

LAPORAN PRAKTIKUM
PROGRAM APLIKASI SEDERHANA JAVA
UJIAN AKHIR SEMESTER I



Disusun Oleh:

HENDRIANTO
NIM 202013031

DOSEN PENGAMPU

SLAMET TRIYANTO, S.ST

PROGRAM STUDI TEKNIK
INFORMATIKAPOLITEKNIK KAMPAR

2021

KATA PENGANTAR

Puji dan syukur penulis ucapkan kepada Tuhan Yang Maha Esa yang sudah memberikan karunia dan rahmat-Nya sehingga penulis dapat melaksanakan tugas praktikum Algoritma Pemrograman ini. Sehingga tersusunlah materi laporan praktikum ini yang berjudul –Program Aplikasi Sederhana Java, berdasarkan hasil praktek yang sudah dilakukan oleh penulis.

Dengan selesainya praktikum ini, maka tidak lupa penulis ucapkan trimakasih kepada semua pihak yang telah membantu penulis dan khususnya penulis ucapkan kepada:

1. Bapak Slamet Triyanto, S.ST selaku Dosen Mata Kuliah Algoritma Pemrograman.
2. Dan kepada teman-teman seperjuangan Teknik Informatika di Politeknik Kampar angkatan 2020 yang telah memberi semangat.

Demikian laporan yang penulis buat, mohon kritik dan sarannya atas kekurangan dalam penyusunan laporan ini. Semoga laporan ini dapat bermanfaat bagi semua pihak dan bagi penulis sendiri. Trimakasih.

Bangkinang, 12 Maret 2021

Penulis

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI.....	ii
DAFTAR GAMBAR.....	iv
DAFTAR TABEL.....	v
BAB I.....	1
TINJAUAN PUSTAKA	1
PEMBAHASAN	4
A. Struktur Bahasa C	4
1. Header File Java	4
2. Deklarasi.....	4
3. Deskripsi.....	4
B. Variabel.....	4
C. Operator.....	6
1. Operator Aritmatika.....	6
2. Operator <i>Bitwise</i>	7
3. Operator Penugasan.....	8
4. Operator Perbandingan.....	9
5. Operator Logika	10
D. Deklarasi Fungsi.....	10
E. Input	10
1. <i>Class Scanner</i>	10
2. <i>Class BufferedReader</i>	11
3. <i>Class Concole</i>	11
F. Ouput.....	11
Fungsi print() vs println().....	12
G. Penyeleksian Kondisi	12
1. Struktur Kondisi -If 	12
2. Struktur Kondisi -If Else 	13
3. Struktur Kondisi -Switch Case Default 	14
BAB III.....	15
LANGKAH KERJA MENJALANKAN PROGRAM.....	15
A. Instalasi Java Jdk.....	15

1. Download java Jdk	15
2. Mengekstrak MinGW	15
3. Instal MinGw	15
B. Proses Source Code	18
C. Penjelasan <i>Source Code</i>	22
BAB IV	26
PENUTUP.....	26
A. Kesimpulan	26
B. Saran	26
DAFTAR PUSTAKA	27

DAFTAR GAMBAR

Gambar 2. 1 Penggunaan IF.....	13
Gambar 3. 1 Tampilan Folder Jdk.....	15
Gambar 3. 2 Tampilan Folder Bin	16
Gambar 3. 3 Tampilan System.....	16
Gambar 3. 4 Tampilan Environment Variabel	17
Gambar 3. 5 Tampilan Versi Java.....	17
Gambar 3. 6 Membuka Software Visual Code	18
Gambar 3. 7 Membuat Source Code	18
Gambar 3. 8 Save Source Code.....	19
Gambar 3. 9 Save As.....	19
Gambar 3. 10 All File.....	19
Gambar 3. 11 Membuka CMD.....	20
Gambar 3. 12 Melihat File pada Data C.....	20
Gambar 3. 13 Membuka Folder	20
Gambar 3. 14 Compile Program	21
Gambar 3. 15 Tampilan Awal dan Pemilihan Program	21
Gambar 3. 16 Tampilan Yang Telah di Pilih	21
Gambar 3. 17 Header	22
Gambar 3. 18 Deklarasi.....	22
Gambar 3. 19 Hello Word.....	22
Gambar 3. 20 Tipe Data	23
Gambar 3. 21 Deskripsi.....	23
Gambar 3. 22 Menghitung Luas Bangun Datar	24
Gambar 3. 23 Switch Case	24
Gambar 3. 24 Pengurutan Data	25

DAFTAR TABEL

Tabel 2. 1 Operator Aritmatika	7
Tabel 2. 2 Operator Bitwise	7
Tabel 2. 3 Operator Penugasan	8
Tabel 2. 4 Perbandingan.....	9
Tabel 2. 5 Operator Logika	10

BAB I

TINJAUAN PUSTAKA

A. Pengertian Bahasa Java

Dalam perkembangan teknologi informasi sekarang ini telah banyak ditemukan penggunaan aplikasi yang berbasis Java pada beberapa perangkat mobile device yang bersifat innovative. Java adalah bahasa pemrograman berorientasi objek yang dapat digunakan untuk pengembangan aplikasi mandiri, aplikasi berbasis internet maupun intranet serta aplikasi untuk perangkat-perangkat cerdas yang dapat berkomunikasi lewat Intranet/jaringan komunikasi. Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di Sun Microsystems saat ini merupakan bagian dari Oracle dan dirilis tahun 1995.

B. Kelebihan dan Kekurangan Bahasa Java

1. Berikut beberapa kelebihan bahasa pemrograman Java :

a. Mudah untuk dikembangkan

Kelebihan dari Java yaitu kemudahan dalam hal pengembangan aplikasi. Setiap aplikasi maupun program yang dibuat dengan menggunakan dasar bahasa pemrograman Java mempunyai kemampuan yang sangat baik untuk dilakukan pengembangan lebih lanjut. Hal ini akan sangat membantu para programmer-programmer dan developer untuk lebih baik lagi dalam mengembangkan satu aplikasi yang berbasis Java.

b. Sifatnya multiplatform

Bahasa pemrograman java yang pada saat ini banyak diminati oleh para developer dan programmer adalah Java karena Java menjadi salah satu bahasa pemrograman yang sifatnya multiplatform, alias universal dan dapat digunakan dalam platform apapun. Hal ini membuat banyak sekali para pengembang aplikasi yang menggunakan basis bahasa pemrograman Java ini untuk membuat aplikasi yang diinginkan oleh programmer tersebut.

- c. Java sangat mendukung programmer dalam berorientasi pada usability

Keunggulan bahasa pemrograman java yang satu ini sangat berhubungan dengan kemampuan aplikasi – aplikasi yang dibuat dengan menggunakan atau berbasis Java yang mampu bekerja di platform manapun. Hal ini berhubungan dengan usability, atau kegunaan dari suatu aplikasi.

