

LAPORAN AKHIR
PEMBUATAN PROGRAM APLIKASI MENGHITUNG LUAS
BANGUN DATAR DAN VOLUME BANGUN RUANG



Disusun Oleh :
Yohana Resty Agatha M. (202013044)

Dosen Pengampu:
Slamet Triyanto, S.ST

PROGRAM STUDI TEKNIK INFORMATIKA
POLITEKNIK KAMPAR
2021

KATA PENGANTAR

Puji dan syukur atas kehadiran Tuhan Yang Maha Esa, atas segala limpahan Nya yang telah memberi penulis kesempatan untuk menyelesaikan Laporan Algoritma dan Pemograman ini. Adapun tujuan penulisan Laporan ini adalah untuk melengkapi Tugas ujian akhir semester Algoritma dan Pemograman.

Dalam proses pembuatan Laporan ini, tentunya penulis mendapat bimbingan, arahan, koreksi dan saran. Untuk itu penulis mengucapkan terima kasih kepada Bapak Slamet Triyanto, S.ST selaku dosen pengampu Algoritma dan Pemograman.

Penulis menyadari bahwa baik dari segi penulisan maupun isi, laporan ini masih memiliki kekurangan, oleh karena itu penulis sangat mengharapkan kritik yang membangun dan saran dari pembaca agar terbentuknya kesempurnaan laporan ini. Atas partisipasinya penulis mengucapkan terima kasih.

Bangkinang, 12 Maret 2021

Penulis

DAFTAR ISI

KATA PENGANTAR	i
DAFTAR GAMBAR	v
DAFTAR TABEL	vi
BAB I	1
TINJAUAN PUSTAKA	1
A. Pengertian	1
1. Pengertian Bahasa Java	1
B. Tujuan Praktikum	2
C. Alat Dan Bahan	2
BAB II	3
PEMBAHASAN	3
A. Variabel Class/Static	3
A. Tipe Data <i>Primitif</i>	4
B. Literasi dalam <i>Java</i>	6
C. Operator Aritmatika	6
D. Operator Relasional	7
E. Operator Logika	8
B. Modifier Akses	9
C. Perulangan Java	9
D. Kondisional If	11
E. Algoritma Sorting	14
BAB III	16
LANGKAH KERJA	16
A. Program Bangun Datar dan Bangun Ruang	16
B. A. Program Aplikasi Menghitung Luas Bangun Datar dan Volume Bangun Ruang.	17
C. Hasil Compaile:	24
BAB IV	24

PENUTUP	24
1. Kesimpulan.....	24
2. Saran	25
DAFTAR PUSTAKA	29

DAFTAR GAMBAR

Gambar 2.1 variabel kelas.....	3
Gambar 2.2 hasil output	4
Gambar 3.1 Aplikasi Sublime	16
Gambar 3.2 membuat file baru	17
Gambar 3.3 tampilan awal new file	17
Gambar 3.4 menampilkan header file	17
Gambar 3.5 menampilkan penamaan kelas dan method utama	17
Gambar 3.6 tampilan baris source code form login dan pemilihan	18
Gambar 3.7 pemilihan menu	19
Gambar 3.8 program aplikasi volume bangun ruang	20
Gambar 3.9 case 5 untuk sorting.....	21
Gambar 3.10 untuk pengurutan	21
Gambar 3.11 Aplikasi perhitungan luas bangun datar	22
Gambar 3.12 Gambar pemilihan.....	23
Gambar 3.13 Akhir Program	23
Gambar 3.14 Membuka tampilan cmd	24
Gambar 3.15 memanggil directory D.....	24
Gambar 3.16 memanggil directory D.....	25
Gambar 3.17 Form Login	25
Gambar 3.18 Tampilan Menu	26
Gambar 3.19 Program Luas.....	26
Gambar 3.20 Hasil Sorting	27
Gambar 3.21 Tampilan Menu Ketiga	28
Gambar 3.22 Tampilan Akhir Program	28

DAFTAR TABEL

Tabel 2.1 karakter <i>escape-sequence</i>	6
Tabel 2.2 <i>Operator</i> aritmatika.....	6
Tabel 2.3 <i>Operator Relasional</i>	7
Tabel 2.4 <i>Operator Logika</i>	8

BAB I

TINJAUAN PUSTAKA

A. Pengertian

1. Pengertian Bahasa Java

Menurut (Putri Indriyani, 2012) *Java* adalah sebuah teknologi yang diperkenalkan oleh Sun Microsystems pada pertengahan tahun 1990. Menurut definisi dari Sun, *Java* adalah nama untuk sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada komputer *standalone* ataupun pada lingkungan jaringan. *Java* lebih banyak disebut sebagai sebuah teknologi dibanding hanya sebuah bahasa pemrograman, karena *Java* lebih lengkap dibanding sebuah bahasa pemrograman konvensional. Teknologi *Java* memiliki tiga komponen penting, yaitu :

- a) *Programming*
- b) *Language Specification*
- c) *Application-Programming Interface*
- d) *Virtual-Machine Specification*

Kelebihan *Java* :

- a) Berorientasi objek, Memudahkan untuk mendesain dan mengembangkan program dengan cepat dan teliti , sehingga mudah digunakan. Salah satu bahasa pemrograman yang berorientasi objek secara murni .
- b) Mirip C++, mempunyai sintaks yang mirip dengan bahasa pemrograman C++. Sehingga para pengguna C++ banyak yang hijrah menggunakan Java.
- c) *Multiplatform*, Dapat digunakan di banyak sistem operasi.
- d) Perkembangan yang luas, dari *game* sampai sistem operasi handphone menggunakan program *java*. Misalnya handphone Sony Ericsson dan Opera Mini 3 yang bertipe .jar (*Java Archive*).
- e) Mempunyai pengumpulan sampah otomatis.

Kekurangan Java :

- a) Proses *Compile*, mengharuskan pengguna mengcompile programnya sebelum dijalankan, berbeda dengan bahasa pemrograman python yang tidak perlu mengcompile terlebih dahulu.
- b) Penggunaan Memori yang besar, berbeda dengan bahasa pemrograman lain yang hanya membutuhkan memori sedikit.
- c) Program yang dibuat oleh java lebih lambat dibanding program yang dibuat dengan bahasa lain seperti c atau C++

B.Tujuan Praktikum

- 1. Untuk mahasiswa lebih memahami Bahasa Java
- 2. Untuk memahami tipe data, percabangan, perulangan dan pengurutan
- 3. Untuk mengetahui *source code* yang digunakan untuk membuat aplikasi mesin kasir.

C. Alat Dan Bahan

- 1. Alat
 - a) Laptop
 - b) Mouse
- 2. Bahan
 - a) Aplikasi Sublime
 - b) Rumus Matematika

BAB II

PEMBAHASAN

A. Variabel Class/Static

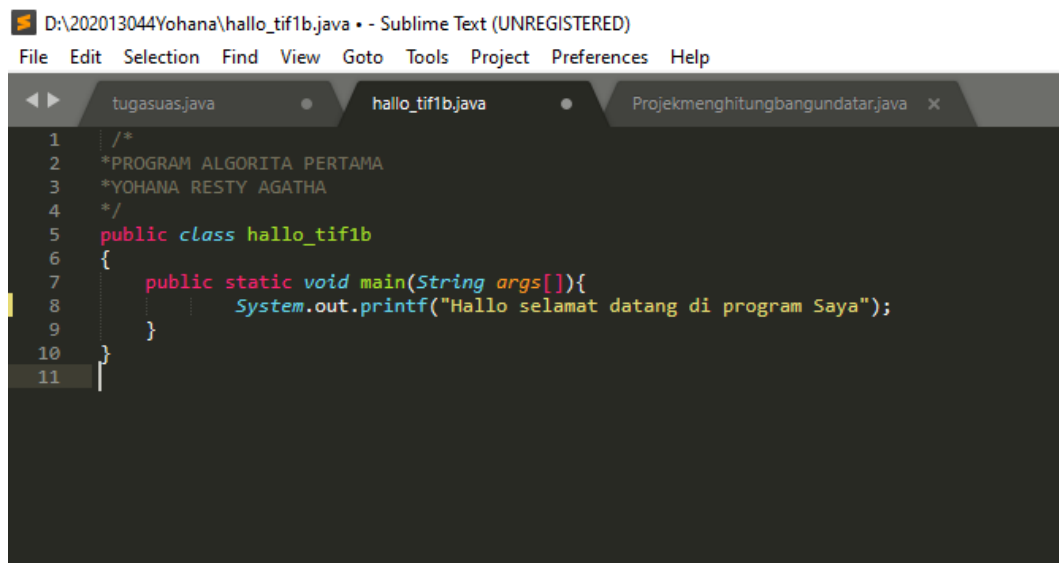
Variabel *class* atau biasa disebut *static* dideklarasikan dengan kata kunci *static* di dalam *class* tetapi diluar method, konstruktor, ataupun blok. Hanya ada satu salinan dari *variable class/static* ini per *class*.

