**JOBSHEET**

**penggunaan tipe data, variabel, konstanta**

**A. TUJUAN**

* 1. Siswa mampu memahami konsep TIPE DATA
  2. Siswa mampu memahami VARIABEL & KONSTANTA
  3. Siswa mampu menyajikan tipe data primitif dan referensi
  4. Siswa mampu menyajikan deklarasi, aturan penamaan

**b. dasar teori**

**tipe data**

Tiap bahasa pemrograman mempunyai bentuk tipe-tipe data sendiri-sendiri yang merupakan pengembangan tipe data algoritma. Dalam algoritma dikenal 5 tipe data, yaitu : tipe data bilangan bulat (integer), tipe data bilangan real, tipe data karakter, tipe data string dan tipe data logika atau boolean.

**variabel & konstanta**

Variabel adalah suatu lokasi memori komputer yang digunakan untuk menampung dan menyimpan data yang akan diolah.

Tipe dari sebuah variabel ditentukan oleh jenis data yang akan disimpan. Penamaannya bersifat UNIK, yaitu tidak boleh ada 2 variabel atau lebih dengan nama yang sama. Sebuah variabel hanya dapat digunakan untuk menyimpan satu jenis data, tidak tergantung besarnya data tersebut dan nilainya boleh berubah-ubah.

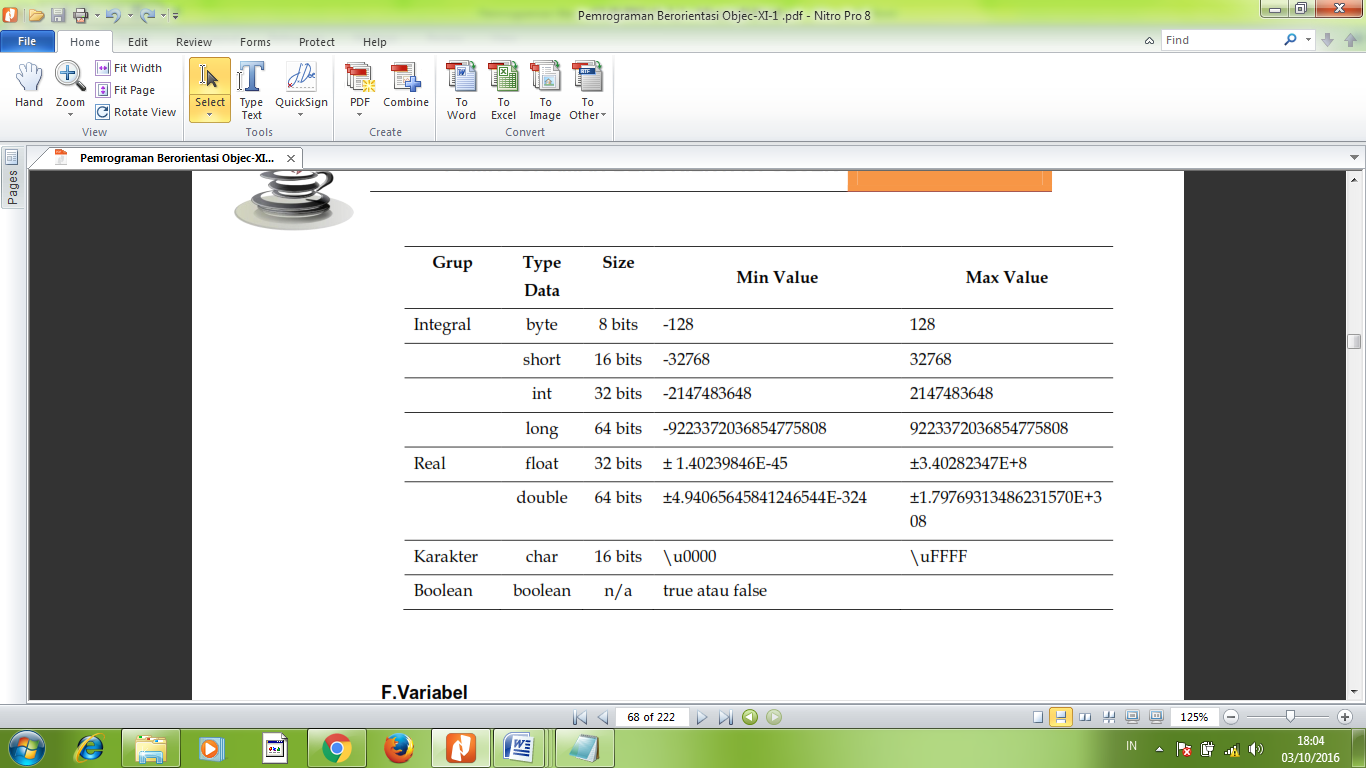
Konstanta adalah besaran yang mempunyai nilai tetap selama program dijalankan. Pemberian nilai akan disimpan pada sebuah konstanta dilakukan di awal program/ algoritma dan selama program dijalankan nilainya tidak bisa diubah.

