

LAPORAN UJIAN SEMESTER AKHIR

PROGRAM KASIR

BAHASA JAVA



Disusun Oleh:

PUTRI ATIKA DEWI KATAREN

NIM. 202013039

DOSEN PENGAMPU

SLAMET TRIYANTO, S.ST

PROGRAM STUDI TEKNIK INFORMATIKA

POLITEKNIK KAMPAR

2020/2021

KATA PENGANTAR

Puji dan syukur atas kehadiran Allah SWT, atas segala limpahan taufiq serta hidayah-Nya yang telah memberi penulis kesempatan untuk menyelesaikan Laporan Ujian Akhir Semester, Tugas Algoritma Pemrograman ini. Adapun tujuan penulisan Laporan ini adalah untuk melengkapi Tugas Perkuliahan Algoritma Pemrograman.

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

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.

Medan, 16 Maret 2021

Penulis

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI.....	ii
BAB I.....	1
TINJAUAN PUSTAKA	1
1.1 BAHASA JAVA.....	1
1.2 Keunggulan Bahasa Java.....	2
1.3 Kekurangan Bahasa Java	3
1.4 Variabel	4
1.5 Operator	5
1.6 Penyeleksian Kondisi	9
BAB II	12
PEMBAHASAN	12
2.1 Proses Instalasi JDK	12
2.2 Hasil Program.....	17
BAB III.....	20
KESIMPULAN.....	20
SARAN.....	20
DAFTAR PUSTAKA.....	21

BAB I

TINJAUAN PUSTAKA

1.1 BAHASA JAVA

Pada tahun 1991, bahasa pemrograman Java pertama kali dikembangkan lewat sebuah proyek bernama “The Green Project” yang dibentuk oleh Sun Microsystems. Beranggotakan James Gosling, Patrick Naughton, dan Mike Sheridan, proyek ini memiliki tujuan untuk menciptakan bahasa pemrograman baru yang lebih andal dari C dan C++. Setelah proyek berjalan selama 18 bulan, James Gosling, sang *developer*, akhirnya berhasil mengembangkan bahasa pemrograman baru yang diberi nama Oak. Sayangnya, nama tersebut telah terdaftar sebagai merek dagang dari perusahaan Oak Technology. Akhirnya, sekitar tahun 1995, Sun Microsystem kemudian resmi mengubah nama Oak menjadi Java yang kabarnya terinspirasi dari *java coffee*/kopi Jawa. Itu sebabnya, logo bahasa pemrograman Java bergambar secangkir kopi.

Java dikenal memiliki moto “*Write Once, Run Anywhere*”. Hal ini menunjukkan bahwa setiap kode program yang ditulis menggunakan Java bisa langsung dijalankan pada *platform* berbeda-beda tanpa perlu menyusun ulang. Mengapa bisa demikian? Penulisan kode program Java biasanya disimpan dalam *file* berekstensi java. Dengan menggunakan *javac compiler*, kamu bisa menyusun *file* berekstensi java menjadi *file* berekstensi class. File berekstensi class inilah yang disebut sebagai *bytecode* dan bisa dijalankan di seluruh *Java Virtual Machine* (JVM). Dilihat dari penggunaannya, sebagai bahasa pemrograman umum kamu bisa

memanfaatkan Java untuk membuat berbagai bentuk aplikasi. Hal itu berlaku mulai dari aplikasi berbasis *desktop*, *website*, *mobile*, hingga aplikasi *embedded device* seperti perangkat pintar atau *microprosesor*. Sementara dalam praktiknya, bahasa Java tidak mendukung pemrograman prosedural melainkan menggunakan konsep pemrograman berbasis *object*.¹

1.2 Keunggulan Bahasa Java

Bahasa java memiliki keunggulan-keunggulan seperti berikut:

A. Mudah digunakan

Dasar dari Java adalah bahasa pemrograman C++. Meskipun bahasa pemrograman tersebut cukup kuat, tetapi tergolong konteks dan tidak cukup untuk berbagai kebutuhan Java.

