

**LAPORAN PRAKTIKUM**  
**ALGORITMA DAN PEMROGRAMAN LANJUT**  
**PERTEMUAN KE-10**



**Disusun Oleh :**

**NAMA : Raden Isnawan Argi Aryasatya**

**NIM : 195410257**

**JURUSAN : Informatika**

**JENJANG : S1**

**SEMESTER ANTARA**

**Sekolah Tinggi Manajemen Informatika Komputer**

**AKAKOM**

**YOGYAKARTA**

**2021**

## **PERTEMUAN KE-10** **(ITERASI SELEKSI DALAM METHOD)**

### **TUJUAN**

Mahasiswa dapat menyelesaikan kasus dengan menggabungkan konsep iterasi, seleksi dalam method serta dapat memanggil method dari class lain

### **DASAR TEORI**

Pernyataan seleksi, perulangan dan fungsi sudah dibahas pada pertemuan sebelumnya. Pada dasarnya pemakaian ketiganya dapat digabungkan dalam suatu array, baik untuk data tunggal maupun data berupa array. Untuk modul kali ini akan dipraktekkan beberapa program yang menggabungkan seleksi, perulangan dalam suatu fungsi.

Bentuk iterasi / perulangan dalam fungsi adalah sebagai berikut:

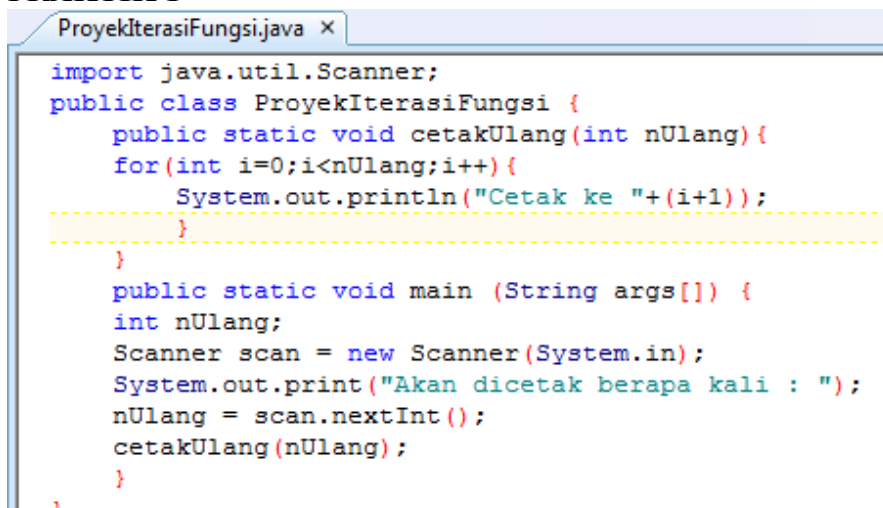
```
Fungsi(){  
    .....  
    Perulangan dengan for, do...while, atau while  
    .....  
}
```

Bentuk seleksi dalam fungsi adalah sebagai berikut.

```
Fungsi(){  
    .....  
    Pilihan dengan if...else atau switch...case  
    .....  
}
```

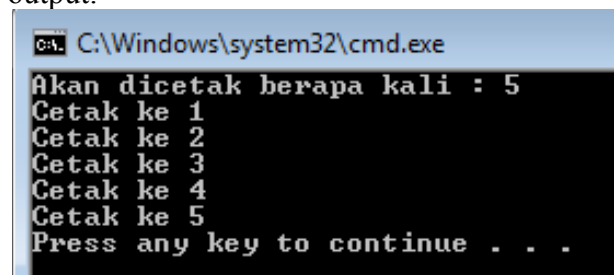
---

### **PRAKTIK 1**



```
ProyekIterasiFungsi.java x  
import java.util.Scanner;  
public class ProyekIterasiFungsi {  
    public static void cetakUlang(int nUlang) {  
        for(int i=0; i<nUlang; i++) {  
            System.out.println("Cetak ke " + (i+1));  
        }  
    }  
    public static void main (String args[]) {  
        int nUlang;  
        Scanner scan = new Scanner(System.in);  
        System.out.print("Akan dicetak berapa kali : ");  
        nUlang = scan.nextInt();  
        cetakUlang(nUlang);  
    }  
}
```

output:



```
C:\Windows\system32\cmd.exe  
Akan dicetak berapa kali : 5  
Cetak ke 1  
Cetak ke 2  
Cetak ke 3  
Cetak ke 4  
Cetak ke 5  
Press any key to continue . . .
```

penjelasan:

inti dari program di praktik ini adalah kita membuat sebuah perulangan di dalam fungsi atau method yang kita buat di program tersebut. Dengan kata lain, kita menggunakan prinsip iterasi di dalam program ini. Di dalam komputer/pemrograman, iterasi adalah sifat tertentu dari algoritma atau program komputer di mana suatu urutan dari langkah algoritmik dilaksanakan di loop program. Iterasi juga dapat diartikan sebagai suatu metode atau cara yang dipakai secara berulang-ulang (pengulangan) dalam menyelesaikan suatu permasalahan matematik. Untuk melaksanakan konsep iterasi di dalam method yang kita buat, kita gunakan perulangan atau loop for. Perulangan for adalah perulangan yang digunakan ketika kita ingin mengenskusi perintah program yang sama dengan jumlah proses perulangan yang sudah diketahui dengan mengacu pada kondisi yang ditetapkan.

Untuk membuat program di atas, kita awali dengan kita import scanner dengan menuliskan baris kode **import java.util.Scanner;**. Kemudian kita tulis **public class ProyekIterasiFungsi** untuk mendeklarasikan class. Class tersebut bermodifier public yaitu keyword yang berfungsi untuk menentukan sifat akses ke semua member yang mengikutinya (di bawahnya), sehingga memiliki sifat dapat di akses dari manapun, dan dapat di akses dari dalam class itu sendiri, dari anak class (derived class) dan juga dari luar class.

Di dalamnya langsung kita buat sebuah method bernama cetakUlang yang memiliki parameter berupa sebuah variabel bertipe data integer yaitu int nUlang. Untuk melakukan itu, kita tulis baris kode **public static void cetakUlang(int nUlang)**. Di dalam method atau fungsi tersebut kita beri perulangan **for(int i=0;i<nUlang;i++)**. Di perulangan itu, **i=0** berarti perulangan i dimulai dari 0. **i<nUlang** berarti selama i bernilai kurang dari **nUlang** (berapa banyak data yang kita perintahkan kepada program untuk diproses), maka perulangan akan terus berjalan. **i++** fungsinya adalah untuk menambah satu (+1) nilai hitungan pada setiap perulangan. Kemudian kita beri perintah pada program untuk menampilkan sebuah pesan dari method cetakUlang yaitu dengan menuliskan baris kode **System.out.println("Cetak ke "+(i+1))**.