**tipe data primitif**

## Macam-macam Tipe Data

Berikut ini macam-macam tipe data pada Java:

* char: Tipe data karakter, contoh Z
* int: angka atau bilangan bulat, contoh 29
* float: bilangan desimal, contoh 2.1
* double: bilangan desimal juga, tapi lebih besar kapasistanya, contoh 2.1
* String: kumpulan dari karakter yang membentuk teks, contoh Hello World!
* boolean: tipe data yang hanya bernilai true dan false



**tipe data referensi**

Tipe data Referensi adalah tipe data yang digunakan dalam OOP atau Object Oriented Programming (Pemrograman berorientasi Objek). Jadi dalam OOP ini kita dapat membuat atau mendefinisikan tipe data baru yang merupakan objek dari class tertentu. Semua tipe data referensi nilai defaultnya adalah nol atau null, tidak seperti tipe data primitif yang memiiliki nilai default variatif.  
Fungsi yang digunakan :

Beberapa perbedaan antara Tipe data Primitif dan Referensi :  
Tipe Data PrimitifTipe data primitif langsung di definisikan oleh java.  
Tipe primitif merepresentasikan nilai tunggal.

Tipe Data Relatif  
Tipe data Referensi di definiskan oleh User.  
Tipe dara referensi dapat menapung nilai nol atau lebih nilai primitif atau object.

**aturan java**

Ada aturan yang harus diikuit, diantaranya:

* Nama variabel tidak boleh menggunakan kata kunci dari Java *(reserved word)* seperti if, for, switch, dll.
* Nama variabel boleh menggunakan huruf, angka (0-9), garis bawah *(underscore)*, dan symbol dollar ($), namun penggunaan garis bawah dan symbol lebih baik dihindari.
* Nama variabel harus diawali dengan huruf kecil, karena Java menggunakan [gaya CamelCase](https://www.petanikode.com/gaya-penulisan-case-dalam-pemrograman/).
* Apabila nama variabel lebih dari 1 suku kata, maka kata ke-2 dituliskan dengan diawali dengan huruf besar dan seterusnya, contoh namaVariabel.

**c. latihan**

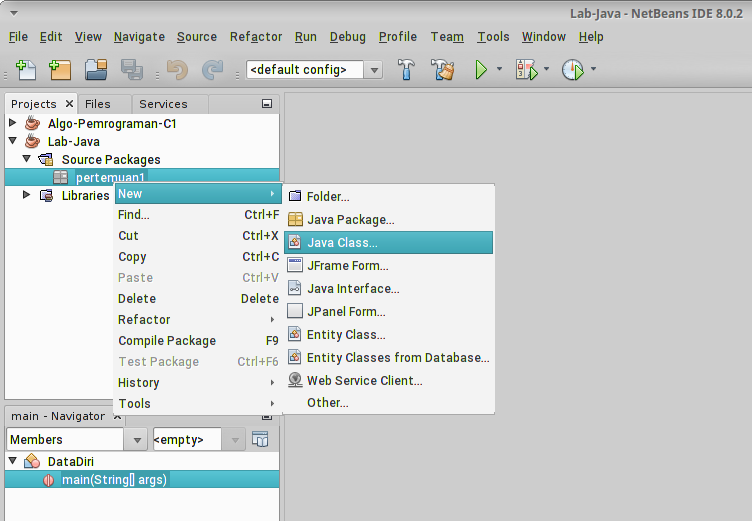
* Apa yang terjadi jika variabel bertipe data int diisi dengan angka desimal?
* Apa yang terjadi jika variabel bertipe data int diisi dengan teks atau sebaliknya?
* Apa yang terjadi jika variabel bertipe data byte diisi nilai 257?
* Manakah deklarasi variabel yang benar sesuai aturan penamaan di Java

