.print("Masukkansuatu bilangan : ");
07 bil = masukan.nextInt();
08 hasil = faktorial(bil);
09 System.out.println("Nilai faktorial "+ bil +" adalah "+ hasil);
10 }
11
12 private static int faktorial(int a){
13 if (a==1)
14 return 1;
15 Else
16 return (a * faktorial(a-1));
17 }
18 }
```

#### IV. Soal latihan

##### Soal 4.1 (Soal\_401.java)

Buat program yang memiliki suatu fungsi untuk menentukan apakah suatu bilangan itu termasuk bilangan prima atau bukan. Kemudian tampilkan bilangan prima dari 1 sampai 100.

##### Soal 4.2 (Soal\_402.java)

Buat program yang memiliki suatu fungsi rekursif untuk mencetak suatu deret fibbonacci. Deret fibbonacci adalah suatu deret dimana bilangan berikutnya merupakan penjumlahan 2 bilangan sebelumnya. (mis : 1, 1, 2, 3, 5, 8, 13, 21, 33,...)

##### Soal 4.3 (Soal\_403.java)

Buat program untuk mencetak bilangan amisabel dari 1 sampai 1000. Bilangan amisabel adalah bilangan yang jumlah faktor-faktornya sama dengan bilangan itu sendiri.

Contoh :

$$1 = 1$$

$$6 = 1 + 2 + 3$$

$$28 = 1 + 2 + 4 + 7 + 14$$



## Array / Larik

### Tujuan Pembelajaran

Setelah mempelajari bab ini, diharapkan mampu:

- Array
- Array 2 dimensi

### Pokok Bahasan

- Array
- Array 2 dimensi

#### I. Program dengan Array 1 dimensi

Larik adalah sebuah struktur data yang terdiri dari data yang bertipe sama. Ukuran larik bersifat tetap, larik akan mempunyai ukuran yang sama pada saat sekali dibuat. Larik dalam Java adalah obyek, disebut juga sebagai tipe referensi. Sedangkan elemen dalam larik Java bisa primitif atau referensi. Posisi dari larik biasa disebut sebagai elemen. Elemen larik dimulai dari 0 (nol). Penyebutan larik diberikan dengan cara menyebutkan nama lariknya dan diikuti dengan indeksinya, dimana indeks dituliskan diantara tanda kurung siku. Gambar 1. memperlihatkan gambaran larik dengan 10 elemen, dimana setiap elemennya bertipe integer, dengan nama A.

|           |      |      |      |      |      |      |      |      |      |      |
|-----------|------|------|------|------|------|------|------|------|------|------|
| Nama      | A[0] | A[1] | A[2] | A[3] | A[4] | A[5] | A[6] | A[7] | A[8] | A[9] |
| Isi larik | 12   | -56  | 23   | 45   | -16  | -2   | 85   | 41   | 15   | 20   |

#### Deklarasi dan Menciptakan Larik

Sebagai sebuah obyek, larik harus diciptakan dengan menggunakan kata cadang new. Deklarasi dan penciptaan variabel larik gambar 1 adalah sebagai berikut.

```
int A[] = new int[10];
```

larik dideklarasikan dan langsung diciptakan . Atau

```
int A[];
```

```
A = new int[10];
```

larik dideklarasikan, baru pada pernyataan berikutnya larik diciptakan.

Program berikut adalah program untuk mencari nilai rata-rata sekelompok bilangan, dimana bilangan yang akan dirata-rata dimasukkan terlebih dahulu ke dalam array. Tulis dan jalankan program berikut (**Prak\_501.java**).

```
01 import java.util.Scanner;
02 public class Prak_501 {
03 public static void main (String[] args) {
04 Scanner masukan = new Scanner(System.in);
05 int jumbil, total=0;
06 float rata2;
07 System.out.print("Berapa banyaknya bilangan ? ");
08 jumbil = masukan.nextInt();
09 int data[] = new int[jumbil];
10 for(int i=0;i<jumbil;i++){
11 System.out.print("Masukkan data ke-"+(i+1)+" : ");
12 data[i] = masukan.nextInt();
13 total = total + data[i];
14 }
15 System.out.print("Nilai rata-rata dari : ");
16 for(int i=0;i<(jumbil-1);i++)
17 System.out.print(data[i] + " + ");
18 System.out.print(data[jumbil-1]+ " adalah ");
19 rata2 = total/jumbil;
20 System.out.println(rata2);
21 }
22 }
```

## Program dengan Array 2 dimensi

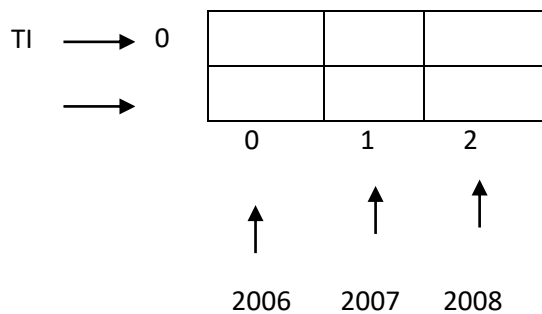
Kita juga bisa membuat variabel larik yang tipe elemennya adalah larik. Dengan cara demikian, kita membuat larik dua dimensi. Dengan larik dua dimensi, maka kita mempunyai elemen yang berindeks tidak hanya satu, tetapi dua. Kita bisa membayangkan larik dua dimensi tersebut seperti sebuah tabel yang berisi baris dan kolom. Penyebutan sel tabel selalu diikuti dengan penyebutan baris berapa dan kolom berapa.

Contoh :

Diberikan data kelulusan mahasiswa sebuah perguruan tinggi sebagai berikut.

| Jurusan            | 2006 | 2007 | 2008 |
|--------------------|------|------|------|
| Teknik Informatika | 110  | 125  | 135  |
| Sistem Informasi   | 56   | 75   | 80   |