Langkah selanjutnya adalah membuat main method yang eksekusinya diawali dengan **public static void main (String args[])**. Public, dapat diartikan bahwa argumen atau metode main() merupakan sebuah metode yang bersifat publik. void merupakan suatu tipe data yang menyatakan bahwa deklarasi code tidak memerlukan nilai balik atau return. String menyatakan bahwa argumen yang terdapat pada code java tersebut merupakan String. Args merupakan argumen bertipe data string yang mengandung array, Args dapat di misalkan sebagai memory untuk menampung program. Lalu dilanjutkan dengan deklarasi variabel nUlang (**int nUlang**). Lalu dilanjutkan dengan memasukkan fungsi scanner supaya kita nanti bisa membuat kode perintah supaya program bisa memasukkan data yang diinput oleh user yaitu dengan menuliskan **Scanner scan = new Scanner(System.in)**. Dengan begitu program akan menampilkan sebuah tempat bagi user untuk memasukkan data yaitu berupa pesan **System.out.print("Akan dicetak berapa kali : ")**. Program akan memproses input data yang dimasukkan oleh user tadi dengan **nUlang = scan.nextInt();**. Terakhir, kita tampilkan hasil dengan memanggil method cetakUlang beserta parameternya yaitu **cetakUlang(nUlang)**.

## PRAKTIK 2

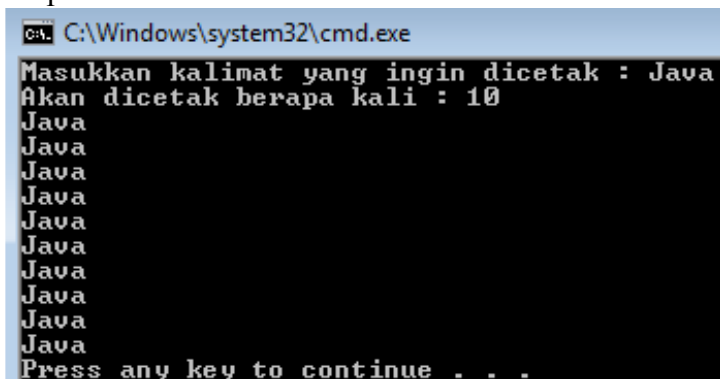
```
ProyekIterasiFungsi.java x
1  import java.util.Scanner;
2  public class ProyekIterasiFungsi {
3      public static void cetakUlang(String cetak, int nUlang){
4          for(int i=0;i<nUlang;i++){
5              System.out.println(cetak);
6          }
7      }
8      public static void main (String args[]) {
```

```

9      String cetak;
10     int nUlang;
11     Scanner scan = new Scanner(System.in);
12     System.out.print("Masukkan kalimat yang ingin dicetak : ");
13     cetak = scan.next();
14     System.out.print("Akan dicetak berapa kali : ");
15     nUlang = scan.nextInt();
16     cetakUlang(cetak, nUlang);
17 }
18

```

output:



```

C:\Windows\system32\cmd.exe
Masukkan kalimat yang ingin dicetak : Java
Akan dicetak berapa kali : 10
Java
Java
Java
Java
Java
Java
Java
Java
Java
Java
Java
Press any key to continue . . .

```

penjelasan:

program ini hampir sama dengan program di praktik 1 tadi. Perbedaannya adalah, di program ini user bebas untuk menginput pesan atau kalimat apa yang akan dicetak. User juga bisa menentukan berapa kali kalimat akan dicetak. Untuk membuat program di atas, kita import scanner dengan menuliskan kode **import java.util.Scanner;**. Kemudian tulis **public class ProyekIterasiFungsi** untuk mendeklarasikan nama class. Class tersebut bermodifler public yaitu keyword yang berfungsi untuk menentukan sifat akses ke semua member yang mengikutinya (di bawahnya), sehingga memiliki sifat dapat di akses dari manapun, dan dapat di akses dari dalam class itu sendiri, dari anak class (derived class) dan juga dari luar class.

Kemudian langsung kita buat method cetakUlang seperti di praktik 1 tadi. Bedanya, di method ini kita menambahkan satu parameter lagi yaitu parameter berupa variabel String bernama "cetak" yang merupakan variabel untuk menampung data kalimat atau pesan apa yang ingin kita cetak nanti. Jadi, kita membuat method cetakUlang dengan menuliskan **public static void cetakUlang(String cetak, int nUlang)**. Di dalam method tersebut kita beri perulangan **for(int i=0;i<nUlang;i++)**. Di perulangan itu, **i=0** berarti perulangan i dimulai dari 0. **i<nUlang** berarti selama i bernilai kurang dari **nUlang** (berapa banyak data yang kita perintahkan kepada program untuk diproses), maka perulangan akan terus berjalan. **i++** fungsinya adalah untuk menambah satu (+1) nilai hitungan pada setiap perulangan. Lalu kita perintahkan program untuk menampilkan kalimat atau pesan yang sudah kita masukkan sebelumnya yaitu **System.out.println(cetak)**.

Langkah selanjutnya adalah membuat main method yang eksekusinya diawali dengan public static void main (String args[]). Public, dapat diartikan bahwa argumen atau metode main() merupakan sebuah metode yang bersifat publik. void merupakan suatu tipe data yang menyatakan bahwa deklarasi code tidak memerlukan nilai balik atau return. String menyatakan bahwa argumen yang terdapat pada code java tersebut merupakan String. Args merupakan argumen bertipe data string yang mengandung array, Args dapat di misalkan sebagai memory untuk menampung program. Lalu kita deklarasikan dua variabel yaitu **String cetak;** dan **int nUlang;**. Dilanjutkan dengan memasukkan fungsi scanner supaya kita nanti bisa membuat kode perintah supaya program bisa memasukkan data yang diinput oleh user yaitu dengan menuliskan baris kode **Scanner scan = new Scanner(System.in)**.

Lalu program memberikan field atau tempat untuk user supaya bisa memasukkan kalimat yang ingin dicetak yaitu dengan **System.out.print("Masukkan kalimat yang ingin dicetak : ")**. Lalu program akan langsung memproses input data yang dimasukkan oleh user tadi dengan **cetak = scan.next()**. Kemudian program memberikan field lagi supaya user bisa memasukkan data untuk berapa kali kalimat akan dicetak yaitu dengan **System.out.print("Akan dicetak berapa kali : ")**. program akan langsung memproses input data yang dimasukkan oleh user tadi dengan **nUlang = scan.nextInt()**. Terakhir, kita tampilkan hasilnya dengan memanggil method cetakUlang beserta seluruh parameternya yaitu dengan menuliskan baris kode **cetakUlang(cetak, nUlang)**.