Java dibangun dari dan menjadi semacam peningkatan dari bahasa pemrograman tersebut. Hal ini membuat Java menjadi bahasa pemrograman yang kuat dan sederhana untuk digunakan.

Secara khusus, jika kamu sudah familiar dengan C++, kamu kemungkinan besar tidak akan banyak kesulitan untuk mempelajari Java.

B. Berorientasi pada objek

Seperti disebutkan di atas, hal ini terkait dengan sifatnya yang merupakan bahasa pemrograman berorientasi objek. Hal ini berbeda dengan bahasa pemrograman C++ yang bisa dianggap semi berorientasi pada objek.

Java memiliki beberapa fitur dari bahasa pemrograman berorientasi objek atau *object-oriented programming language* (OOP). Hal-hal tersebut di antaranya adalah *abstraction*, *encapsulation*, *inheritance* dan *polymorphism*.

C. Keamanan

¹ <https://glints.com/id/lowongan/bahasa-pemrograman-java/#.YE1npgzZPY>

Pada awalnya, Java ditujukan untuk perangkat *mobile* yang bertukar data lewat jaringan. Hal ini membuat bahasa Java dibangun dengan tingkat keamanan tinggi.

Java sendiri bisa dikatakan adalah salah satu bahasa pemrograman dengan tingkat keamanan yang tinggi. Menurut CodeJava, sekarang ini tidak terdengar ada virus yang menyerang aplikasi Java.

D. Bisa digunakan di berbagai *platform*

Sebuah program idealnya bisa bekerja terlepas dari platform apa yang digunakan untuk mengeksekusinya. Java ditulis sebagai bahasa pemrograman yang portabel dan bisa digunakan lintas *platform*.

Hal tersebut membuatnya dapat digunakan di berbagai sistem operasi, *hardware*, ataupun perangkat. Ini sejalan dengan slogan *write once, run anywhere* yang identik dengan bahasa pemrograman ini.

1.3 Kekurangan Bahasa Java

Kekurangan dari Java adalah sebagai berikut

- A. Java memiliki kecepatan yang kurang dari C++
- B. Java adalah bahasa pemrograman yang kompleks
- C. Java menggunakan memori yang banyak
- D. Program yang dibuat oleh Java lebih lambat dibanding program yang dibuat dengan bahasa lain seperti C atau C++.
- E. Java mudah didekompilasi yang dikarenakan kode jadi Java merupakan bytecode yang menyimpan banyak atribut bahasa tingkat tinggi, seperti nama-nama kelas, metode, dan tipe data. algoritma yang digunakan program akan lebih sulit disembunyikan dan mudah dibajak

1.4 Variabel

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;
```

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

1. Variabel bisa terdiri dari huruf, angka dan karakter underscore / garis bawah (_).
2. Karakter pertama dari variabel hanya boleh berupa huruf dan underscore (_), tidak bisa berupa angka. Meskipun dibolehkan, sebaiknya tidak menggunakan karakter underscore sebagai awal dari variabel karena bisa bentrok dengan beberapa variabel settingan program.
3. Variabel harus selain dari keyword. Sebagai contoh, kita tidak bisa memakai kata `int` sebagai nama variabel, karena `int` merupakan keyword untuk menandakan tipe data integer.

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**.
4. **Tipe data string**, yakni tipe data gabungan dari char, contoh "nama_saya".

1.5 Operator

Operator dan Operand dalam Bahasa C 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 C 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 C dan berikut contoh penggunaannya :

A. Operator Aritmatika

Operator Aritmatika adalah operator yang digunakan untuk mengoperasikan bilangan integer, float dan lain sebagainya. Biasanya operatornya berupa penjumlahan, pengurangan, pembagian, perkalian, increament, decreament dan modulus. Increament merupakan operasi untuk menambahkan value sejumlah 1 sedangkan decreament 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	$a = b \% c$	sisanya dari b dibagi c
++	Increment	$a++$	$a = a + 1$
--	Decrement	$a--$	$a = a - 1$

B. 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

C. 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	Sama dengan	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

D. 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 sama dengan	a >= b	benar jika a lebih besar atau sama dengan b
<=	Lebih kecil atau sama dengan	a <= b	benar jika a lebih kecil atau sama dengan b

E. 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

1.6 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.

A. 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 */

