

Tugas 3 PBO

Muh.Afrizaldi Attalah/13020220057/A2

1. Baca String

```
1 import java.io.BufferedReader;
2 import java.io.IOException;
3 import java.io.InputStreamReader;
4 import java.sql.SQLOutput;
5 import javax.swing.*;
6
7 public class BacaString {
8     public static void main(String[] args) throws IOException {
9         String str;
10
11         BufferedReader input = new BufferedReader(new InputStreamReader(System.in));
12
13         System.out.println("\nBaca String dan Integer: \n");
14         System.out.println("Masukkan sebuah String: \n");
15         str= input.readLine();
16         System.out.print ("Sting yang dibaca: "+ str);|
17     }
18 }
```

```
Baca String dan Integer:

Masukkan sebuah String:

hallo zzz
Sting yang dibaca: hallo zzz
```

Penjelasan:

code program diatas menjelaskan tentang cara program membaca input bertipe data string dengan menggunakan class Buffereader dan inputstreamreader, dimana inputStreamReader berguna untuk menerima data dengan lebih efisien, lalu data akan dibaca dan diinput degan bufferreader ke variable str. Kelas IOException berguna sebagai manajemen kesalahan dalam program. Misal, file tidak ditemukan atau tidak dapat diakses,dapat menggunakan throws IOException untuk memberi tahu pemanggil metode bahwa perlu menangani situasi tersebut.

2. ForEver

```
1 public class ForEver {
2     public static void main(String[] args){
3         System.out.println("Program akan looping, akhiri dengan ^c");
4         while(true){
5             System.out.println("Print satu baris ...\n");
6         }
7     }
8 }
9
```

```
Print satu baris ...  
  
Print satu baris ...  
  
Print satu baris ...  
  
Print satu baris ...  
  
Print satu baris ...  
  
Print satu baris ...  
  
Print satu baris ...
```

Penjelasan:

Pada code Program ini, program akan menunjukkan program yang me looping kata “print satu baris”. Perulangan yang dipakai yaitu perulangan while, dimana looping while yaitu looping yang berjenis uncounted loop yaitu looping yang digunakan ketika belum mengetahui jumlah proses perulangan, banyaknya perulangan ditentukan dari proses inialisasi dan kondisi yang didefinisikan. dikarenakan kondisi dari while bernilai true dan tidak ada Batasan yang menyatakan program untuk berhenti maka looping akan terus dilakukan selamanya.

3. IF1

```
1  import java.util.Scanner;  
2  
3  public class If1 {  
4      public static void main(String []args){  
5          Scanner Input= new Scanner(System.in);  
6  
7          System.out.println("Contoh IF satu kasus \n");  
8          System.out.println("Ketikkan suatu nilai integer: ");  
9          int a = Input.nextInt();  
10         if(a >= 0){  
11             System.out.print("\nNilai a Positif "+ a);  
12         }  
13     }  
14 }
```

Contoh IF satu kasus

Ketikkan suatu nilai integer:

5

Nilai a Positif 5

Penjelasan:

Pada program diatas merupakan implementasi dari if. Dimana ekspresi dari if yang menentukan jika bilangan bulat yang diinput di atas angka 0 atau sama dengan 0, maka akan tampil teks yang menyatakan hasil inputan adalah bilangan positif.

4. IF2

```
1  import java.util.Scanner;
2  public class If2 {
3      public static void main(String[] args){
4          int a;
5          Scanner input= new Scanner(System.in);
6
7          System.out.println("Contoh IF dua kasus\n");
8
9          System.out.println("Ketikkan suatu nilai integer: ");
10         a= input.nextInt();
11
12         if(a>=0){
13             System.out.println("Nilai suatu nilai "+ a);
14         }else{
15             System.out.println("Nilai a negatif "+ a);
16         }
17     }
18 }
```

```
Contoh IF dua kasus

Ketikkan suatu nilai integer:
-1
Nilai a negatif -1
```

Penjelasan:

Kode program IF2 mirip dengan yang pertama namun dengan kondisi yang ditambahkan pada else. Dimana kondisi tersebut berarti yang hasil yang akan masuk diluar dari kondisi pertama yang berarti selain kondisi tersebut yaitu bilangan negative.