### PRAKTIK 3

```
ProyekHitungJumlah.java x
import java.util.Scanner;

public class ProyekHitungJumlah{

    public static int hitungJumlah (int[] x){
        int jum=0;
        for(int i=0;i<x.length;i++){
            jum+=x[i];
        }
        return jum;
    }
    public static void main (String args[]){
        int data[] = new int[10];
        int jumlah;
        Scanner scan = new Scanner(System.in);
        for(int i=0;i<10;i++){
            System.out.print("Masukkan data ke-"+(i+1)+":");
            data[i] = scan.nextInt();
        }
        jumlah = hitungJumlah(data);
        System.out.println("Jumlah data = "+jumlah);
    }
}
```

output:

```
C:\Windows\system32\cmd.exe
Masukkan data ke-1:2
Masukkan data ke-2:4
Masukkan data ke-3:6
Masukkan data ke-4:8
Masukkan data ke-5:10
Masukkan data ke-6:12
Masukkan data ke-7:14
Masukkan data ke-8:16
Masukkan data ke-9:18
Masukkan data ke-10:20
Jumlah data = 110
Press any key to continue . . .
```

penjelasan:

program ini merupakan program yang berfungsi untuk menghitung hasil penjumlahan dari 10 data yang dimasukkan oleh user. Untuk membuat program di atas, bisa kita awali dengan mengimport scanner yaitu dengan **import java.util.Scanner**; Setelah itu kita deklarasikan nama class yaitu dengan menuliskan baris kode **public class ProyekHitungJumlah**. Class tersebut bermodififier public yaitu keyword yang berfungsi untuk menentukan sifat akses ke semua member yang mengikutinya (di bawahnya), sehingga memiliki sifat dapat di akses dari manapun, dan dapat di akses dari dalam class itu sendiri, dari anak class (derived class) dan juga dari luar class. Kemudian

langsung kita buat sebuah method untuk menghitung penjumlahan sekaligus membuat perulangan yaitu **public static int hitungJumlah (int[] x)**. Di dalamnya kita buat variabel **jum=0** yang artinya value jum dimulai dari 0. Lalu kita tambahkan perulangan for. Perulangan for adalah perulangan yang digunakan ketika kita ingin mengeksekusi perintah program yang sama dengan jumlah proses perulangan yang sudah diketahui dengan mengacu pada kondisi yang ditetapkan. Perulangan yang kita buat di dalam method hitungJumlah adalah **for(int i=0;i<x.length;i++)**. **i=0** berarti perulangan i dimulai dari 0. **i<x.length** berarti selama i bernilai kurang dari **x.length** (berapa banyak data yang kita perintahkan kepada program untuk diproses), maka perulangan akan terus berjalan. **i++** fungsinya adalah untuk menambah satu (+1) nilai hitungan pada setiap perulangan. Lalu kita tulis **jum+=x[i]** untuk menambahkan seluruh data yang telah dimasukkan oleh user (i adalah data-data tersebut). Kemudian kita return untuk mengembalikan nilai supaya hasil penjumlahan berhasil dilakukan dan ditampilkan di program.

Kita buat main method yang eksekusinya diawali dengan **public static void main (String args[])**. Public, dapat diartikan bahwa argumen atau metode main() merupakan sebuah metode yang bersifat publik. void merupakan suatu tipe data yang menyatakan bahwa deklarasi code tidak memerlukan nilai balik atau return. String menyatakan bahwa argumen yang terdapat pada code java tersebut merupakan String. Args merupakan argumen bertipe data string yang mengandung array, Args dapat di misalkan sebagai memory untuk menampung program. Di dalamnya kita buat variabel data dan kita beri nilai 10 (integer) supaya user nantinya bisa memasukkan 10 data. Untuk melakukan itu kita tulis baris kode **int data[] = new int[10];**. Selain itu kita juga buat variabel jumlah dengan **int jumlah;**.

Kemudian dilanjutkan dengan memasukkan fungsi scanner supaya kita nanti bisa membuat kode perintah supaya program bisa memasukkan data yang diinput oleh user yaitu dengan menuliskan baris kode **Scanner scan = new Scanner(System.in)**. Kita buat perulangan **for(int i=0;i<10;i++)** yang fungsinya untuk mengulang kalimat **System.out.print("Masukkan data ke-"+(i+1)+":")**. **i=0** berarti perulangan i dimulai dari 0. **i<10** berarti selama i bernilai kurang dari 10, maka perulangan akan terus berjalan. **i++** fungsinya adalah untuk menambah satu (+1) nilai hitungan pada setiap perulangan. User bisa memasukkan data untuk diproses program dengan baris kode **data[i] = scan.nextInt()**. Terakhir, kita panggil method hitungJumlah dengan **jumlah = hitungJumlah(data)** sekaligus kita tampilkan hasilnya yaitu dengan **System.out.println("Jumlah data = "+jumlah)**.

## PRAKTIK 4

```
ProyekHitungJumlah.java x
1  import java.util.Scanner;
2      public class ProyekHitungJumlah{
3          public static int hitungJumlah (int[] x){
4
5              int jum=0;
6              for(int i=0;i<x.length;i++){
7                  jum+=x[i];
8              }
9              return jum;
10         }
11         public static float hitungMean (int[]x){
12             float jum=0;
13             for(int i=0;i<x.length;i++){
14                 jum+=x[i];
15             }
16             return (jum/10);
17         }
18         public static void main (String args[]){
19             int data[] = new int[10];
20             int jumlah;
```



```

21         float mean;
22         Scanner scan = new Scanner(System.in);
23         for(int i=0;i<10;i++){
24             System.out.print("Masukkan data ke-"+(i+1)+" :");
25             data[i] = scan.nextInt();
26         }
27         jumlah = hitungJumlah(data);
28         mean = hitungMean(data);
29         System.out.println("Jumlah data = "+jumlah);
30         System.out.println("Rata-rata = "+mean);
31     }
32 }

```

output:

```

C:\Windows\system32\cmd.exe
Masukkan data ke-1:5
Masukkan data ke-2:10
Masukkan data ke-3:15
Masukkan data ke-4:20
Masukkan data ke-5:25
Masukkan data ke-6:30
Masukkan data ke-7:35
Masukkan data ke-8:40
Masukkan data ke-9:45
Masukkan data ke-10:50
Jumlah data = 275
Rata-rata = 27.5
Press any key to continue . . .