```
Import java.util.scanner;  
Public class contoh{  
Public static void main(String [] args) {  
float nilai;  
System.out.print("Masukan nilai yang didapat : ");  
Scanner terima=new Scanner(System.in);  
Nilai=terima.nextInt();  
    if(nilai > 65){  
System.out.print ("\\n ANDA LULUS );  
}  
}  
}
```

Gambar 2. 1 Penggunaan IF

Bila program tersebut dijalankan dan kita memasukan nilai 80, maka perintah mencetak perkataan LULUS !!!! akan dilaksanakan, namun sebaliknya bila kita memasukan sebuah nilai yang kurang dari 65 maka program akan berhenti dan tidak dihasilkan apa-apa.

B. 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 :

```
Import java.util.scanner;  
Public class contoh{  
Public static void main(String [] args) {  
float nilai;  
System.out.print("Masukan nilai yang didapat : ");  
Scanner terima=new Scanner(System.in);  
Nilai=terima.nextInt();  
if(nilai > 65){
```

```
System.out.print ("\n ANDA LULUS ");
}
Else {
System.out.print ("\n ANDA TIDAK LULUS ");
}

}
}
```

C. 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 :

```
switch(kondisi) {

    case 1      : pernyataan-1;
                  break;

    case 2      : pernyataan-2;
                  break;

    case n      : pernyataan-n;
                  break;