1. int 4ngka;
2. int BilanganGenap
3. int Bilangan-Ganjil
4. int namaSiswa
5. dst.

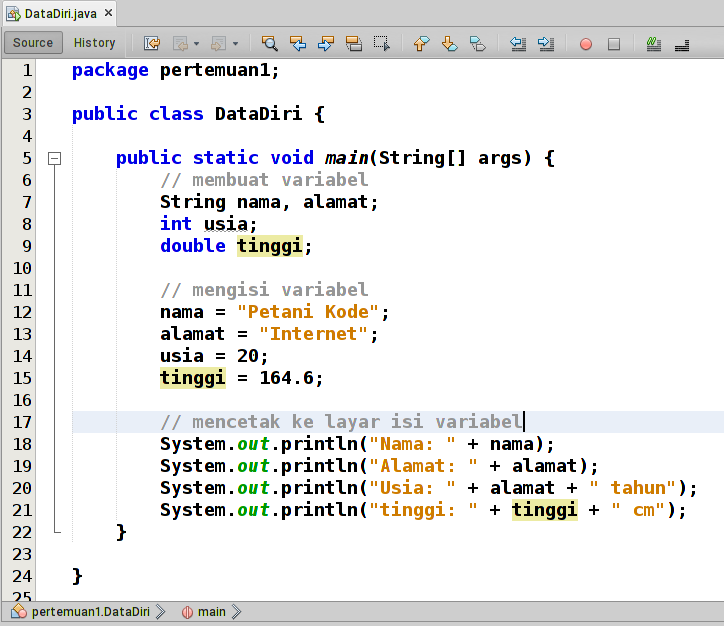
**d. tugas praktikum**

PRAKTIKUM 1

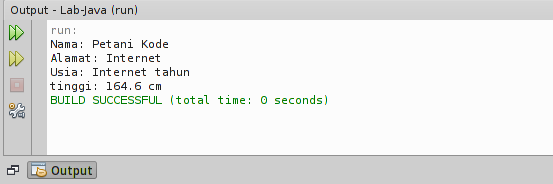
Buatlah program baru bernama DataDiri. Silahkan tambahkan kelas *(java class)* baru di paket *(package)* pertemuan1.



Pada jendela yang muncul, isi nama kelasnya dengan *DataDiri*. Setelah itu, silahkan ikuti kode berikut.



Setelah itu, coba jalankan dengan menekan tombol [Shift]+[F6]. Analisa dan pahami maksud kode-kode di atas.



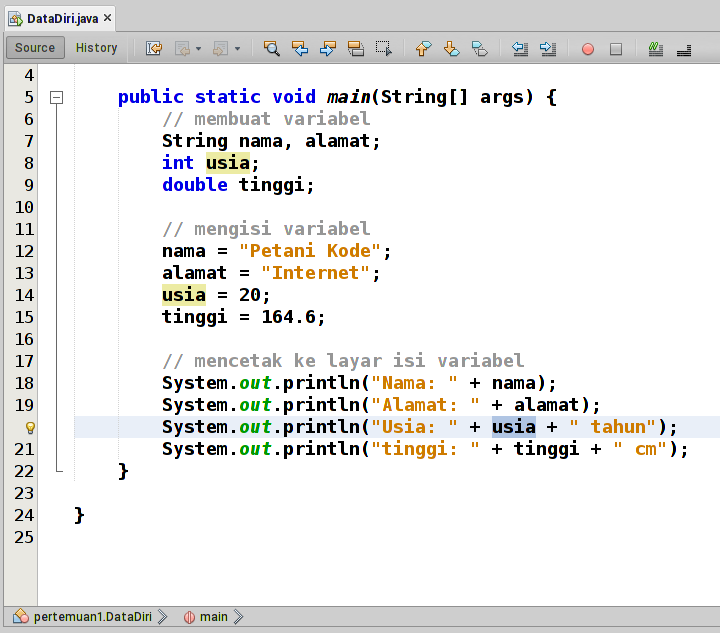
Hasil outputnya sepertinya salah.