Variabel *static* ini biasa nya di deklarasikan sebagai konstanta. Variabel konstanta nilainya tetap dan tidak berubah. Variabel *static* dibuat ketika program dimulai dan dihapus saat program berhenti. Nilai *default* nya sama dengan variabel *instance*.

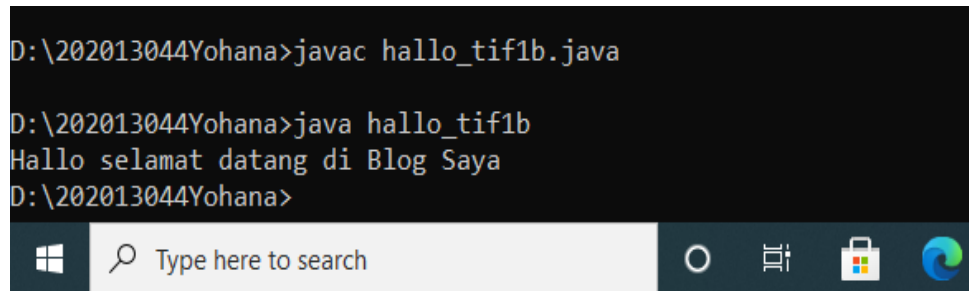
Variabel *static* dapat diakses dengan memanggil nama *class*, misalnya Nama *Class*, NamaVariabel. Saat mendeklarasikan variabel *class* dengan *public static final*, maka nama variabel (konstanta) dituliskan semua dalam huruf kapital. Jika variabel *static* tidak dideklarasikan dengan *public final*, *syntax* penamaannya sama dengan variabel *instance* dan *local*.

Berikut ini contoh penggunaan variabel kelas:

Gambar 1.1 variabel kelas



Berikut adalah hasil output dari kode di atas :



```
D:\202013044Yohana>javac halo_tif1b.java

D:\202013044Yohana>java halo_tif1b
Halo selamat datang di Blog Saya
D:\202013044Yohana>
```

Gambar 2.2 hasil output

A. Tipe Data *Primitif*

Terdapat delapan tipe data *primitif* yang didukung dalam *java*. Tipe data *primitif* didefinisikan oleh bahasa dan diberi nama dengan sebuah kata kunci. Berikut ini tipe data primitif dalam *java*.

1. *Byte*

Tipe data *byte* merupakan *integer* 8-bit yang bersifat *signed*. Nilai *minimum* adalah -128, nilai *maksimum* adalah 127 dan nilai *default* adalah 0. Tipe data *byte* digunakan untuk menghemat ruang pada array yang besar. Contoh : *byte* a = 100, *byte* b = 50.

2. *Short*

Tipe data *short* merupakan *integer* 16-bit yang bersifat *signed*. Nilai *minimum* adalah -32,768, nilai *maksimum* adalah 32,767 dan nilai *default* adalah 0. Tipe data *short* juga dapat menghemat ruang seperti pada tipe data *byte*. Contoh: *short* s = 10000, *short* r = -20000.

3. *Int*

Tipe data *int* merupakan *integer* 32-bit yang bersifat *signed*. Nilai *minimum* adalah -2,147,483,648, nilai *maksimum* adalah 2,147,483,647 dan nilai *default* 0. Tipe data *int* digunakan secara umum untuk menampung nilai-nilai integral. Contoh: *Int* a = 100000, *int* b = -200000.

4. *Long*

Tipe data *long* merupakan integer 64-bit yang bersifat signed. Nilai minimum adalah -9,223,372,036,854,775,808, nilai maksimum adalah -9,223,372,036,854,775,807 dan *default* adalah 0L. Tipe data *long* digunakan untuk menampung nilai-nilai dengan jangkauan nilai yang lebih dari tipe data *int*. Contoh: *int* a = 100000L, *int* b = -200000L.

5. *Float*

Tipe data *float* merupakan nilai *decimal* 32-bit presisi tunggal. *Float* biasanya digunakan untuk menghemat ruang pada *array* besar yang berisi nilai-nilai *decimal* dan nilai *default* adalah 0.0f. Tipe data *float* tidak pernah digunakan untuk nilai-nilai seperti nilai mata uang. Contoh: *float* f1 = 234.5f.

6. *Double*

Tipe data *double* merupakan nilai desimal 64-bit presisi ganda. *Double* biasanya digunakan sebagai tipe data *default* untuk nilai *decimal*. Nilai *default* adalah 0.0d. Tipe data *double* tidak pernah digunakan untuk nilai-nilai presisi seperti nilai mata uang. Contoh: *double* d1 = 123.4.

7. *Boolean*

Tipe data *Boolean* merepresentasikan satu bit informasi. Hanya ada dua nilai yang mungkin: *true* (benar) dan *false* (salah), tipe data ini digunakan sebagai penanda kondisi benar/salah. Nilai *default* adalah *false*. Contoh: *Boolean* satu = *true*.

8. *Char*

Tipe data *char* merupakan karakter *Unicode* 16-bit tunggal. Nilai minimum adalah ``\u0000`` dan nilai maksimum adalah ``\uffff``. Tipe data *char* digunakan untuk menampung karakter apa saja. Contoh: *char* huruf A = ``A``.

B.Literasi dalam Java

Bahasa Java mendukung beberapa *escape sequence* untuk literal *String* dan *char*.

Tabel 2.1 karakter *escape-sequence*

<code>\n</code>	Fungsi CR/LF (<i>Carriage Return and Line Feed</i>).Cursor pindah ke kolom pertama baris berikutnya.
<code>\b</code>	<i>Back Space</i> , <i>Cursor</i> mundur satu langkah pada baris yang sama.
<code>\r</code>	<i>Awal Baru</i>
<code>\t</code>	<i>Tab</i> . <i>Cursor</i> pindah ke posisi tabulasi berikutnya.
<code>\f</code>	Halaman Baru
<code>\'</code>	Untuk menampilkan karakter tanda petik tunggal(<i>single quotation mark</i>)
<code>\"</code>	Untuk menampilkan karakter tanda petik ganda(<i>double quotation mark</i>).

C.Oprator Aritmatika

Operator aritmatika digunakan pada ekspresi matematik seperti pada operasi aljabar. Diansumsi kan bahwa variabel A bernilai 10 dan B bernilai 20, maka berikut ini daftar operator aritmatika:

Tabel 2.2 Operator aritmatika

Operator	Deskripsi	Contoh
+	Operator penjumlahan. Berfungsi untuk menambahkan	A + B hasilnya 30

	operan	
-	Operator pengurangan. Berfungsi untuk mengurangi operan	A – B hasilnya - 10
*	Operator perkalian. Berfungsi untuk mengalikan operan.	A * B hasilnya 200
/	Operator pembagian. Berfungsi untuk membagi operan.	B/A hasilnya 2
%	Operator Modulus. Berfungsi mengurangi nilai operan di sebelah kiri dengan nilai operan di sebelah kanan dan mengembalikan sisa nilainya.	B%A hasilnya 0
++	Peningkatan(menambahkan nilai 1 operan)	B++ hasilnya 21
--	Penurunan (mengurangkan 1 nilai operan)	B—hasilnya 19

D.Operator Relasional

Operator ini digunakan untuk membandingkan dua operan yang umumnya dimanfaatkan dalam pembuatan struktur if. Berikut ini operator *relasional* yang didukung dalam java. Diasumsikan bahwa variabel A bernilai 10 dan B bernilai 20,maka :

Tabel 2.3 *Operator Relasional*

Operator	Deskripsi	Contoh
==	Memeriksa apakah nilai kedua operan sama atau tidak, jika sama maka kondisi bernilai benar.	(A==B) adalah tidak benar

!=	Memeriksa apakah nilai kedua operan sama atau tidak , jika tidak sama maka kondisi bernilai benar.	(A!=B) adalah benar
>	Memeriksa apakah nilai operan disebelah kiri lebih dari nilai operan disebelah kanan, jika ya maka kondisi bernilai benar.	(A>B) adalah tidak benar
<	Memeriksa apakah nilai operan disebelah kiri kurang dari nilai operan disebelah kanan, jika ya maka kondisi bernilai benar.	(A<B) adalah benar
>=	Memeriksa apakah nilai operan disebelah kiri lebih dari atau sama dengan nilai operan di sebelah kanan, jika ya maka kondisi bernilai benar	(A>=B) adalah tidak benar
<=	Memeriksa apakah nilai operan disebelah kiri kurang dari atau sama dengan nilai operan di sebelah kanan, jika ya maka kondisi bernilai benar	(A<=B) adalah benar

E.Operator Logika

Oprator logika digunakna untuk mengetahui kondisi dari dua operan. Operator logika sering digunakan untuk membentuk struktur if. Berikut ini daftar operator logika. Diasumsikan bahwa nilai Boolean variabel A adalah *true* dan B adalah *false*, Maka :

Tabel 2.4 *Operator Logika*

Operator	Deskripsi	Contoh
&&	Operator logika <i>AND</i> . Jika kedua operator bukan nol, maka kondisi bernilai	(A && B) adalah tidak benar.

	benar.	
	Operator logika OP. Jika ada operan yang bukan nol, maka kondisi bernilai benar.	(A B) adalah benar.
!	Operator logika NOT. Digunakan untuk membalik keadaan logika dari operan. Jika kondisi bernilai benar maka operan NOT akan membuatnya menjadi tidak benar.	!(A && B) adalah benar.