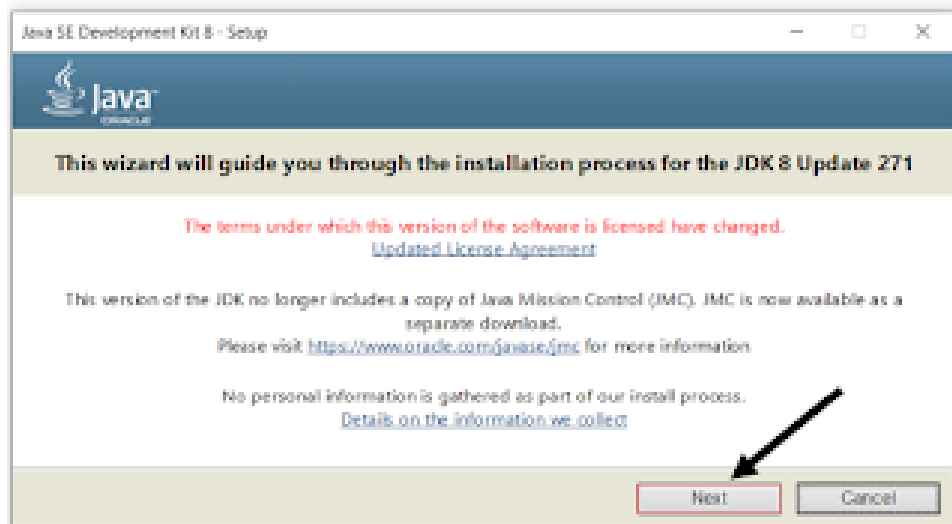
    default }   : pernyataan-m
```

BAB II

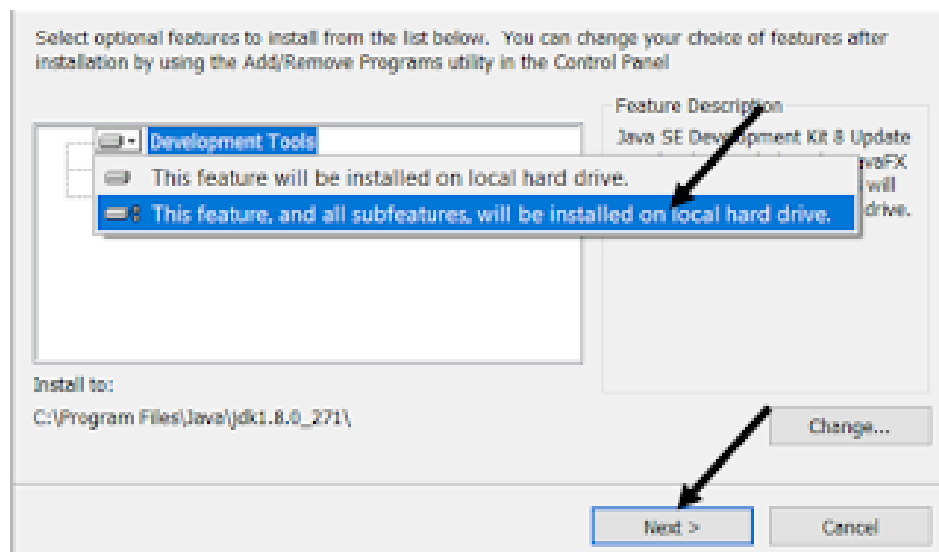
PEMBAHASAN

2.1 Proses Instalasi JDK

1. Klik 2 kali pada software JDK untuk menjalankan setup nya.
2. Setelah itu, klik next

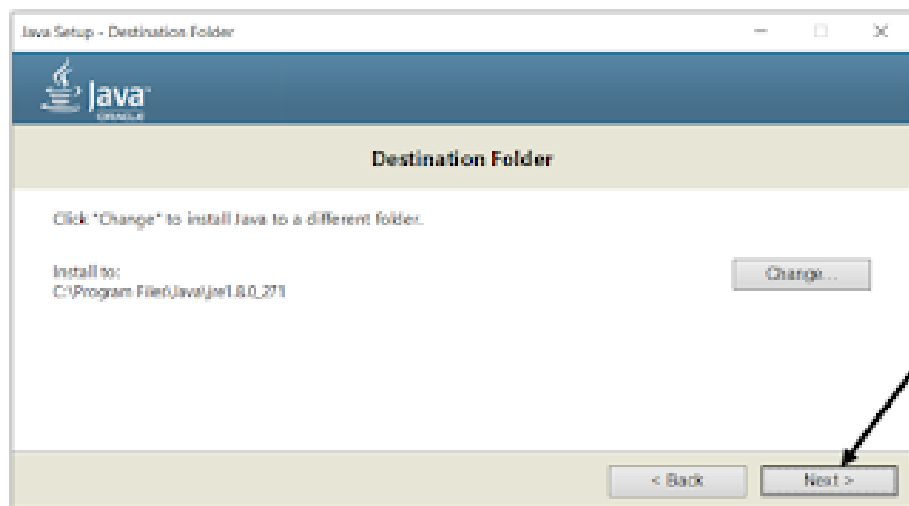


3. kemudian tunggu prosesnya hingga selesai. setelah tampilan seperti dibawah ini, klik panah tiga kebawah yang ada pada development tools.



kemudian pilih this feature and all subfeatures , will be installed on local harddrive . Setelah itu klik next