- d. Bahasa pemrograman yang berorientasi terhadap objek

Selain itu, bahasa pemrograman Java yang satu ini lebih bersifat teknis. Bahasa pemrograman Java adalah salah satu bentuk atau jenis bahasa pemrograman yang berorientasi terhadap objek. Itu artinya setiap aplikasi yang dibuat dengan menggunakan bahasa pemrograman java akan disesuaikan dengan objek atau dapat juga dengan tampilan dan interface dari aplikasi tersebut.

- e. Dinamis

Kelebihan dari bahasa pemrograman Java yang lainnya yaitu sifatnya yang dinamis. Sifat dinamis dari bahasa pemrograman Java ini sangat berkaitan dengan kemampuan dari bahasa pemrograman Java yang sangat mudah untuk dikembangkan. Struktur kodenya dapat dengan mudah dimodifikasi dan dikembangkan, sesuai dengan kebutuhan dari user. Meskipun nampaknya bahasa pemrograman Java memiliki banyak kelebihan, namun Java tetap tidak luput dari kekurangan.

- 2. Berikut beberapa kekurangan bahasa pemrograman Java adalah :

- a. Penggunaan memory yang cukup tinggi

Bahasa pemrograman Java memang banyak sekali menawarkan fitur-fitur yang luar biasa, baik itu kemudahan dalam Menyusun script, hingga fitur object oriented, yang menjadi salah satu ciri khas dari bahasa pemrograman Java. Namun sayangnya, kelebihan tersebut harus dikompensasi dengan kebutuhan memory yang besar.

Hal ini menjadikan beberapa aplikasi Java membutuhkan resource memory yang cukup besar untuk bisa berjalan dengan baik. Begitu pula pada saat seorang developer akan mengembangkan aplikasi Java, developer juga membutuhkan memory yang besar dalam mengembangkan aplikasi ini. Itu berarti developer membutuhkan komputer yang memiliki kapasitas ram yang besar untuk melancarkan develop agar lebih optimal.

b. Mudah didekompilasi

Istilah dekompilasi ini adalah pengambilan source code. Jadi, Java menjadi salah satu bahasa pemrograman yang mudah mengalami dekompilasi. Source code dan juga script yang dibuat dengan bahasa pemrograman Java bisa dengan mudah diambil dan juga dibajak oleh orang lain, sehingga menimbulkan banyak hack dan juga pengcopyan dari aplikasi yang menggunakan bahasa pemrograman Java tersebut.

C. Tujuan Praktikum

Praktikum ini memiliki beberapa tujuan yaitu :

1. Agar mahasiswa lebih memahami tentang bahasa Java.
2. Agar mahasiswa memahami dasar-dasar bahasa Java untuk membuat program menghitung volume bangun ruang.
3. Untuk memahami *source code* yang digunakan pada perhitungan volume bangun ruang.

D. Alat dan Bahan

1. Alat
 - a. Laptop
 - b. Mouse
2. Bahan
 - a. Visual Studio Code
 - b. CMD
 - c. Rumus-rumus Matematika

BAB II

PEMBAHASAN

A. Struktur Bahasa C

Java merupakan bahasa pemrograman yang bersifat umum/non-spesifik (*general purpose*), dan secara khusus di disain untuk memanfaatkan dependensi implementasi seminimal mungkin.

1. Header File Java

Pada bagian ini, kita melakukan impor library yang dibutuhkan pada program. Library merupakan sekumpulan *class* dan fungsi yang bisa kita gunakan dalam membuat program.

Contoh impor library:

```
Import java.util.Scanner;
```

Pada contoh tersebut, kita mengimpor class Scanner dari package java.util.Scanner;

2. Deklarasi

Deklarasi adalah bagian untuk mendefinisikan semua nama yang dipakai dalam program. Nama tersebut dapat berupa nama variabel, nama tipe, nama prosedur, nama fungsi.

3. Deskripsi

Bagian inti dari suatu program yang berisi uraian langkah-langkah penyelesaian masalah. Program Java pada hakekatnya tersusun atas sejumlah blok fungsi. Sebuah program minimal mengandung sebuah fungsi. Setiap fungsi terdiri dari satu atau beberapa pernyataan, yang secara keseluruhan dimaksudkan untuk melaksanakan tugas khusus. Bagian pernyataan fungsi (disebut tubuh fungsi) diawali dengan tanda `--` (kurung kurawal buka) dan diakhiri dengan tanda `--` (kurung kurawal tutup).

B. Variabel

Variabel dalam program digunakan untuk menyimpan suatu nilai tertentu dimana nilai tersebut dapat berubah-ubah. Setiap variabel mempunyai tipe dan

hanya data yang bertipe sama dengan tipe variabel yang dapat disimpan di dalam variabel tersebut. Setiap variabel mempunyai nama. Pemisahan antar variabel dilakukan dengan memberikan tanda koma. *Syntax*:
<nama_tipe_data> <nama_variabel>.

Contoh :

```
int jumlah;  
float harga_per_unit, total_biaya;
```

Aturan Penamaan Variabel dalam Bahasa Java

Penamaan **variabel** merujuk ke aturan identifier yang pernah kita bahas beberapa tutorial sebelumnya. Berikut aturan penamaan variabel di dalam bahasa pemrograman java :

1. Nama variabel tidak boleh menggunakan kata kunci dari Java (*reserved word*) seperti if, for, switch, dll.
2. Nama variabel boleh menggunakan huruf, angka (0-9), garis bawah (*underscore*), dan symbol dollar (\$), namun penggunaan garis bawah dan symbol lebih baik dihindari.
3. Nama variabel harus diawali dengan huruf kecil, karena Java menggunakan gaya CamelCase.
4. Apabila nama variabel lebih dari 1 suku kata, maka kata ke-2 dituliskan dengan diawali dengan huruf besar dan seterusnya, contoh namaVariabel.

Cara Penulisan Variabel di dalam bahasa Java.

Ketika akan menulis variabel, dalam hampir semua bahasa pemrograman terdapat 2 proses: deklarasi dan inisialisasi.

Deklarasi adalah proses untuk memberitahukan *compiler* bahasa java bahwa kita akan membuat sebuah variabel. Bahasa java termasuk bahasa pemrograman yang menggunakan konsep *strongly typed programming language*, yang artinya untuk setiap variabel harus ditulis akan berisi tipe

data apa. Apakah itu angka bulat (*integer*), angka pecahan (*float*), huruf (*char*), atau yang lain. Ketika ingin membuat variabel, harus ditentukan dulu apa tipe data dari variabel tersebut. Pembahasan mengenai tipe data di dalam bahasa java akan kita bahas dalam tutorial tersendiri, namun berikut tipe data yang sering dipakai:

1. Tipe data *integer*, yakni tipe data angka bulat seperti 1, 5 atau 1000. Tipe data *integer* ditulis dengan *keyword* *int*.
2. Tipe data *float*, yakni tipe data angka pecahan seperti 1.33, 5.90 atau 1000.99. Tipe data *float* ditulis dengan *keyword* *float*.
3. Tipe data *character*, yakni tipe data huruf seperti `'A'`, `'a'`, atau `'Z'`. Tipe data *character* ditulis dengan *keyword* *char*.

