

**PEMBUATAN PROGRAM MESIN KASIR**  
**MENGGUNAKAN BAHASA JAVA**  
**UJIAN AKHIR SEMESTER 1**



**Disusun Oleh :**  
**IWAN(202013033)**

**DOSEN PENGAMPU :**  
**SLAMET TRIYANTO,S.ST.**

**PROGRAM STUDI TEKNIK INFORMATIKA**  
**POLITEKNIK KAMPAR**

**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 Pratikum Tugas Praktik Algoritma. Adapun tujuan penulisan Laporan ini adalah untuk melengkapi Tugas Perkuliahan 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 mata kuliah 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.

RantauPrapat, 15 Maret 2021

Penulis

## DAFTAR ISI

<b>KATA PENGANTAR .....</b>	<b>i</b>
<b>DAFTAR ISI .....</b>	<b>ii</b>
<b>BAB I.....</b>	<b>1</b>
<b>TINJAUAN PUSTAKA .....</b>	<b>1</b>
A. Pengertian .....	1
1. Pengertian Bahasa Java.....	1
2. Kelebihan dan Kekurangan bahasa Java .....	1
B. Tujuan Praktikum.....	3
C. Alat dan Bahan.....	3
<b>BAB II .....</b>	<b>5</b>
<b>PEMBAHASAN.....</b>	<b>5</b>
A. Struktur Bahasa Java .....	5
B. Tipe Data dan Variabel .....	6
C. Percabangan .....	7
D. Percabangan IF ELSE.....	7
E. Pemilihan Switch Case.....	9
F. Perulangan.....	10
G. Bubble Sort(pengurutan) .....	13
<b>BAB III.....</b>	<b>16</b>
<b>LANGKAH KERJA .....</b>	<b>16</b>
A. Instalasi Java Jdk.....	16
1. Download java Jdk.....	16
2. Mengekstrak Java Jdk .....	16
3. Instal Java Jdk .....	16
<b>Bab Iv .....</b>	<b>19</b>
<b>Penutup .....</b>	<b>19</b>

A. Kesimpulan .....	19
B. Saran .....	19
<b>Daftar Pustaka .....</b>	<b>20</b>

# **BAB I**

## **TINJAUAN PUSTAKA**

### **A. Pengertian**

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

Java ini adalah salah satu bahasa pemrograman yang dianggap powerful karena merupakan bahasa pemrograman yang bersifat multiplatform. Selain itu, library yang dimilikinya juga terbilang lengkap. Karena itulah, Java ini menjadi bahasa pemrograman yang banyak digunakan oleh para programmer.

Pengertian Java adalah sebuah bahasa pemrograman umum tingkat tinggi yang dipakai untuk membuat beragam aplikasi. Bahasa pemrograman pada Java berorientasi pada object. Sementara programnya tersusun dari kelas-kelas atau bagian-bagian tertentu. Kelas ini sendiri terdiri atas metode-metode yang bertugas melakukan pekerjaan kemudian mengembalikan informasi setelah pekerjaan tersebut selesai.

Java ini adalah bahasa pemrograman yang non-spesifik atau bersifat umum. Artinya, bahasa pemrograman Java yang berorientasi pada object (OOP) bisa dijalankan untuk berbagai platform atau sistem operasi. Selain itu, OOP juga memungkinkan program komputer sebagai kelompok-kelompok objek yang bisa saling berinteraksi. Sementara itu, Object Oriented Programming (OOP) ini berperan untuk mengorganisir program sebagai sebuah kumpulan komponen yang kemudian disebut sebagai objek. Object-object tersebut sifatnya independen, dapat berkomunikasi dengan object lain dengan aturan-aturan tertentu.

#### **2. Kelebihan dan Kekurangan bahasa Java**

##### **a. Kelebihan bahasa Java :**

##### **1. 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.

2. 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.

3. 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.

4. 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.

5. 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.

**b. Kekurangan bahasa java :**

1. Penggunaan memory yang cukup tinggi

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.

## 2. 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.

### **B. Tujuan Praktikum**

- a) Untuk mengetahui cara menggunakan aplikasi bahasa Java dalam pembuatan program kasir.
- b) Untuk memahami dasar – dasar bahasa Java untuk membuat program kasir.
- c) Untuk memahami source code yang digunakan untuk membuat program kasir.