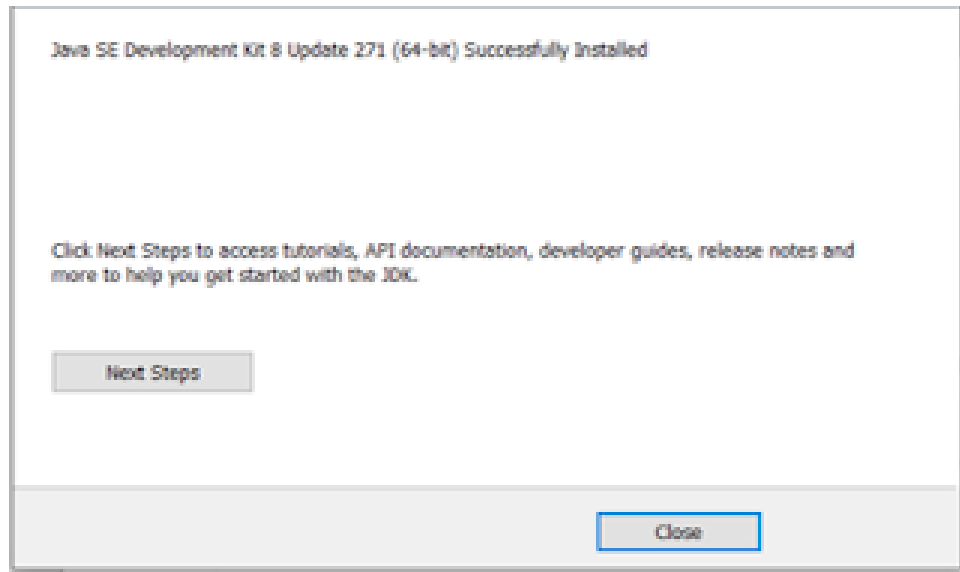
4. Pilih folder direktori instalasi, saya menyarankan teman-teman mengikuti folder instalasi defaultnya saja



5. Tunggu hingga proses instalasi selesai

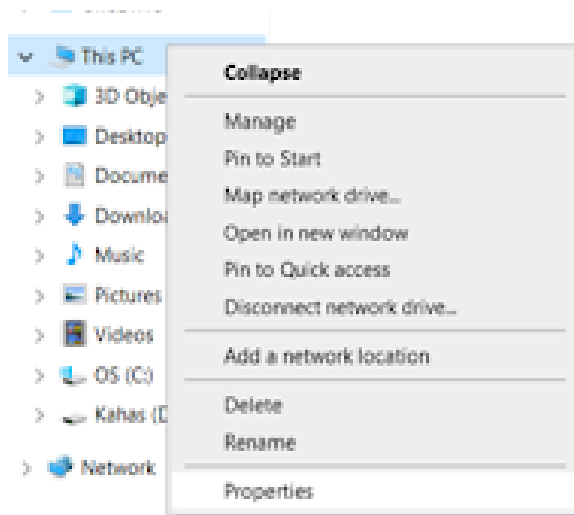


6. setelah itu klik close, dan proses installasi pun selsai.

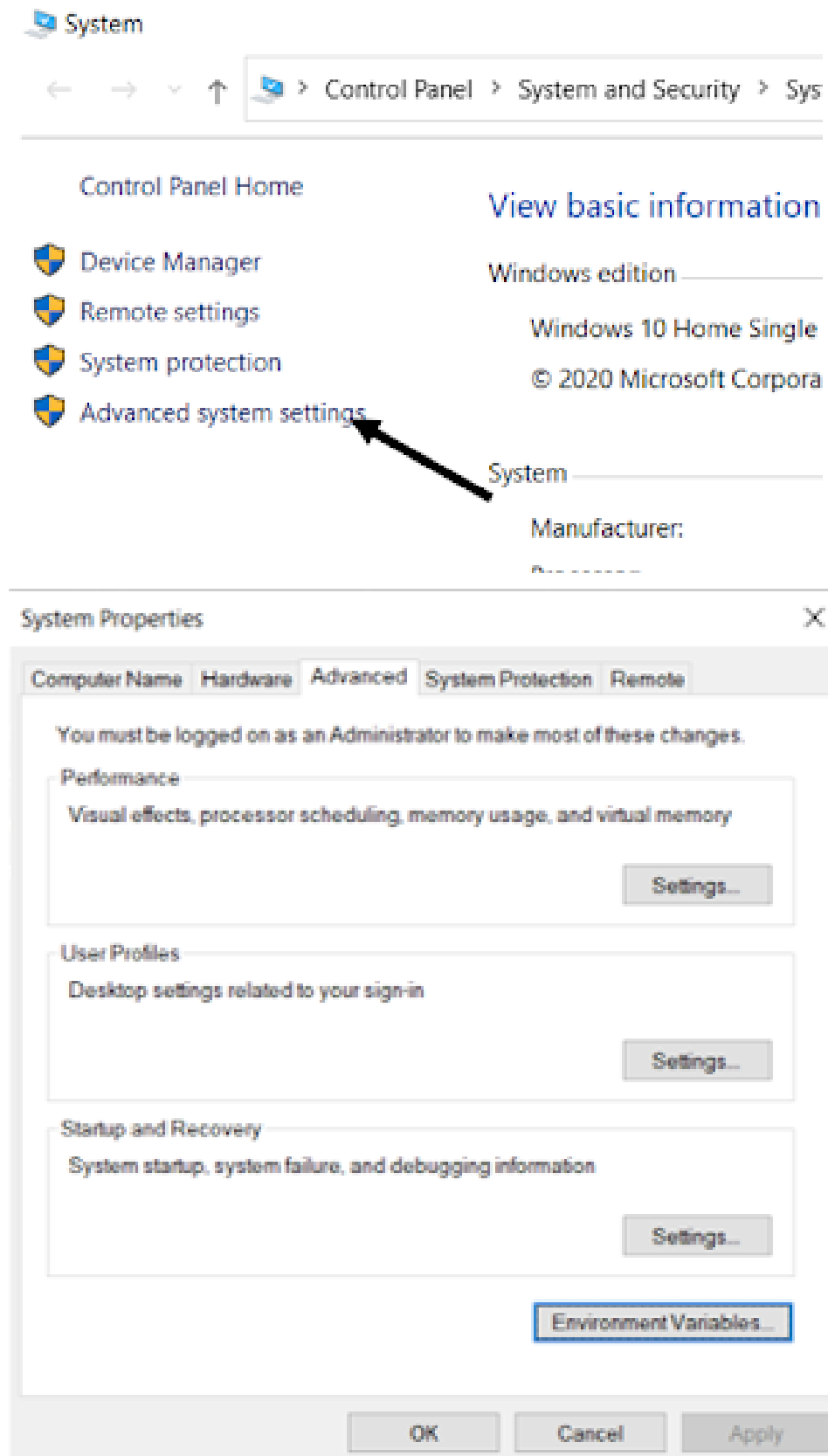


Setelah proses installasi selesai, kita harus mengkonfigurasi agar JDK dapat digunakan. Langkah-langkah konfigurasi JDK 8.0 :

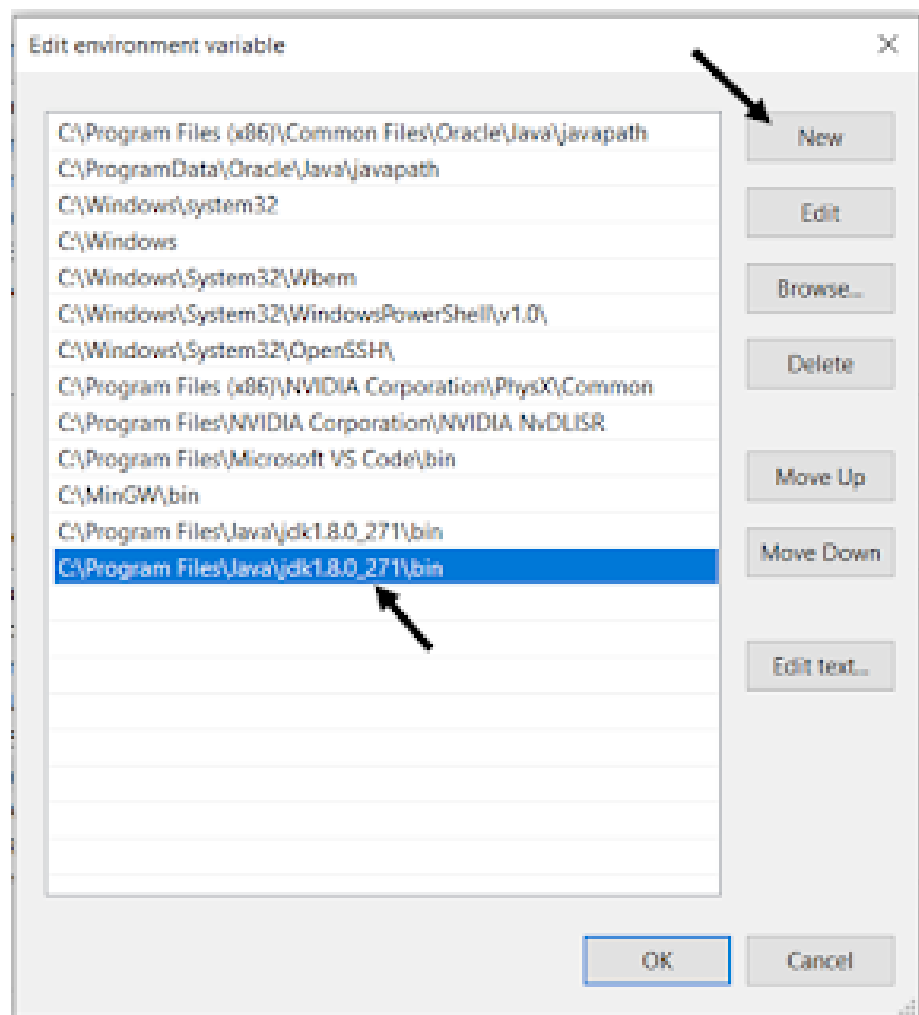
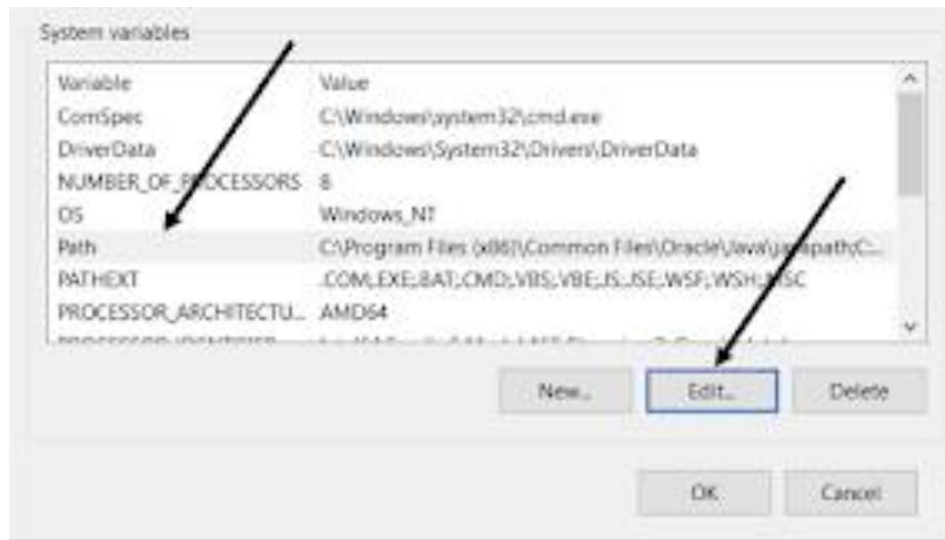