```

penjelasan:

program ini struktur dan konsepnya sama persis dengan program di praktik 3 tadi. Perbedaannya adalah di program ini kita menambahkan method untuk menghitung rata-rata dari seluruh data yang dimasukkan oleh user. Untuk membuat program di atas, kita awali dengan mengimport scanner yaitu dengan **import java.util.Scanner;**. Setelah itu kita deklarasikan nama class yaitu dengan menuliskan baris kode **public class ProyekHitungJumlah**.

Kita buat sebuah method untuk menghitung penjumlahan sekaligus membuat perulangan yaitu **public static int hitungJumlah(int[] x)**. Di dalamnya kita buat variabel **jum=0** yang artinya value jum dimulai dari 0. Lalu kita tambahkan perulangan for. Perulangan yang kita buat di dalam method hitungJumlah adalah **for(int i=0;i<x.length;i++)**. **i=0** berarti perulangan i dimulai dari 0. **i<x.length** berarti selama i bernilai kurang dari **x.length**, maka perulangan akan terus berjalan. **i++** fungsinya adalah untuk menambah satu (+1) nilai hitungan pada setiap perulangan. Lalu kita tulis **jum+=x[i]** untuk menambahkan seluruh data yang telah dimasukkan oleh user (i adalah data-data tersebut). Kemudian kita return untuk mengembalikan nilai supaya hasil penjumlahan berhasil dilakukan dan ditampilkan di program dengan **return jum**.

Method kedua yang kita buat adalah method yang berfungsi untuk menghitung nilai rata-rata seluruh data yang dimasukkan oleh user. Method tersebut adalah **public static float hitungMean(int[]x)**. Di dalamnya kita buat variabel jum bertipe data float supaya bisa menampilkan bilangan desimal. Untuk itu kita tulis **float jum=0**. Lalu buat perulangan **for(int i=0;i<x.length;i++)**. **i=0** berarti perulangan i dimulai dari 0. **i<x.length** berarti selama i bernilai kurang dari **x.length**, maka perulangan akan terus berjalan. **i++** fungsinya adalah untuk menambah satu (+1) nilai hitungan pada setiap perulangan. Lalu kita tulis **jum+=x[i]** untuk menambahkan seluruh data yang telah dimasukkan oleh user (i adalah data-data tersebut). Kemudian kita return untuk mengembalikan nilai supaya hasil penjumlahan sekaligus pembagian (supaya bisa menghasilkan nilai rata-rata) berhasil dilakukan dan ditampilkan di program dengan **return jum/10**.

Kita buat main method yang eksekusinya diawali dengan **public static void main (String args[])**. kita buat variabel data dan kita beri nilai 10 (integer) supaya user nantinya bisa memasukkan 10

data. Untuk melakukan itu kita tulis baris kode `int data[] = new int[10];`. Selain itu kita juga buat variabel jumlah dengan **int jumlah**; dan **float mean**;. Selanjutnya kita masukkan fungsi scanner supaya kita nanti bisa membuat kode perintah supaya program bisa memasukkan data yang diinput oleh user yaitu dengan menuliskan baris kode **Scanner scan = new Scanner(System.in)**

Kita buat perulangan **for(int i=0;i<10;i++)** yang fungsinya untuk mengulang kalimat **System.out.print("Masukkan data ke-"+(i+1)+":")**. **i=0** berarti perulangan i dimulai dari 0. **i<10** berarti selama i bernilai kurang dari 10, maka perulangan akan terus berjalan. **i++** fungsinya adalah untuk menambah satu (+1) nilai hitungan pada setiap perulangan. User bisa memasukkan data untuk diproses program dengan baris kode **data[i] = scan.nextInt()**. Terakhir, kita panggil method hitungJumlah dan hitung Mean dengan **jumlah = hitungJumlah(data)** dan juga **mean = hitungMean(data)**, sekaligus kita tampilkan hasilnya yaitu dengan **System.out.println("Jumlah data = "+jumlah)** dan **System.out.println("Rata-rata = "+mean)**.

## PRAKTIK 5

```
ProyekCekGenap1.java x
import java.util.Scanner;
public class ProyekCekGenap1{
    public static boolean cekGenap(int a){
        boolean status;
        if((a%2)==0){
            status = true;
        }else{
            status = false;
        }
        return status;
    }
    public static void main(String args[]){
        Scanner scan = new Scanner(System.in);
        int x;
        boolean genap;
        System.out.print("Masukkan bilangan : ");
        x = scan.nextInt();
        genap = cekGenap(x);
        if(genap){
            System.out.println("Bilangan yang dicek ternyata genap");
        }else{
            System.out.println("Bilangan yang dicek ternyata ganjil");
        }
    }
}
```

output:

```
C:\Windows\system32\cmd.exe
Masukkan bilangan : 7
Bilangan yang dicek ternyata ganjil
Press any key to continue . . .
```

```
C:\Windows\system32\cmd.exe
Masukkan bilangan : 12
Bilangan yang dicek ternyata genap
Press any key to continue . . .
```

penjelasan:

program ini adalah program yang berfungsi supaya bisa mengecek bilangan mana yang dimasukkan user yang berupa ganjil atau genap. Untuk membuat program ini kita awali dengan import Scanner



yaitu dengan menuliskan baris kode **import java.util.Scanner**. Lalu kita deklarasikan nama class yaitu dengan menuliskan baris kode **public class ProyekCekGenap1**. Class tersebut bermodifikasi public yaitu keyword yang berfungsi untuk menentukan sifat akses ke semua member yang mengikutinya (di bawahnya), sehingga memiliki sifat dapat di akses dari manapun, dan dapat di akses dari dalam class itu sendiri, dari anak class (derived class) dan juga dari luar class.

Kemudian kita buat method cekGenap dengan menuliskan **public static boolean cekGenap(int a)**. parameter int a adalah sebuah variabel yang nantinya akan bergungsi sebagai penampung data yang diinput oleh user. Kita lanjutkan dengan membuat variabel **boolean status**. Kita buat seleksi kondisi di dalam method tersebut. **if((a%2)==0)** yang artinya "jika variabel a modulo 2 (sisa setelah dibagi 2) hasilnya sama dengan 0, maka.." lalu dilanjutkan dengan **status = true**. Lalu ada else yang akan menghasilkan **status = false**. Kemudian kita return untuk mengembalikan nilai yaitu dengan **return status**.