### **C. Alat dan Bahan**

1. Alat
  - a) Laptop
2. Bahan
  - a) Sublime Text
  - b) CMD
  - c) Aplikasi Bahasa Java

## **BAB II**

### **PEMBAHASAN**

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

Bentuk program Java mirip dengan kebanyakan program bahasa tingkat tinggi lainnya. Bentuk programnya ada Deklarasi package, Impor Library, class dan method main

##### **1. Deklarasi Package**

Package merupakan sebuah folder yang berisi sekumpulan program Java. Deklarasi package biasanya dilakukan saat membuat program atau aplikasi besar. Contoh deklarasi package

```
package com.imastudio.program;
```

Biasanya nama package mengikuti nama domain dari sebuah vendor yang mengeluarkan program tersebut. Pada contoh di atas, com.imastudio adalah nama domain dari imastudio. Aturannya: nama domain dibalik, lalu diikuti nama programnya. Bagaimana kalau kita tidak mendeklarasikan package? Boleh-boleh saja dan programnya akan tetap bisa jalan. Tapi nanti saat produksi, misalnya saat membuat aplikasi Android, kita wajib mendeklarasikan package.

##### **2. Bagian Impor**

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*.

##### **3. Bagian Class**

Java merupakan bahasa pemrograman yang menggunakan paradigma OOP (Object Oriented Programming). Setiap program harus dibungkus di dalam class agar nanti bisa dibuat menjadi object. Kalau kamu belum paham apa itu OOP? Cukup pahami class sebagai deklarasi nama program.

```
class Program {  
  
    public static void main(String args[]){  
  
        system.out.println("hello World");  
    }  
}
```



Ini adalah blok class. Blok class dibuka dengan tanda kurung kurawal { kemudian ditutup atau diakhiri dengan }. Di dalam blok class, kita dapat mengisinya dengan method atau fungsi-fungsi dan juga variabel. Pada contoh di atas, terdapat method main().

#### 4. Method Main

Method main() atau fungsi main() merupakan blok program yang akan dieksekusi pertama kali. Ini adalah entri point dari program. Method main() wajib kita buat. Kalau tidak, maka programnya tidak akan bisa dieksekusi. Contoh method main().

```
Public class program{  
    public static void main(String args[]){  
        System.out.println("Hello World"); }  
}
```

Penulisannya harus seperti ini... Method `main()` memiliki parameter `args[]`. Parameter ini nanti akan menyimpan sebuah nilai dari argumen di command line. Lalu di dalam method main(), terdapat statement atau fungsi: `System.out.println("Hello World");` Ini adalah fungsi untuk menampilkan output teks ke layar monitor

#### B. Tipe Data dan Variabel

Variabel sebagai tempat menyimpan nilai sementara. Lalu, apa itu tipe data? Tipe data adalah jenis data yang tersimpan dalam variabel.

Macam-macam Tipe Data Berikut ini macam-macam tipe data pada Java:

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

### C. Percabangan

Percabangan IF merupakan bentuk percabangan yang paling sederhana. Dimana hanya terdapat sebuah kondisi pada pemilihan IF. Bentuk umumnya.

```
if (kondisi){  
    pernyataan1; pernyataan2;  
}
```

Pernyataan 1 dan pernyataan 2 akan dijalankan/di eksekusi ketika kondisi bernilai benar (true). Sementara ketika kondisi bernilai salah (false) kedua pernyataan tersebut tidak dijalankan oleh program.

```
public class Percabangan {  
    public static void main(String[] args) {  
        int bil=0;  
        if (bil==0){  
            System.out.println("Bilangan Nol");  
        }  
    }  
}
```

Pada program diatas kita membuat sebuah class dengan nama Percabangan. Class adalah suatu ‘blue print’ untuk menciptakan instance dari suatu object. Program dibuat pada fungsi utama main(), dimana kita inisialisasi variabel bil menggunakan tipe data integer dengan nilai 0. Kemudian terdapat percabangan IF dengan kondisi bil==0 ? jika kondisi tersebut benar (true) maka pernyataan untuk mencetak output ‘Bilangan Nol’ akan di eksekusi. Sementara jika kondisi salah (false) program tidak mengerjakan perintah apapun.

### D. Percabangan IF ELSE

Pada bentuk percabangan IF yang kedua terdapat bagian ELSE dimana jika kondisi tidak terpenuhi maka pernyataan pada bagian ELSE yang akan dikerjakan. Perhatikan bentuk umumnya:

```
if (kondisi){  
    pernyataan1;  
}
```

```

    pernyataan2; }
else
{ alternatif_pernyataan1;
  alternatif_pernyataan2; }

```

Alternatif\_pernyataan1 dan alternatif\_pernyataan2 pada bagian ELSE akan dijalankan ketika kondisi bernilai salah (false).

```

public class Percabangan {
public static void main(String[] args){
int bil=1;
if (bil==0){
System.out.println("Bilangan Nol");
}else { System.out.println("Bilangan Bukan Nol");
} } }

```

Program diatas terdapat variabel bil dengan nilai 1. Kemudian ada percabangan IF dengan kondisi apakah nilai bil adalah 0 ? Jika benar maka tampilkan kalimat ‘Bilangan Nol’ namun jika salah tampilkan ‘Bilangan bukan nol’. Dari contoh ini dapat kita lihat dengan sederhana bahwa nilai bil adalah 1 dan bukan 0 sehingga kondisi bernilai salah (false) maka pernyataan yang di eksekusi adalah pada bagian ELSE yaitu menampilkan kalimat **‘Bilangan bukan nol’**.

Percabangan IF ELSE IF Bentuk ke-3 dari percabangan IF adalah IF ELSE IF dimana jika pada bentuk pertama dan ke-2 hanya memiliki satu kondisi pada bentuk yang ketiga percabangan memiliki lebih dari satu kondisi. Bentuk Umumnya:

```

if (kondisi1){
    pernyataan1;
    pernyataan2;
}else if (kondisi2){
    pernyataan3;
    pernyataan4;
}else if (kondisi3)

```

```
{ pernyataan5;  
pernyataan6; }  
else{ alternatif_pernyataan;  
}
```

### **E. Pemilihan Switch Case**

Struktur pemilihan switch case sebetulnya hampir sama dengan percabangan IF ELSE dimana jika di percabangan IF terdapat beberapa kondisi, pada pemilihan switch akan ada beberapa case yang dapat kita buat, jika nilai yang menjadi nilai pembanding sama dengan case yang ada atau kata lainnya jika suatu case pada pemilihan switch bernilai true maka pernyataan pada case tersebut yang akan dieksekusi. Bentuk umum:

```
switch (ekspresi){  
case nilai:  
pernyataan1;  
break;  
case nilai:  
pernyataan2;  
break;  
case nilai:  
pernyataan3;  
break;  
default;  
pernyataan_default; }
```

Pada percabangan switch harus ada nilai ekspresi yang nantinya akan dibandingkan dengan setiap case nilai ekspresi berupa bilangan bulat atau karakter. Jika terdapat case yang sama dengan nilai ekspresi (bernilai true) maka pernyataan pada case tersebut akan dieksekusi. Diakhir sebuah case selalu ada kata kunci break sebagai tanda bahwa berakhirnya eksekusi pernyataan pada case tersebut.

### Contoh Program

```
package percabangan;
import java.util.Scanner;
public class Percabangan {
    public static void main(String args[]){
        Scanner masuk=new Scanner(System.in);
        int pil;
        System.out.print("Masukkan Pilihan Jurusan : ");
        pil=masuk.nextInt();
        switch(pil) {
            case 1: System.out.println("Manajemen Informatika");
                break;
            case 2: System.out.println("Teknik Komputer");
                break;
            case 3: System.out.println("Komputerisasi Akuntansi");
                break;
            case 4: System.out.println("Teknik Informatika");
                break;
            case 5: System.out.println("Sistem Informasi");
                break;
            default: System.out.println("Pilihan Salah!!!"); } } }
```

### F. Perulangan

Dalam Java, ada tiga struktur kontrol perulangan yaitu: for, while, dan do-while.

Untuk yang belum tahu: Perulangan ( atau yang disebut Looping) adalah suatu proses yang dilakukan secara berulang-ulang hingga mencapai kondisi tertentu. Sebagai contoh ketika anda ingin mencetak deretan angka hingga batas tertentu (contoh: 1-100), maka anda bisa menggunakan fungsi looping dalam program.

Biasanya fungsi looping digunakan dan berperan penting dalam algoritma sorting, karena kita akan menukar nilai variabel hingga menghasilkan nilai berurutan. Bentuk umum perulangan pada java dapat kita temukan dalam beberapa program yang bekerja menggunakan tipe data array sebagai variabel yang digunakan.

#### a) Strukur dan Aturan Penulisan For Loop

Pada dasarnya konsep perulangan pada java memiliki sintaks yang sama dengan C Perhatikan baris kode berikut ini:

```
for (start; condition; increment)
{
    statement;
}
```

- 1) **Start** adalah kondisi pada saat awal perulangan. Biasanya kondisi awal ini digunakan untuk membuat dan memberikan nilai kepada variabel yang digunakan untuk mengontrol perulangan. Misalkan, kita akan membuat variabel *i*, maka untuk kondisi start ini, kita juga harus memberikan nilai awal untuk variabel *i*, misalnya dengan 1, maka *i*=1.
- 2) **Condition** adalah kondisi yang harus terpenuhi agar looping dapat berjalan. Selama kondisi terpenuhi, maka Java akan terus melakukan perulangan. Jadi anda harus lebih teliti dan berhati-hati dalam memberikan condition dalam perulangan for supaya tidak terjadi infinity loop atau malah loop tidak bekerja.
- 3) **Statement** adalah bagian baris kode program yang akan diproses secara terus-menerus selama proses perulangan berlangsung.
- 4) **Increment** adalah bagian yang digunakan untuk memproses variabel agar bisa memenuhi kondisi akhir perulangan. Biasa ditulis *i++* atau *++i*.

Berikut contoh sederhana perulangan for pada java:

```
public class CetakAngka {
    public static void main(String[] args){
        for(int x=0;x<10;x++)
        {
```

```
System.out.println(x);  
}  
}  
}
```

Dalam program diatas kita hanya ingin mencetak 10 angka mulai dari 0 sampai 9. Mengapa dimulai dari nol? karena start kita di angka nol.\

#### b) Perbedaan While dan Do While

Perulangan yang mungkin anda akan temukan adalah while dan do while. Kedua looping ini akan bekerja selama kondisi masih terpenuhi. Namun yang membedakan kedua perulangan tersebut adalah lokasi kondisi. Untuk detailnya mari kita langsung saja lihat kedalam contoh program.

```
public class WhileLoop {  
    public static void main(String[] args) {  
        int batas = 0; while (batas<10) {  
            System.out.print(batas); batas++;  
        }  
    }  
}
```

dan bandingkan dengan baris kode berikut:

```
public class DoWhileLoop {  
    public static void main(String[] args) {  
        int batas = 0;  
        do  
        {  
            System.out.print(batas);  
            batas++;  
        }while (batas<10); } }
```

Sekilas memang hasilnya sama saja, namun sebenarnya ada perbedaan hasil jika kita menggunakan while loop dengan menggunakan do while. Pada kasus do while loop pengecekan dilakukan di akhir sehingga minimal

program akan berjalan 1 kali meskipun kondisi tidak terpenuhi. Sebenarnya konsepnya sama dengan perulangan for, namun hanya sedikit berbeda dalam penulisan kodenya. Kita tetap membutuhkan start, condition, dan increment.

#### **G. Bubble Sort(pengurutan)**

Bubble sort adalah salah satu algoritma pengurutan data dengan cara membandingkan sebuah data dengan sebelahnya, kemudian dilakukan pertukaran data jika berada dalam urutan yang salah. Jika tidak ada pertukaran data lagi, berarti data-data tersebut sudah terurut. Dengan menggunakan algoritma Bubble sort, tentunya kita dapat mengurutkan sebuah bilangan dari terkecil sampai terbesar (Ascending) maupun bilangan dari terbesar sampai dengan terkecil (Descending).