5. IF3

```
1 import java.util.Scanner;
2 public class If3 {
3     public static void main(String[] args){
4         System.out.println("Contoh IF tiga kasus \n");
5         System.out.println("Input suatu nilai integer: ");
6         Scanner input=new Scanner(System.in);
7         int a= input.nextInt();
8
9         if(a > 0){
10             System.out.println("Nilai a positif "+ a);
11         }else if(a == 0){
12             System.out.println("Nilai 0 "+ a);
13         }else{
14             System.out.println("Nilai a negatif "+a);
15         }
16     }
17 }
```

Contoh IF tiga kasus

Input suatu nilai integer:

0

Nilai 0 0

Penjelasan:

Pada kode program diatas menjelaskan tentang Penggunaan if dengan tiga kasus, dimana kasus terbaru nya yaitu Ketika inputan yang dimasukkan adalah angka 0 maka program akan dieksekusi di kasus tersebut.

6. KasusBoolean

```
1 public class KasusBoolean {
2     public static void main(String[] args){
3         boolean b;
4         b= true;
5
6         if(b){
7             System.out.println("true\n");
8         }else{
9             System.out.println("false\n");
10        }
11        if(!b){
12            System.out.println("false\n");
13        }else{
14            System.out.println("benar\n");
15        }
16    }
17 }
```

Encoding on 0 Boolean: true

benar

Penjelasan:

Kode program diatas menjelaskan tentang penggunaan if namun dengan variable dengan tipe data Boolean, dimana dalam ekspresi hanya menampilkan variable saja tanpa operator dikarenakan variable sudah memiliki nilai true yang berarti sudah bisa dieksekusi dan memenuhi syarat ekspresinya.

lalu untuk ekspresi dengan tanda seru b(!b) berarti dapat dieksekusi jika nilai dari b yaitu false atau bisa dibilang negasi dari b(true). Maka dari itu yang tereksekusi yaitu bagian elsenya.

7. Konstant

```
1  import java.util.Scanner;
2  public class Konstant {
3      public static void main(String[] args){
4          final float PHI= 3.1415f;
5          float r;
6          Scanner input= new Scanner(System.in);
7
8          System.out.println("Jari-jari lingkaran= ");
9          r= input.nextFloat();
10
11         System.out.println("Luas lingkaran= "+ (PHI *r*r) + "\n");
12         System.out.println("Akhiri program \n");
13     }
```

```
Jari-jari lingkaran=
12
Luas lingkaran= 452.37598

Akhiri program
```

Penjelasan:

program diatas sedang melakukan proses untuk mrnghitung luas lingkaran dengan rumus yang terdapat pada pemanggilannya. Sehingga tidak memerlukan variable lain untuk membuat rumus. tipe data yang digunakan yaitu tipe data float sesuai kegunaannya dikarenakan memerlukan bilangan pecahan(decimal) untuk mendapat hasil yang sesuai.

8. Konstant

```
1 import java.util.Scanner;
2 public class Max2 {
3     public static void main(String[] args){
4         int a, b;
5         Scanner input= new Scanner(System.in);
6
7         System.out.println("Maksimum dua bilangan: \n");
8         System.out.println("Ketikkan dua bilangan, pisahkan dengan return: \n");
9
10        a=input.nextInt();
11        b=input.nextInt();
12        System.out.println("Ke dua bilangan : a = "+a+" b= "+b);
13        if (a >= b){
14            System.out.println("Nilai a yang maksimum "+ a);
15        }else{
16            System.out.println("Nilai b yang maksimum: "+b);
17        }
18    }
19 }
```

```
Maksimum dua bilangan:

Ketikkan dua bilangan, pisahkan dengan return:

2
3
Ke dua bilangan : a = 2 b= 3
Nilai b yang maksimum: 3
```

Penjelasan:

pada program diatas menjelaskan tentang program yang dapat membaca nilai tertinggi dari dua bilangan, dimana hanya menggunakan if untuk membandingkan diantara dua variable yang telah diinput suatu nilai bilangan bulat, dimana jika bilangan pertama(a) lebih besar atau sama dengan bilanga kedua(b) maka program yang dieksekusi menampilkan teks bahwa bilangan kedua adalah bilangan maksimal, begituoun dengan kondisi berikutnya(sebaliknya).