B.Modifier Akses

Java menyediakan beberapa *modifier* akses. Fungsinya untuk mengatur level akses untuk *class*, variabel, method, dan *constructor*. Berikut ini empat level akses:

9. *Default*: Dapat diakses oleh *package*, yang merupakan level akses *default*. Tidak memerlukan *modifier*.
10. *Private*: Hanya dapat diakses oleh *class*.
11. *Public*: Dapat diakses semua.
12. *Protected*: Dapat diakses oleh *package* dan semua *sub-class*.

Untuk menggunakan *modifier*, anda menambahkan kata kunci pada definisi *class*, method, atau variabel. *Modifier* berada di bagian awal statemen.

C.Perulangan Java

Perulangan digunakan untuk mengulang bagian tertentu dari struktur pemrograman sampai ditemukan kondisi dimana perulangan tersebut harus dihentikan. Bahasa pemrograman memiliki struktur pemrograman yang sama namun ditulis dengan *syntax* yang berbeda satu dengan lainnya. Didalam bab ini akan dibahas beberapa perulangan yang dapat digunakan dalam pemrograman *java*.

1. Perulangan While

Perulangan *while* adalah struktur *control* yang memungkinkan anda mengulangi suatu proses dengan jumlah

perulangan tertentu sampai ditemukan kondisi dimana perulangan akan dihentikan. Berikut ini *sintaks* dari perulangan *while*:

```
While(ekspresi_Boolean)
{
    //kode program yang akan diulang
}
```

Selama ekspresi_Boolean bernilai benar (*True*), maka statemen dalam struktur *while* akan terus dieksekusi.

2. Perulangan *do...while*

Perulangan *do...while* sama seperti perulangan *while*, tetapi perulangan *do...while* pasti akan dieksekusi minimal satu kali. Berikut sintaksnya:

```
Do
{
    //Statemen
}
While(ekspresi_boolean);
```

3. Perulangan *For*

Perulangan *for* adalah struktur *control repetitive* yang memungkinkan anda untuk menjalankan proses dengan jumlah perulangan tertentu (jumlah perulangan sudah diketahui sebelumnya). Berikut ini *sintaks* nya:

```
For(inisialisasi;ekspresi_boolean;update)
```

```
{  
    //Statemen  
}
```

Berikut ini aliran proses dalam perulangan *for*:

- b. Bagian inisialisasi dieksekusi pertama kali dan hanya sekali. Bagian ini memungkinkan anda untuk mendeklarasikan dan menginisialisasi variabel *control* perulangan.
- c. Setelah itu, ekspresi *Boolean* dievaluasi. Jika bernilai benar, statemen dalam struktur *for* akan dieksekusi. Jika tidak, statemen tidak dieksekusi dan proses berlanjut pada bagian setelah perulangan *for*.
- d. Setelah statemen dalam *for* dieksekusi, aliran proses kembali pada bagian *update*. Statemen *update* ini memungkinkan anda untuk meng-update nilai di dalam variabel *control* dalam *for*.
- e. Ekspresi *Boolean* kemudian dievaluasi lagi. Jika benar, statemen dalam perulangan *for* kembali dieksekusi, dan dilanjutkan ke bagian *update*. Jika tidak, perulangan *for* akan selesai dan proses berlanjut pada bagian setelah perulangan *for*.
- f. Ekspresi *Boolean* kemudian dievaluasi lagi. Jika benar, statemen dalam perulangan *for* kembali dieksekusi, dan dilanjutkan ke bagian *update*. Jika tidak, perulangan *for* akan selesai dan proses berlanjut pada bagian setelah perulangan *for*.

D.Kondisional If

Statemen *if* digunakan untuk membuat percabangan dari struktur program *java*. Dengan demikian, Anda bisa membuat beberapa struktur program yang akan dijalankan salah satunya, atau beberapa diantaranya,

apabila ada kondisi yang terpenuhi. Struktur *if* ini dikenali di seluruh bahasa pemrograman dan secara umum, cara penggunaannya tidak berbeda jauh.

1. Statemen *if*

Statemen *if* dalam *java* terdiri dari sebuah ekspresi *Boolean* yang diikuti dengan satu statemen atau lebih. Berikut ini sintaks untuk statemen *if*:

```
If(ekspresi_boolean)
{
    //Statemen akan dieksekusi jika ekspresi Boolean
    //bernilai benar
}
```

2. Statemen *if...else*

Statemen *if* dapat diikuti dengan statemen *else* yang opsional agar anda bisa menyusun struktur program yang akan dijalankan apabila ekspresi *Boolean* yang diuji menghasilkan nilai salah. Berikut contoh sintaksnya:

```
If(ekspresi_boolean)
{
    //Dieksekusi jika ekspresi Boolean bernilai benar
}
else
{
    //Dieksekusi jika ekspresi Boolean bernilai salah
}
```

3. Statemen *if...else if...else*

Statemen *if* dapat diikuti dengan statemen *else if...else opsional*. Saat menggunakan statemen *if*, *else if*, *else* ada beberapa hal yang harus anda perhatikan:

- a. Sebuah statemen *if* dapat memiliki nol atau satu statemen *else* dan harus digunakan setelah statemen *else if*.

- b. Sebuah statemen *if* dapat memiliki nol atau banyak statemen *else if* dan harus digunakan sebelum statemen *else*.
- c. Setelah sebuah statemen *else if* dieksekusi, statemen *else if* atau *else* yang lain tidak akan diperiksa(akan langsung melompat pada kode setelah *else* terakhir)

4. Statemen *Switch*

Statemen *Switch* memungkinkan sebuah variabel diperiksa kesamaannya dengan daftar nilai-nilai yang ada. Setiap nilai dalam daftar disebut *case*. Berikut ini sintaks statemen *switch*:

```
Switch(expression)
{
case nilai:
//Statemen
break; //opsional
case nilai:
//Statemen
break; //opsional
// anda dapat menggunakan beberapa statemen case
default: //optional
//Statements
}
```

Berikut ini aturan penggunaan statemen *switch*:

Variabel yang digunakan dalam *switch* hanya boleh berupa *byte*, *short*, *int*, atau *char*. Anda dapat menggunakan beberapa *case* didalam *switch*. Setiap *case* diikuti dengan nilai yang akan dibandingkan dan tanda titik dua(:). Nilai *case* harus sama tipe datanya dengan variabel dalam *switch* dan harus berupa konstanta atau *literal*. Ketike variabel yang dibandingkan sama dengan dengan *case*, statemen pada case tersebut akan dieksekusi sampai pada kata kunci *break*. Ketika sampai pada kata kunci

break, *switch* berhenti dieksekusi, dan aliran proses melompat pada kode sesudah statemen *switch*.

Tidak semua *case* memerlukan *break*. Jika tidak ada *break*, aliran proses akan terus dijalankan pada *case* selanjutnya sampai pada *break*. Statemen *switch* dapat memiliki *case default*, yang diletakan pada akhir statemen. *Default* dijalankan jika tidak ada *case* yang dijalankan.

E.Algoritma Sorting

5. Insertion Sorting

Insertion Sorting adalah algoritma pengurutan sederhana yang *relative efisien* untuk *list* data kecil dan sering digunakan sebagai bagian dari algoritma yang lebih digunakan sebagai bagian dari algoritma yang lebih mutakhir. Algoritma ini bekerja dengan cara mengambil elemen dari list secara satu per satu dan menyisipkan elemen tersebut ke dalam posisi yang benar di list baru yang telah diurutkan (N. Wirth 1985).

6. Selection Sort

Algoritma *Selection Sort* memilih elemen elemen maksimum/minimum *array*, lalu menempatkan elemen maksimum/minimum itu pada awal atau akhir *array* (tergantung pada urutannya *ascending/descending*).Selanjutnya elemen tersebut tidak disertakan pada proses selanjutnya. Karena setiap kali *Selection Sort* harus membandingkan elemen – elemen data, algoritma ini termasuk dalam *Comparison Based Sorting* (Ananda,et al.2009).