Berikut ini adalah sistem kerja dari algoritma Bubble Sort dari sebuah n elemen data yang belum terurut (unsorted data):

- a) Bandingkan data ke-i dengan data ke-(i+1) . Jika tidak sesuai lakukan pertukaran data untuk menempati urutan yang benar dimana : data ke-i = data ke-(i+1) dan data ke-(i+1) = data ke-i. Urutannya bergantung apakah data tersebut mau diurutkan secara ascending (A-Z) maupun secara descending (Z-A).
- b) Lalu bandingkan data berikutnya : data ke-(i+1) dengan data ke-(i+2). Jika tidak sesuai urutannya, tukar lagi data tersebut. Proses perbandingan data dilanjutkan sampai data terakhir.
- c) Lakukan sampai beberapa kali iterasi, sampai didapat kondisi tidak terjadinya pertukaran data dalam satu iterasi. Artinya kita sudah mendapatkan data yang terurut.

#### **Program Java Bubble Sort**

Setelah kita memahami prinsip sorting data dengan algoritma Bubble Sort, sekarang kita akan mengimplementasikan Bubble Sort dalam program Java. Dalam program ini, kita menerima jumlah inputan bilangan yang akan dimasukkan oleh user ketika program dijalankan. Kemudian kita



memasukkan bilangan satu persatu, setelah itu akan diurutkan dari yang terkecil sampai dengan terbesar (Ascending).

```
import java.util.Scanner;

public class BublbleSort
{
    public static void main(String[] args)
    {
        int jumlah,i,j, swap;
        Scanner scan = new Scanner(System.in);

        System.out.print("Masukkan Jumlah Bilangan Yang Mau DiInputkan :");
        jumlah = scan.nextInt();

        int array[] = new int[jumlah];

        System.out.println("\nMasukkan " + jumlah+" Buah Bilangan Integer");
        System.out.println("=====");
        for(i=0; i<jumlah;i++)
        {
            System.out.print("Bilangan Ke- " + (i+1)+" =");
            array[i]=scan.nextInt();
        }

        System.out.println("\nBilangan Belum Terurutnya");
        for(int a=0;a<jumlah;a++)
        {
            System.out.print(array[a] + " ");
        }

        for(i=0; i<(jumlah-1);i++)
        {
            for(j=0;j<jumlah-i-1;j++)
            {
                if (array[j] > array[j+1])
                {
                    swap = array[j];
                    array[j] = array[j+1];
                    array[j+1]=swap;
                }
            }
        }

        System.out.println("\n\nBilangan Terurutnya adalah :");
        for(i=0;i<jumlah; i++)
            System.out.print(array[i] + " ");
    }
}
```

Outputnya di tunjukkan oleh gambar di bawah ini

```
C:\Program Files\Xinox Software\JCreatorV3\GE2001.exe
Masukkan Jumlah Bilangan Yang Mau DiInputkan :5
Masukkan 5 Buah Bilangan Integer
=====
Bilangan Ke- 1 =5
Bilangan Ke- 2 =1
Bilangan Ke- 3 =12
Bilangan Ke- 4 =-5
Bilangan Ke- 5 =16
Bilangan Belum Terurutnya
5 1 12 -5 16
Bilangan Terurutnya adalah :
-5 1 5 12 16 Press any key to continue...
```

## BAB III

### LANGKAH KERJA

#### A. Instalasi Java Jdk

Java merupakan bahasa pemrograman yang bersifat umum/non-spesifik (general purpose), dan secara khusus di disain untuk memanfaatkan dependensi implementasi seminimal mungkin. Sehingga komputer mengerti apa yang dimaksud dengan bahasa java. Java jdk adalah salah satu compiler bahasa Java