```
int data_lulus [2] [3]
```



|         |         |         |
|---------|---------|---------|
| [0] [0] | [0] [1] | [0] [2] |
| [1] [0] | [1] [1] | [1] [2] |

Program berikut adalah program untuk menjumlahkan dua buah matriks. Untuk membuat suatu matriks, maka digunakan array 2 dimensi. Tulis dan jalankan program berikut (**Prak\_502.java**).

```
01 import java.util.Scanner;
02 public class Prak_502 {
03 public static void main (String[] args) {
04 int baris, kolom;
05 Scanner masukan = new Scanner(System.in);
06 System.out.println("Program Penjumlahan 2 buah matriks");
07 System.out.print("Masukkan jumlah baris Matriks : ");
08 baris = masukan.nextInt();
09 System.out.print("Masukkan jumlah kolom Matriks : ");
10 kolom = masukan.nextInt();
11 System.out.println();
12 System.out.println("Masukkan data : ");
13 System.out.println("Matriks 1 : ");
14 int matriks1[][] = new int[baris][kolom];
15 for(int i=0;i<baris;i++){
16 for(int j=0;j<kolom;j++){
17 System.out.print("Bil baris ke-"+i+" kolom ke-"+j+" : ");
18 matriks1[i][j] = masukan.nextInt();
19 }
20 }
21 System.out.println();
22 System.out.println("Matriks 2 : ");
23 int matriks2[][] = new int[baris][kolom];
24 for(int i=0;i<baris;i++){
25 for(int j=0;j<kolom;j++){
26 System.out.print("Bil baris ke-"+i+" kolom ke-"+j+" : ");
27 matriks2[i][j] = masukan.nextInt();
28 }
29 }
30 System.out.println("Data sebelum dijumlahkan : ");
31 System.out.println("Matriks 1 : ");
32 for(int i=0;i<baris;i++){
33 for(int j=0;j<kolom;j++)
34 System.out.print(matriks1[i][j]+ " ");
```



```

35 System.out.println();
36 }
37 System.out.println();
38 System.out.println("Matriks2 : ");
39 for(int i=0;i<baris;i++){
40 for(int j=0;j<kolom;j++){
41 System.out.print(matriks2[i][j]+ " ");
42 System.out.println();
43 }
44 System.out.println();
45 int hasil[][] = new int[baris][kolom];
46 for(int i=0;i<baris;i++){
47 for(int j=0;j<kolom;j++){
48 hasil[i][j] = matriks1[i][j]+matriks2[i][j];
49 }
50 System.out.println("Hasil penjumlahan Matriks tersebut : ");
51 for(int i=0;i<baris;i++){
52 for(int j=0;j<kolom;j++){
53 System.out.print(hasil[i][j]+ " ");
54 System.out.println();
55 }
56 System.out.println();
57 }
58 }

```

## II. Soal latihan

### Soal 5.1 (Soal\_501.java)

Buat sebuah program yang memiliki menu sebagai berikut :

Menu Utama :

1. Input Data
2. Tampilkan Data
3. Distribusi Frekuensi
4. Statistik
5. Keluar

Dimana input data adalah memasukkan data nilai (antara 0 sampai 100) ke dalam suatu array. Menampilkan data adalah menampilkan semua isi array. Distribusi frekuensi adalah mencari nilai frekuensi untuk masing-masing range nilai (dengan interval 20) . Statistik adalah mencari nilai mean, minimum dan maksimum dari data.



## Operasi File

### Tujuan Pembelajaran

Setelah mempelajari bab ini, diharapkan mampu:

- Operasi pembacaan file teks
- Operasi penulisan file teks

### Pokok Bahasan

- Operasi pembacaan file teks
- Operasi penulisan file teks

#### I. Program untuk membaca file teks

Program di bawah ini adalah program untuk membaca isi suatu file teks dan menampilkan ke layar dengan menambahkan nomor baris. Tulis dan jalankan program berikut (**Prak\_601.java**).