Terdapat dua pendekatan dalam metode pengurutan dengan *Selection Sort*

- a. Algoritma pengurutan maksimum(*Maximum Selection Sort*), yaitu memilih elemen maksimum sebagai basis pengurutan.

- b. Algoritma pengurutan minimum(*Minimum Selection Sort*), yaitu memilih elemen minimum sebagai basis pengurutan.

7. *Quick Sort*

Quick Sort adalah salah satu jenis algoritma jenis pengurutan dengan konsep divide and conquer dengan mengandalkan operasi partisi, untuk mempartisi sebuah elemen array yang dipilih yang disebut sebagai *pivot* (Cormen. At al.2009).

8. *Merge Sort*

Algoritma *Merge Sort* mengambil keuntungan dari kemudahan menggabungkan daftar yang sudah diurutkan ke dalam daftar urutan yang baru. Dimulai dengan proses membandingkan setiap dua elemen(misalnya, 1 dengan 2,3 dengan 4, dan seterusnya) dan menukarnya jika nilai pertama lebih besar. Kemudian menggabungkan masing-masing dua elemen tersebut menjadi empat elemen dan seterusnya.(N.Wirth 1985).

9. *Array.sort() Java*

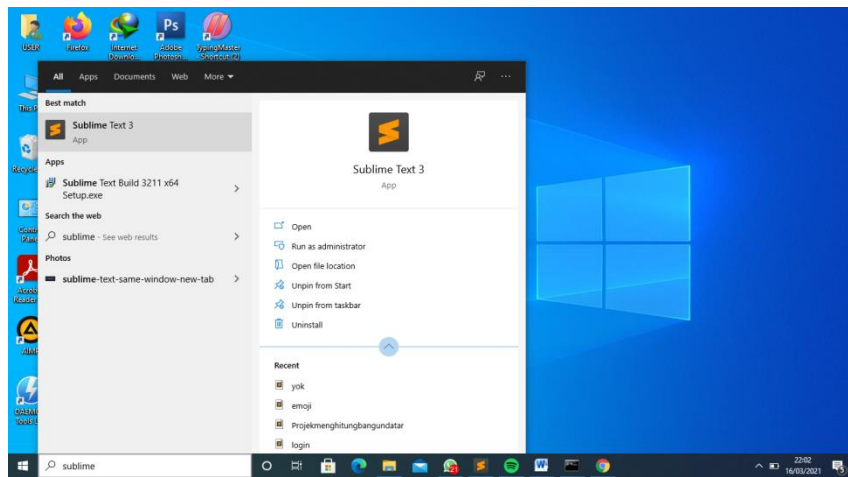
Di dalam *class library Java* terdapat sebuah method yang digunakan untuk mengurutkan data. Method tersebut berada di dalam *class Array*.

BAB III

LANGKAH KERJA

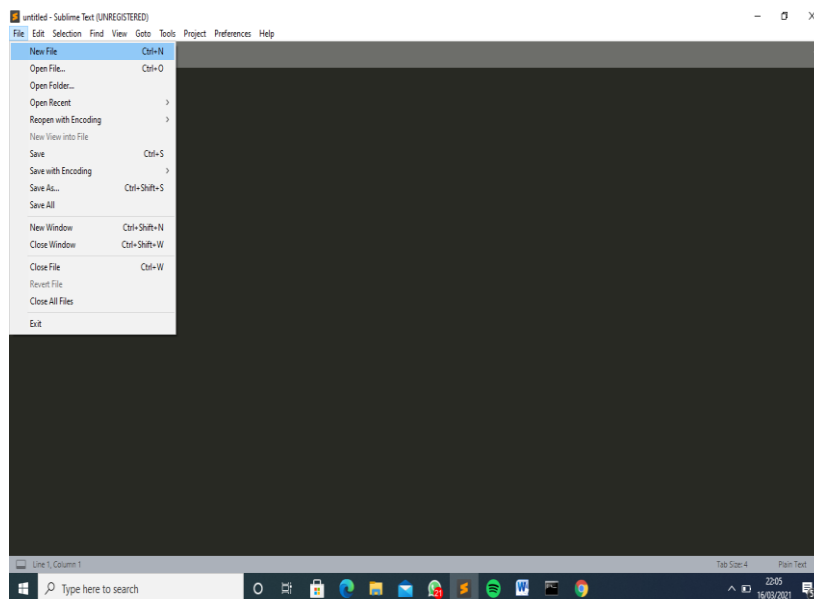
A. Program Bangun Datar dan Bangun Ruang

1. Pastikan dilaptop/*computer* anda sudah tersedia aplikasi sublime text. Untuk membuat *Source Code*.



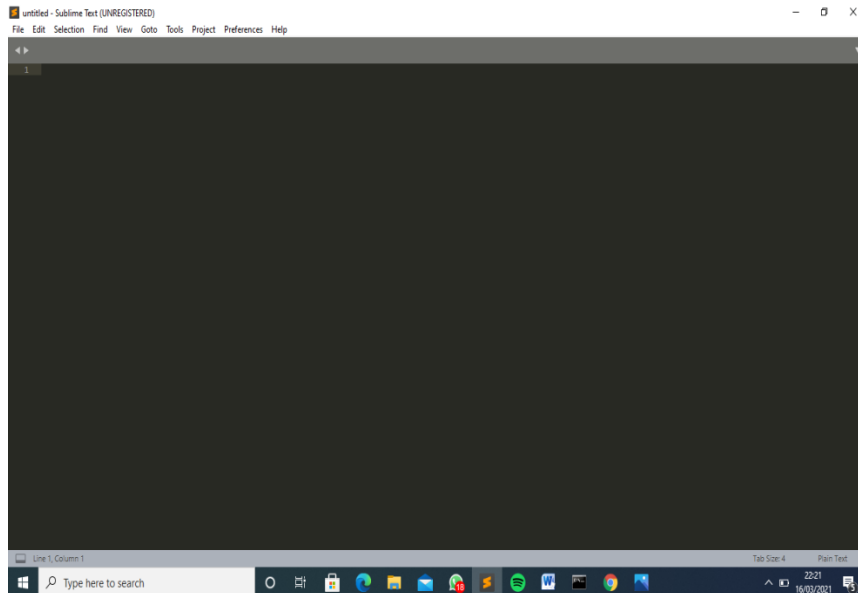
Gambar 3.1 Aplikasi Sublime

2. Buka aplikasi Pilih File > New File.



Gambar 4.2 membuat file baru

3. Maka akan muncul tampilan kosong seperti ini.



Gambar 5.3 tampilan awal new file

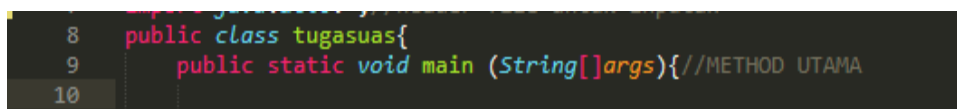
4. Kemudian buat *Source Code* mengenai perhitungan luas bangun datar dan volume bangun ruang

B. A. Program Aplikasi Menghitung Luas Bangun Datar dan Volume Bangun Ruang.



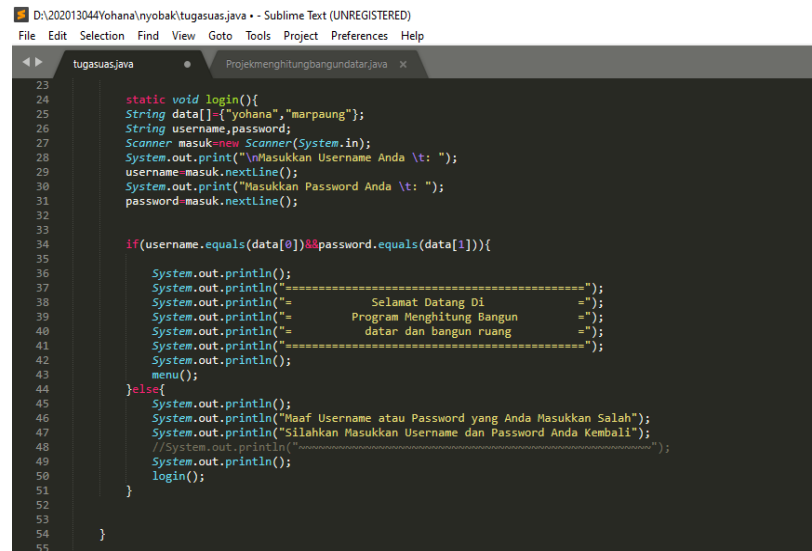
Gambar 6.4 menampilkan header file

Pada baris ke-7 ialah header file untuk program perhitungan bangun datar dan bangun ruang ini fungsinya untuk memanggil inputan