1. Download java Jdk

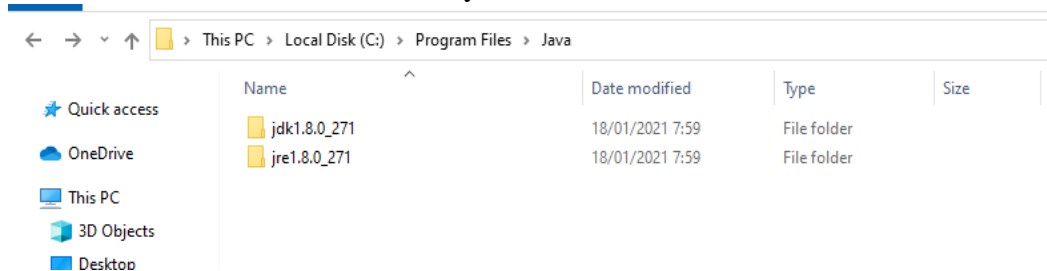
- a. Nyalakan komputer atau laptop, setelah itu beri koneksi internet.
- b. Buka browser dan carilah compiler java Jdk
- c. Setelah itu download

2. Mengekstrak Java Jdk

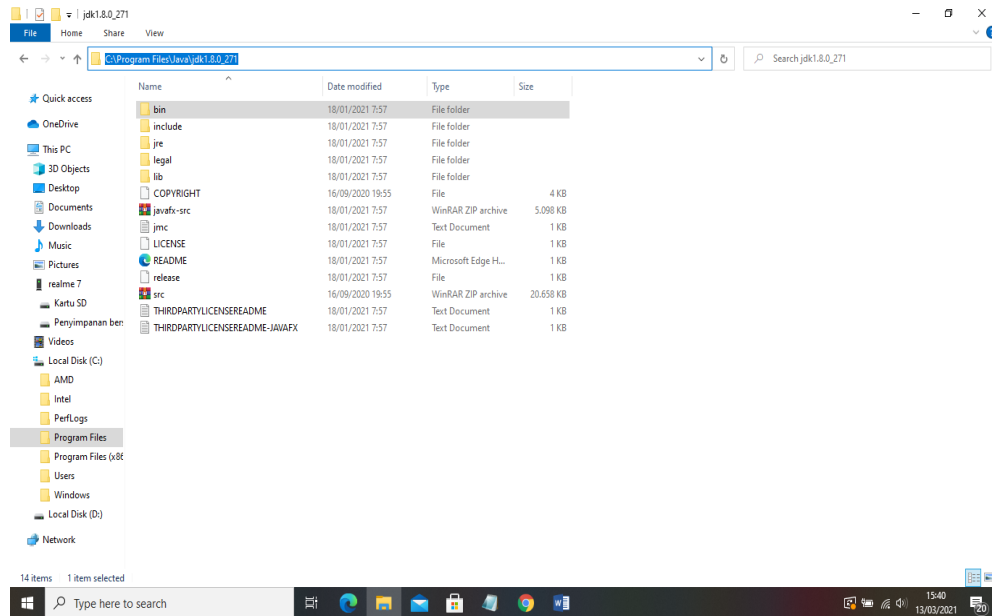
- a. Setelah Java jdk berhasil terdownload lalu buka
- b. Kemudian klik ekstrak to dan pilihlah lokasi penyimpanan kemudian klik “ok”. Setelah itu akan muncul file baru jdk yang berisi folder bin, include, lib, libexec.

3. Instal Java Jdk

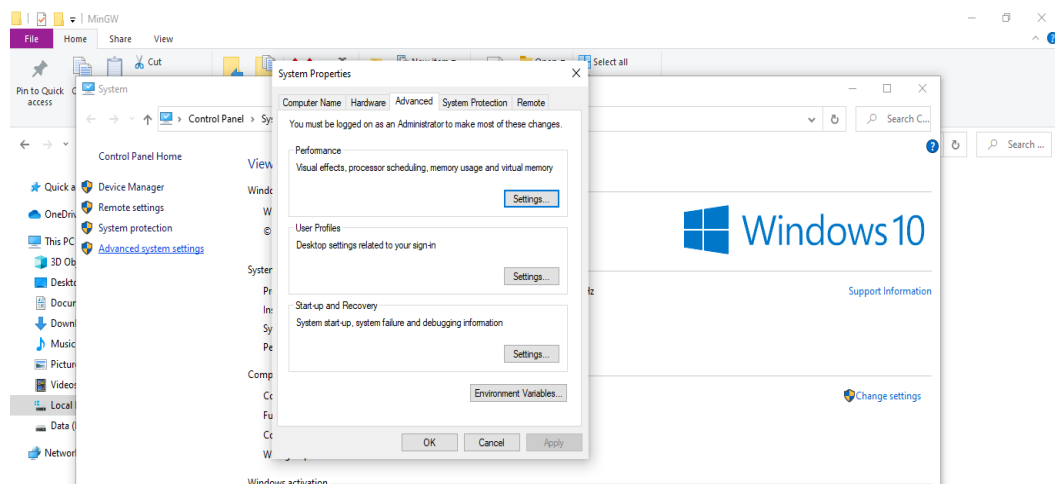
- a. Langkah awal penginstalan Java Jdk adalah masuk ke file yang sudah di ekstrak contohnya di local disk C.