C. Operator

Operator dan Operand dalam Bahasa java merupakan sebuah istilah yang tidak asing lagi dalam pemrograman. Operator adalah simbol atau karakter yang digunakan oleh program untuk melakukan sebuah operasi dalam sebuah proses program seperti operasi bilangan dan operasi string. Bahasa java mengenal penggunaan beberapa operator dengan fungsi yang berbeda-beda. Setiap operator memiliki kedudukan atau hirarki saat penanganan program. Operator dengan hirarki lebih tinggi akan dikerjakan lebih dahulu dibandingkan operator dengan hirarki lebih rendah. Ada beberapa operator dalam bahasa java dan berikut contoh penggunaannya :

1. Operator Aritmatika

Operator Aritmatika adalah operator yang digunakan untuk mengoperasikan bilangan *integer*, *float* dan lain sebagainya. Biasanya operatornya berupa penjumlahan, pengurangan, pembagian, perkalian, *increment*, *decrement* dan modulus. *Increment* merupakan operasi untuk menambahkan *value* sejumlah 1 sedangkan *decrement* akan mengurangi *value* sejumlah 1.

Tabel 2. 1 Operator Aritmatika

Operator	Nama	Contoh	Hasil
+	Pemjumlahan	$a = b + c$	b ditambah dengan c
-	Pengurangan	$a = b - c$	b dikurang dengan c
*	Perkalian	$a = b * c$	b dikali dengan c
/	Pembagian	$a = b / c$	b dibagi dengan c
%	Modulus	$a = b \% c$	sisir dari b dibagi c
++	Increment	$a++$	$a = a + 1$
--	Decrement	$a--$	$a = a - 1$

2. Operator *Bitwise*

Adalah operator yang menangani operasi bilangan biner seperti *And*, *Or*, *Not* dan sebagainya. Operator *bitwise* ini akan menangani data sesuai dengan tipenya. Misalnya sebuah data bertipe *char* atau *byte* maka bilangan yang dihasilkan adalah sebesar 8 bit.

Tabel 2. 2 Operator *Bitwise*

Operator	Nama	Contoh	Biner	Hasil Biner	Hasil Decimal
&	AND	$x = 5 \& 1$	0101 & 0001	0001	1
	OR	$x = 5 1$	0101 0001	0101	5
~	NOT	$x = \sim 5$	~0101	1010	10

\wedge	XOR	$x = 5 \wedge 1$	$0101 \wedge 0001$	0100	4
\ll	Left shift	$x = 5 \ll 1$	$0101 \ll 1$	1010	10
\gg	Right shift	$x = 5 \gg 1$	$0101 \gg 1$	0010	2

3. Operator Penugasan

Adalah operator yang digunakan untuk memberi nilai pada sebuah variabel. Operator penugasan yang paling dasar adalah sama dengan ($=$). Dari operator ini dapat dikembangkan beberapa operator penugasan lain seperti $+=$, $-=$ dan sebagainya.

Tabel 2. 3 Operator Penugasan

Operator	Penugasan	Samadengan	Deskripsi
$=$	$x = y$	$x = y$	variabel x memperoleh nilai dari variabel y
$+=$	$x += y$	$x = x + y$	variabel x memperoleh nilai dari $x + y$
$-=$	$x -= y$	$x = x - y$	variabel x memperoleh nilai dari $x - y$
$*=$	$x *= y$	$x = x * y$	variabel x memperoleh nilai dari $x * y$
$/=$	$x /= y$	$x = x / y$	variabel x memperoleh nilai dari x / y
$\% =$	$x \% = y$	$x = x \% y$	variabel x memperoleh nilai dari $x \% y$

<<=	x <<= y	x = x << y	variabel x memperoleh nilai dari x << y
>>=	x >>= y	x = x >> y	variabel x memperoleh nilai dari x >> y
&=	x &= y	x = x & y	variabel x memperoleh nilai dari x & y
=	x = y	x = x y	variabel x memperoleh nilai dari x y
^=	x ^= y	x = x ^ y	variabel x memperoleh nilai dari x ^ y

4. Operator Perbandingan

Adalah operator yang digunakan untuk membandingkan dua buah nilai atau variabel. Nilai yang dibandingkan bisa berupa angka maupun string. Hasil dari perbandingan ini berupa nilai *boolean*, yaitu *true* (benar) atau *false* (salah).

Tabel 2. 4 Perbandingan

Operator	Nama	Contoh	Hasil
==	Sama dengan	a == b	benar jika a sama dengan b
!=	Tidak sama dengan	a != b	benar jika a berbeda dengan b
>	Lebih besar	a > b	benar jika a lebih besar dari b
<	Lebih kecil	a < b	benar jika a lebih kecil dari b
>=	Lebih besar atau	a >= b	benar jika a lebih besar atau

	sama dengan		sama dengan b
<=	Lebih kecil atau sama dengan	a <= b	benar jika a lebih kecil atau sama dengan b

5. Operator Logika

Adalah operator yang digunakan untuk menangani tipe data *boolean*.

Nilai data *boolean* bisa berupa kondisi benar (*true*) atau salah (*false*) dan bisa juga 1 atau 0.

Tabel 2. 5 Operator Logika

Operator	Nama	Contoh	Hasil
&&	And	a && b	Benar jika a and b bernilai benar
	Or	a b	Benar jika salah satu a atau b bernilai benar
!	Not	!a	Benar jika a tidak benar

D. Deklarasi Fungsi

Fungsi merupakan bagian yang terpisah dari program dan dapat diaktifkan atau dipanggil di manapun di dalam program. Bentuk umum deklarasi sebuah fungsi adalah :

Tipe_fungsi nama_fungsi(parameter_fungsi); Contohnya :

float luas_lingkaran(int jari); void tampil(); int tambah(int x, int y);

E. Input

Dalam bahasa java proses memasukkan suatu data bisa menggunakan beberapa fungsi pustaka yang telah tersedia. Beberapa fungsi pustaka yang bisa digunakan adalah :

1. Class Scanner

Scanner merupakan *class* yang menyediakan fungsi-fungsi untuk mengambil input dari keyboard. Agar kita bisa menggunakan *Scanner*, kita perlu mengimpornya ke dalam kode: Perlu diperhatikan, penggunaan fungsi untuk mengambil data bergantung dari tipe data yang digunakan. Misal, tipe datanya adalah *String*, maka fungsi atau *method* yang dipakai adalah `nextLine()`.

Begitu juga dengan tipe data lain, *Integer* menggunakan `nextInt()`, *Double* menggunakan `nextDouble()`, dsb.

2. *Class BufferedReader*

Class *BufferedReader* sebenarnya tidak hanya untuk mengambil input dari keyboard saja. *Class* ini juga dapat digunakan untuk membaca input dari file dan jaringan. *Class* ini terletak di dalam paket `java.io`.

Silahkan diimpor untuk dapat menggunakan *class* *BufferedReader*.

```
Import java.io.BufferedReader;
```

Perbedaan *BufferedReader* dengan *Scanner* terlihat dari fungsi atau *method* yang dipakai. *Scanner* menggunakan `next()`, sedangkan *BufferedReader* menggunakan `readLine()`. Lalu untuk tipe data *integer*, *BufferedReader* menggunakan fungsi `read()` saja.

3. *Class Concole*

Class *Console* hampir sama dengan *BufferedReader*. Dia juga menggunakan fungsi `readLine()` untuk mengambil input. Akan tetapi *Class* ini hanya bisa digunakan di lingkungan console saja, seperti Terminal dan CMD. *Class* *Console* tidak bisa digunakan langsung di Netbeans. Maka dari itu, kita harus kompilasi secara manual. Untuk menggunakan *class* ini, kita perlu mengimpornya terlebih dahulu.

F. Ouput

Kita sudah mengenal beberapa cara mengambil input dari keyboard untuk program berbasis teks. Sekarang bagaimana dengan outputnya?

Ada beberapa fungsi yang sudah disediakan oleh Java:

1. Fungsi `System.out.print()`
2. Fungsi `System.out.println()`

Fungsi `print()` vs `println()`

Fungsi `print()` dan `println()` sama-sama digunakan untuk menampilkan teks.

Lalu apa bedanya?

Fungsi `print()` akan menampilkan teks apa adanya. Sedangkan `println()` akan menampilkan teks dengan ditambah baris baru.

Mari kita coba dalam kode:

```
package eksperimen;

public class PrintVsPrinln {

    public static void main(String[] args) {

        System.out.print("ini teks yang dicetak dengan print()");

        System.out.println("sedangkan ini teks yang dicetak dengan println()");

        System.out.print("pake print() lagi");

    }

}
```

G. Penyeleksian Kondisi

Penyeleksian kondisi digunakan untuk mengarahkan perjalanan suatu proses. Penyeleksian kondisi dapat diibaratkan sebagai katup atau kran yang mengatur jalannya air. Bila katup terbuka maka air akan mengalir dan sebaliknya bila katup tertutup air tidak akan mengalir atau akan mengalir melalui tempat lain. Fungsi penyeleksian kondisi penting artinya dalam penyusunan bahasa C, terutama untuk program yang kompleks.

1. Struktur Kondisi -If

Struktur `if` dibentuk dari pernyataan `if` dan sering digunakan untuk menyeleksi suatu kondisi tunggal. Bila proses yang diseleksi terpenuhi atau bernilai benar, maka pernyataan yang ada di dalam blok `if` akan

diproses dan dikerjakan. Bentuk umum struktur kondisi if adalah : if

(kondisi) pernyataan;

contoh program : /* Program struktur kondisi if untuk memeriksa suatu kondisi */

```
if (condition_1) {  
    System.out.println("—Anda ingin makan");  
}  
else {  
    System.out.println("—Anda ingin minum"); }  

```

Gambar

2. 1 Penggunaan IF

Bila program tersebut dijalankan dan kita memasukan nilai 1, maka perintah mencetak perkataan ANDA INGIN MAKAN !!!! akan dilaksanakan, namun sebaliknya bila kita memasukan sebuah nilai yang lebih dari 1 seperti 2 sampai seterusnya maka program akan mencetak perkataan ANDA INGIN MINUM.

2. Struktur Kondisi -If Else

Dalam struktur kondisi if.....else minimal terdapat dua pernyataan. Jika kondisi yang diperiksa bernilai benar atau terpenuhi maka pernyataan pertama yang dilaksanakan dan jika kondisi yang diperiksa bernilai salah maka pernyataan yang kedua yang dilaksanakan. Bentuk umumnya adalah sebagai berikut :

```
if (condition_1) {  
    // Kode program yang dijalankan jika condition_1 berisi nilai True  
}  
else if (condition_2) {  
    // Kode program yang dijalankan jika condition_2 berisi nilai True  
}  
else {  
    // Kode program yang dijalankan jika semua kondisi tidak  
    terpenuhi  
}
```

3. Struktur Kondisi –Switch Case Default

Struktur kondisi switch....case....default digunakan untuk penyeleksian kondisi dengan kemungkinan yang terjadi cukup banyak. Struktur ini akan melaksanakan salah satu dari beberapa pernyataan `_case` tergantung nilai kondisi yang ada di dalam switch. Selanjutnya proses diteruskan hingga ditemukan pernyataan `_break`. Jika tidak ada nilai pada case yang sesuai dengan nilai kondisi, maka proses akan diteruskan kepada pernyataan yang ada di bawah `_default`. Bentuk umum dari struktur kondisi ini adalah :

```
Swich (kondisi){  
  Case 1 : pernyataan 1;  
    Break;  
  Case 2 : pernyataan 2;  
    Break ;  
  Case n : pernyataan n;  
    Break;  
  Default : Pernyataan m;  
    Break;  
}
```

BAB III

LANGKAH KERJA MENJALANKAN PROGRAM

A. Instalasi Java Jdk

Bahasa java merupakan bahasa tingkat tinggi yang mempunyai penyusun atau Compiler dan penerjemah Interpreter. Sehingga komputer mengerti apa yang dimaksud dengan bahasa C. MinGW adalah salah satu compiler bahasa c.

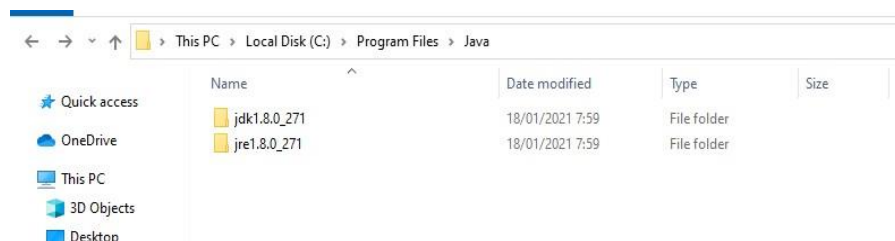
1. Download java Jdk
 - a. Nyalakan komputer atau laptop, setelah itu beri koneksi internet.
 - b. Buka browser dan carilah compiler java Jdk
 - c. Setelah itu download
2. Mengekstrak MinGW

Berikut adalah langkah untuk mengekstrak MinGw:

 - a. Setelah Java jdk berhasil terdownload lalu buka
 - b. Kemudian klik ekstrak to dan pilihlah lokasi penyimpanan kemudian klik -ok|. Setelah itu akan muncul file baru jdk yang berisi folder bin, include, lib, libexec.
3. Instal MinGw

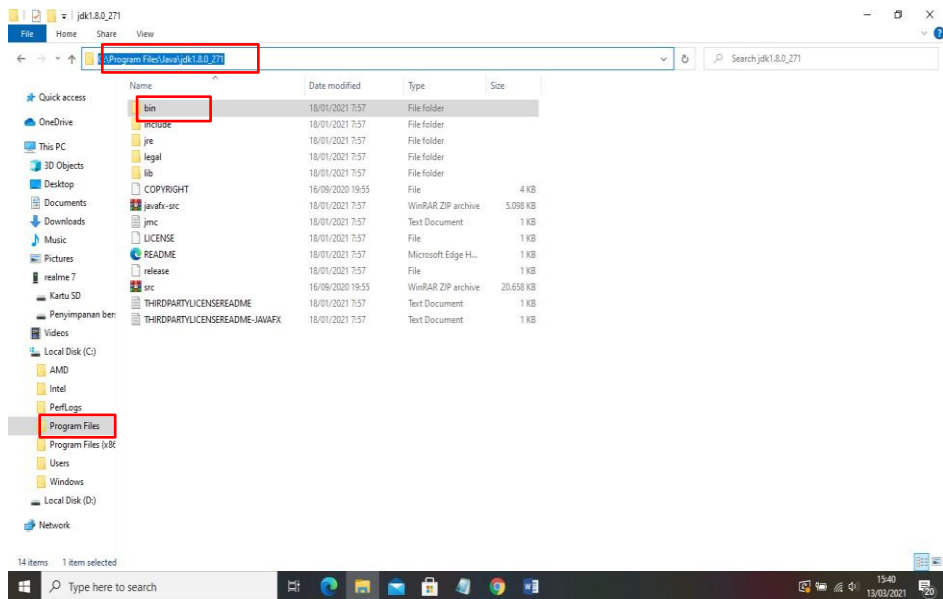
Berikut langkah penginstalan Java Jdk:

 - a. Langkah awal penginstalan Java Jdk adalah masuk ke file yang sudah di ekstrak contohnya di local disk C.



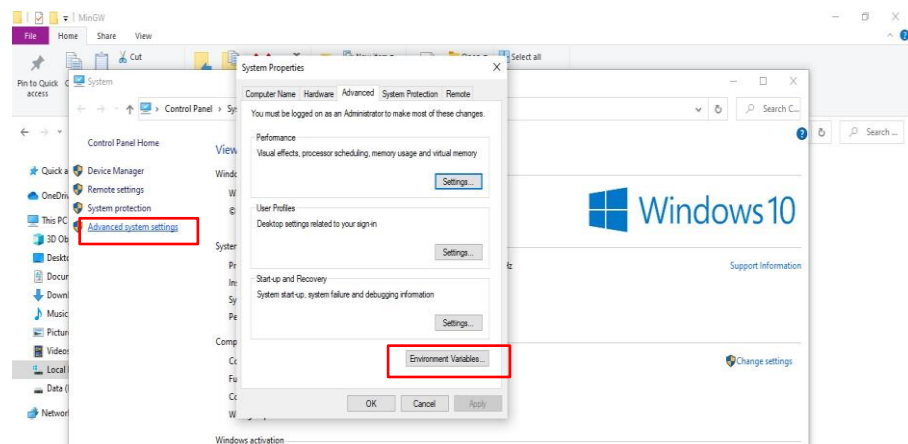
Gambar 3. 1 Tampilan Folder Jdk

- b. Setelah masuk pilih folder Java Jdk kemudian cari folder -bin| kemudian klik sekali dan copy link nya, seperti gambar dibawah ini.



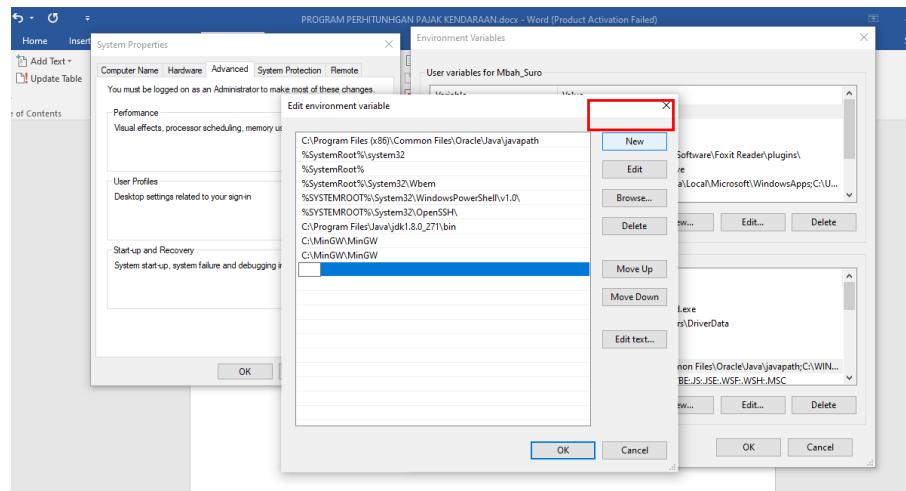
Gambar 3. 2 Tampilan Folder Bin

- c. Kemudian masuk ke this Pc kemudian klik kanan dan pilih *propertis*.
- d. Lalu pilih *Advanced System Setting*, kemudian pada sistem *propertis* pilih *-Environment Variabels*||.



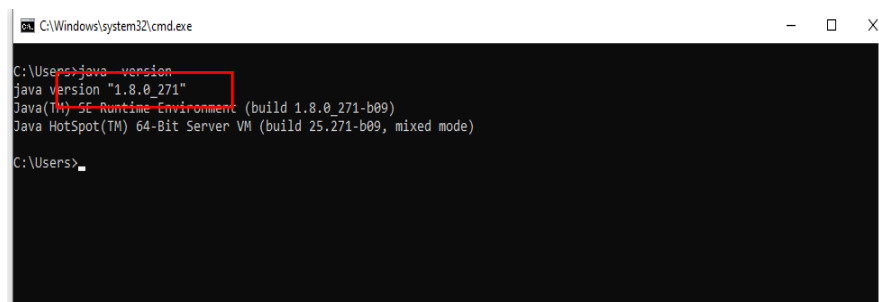
Gambar 3. 3 Tampilan *System*

- e. Kemudian lihat di sistem variabel cari *-Path* kemudian klik *new* lalu klik paste pada link yang sudah di *copy* kemudian OK.



Gambar 3. 4 Tampilan Environment Variabel

- f. Untuk menguji *compiler* nya sudah terinstal cek di cmd dengan tekan win+R kemudian masuk setelah itu ketik java spasi -version untuk melihat versi compiler yang sudah d instal, dari gambar di bawah versi Java adalah 1.8.0_271.



Gambar 3. 5 Tampilan Versi Java

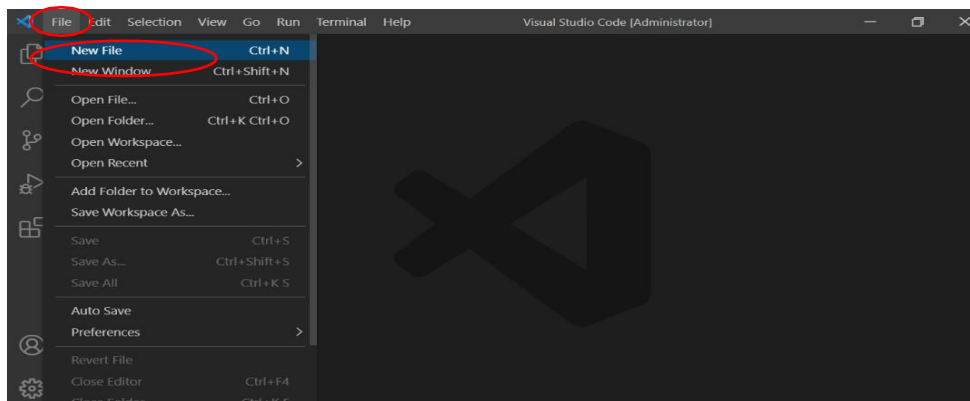
B. Proses Source Code

1. Langkah dalam membuat *Soursode* menggunakan *Visual Studio Code* dengan mengklik kanan lalu pilih *Open* , maka *Visual Studio Code* akan terbuka.



Gambar 3. 6 Membuka *Software Visual Code*

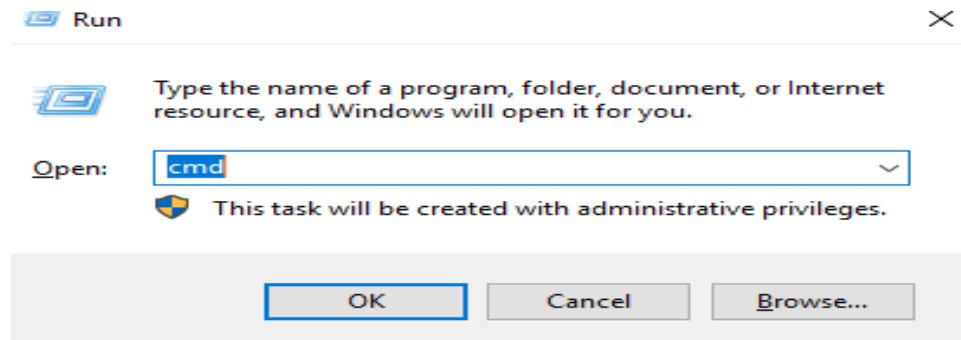
2. Setelah jendela *Visual Studio Code* terbuka pilih *file* => *New File* maka tuliskan *Sourcode* Program Penghitung Pajak Kendaraan yang akan di buat Programnya.



Gambar 3. 7 Membuat *Source Code*

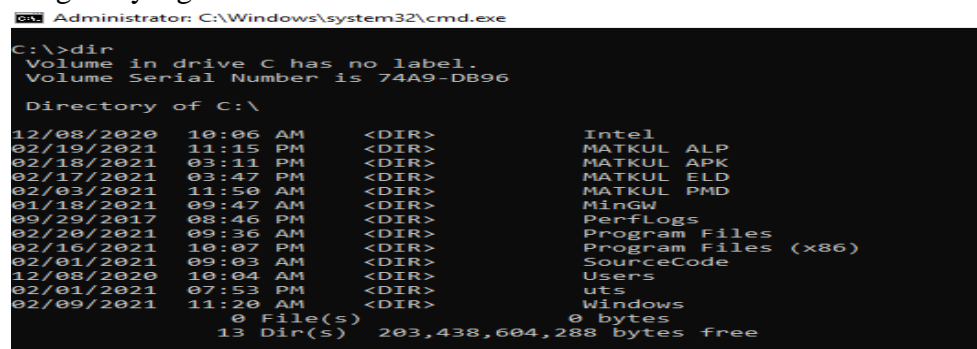
3. Setelah *Sourece Code* selesai ditulis maka di simpan.

6. Setelah selesai menyimpan *File* maka untuk menjalankan Program yang telah di buat dengan *Mengcompile* menggunakan *CMD*.



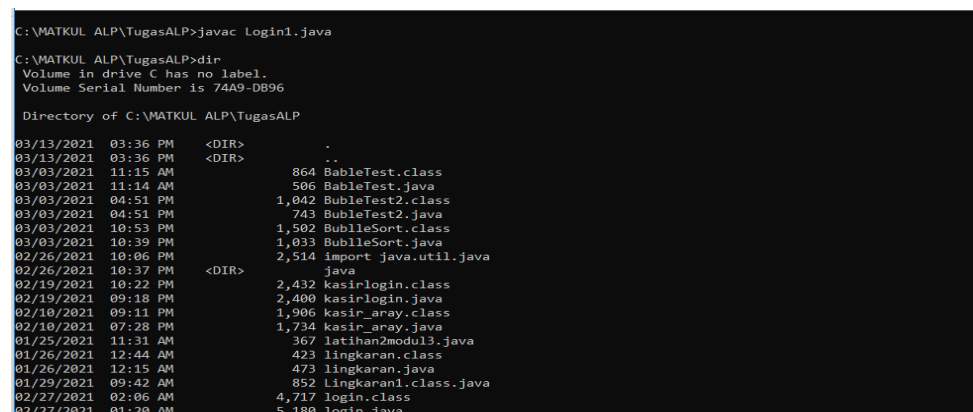
Gambar 3. 11 Membuka CMD

7. Setelah terbuka jendela *CMD* maka untuk melihat folder penyimpanan file Program yang telah di buat di Data C.



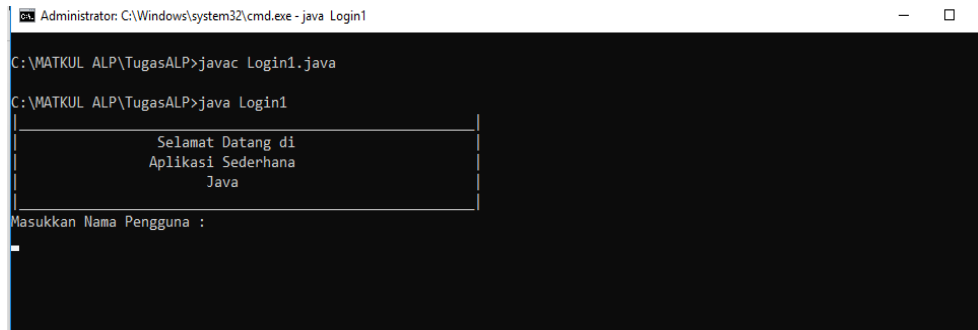
Gambar 3. 12 Melihat File pada Data C

8. Setelah masuk ke dalam data C langkah selanjutnya masuk ke dalam folder tempat dimana menyimpan folder yang telah di save.



Gambar 3. 13 Membuka Folder

9. Untuk meng-compile program yang telah di simpan dengan rumus Javac nama *file* .Java.

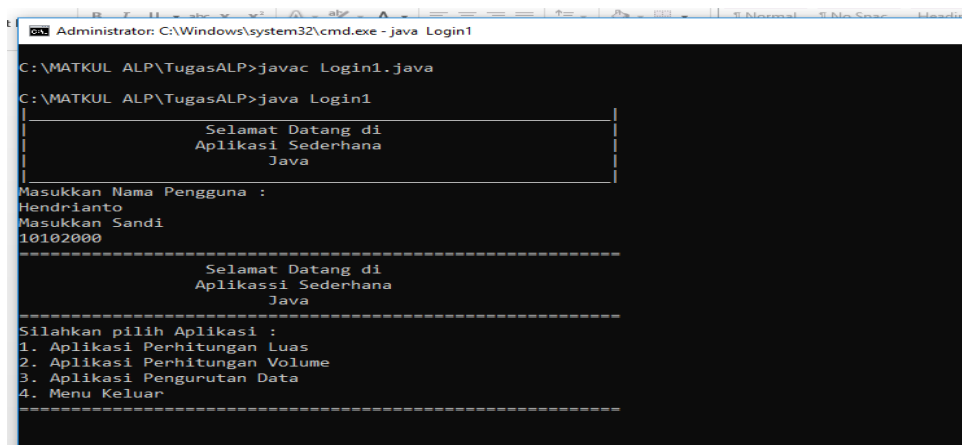


```
Administrator: C:\Windows\system32\cmd.exe - java Login1
C:\MATKUL ALP\TugasALP>javac Login1.java
C:\MATKUL ALP\TugasALP>java Login1
Selamat Datang di
Aplikasi Sederhana
Java
Masukkan Nama Pengguna :

```

Gambar 3. 14 *Compile Program*

10. Tampilan awal Program dan Pemilihan Program.

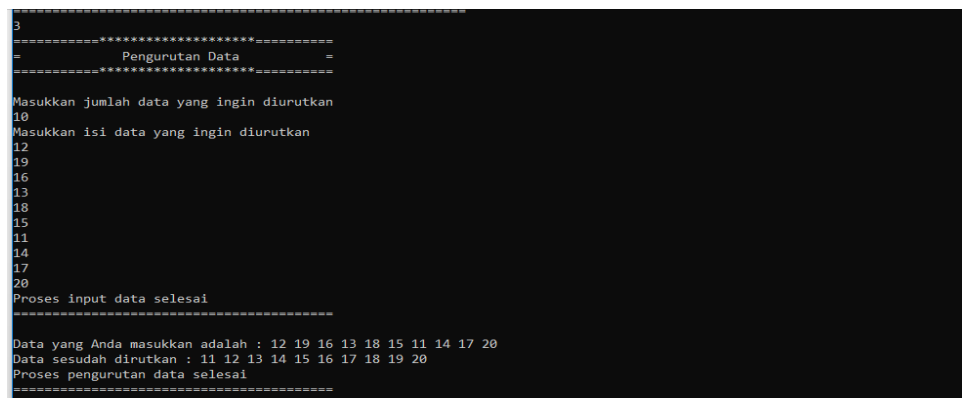


```
Administrator: C:\Windows\system32\cmd.exe - java Login1
C:\MATKUL ALP\TugasALP>javac Login1.java
C:\MATKUL ALP\TugasALP>java Login1
Selamat Datang di
Aplikasi Sederhana
Java
Masukkan Nama Pengguna :
Hendrianto
Masukkan Sandi
10102000
Selamat Datang di
Aplikasi Sederhana
Java
Silahkan pilih Aplikasi :
1. Aplikasi Perhitungan Luas
2. Aplikasi Perhitungan Volume
3. Aplikasi Pengurutan Data
4. Menu Keluar

```

Gambar 3. 15 Tampilan Awal dan Pemilihan Program

11. Salah satu tampilan Program yang di pilih.



```
3
=====
= Pengurutan Data =
=====
Masukkan jumlah data yang ingin diurutkan
10
Masukkan isi data yang ingin diurutkan
12
19
16
13
18
15
11
14
17
20
Proses input data selesai
=====
Data yang Anda masukkan adalah : 12 19 16 13 18 15 11 14 17 20
Data sesudah diurutkan : 11 12 13 14 15 16 17 18 19 20
Proses pengurutan data selesai
=====

```

Gambar 3. 16 Tampilan Yang Telah di Pilih

C. Penjelasan Source Code

1. Header File

```
1 import java.util.Scanner;
```

Gambar 3. 17 Header

Penjelasan:

Pada bagian ini, nomor 1, kita melakukan impor library yang dibutuhkan pada program. Library merupakan sekumpulan *class* dan fungsi yang bisa kita gunakan dalam membuat program.

2. Deklarasi

```
2 public class Login1{
3     public static void main(String[] args){
4         Scanner uname=new Scanner(System.in);
5         Scanner pw=new Scanner(System.in);
6     }
```

Gambar 3. 18 Deklarasi

Penjelasan:

Deklarasi diatas adalah bagian untuk mendefinisikan semua yang dipakai dalam pemograman. Dari no 2 – 5 merupakan sebagai wadah dalam sebuah pemograman atau juga bisa disebut sebagai *Pendeklarasian*.

3. Hello World

```
7     String nama_pengguna[]={ "Hendrianto"}, nama;
8     String passwd[]={ "10102000"}, sandi;
9     System.out.println("Masukkan Nama Pengguna :");
10    System.out.println("Selamat Datang di");
11    System.out.println("Aplikasi Sederhana");
12    System.out.println("Java");
13    System.out.println("Masukkan Sandi");
14    System.out.println("Masukkan Nama Pengguna :");
15    nama = uname.nextLine();
16    System.out.println("Masukkan Sandi");
17    sandi=pw.nextLine();
```

Gambar 3. 19 Hello Word

Penjelasan:

Dari nomor 7-17, adalah merupakan Hello World! atau Halo Dunia adalah sebuah kata salam yang biasanya selalu dipelajari untuk pertama kalinya dalam belajar sebuah bahasa pemrograman komputer. jadi dapat kita

sebut bahwa Program Hello World! adalah program awal, yaitu program yang simple dan biasanya hanya menampilkan teks "Hello World!" pada layar komputer maupun Mikrokontroler.

4. Tipe Data

```

47      menu=nextInt();
48      if (menu==0) {
49      }
50      else if (menu==1){
51          PerhitunganLuas();
52      }
53      else if (menu==2){
54          PerhitunganVolume();
55      }
56      else if (menu==3){
57          PengurutanData();
58      }
59      else if (menu==4){
60          System.exit(0);
61      }
62      else {
63          System.out.println("Maaf tidak ada Menu untuk inputan Anda");
64          menu();
65      }
66  }
67  }

```

Gambar 3. 20 Tipe Data

Penjelasan :

Tipe data merupakan bagian program yang paling penting karena tipe data mempengaruhi setiap instruksi yang akan dilaksanakan oleh computer. Jika dalam kehidupan sehari-hari tidak ada batasan untuk melakukan operasi pada bilangan, akan sangat berbeda jika operasi bilangan tersebut dilakukan dengan komputer. Kelemahan - kelimahan pada komputer tidak memungkinkan untuk melakukan perhitungandiluar dari kemampuan komputer tersebut. Untuk itu dalam bahasa pemrograman komputer dikenal dengan istilah tipe data.

5. Deskripsi

```

33      static void menu(){
34          int menu;
35          Scanner h1=new Scanner(System.in);
36          System.out.println("=====");
37          System.out.println("                Selamat Datang di                ");
38          System.out.println("                Aplikasi Sederhana                ");
39          System.out.println("                Java                ");
40          System.out.println("=====");
41          System.out.println("Silahkan pilih Aplikasi :");
42          System.out.println("1. Aplikasi Perhitungan Luas");
43          System.out.println("2. Aplikasi Perhitungan Volume");
44          System.out.println("3. Aplikasi Pengurutan Data");
45          System.out.println("4. Menu Keluar");
46          System.out.println("=====");
47          menu=h1.nextInt();

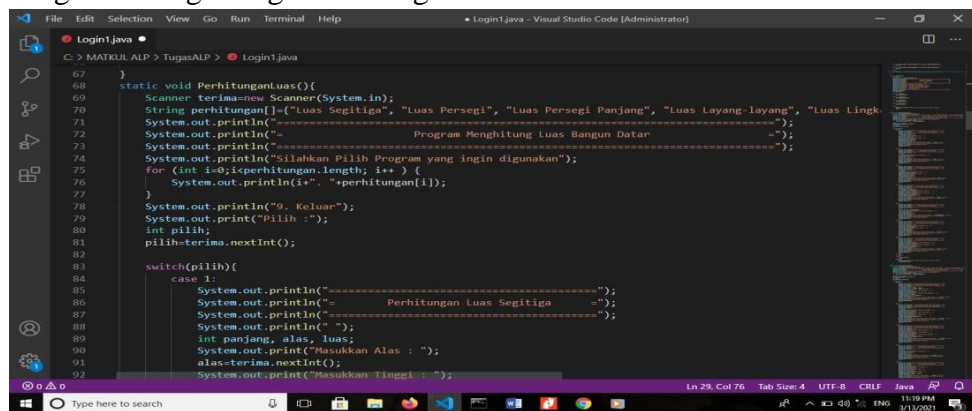
```

Gambar 3. 21 Deskripsi

Penjelasan:

Dari no 33 – 47 merupakan inti dari sebuah Deklarasi yang merupakan Inti suatu programan yang berisi uraian langkah- langkah penyelesaian masalah. Program C pada hakikatnya tersusun atas sejumlah blok fungsi sebuah program minimal mengandung sebuah fungsi, setiap fungsi terdiri dari satu atau beberapa pernyataan yang secara keseluruhan yang dimaksudkan untuk melaksanakan tugas.

6. Program Menghitung Luas Bangun Datar

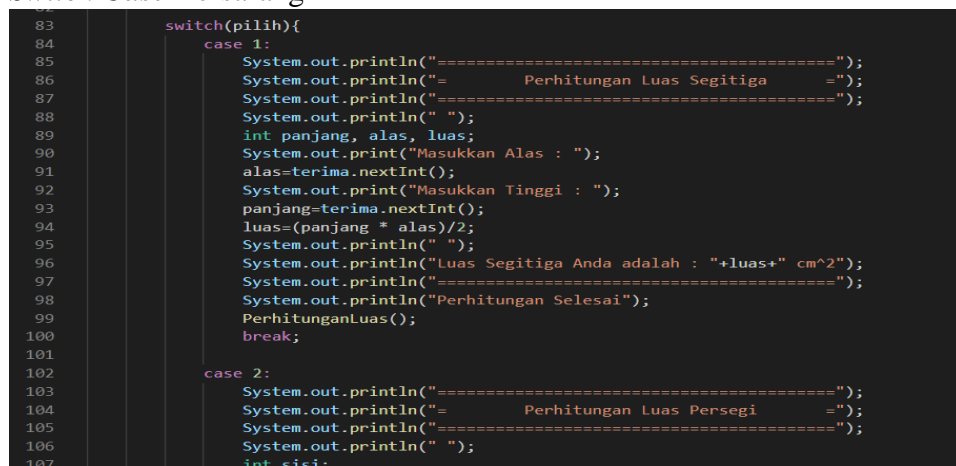


Gambar 3. 22 Menghitung Luas Bangun Datar

Penjelasan :

Meurpakan salah satu Program yang ada pada daftar menu dalam Aplikasi Program Sederhana Java Ini, yang terdiri beberapa Bangun Datar yang terdapat didalam Aplikasi Sederhana Java ini.

7. Switch Case Bersarang



Gambar 3. 23 Switch Case

Penjelasan :

Pada program diatas menggunakan switch case bersarang untuk melakukan pemilihan Bangun datar yang akan ditampilkan. Pada program ini menggunakan 9 switch case , program ini juga menggunakan **fungsi pindah program secara otomatis** untuk mengarahkan eksekusi program ke pernyataan selanjutnya . Dan didalam setiap block case diakhiri dengan perintah break, agar struktur case langsung berhenti begitu kondisi terpenuhi.

8. Pengurutan Data

```
367 }
368 static void PengurutanData(){
369     Scanner terima=new Scanner(System.in);
370     int n;
371     System.out.println("=====*****");
372     System.out.println("=          Pengurutan Data          =");
373     System.out.println("=====*****");
374     System.out.println();
375     System.out.println("Masukkan jumlah data yang ingin diurutkan");
376     n=terima.nextInt();
377     int data[]=new int [n];
378     System.out.println("Masukkan isi data yang ingin diurutkan");
379     for (int i=0; i<n; i++){
380         data[i]=terima.nextInt();
381     }
382     System.out.println("Proses input data selesai");
383     System.out.println("=====");
384     System.out.println();
385     System.out.print("Data yang Anda masukkan adalah : ");
386     for (int j=0;j<n;j++){
387         System.out.print(data[j]+" ");
388     }
389     System.out.println();
390     int tmp;
391     for (int i=0;i<data.length;i++){
392         for (int j=0;j<(data.length-1);j++) {
```

Gambar 3. 24 Pengurutan Data

Penjelasan :

Pengurutan data atau sering di sebut juga *Bable Short* merupakan pengurutan data mulai dari terkecil hingga ke terbesar atau sebalik nya dari terbesar hingga ke terkecil atau di sebut dengan istila (*Asscending* dan *Dissending*)

BAB IV

PENUTUP

A. Kesimpulan

Program Aplikasi Sederhana Java ini sudah berhasil dibuat, pada program ini menggunakan Tipe Data, Percabangan, Perulangan, Pengurutan untuk menyeleksi suatu kondisi tunggal, else if bersarang, dan switch case bersarang. Program ini juga menggunakan beberapa perintah seperti printf dan scanf. Untuk memperindah tampilan program menggunakan system color yang berada pada setiap blok program.

B. Saran

Program ini masih jauh dari kata sempurna dan untuk kedepannya agar lebih memperindah tampilan dan lebih banyak lagi pilihan aplikasi di dalam Program Aplikasi Sederhana Java ini kedepannya. untuk menyempurnakan program ini membutuhkan perkembangan lagi mulai dari tampilan program, kerapian, dibutuhkan lagi database agar dapat menampung data secara permanen agar lebih efisien.

DAFTAR PUSTAKA

Fakmaludin, I. S. (2018). *PEMOGRAMAN C UNTUK PENYELESAIAN KASUS*. Yogyakarta: Penerbit Periuk.