

Nama : muhammad shabran

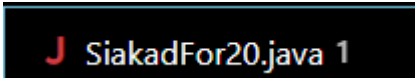
Kelas : 1 D TI

No absen : 20

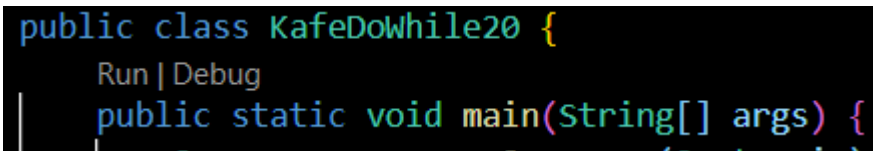
NIM : 244107020112

Percobaan 1 :

1. Buat file baru, beri nama SiakadForNoAbsen.java

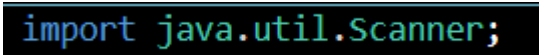


2. Buatlah struktur dasar program Java yang terdiri dari fungsi main().



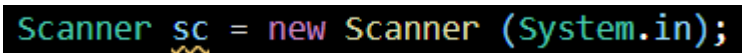
```
public class KafeDoWhile20 {  
    Run | Debug  
    public static void main(String[] args) {
```

3. Tambahkan library Scanner di bagian atas (luar) class



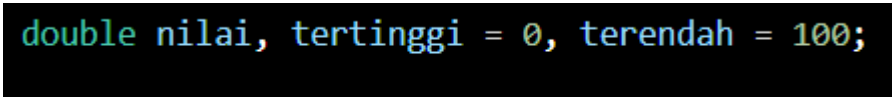
```
import java.util.Scanner;
```

4. Buat deklarasi Scanner dengan nama variabel sc di dalam fungsi main()



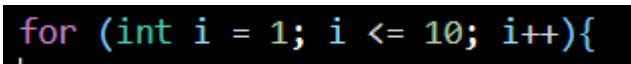
```
Scanner sc = new Scanner (System.in);
```

5. Deklarasikan variabel nilai, tertinggi, dan terendah bertipe double. Inisialisasi tertinggi dengan 0 dan terendah dengan 100



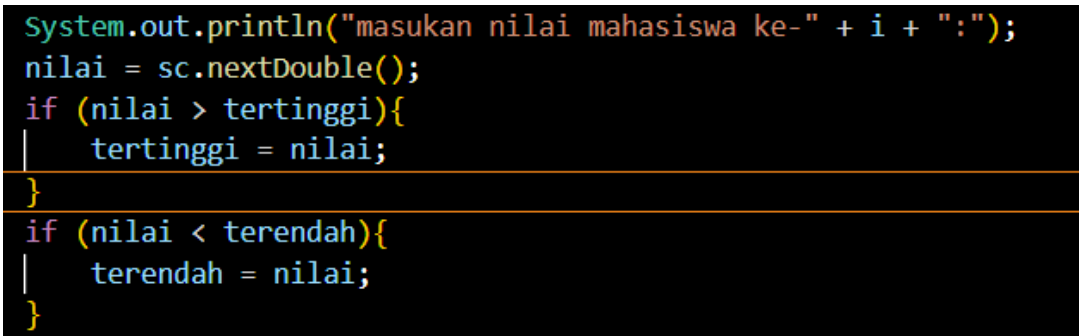
```
double nilai, tertinggi = 0, terendah = 100;
```

6. Buat struktur perulangan FOR dengan batas kondisi sesuai jumlah mahasiswa yaitu 10



```
for (int i = 1; i <= 10; i++){
```

7. Di dalam perulangan FOR tersebut, tambahkan perintah untuk memasukkan nilai mahasiswa. Setelah itu, buat dua kondisi pemilihan secara terpisah untuk mengecek nilai tertinggi dan terendah dengan membandingkan nilai masukan dengan variabel tertinggi dan variabel terendah



```
System.out.println("masukan nilai mahasiswa ke-" + i + ":");  
nilai = sc.nextDouble();  
if (nilai > tertinggi){  
    tertinggi = nilai;  
}  
if (nilai < terendah){  
    terendah = nilai;  
}
```

8. Di luar perulangan FOR, tampilkan nilai tertinggi dan terendah

```

System.out.println("masukan nilai mahasiswa ke-" + i + ":");
nilai = sc.nextDouble();
if (nilai > tertinggi){
    tertinggi = nilai;
}
if (nilai < terendah){
    terendah = nilai;
}

```

9. Di luar perulangan FOR, tampilkan nilai tertinggi dan terendah

```

System.out.println("nilai tertinggi :" + tertinggi);
System.out.println("nilai terendah :" + terendah);

```

10. Cocokkan hasil compile kode program Anda dengan gambar berikut ini.

```

Masukkan nilai mahasiswa ke-1: 76.5
Masukkan nilai mahasiswa ke-2: 82.3
Masukkan nilai mahasiswa ke-3: 62.1
Masukkan nilai mahasiswa ke-4: 88.4
Masukkan nilai mahasiswa ke-5: 65.9
Masukkan nilai mahasiswa ke-6: 67.9
Masukkan nilai mahasiswa ke-7: 90.1
Masukkan nilai mahasiswa ke-8: 55.3
Masukkan nilai mahasiswa ke-9: 73.7
Masukkan nilai mahasiswa ke-10: 78.6
Nilai tertinggi: 90.1
Nilai terendah: 55.3