9. PriFor

```
1 import java.util.Scanner;
2 public class PriFor {
3     public static void main(String[] args){
4         int i, N;
5         Scanner input= new Scanner(System.in);
6
7         System.out.println("Baca N, print 1 sampai N");
8         System.out.println("N = ");
9
10        N=input.nextInt();
11
12        for(i=1; i<=N; i++) {
13            System.out.println(i);
14            System.out.println("Akhiri program \n");
15        }
```

```
Baca N, print 1 sampai N
N =
2
1
Akhiri program

2
Akhiri program
```

Penjelasan:

Program diatas merupakan implementasi dari perulangan for, dimana for adalah looping yang termasuk jenis counted loop artinya bisa diketahui batas loopnya. Pada program diatas telah ditentukan i=1 yang berarti akan dimulai dari satu hingga parameter kedua terpenuhi atau bernilai false.

10. PrintIterasi

```
1 import java.util.Scanner;
2 public class PrintIterasi {
3     public static void main(String[] args){
4         int N;
5         int i;
6         Scanner input= new Scanner(System.in);
7
8         System.out.println("Nilai N > 0 = ");
9         N= input.nextInt();
10
11         i= 1;
12         System.out.println("Print i dengan ITERATE: \n");
13         for (;;){
14             System.out.println(i);
15             if(i == N){
16                 System.out.println("Kondisi berhenti");
17                 break;
18             }else{
19                 System.out.println(" ");i++;
```

```
Nilai N > 0 =
2
Print i dengan ITERATE:

1

2
Kondisi berhenti
```

Penjelasan:

pada program diatas sama seperti sebelumnya namun dengan cara yang berbeda, dimana perulangan loop for tidak memiliki parameter yang di set, dan agar program dapat melakukan increment maka incremennya ditambahkan di dalam perulangan sebelum loop dilakukan Kembali,dan untuk membuat program berhenti maka digunakan if dengan kondisi Ketika bilangan telah mencapai batas dari if atau n. dan else nya tetap akan melanjutkan looping dan melakukan increment.

.11.PrintRepeat

```
1 import java.util.Scanner;
2 public class PrintRepeat {
3     public static void main(String[] args){
4         int N, i;
5         Scanner input= new Scanner(System.in);
6         System.out.println("Nilai N>0 = ");
7         N= input.nextInt();
8
9         i=1;
10        System.out.println("Print i dengan REPEAT");
11        do{
12            System.out.println(i+"\n");
13            i++;
14        }while (i <= N);
15    }
16 }
```

```
Nilai N>0 =
2
Print i dengan REPEAT
1
2
```

Penjelasan:

program diatas sama bertujuan sama dengan sebelumnya namun dengan menggunakan metode looping yang berbeda yaitu dengan menggunakan do-while, dimana sebelum masuk kepada perulangan, variable dideklarasikan terlebih dahulu dengan nilai awalnya yaitu i=1. Lalu masuk kepada perulangan dimana langsung di print dan di increment lalu Ketika i belum mencapai N program akan di looping Kembali.

12. PrintwHile

```
import java.util.Scanner;
public class PrintWhile {
    public static void main(String[] args){
        int N, i;

        Scanner input= new Scanner(System.in);
        N= input.nextInt();
        i=1;

        System.out.println("Print i dengan WHILE: \n");
        while (i <= N){
            System.out.println(i);
            i++;
        }
    }
}
```

```
3
Print i dengan WHILE:

1
2
3
```

Penjelasan:

Program diatas juga mirip dengan sebelumnya namun dengan metode perulangan while, dimana yang parameter dari while hanyalah ekspresi yang berguna untuk menghentikan loop, dan untuk increment dilakukan didalam while, lalu untuk pendeklarasian nilai awal dilakukan diluar while atau loop.