Usia kok nilianya alamat, mengapa demikian?

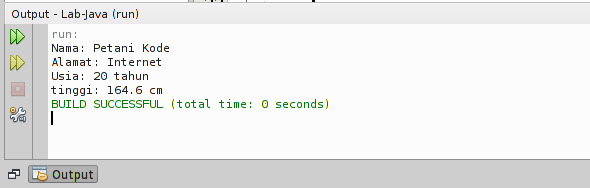
Itu disebabkan karena kita memanggil variabel alamat pada baris *usia*.

Jadi yang akan tampil adalah isi dari variable alamat, bukan isi dari variabel usia.

Perbaikilah kodenya menjadi seperti ini:



Dan, … silahkan coba jalankan kembali programnya.



Yap! hasilnya sudah benar.

**Latihan Ke-2: Program LuasLingkaran**

Mari kita mantapkan pemahaman dengan membuat program *LuasLingkaran*.

Program ini fungsinya untuk menghitung luas lingkaran.

Luas lingkaran dapat kita hitung dengan rumus PI x r2.

Sebelum memulai memrogram, sebaiknya kita pahami dulu algoritma dan flowchart-nya:

**Algoritma**

Deklarasi:

Double luas, PI

int r

Deskripsi:

- Input

PI = 3.14

r = 18

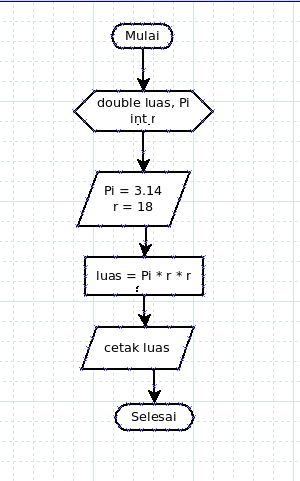
- Proses

luas = PI \* r \* r

- Output

cetak luas

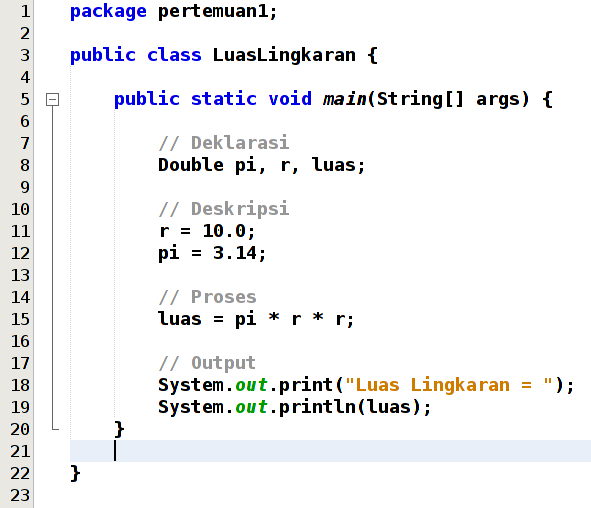
**Flow Chart:**



Nah, sudah jelas kan?

Sekarang kita tinggal *coding* saja.

PRAKTIKUM 2



Silahkan dieksekusi dan pertahikan hasilnya.

