

LAPORAN PRAKTIKUM STRUKTUR DATA

PERTEMUAN KE-7



Disusun Oleh :

NAMA : Raden Isnawan Argi Aryasatya

NIM : 195410257

JURUSAN : Teknik Informatika

JENJANG : S1

Laboratorium Terpadu

Sekolah Tinggi Management Informatika Komputer

AKAKOM

YOGYAKARTA

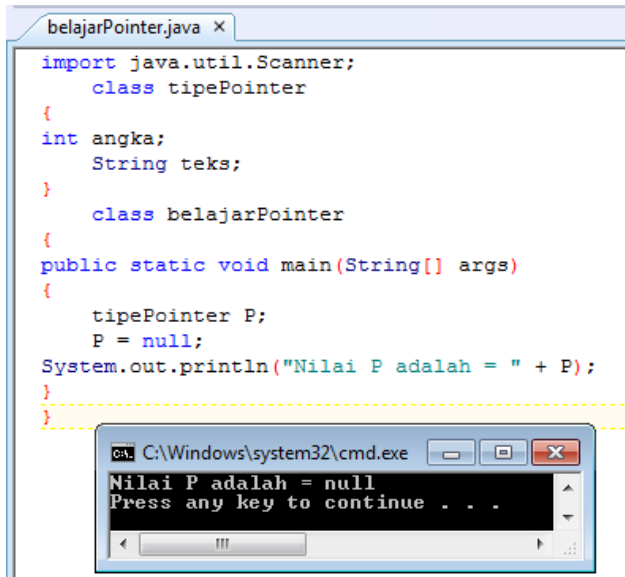
2020

PERTEMUAN KE-7 (POINTER DALAM JAVA)

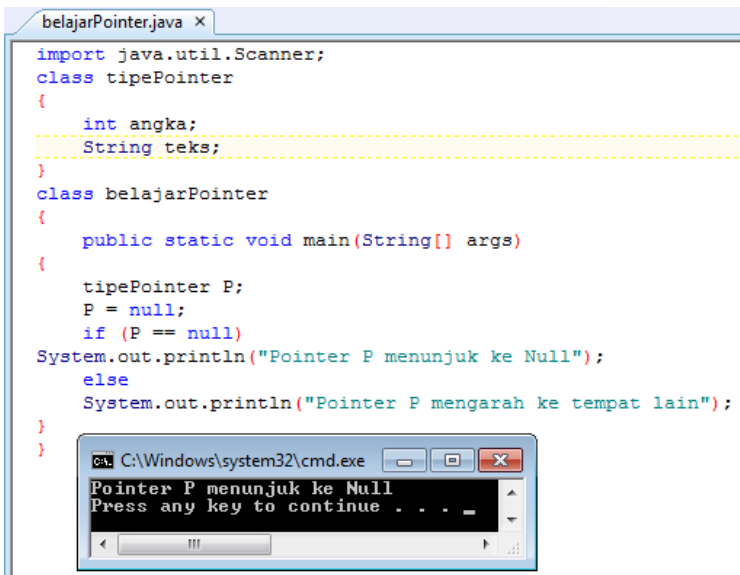
TUJUAN

Mahasiswa dapat mengeksploitasi variabel pointer di dalam java

PRAKTIK 1



penjelasan: yang terjadi adalah P sama dengan null. Nilai pointer P dapat dilihat yaitu null. Null adalah sebuah tempat kosong di memori yang tidak memiliki nilai. Null dapat diumpamakan seperti ground pada sebuah rangkaian elektronika.