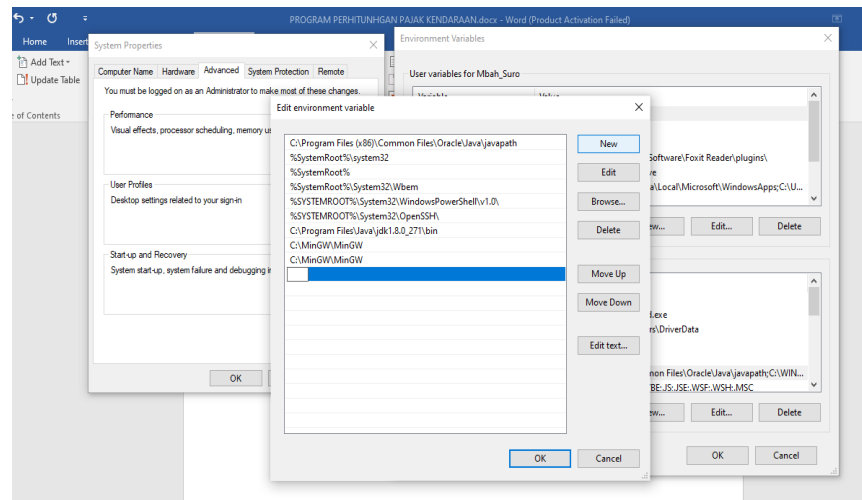
- b. Setelah masuk pilih folder Java Jdk kemudian cari folder “bin” kemudian klik sekali dan copy link nya, seperti gambar dibawah ini.



- c. Kemudian masuk ke this Pc kemudian klik kanan dan pilih properti
- d. Lalu pilih Advanced System Setting, kemudian pada sistem properti pilih “environment variabels”



- e. Kemudian lihat di sistem variabel cari “path” kemudian klik new dan pastekan link tadi yang sudah di copy kemudian ok



- 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 dibawah versi Java adalah 1.8.0\_271.

```

C:\Windows\system32\cmd.exe
C:\Users>java -version
java version "1.8.0_271"
Java(TM) SE Runtime Environment (build 1.8.0_271-b09)
Java HotSpot(TM) 64-Bit Server VM (build 25.271-b09, mixed mode)
C:\Users>

```

## **Bab Iv**

### **Penutup**

#### **A. Kesimpulan**

Program Aplikasi kasir sudah berhasil dibuat oleh programers, pada program ini programers menggunakan hello world, tipe data dan variable, perulangan, percabangan if untuk menyeleksi suatu kondisi tunggal, else if bersarang, switch case bersarang, dan sorting bubble sort. Program ini juga menggunakan tipe data integer dan string. Program ini dapat memudahkan dalam pembuatan program kasir yang didalamnya menggunakan pemilihan dan perulangan.

#### **B. Saran**

Saran penulis untuk pengembangan program aplikasi kasir selanjutnya, 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**

<https://www.duniailkom.com/tutorial-belajar-java-pengertian-bahasa-pemrograman-java/#:~:text=Terjemahan%20bebasnya%2C%20Java%20adalah%20%E2%80%9Cbahasa,tidak%20bergantung%20pada%20sistem%20tertentu%E2%80%9C.>

<http://dauren-seo.blogspot.com/2018/02/logika-bubble-sort-di-java-dan-contoh.html>

<https://dosenit.com/kuliah-it/pemrograman/kelebihan-dan-kekurangan-java>