Gambar 7.5 menampilkan penamaan kelas dan method utama

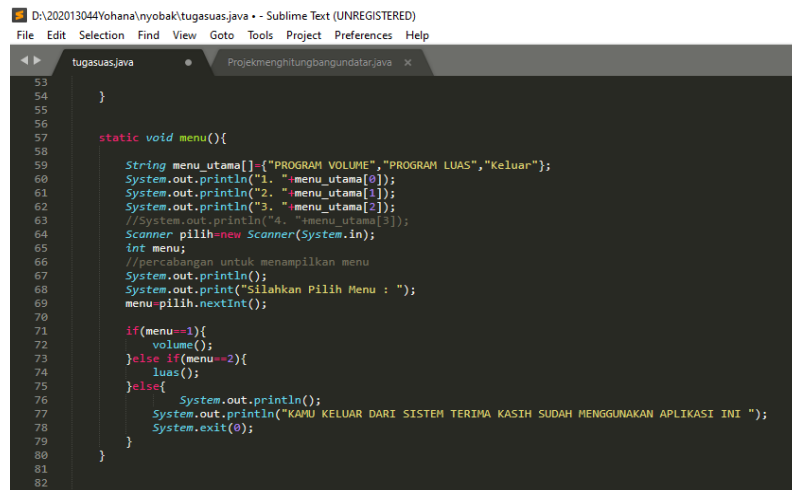
Pada baris ke-8 adalah penamaan kelas dimana penamaan ini sangat penting karena pada bahasa java nama kelas harus sama dengan file yang disimpan dengan *indeks.java* .Selanjutnya baris ke-10 adalah method utama



```
23
24
25 static void login(){
26     String data[]={"yohana","marpaung"};
27     String username,password;
28     Scanner masuk=new Scanner(System.in);
29     System.out.print("\nMasukkan Username Anda \t: ");
30     username=masuk.nextLine();
31     System.out.print("\nMasukkan Password Anda \t: ");
32     password=masuk.nextLine();
33
34     if(username.equals(data[0])&&password.equals(data[1])){
35
36         System.out.println();
37         System.out.println("=====");
38         System.out.println("                Selamat Datang Di                ");
39         System.out.println("                Program Menghitung Bangun            ");
40         System.out.println("                datar dan bangun ruang                ");
41         System.out.println("=====");
42         System.out.println();
43         menu();
44     }else{
45         System.out.println();
46         System.out.println("Maaf Username atau Password yang Anda Masukkan Salah");
47         System.out.println("Silahkan Masukkan Username dan Password Anda Kembali");
48         //System.out.println("=====");
49         System.out.println();
50         login();
51     }
52
53
54
55 }
```

Gambar 8.6 tampilan baris source code form login dan pemilihan

Baris ke-24 merupakan fungsi *void* penamaan kelas untuk *login* ke program. Pada baris ke-25 merupakan *array* yang akan diisi username dan password yang sudah ditanamkan untuk login. Baris ke-26 untuk mendeklarasikan tipe data *string*. Form login ini digunakan untuk masuk ke aplikasi menghitung luas bangun datar dan volume bangun ruang. Baris ke-34 adalah percabangan untuk memeriksa username dan password, baris ke-37 sampai baris ke-42 digunakan untuk menampilkan hasil output jika username dan password yang diinputkan benar sedangkan pada baris ke-44 sampai baris ke-49 adalah hasil output jika username atau password yang diinputkan salah.

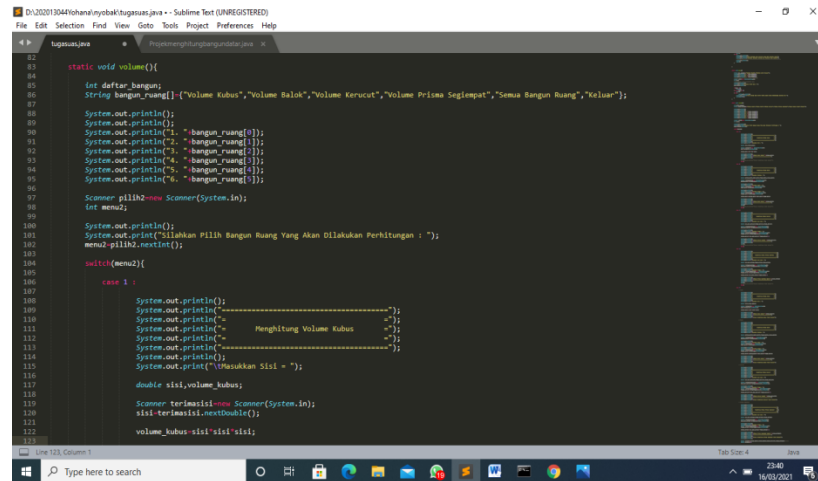


```
53
54
55
56
57 static void menu(){
58
59     String menu_utama[]={"PROGRAM VOLUME","PROGRAM LUAS","Keluar"};
60     System.out.println("1. "+menu_utama[0]);
61     System.out.println("2. "+menu_utama[1]);
62     System.out.println("3. "+menu_utama[2]);
63     //System.out.println("4. "+menu_utama[3]);
64     Scanner pilih=new Scanner(System.in);
65     int menu;
66     //percabangan untuk menampilkan menu
67     System.out.println();
68     System.out.print("Silahkan Pilih Menu : ");
69     menu=pilih.nextInt();
70
71     if(menu==1){
72         volume();
73     }else if(menu==2){
74         luas();
75     }else{
76         System.out.println();
77         System.out.println("KAMU KELUAR DARI SISTEM TERIMA KASIH SUDAH MENGGUNAKAN APLIKASI INI ");
78         System.exit(0);
79     }
80
81
82 }
```

Gambar 9.7 pemilihan menu

Disini terdapat beberapa *static void main(){* agar mempermudah dalam mengelompokan program. Pada baris ke-57 merupakan fungsi *void* untuk pemilihan menu. Terdapat perintah *System.out.println* yang digunakan untuk menampilkan inputan kelayar, pada baris ke-65 merupakan deklarasi tipe data *integer*.

Disini juga terdapat pemilihan kondisi dengan menggunakan percabangan *if else*. Baris ke-71 kondisi satu dimana jika *user* memilih satu maka program yang dijalankan adalah kondisi dari *if* satu yaitu perhitungan volume. Sementara baris ke-73 berisi *else if* dimana ketika *user* memilih dua maka program yang dijalankan adalah perhitungan luas. Apabila *user* memilih tiga maka program akan keluar dan baris ke-75 apabila *user* memilih diluar data yang disediakan maka program juga otomatis akan keluar.



```
82
83
84 static void volume(){
85     int daftar_bangun;
86     String bangun_ruang[]={"Volume Kubus","Volume Balok","Volume Kerucut","Volume Prisma Segiempat","Semua Bangun Ruang","Keluar"};
87
88     System.out.println();
89     System.out.println();
90     System.out.println("1. " + bangun_ruang[0]);
91     System.out.println("2. " + bangun_ruang[1]);
92     System.out.println("3. " + bangun_ruang[2]);
93     System.out.println("4. " + bangun_ruang[3]);
94     System.out.println("5. " + bangun_ruang[4]);
95     System.out.println("6. " + bangun_ruang[5]);
96
97     Scanner pilib2=new Scanner(System.in);
98     int menu;
99
100     System.out.println();
101     System.out.print("Silahkan Pilih Bangun Ruang Yang Akan Dilakukan Perhitungan : ");
102     menu= pilib2.nextInt();
103
104     switch(menu){
105
106         case 1 :
107             System.out.println();
108             System.out.println("-----");
109             System.out.println("1. " + bangun_ruang[0]);
110             System.out.println("2. " + bangun_ruang[1]);
111             System.out.println("3. " + bangun_ruang[2]);
112             System.out.println("4. " + bangun_ruang[3]);
113             System.out.println("5. " + bangun_ruang[4]);
114             System.out.println("6. " + bangun_ruang[5]);
115             System.out.print("\nMasukkan Sisi = ");
116
117             double sisi,volume_kubus;
118
119             Scanner terimasisi=new Scanner(System.in);
120             sisi=terimasisi.nextDouble();
121
122             volume_kubus=sisi*sisi*sisi;
```