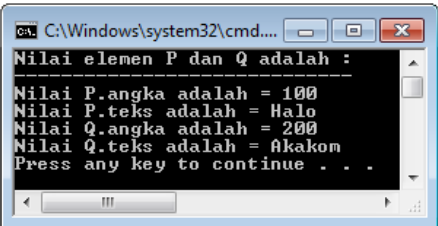


penjelasan: pointer P bisa menunjuk null karena sudah dideklarasikan jika `P == null`. Null juga bisa dilihat sebagai null. Instruksi yang menyebabkan P menunjuk ke null adalah :

```
tipePointer P;  
    P = null;  
    if (P == null)  
System.out.println("Pointer P menunjuk ke Null");
```

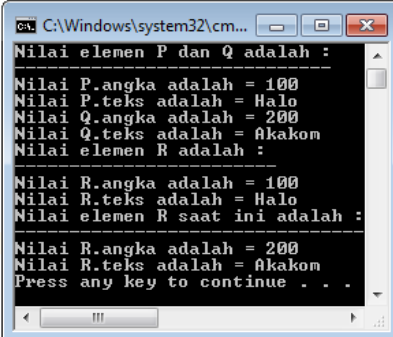
PRAKTIK 2

```
belajarPointer.java x
import java.util.Scanner;
class tipePointer
{ int angka;
String teks;
}
class belajarPointer
{
    public static void main(String[] args)
    {
        tipePointer P;
        P = new tipePointer();
        P.angka = 100;
        P.teks = "Halo";
        tipePointer Q = new tipePointer();
        Q.angka = 200;
        Q.teks = "Akakom";
        System.out.println("Nilai elemen P dan Q adalah :");
        System.out.println("-----");
        System.out.println("Nilai P.angka adalah = " + P.angka);
        System.out.println("Nilai P.teks adalah = " + P.teks);
        System.out.println("Nilai Q.angka adalah = " + Q.angka);
        System.out.println("Nilai Q.teks adalah = " + Q.teks);
    }
}
```



penjelasan: Nilai P angka adalah 100 dan Nilai P teks adalah Halo. Nilai Q angka adalah 200 dan Nilai Q teks adalah Akakom. Hal ini terjadi karena P.angka, P.teks, Q.angka, Q.teks sudah dideklarasikan di dalam tipePointer P dan tipePointer Q. Dan dipanggil kembali dengan System.out.println.

```
belajarPointer.java x
import java.util.Scanner;
class tipePointer
{
    int angka;
    String teks;
}
class belajarPointer
{
    public static void main(String[] args)
    {
        tipePointer P;
        P = new tipePointer();
        P.angka = 100;
        P.teks = "Halo";
        tipePointer Q = new tipePointer();
        Q.angka = 200;
        Q.teks = "Akakom";
        System.out.println("Nilai elemen P dan Q adalah :");
        System.out.println("-----");
        System.out.println("Nilai P.angka adalah = " + P.angka);
        System.out.println("Nilai P.teks adalah = " + P.teks);
        System.out.println("Nilai Q.angka adalah = " + Q.angka);
        System.out.println("Nilai Q.teks adalah = " + Q.teks);
        tipePointer R;
        R = P;
        System.out.println("Nilai elemen R adalah :");
        System.out.println("-----");
        System.out.println("Nilai R.angka adalah = " + R.angka);
        System.out.println("Nilai R.teks adalah = " + R.teks);
        R = Q;
        System.out.println("Nilai elemen R saat ini adalah :");
        System.out.println("-----");
        System.out.println("Nilai R.angka adalah = " + R.angka);
        System.out.println("Nilai R.teks adalah = " + R.teks);
    }
}
```



penjelasan: nilai angka dan teks yang ditunjuk pointer R pada percobaan pertama adalah 100 dan Halo, sementara pada percobaan kedua adalah 200 dan Akakom. Hal yang menyebabkan perbedaan antara percobaan pertama dan percobaan kedua adalah: dideklarasikannya R=P dan R=Q sebelum memunculkan angka dan teks. Pada percobaan pertama R mengikuti pointer P sementara di percobaan kedua R mengikuti pointer Q.

```
belajarPointer.java x
import java.util.Scanner;
class tipePointer
{ int angka;
  String teks;
}

class belajarPointer
{
    public static void main(String[] args)
    {
        tipePointer P = new tipePointer();
        P.angka = 100;
        P.teks = "Halo";
        tipePointer Q = new tipePointer();
        Q.angka = 200;
        Q.teks = "Akakom";
        System.out.println("Nilai P dan Q sebelum pointer dimanipulasi :");
        System.out.println("-----");
        System.out.println("Nilai P.angka adalah = " + P.angka);
        System.out.println("Nilai P.teks adalah = " + P.teks);
        System.out.println("Nilai Q.angka adalah = " + Q.angka);
        System.out.println("Nilai Q.teks adalah = " + Q.teks);
        tipePointer R;
        R = P;
        P = Q;
        Q = R;
        System.out.println("Nilai P dan Q setelah pointer dimanipulasi :");
        System.out.println("-----");
        System.out.println("Nilai P.angka adalah = " + P.angka);
        System.out.println("Nilai P.teks adalah = " + P.teks);
        System.out.println("Nilai Q.angka adalah = " + Q.angka);
        System.out.println("Nilai Q.teks adalah = " + Q.teks);
    }
}

C:\Windows\system32\cmd.exe
Nilai P dan Q sebelum pointer dimanipulasi :
-----
Nilai P.angka adalah = 100
Nilai P.teks adalah = Halo
Nilai Q.angka adalah = 200
Nilai Q.teks adalah = Akakom
Nilai P dan Q setelah pointer dimanipulasi :
-----
Nilai P.angka adalah = 200
Nilai P.teks adalah = Akakom
Nilai Q.angka adalah = 100
Nilai Q.teks adalah = Halo
Press any key to continue . . . _
```

Penjelasan: Sebelum manipulasi: nilai P angka dan teks adalah 100 dan Halo. Nilai Q angka dan teks adalah 200 dan Akakom. Setelah manipulasi: nilai P angka dan teks adalah 100 dan Akakom. Nilai Q angka dan teks adalah 100 dan Halo. Ini terjadi karena di tipePointer R (nilai setelah manipulasi) semua dideklarasikan terbalik. Maksudnya adalah R = P dan Q = R yang membuat nilai P menjadi nilai Q menjadi tertukar.

PRAKTIK 3 di halaman selanjutnya

PRAKTIK 3

```
belajarPointer.java x
1  import java.util.Scanner;
2  class tipePointer
3  {
4      String namaKota;
5      tipePointer kanan;
6  }
7  class belajarPointer
8  {
9      public static void main(String[] args)
10     {
11         tipePointer P;
12         P = new tipePointer();
13         P.namaKota = "Yogyakarta";
14         tipePointer Q;
15             Q = new tipePointer();
16             Q.namaKota = "Klaten";
17         tipePointer R;
18             R = new tipePointer();
19             R.namaKota = "Solo";
20         tipePointer S;
21             S = new tipePointer();
22             S.namaKota = "Sragen";
23         tipePointer T;
24             T = new tipePointer();
25             T.namaKota = "Ngawi";
26         System.out.println("Nilai P,Q,R,S,T adalah :");
27         System.out.println("-----");
28         System.out.println("Nilai P.namaKota adalah = " + P.namaKota);
29         System.out.println("Nilai Q.namaKota adalah = " + Q.namaKota);
30         System.out.println("Nilai R.namaKota adalah = " + R.namaKota);
31         System.out.println("Nilai S.namaKota adalah = " + S.namaKota);
32         System.out.println("Nilai T.namaKota adalah = " + T.namaKota);
33         P.kanan = Q;
34         Q.kanan = R;
35         R.kanan = S;
36         S.kanan = T;
37         T.kanan = null;
38         System.out.println("Nilai-nilai yang dapat diakses dari pointer P adalah :");
39         System.out.println("-----");
40         System.out.println(P.namaKota);
41         System.out.println(P.kanan.namaKota);
42         System.out.println(P.kanan.kanan.namaKota);
43         System.out.println(P.kanan.kanan.kanan.namaKota);
44         System.out.println(P.kanan.kanan.kanan.kanan.namaKota);
45     }
46 }
```

```
C:\Windows\system32\cmd.exe
Nilai P,Q,R,S,T adalah :
-----
Nilai P.namaKota adalah = Yogyakarta
Nilai Q.namaKota adalah = Klaten
Nilai R.namaKota adalah = Solo
Nilai S.namaKota adalah = Sragen
Nilai T.namaKota adalah = Ngawi
Nilai-nilai yang dapat diakses dari pointer P adalah :
-----
Yogyakarta
Klaten
Solo
Sragen
Ngawi
Press any key to continue . . .
```

penjelasan:

nilai masing-masing elemen:

System.out.println("Nilai P.namaKota adalah = " + P.namaKota); = Yogyakarta

System.out.println("Nilai Q.namaKota adalah = " + Q.namaKota); = Klaten

System.out.println("Nilai R.namaKota adalah = " + R.namaKota); = Solo

System.out.println("Nilai S.namaKota adalah = " + S.namaKota); = Sragen

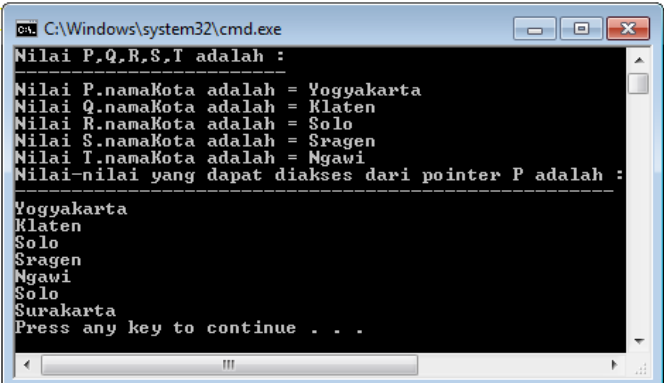
System.out.println("Nilai T.namaKota adalah = " + T.namaKota); = Ngawi

hasil yang diperoleh dengan deretan perintah:

System.out.println(P.namaKota);	= Yogyakarta
System.out.println(P.kanan.namaKota);	= Klaten
System.out.println(P.kanan.kanan.namaKota);	= Solo
System.out.println(P.kanan.kanan.kanan.namaKota);	= Sragen
System.out.println(P.kanan.kanan.kanan.kanan.namaKota);	= Ngawi

yang dapat disimpulkan adalah P.namaKota adalah Yogyakarta yang berposisi di sebelah paling kiri. Pointer kanan bertugas menghubungkan dengan heap sebelah kanan. Maka jika P.kanan.namaKota maka sama dengan Yogyakarta ditambah satu ke kanan yaitu ke Klaten. Jika ditambah pointer kanan lagi maka ke Solo dan begitu seterusnya.

```
38 System.out.println("Nilai-nilai yang dapat diakses dari pointer P adalah :");
39 System.out.println("-----");
40 System.out.println(P.namaKota);
41 System.out.println(P.kanan.namaKota);
42 System.out.println(P.kanan.kanan.namaKota);
43 System.out.println(P.kanan.kanan.kanan.namaKota);
44 System.out.println(P.kanan.kanan.kanan.kanan.namaKota);
45 System.out.println(R.namaKota);
46 P.kanan.kanan.namaKota = ("Surakarta");
47 System.out.println(R.namaKota);
48 }
49 }
```



Nilai P,Q,R,S,T adalah :

Nilai P.namaKota adalah = Yogyakarta
Nilai Q.namaKota adalah = Klaten
Nilai R.namaKota adalah = Solo
Nilai S.namaKota adalah = Sragen
Nilai T.namaKota adalah = Ngawi
Nilai-nilai yang dapat diakses dari pointer P adalah :
Yogyakarta
Klaten
Solo
Sragen
Ngawi
Solo
Surakarta
Press any key to continue . . .