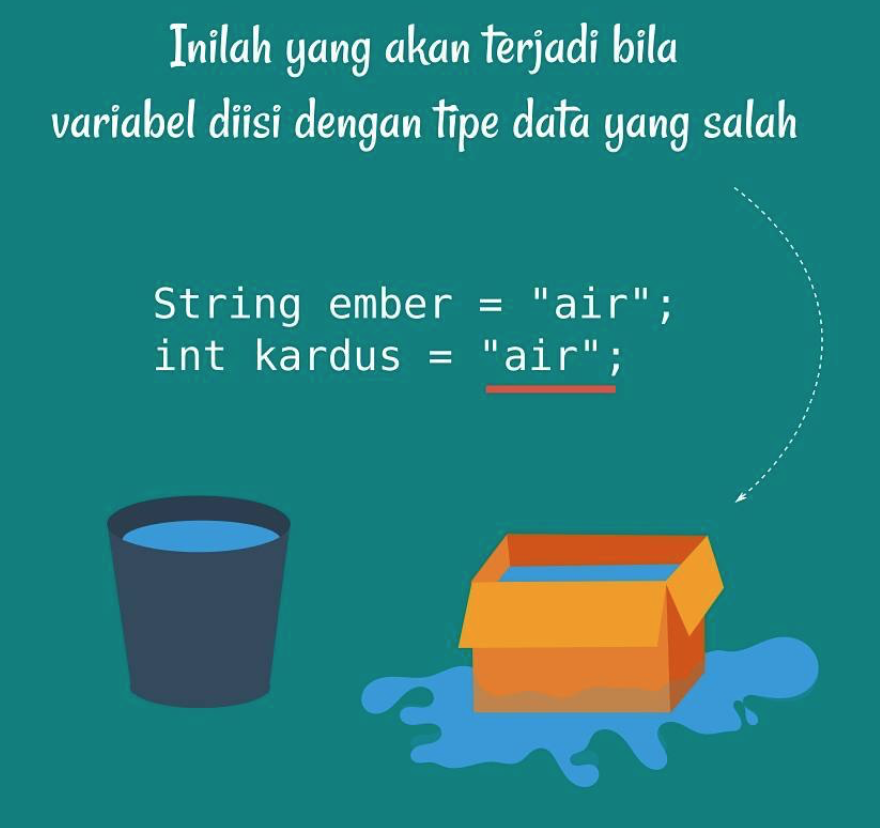
PRAKTIKUM 3

**Konversi Tipe Data**

Konversi artinya merubah ke jenis yang lain.

Kenapa kita perlu konversi tipe data?

Untuk menjawabnya, saya ingin tunjukan ilustrasi berikut ini:

****

Air yang bentuknya cair tidak bisa disimpan dalam kardus. Karena itu, air harus konversi dulu menjadi bentuk padat (es) agar bisa disimpan dalam kardus.

Begitu juga dengan variabel.

Tipe data string tidak akan bisa disimpan dalam variabel dengan tipe integer.

### Cara Konversi Tipe Data

Contoh: Konversi ke tipe integer

Cara 1:

Integer.perseInt(variabel);

Integer.perseInt(1.2);

Cara 2:

Integer.valueOf(2.1);

Cara 3:

objek.toInt();

Cara 4:

(int) variabel;

(int) 2.1;

Ternyata, ada banyak sekali cara.

### Mari Kita Coba dalam Program…

Buatlah kelas (java class) baru di paket pertemuan2.

Berikan nama: LuasSegitiga.

Kemudian ikuti kode program berikut:

package pertemuan2;

import java.util.Scanner;

public class LuasSegitia {

public static void main(String[] args) {

// deklarasi

Double luas;

int alas, tinggi;

// mebuat scanner baru

Scanner baca = new Scanner(System.in);

// Input

System.out.println("== Program hitung luas Segitiga ==");

System.out.print("Input alas: ");

alas = baca.nextInt();

System.out.print("Input tinggi: ");

tinggi = baca.nextInt();

// proses

luas = Double.valueOf((alas \* tinggi) / 2);

// output

System.out.println("Luas = " + luas);

}

}

Variabel luas bertipe data Double, berarti nilai yang bisa disimpan adalah Double.

Sedangkan variabel alas dan tinggi bertipe Integer.

Agar hasil operasi data integer dapat disimpan dalam variabel bertipe double, maka perlu dikonversi.

Kalau dijalankan, akan menghasilkan:

