

LAPORAN  
APLIKASI PERHITUNGAN LUAS BANGUN DATAR  
MENGGUNAKAN JAVA

UJIAN AKHIR SEMESTER I



Disusun oleh :

ATIQAH NAJWA ANGGRAINI  
NIM 202013024

DOSEN PENGAMPU  
SLAMET TRIYANTO, S. ST.

PROGRAM STUDI TEKNIK INFORMATIKA  
POLITEKNIK KAMPAR  
2020-2021

## **KATA PENGANTAR**

Puji dan syukur atas kehadirat Allah SWT, atas segala limpahan taufiq serta hidayah-Nya yang telah memberi penulis kesempatan untuk menyelesaikan Laporan Pratikum Tugas Praktik Algoritma ini. Adapun tujuan penulisan Laporan ini adalah untuk melengkapi Tugas Perkuliahinan Praktik Algoritma.

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 Praktik Algoritma.

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, 14 Maret 2021

Penulis

## DAFTAR ISI

<b>KATA PENGANTAR.....</b>	i
<b>DAFTAR ISI.....</b>	ii
<b>DAFTAR GAMBAR.....</b>	iv
<b>DAFTAR TABEL .....</b>	vi
<b>BAB I TINJAUAN PUSTAKA.....</b>	1
A. Pengertian .....	1
1. Pengertian Bahasa Java.....	1
2. Pengertian CLI (Command Line Interface) .....	3
B. Tujuan Praktikum.....	4
C. Alat Dan Bahan.....	4
1. Alat .....	4
2. Bahan .....	4
<b>BAB II PEMBAHASAN.....</b>	5
A. Struktur Java .....	5
1. Comments .....	5
2. Reserved words .....	5
3. Modifiers.....	5
4. Statements .....	5
5. Blocks .....	5
6. Classes.....	5
7. Methods.....	5
B. Variabel dan Tipe Data.....	6
1. Variabel.....	6
2. Tipe Data.....	7
C. Escape Sequence .....	11
D. Operator.....	11
1. Operator Aritmatika.....	11
2. Operator Penugasan.....	12
3. Operator Relasional .....	12
4. Operator Logika.....	12
5. Operator Penggabungan.....	13
E. Perulangan .....	13
1. Perulangan While .....	13
2. Perulangan Do ... While.....	13
3. Perulangan For .....	14
F. Percabangan.....	14
1. Struktur If.....	14
2. Struktur If ... Else .....	15
3. Struktur If ... Else If ... Else.....	15
4. Struktur Switch.....	16
G. Array.....	17
H. Pengurutan .....	17
1. Jenis Pengurutan.....	17
2. Metode Pengurutan.....	17

<b>BAB III LANGKAH KERJA .....</b>	<b>19</b>
A. Mengunduh Aplikasi JDK 8 .....	19
B. Instalasi Aplikasi JDK.....	20
C. Setting Path.....	22
D. Menguji Instalasi JDK 8.....	25
E. Program Aplikasi Perhitungan Luas Bangun Datar.....	26
F. Tampilan Aplikasi.....	35
1. Tampilan awal.....	35
2. Username dan password yang dimasukkan benar.....	36
3. Username dan password yang dimasukkan salah.....	36
4. Jika memilih menu 1 .....	37
<b>BAB IV PENUTUP .....</b>	<b>44</b>
A. Kesimpulan .....	44
B. Saran.....	44
<b>DAFTAR PUSTAKA .....</b>	<b>45</b>

## DAFTAR GAMBAR

Gambar 2. 1 <i>Method main()</i> .....	6
Gambar 2. 2 <i>Deklarasi Variabel 1</i> .....	10
Gambar 2. 3 <i>Deklarasi Variabel 2</i> .....	10
Gambar 2. 4 <i>Deklarasi Variabel 3</i> .....	10
Gambar 2. 5 <i>Perulangan While</i> .....	13
Gambar 2. 6 <i>Perulangan Do ... While</i> .....	13
Gambar 2. 7 <i>Perulangan For</i> .....	14
Gambar 2. 8 <i>Percabangan If</i> .....	15
Gambar 2. 9 <i>Percabangan If ... Else</i> .....	15
Gambar 2. 10 <i>Percabangan If ... Else If ... Else</i> .....	16
Gambar 2. 11 <i>Percabangan Switch</i> .....	16
Gambar 2. 12 <i>Deklarasi Array 1</i> .....	17
Gambar 2. 13 <i>Deklarasi Array 2</i> .....	17
Gambar 3. 1 Tampilan Website Resmi Download JDK 8 .....	19
Gambar 3. 2 Versi Aplikasi JDK 8.....	19
Gambar 3. 3 <i>Accept License Agreement</i> .....	20
Gambar 3. 4 Instal JDK 8 1 .....	20
Gambar 3. 5 Instalasi JDK 8 2 .....	21
Gambar 3. 6 Instalasi JDK 8 3 .....	21
Gambar 3. 7 <i>Instalasi JDK 8 4</i> .....	22
Gambar 3. 8 <i>Salin Lokasi JDK 8</i> .....	22
Gambar 3. 9 <i>Membuka Jendela System</i> .....	23
Gambar 3. 10 <i>Jendela System</i> .....	23
Gambar 3. 11 <i>Jendela System Properties</i> .....	24
Gambar 3. 12 <i>Jendela Environment Variabel</i> .....	24
Gambar 3. 13 <i>Jendela Edit System Variable</i> .....	25
Gambar 3. 14 <i>Jendela Run</i> .....	25
Gambar 3. 15 <i>Penginstalan JDK Berhasil</i> .....	26
Gambar 3. 16 <i>Source Code 1</i> .....	26
Gambar 3. 17 <i>Source Code 2</i> .....	27
Gambar 3. 18 <i>Source Code 3</i> .....	27
Gambar 3. 19 <i>Source Code 4</i> .....	28
Gambar 3. 20 <i>Source Code 5</i> .....	29
Gambar 3. 21 <i>Source Code 6</i> .....	29
Gambar 3. 22 <i>Source Code 7</i> .....	29
Gambar 3. 23 <i>Source Code 8</i> .....	30
Gambar 3. 24 <i>Source Code 9</i> .....	31
Gambar 3. 25 <i>Source Code 10</i> .....	31
Gambar 3. 26 <i>Source Code 11</i> .....	32
Gambar 3. 27 <i>Source Code 12</i> .....	33
Gambar 3. 28 <i>Source Code 13</i> .....	33
Gambar 3. 29 <i>Source code 14</i> .....	34
Gambar 3. 30 <i>Source Code 15</i> .....	34
Gambar 3. 31 <i>Source Code 16</i> .....	35

Gambar 3. 32 <i>Source Code 17</i> .....	35
Gambar 3. 33 <i>Tampilan Aplikasi 1</i> .....	35
Gambar 3. 34 <i>Tampilan Aplikasi 2</i> .....	36
Gambar 3. 35 <i>Tampilan Aplikasi 3</i> .....	36
Gambar 3. 36 <i>Tampilan Aplikasi 4</i> .....	37
Gambar 3. 37 <i>Tampilan Aplikasi 5</i> .....	37
Gambar 3. 38 <i>Tampilan Aplikasi 6</i> .....	38
Gambar 3. 39 <i>Tampilan Aplikasi 7</i> .....	38
Gambar 3. 40 <i>Tampilan Aplikasi 8</i> .....	39
Gambar 3. 41 <i>Tampilan Aplikasi 9</i> .....	39
Gambar 3. 42 <i>Tampilan Aplikasi 10</i> .....	40
Gambar 3. 43 <i>Tampilan Aplikasi 11</i> .....	40
Gambar 3. 44 <i>Tampilan Aplikasi 12</i> .....	41
Gambar 3. 45 <i>Tampilan Aplikasi 13</i> .....	41
Gambar 3. 46 <i>Tampilan Aplikasi 14</i> .....	42
Gambar 3. 47 <i>Tampilan Aplikasi 15</i> .....	42
Gambar 3. 48 <i>Tampilan Aplikasi 16</i> .....	43
Gambar 3. 49 <i>Tampilan Aplikasi 17</i> .....	43

## **DAFTAR TABEL**

Tabel 2. 1 Daftar Escape Sequence .....	11
Tabel 2. 2 Operator Aritmatika .....	11
Tabel 2. 3 Operator Penugasan.....	12
Tabel 2. 4 Operator Relasional.....	12
Tabel 2. 5 Operator Logika .....	12

# **BAB I**

## **TINJAUAN PUSTAKA**

### **A. Pengertian**

#### **1. Pengertian Bahasa Java**

Java adalah nama sebuah bahasa pemrograman yang sangat terkenal. Sebagai bahasa pemrograman, java dapat digunakan untuk menulis program. Bahasa java dikembangkan di Sun Microsystem dan mulai diperkenalkan kepada public pada tahun 1995. Java merupakan bahasa yang berorientasi objek. Java mempunyai keunggulan yakni bersifat universal. Sebagai bahasa yang universal, java bias dijumpai di berbagai platform (Linux, UNIX, Windows, Mac, dan lain – lain). Hasil kompilasi java yang dinamakan bytecode dapat dijalankan di berbagai platform sepanjang di system target memiliki Java Runtime Environment (JRE).

Menurut Garling dan Lestari (2010:1) Java adalah sebuah bahasa pemrograman scripting yang sering digunakan dalam pembuatan aplikasi berbasis handphone dan juga dapat digunakan untuk menyediakan akses objek yang disisipkan di aplikasi lain. Java berfungsi sebagai penambah tingkah laku agar widget dapat tampil lebih atraktif. Menurut definisi dari Sun, Java adalah nama untuk sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada komputer standalone ataupun pada lingkungan jaringan.

Kelebihan Java :

- a) Multiplatform dan multidevice. Sekali anda menuliskan sebuah program dengan menggunakan Java, anda dapat menjalankannya hampir di semua komputer dan perangkat lain yang support Java, dengan sedikit perubahan atau tanpa perubahan sama sekali dalam kodennya.
- b) OOP atau Object Oriented Programming. Memudahkan untuk mendesign dan mengembangkan program dengan cepat dan

teliti , sehingga mudah digunakan. Salah satu bahasa pemrograman yang berorientasi objek secara murni.

- c) Bergaya C++, memiliki sintaks seperti bahasa pemrograman C++ sehingga menarik banyak pemrogram C++ untuk pindah ke Java. Saat ini pengguna Java sangat banyak, sebagian besar adalah pemrogram C++ yang pindah ke Java. Universitas-universitas di Amerika Serikat juga mulai berpindah dengan mengajarkan Java kepada murid-murid yang baru karena lebih mudah dipahami oleh murid dan dapat berguna juga bagi mereka yang bukan mengambil jurusan komputer.
- d) Java memiliki library yang lengkap. Library disini adalah sebuah kumpulan dari program yang disertakan dalam Java. Hal ini akan memudahkan pemrograman menjadi lebih mudah. Kelengkapan library semakin beragam jika ditambah dengan karya komunitas Java. Setiap hal pasti memiliki kelebihan dan kekurangan.

Kekurangan Java :

- a) Tulis sekali, jalankan di mana saja - Masih ada beberapa hal yang tidak kompatibel antara platform satu dengan platform lain. Untuk J2SE, misalnya SWT-AWT bridge yang sampai sekarang tidak berfungsi pada Mac OS X.
- b) Mudah didekompilasi. Dekompilasi adalah proses membalikkan dari kode jadi menjadi kode sumber. Ini dimungkinkan karena kode jadi Java merupakan bytecode yang menyimpan banyak atribut bahasa tingkat tinggi, seperti nama-nama kelas, metode, dan tipe data. Hal yang sama juga terjadi pada Microsoft .NET Platform. Dengan demikian, algoritma yang digunakan program akan lebih sulit disembunyikan dan mudah dibajak/direverse-engineer.

- c) Pengumpulan sampah otomatis, memiliki fasilitas pengaturan penggunaan memori sehingga para pemrogram tidak perlu melakukan pengaturan memori secara langsung seperti halnya dalam bahasa C++ yang dipakai secara luas (Aulia, 2017).

## 2. Pengertian CLI (Command Line Interface)

CLI (Command Line Interface) atau Antarmuka Baris Perintah adalah suatu bentuk penggunaan komputer berbasis teks, dimana pengguna komputer memberikan perintah (input) kepada komputer melalui pengetikan sejumlah baris perintah untuk kemudian diolah (diproses) agar menjadi sebuah keluaran (output).

Tampilan antarmuka pengguna atau user interface (UI) pada CLI biasanya adalah berbasis teks yang sangat sederhana dan tidak serumit tampilan UI berbasis window. Pengguna komputer cukup memberikan input pada layar berbasis teks sederhana yang ditandai dengan cursor yang berkedip yang berarti sebuah tanda bahwa komputer siap menerima perintah dari pengguna.

CLI tidak menyediakan grafik, baik berupa gambar-gambar, jendela-jendela ataupun animasi-animasi yang bisa memanjakan pengguna, melainkan hanya berupa teks yang harus diketikkan oleh pengguna. Jadi, apabila user ingin melakukan operasi dalam sistem operasi tersebut, misalnya melakukan copy, rename, cut, delete, dan sebagainya, maka pengguna harus mengetikkan perintah berupa teks dengan cara manual.

Pada CLI, pengguna mengetikkan perintah-perintah atau menekan tombol-tombol khusus pada keyboard untuk memasukkan data dan perintah. CLI seringkali sulit digunakan karena memerlukan ejaan, tata bahasa, dan tanda baca yang tepat. Kesalahan kecil seperti kurangnya tanda titik akan menghasilkan pesan kesalahan. Akan tetapi, CLI memberi penggunaan lebih banyak kendali untuk mengatur hal-hal kecil dan rinci.

**B. Tujuan Praktikum**

1. Tujuan yang pertama agar mahasiswa lebih memahami tentang bahasa Java.
2. Tujuan yang kedua agar memahami dasar-dasar bahasa Java untuk membuat program menghitung luas bangun datar.
3. Tujuan yang ketiga untuk memahami sourcode yang digunakan pada pembuatan aplikasi perhitungan luas bangun datar.

**C. Alat Dan Bahan**

1. Alat
  - a. Laptop
2. Bahan
  - a. Aplikasi Sublime Text 3
  - b. CMD
  - c. Aplikasi JDK

## **BAB II**

### **PEMBAHASAN**

#### **A. Struktur Java**

##### 1. Comments

*Comments* digunakan untuk memberikan keterangan/penjelasan suatu baris program. *Comments* tidak akan dikompilasi oleh *compiler*.

Lambang untuk *comments* :

- a. *double slash* ( // ) untuk memberikan penjelasan satu baris
- b. /\* dan \*/ untuk memberikan penjelasan satu baris atau lebih.

##### 2. Reserved words

*Reserved words* atau *keywords* merupakan kata-kata spesifik digunakan oleh *compiler* untuk keperluan pemrograman. Contoh *reserved words class* digunakan untuk membuat *class* pada java.

##### 3. Modifiers

*Modifiers* digunakan untuk menentukan penggunaan dari suatu data, methods dan *class*. Contoh *modifiers* adalah *public, static, private, final, abstract* dan *protected*.

##### 4. Statements

*Statements* merupakan baris perintah atau kumpulan perintah. Setiap *statements* pada java selalu diakhiri dengan titik koma ( ; ).

##### 5. Blocks

*Blocks* digunakan untuk membentuk suatu grup *statements*. *Blocks* diawali dengan kurung kurawal buka ( { ) dan kurung kurawal tutup ( } ).  
*Blocks* dapat digunakan secara *nested* (*blocks* didalam *blocks*).

##### 6. Classes

*Classes* merupakan inti dari program Java. Suatu *class* merupakan “*blueprint*” untuk menciptakan suatu object.

##### 7. Methods

*Methods* merupakan kumpulan dari *statements* yang melakukan suatu fungsi atau operasi secara sekuens. Setiap java aplikasi harus

mempunyai *method main()* (kecuali applet). *Method main()* pada java selalu berbentuk :

```
public static void main(String [] args) {  
//statements  
}
```

Gambar 2. 1 *Method main()*

## B. Variabel dan Tipe Data

### 1. Variabel

Variabel adalah lokasi yang ada didalam memori komputer yang nilainya dapat berubah-ubah atau yang memiliki variasi nilai, nilai variable tersebut dapat berubah di pernyataan atau program. Sebuah class dapat memiliki jenis-jenis variabel berikut :

- a. Variabel lokal adalah variabel yang didefinisikan didalam sebuah method, konstruktor, atau blok. Variabel ini dideklarasikan dan diinisialisasi di dalam method dan kemudian akan dihapus setelah method selesai dijalankan. Modifier akses tidak dapat digunakan untuk variabel lokal. Variabel lokal hanya dapat digunakan di dalam method, constructor, atau blok tempat dimana ia dideklarasikan. Tidak ada nilai default untuk variabel lokal sehingga variabel lokal harus dideklarasikan dan dinisialisasi sebelum digunakan. Variabel lokal harus diinisialisasi (diberi nilai). Jadi, jika bagian ini dihilangkan, dan hanya ditulis int umur saja, maka akan terjadi error: *variable might not have been initialized.*
- b. Variabel instance merupakan variabel yang ada di dalam class tetapi berada di luar method. Variabel ini diinisialisasi saat kelas dimuat. Variabel ini juga dapat diakses dari dalam method, konstruktor, atau blok dari kelas tersebut. Saat sebuah objek dibuat, tempat untuk setiap variabel instance disiapkan dalam memori. Variabel instance dibuat saat sebuah objek dibuat dengan menggunakan kata kunci new dan dihapus saat objek tersebut

dihapus. Variabel instance dapat dideklarasikan dalam class sebelum atau sesudah Modifier akses dapat digunakan pada variabel instance. Variabel instance dapat digunakan oleh semua method, *constructor*, dan blok di dalam kelas. Variabel instance mempunyai nilai default: untuk tipe numerik sebesar 0, untuk Boolean bernilai *false*, dan untuk referensi objek adalah null.

- c. Variabel class merupakan variabel yang dideklarsikan di dalam sebuah class di luar method, dan menggunakan kata kunci statis. Hanya ada satu salinan dari variabel class/static ini per class. Variabel static biasanya dideklarasikan sebagai konstanta. Variabel konstanta nilainya tetap dan tidak berubah. Variabel static dibuat ketika program dimulai dan dihapus saat program berhenti. Nilai defaultnya sama dengan variabel instance. Variabel static dapat diakses dengan memanggil nama class, misalnya **NamaClass.NamaVariabel**. Saat mendeklarasi 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 lokal.

## 2. Tipe Data

Tipe data adalah jenis data pada suatu bahasa pemrograman yang mempunyai batasan tempat dan karakteristik tertentu dengan interpretasi data dan dapat diolah oleh komputer untuk kepentingan pemograman. Setiap tipe data memiliki karakteristik dan jangkauan yang berbeda-beda. Jangkauan adalah kapasitas tampung angka atau data dalam setiap tipe data.

### a. Tipe Data Primitif

#### 1) Byte

- a) Tipe data byte merupakan integer 8-bit yang bersifat signed.
- b) Nilai minimum adalah 128.

- c) Nilai maksimum adalah 127.
  - d) Nilai default adalah 0.
  - e) Tipe data byte digunakan untuk menghemat ruang pada array yang besar.
- 2) Short
- a) Tipe data short merupakan integer 16-bit yang bersifat signed.
  - b) Nilai minimum adalah -32.768.
  - c) Nilai maksimum adalah 32.767.
  - d) Nilai default adalah 0.
  - e) Tipe data short juga dapat menghemat ruang seperti pada tipe data byte.
- 3) Int
- a) Tipe data int merupakan integer 32-bit yang bersifat signed.
  - b) Nilai minimum adalah -2.147.483.648 .
  - c) Nilai maksimum adalah 2.147.483.647 .
  - d) Nilai default adalah 0.
  - e) Tipe data int digunakan secara umum untuk menampung nilai-nilai integral.
- 4) Long
- a) Tipe data long merupakan integer 64-bit yang bersifat signed.
  - b) Nilai minimum adalah -9.223.372.036.854.775.808 .
  - c) Nilai maksimum adalah 9.223.372.036.854.775.807 .
  - d) Nilai default adalah 0L.
  - e) Tipe data long digunakan untuk menampung nilai-nilai dengan jangkauan nilai yang lebih dari tipe data int.
- 5) Float
- a) Tipe data float merupakan nilai desimal 32-bit presisi tunggal.

- b) Float biasanya digunakan untuk menghemat ruang pada array besar yang berisi nilai-nilai desimal.
  - c) Nilai default adalah 0,0f.
  - d) Tipe data float tidak pernah digunakan untuk nilai-nilai seperti nilai mata uang,
- 6) Doubel
- a) Tipe data doubel merupakan desimal 64-bit presisi ganda.
  - b) Doubel biasanya digunakan sebagai tipe data default untuk nilai desimal.
  - c) Nilai deafault adalah 0,0d.
  - d) Tipe data doubel tidak pernah digunakan untuk nilai-nilai presisi seperti nilai mata uang.
- 7) Boolean
- a) Tipe data boolean merepresentasikan satu bit informasi.
  - b) Hanya ada dua nilai yang mungkin : *true* (benar) dan *false* (salah).
  - c) Tipe data ini digunakan sebagai penanda kondisi benar/salah.
  - d) Nilai default adalah *false*.
- 8) Char
- a) Tipe data char merupakan karakter Unicode 16-bit tunggal.
  - b) Nilai minimum adalah '\u0000'.
  - c) Nilai maksimum adalah '\uffff'.
  - d) Tipe data char digunakan untuk menampung karakter apa saja.
- 9) String

Tipe data String digunakan untuk tipe data teks yang dapat menggabungkan tipe data angka, huruf, symbol dan lain-lain.

### b. Tipe Data Referensi

Variabel referensial dibuat dengan menggunakan konstruktor yang didefinisikan dalam class. Variabel ini digunakan untuk mengakses objek-objek. Variabel ini dideklarasikan sebagai tipe tertentu yang tidak dapat diubah.

Objek class dan berbagai tipe variabel array merupakan tipe data referensi. Nilai default adalah null. Variabel referensi dapat digunakan untuk mengacu pada objek dengan tipe yang sama atau yang kompatibel.

Setiap variabel dalam Java memiliki tipe spesifik, yang menentukan ukuran dan layout memori, jangkauan memori, jangkauan nilai yang dapat disimpan, dan operasi-operasi yang dapat dijalankan terhadap variabel.

Sebelum variabel digunakan, harus terlebih dahulu dideklarasikan. Ada beberapa cara mendeklarasikan variabel. Berikut bentuk-bentuk dasar mendeklarasikan variabel :

1. Tulis tipe data yang digunakan dan diikuti dengan nama variabel yang akan dibuat. Contoh :

```
String nama;
int umur;
```

Gambar 2. 2 Deklarasi Variabel 1

2. Untuk mendeklarasikan lebih dari satu variabel dengan tipe tertentu, dapat menggunakan tanda koma sebagai pemisah. Contoh :

```
String nama, alamat, kota;
int umur, berat, tinggi;
```

Gambar 2. 3 Deklarasi Variabel 2

3. Variabel dapat langsung diinisialisasi (memasukkan nilai) ke dalam variabel pada saat deklarasi secara langsung. Contoh :

```
int umur = 30;
```

Gambar 2. 4 Deklarasi Variabel 3

### C. Escape Sequence

Bahasa java mendukung beberapa escape sequence untuk literal String dan Char.

Tabel 2. 1 Daftar Escape Sequence

No	Notasi	Karakter yang diresentasikan
1	\b	backspace
2	\f	halaman baru
3	\n	baris baru
4	\r	awal baris
5	\s	Spasi
6	\t	Tab
7	\'	Tanda petik tunggal
8	\"	Tanda petik ganda
9	\	garis miring
10	\ddd	Karakter Octal
11	\xxxxx	Karakter Unicode Hexadesimal

### D. Operator

Operator adalah simbol atau karakter khusus (matematika) yang digunakan dalam suatu ekspresi. Dalam bahasa pemrograman baik java atau bahasa pemrograman lainnya membutuhkan berbagai macam operator dalam pembuatan source code, contoh operator aritmatika dan operator logika.

#### 1. Operator Aritmatika

Tabel 2. 2 Operator Aritmatika

Operator	Keterangan
+	Operator penjumlahan.
-	Operator pengurangan.
*	Operator perkalian.
/	Operator pembagian
%	Operator modulus (sisa bagi)

## 2. Operator Penugasan

Tabel 2. 3 Operator Penugasan

Simbol	Keterangan
$x += y$	$x = x + y$
$x -= y$	$x = x - y$
$x *= y$	$x = x * y$
$x /= y$	$x = x / y$
$x \%= y$	$x = x \% y$

## 3. Operator Relasional

Tabel 2. 4 Operator Relasional

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
$\geq$	Lebih besar atau sama dengan	$a \geq b$	benar jika a lebih besar atau sama dengan b
$\leq$	Lebih kecil atau sama dengan	$a \leq b$	benar jika a lebih kecil atau sama dengan b

## 4. Operator Logika

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

## 5. Operator Penggabungan

Operator penggabungan dinyatakan dengan symbol (+), operator penggabungan ini dapat di gunakan untuk menggabungkan antara tipe data String dengan tipe data String atau tipe data String dengan tipe data angka/bilangan.

## E. Perulangan

Perulangan digunakan untuk mengulang bagian tertentu dari struktur pemrograman sampai ditemukan kondisi dimana perulangan tersebut harus dihentikan. Bahasa pemrograman memiliki struktur pemprograman yang sama namun ditulis dengan syntax yang berbeda satu dengan lainnya.

### 1. Perulangan While

Perulangan while adalah struktur kontrol yang memungkinkan untuk mengulangi suatu proses dengan jumlah perulangan tertentu sampai ditemukan kondisi dimana perulangan akan dihentikan.

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

Gambar 2. 5 Perulangan While

### 2. Perulangan Do ... While

Perulangan do ... while sama seperti pengulangan while, tetapi perulangan do ... while pasti akan dieksekusi minimal satu kali.

```
do
{
    // statemen
}
while(ekspresi_Boolean);
```

Gambar 2. 6 Perulangan Do ... While

### 3. Perulangan For

Perulangan for adalah struktur kontrol repetitif yang memungkinkan untuk menjalankan proses dengan jumlah perulangan tertentu (jumlah perulangan sudah diketahui sebelumnya).

```
for(inisialisasi; ekspresi_Boolean; update)
{
    // statemen
}
```

Gambar 2. 7 Perulangan For

Berikut ini aliran proses dalam perulangan for:

- Bagian inisialisasi dieksekusi pertama kali dan hanya sekali. Bagian ini memungkin untuk mendeklarasikan dan menginisialisasikan variabel kontrol perulangan.
- Setelah itu, ekspresi Boolean dievaluasi. Jika bernilai benar, statemen dalam struktur for akan dieksekusi. Jika tidak, statemen tidak dieksekusi dan proses berlanjut setelah perulangan for.
- Setelah statemen dalam for dieksekusi, aliran proses kembali pada bagian update. Statemen update ini memungkin untuk meng-updatr nilai variabel kontrol dalam for.
- Ekspsi boolean kemudian dievaluasi lagi. Jika benar, statemen dalam perulangsn for kembali dieksekusi, dan dilanjutkan ke bagian update. Jika tidak, perulangan for akan selesai dan proses berlanjut pada bagian setelah perulangan for.

## F. Percabangan

Percabangan dapat membuat beberapa program yang akan dijalankan salah satunya atau beberapa di antaranya, apabila ada kondisi yang terpenuhi.

### 1. Struktur If

Struktur if dalam Java terdiri dari sebuah ekspresi Boolean yang diikuti dengan satu statemen atau lebih.

```

if (ekspresi_boolean)
{
    //statemen akan dieksekusi
//jika ekspresi Boolean bernilai benar
}

```

Gambar 2. 8 *Percabangan If*

## 2. Struktur If ... Else

Struktur if dapat diikuti dengan perintah else yang opsional agar bisa menyusun struktur program yang akan dijalankan apabila ekspresi Boolean yang diuji menghasilkan nilai salah.

```

if (ekspresi_boolean)
{
    //statemen akan dieksekusi
    //jika ekspresi Boolean bernilai benar
}
else
{
    //statemen akan dieksekusi
    //jika ekspresi Boolean bernilai salah
}

```

Gambar 2. 9 *Percabangan If... Else*

## 3. Struktur If ... Else If ... Else

Struktur if dapat diikuti dengan perintah else if .. else yang opsional. Saat menggunakan struktur if ... else if ... else ada beberapa hal yang harus diperhatikan.

- Sebuah perintah if dapat memiliki nol atau satu statemen else dan harus digunakan setelah statemen else if.
- Sebuah perintah if dapat memiliki nol atau banyak statemen else if dan harus digunakan sebelum statemen else.

```

if (ekspresi_boolean1)
{
    //statemen akan dieksekusi
    //jika ekspresi Boolean 1 bernilai benar
}
else if (ekspresi_boolean2)
{
    //statemen akan dieksekusi
    //jika ekspresi Boolean 2 bernilai benar
}
else if (ekspresi_boolean3)
{
    //statemen akan dieksekusi
    //jika ekspresi Boolean 3 bernilai benar
}
else
{
    //statemen akan dieksekusi
    //jika semua ekspresi Boolean bernilai salah
}

```

Gambar 2. 10 *Percabangan If... Else If... Else*

#### 4. Struktur Switch

Struktur switch memungkinkan sebuah variabel diperiksa kesamaannya dengan daftar nilai-nilai yang ada. Setiap nilai dalam daftar disebut case.

```

switch(ekspresi)
{
    case nilai1 :
        //statemen
    break; //opsional
    case nilai2 :
        //statemen
    break; //opsional
    default: //opsional
        //statemen
}

```

Gambar 2. 11 *Percabangan Switch*

## G. Array

Java menyediakan sebuah struktur data yaitu array. Array menampung koleksi elemen-elemen dengan tipe yang sama. Ada beberapa cara untuk mendeklarasikan array, yaitu :

1. Deklarasi array dengan memasukkan ukuran (pembatasan) array.

```
tipeData[] namaArray = new tipeData[ukuranArray];
```

Gambar 2. 12 *Deklarasi Array 1*

2. Deklarasi array dengan memasukkan nilai elemen-elemen ke dalam array.

```
tipeData[] namaArray ={nilai0,nilai1,...,nilai};
```

Gambar 2. 13 *Deklarasi Array 2*

## H. Pengurutan

Pengurutan atau sorting adalah suatu objek untuk menyusun kembali himpunan objek menggunakan aturan tertentu. Sorting juga dapat didefinisikan sebagai pengurutan data yang sebelumnya disusun secara acak, sehingga menjadi tersusun secara teratur.

1. Jenis Pengurutan

- a. Urutan Naik (Ascending) adalah pengurutan dimulai dari data yang paling kecil sampai paling besar.
- b. Urutan Turun (Descending) adalah pengurutan dimulai dari data yang paling besar sampai paling kecil.

2. Metode Pengurutan

- a. Bubble Sort

Bubble sort sering juga disebut dengan metode penukaran (exchange sort). Pengurutan bubble sort dilakukan dengan cara:

- 1) Bandingkan masing-masing item / data dalam suatu list secara berpasangan
- 2) Tukar posisi item tersebut jika diperlukan (ascending/descending)

- 3) Ulangi hingga akhir list sehingga tidak ada lagi item yang dapat ditukar.

b. Selection Sort

Cara pengurutan selection sort :

- 1) Memilih elemen dengan nilai paling rendah.
- 2) Tukar elemen yang terpilih tersebut dengan elemen ke-i.
- 3) Nilai dari i dimulai dari 1 ke n, dimana n adalah jumlah total elemen dikurangi 1.

c. Insertion Sort

Insertion sort merupakan salah satu metode sorting dengan cara menyisipkan (insert). Cara pengurutan insertion sort :

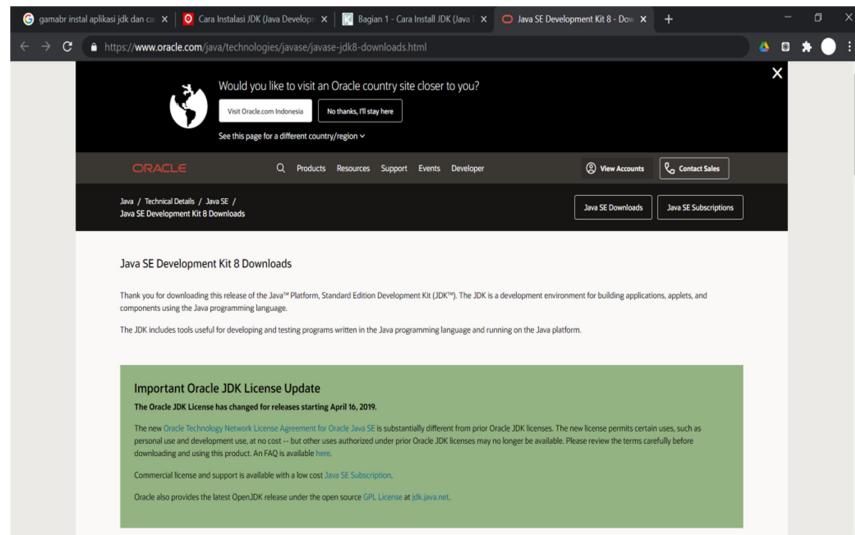
- 1) Data yang akan diurutkan dibagi menjadi dua bagian, yang belum diurutkan dan yang sudah diurutkan.
- 2) Elemen pertama diambil dari bagian array yang belum diurutkan dan kemudian diletakkan sesuai posisinya pada bagian lain dari array yang telah diurutkan.
- 3) Langkah ini dilakukan secara berulang hingga tidak ada lagi elemen yang tersisa pada bagian array yang belum diurutkan tersebut.

## BAB III

### LANGKAH KERJA

#### A. Mengunduh Aplikasi JDK 8

1. Download JDK 8 di *website* resminya atau di link berikut :  
<https://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html>



Gambar 3. 1 Tampilan Website Resmi Download JDK 8

2. Pilih versi aplikasi JDK 8 sesuai dengan spesifikasi laptop yang dipakai.

Platform	File Size	Download Link
Linux ARM 32 Hard Float ABI	73.47 MB	<a href="#">jdk-8u281-linux-arm32-vfp-hfll.tar.gz</a>
Linux x86 RPM Package	108.46 MB	<a href="#">jdk-8u281-linux-i586.rpm</a>
Linux x86 Compressed Archive	136.95 MB	<a href="#">jdk-8u281-linux-i586.tar.gz</a>
Linux x64 RPM Package	108.06 MB	<a href="#">jdk-8u281-linux-x64.rpm</a>
Linux x64 Compressed Archive	137.06 MB	<a href="#">jdk-8u281-linux-x64.tar.gz</a>
macOS x64	205.26 MB	<a href="#">jdk-8u281-macosx-x64.dmg</a>
Solaris SPARC 64-bit (SVR4 package)	125.96 MB	<a href="#">jdk-8u281-solaris-sparcv9tar.Z</a>
Solaris SPARC 64-bit	88.77 MB	<a href="#">jdk-8u281-solaris-sparcv9tar.gz</a>
Solaris x64 (SVR4 package)	134.68 MB	<a href="#">jdk-8u281-solaris-x64tar.Z</a>
Solaris x64	92.66 MB	<a href="#">jdk-8u281-solaris-x64tar.gz</a>
Windows x86	154.69 MB	<a href="#">jdk-8u281-windows-i586.exe</a>
Windows x64	166.97 MB	<a href="#">jdk-8u281-windows-x64.exe</a>

Gambar 3. 2 Versi Aplikasi JDK 8

3. Tekan kotak untuk menerima perjanjian lisensi



Gambar 3. 3 Accept License Agreement

4. Kemudian, tekan kotak hijau untuk mengunduhnya.
5. Tunggu sampai aplikasi selesai diunduh.

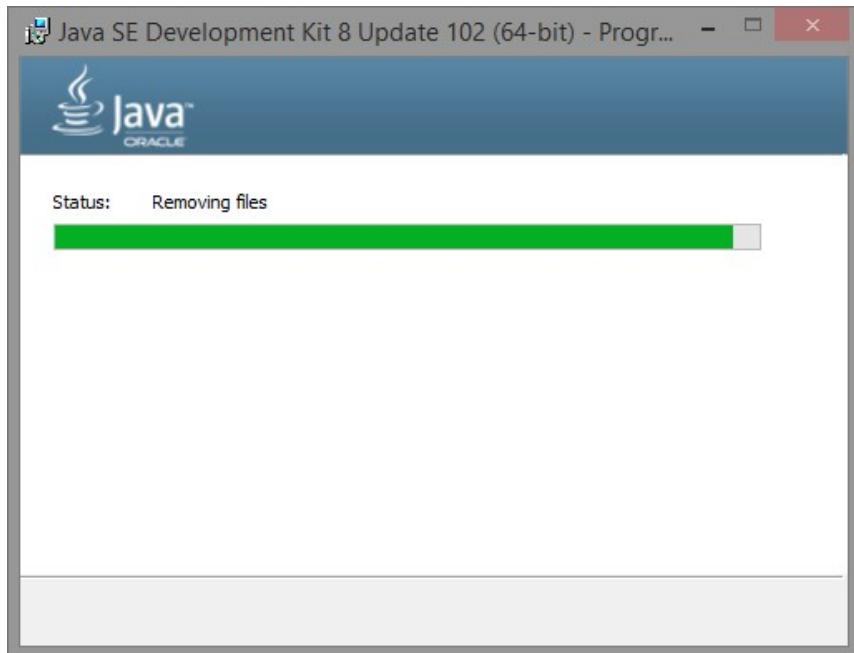
## B. Instalasi Aplikasi JDK

1. Klik dua kali aplikasi yang telah diunduh, lalu klik next.



Gambar 3. 4 Instal JDK 8 1

2. Kemudian tunggu proses instalasi selesai.



Gambar 3. 5 Instalasi JDK 8 2



Gambar 3. 6 Instalasi JDK 8 3

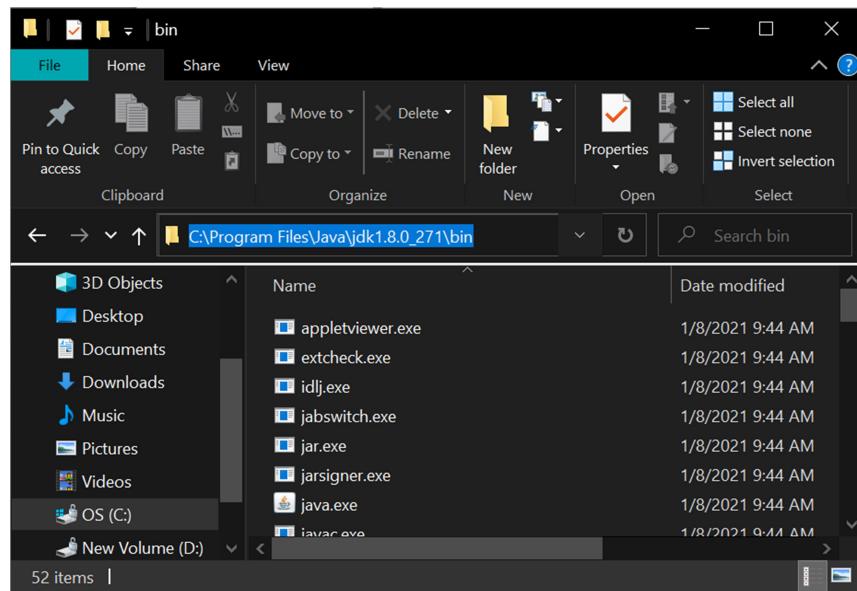
3. Setelah selesai, klik Close untuk menutup proses instalasi.



Gambar 3. 7 Instalasi JDK 8 4

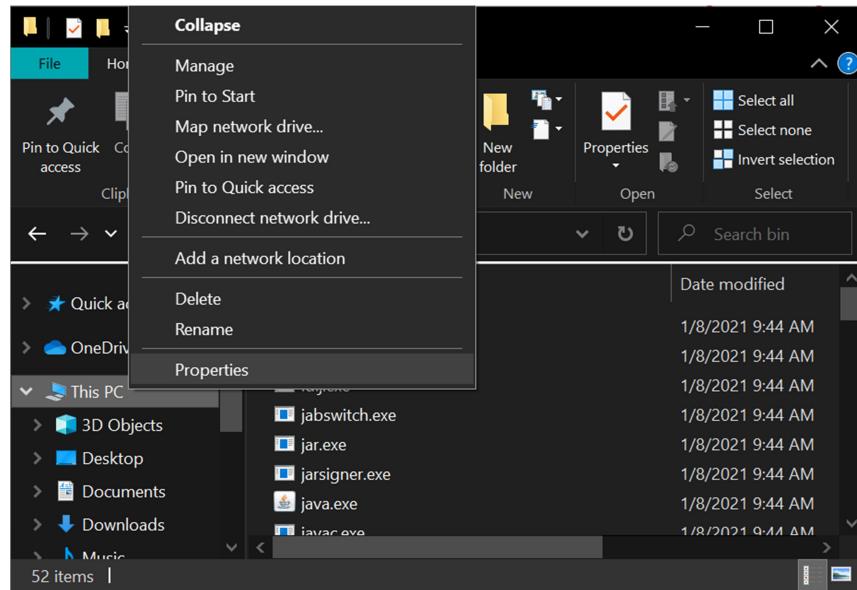
### C. Setting Path

#### 1. Salin lokasi folder instalasi JDK 8



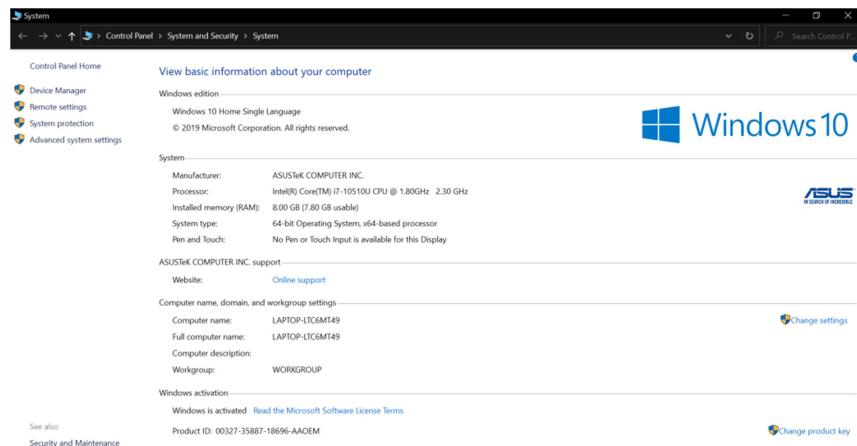
Gambar 3. 8 Salin Lokasi JDK 8

#### 2. Selanjutnya klik kanan pada This PC, kemudian pilih Properties.



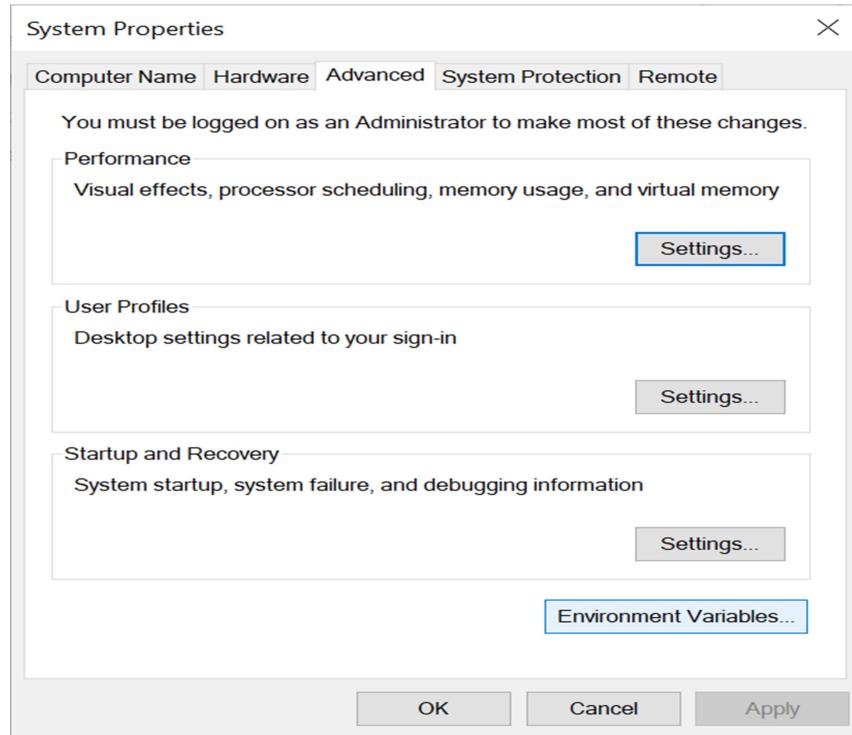
Gambar 3. 9 Membuka Jendela System

3. Setelah itu akan muncul Jendela baru (System) dan pilih Advanced System Settings.



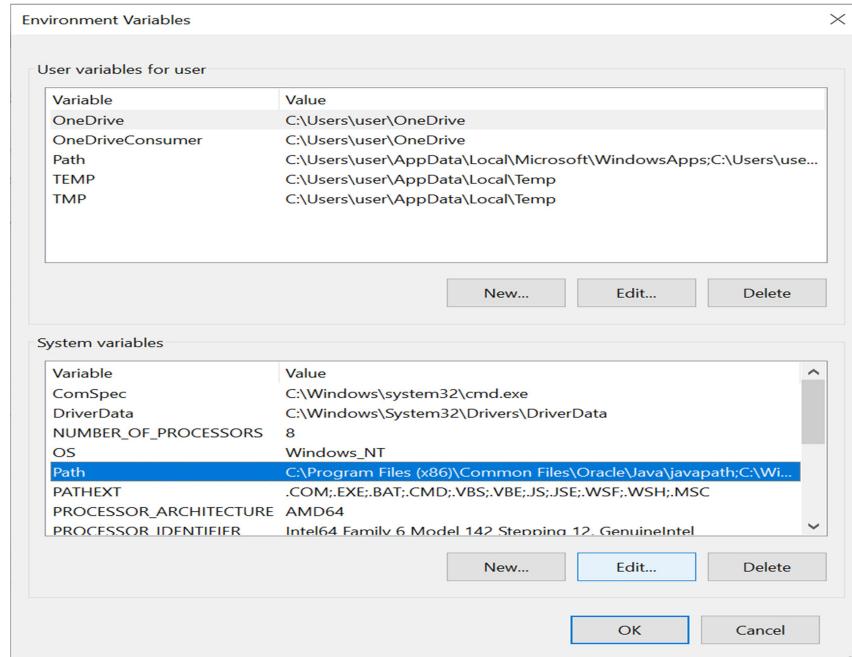
Gambar 3. 10 Jendela System

4. Selanjutnya pilih Environment Variables.



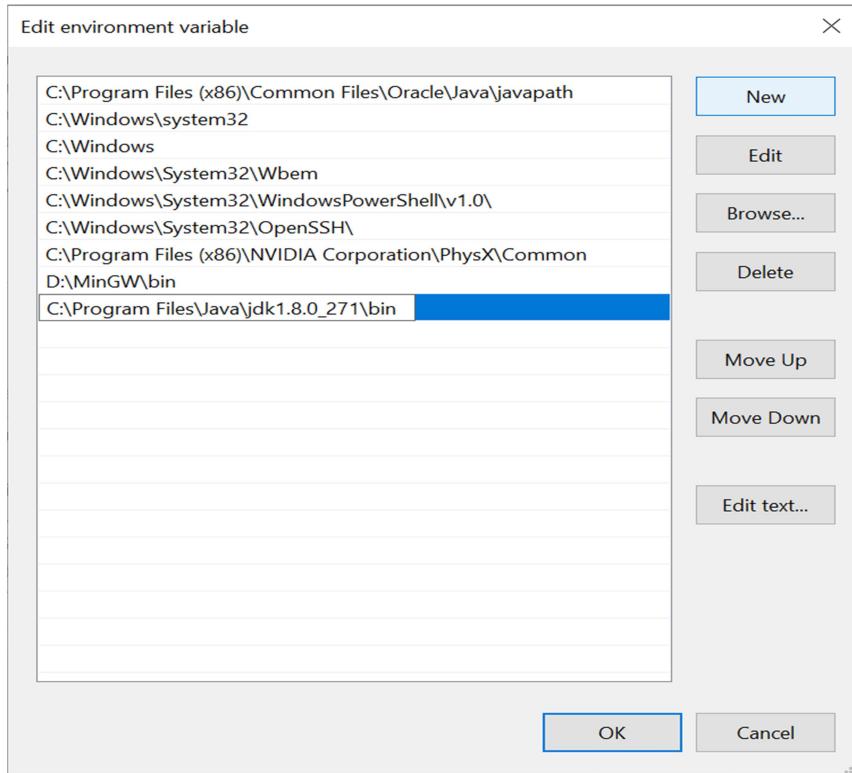
Gambar 3. 11 Jendela System Properties

5. Pada bagian System Variables cari Path kemudian klik Edit.



Gambar 3. 12 Jendela Environment Variabel

6. Maka akan terbuka window baru yaitu Edit System Variable.

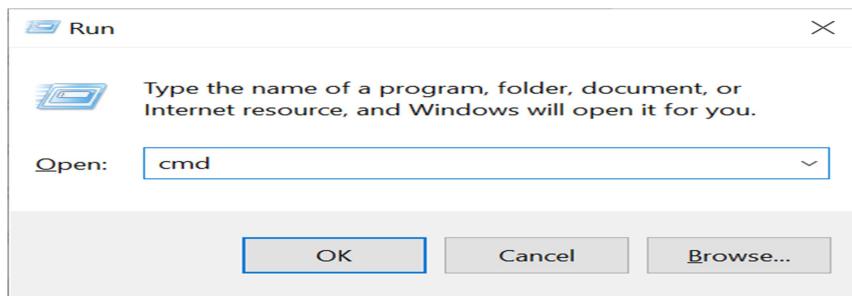


Gambar 3. 13 Jendela Edit System Variable

7. Klik New kemudian paste alamat lokasi folder JDK 8 yang telah di instal sebelumnya.
8. Kemudian klik OK disemua jendela yang telah terbuka.
9. Instalasi JDK 8 telah selesai.

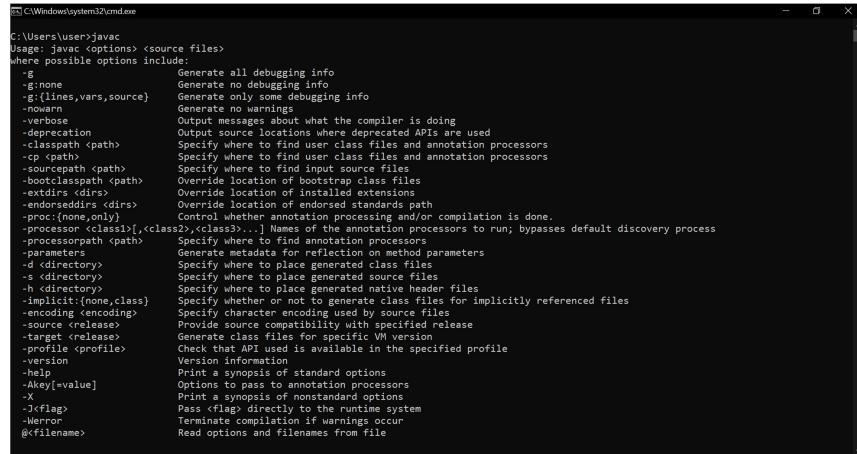
#### D. Menguji Instalasi JDK 8

1. Buka jendela Run dengan menekan logo windows dan R secara bersamaan.



Gambar 3. 14 Jendela Run

2. Ketik “cmd” kemudian klik OK.
3. Di jendela Command Prompt ketik “javac”, lalu enter.



```
C:\Users\user>javac
Usage: javac <options> <source files>
where possible options include:
  -g                         Generate all debugging info
  -g:none                     Generate no debugging info
  -g:{lines,vars,source}        Generate only some debugging info
  -nowarn                     Generate no warnings
  -verbose                    Output messages about what the compiler is doing
  -deprecation                Output source locations where deprecated APIs are used
  -classpath <path>           Specify where to find user class files and annotation processors
  -cp <path>                  Specify where to find user class files and annotation processors
  -sourcepath <path>          Specify where to find input source files
  -bootclasspath <path>       Override location of bootstrap class files
  -extdirs <dirs>             Override location of installed extensions
  -endorseddirs <dirs>        Override location of endorsed standard path
  -proc:{none,only}            Control whether annotation processing and/or compilation is done.
  -processor <class1>,<class2>,... Names of the annotation processors to run; bypasses default discovery process
  -processorpath <path>        Specify where to find annotation processors
  -parameters                 Generate metadata for reflection on method parameters
  -d <directory>              Specify where to place generated class files
  -s <directory>              Specify where to place generated source files
  -h <directory>              Specify where to place generated native header files
  -implicit:{none,class}      Specify whether or not to generate class files for implicitly referenced files
  -encoding <encoding>         Specify character encoding used by source files
  -source <release>            Provide source compatibility with specified release
  -target <release>            Generate class files for specific VM version
  -profile <profile>           Check that the profile is available in the specified profile
  -version                     Version information
  -help                        Print a synopsis of standard options
  -Akey[=value]                Options to pass to annotation processors
  -X                          Print a synopsis of nonstandard options
  -J<flag>                   Pass <flag> directly to the runtime system
  -Werror                      Terminate compilation if warnings occur
  @<filename>                 Read options and filenames from file
```

Gambar 3. 15 Penginstalan JDK Berhasil

4. Jika muncul tampilan seoerti gambar di atas, artinya penginstalan aplikasi JDK 8 berhasil.

## E. Program Aplikasi Perhitungan Luas Bangun Datar

1. Komentar yang berisi keterangan judul, nama, NIM, kelas, mata kuliah mahasiswa dan nama dosen pengampu.

```
1  /*
2   *
3   *   NAMA : ATIQAH NAJWA ANGGRAINI
4   *   NIM  : 202013024
5   *   KELAS : TIF 1B
6   *   MATA KULIAH : PRAKTIK ALGORITMA
7   *   DOSEN PENGAMPU : SLAMET TRIYANTO S. ST.
8   *
9  */
```

Gambar 3. 16 Source Code 1

2. Deklarasikan library untuk scanner. Membuat class dengan nama “tugas\_Uas\_Perhitungan\_Luas\_Bangun\_Datar”. Membuat method utama yang berisi judul awal aplikasi dan menuju ke method login\_program().

```

11 import java.util.*;
12 public class tugas_Uas_Perhitungan_Luas_Bangun_Datar{
13     public static void main (String[] args){
14
15     // Tampilan Awal
16
17     System.out.println("= = = = = = = = = = = = = = = = = = = = = =");
18     System.out.println("-----PROGRAM PERHITUNGAN LUAS BANGUN DATAR-----");
19     System.out.println("= = = = = = = = = = = = = = = = = = = = = = =");
20     login_pogram();
21 }
22 }
```

Gambar 3. 17 Source Code 2

3. Method `login_program()` berisi inisialisasi variabel “`data[]`” yang merupakan array dengan dua elemen, bertipe data string. Variabel “`username`” dan “`password`” juga bertipe data string. Terdapat perintah untuk menginputkan nilai ke variabel “`username`” dan “`password`” menggunakan scanner. Terdapat percabangan dengan kondisi jika variabel “`username`” nilainya sama dengan variabel “`data[0]`” dan “`password`” nilainya sama dengan variabel “`data[1]`”. Jika kondisi benar, maka akan muncul tulisan “selamat datang” dan masuk ke method `menu_utama()`. Jika kondisi salah, maka akan muncul tulisan “Maaf Username dan Password tidak tersedia” dan “Silahkan Masukkan Username dan Password Kembali”. Lalu kembali masuk ke method `login_program()`.

```

23 // Masukkan Username dan Password
24
25     static void login_pogram(){
26         String data[]={“atiqah”, “202013024”};
27         String username,password;
28         Scanner masuk=new Scanner(System.in);
29         System.out.print(“\nUsername\t: ”);
30         username=masuk.nextLine();
31         System.out.print(“\nPassword\t: ”);
32         password=masuk.nextLine();
33
34         /**
35          * Percabangan untuk mengecek keberadaan data yang sudah kita inputkan di atas
36          */
37
38         if(username.equals(data[0])&&password.equals(data[1])){
39
40             System.out.println();
41             System.out.println(“= = = = = = = = = = = = = = = = = = = = = =”);
42             System.out.println(“-----Selamat Datang-----”);
43             System.out.println(“= = = = = = = = = = = = = = = = = = = = = =”);
44             System.out.println();
45             menu_utama();
46         }else{
47             System.out.println();
48             System.out.println(“Maaf Username dan Password tidak tersedia”);
49             System.out.println(“Silahkan Masukkan Username dan Password Kembali”);
50             login_pogram();
51         }
52 }
```

Gambar 3. 18 Source Code 3

4. Method menu\_utama() berisi variabel “menu” dengan tipe data integer.

Lalu terdapat perintah untuk menampilkan daftar menu. Kemudian ada perintah untuk menginputkan nilai ke variabel “menu”. Jika variabel “menu” bernilai 1 maka akan masuk ke method perhitungan(). Jika variabel “menu” bernilai 2 maka akan masuk ke method keluar(). Jika tidak ada kondisi yang memenuhi maka akan muncul tulisan “Maaf Pilihan tidak tersedia” dan “Silahkan Pilih Menu Sesuai yang Ditetapkan : ”. Kemudian masuk kembali ke method menu\_utama().

```

55 // Pemilihan di menu awal
56
57 static void menu_utama(){
58     int menu;
59     Scanner terima=new Scanner(System.in);
60     System.out.println("");
61     System.out.println("");
62     System.out.println("      =====");
63     System.out.println("      |          M E N U      |");
64     System.out.println("      =====");
65     System.out.println("      | 1. Menghitung Luas Bangun Datar |");
66     System.out.println("      | 2. Keluar                  |");
67     System.out.println("      =====");
68     System.out.println("-----");
69
70     System.out.print("Silahkan Masukan Pilihan Menu\t: ");
71     menu=terima.nextInt();
72     System.out.println();
73     System.out.println();
74     System.out.println();
75
76     if(menu==1){
77         perhitungan();
78     }else if(menu==2){
79         keluar();
80     }else{
81         System.out.println("Maaf Pilihan tidak tersedia");
82         System.out.print("Silahkan Pilih Menu Sesuai yang Ditetapkan\t: ");
83         menu_utama();
84     }
85 }
```

Gambar 3. 19 Source Code 4

5. Di method perhitungan() inisialisasikan variabel “daftar” dengan tipe data integer. Kemudian terdapat perintah untuk menampilkan daftar bangun datar, yaitu : persegi, persegi panjang, segitiga, lingkaran, jajar genjang, belah ketupat, trapesium, dan layang-layang. Dan menu sebelumnya untuk kembali ke method menu\_utama(). Jika variabel “daftar” bernilai sembilan maka akan masuk ke method menu\_utama(). Jika variabel “daftar” bernilai kurang dari sembilan maka akan masuk ke percabangan switch. Sebelum percabangan switch terlebih dahulu inisialissi variabel-variabel yang berhubungan dengan rumus luas bangun datar dan masukkan banyak bangun datar yang ingin dihitung ke variabel “banyak”.

```

87 // Pemilihan bangun datar yang ingin dihitung
88 static void perhitungan(){
89     int daftar;
90     Scanner terima=new Scanner(System.in);
91     System.out.println("-----");
92     System.out.println("-----| BANGUN DATAR |-----");
93     System.out.println("-----| 1. Persegi      |-----");
94     System.out.println("-----| 2. Persegi Panjang |-----");
95     System.out.println("-----| 3. Segitiga      |-----");
96     System.out.println("-----| 4. Lingkaran      |-----");
97     System.out.println("-----| 5. Jajar Genjang |-----");
98     System.out.println("-----| 6. Belah Ketupat |-----");
99     System.out.println("-----| 7. Trapesium      |-----");
100    System.out.println("-----| 8. Layang-Layang |-----");
101    System.out.println("-----| 9. Menu Sebelumnya |-----");
102    System.out.println("-----");
103    System.out.println("-----");
104    System.out.println("-----");
105    System.out.print("Silahkan Pilih Bangun Datar\t: ");
106    daftar=terima.nextInt();
107    if(daftar==9){
108        menu_utama();

```

Gambar 3. 20 Source Code 5

```

109 }if(daftar<9){
110     int banyak;
111     System.out.print("Silahkan Masukan Banyak Bangun yang Dihitung\t: ");
112     banyak=terima.nextInt();
113     double luas[] =new double[banyak];
114     double jari[] =new double[banyak];
115     double tinggi[] =new double[banyak];
116     double alas[] =new double[banyak];
117     double diagonal1[] =new double[banyak];
118     double diagonal2[] =new double[banyak];
119     double sisi[] =new double[banyak];
120     double sisiAtas[] =new double[banyak];
121     double sisiBawah[] =new double[banyak];
122     double panjang[] =new double[banyak];
123     double lebar[] =new double[banyak];
124     double pi=3.14;
125     int i;

```

Gambar 3. 21 Source Code 6

6. Case 1 berisi program untuk menghitung luas persegi. Dengan perulangan for akan menghitung luas bangun persegi sebanyak nilai variabel "banyak". Dalam perulangan for terdapat perintah untuk menginputkan nilai ke variabel "sisi". Kemudian luas persegi yang telah dihitung akan diurutkan secara *ascending* dengan metode *bubble sort*. Setelah itu masukkan ke method lanjut().

```

127 switch(daftar){
128 case 1:
129     System.out.println("");
130     System.out.println("*****");
131     System.out.println("      PROGRAM MENGHITUNG LUAS PERSEGI ");
132     System.out.println("*****");
133     System.out.println();
134     for(i=0;i

```

Gambar 3. 22 Source Code 7

7. Case 2 berisi program untuk menghitung luas persegi panjang. Dengan perulangan for akan menghitung luas bangun persegi panjang sebanyak nilai variabel “banyak”. Dalam perulangan for terdapat perintah untuk menginputkan nilai ke variabel ”panjang” dan ”lebar”. Kemudian luas persegi panjang yang telah dihitung akan diurutkan secara *ascending* dengan metode *bubble sort*. Setelah itu masukkan ke method lanjut().

Gambar 3. 23 Source Code 8

```

167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
    case 2:
        System.out.println("");
        System.out.println("*****");
        System.out.println(" PROGRAM MENGHITUNG LUAS PERSEGI PANJANG ");
        System.out.println("*****");
        System.out.println();
        for(i=0;i

```

8. Case 3 berisi program untuk menghitung luas segitiga. Dengan perulangan for akan menghitung luas bangun segitiga sebanyak nilai variabel “banyak”. Dalam perulangan for terdapat perintah untuk menginputkan nilai ke variabel ”alas” dan ”tinggi”. Kemudian luas segitiga yang telah dihitung akan diurutkan secara *ascending* dengan metode *bubble sort*. Setelah itu masukkan ke method lanjut().

```

206
207     case 3:
208         System.out.println("");
209         System.out.println("***** PROGRAM MENGHITUNG LUAS SEGITIGA *****");
210         System.out.println("*****");
211         System.out.println();
212         for(i=0;i<banyak;i++){
213
214             System.out.print("Segitiga "+(i+1));
215             System.out.println("-----");
216             System.out.print("Silahkan Masukkan Alas = ");
217             alas[i]=terima.nextDouble();
218
219             System.out.print("Silahkan Masukkan Tinggi = ");
220             tinggi[i]=terima.nextDouble();
221
222             luas[i]=(alas[i]*tinggi[i])/2;
223
224             System.out.println("");
225             System.out.println("***** LUAS SEGITIGA = "+(i+1)+" = "+luas[i]);
226             System.out.println("*****");
227             System.out.println();
228         }
229
230         for (i=0;i<luas.length;i++){
231             for(int k=0;k<(luas.length-1);k++){
232                 if(luas[k]<luas[k+1]){
233                     penampung=luas[k];
234                     luas[k]=luas[k+1];
235                     luas[k+1]=penampung;
236                 }
237             }
238         }
239         System.out.println("Luas Segitiga Diurutkan Dari yang Terkecil : ");
240         for(i=0;i<luas.length;i++){
241             System.out.println((i+1)+". "+luas[i]);
242         }
243         lanjut();
244     break;

```

Gambar 3. 24 Source Code 9

9. Case 4 berisi program untuk menghitung luas lingkaram. Dengan perulangan for akan menghitung luas bangun lingkaran sebanyak nilai variabel “banyak”. Dalam perulangan for terdapat perintah untuk menginputkan nilai ke variabel ”jari”. Kemudian luas lingkaran yang telah dihitung akan diurutkan secara ascending dengan metode bubble sort. Setelah itu masukkan ke method lanjut().

```

245
246     case 4:
247         System.out.println("");
248         System.out.println("***** PROGRAM MENGHITUNG LUAS LINGKARAN *****");
249         System.out.println("*****");
250         System.out.println();
251         for(i=0;i<banyak;i++){
252
253             System.out.print("Lingkaran "+(i+1));
254             System.out.println("-----");
255             System.out.print("Silahkan Masukkan Jari-Jari = ");
256             jari[i]=terima.nextDouble();
257
258             luas[i]=pi*jari[i]*jari[i];
259
260             System.out.println("");
261             System.out.println("***** LUAS LINGKARAN "+(i+1)+" = "+luas[i]);
262             System.out.println("*****");
263             System.out.println();
264         }
265
266         for (i=0;i<luas.length;i++){
267             for(int k=0;k<(luas.length-1);k++){
268                 if(luas[k]<luas[k+1]){
269                     penampung=luas[k];
270                     luas[k]=luas[k+1];
271                     luas[k+1]=penampung;
272                 }
273             }
274         }
275         System.out.println("Luas Lingkaran Diurutkan Dari yang Terkecil : ");
276         for(i=0;i<luas.length;i++){
277             System.out.println((i+1)+". "+luas[i]);
278         }
279         lanjut();
280     break;

```

Gambar 3. 25 Source Code 10

10. Case 5 berisi program untuk menghitung luas jajar genjang. Dengan perulangan for akan menghitung luas bangun jajar genjang sebanyak nilai variabel “banyak”. Dalam perulangan for terdapat perintah untuk menginputkan nilai ke variabel ”alas” dan ”tinggi”. Kemudian luas jajar genjang yang telah dihitung akan diurutkan secara ascending dengan metode bubble sort. Setelah itu masukkan ke method lanjut().

```

281      case 5:
282          System.out.println("");
283          System.out.println("*****");
284          System.out.println(" PROGRAM MENGHITUNG LUAS JAJAR GENJANG ");
285          System.out.println("*****");
286          System.out.println();
287          for(i=0;i<banyak;i++){
288
289              System.out.print("Jajar Genjang "+(i+1));
290              System.out.print("-----");
291              System.out.print("Silahkan Masukkan Alas = ");
292              alas[i]=terima.nextDouble();
293
294              System.out.print("Silahkan Masukkan Tinggi = ");
295              tinggi[i]=terima.nextDouble();
296
297              luas[i]=(alas[i]*tinggi[i])/2;
298
299              System.out.println("");
300              System.out.println("*****");
301              System.out.println("LUAS JAJAR GENJANG = "+(i+1)+" = "+luas[i]);
302              System.out.println("*****");
303              System.out.println();
304          }
305          for (i=0;i<luas.length;i++){
306              for(int k=0;k<(luas.length-1);k++){
307                  if(luas[k]>luas[k+1]){
308                      penampung=luas[k];
309                      luas[k]=luas[k+1];
310                      luas[k+1]=penampung;
311                  }
312              }
313          }
314          System.out.println("Luas Jajar Genjang Diurutkan Dari yang Terkecil : ");
315          for(i=0;i<luas.length;i++){
316              System.out.println((i+1)+" . "+luas[i]);
317          }
318          lanjut();
319      break;

```

Gambar 3. 26 Source Code 11

11. Case 6 berisi program untuk menghitung luas belah ketupat. Dengan perulangan for akan menghitung luas bangun belah ketupat sebanyak nilai variabel “banyak”. Dalam perulangan for terdapat perintah untuk menginputkan nilai ke variabel ”diagonal1” dan ”diagonal2”. Kemudian luas belah ketupat yang telah dihitung akan diurutkan secara ascending dengan metode bubble sort. Setelah itu masukkan ke method lanjut().

```

320     case 6:
321         System.out.println("");
322         System.out.println("*****");
323         System.out.println(" PROGRAM MENGHITUNG LUAS BELAH KETUPAT ");
324         System.out.println("*****");
325         System.out.println();
326         for(i=0;i

```

Gambar 3. 27 Source Code 12

12. Case 7 berisi program untuk menghitung luas trapesium. Dengan perulangan for akan menghitung luas bangun trapesium sebanyak nilai variabel “banyak”. Dalam perulangan for terdapat perintah untuk menginputkan nilai ke variabel ”sisiAtas”, ”sisiBawah” dan ”tinggi”. Kemudian luas segitiga yang telah dihitung akan diurutkan secara ascending dengan metode bubble sort. Setelah itu masukkan ke method lanjut().

```

359
360     case 7:
361         System.out.println("");
362         System.out.println("*****");
363         System.out.println(" PROGRAM MENGHITUNG LUAS TRAPESIUM ");
364         System.out.println("*****");
365         System.out.println();
366         for(i=0;i

```

Gambar 3. 28 Source Code 13

13. Case 8 berisi program untuk menghitung luas layang-layang. Dengan perulangan for akan menghitung luas bangun layang-layang sebanyak nilai variabel “banyak”. Dalam perulangan for terdapat perintah untuk menginputkan nilai ke variabel ”diagonal1” dan ”diagonal2”. Kemudian luas layang-layang yang telah dihitung akan diurutkan secara ascending dengan metode bubble sort. Setelah itu masukkan ke method lanjut().

```

397
398     case 8:
399         System.out.println("");
400         System.out.println("***** PROGRAM MENGHITUNG LUAS LAYANG-LAYANG *****");
401         System.out.println("*****");
402         System.out.println();
403         for(i=0;i<banyak;i++){
404
405             System.out.println("Layang-Layang "+(i+1));
406             System.out.println("-----");
407             System.out.print("Silahkan Masukkan Diagonal 1 = ");
408             diagonal1[i]=terima.nextDouble();
409
410             System.out.print("Silahkan Masukkan Diagonal 2 = ");
411             diagonal2[i]=terima.nextDouble();
412
413             luas[i]=(diagonal1[i]*diagonal2[i])/2;
414
415             System.out.println("");
416             System.out.println("*****");
417             System.out.println("LUAS LAYANG-LAYANG = "+(i+1)+" = "+luas[i]);
418             System.out.println("*****");
419             System.out.println();
420         }
421         for (i=0;i<luas.length;i++){
422             for(int k=0;k<(luas.length-1);k++){
423                 if(luas[k]>luas[k+1]){
424                     penampung=luas[k];
425                     luas[k]=luas[k+1];
426                     luas[k+1]=penampung;
427                 }
428             }
429         }
430         System.out.println("Luas Layang-Layang Diurutkan Dari yang Terkecil : ");
431         for(i=0;i<luas.length;i++){
432             System.out.println((i+1)+". "+luas[i]);
433         }
434         lanjut();
435     break;
436 }
```

Gambar 3. 29 Source code 14

14. Bagian default akan menampilkan “Pilihan yang anda masukkan tidak tersedia” dan “Silahkan pilih bangun datar yang tertera”. Kemudian masuk ke method perhitungan().

```

436         default :
437             System.out.println("Pilihan yang anda masukkan tidak tersedia");
438             System.out.println("Silahkan pilih bangun datar yang tertera");
439             System.out.println();
440             perhitungan();
441             break;
442         }
443     }
444 }
```

Gambar 3. 30 Source Code 15

15. Method lanjut() berisi variabel “ulang” dengan tipe data integer. Dalam method ini akan menampilkan tulisan “Perhitungan Telah Selesai” dan pilihan untuk melanjutkan aplikasi. Jika variabel “ulang” bernilai satu maka akan masuk method perhitungan(). Jika variabel “ulang” bernilai

dua maka akan masuk method menu\_utama(). Jika variabel "ulang" bernilai tiga maka akan masuk method keluar().

```
446 static void lanjut(){
447     Scanner terima=new Scanner(System.in);
448     int ulang;
449     System.out.println();
450     System.out.println("-----");
451     System.out.println("|~~~~~Perhitungan Telah Selesai~~~~~|");
452     System.out.println("-----");
453     System.out.println();
454     System.out.println("Silahkan Masukkan Pilihan Untuk Melanjutkan");
455     System.out.println("    Tekan 1 untuk Menu Sebelumnya");
456     System.out.println("    Tekan 2 untuk kembali ke Menu");
457     System.out.println("    Tekan 3 untuk Keluar");
458     System.out.print("Masukan Pilihan : ");
459     ulang=terima.nextInt();
460     if(ulang==1){
461         perhitungan();
462     }if(ulang==2){
463         menu_utama();
464     }if(ulang==3){
465         keluar();
466     }
467     else{
468         System.out.println("Pilihan yang anda masukkan tidak tersedia")
469         lanjut();
470     }
471 }
```

Gambar 3.31 Source Code 16

16. Method `keluar()` berisi perintah untuk menampilkan “System Keluar” dan “Terima Kasih Sudah Berkunjung”. Kemudian terdapat perintah `System.exit(0)` yang berfungsi untuk keluar dari aplikasi.

```
472  
473     static void keluar(){  
474         System.out.println("-----");  
475         System.out.println(" | ~ |");  
476         System.out.println(" |       System Keluar   |");  
477         System.out.println(" | Terima Kasih Sudah Berkunjung |");  
478         System.out.println(" | ~ |");  
479         System.out.println("-----");  
480         System.exit(0);  
481     }  
482 }
```

Gambar 3.32 Source Code 17

#### **F. Tampilan Aplikasi**

- ## 1. Tampilan awal.

Gambar 3. 33 Tampilan Aplikasi 1

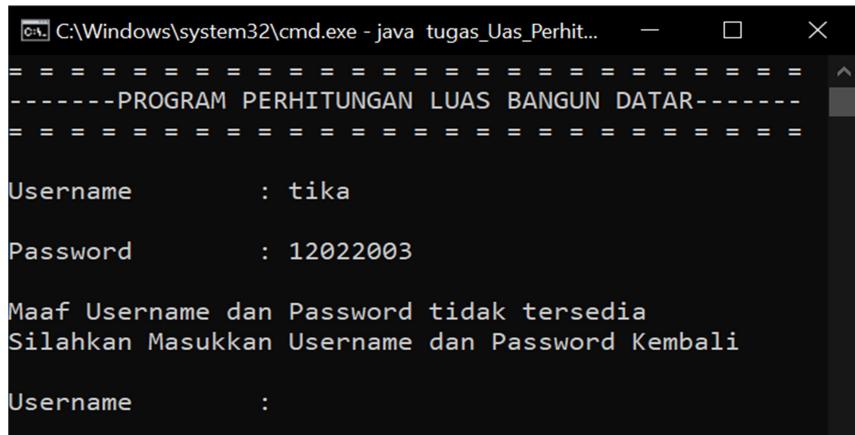
2. Username dan password yang dimasukkan benar.



```
C:\Windows\system32\cmd.exe - java tugas_Uas_Perhit...
= = = = = = = = = = = = = = = = = = = = = = = = = = = = =
-----PROGRAM PERHITUNGAN LUAS BANGUN DATAR-----
= = = = = = = = = = = = = = = = = = = = = = = = = = = = =
Username : atiqah
Password : 202013024
= = = = = = = = = = = = = = = = = = = = = = = = = = = =
-----Selamat Datang-----
= = = = = = = = = = = = = = = = = = = = = = = = = = = =
=====
|           M E N U           |
=====
| 1. Menghitung Luas Bangun Datar | 
| 2. Keluar                  | 
-----
Silahkan Masukan Pilihan Menu :
```

Gambar 3. 34 Tampilan Aplikasi 2

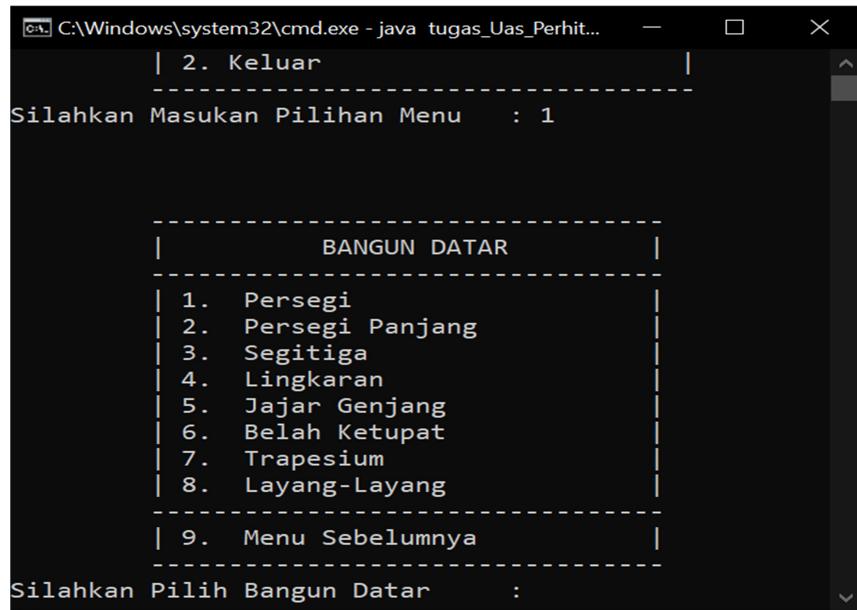
3. Username dan password yang dimasukkan salah.



```
C:\Windows\system32\cmd.exe - java tugas_Uas_Perhit...
= = = = = = = = = = = = = = = = = = = = = = = = = = = = =
-----PROGRAM PERHITUNGAN LUAS BANGUN DATAR-----
= = = = = = = = = = = = = = = = = = = = = = = = = = = = =
Username : tika
Password : 120222003
Maaf Username dan Password tidak tersedia
Silahkan Masukkan Username dan Password Kembali
Username :
```

Gambar 3. 35 Tampilan Aplikasi 3

4. Jika memilih menu 1.



C:\Windows\system32\cmd.exe - java tugas\_Uas\_Perhit...

| 2. Keluar

Silahkan Masukan Pilihan Menu : 1

BANGUN DATAR

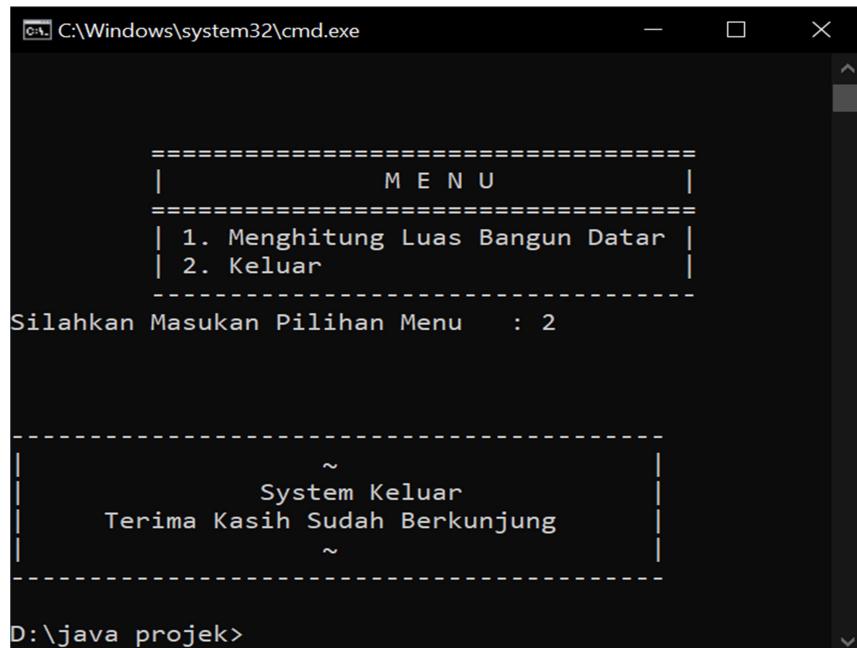
1. Persegi  
2. Persegi Panjang  
3. Segitiga  
4. Lingkaran  
5. Jajar Genjang  
6. Belah Ketupat  
7. Trapesium  
8. Layang-Layang

9. Menu Sebelumnya

Silahkan Pilih Bangun Datar :

Gambar 3. 36 Tampilan Aplikasi 4

5. Jika memilih menu 2.



C:\Windows\system32\cmd.exe

MENU

1. Menghitung Luas Bangun Datar  
2. Keluar

Silahkan Masukan Pilihan Menu : 2

~  
System Keluar  
Terima Kasih Sudah Berkunjung  
~

D:\java projek>

Gambar 3. 37 Tampilan Aplikasi 5

6. Jika memilih bangun datar 1.

```
C:\Windows\system32\cmd.exe - java tugas_Uas_Perhit...
Silahkan Pilih Bangun Datar : 1
Silahkan Masukan Banyak Bangun yang Dihitung : 3
*****
PROGRAM MENGHITUNG LUAS PERSEGI
*****
Persegi 1
-----
Silahkan Masukkan Sisi = 2
*****
LUAS PERSEGI 1 = 4.0
*****
Persegi 2
-----
Silahkan Masukkan Sisi = 5
*****
LUAS PERSEGI 2 = 25.0
*****
Persegi 3
-----
Silahkan Masukkan Sisi = 2
*****
LUAS PERSEGI 3 = 4.0
*****
```

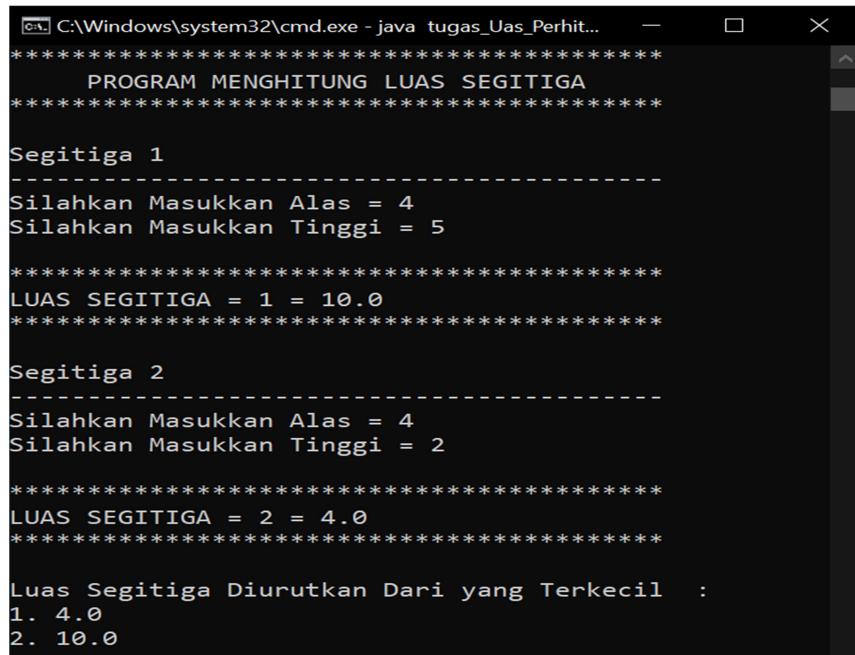
Gambar 3. 38 Tampilan Aplikasi 6

7. Jika memilih bangun datar 2.

```
C:\Windows\system32\cmd.exe - java tugas_Uas_Perhit...
-----
Silahkan Pilih Bangun Datar : 2
Silahkan Masukan Banyak Bangun yang Dihitung : 2
*****
PROGRAM MENGHITUNG LUAS PERSEGI PANJANG
*****
Persegi Panjang 1
-----
Silahkan Masukkan Panjang = 4
Silahkan Masukkan Lebar = 5
*****
LUAS PERSEGI PANJANG 1 = 20.0
*****
Persegi Panjang 2
-----
Silahkan Masukkan Panjang = 1
Silahkan Masukkan Lebar = 2
*****
LUAS PERSEGI PANJANG 2 = 2.0
*****
Luas Persegi Panjang Diurutkan Dari yang Terkecil :
1. 2.0
2. 20.0
```

Gambar 3. 39 Tampilan Aplikasi 7

8. Jika memilih bangun datar 3.



```
C:\Windows\system32\cmd.exe - java tugas_Uas_Perhit...
*****
PROGRAM MENGHITUNG LUAS SEGITIGA
*****
Segitiga 1
-----
Silahkan Masukkan Alas = 4
Silahkan Masukkan Tinggi = 5

*****
LUAS SEGITIGA = 1 = 10.0
*****

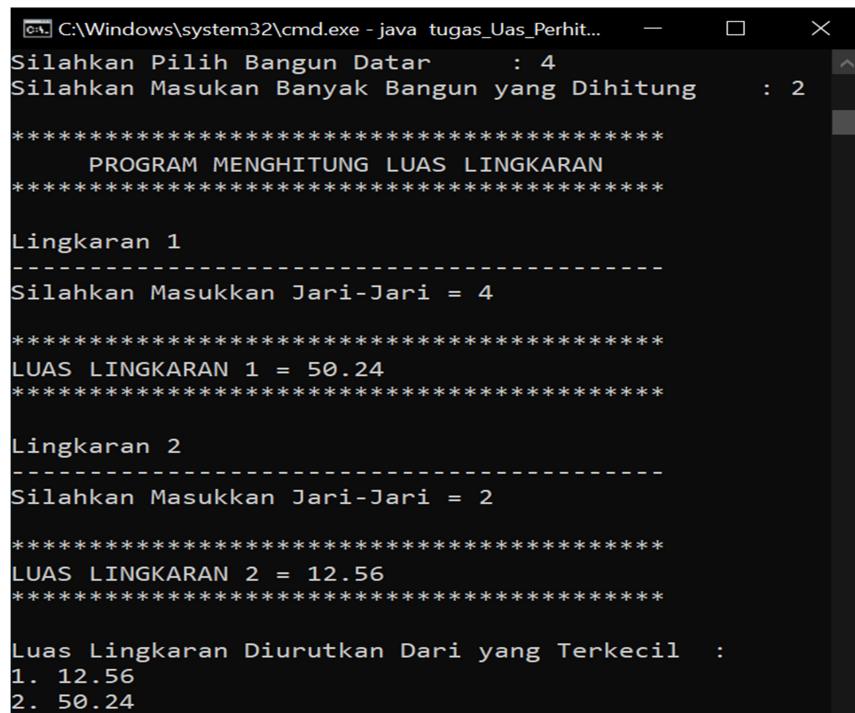
Segitiga 2
-----
Silahkan Masukkan Alas = 4
Silahkan Masukkan Tinggi = 2

*****
LUAS SEGITIGA = 2 = 4.0
*****

Luas Segitiga Diurutkan Dari yang Terkecil :
1. 4.0
2. 10.0
```

Gambar 3. 40 Tampilan Aplikasi 8

9. Jika memilih bangun datar 4.



```
C:\Windows\system32\cmd.exe - java tugas_Uas_Perhit...
*****
Silahkan Pilih Bangun Datar : 4
Silahkan Masukan Banyak Bangun yang Dihitung : 2

*****
PROGRAM MENGHITUNG LUAS LINGKARAN
*****

Lingkaran 1
-----
Silahkan Masukkan Jari-Jari = 4

*****
LUAS LINGKARAN 1 = 50.24
*****
```

```
Lingkaran 2
-----
Silahkan Masukkan Jari-Jari = 2

*****
LUAS LINGKARAN 2 = 12.56
*****
```

```
Luas Lingkaran Diurutkan Dari yang Terkecil :
1. 12.56
2. 50.24
```

Gambar 3. 41 Tampilan Aplikasi 9

10. Jika memilih bangun datar 5.

```
C:\Windows\system32\cmd.exe -java tugas_Uas_Perhit...
Silahkan Pilih Bangun Datar : 5
Silahkan Masukan Banyak Bangun yang Dihitung : 2
*****
PROGRAM MENGHITUNG LUAS JAJAR GENJANG
*****
Jajar Genjang 1
-----
Silahkan Masukkan Alas = 4
Silahkan Masukkan Tinggi = 5
*****
LUAS JAJAR GENJANG = 1 = 10.0
*****
Jajar Genjang 2
-----
Silahkan Masukkan Alas = 3
Silahkan Masukkan Tinggi = 2
*****
LUAS JAJAR GENJANG = 2 = 3.0
*****
Luas Jajar Genjang Diurutkan Dari yang Terkecil :
1. 3.0
2. 10.0
```

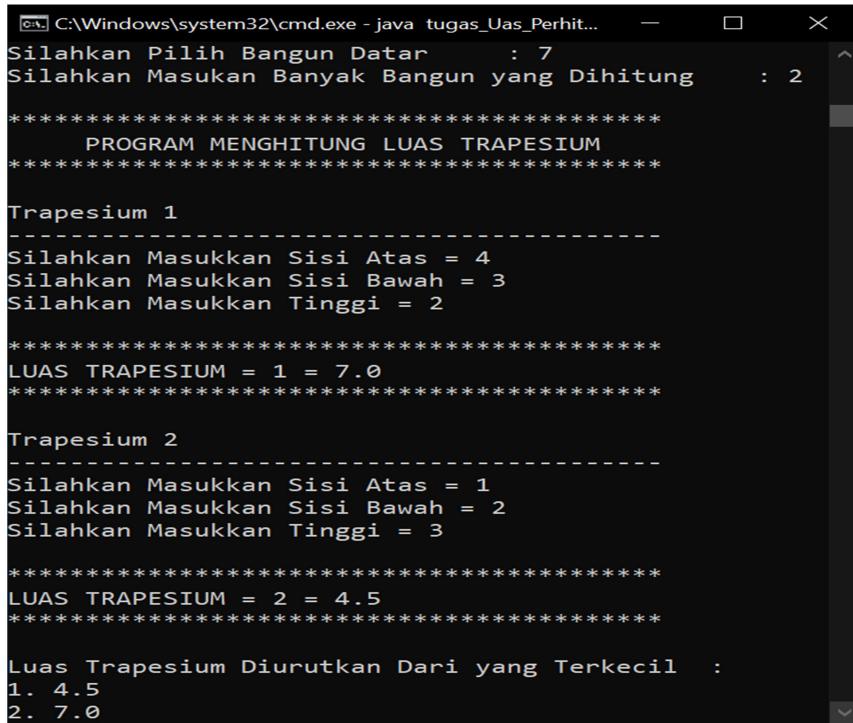
Gambar 3. 42 Tampilan Aplikasi 10

11. Jika memilih bangun datar 6.

```
C:\Windows\system32\cmd.exe -java tugas_Uas_Perhit...
Silahkan Pilih Bangun Datar : 6
Silahkan Masukan Banyak Bangun yang Dihitung : 2
*****
PROGRAM MENGHITUNG LUAS BELAH KETUPAT
*****
Belah Ketupat 1
-----
Silahkan Masukkan Diagonal 1 = 4
Silahkan Masukkan Diagonal 2 = 2
*****
LUAS BELAH KETUPAT = 1 = 4.0
*****
Belah Ketupat 2
-----
Silahkan Masukkan Diagonal 1 = 1
Silahkan Masukkan Diagonal 2 = 3
*****
LUAS BELAH KETUPAT = 2 = 1.5
*****
Luas Belah Ketupat Diurutkan Dari yang Terkecil :
1. 1.5
2. 4.0
```

Gambar 3. 43 Tampilan Aplikasi 11

12. Jika memilih bangun datar 7.



```
C:\Windows\system32\cmd.exe - java tugas_Uas_Perhit...
Silahkan Pilih Bangun Datar : 7
Silahkan Masukan Banyak Bangun yang Dihitung : 2
*****
PROGRAM MENGHITUNG LUAS TRAPESIUM
*****

Trapesium 1
-----
Silahkan Masukkan Sisi Atas = 4
Silahkan Masukkan Sisi Bawah = 3
Silahkan Masukkan Tinggi = 2

*****
LUAS TRAPESIUM = 1 = 7.0
*****
```

```
Trapesium 2
-----
Silahkan Masukkan Sisi Atas = 1
Silahkan Masukkan Sisi Bawah = 2
Silahkan Masukkan Tinggi = 3

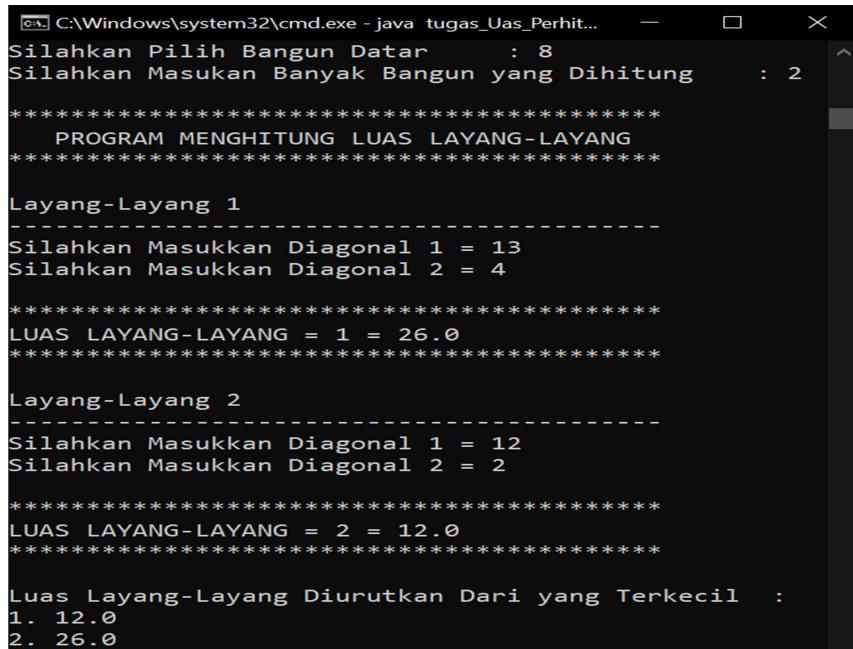
*****
LUAS TRAPESIUM = 2 = 4.5
*****
```

```
Luas Trapesium Diurutkan Dari yang Terkecil :
1. 4.5
2. 7.0
```

Gambar 3. 44 Tampilan Aplikasi 12

13. Jika memilih bangun datar 8.



```
C:\Windows\system32\cmd.exe - java tugas_Uas_Perhit...
Silahkan Pilih Bangun Datar : 8
Silahkan Masukan Banyak Bangun yang Dihitung : 2
*****
PROGRAM MENGHITUNG LUAS LAYANG-LAYANG
*****
```

```
Layang-Layang 1
-----
Silahkan Masukkan Diagonal 1 = 13
Silahkan Masukkan Diagonal 2 = 4

*****
LUAS LAYANG-LAYANG = 1 = 26.0
*****
```

```
Layang-Layang 2
-----
Silahkan Masukkan Diagonal 1 = 12
Silahkan Masukkan Diagonal 2 = 2

*****
LUAS LAYANG-LAYANG = 2 = 12.0
*****
```

```
Luas Layang-Layang Diurutkan Dari yang Terkecil :
1. 12.0
2. 26.0
```

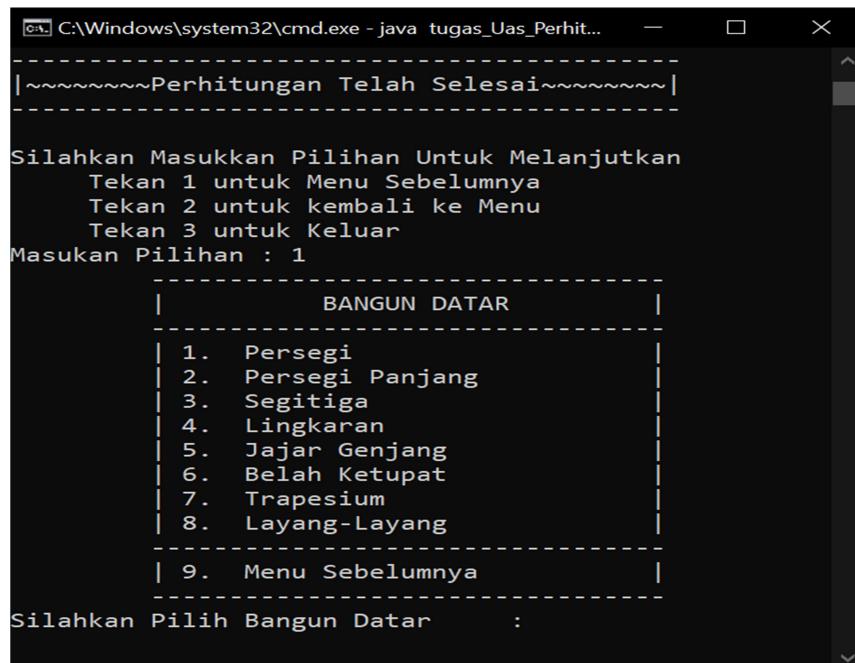
Gambar 3. 45 Tampilan Aplikasi 13

14. Jika memilih menu 9 (menu sebelumnya).



**Gambar 3.46 Tampilan Aplikasi 14**

15. Jika memilih 1 (menu sebelumnya) setelah proses perhitungan



Gambar 3. 47 Tampilan Aplikasi 15

16. Jika memilih 2 (kembali ke Menu) setelah proses perhitungan.

```
Silahkan Masukkan Pilihan Untuk Melanjutkan
Tekan 1 untuk Menu Sebelumnya
Tekan 2 untuk kembali ke Menu
Tekan 3 untuk Keluar
Masukan Pilihan : 2

=====
|           M E N U           |
=====
| 1. Menghitung Luas Bangun Datar |
| 2. Keluar                      |
-----
Silahkan Masukan Pilihan Menu : -
```

Gambar 3. 48 Tampilan Aplikasi 16

17. Jika memilih 3 (keluar) setelah proses perhitungan.

```
Silahkan Masukkan Pilihan Untuk Melanjutkan
Tekan 1 untuk Menu Sebelumnya
Tekan 2 untuk kembali ke Menu
Tekan 3 untuk Keluar
Masukan Pilihan : 3

-----
|           ~           |
|           System Keluar          |
|   Terima Kasih Sudah Berkunjung |
|           ~           |
-----
```

Gambar 3. 49 Tampilan Aplikasi 17

## **BAB IV**

## **PENUTUP**

### **A. Kesimpulan**

Aplikasi perhitungan luas bangun datar dapat dibuat menggunakan bahasa Java. Tipe data yang digunakan pada aplikasi adalah int, string, dan double. Aplikasi ini menggunakan percabangan if, percabangan if ... else, percabangan if ... else if ... else, dan percabangan switch. Perulangan for juga digunakan untuk menghitung luas bangun datar sebanyak yang dimasukkan. Selain itu, pengurutan juga menggunakan percabangan for, untuk mengurutkan luas bangun datar secara *ascending*.

### **B. Saran**

Aplikasi perhitungan luas bangun datar yang telah dibuat ini masih sederhana, sehingga masih perlu pengembangan dan penyempurnaan dalam segi tampilan, kerapian, database, dan sebagainya.

## **DAFTAR PUSTAKA**

- Enterprise, J. (2019). *Lancar Java dan Javascript*. Jakarta: Elex Media Komputindo.
- Lie, S. (2018). *Struktur Dasar Pemrograman Java*. Diakses pada 15 Maret 2021 dari <https://docplayer.info/37852758-Struktur-dasar-pemrograman-java.html>
- Sallaby, A. F., Utami, F. H., & Arliando, Y. (2015). Aplikasi widget berbasis java. *Jurnal Media Infotama*, 11(2).
- Irsan, M. (2015). Rancang bangun aplikasi mobile notifikasi berbasis android untuk mendukung kinerja di instansi pemerintahan. *JustIN (Jurnal Sistem dan Teknologi Informasi)*, 3(1), 115-120.
- Frasiwi, A. R. (2019). PERANCANGAN MEDIA INFORMASI BERBASIS DIGITAL MENGGUNAKAN ANDROID DI JURUSAN TEKNIK KOMPUTER (Doctoral dissertation, Politeknik Negeri Sriwijaya).
- Sulaiman, R. (2021). USABILITY TESTING PADA WEBSITE E-LEARNING AKAKOM (ELA) STMIK AKAKOM YOGYAKARTA DENGAN HEURISTIC EVALUATION (Doctoral dissertation, STMIK AKAKOM YOGYAKARTA).
- Endra, R. Y. (2019). Belajar Mudah Algoritma Dan Pemograman Java.