13. PrintXiterasi

```
1 import java.util.Scanner;
2 public class PrintXiterasi {
3     public static void main(String[] args){
4         int Sum=0;
5         int x;
6
7         Scanner input= new Scanner(System.in);
8         System.out.println("Masukkan nilai x (int), akhiri dengan 999: ");
9
10        x= input.nextInt();
11        if(x == 999){
12            System.out.println("Kasus kosong\n");
13        }else{
14            Sum= x;
15            for(;;){
16                System.out.println("Masukkan nilai x (int), akhiri dengan 999: ");
17                x= input.nextInt();
18                if(x==999){
19                    break;
20                }else{
21                    Sum= Sum + x;
22                }
23            }
24            System.out.println("Hasil penjumlahan= "+ Sum);
25        }
```

Penjelasan:

Program diatas menjelaskan proses penjumlahan dari angka yang diinput, dimana proses nya terjafi pada didalam pengkondisian if Ketika inputan bukan bernilai 999, dan kemudian dibungkus oleh for agar perulangan terus dilakukan Ketika program belum mencapai break; atau saat input yang dimasukkan adalah 999.

14. PrintXRepeat

```
1 import java.util.Scanner;
2 public class PrintXRepeat {
3     public static void main(String[] args){
4         int Sum, x;
5
6         Scanner input= new Scanner(System.in);
7
8         System.out.println("Masukkan nilai x (int), akhiri dengan 999: ");
9
10        x= input.nextInt();
11        if(x==999){
12            System.out.println("Kasus kosong \n");
13        }else{
14            Sum= 0;
15
16            do{
17                Sum= Sum+ x;
18                System.out.println("Masukkan nilai x (int), akhiri dengan 999: ");
19                x= input.nextInt();
20            }while(x != 999);
21            System.out.println("Hasil penjumlahan= " + Sum);
22        }
```

```
Masukkan nilai x (int), akhiri dengan 999:
12
Masukkan nilai x (int), akhiri dengan 999:
12
Masukkan nilai x (int), akhiri dengan 999:
999
Hasil penjumlahan= 24
```

Penjelasan:

Program diatas sama dengan program sebelumnya dimana melakukan proses penambahan terhadap jumlah penambahan sebelumnya. Namun, di program ini menggunakan cara yang berbeda yakni dengan menggunakan perulangan do-while yang dibungkus dengan pengkondisian else dari if yang bereksoresi = 999, yang berarti Ketika input bernilai 999 lalu program akan di berhentikan dan perulangan tidak akan dieksekusi sedangkan else merupakan inputan selain 999, proses penjumlahan dilakukan didalam loop dan akan berhenti Ketika inputan juga bernilai 999 pada ekspresi while.

15. PrintXRepeat

```
1 import java.util.Scanner;
2 public class PrintXWhile {
3     public static void main(String[] args) {
4         int Sum, x;
5         Scanner input= new Scanner(System.in);
6         Sum= 0;
7
8         System.out.println("Masukkan nilai x (int, akhiri dengan 999: );");
9         x= input.nextInt();
10        while(x != 999){
11            Sum=Sum+x;
12            System.out.println("Masukkan nilai x (int), akhiri dengan 999: ");
13            x= input.nextInt();
14        }
15        System.out.println("Hasil penjumlahan= "+ Sum);
16    }
```

```
Masukkan nilai x (int, akhiri dengan 999: )
12
Masukkan nilai x (int), akhiri dengan 999:
12
Masukkan nilai x (int), akhiri dengan 999:
999
Hasil penjumlahan= 24
```

Penjelasan:

program diatas juga sama dengan program sebelumnya namun dengan cara yang berbeda, para program diatas menggunakan perulangan while dan loop akan bekerja selama ekspresi while tidak mencapai nilai false, dimana nilai falsenya Ketika nilai yang diinput adalah 999,

16.SubProgram

```
import java.util.Scanner;
public class SubProgram {
    public static int maxab(int a, int b){
        return ((a >= b) ? a:b);
    }

    public static void tukar(int a, int b){
        int temp;
        temp= a;
        a=b;
        b=temp;
        System.out.println("Ke dua bilangan setelah tukar: a= "+a+" b= "+b);
    }

    public static void main(String[] args){
        int a, b;
        Scanner input= new Scanner(System.in);
        System.out.println("Maksimum dua bilangan\n");
        System.out.println("Ketikkan dua bilangan, pisahkan dengan RETURN: \n");
        a= input.nextInt();
        b= input.nextInt();
        System.out.println("Ke dua bilangan : a = "+a+" b= "+b);
        System.out.println("Maksimum= "+ (maxab(a, b)));
        System.out.println("Tukar kedua bilangan...\n");
        tukar(a, b);
    }
```

```

Maksimum dua bilangan

Ketikkan dua bilangan, pisahkan dengan RETURN:

2
3
Ke dua bilangan : a = 2 b= 3
Maksimum= 3
Tukar kedua bilangan...

Ke dua bilangan setelah tukar: a= 3 b= 2