Penjelasan: Yang terjadi pada R.namaKota adalah nilainya bertambah satu yaitu Surakarta. Pada program di atas terlihat ada upaya untuk mengganti elemen “Solo” yang sedang ditunjuk oleh pointer R menjadi “Surakarta” namun proses pengantiannya dilakukan melalui pointer P (perhatikan instruksi P.kanan.kanan.namaKota = "Surakarta"); Hal ini mengakibatkan nilai elemen R.namaKota akan berubah dari yang semula menampilkan “Solo” berubah menjadi “Surakarta”.

```
belajarPointer.java x
1 import java.util.Scanner;
2 class tipePointer
3 {
4     String namaKota;
5 tipePointer kanan;
6 }
7 class belajarPointer
8 {
9     public static void main(String[] args)
10 {
11     tipePointer P;
12     P = new tipePointer();
13     P.namaKota = "Yogyakarta";
14     tipePointer Q;
15     Q = new tipePointer();
16     Q.namaKota = "Klaten";
17     tipePointer R;
18     R = new tipePointer();
19     R.namaKota = "Solo";
20     tipePointer S;
21     S = new tipePointer();
22     S.namaKota = "Sragen";
23     tipePointer T;
24     T = new tipePointer();
25     T.namaKota = "Ngawi";
```

```

26 System.out.println("Elemen namaKota untuk pointer P,Q,R,S,T adalah :");
27 System.out.println("-----");
28 System.out.println("Nilai P.namaKota adalah = " + P.namaKota);
29 System.out.println("Nilai Q.namaKota adalah = " + Q.namaKota);
30 System.out.println("Nilai R.namaKota adalah = " + R.namaKota);
31 System.out.println("Nilai S.namaKota adalah = " + S.namaKota);
32 System.out.println("Nilai T.namaKota adalah = " + T.namaKota);
33     P.kanan = Q;
34     Q.kanan = R;
35     R.kanan = S;
36     S.kanan = T;
37 T.kanan = null;
38 System.out.println("Elemen namaKota untuk pointer P,Q,R,S,T adalah :");
39 System.out.println("-----");
40 System.out.println("Nilai P.namaKota adalah = " + P.namaKota);
41 System.out.println("Nilai Q.namaKota adalah = " + P.kanan.namaKota);
42 System.out.println("Nilai R.namaKota adalah = " + P.kanan.kanan.namaKota);
43 System.out.println("Nilai S.namaKota adalah = " +
44     P.kanan.kanan.kanan.namaKota);
45 System.out.print("Nilai T.namaKota adalah= "+
46     P.kanan.kanan.kanan.kanan.namaKota);
47     tipePointer BANTU;
48     BANTU = P;
49     while (BANTU!=null)
50     {
51 System.out.println("Nilai BANTU.namaKota adalah = " + BANTU.namaKota);
52 BANTU = BANTU.kanan;
53     }
54 }
55 }

```

```

C:\Windows\system32\cmd.exe
Elemen namaKota untuk pointer P,Q,R,S,T adalah :
-----
Nilai P.namaKota adalah = Yogyakarta
Nilai Q.namaKota adalah = Klaten
Nilai R.namaKota adalah = Solo
Nilai S.namaKota adalah = Sragen
Nilai T.namaKota adalah = Ngawi
Elemen namaKota untuk pointer P,Q,R,S,T adalah :
-----
Nilai P.namaKota adalah = Yogyakarta
Nilai Q.namaKota adalah = Klaten
Nilai R.namaKota adalah = Solo
Nilai S.namaKota adalah = Sragen
Nilai T.namaKota adalah= NgawiNilai BANTU.namaKota adalah = Yogyakarta
Nilai BANTU.namaKota adalah = Klaten
Nilai BANTU.namaKota adalah = Solo
Nilai BANTU.namaKota adalah = Sragen
Nilai BANTU.namaKota adalah = Ngawi
Press any key to continue . . .