```
01 import java.io.*;
02 public class Prak_601 {
03 public static void main (String[] args) throws Exception {
04 File file = new File("Prak_601.java");
05 if (!file.exists() || !file.canRead()){
06 System.out.println("Can't read " + file);
07 return;
08 }
09 try {
10 int i=0;
11 BufferedReader fln = new BufferedReader(new FileReader(file));
12 String line;
13 while((line = fln.readLine()) != null){
14 i++;
15 System.out.println(i+" : "+line);
16 }
17 }
18 catch (FileNotFoundException e){
19 System.out.println("File tidak ditemukan");
20 }
21 }
22 }
```

Lakukan modifikasi program sehingga dapat menampilkan banyaknya huruf, angka dan karakter lain dari file yang ditampilkan.

## II. Program untuk menulis file teks

Program di bawah ini merupakan modifikasi dari program di atas, yaitu dengan membaca file dan kemudian menuliskannya ke file lain dngan menambahkan nomor baris. Tulis dan jalankan program berikut (**Prak\_602.java**).

```
01 import java.io.*;
02 public class Prak_602 {
03 public static void main (String[] args) throws Exception {
04 File fileIn = new File("Prak_602.java");
05 File fileOut = new File("hasil.txt");
06 if (!fileIn.exists() || !fileIn.canRead()){
07 System.out.println("Can't read " + fileIn);
08 return;
09 }
10 try {
11 int i=0;
12 BufferedReader fln=new BufferedReader(new FileReader(fileIn));
13 BufferedWriter fOut=new BufferedWriter(new FileWriter(fileOut));
14 String line;
15 while((line = fln.readLine()) != null){
16 i++;
17 if (i<10) fOut.write(" ");
18 else if (i<100) fOut.write(" ");
19 fOut.write(i+" : "+line);
20 fOut.newLine();
21 }
22 fOut.close();
23 fln.close()
24 }
25 catch (FileNotFoundException e){
26 System.out.println("File tidak ditemukan");
27 }
28 }
29 }
```

### III. Soal latihan

#### Soal 6.1 (Soal\_601.java)

Buat program untuk membaca file teks mhs.txt yang memiliki isi sebagai berikut :

1234567890123456789023456789 ← tulisan ini tidak disimpan

|       |       |    |    |    |
|-------|-------|----|----|----|
| 01001 | adi   | 45 | 65 | 89 |
| 01002 | budi  | 80 | 45 | 70 |
| 01003 | cahyo | 60 | 80 | 30 |
| 01004 | dedi  | 70 | 80 | 70 |

Dimana data yang disimpan adalah nim, nama, nilai tugas, nilai uts, dan nilai uas.



## Project Akhir

### Tujuan Pembelajaran

Setelah mempelajari bab ini, diharapkan mampu:

- Project Akhir

### Pokok Bahasan

- Project Akhir

Buat program Sistem Informasi Perhotelan dengan menggunakan Java, dimana data yang disimpan adalah :

1. Data kamar, yang berisi data jumlah kamar yang dimiliki oleh hotel tersebut, termasuk status dari kamar tersebut, apakah terisi, dibooking atau kosong. Kamar yang ada terdiri dari tiga jenis kamar, yaitu single (harga sewanya Rp. 175.000,-), double (harga sewanya Rp. 225.000,-) dan suite (harga sewanya Rp. 300.000,-). Jumlah kamar yang dimiliki suatu hotel ditentukan oleh user dan disimpan pada suatu file konfigurasi.
2. Data penyewa, yang berisi data tamu yang menginap di hotel tersebut, yang minimal menyimpan data no KTP/SIM, nama, alamat, dan no telepon.
3. Data transaksi, yang berisi proses transaksi antara penyewa dan kamar yang disewa

Adapun fasilitas / proses yang dimiliki oleh program ini adalah sebagai berikut :

1. Dapat menampilkan data kamar beserta statusnya. Termasuk memberikan summary berapa jumlah kamar yang terisi, yang kosong, dan yang dibooking.
2. Dapat menampilkan data penyewa termasuk kamar yang disewa. Juga dapat memberikan summary tentang penyewa yang sering menginap di hotel tersebut.
3. Dapat mengetahui total transaksi yang dilakukan pada suatu waktu tertentu (perbulan atau pertahun tertentu).
4. Tambahan fasilitas lain akan memberikan nilai tambah.

Untuk proses pemilihan proses digunakan sistem menu, yang formatnya ditentukan sendiri oleh pembuat program.