1. Buka windows explorer dengan shortcut *Windows + R*, klik kanan pada *my computer*, kemudian temen-temen pilih *properties*.



2. Setelah jendela baru terbuka, temen-temen pilih advanced system settings, kemudian pilih envirotnmen variables.

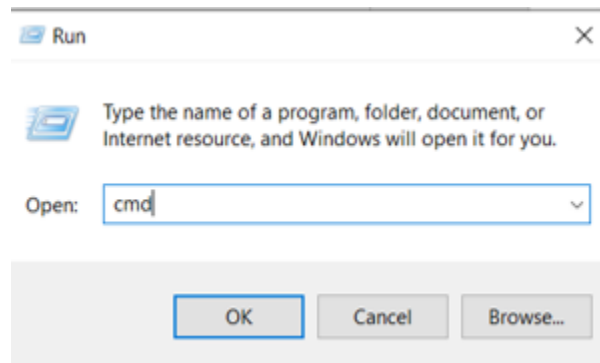


3. setelah itu, cari variable path, kemudian edit setelah itu copy kan path tempat JDK tadi di install dalam hal ini saya menyimpannya di C:\Program Files\Java\jdk1.8.0_271\bin



4. Setelah itu klik ok sampai semua jendela tertutup

5. kemudian teman-teman bisa menguji apakah JDK teman-teman sudah terinstall. dengan cara klik windows + R, setelah itu ketikkan CMD



6. setelah itu ketikkan `javac -version` pada terminal cmd, Proses konfigurasi selesai

2.2 Hasil Program

Hasil output dari program berupa aplikasi berbasis Command Line Interface (CLI) yaitu tatap muka baris perintah dimana untuk menjalankan program ini kita pakai Command Prompt aplikasi bawaan dari Sistem Operasi Windows. Aplikasi yang sudah dibuat akan berjalan seperti berikut:

1. Tampilan awal kita akan diminta untuk login ke Aplikasi