Kita buat main method yang eksekusinya diawali dengan **public static void main(String args[])**. Public, dapat diartikan bahwa argumen atau metode main() merupakan sebuah metode yang bersifat publik. void merupakan suatu tipe data yang menyatakan bahwa deklarasi code tidak memerlukan nilai balik atau return. String menyatakan bahwa argumen yang terdapat pada code java tersebut merupakan String. Args merupakan argumen bertipe data string yang mengandung array, Args dapat di misalkan sebagai memory untuk menampung program. Kita juga kita buat fungsi scanner supaya kita nanti bisa membuat kode perintah supaya program bisa memasukkan data yang diinput oleh user yaitu dengan menuliskan baris kode **Scanner scan = new Scanner(System.in)**.

Langkah selanjutnya adalah membuat variabel **int x** dan **boolean genap**. Kemudian kita beri tempat bagi user untuk memasukkan bilangan yaitu dengan **System.out.print("Masukkan bilangan : ")**, yang input data nya akan diproses dengan baris kode **x = scan.nextInt()**. Kemudian kita tulis **genap = cekGenap(x)** untuk mengecek jika data yang dimasukkan adalah genap. Jika data yang dimasukkan memenuhi kondisi bilangan genap yaitu **if(genap)** maka program akan memunculkan pesan **System.out.println("Bilangan yang dicek ternyata genap")**. Jika else (ganjil), maka program akan memunculkan pesan **System.out.println("Bilangan yang dicek ternyata ganjil")**.

## PRAKTIK 6

```
ProyekGanjilGenapArray.java x
1  import java.util.Scanner;
2  public class ProyekGanjilGenapArray{
3
4      public static boolean[] cekGanjilGenapArray(int[]x){
5          boolean hasil[] = new boolean[10];
6          for (int i=0;i<x.length;i++){
7              if((x[i]%2)==0){
8                  hasil[i] = true;
9              }else{
10                 hasil[i] = false;
11             }
12         }
13         return hasil;
14     }
15
16     public static void main(String args[]){
17         int data [] = new int[10];
18         boolean hasilCek[] = new boolean[10];
19         Scanner scan = new Scanner(System.in);
20         for (int i=0;i<10;i++){
21             System.out.print("Masukkan data ke-"+(i+1)+":");
22             data[i] = scan.nextInt();
23         }
```

```

24     hasilCek = cekGanjilGenapArray(data);
25     System.out.println("=====");
26     System.out.println("==Hasil Pengecekan==");
27     System.out.println("=====");
28
29     for(int i=0;i<10;i++){
30         System.out.print(" "+data[i]);
31         System.out.print(" "+hasilCek[i]);
32         System.out.println();
33     }
34 }
35 }

```

output:

```

C:\Windows\system32\cmd.exe
Masukkan data ke-1:2
Masukkan data ke-2:5
Masukkan data ke-3:4
Masukkan data ke-4:6
Masukkan data ke-5:7
Masukkan data ke-6:3
Masukkan data ke-7:4
Masukkan data ke-8:7
Masukkan data ke-9:9
Masukkan data ke-10:8
=====
==Hasil Pengecekan==
=====
2 true
5 false
4 true
6 true
7 false
3 false
4 true
7 false
9 false
8 true
Press any key to continue . . .

```

penjelasan:

untuk membuat program di atas, kita awali dengan mengimport scanner yaitu dengan menuliskan **import java.util.Scanner**. Kemudian kita deklarasi nama class dengan menuliskan **public class ProyekGanjilGenapArray**. Dibawahnya kita buat method untuk mengecek ganjil dan genapnya bilangan yang yaitu method **public static boolean[]cekGanjilGenapArray(int[]x)**. Method tersebut bertipe data boolean karena hanya akan menghasilkan dua nilai yaitu true dan false. Lalu kita buat variabel bernama hasil yang kita beri nilai 10 yaitu **boolean hasil[] = new boolean[10]**.

Di dalamnya, kita beri perulangan **for (int i=0;i<x.length;i++)** yang berfungsi untuk menampilkan pesan true atau false sesuai dengan data yang nanti dimasukkan. **i=0** berarti perulangan i dimulai dari 0. **i<x.length** berarti selama i bernilai kurang dari **x.length**, maka perulangan akan terus berjalan. **i++** fungsinya adalah untuk menambah satu (+1) nilai hitungan pada setiap perulangan. Kita buat seleksi kondisi di dalam method tersebut. **if((x[i]%2)==0** yang artinya "jika variabel x[i] modulo 2 (sisa setelah dibagi 2) hasilnya sama dengan 0, maka.." lalu dilanjutkan dengan **hasil[i] = true**. Lalu ada else yang akan menghasilkan **hasil[i] = false**. Kemudian kita return untuk mengembalikan nilai yaitu dengan return hasil.

Lalu langsung saja kita buat main method yang diawali dengan baris kode **public static void main(String args[])**. Di dalamnya kita beri nilai 10 pada variabel data yaitu dengan **int data [] = new int[10]**. Kemudian kita juga berikan nilai 10 untuk data yang nanti telah dicek yaitu dengan baris kode **boolean hasilCek[] = new boolean[10]**. Buat fungsi scanner supaya kita nanti bisa membuat kode perintah supaya program bisa memasukkan data yang diinput oleh user yaitu dengan menuliskan baris kode **Scanner scan = new Scanner(System.in)**. Lalu buat perulangan for yaitu