```

Penjelasan: System.out.println("Nilai BANTU.namaKota adalah = " + BANTU.namaKota); dieksekusi sebanyak 4 kali yang menghasilkan nilai berbeda yaitu Klaten, Solo, Sragen, dan Ngawi. Mengapa demikian? Karena perintahnya adalah BANTU.kanan maka yang dieksekusi adalah nilai di sebelah kanan Yogyakarta yaitu ke-4 kota tersebut

```

1 import java.util.Scanner;
2     class tipePointer
3     {
4 String namaKota;
5 tipePointer kiri;
6 tipePointer kanan;
7     }
8     class belajarPointer
9     {
10     public static void main(String[] args)
11     {
12 tipePointer P;
13 P = new tipePointer();
14 P.namaKota = "Yogyakarta";
15 tipePointer Q;
16 Q = new tipePointer();
17 Q.namaKota = "Klaten";
18 tipePointer R;
19 R = new tipePointer();
20 R.namaKota = "Solo";
21 tipePointer S;
22 S = new tipePointer();
23 S.namaKota = "Sragen";
24 tipePointer T;

```

```
belajarPointer.java x
25 T = new tipePointer();
26 T.namaKota = "Ngawi";
27 System.out.println("Elemen namaKota untuk pointer P,Q,R,S,T adalah :");
28 System.out.println("-----");
29 System.out.println("Nilai P.namaKota adalah = " + P.namaKota);
30 System.out.println("Nilai Q.namaKota adalah = " + Q.namaKota);
31 System.out.println("Nilai R.namaKota adalah = " + R.namaKota);
32 System.out.println("Nilai S.namaKota adalah = " + S.namaKota);
33 System.out.println("Nilai T.namaKota adalah = " + T.namaKota);
34 P.kanan = Q;
35     Q.kanan = R;
36     R.kanan = S;
37     S.kanan = T;
38     T.kanan = null;
39     P.kiri = null;
40     Q.kiri = P;
41     R.kiri = R;
42     S.kiri = R;
43     T.kiri = S;
44     System.out.println("Elemen namaKota untuk pointer P adalah :");
45     System.out.println("-----");
46     System.out.println(P.namaKota);
47     System.out.println(P.kanan.kiri.namaKota);
48     System.out.println(P.kanan.kanan.kiri.kiri.namaKota);
49 tipePointer BANTU;
50 BANTU = T;
51 while (BANTU!=null)
52 { System.out.println("Nilai BANTU.namaKota adalah = " + BANTU.namaKota);
53 BANTU = BANTU.kanan;
54 }
55 }
56 }
```

```
C:\Windows\system32\cmd.exe
Elemen namaKota untuk pointer P,Q,R,S,T adalah :
-----
Nilai P.namaKota adalah = Yogyakarta
Nilai Q.namaKota adalah = Klaten
Nilai R.namaKota adalah = Solo
Nilai S.namaKota adalah = Sragen
Nilai T.namaKota adalah = Ngawi
Elemen namaKota untuk pointer P adalah :
-----
Yogyakarta
Yogyakarta
Solo
Nilai BANTU.namaKota adalah = Ngawi
Press any key to continue . . .
```

penjelasan: ketika BANTU.kanan dieksekusi maka muncul nilai berupa Ngawi. BANTU.kanan dieksekusi sebanyak satu kali. Hal ini bisa terjadi karena BANTU = T dan T.namaKota = Ngawi. Yogyakarta bisa muncul 2 kali karena diputar balik menggunakan P.kanan.kiri.namaKota. Lalu Solo karena P.kanan.kanan.kiri.kiri.namaKota yang membuat posisi selangkah ke kanan dari Yogyakarta.

KESIMPULAN

Pointer (Penunjuk) adalah sebuah variabel bertipe khusus yang dapat menunjuk (menyimpan alamat) sebuah variabel/ heap/ obyek. Tipe data dari variabel pointer (selanjutnya disebut pointer) tidak bertipe primitif sebagaimana int, char, boolean melainkan bertipe data buatan sehingga harus dideklarasikan terlebih dahulu menggunakan sebuah class (pelajari kembali modul 2 bagian “Struktur Penyimpan Terstruktur berbasis Record/Rekaman”).

Ada tiga karakteristik dari sebuah pointer yaitu :

- a. Pointer dapat menunjuk ke NULL
- b. Pointer dapat menunjuk ke sebuah HEAP
- c. Pointer dapat saling mengcopy (alamat) heap yang sedang ditunjuk