Gambar 10.8 program aplikasi volume bangun ruang

Disini menggunakan *static void volume()*, pada baris ke-86 adalah array dengan tipe data string untuk menyimpan volume empat bangun ruang. Selanjutnya, baris ke-85 merupakan pendeklarasian tipe data *integer*. Sedangkan baris ke-96 sampai baris ke-101 adalah *indeks array* dari 0 sampai 5. Lanjut ke percabangan *switch case* yang dimulai pada baris ke-104 yaitu *switch* dan *case 1* pada baris ke-106 dimana kita akan menghitung volume kubus. Pada baris ke-117 kita menggunakan tipe data double yaitu untuk bilangan decimal atau pun bilangan biasa. Pada baris ke-122 adalah rumus untuk perhitungan volume kubus sereta akan dilakukan perhitungan aritmatik untuk rumus perhitungan volume kubus. Baris ke-128 *break*; untuk mengakhiri dari *case 1*.

```

230
231
232
233 // Kubus
234
235
236 System.out.println();
237 System.out.println("=====");
238 System.out.println(" ");
239 System.out.println(" Menghitung Volume Kubus ");
240 System.out.println("=====");
241 System.out.println();
242 System.out.print("\tMasukkan Sisi = ");
243
244 double sisi,volume_kubus;
245
246 Scanner terimasisi=new Scanner(System.in);
247 sisi=terimasisi.nextDouble();
248
249 volume_kubus=sisi*sisi*sisi;
250
251 System.out.println();
252 System.out.println("\tVolume Kubus Adalah " +volume_kubus);
253 System.out.println(" ");
254 System.out.println("=====");
255 System.out.println("\tProses Menghitung Kubus Telah Selesai");
256
257 // Balok
258
259 System.out.println();
260 System.out.println();
261 System.out.println("=====");
262 System.out.println(" ");
263 System.out.println(" Menghitung Volume Balok ");
264 System.out.println(" ");
265 System.out.println("=====");
266 System.out.println();
267 System.out.print("\tMasukkan Panjang = ");
268
269 double panjang_balok,lebar_balok,tinggi_balok,volume_balok;
270

```

Gambar 11.9 case 5 untuk sorting

```

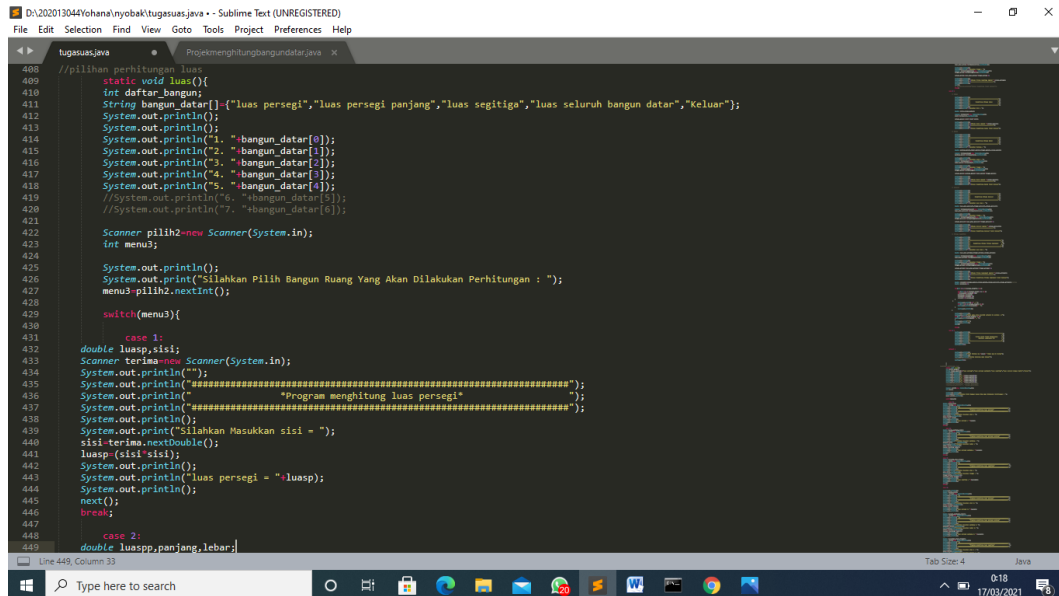
343
344
345 tinggi_prisma_tertinggi=prisma.nextDouble();
346
347 volume_prisma=luas_alas_prisma*tinggi_prisma/2;
348
349 System.out.println();
350 System.out.println("\tVolume Prisma Segiempat Adalah " +volume_prisma);
351 System.out.println(" ");
352 System.out.println("=====");
353 System.out.println("\tProses Menghitung Prisma Segiempat Telah Selesai");
354 System.out.println();
355
356 double volume[]={volume_kubus,volume_balok,volume_kerucut,volume_prisma}; //array
357 double penampung=0;
358
359 for(int h=0; h<(volume.length); h++){
360
361     for(int i=0; i<(volume.length-1); i++){
362         if(volume[i]>volume[i+1]){
363             penampung=volume[i];
364             volume[i]=volume[i+1];
365             volume[i+1]=penampung;
366         }
367     }
368     System.out.print(h+1+" = ");
369     for(int d=0; d<volume.length; d++){
370         System.out.print(volume[d]+" ");
371     }
372     System.out.println();
373
374
375 System.out.println();
376 System.out.print("\tData pada array hasil[] setelah di urutkan = (");
377 for(int i=0; i<volume.length; i++){
378     System.out.print(volume[i]+" , ");
379 }
380 System.out.print("\t");
381 System.out.println();
382 next();
383 break;

```

Gambar 12.10 untuk pengurutan

Alur program case 1 hingga case 4 itu sama yang membedakan hanya rumus matematikanya saja,tetapi untuk case 5 ini kita akan mengcompaile seluruh volume bangun ruang diatas setelah dilakukan compaile selanjutnya akan dilakukan sorting dimana menggunakan perulangan for pada baris ke-358 sampai baris ke-380 memakai double penampung untuk menampung hasil volume empat

bangun ruang kemudian di sorting yang hasil nya nanti terurut secara ascending. Dan di akhiri dengan *break*; baris ke-383 untuk mengakhiri dari *case 5*



```
408 //pilihan perhitungan luas
409 static void luas(){
410     int daftar_bangun;
411     String bangun_datar[]={"luas persegi","luas persegi panjang","luas segitiga","luas seluruh bangun datar","Keluar"};
412     System.out.println();
413     System.out.println();
414     System.out.println("1. bangun_datar[0]);
415     System.out.println("2. bangun_datar[1]);
416     System.out.println("3. bangun_datar[2]);
417     System.out.println("4. bangun_datar[3]);
418     System.out.println("5. bangun_datar[4]);
419     //System.out.println("6. bangun_datar[5]);
420     //System.out.println("7. bangun_datar[6]);
421
422     Scanner pilih2=new Scanner(System.in);
423     int menu3;
424
425     System.out.println();
426     System.out.print("Silahkan Pilih Bangun Ruang Yang Akan Dilakukan Perhitungan : ");
427     menu3=pilih2.nextInt();
428
429     switch(menu3){
430
431         case 1:
432             double luaspp,sisi;
433             Scanner terima=new Scanner(System.in);
434             System.out.println("");
435             System.out.println("=====");
436             System.out.println("Program menghitung luas persegi");
437             System.out.println("=====");
438             System.out.println();
439             System.out.print("Silahkan Masukkan sisi = ");
440             sisi=terima.nextDouble();
441             luaspp=(sisi*sisi);
442             System.out.println();
443             System.out.println("luas persegi = "+luaspp);
444             System.out.println();
445             next();
446             break;
447
448         case 2:
449             double luaspp,panjang,lebar;
```

Gambar 13.11 Aplikasi perhitungan luas bangun datar

Program diatas adalah program untuk menghitung luas bangun datar, jika user memilih menu dua maka akan tampil perhitungan luas,dimana terdapat tiga bangun datar , disini menggunakan static void luas untuk awal kelas nya dan dilanjutkan pada baris ke-411 pendeklarasian array dengan tipe data string. Switch case dimulai pada baris ke-429 . case 1 dimulai pada baris ke-431,baris ke-432 pendeklrasian tipe data yang digunakan untuk perhitungan luas persegi,pada baris ke-433 menggunakan scanner new untuk menginput variabel lalu terdapat operator aritmatika yaitu pada baris ke-441 untuk rumus matematika.pada baris 446 break untuk mengakhiri program dan dilanjutkan dengan case seterusnya.

```

562     case 5 :
563         System.out.println();
564         System.out.println("*****");
565         System.out.println("*****");
566         System.out.println("    TERIMA KASIH TELAH    ");
567         System.out.println("    MEMAKAI APLIKASI KAMI ");
568         System.out.println("    SISTEM AKAN DITUTUP ");
569         System.out.println("    BYEE                ");
570         System.out.println("*****");
571         System.out.println("    ");
572         System.out.println();
573         System.exit(0);
574     default :
575
576         System.out.println();
577         System.out.print("Maaf Pilihan No. "+menu3+" Tidak Ada Di Sistem");
578         System.out.println();
579         System.out.println("Sistem Akan Keluar");
580         System.exit(0);
581     }
582 }
583
584
585 static void next(){
586
587     Scanner lagi=new Scanner(System.in);
588     int ulang;
589     System.out.println();
590     System.out.print("Apakah Anda Akan Melakukan Perhitungan Lagi ? ( 1/0 ) = ");
591     ulang=lagi.nextInt();
592     if(ulang==1){
593         luas();
594     }else{
595         penutup();
596     }
597 }
598 }