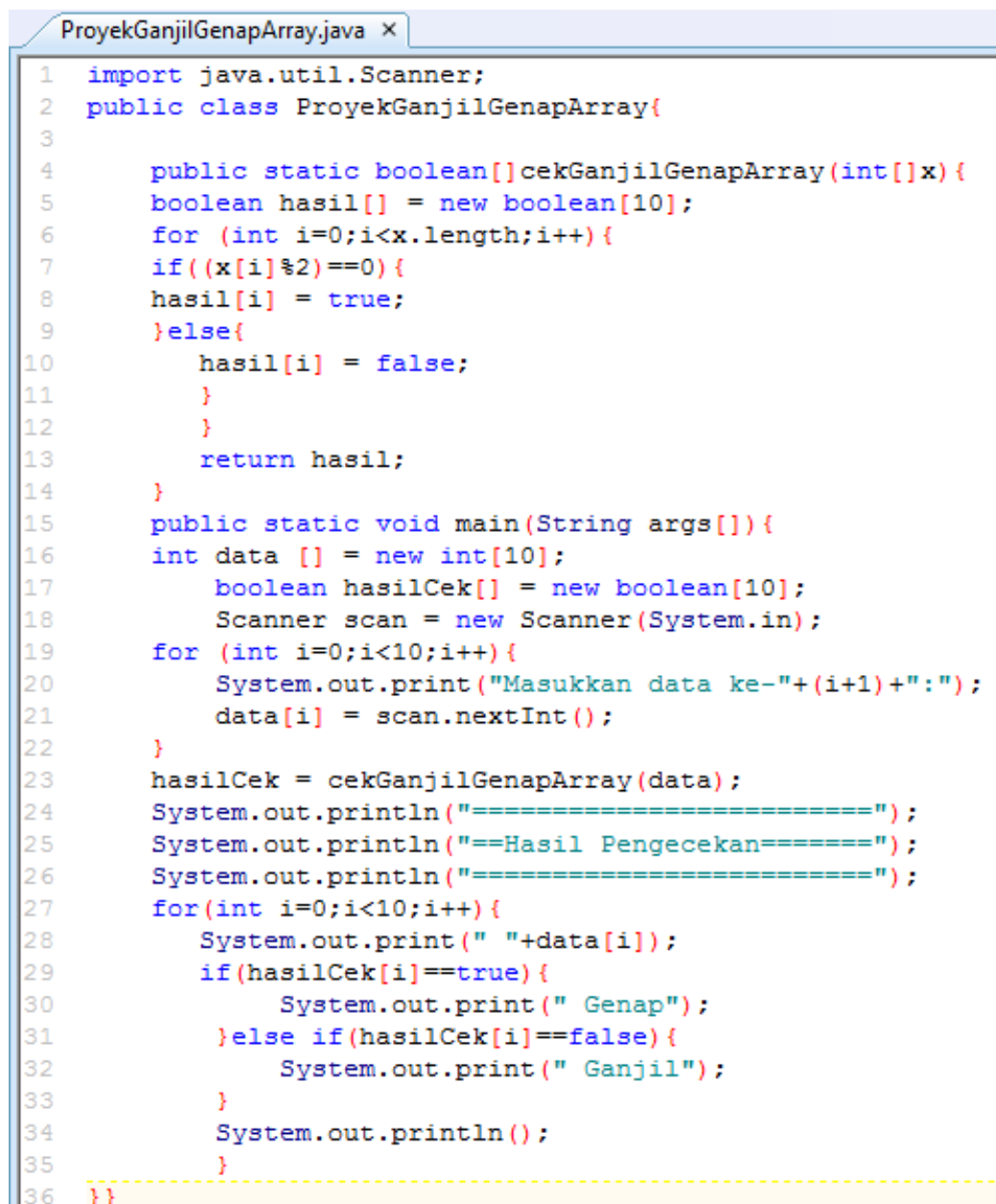
**for (int i=0;i<10;i++)**. perulangan tersebut digunakan untuk mengulang kalimat ""Masukkan data ke-"+(i+1)+":". **i=0** berarti perulangan i dimulai dari 0. **i<10** berarti selama i bernilai kurang dari 10, maka perulangan akan terus berjalan. **i++** fungsinya adalah untuk menambah satu (+1) nilai hitungan pada setiap perulangan. Kalimat yang diulang adalah **System.out.print("Masukkan data ke-"+(i+1)+" :")**. Lalu program akan memasukkan dan memproses data dengan baris kode **data[i] = scan.nextInt()**. Lalu kita panggil method dengan **hasilCek = cekGanjilGenapArray(data)**. Lalu kita buat kalimat-kalimat pembatas antara data yang belum dicek dan sudah dicek:

```
System.out.println("=====");
System.out.println("==Hasil Pengecekan=====");
System.out.println("=====");
```

kemudian kita buat perulangan **for(int i=0;i<10;i++)** lagi. Lalu program diakhiri dengan menampilkan data yang diinput dan data hasil pengecekan yaitu:

```
System.out.print(" "+data[i]);
System.out.print(" "+hasilCek[i]);
```

## PRAKTIK 7



```
ProyekGanjilGenapArray.java x
1  import java.util.Scanner;
2  public class ProyekGanjilGenapArray{
3
4      public static boolean[] cekGanjilGenapArray(int[] x){
5          boolean hasil[] = new boolean[10];
6          for (int i=0;i<x.length;i++){
7              if((x[i]%2)==0){
8                  hasil[i] = true;
9              }else{
10                 hasil[i] = false;
11             }
12         }
13         return hasil;
14     }
15     public static void main(String args[]){
16         int data [] = new int[10];
17         boolean hasilCek[] = new boolean[10];
18         Scanner scan = new Scanner(System.in);
19         for (int i=0;i<10;i++){
20             System.out.print("Masukkan data ke-"+(i+1)+" :");
21             data[i] = scan.nextInt();
22         }
23         hasilCek = cekGanjilGenapArray(data);
24         System.out.println("=====");
25         System.out.println("==Hasil Pengecekan=====");
26         System.out.println("=====");
27         for(int i=0;i<10;i++){
28             System.out.print(" "+data[i]);
29             if(hasilCek[i]==true){
30                 System.out.print(" Genap");
31             }else if(hasilCek[i]==false){
32                 System.out.print(" Ganjil");
33             }
34             System.out.println();
35         }
36     }}
```

output:

```
C:\Windows\system32\cmd.exe
Masukkan data ke-1:2
Masukkan data ke-2:5
Masukkan data ke-3:4
Masukkan data ke-4:6
Masukkan data ke-5:8
Masukkan data ke-6:7
Masukkan data ke-7:9
Masukkan data ke-8:1
Masukkan data ke-9:2
Masukkan data ke-10:3
=====
==Hasil Pengecekan=====
=====
2 Genap
5 Ganjil
4 Genap
6 Genap
8 Genap
7 Ganjil
9 Ganjil
1 Ganjil
2 Genap
3 Ganjil
Press any key to continue . . .
```

penjelasan:

program ini hampir sama dengan program sebelumnya, hanya saja true diganti dengan "Genap" dan false diganti dengan "Ganjil".

Kita awali dengan mengimport scanner yaitu dengan menuliskan **import java.util.Scanner**. Kemudian kita deklarasi nama class dengan menuliskan **public class ProyekGanjilGenapArray**. Dibawahnya kita buat method untuk mengecek ganjil dan genapnya bilangan yang yaitu method **public static boolean[] cekGanjilGenapArray(int[] x)**. Method tersebut bertipe data boolean karena hanya akan menghasilkan dua nilai yaitu true dan false. Lalu kita buat variabel bernama hasil yang kita beri nilai 10 yaitu **boolean hasil[] = new boolean[10]**.

Di dalamnya, kita beri perulangan **for (int i=0;i<x.length;i++)** yang berfungsi untuk menampilkan pesan true atau false sesuai dengan data yang nanti dimasukkan. **i=0** berarti perulangan i dimulai dari 0. **i<x.length** berarti selama i bernilai kurang dari **x.length**, maka perulangan akan terus berjalan. **i++** fungsinya adalah untuk menambah satu (+1) nilai hitungan pada setiap perulangan. Kita buat seleksi kondisi di dalam method tersebut. **if((x[i]%2)==0** yang artinya "jika variabel x[i] modulo 2 (sisanya setelah dibagi 2) hasilnya sama dengan 0, maka.." lalu dilanjutkan dengan **hasil[i] = true**. Lalu ada else yang akan menghasilkan **hasil[i] = false**. Kemudian kita return untuk mengembalikan nilai yaitu dengan return hasil.