```
D:\Projek>java putri
=====
                Selamat Datang di
                Aplikassi Sederhana
=====
Masukkan Nama Pengguna :
putri
Masukkan Sandi
202013035
```

2. Menu awal yang dapat dipilih setelah login

```
=====
                        Selamat Datang di
                        POLKAM MART
                        Kasir
=====
Silahkan pilih menu :
1. Menu Penjualan
2. Menu Stok
3. Menu Tambah Data
4. Menu Utama
5. Menu Keluar
=====
```

3. Ini adalah tampilan dari menu penjualan

```
=====
                        POLKAM MART
                        Kasir
=====
Nama barang                harga
Popmie                     Rp.5000
Es Lilin                   Rp.1000
Kerupuk                    Rp.500
Masukkan jumlah barang pertama:
```

5. Ini adalah tampilan dari menu stok

```
=====
                        POLKAM MART
                        Kasir
=====
Nama barang                jumlah
Popmie                     10
Es Lilin                   15
Kerupuk                    20
=====
```

6. Dan ini adalah tampilan dari menu Tambah Barang

```
=====
                        POLKAM MART
                        Kasir
=====
Masukkan banyaknya data yang ingin di inputkan : 2
Masukkan Nama barang ke 1 : Mie Gelas
Masukkan Harga barang ke 1 : 2000
Masukkan jumlah barang ke 1 : 10
Penginputan barang ke 1telah selesai
=====
Masukkan Nama barang ke 2 : Beng-Beng
Masukkan Harga barang ke 2 : 3000
Masukkan jumlah barang ke 2 : 5
Penginputan barang ke 2telah selesai
=====

Barang yang diinputkan adalah sebagai berikut
Nama barang ke 1 Mie Gelas  Harga barang ke 1 2000  Jumlah barang ke 1 10
Nama barang ke 2 Beng-Beng  Harga barang ke 2 3000  Jumlah barang ke 2 5
Total harga adalah Rp.35000
=====
```

7. Apabila kita pilih 5 pada menu kasir tadi, maka akan keluar dari Program

```
=====
                        Selamat Datang di
                        POLKAM MART
                        Kasir
=====
Silahkan pilih menu :
1. Menu Penjualan
2. Menu Stok
3. Menu Tambah Data
4. Menu Utama
5. Menu Keluar
=====
5

D:\Projek>
```

BAB III

KESIMPULAN DAN SARAN

3.1 KESIMPULAN

Program Aplikasi kasir sudah berhasil dibuat, pada program ini menggunakan hello world, tipe data dan variable, perulangan, percabangan if untuk menyeleksi suatu kondisi tunggal, else if bersarang, dan switch case bersarang, dan sorting bubble sort. Program ini juga menggunakan beberapa perintah seperti printf dan scanner. Untuk memperindah

Program ini juga menggunakan **fungsi void** untuk mengarahkan eksekusi program ke pernyataan yang diawali oleh sebuah label, seperti static void kasir berarti program akan mengarahkannya ke kasir, static void Pengurutan berarti program akan mengarahkannya ke Pengurutan.

3.2 SARAN

Program ini sudah layak digunakan untuk dijadikan referensi program aplikasi kasir tetapi 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

Tandika, Bima. 2020. *Apa Itu Bahasa Pemrograman Java? Ini Penjelasannya*.

<https://glints.com/id/lowongan/bahasa-pemrograman-java/#.YE1npdgzZPY>. Diakses tanggal 14 Maret 2020.