```

Gambar 14.12 Gambar pemilihan

Program pemilihan dimana pada baris ke-592 jika anda mengetik satu maka akan tampil menu utama saat program pertama kali dijalankan namun jika anda mengetik 0 maka menu perhitungan akan berhenti dan akan tampil terimakasih telah memakai aplikasi kami sistem akan ditutup bye.

```

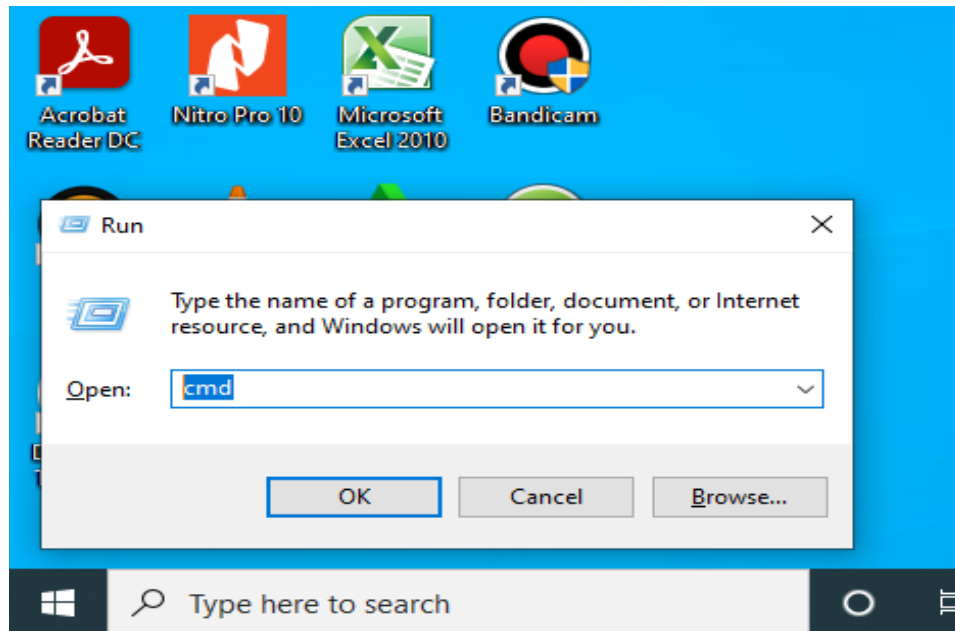
599 static void penutup(){
600
601     System.out.println();
602     System.out.println("Terima Kasih Telah Menggunakan Aplikasi Sederhana Kami");
603     System.out.println("    A  A    ");
604     System.out.println("    A  A    ");
605     System.out.println("    A  v  A  ");
606     System.out.println("    A  A    ");
607     System.out.println("    A  A    ");
608     System.out.println(" >SEMANGAT PETUNJUK< ");
609     System.out.println();
610 }
611
612 }
613
614
615

```

Gambar 15.13 Akhir Program

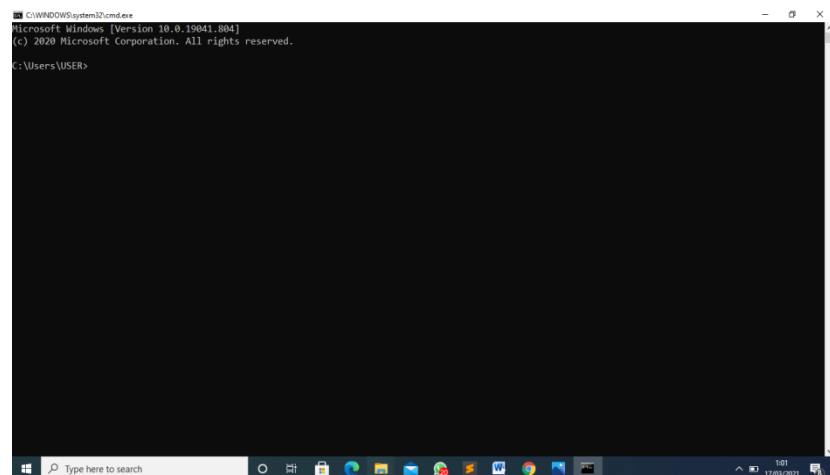
Program diatas adalah static void penutup dari seluruh program jadi diakhir nanti jika anda telah selesai melakukan perhitungan maka akan tampil perintah terimakasih telah menggunakan aplikasis sederhana kami dan muncul emoji seperti diatas, baris ke-613 adalah kurung kurawal tanda program telah selesai.

C. Hasil Compaile:



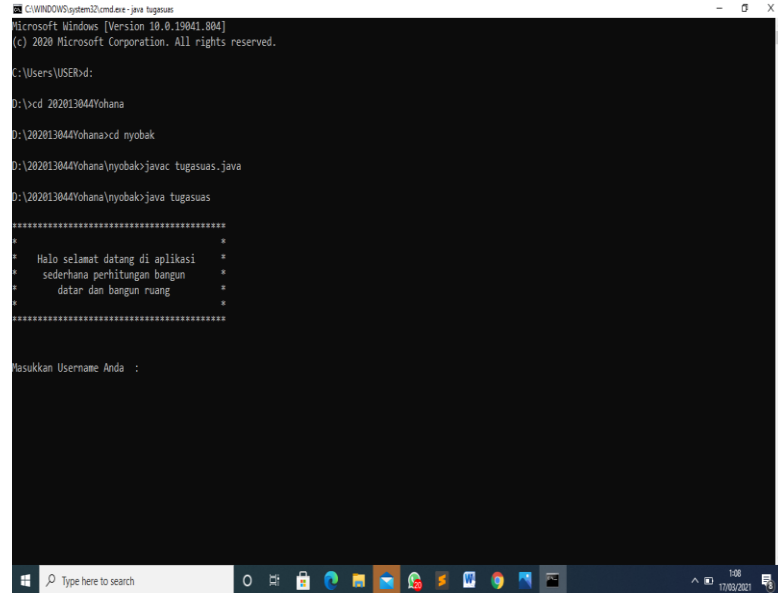
Gambar 16.14 Membuka tampilan cmd

Langkah berikutnya kita masuk ke cmd untuk melihat hasil program. Ketik windows+R > ketik cmd > enter. Maka, akan tampil seperti gambar dibawah samping sebelah kanan.



Gambar 17.15 memanggil directory D

Ketikan nama *directory* tempat menyimpan file program lalu ketik javac beserta namafile(tugasuas) kemudian enter.



```
C:\WINDOWS\system32\cmd.exe - java tugasuas
Microsoft Windows [Version 10.0.19041.804]
(c) 2020 Microsoft Corporation. All rights reserved.

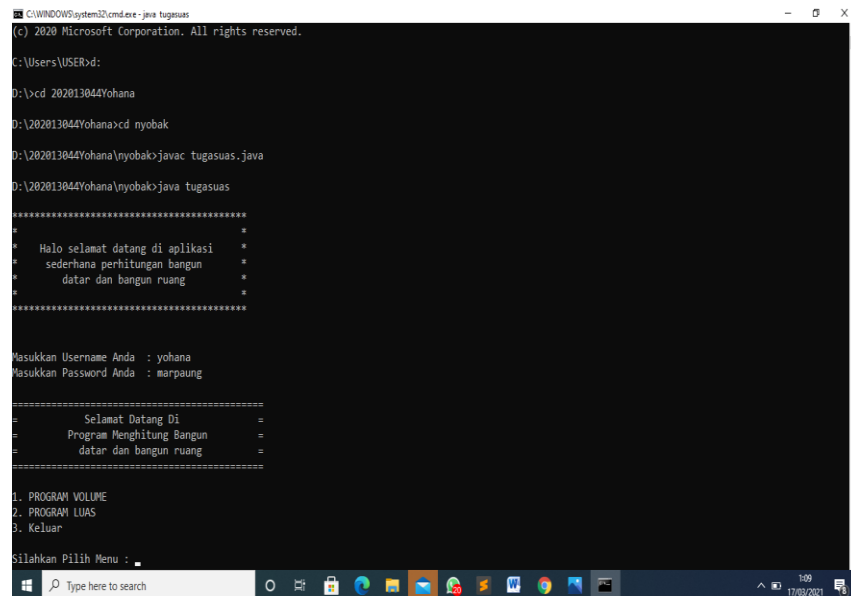
C:\Users\USER>cd
D:\>cd 202013044Yohana
D:\202013044Yohana>cd nyobak
D:\202013044Yohana\nyobak>javac tugasuas.java
D:\202013044Yohana\nyobak>java tugasuas

*****
*      Halo selamat datang di aplikasi      *
*      sederhana perhitungan bangun          *
*      datar dan bangun ruang                *
*****

Masukkan Username Anda :
```