Lalu langsung saja kita buat main method yang diawali dengan baris kode **public static void main(String args[])**. Di dalamnya kita beri nilai 10 pada variabel data yaitu dengan **int data [] = new int[10]**. Kemudian kita juga berikan nilai 10 untuk data yang nanti telah dicek yaitu dengan baris kode **boolean hasilCek[] = new boolean[10]**. Buat fungsi scanner supaya kita nanti bisa membuat kode perintah supaya program bisa memasukkan data yang diinput oleh user yaitu dengan menuliskan baris kode **Scanner scan = new Scanner(System.in)**. Lalu buat perulangan for yaitu **for (int i=0;i<10;i++)**. perulangan tersebut digunakan untuk mengulang kalimat ""Masukkan data ke-""+(i+1)+": ". **i=0** berarti perulangan i dimulai dari 0. **i<10** berarti selama i bernilai kurang dari 10, maka perulangan akan terus berjalan. **i++** fungsinya adalah untuk menambah satu (+1) nilai hitungan pada setiap perulangan. Kalimat yang diulang adalah **System.out.print("Masukkan data ke-""+(i+1)+": ")**. Lalu program akan memasukkan dan memproses data dengan baris kode **data[i]**

= `scan.nextInt()`. Lalu kita panggil method dengan `hasilCek = cekGanjilGenapArray(data)`. Lalu kita buat kalimat-kalimat pembatas antara data yang belum dicek dan sudah dicek:

```
System.out.println("=====");
```

```
System.out.println("==Hasil Pengecekan=====");
```

```
System.out.println("=====");
```

lalu buat perulangan `for(int i=0;i<10;i++)`. Jika kondisi memenuhi `if(hasilCek[i]==true)`, maka program menampilkan pesan `System.out.print(" Genap")`. Jika kondisi memenuhi seleksi `else if(hasilCek[i]==false)`, maka program akan menampilkan pesan `System.out.print(" Ganjil")`.

---

## LATIHAN 1

```
KonversiNilai.java x
1  import java.util.Scanner;
2  public class KonversiNilai{
3      public static void main(String args[]){
4          Scanner scan = new Scanner (System.in);
5          double[] data = new double [10];
6          char[] keluar = new char [10];
7          for (int i=0;i<10;i++){
8              System.out.print("data ke-"+(i+1)+" :");
9              data[i]=scan.nextDouble();
10         }
11         System.out.println("=====");
12         System.out.println(" Hasil Konversi ");
13         System.out.println("=====");
14         keluar = konversiNilai(data);
15         for(int i=0;i<10;i++){
16             System.out.println(data[i]+" "+keluar[i]);
17         }
18     }
19     public static char[] konversiNilai(double[] x){
20         int n = x.length;
21         char hasil[] = new char [n];
22         for (int i=0;i<n;i++){
23             if ((x[i]>80) && (x[i]<=100)){
24                 hasil[i]='A';
25             }else if ((x[i]>60) && (x[i]<=80)){
26                 hasil[i]='B';
27             }else if ((x[i]>40) && (x[i]<=60)){
28                 hasil[i]='C';
29             }else if ((x[i]>20) && (x[i]<=40)){
30                 hasil[i]='D';
31             }else{
32                 hasil[i]='E';
33             }
34         }
35         return hasil;
36     }}
```

output di halaman selanjutnya

output:

```
C:\Windows\system32\cmd.exe
data ke-1:60
data ke-2:90
data ke-3:45
data ke-4:20
data ke-5:36
data ke-6:72
data ke-7:87
data ke-8:95
data ke-9:56
data ke-10:48
=====
  Hasil Konversi
=====
60.0 C
90.0 A
45.0 C
20.0 E
36.0 D
72.0 B
87.0 A
95.0 A
56.0 C
48.0 C
Press any key to continue . . .
```

penjelasan:

kita membuat program untuk mengkonversi masukan berupa angka menjadi huruf. Untuk itu kita import scanner dengan **import java.util.Scanner**. Lalu kita tuliskan baris kode **public static void main(String args[])**. Buat fungsi scanner supaya kita nanti bisa membuat kode perintah supaya program bisa memasukkan data yang diinput oleh user yaitu dengan menuliskan baris kode **Scanner scan = new Scanner(System.in)**. Kemudian kita beri nilai ke dalam variabel data dan variabel keluar yaitu dengan **double[] data = new double [10]** dan **char[] keluar = new char [10]**. Data adalah variabel tempat menampung data nilai berupa angka yang dimasukkan user, sementara keluar adalah hasil konversi dari angka menjadi char.

Kita buat perulangan **for (int i=0;i<10;i++)**. **i=0** berarti perulangan i dimulai dari 0. **i<10** berarti selama i bernilai kurang dari 10, maka perulangan akan terus berjalan. **i++** fungsinya adalah untuk menambah satu (+1) nilai hitungan pada setiap perulangan. Perulangan tersebut berfungsi untuk mengulang kalimat tempat user memasukkan data yaitu **System.out.print("data ke-" + (i+1) + ":")**. Lalu data yang telah diinput diproses dengan **data[i]=scan.nextDouble()**. Kemudian kita buat batas untuk hasil konversi yaitu:

```
System.out.println("=====");
System.out.println(" Hasil Konversi ");
System.out.println("=====");
```

dibawahnya kita tulis **keluar = konversiNilai(data)** untuk mengonversi nilai menjadi char. Kita beri perulangan lagi yaitu **for (int i=0;i<10;i++)**. **i=0** berarti perulangan i dimulai dari 0. **i<10** berarti selama i bernilai kurang dari 10, maka perulangan akan terus berjalan. **i++** fungsinya adalah untuk menambah satu (+1) nilai hitungan pada setiap perulangan. Perulangan itu akan menampilkan kalimat **System.out.println(data[i]+ " "+keluar[i])**.

kita buat method untuk mengubah angka double menjadi char yaitu method **public static char[] konversiNilai(double[] x)**. kita deklarasikan dua variabel yaitu **int n = x.length** dan **char hasil[] = new char [n]**. Kita beri perulangan untuk mengulang char yaitu **for (int i=0;i<n;i++)**. **i=0** berarti perulangan i dimulai dari 0. **i<n** berarti selama i bernilai kurang dari n, maka perulangan akan terus berjalan. **i++** fungsinya adalah untuk menambah satu (+1) nilai hitungan pada setiap perulangan. Kita buat seleksi kondisi dengan if dan else if. Jika nilai 81-100 akan maka akan dikonversi menjadi



'A'. Jika nilai 61-80 akan maka akan dikonversi menjadi 'B'. Jika nilai 41-60 akan maka akan dikonversi menjadi 'C'. Jika nilai 21-40 akan maka akan dikonversi menjadi 'D'. Jika nilai 20 dan dibawah 20 (else) maka akan dikonversi menjadi 'E'. terakhir, kita **return hasil**.