```

Penjelasan:

Pada program diatas terdapat class subprogram dimana kelas tersebut terdapat method yang diberi nama tukar dan method maxab, dalam method maxab terdapat suatu proses return ((a >= b) ? a:b); yang berarti akan menghasilkan nilai true Ketika ekspresi true maka akan menampilkan nilai dari angka 1, sedangkan Ketika ekspresi false akan menampilkan nilai dari b.

lalu pada method tukar terdapat proses untuk menukar nilai dari a dan b, proses tersebut menggunakan variable temp untuk dijadikan wadah sebagai duplikasi nilai dari variable sehingga penukaran nilai bisa dilakukan.

Sedangkan pada main program hanya melakukan proses pemanggilan kedua method dan melakukan sesuai dengan alur program mulai dari input hingga tukar, dan yang perlu diperhatikan disini yaitu Ketika melakukan pemanggilan method tukar, tidak memerlukan objek yang perlu dibuat dikarenakan method menggunakan kata kunci static.

17. Switch

```

1  import java.util.Scanner;
2
3  public class Switch {
4      public static void main(String[] args) {
5          char cc;
6          Scanner input = new Scanner(System.in);
7
8          System.out.println("Ketikkan sebuah huruf, akhiri dengan Return\n");
9          cc = input.next().charAt(0);
10
11         switch (cc) {
12             case 'a': {
13                 System.out.println("Yang anda ketik adalah a\n");
14                 break;
15             }
16             case 'i': {
17                 System.out.println("Yang anda ketik adalah i\n");
18                 break;
19             }
20             case 'u': {
21                 System.out.println("Yang anda ketik adalah u\n");
22                 break;
23             }
24             case 'e': {
25                 System.out.println("Yang anda ketik adalah e\n");

```

```

    }
    case 'o': {
        System.out.println("Yang anda ketik adalah o\n");
        break;
    }
    default:
        System.out.println("Yang anda ketik adalah huruf mati\n");
    }
}
}

```

Ketikkan sebuah huruf, akhiri dengan Return

```

e
Yang anda ketik adalah e

```

Penjelasan:

Pada program diatas menjelaskan tentang penggunaan sederhana dari Switch, dimana switch merupakan selection yang bekerja tidak seperti if-else, switch bekerja dengan cara langsung mengoper inputan yang di jadikan ekspresi pada switch ke case atau code yang sesuai dengan input tanpa mengecek satu satu pengkondisian yang ada(case).

18.Tempair

```

1  import java.util.Scanner;
2  public class Tempair {
3      public static void main(String[] args) {
4          int T;
5          Scanner input=new Scanner(System.in);
6
7          System.out.print ("Contoh IF tiga kasus \n");
8          System.out.print ("Temperatur (der. C) = ");
9          T=input.nextInt();
10         if (T < 0) {
11             System.out.print ("Wujud air beku \n"+ T);
12         }else if ((0 <= T) && (T <= 100)){
13             System.out.print ("Wujud air cair \n"+ T);
14         }else if (T > 100){
15             System.out.print ("Wujud air uap/gas \n"+ T);
16         };
17     }
18 }

```

```
Contoh IF tiga kasus
Temperatur (der. C) = -2
Wujud air beku
-2
Process finished with exit code 0
|
```

Penjelasan:

pada program diatas menjelaskan tentang penggunaan selection if-else pada temperature, dimana Ketika nilai T yang diinput kurang dari 0 maka akan menampilkan teks "wujud air beku".

lalu Ketika T nilai mempunyai yang diinput 0 sampai dengan kurang atau sama dengan 100 maka akan menampillkan wujud cair.

dan yang terakhir yaitu pada bagian elsesnya yang akan menampilkanwujud air uap/gas.