Gambar 18.16 memanggil directory D

Maka akan tampil tampilan form login seperti atas jika program anda jalan atau berhasil di compile.



```
C:\WINDOWS\system32\cmd.exe - java tugasuas
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\USER>cd
D:\>cd 202013044Yohana
D:\202013044Yohana>cd nyobak
D:\202013044Yohana\nyobak>javac tugasuas.java
D:\202013044Yohana\nyobak>java tugasuas

*****
*      Halo selamat datang di aplikasi      *
*      sederhana perhitungan bangun          *
*      datar dan bangun ruang                *
*****

Masukkan Username Anda : yohana
Masukkan Password Anda : marpaung

*****
*      Selamat Datang Di                    *
*      Program Menghitung Bangun            *
*      datar dan bangun ruang                *
*****

1. PROGRAM VOLUME
2. PROGRAM LUAS
3. Keluar

Silahkan Pilih Menu :
```

Gambar 19.17 Form Login

Tampilan diatas jika anda berhasil memasukkan username dan password yang benar maka akan tampil menu.


```
C:\WINDOWS\system32\cmd.exe - java tugasuas
=====
Selamat Datang Di
Program Menghitung Bangun
datar dan bangun ruang
=====

1. PROGRAM VOLUME
2. PROGRAM LUAS
3. Keluar

Silahkan Pilih Menu : 1

1. Volume Kubus
2. Volume Balok
3. Volume Kerucut
4. Volume Prisma Segiempat
5. Semua Bangun Ruang
6. Keluar

Silahkan Pilih Bangun Ruang Yang Akan Dilakukan Perhitungan : 1

=====
Menghitung Volume Kubus
=====

Masukkan Sisi = 2
Volume Kubus Adalah 8.0

Apakah Anda Akan Melakukan Perhitungan Lagi ? ( 1/0 ) =
```

Gambar 20.18 Tampilan Menu

Tampilan diatas jika anda memilih menu 1 maka akan tampil pilihan bangun ruang yang dapat dihitung volume nya , saya memilih bangun ruang satu yaitu kubus dan selanjutnya kita harus menginput panjang sisi setelah itu hasil perhitungan nya akan keluar.

```
C:\WINDOWS\system32\cmd.exe - java tugasuas
Apakah Anda Akan Melakukan Perhitungan Lagi ? ( 1/0 ) = 1

1. luas persegi
2. luas persegi panjang
3. luas segitiga
4. luas seluruh bangun datar
5. Keluar

Silahkan Pilih Bangun Ruang Yang Akan Dilakukan Perhitungan : 1

#####
"Program menghitung luas persegi"
#####

Silahkan Masukkan sisi = 3
luas persegi = 9.0

Apakah Anda Akan Melakukan Perhitungan Lagi ? ( 1/0 ) =
```

Gambar 21.19 Program Luas

Jika anda memilih menu luas maka akan tampil seperti diatas selanjutnya saya memilih perhitunngan bangun datar 1 yaitu persegi selanjutnya anda dihaaruskan

menginput panjang sisi persegi dan hasil perhitungan nya aan keluar,serta akan muncul perintah apakah anda ingin melakukan perhitungan lagi? Jika yam aka ketik 1 jika tidak ketik 0

```

C:\WINDOWS\system32\cmd.exe - java tugasuas
1. luas persegi
2. luas persegi panjang
3. luas segitiga
4. luas seluruh bangun datar
5. Keluar

Silahkan Pilih Bangun Ruang Yang Akan Dilakukan Perhitungan : 4

#####
*Program menghitung luas persegi*
#####

Silahkan Masukkan sisi 1= 3

luas persegi 1= 9.0

#####
*Program menghitung luas persegi panjang*
#####

Silahkan Masukkan panjang 1= 4
Silahkan masukkan lebar 1= 2

luas persegi panjang 1= 8.0

#####
*Program menghitung luas segitiga*
#####

Silahkan Masukkan alas 1 = 3
Silahkan Masukkan Tinggi 1 = 2

```

```

#####
*Program menghitung luas persegi panjang*
#####

Silahkan Masukkan panjang 1= 4
Silahkan masukkan lebar 1= 2

luas persegi panjang 1= 8.0

#####
*Program menghitung luas segitiga*
#####

Silahkan Masukkan alas 1 = 3
Silahkan Masukkan Tinggi 1 = 2

luas segitiga 1 = 3.0

1 = 8.0 3.0 9.0
2 = 3.0 8.0 9.0
3 = 3.0 8.0 9.0

Data pada array hasil[] setelah di urutkan = {3.0 , 8.0 , 9.0 , }

Apakah Anda Akan Melakukan Perhitungan Lagi ? ( 1/0 ) =

```

Gambar 22.20 Hasil Sorting

Selanjutnya saya memilih untuk melakukan perhitungan lagi,lalu saya memilih menu luas dan memilih menu untuk mencopai semua bangun serta hasil sorting nya ,terlihat bahwa hasil semua luas disoritng diakhir dan hasilnya tersusun secara ascending.

```

C:\WINDOWS\system32\cmd.exe
D:\202013044Yohana\nyobak>java tugasuas

*****
*      Halo selamat datang di aplikasi      *
*      sederhana perhitungan bangun         *
*      datar dan bangun ruang              *
*****

Masukkan Username Anda : yohana
Masukkan Password Anda : marpaung

=====
=      Selamat Datang Di                    =
=      Program Menghitung Bangun           =
=      datar dan bangun ruang              =
=====

1. PROGRAM VOLUME
2. PROGRAM LUAS
3. Keluar

Silahkan Pilih Menu : 3

KAMU KELUAR DARI SISTEM TERIMA KASIH SUDAH MENGGUNAKAN APLIKASI INI

D:\202013044Yohana\nyobak>

```

Gambar 23.21 Tampilan Menu Ketiga

Nah diatas adalah hasil program jika anda memilih menu 3 maka tampilan nya seperti diatas artinya anda telah keluar dari aplikasi.

```

C:\WINDOWS\system32\cmd.exe

1. Volume Kubus
2. Volume Balok
3. Volume Kerucut
4. Volume Prisma Segiempat
5. Semua Bangun Ruang
6. Keluar

Silahkan Pilih Bangun Ruang Yang Akan Dilakukan Perhitungan : 1

=====
=      Menghitung Volume Kubus            =
=====

Masukkan Sisi = 3

Volume Kubus Adalah 27.0
=====

Apakah Anda Akan Melakukan Perhitungan Lagi ? ( 1/0 ) = 0

Terima Kasih Telah Menggunakan Aplikasi Sederhana Kami
  A  A
  A  A
  A * v * A
  A  A
  A  A
>SEMANGAT PEJUANG>

D:\202013044Yohana\nyobak>

```

Gambar 24.22 Tampilan Akhir Program

Hasil compaile diatas adalah hasil compaile jka program telah selesai dan anda tidak ingin melakukan perhitungan lagi.

BAB IV

PENUTUP

1.Kesimpulan

Dari tahapan yang telah dilakukan dalam pembuatan program Aplikasi Menghitung Luas Bangun Datar dan Volume bangun ruang, maka penulis menyimpulkan beberapa hal sebagai berikut :

- a. Dalam pemograman aplikasi menghitung luas bangun datar dan volume bangun ruang ini telah berhasil untuk menyelesaikan tugas projek ujian akhir semester 1 mata kuliah Algoritma dan Pemograman .
- b. Program ini menggunakan bahasa *Java* dan perintah berupa:
 1. Daftar *header file*
 2. Tipe data
 - a. *Integer*
 - b. *Double*
 3. Operator Aritmatika
 4. Perulangan *For*
 5. Pemilihan *Switch Case*
 6. Percabangan *if else*
 7. Variabel
 8. *Bubble Sort* (pengurutan)
- c. Program ini dapat mempermudah dalam menghitung program dalam satu perintah.
- d. Program ini menyediakan beberapa menu untuk mempermudah proses perhitungan
- e. Program ini terdiri atas prodesur yang didalam nya terdapat perintah pemilihan dan perulangan.

2.Saran

Saran yang dapat diberikan penulis untuk pengembangan program perhitungan bangun datar dan bangun ruang ini selanjutnya dimasa yang akan datang adalah :

- a. Pada program aplikasi menghitung luas bangun datar dan volume bangun ruang ini belum menggunakan data base sehingga program belum bisa menyimpan data dengan benar.
- b. Program ini juga dapat dibuat dengan keefektivan dari penggunaan fungsi dalam program.

DAFTAR PUSTAKA

AF Sallaby, F. U. (2015). Aplikasi Widget berbasis java. *Jurnal Media Infotama*.

Enterprise, J. (2019). *Lancar Java dan Javascript*. PT Elex Media Komputindo, Jakarta: PT Elex Media Komputindo.

Maulana, M. R. (2017). Komparasi Algoritma Sorting pada Bahasa Pemograman Java. *STMIK Widya Pratama Pekalongan*, 37.