---

## TUGAS

```
KonversiNilai.java x
1  import java.util.Scanner;
2  public class KonversiNilai{
3      public static void main(String[]args){
4          Scanner scan = new Scanner (System.in);
5          int nilai;
6          char huruf;
7          double[] data = new double [10];
8          char[] keluar = new char [10];
9          int i = 0;
10         while (i<10){
11             System.out.print("data ke-"+(i+1)+":");
12             data[i]=scan.nextDouble();
13             i++;
14         }
15         System.out.println("=====");
16         System.out.println(" Hasil Konversi ");
17         System.out.println("=====");
18         keluar = konversiNilai(data);
19
20         int j = 0;
21         while (j<10){
22             System.out.println(data[j] + " " + keluar [j]);
23             j++;
24         }
25     }
26     public static char[] konversiNilai(double[] x){
27         int n = x.length;
28         char hasil[] = new char [n];
29         int i = 0;
30         while(i<n){
31             switch ((int) (x[i]/10)){
32                 case 10:
33                 case 9:
34                 case 8:
35                     hasil[i] = 'A';
36                     break;
37                 case 7:
38                 case 6:
39                     hasil[i] = 'B';
40                     break;
41                 case 5:
42                 case 4:
43                     hasil[i] = 'C';
44                     break;
45                 case 3:
46                 case 2:
47                     hasil[i] = 'D';
48                     break;
49                 case 1:
50                 case 0:
51                     hasil[i] = 'E';
52                     break;
53                 default:
54                     hasil[i] = 'z';
55             }
56             i++;
57         }
58         return hasil;
59     }}
```

output:

```
C:\Windows\system32\cmd.exe
data ke-1:95
data ke-2:23
data ke-3:18
data ke-4:58
data ke-5:75
data ke-6:80
data ke-7:82
data ke-8:42
data ke-9:37
data ke-10:68
=====
  Hasil Konversi
=====
95.0 A
23.0 D
18.0 E
58.0 C
75.0 B
80.0 A
82.0 A
42.0 C
37.0 D
68.0 B
Press any key to continue . . . _
```

penjelasan:

kita import scanner dengan **import java.util.Scanner**. Lalu kita tuliskan baris kode **public static void main(String args[])**. Buat fungsi scanner supaya kita nanti bisa membuat kode perintah supaya program bisa memasukkan data yang diinput oleh user yaitu dengan menuliskan baris kode **Scanner scan = new Scanner(System.in)**. Kita buat variabel **int nilai** dan **char huruf**. Kemudian kita beri nilai ke dalam variabel data dan variabel keluar yaitu dengan **double[] data = new double [10]** dan **char[] keluar = new char [10]**. Data adalah variabel tempat menampung data nilai berupa angka yang dimasukkan user, sementara keluar adalah hasil konversi dari angka menjadi char.

Kita buat perulangan while yang diawali dengan **int i = 0** yang artinya perulangan i dimulai dari 0. **while (i<10)** artinya selama i kurang dari 10, maka perulangan akan dilakukan. Lalu kita buat kalimat untuk tempat user memasukkan data yaitu **System.out.print("data ke-"+(i+1)+":")**. Lalu data yang telah dimasukkan langsung diproses oleh program lewat **data[i]=scan.nextDouble()**. Setelah itu kita tulis **i++** untuk menambah satu (+1) nilai hitungan pada setiap perulangan. Kemudian kita beri kalimat pembatas hasil konversi:

```
System.out.println("=====");
```

```
System.out.println(" Hasil Konversi ");
```

```
System.out.println("=====");
```

kita tulis **keluar = konversiNilai(data)** untuk mengonversi nilai menjadi char. Kita buat perulangan while yang diawali dengan **int j = 0** yang artinya perulangan j dimulai dari 0. **while (j<10)** artinya selama j kurang dari 10, maka perulangan akan dilakukan. Kemudian untuk menampilkan hasil konversi kita tulis **System.out.println(data[j] + " " + keluar [j])**. Setelah itu kita tulis **j++** untuk menambah satu (+1) nilai hitungan pada setiap perulangan.

Kita buat method untuk mengonversi nilai double menjadi char dengan **public static char[] konversiNilai(double[] x)**. Kita tulis **char hasil[] = new char [n]** sebagai tempat char yang sudah diproses. Kemudian kita buat perulangan while dan switch case. Pertama kita deklarasikan **int i = 0**; lalu **while(i<n)**. kemudian kita buat **switch ((int) (x[i]/10))** yang artinya nilai yang dimasukkan user akan dibagi dengan 10. Misalnya kita memasukkan 90, maka program akan membagi 90 dengan 10 sehingga hasilnya 9 dan masuk ke case 9 dan mendapat nilai 'A'. jika nilai 61-80 maka masuk case 6-7 (B). jika nilai 41-60 maka masuk case 4-5(C). jika nilai 21-40 maka masuk case 2-3(D). jika 20 kebawah maka masuk case 0-1(E). Jika salah input maka 'z'. Terakhir tulis baris kode **return hasil**.

---

---

## **KESIMPULAN**

di pertemuan ke-10 ini kita membahas tentang iterasi seleksi dalam method. Saya pun telah menyelesaikan seluruh praktik, latihan, dan tugas di pertemuan ini sehingga saya telah memenuhi tujuan dari pertemuan praktikum ke-10 ini yaitu dapat menyelesaikan kasus dengan menggabungkan konsep iterasi, seleksi dalam method serta dapat memanggil method dari class lain. Kasus yang ada di dalam pertemuan ini bermacam-macam dan berfokus ke iterasi dan seleksi yang di dalam setiap programnya selalu ada perulangan (menggunakan for atau while) dan di beberapa program kita menggunakan seleksi else if juga. Dari semua yang telah saya kerjakan, bisa saya simpulkan jika iterasi adalah sifat tertentu dari algoritma atau program komputer di mana suatu urutan dari langkah algoritmik dilaksanakan di loop program. Setelah itu di setiap program iterasi yang dibuat dipadukan dengan seleksi di dalam method yang dibuat.

**Terima Kasih**