```

```

masukan nilai mahasiswa ke-1:
76,5
masukan nilai mahasiswa ke-2:
82,3
masukan nilai mahasiswa ke-3:
52,1
masukan nilai mahasiswa ke-4:
88,4
masukan nilai mahasiswa ke-5:
65,9
masukan nilai mahasiswa ke-6:
67,9
masukan nilai mahasiswa ke-7:
90,1
masukan nilai mahasiswa ke-8:
55,3
masukan nilai mahasiswa ke-9:
73,7
masukan nilai mahasiswa ke-10:
78,6
nilai tertinggi :90.1
nilai terendah :52.1
PS C:\Users\muham\Desktop\ngoding\pekan 7\jobsheet-7> 

```

### Pertanyaan

1. Sebutkan dan tunjukkan masing-masing komponen perulangan FOR pada kode program Percobaan 1!

Jawab :

```
for (int i = 1; i <= 10; i++){
```

2. Mengapa variabel tertinggi diinisialisasi 0 dan terendah diinisialisasi 100? Apa yang terjadi jika variabel tertinggi diinisialisasi 100 dan terendah diinisialisasi 0?

Jawab : agar nilai terbesar yang di input akan menjadi variabel tertinggi sedangkan nilai terendah akan masuk ke variabel terendah. Jika dibalik maka input yang dimasukan harus lebih besar dari 100 atau lebih kecil dari 0

3. Jelaskan fungsi dan alur kerja dari potongan kode berikut!

```
if (nilai > tertinggi) {  
    tertinggi = nilai;  
}  
if (nilai < terendah) {  
    terendah = nilai;  
}
```

Jawab : program akan membaca input yang paling tinggi dan di masukan ke variabel tertinggi, lalu akan membaca input yang paling rendah akan di masukan ke variabel terendah.

4. Modifikasi kode program sehingga terdapat perhitungan untuk menentukan berapa mahasiswa yang lulus dan yang tidak lulus berdasarkan batas kelulusan (nilai minimal 60). Tampilkan jumlah mahasiswa lulus dan tidak lulus setelah menampilkan nilai tertinggi dan terendah!
5. Commit dan push kode program ke Github

Jawaban nomor 4&5 telah di push ke github

### Percobaan 2

1. Buat file baru, beri nama SiakadWhileNoAbsen.java

```
J Siakadwhile20.java 1
```

2. Buatlah struktur dasar program Java yang terdiri dari fungsi main().

```
public class Siakadwhile20 {  
    Run | Debug  
    public static void main(String[] args) {
```

3. Tambahkan library Scanner di bagian atas (luar) class

```
import java.util.Scanner;
```

4. Buat deklarasi Scanner dengan nama variabel sc di dalam fungsi main()

```
Scanner sc = new Scanner(System.in);
```

5. Deklarasikan variabel nilai, jml, dan i (untuk perulangan) bertipe integer. Inisialisasi i dengan 0 sebagai nilai awal perulangan

```
int nilai, jml, i=0;
```

6. Tuliskan kode program untuk menerima input banyaknya mahasiswa yang disimpan ke variabel jml. Dengan demikian, batas perulangan akan dinamis sesuai masukan dari pengguna melalui keyboard.

```
System.out.print(s:"masukan jumlah mahasiswa : ");  
jml = sc.nextInt();
```

7. Buat struktur perulangan WHILE dengan batas kondisi sesuai jumlah mahasiswa yaitu 5. Perhatikan simbol yang digunakan adalah < karena perulangan variabel i dimulai dari 0, bukan 1

```
while ( i < jml) {
```

8. Di dalam perulangan WHILE tersebut, tambahkan perintah untuk memasukkan nilai mahasiswa. Setelah itu, buat kondisi pemilihan IF untuk mengecek valid atau tidaknya nilai yang dimasukkan, dengan syarat nilai harus berada pada rentang 0 hingga 100. Kemudian tambahkan kondisi pemilihan IF-ELSE IF-ELSE untuk menampilkan kategori nilai huruf berdasarkan ketentuan.

```

System.out.print("masukan nilai mahasiswa ke " + (i+1) + " :");
nilai = sc.nextInt();

if (nilai < 0 || nilai > 100){
    System.out.println( x:"nilai tidak valid masukan nilai yang valid ");
    continue;
}
if (nilai > 80 && nilai <= 100){
    System.out.println("nilai mahasiswa" + (i+1) + "adalah a");
}
else if (nilai > 73 && nilai <= 80){
    System.out.println("nilai mahasiswa" + (i+1) + "adalah b+");
}
else if (nilai > 65 && nilai <= 73){
    System.out.println("nilai mahasiswa" + (i+1) + "adalah b");
}
else if (nilai > 60 && nilai <= 65){
    System.out.println("nilai mahasiswa" + (i+1) + "adalah c+");
}
else if (nilai > 50 && nilai <= 60){
    System.out.println("nilai mahasiswa" + (i+1) + "adalah c");
}
else if (nilai > 39 && nilai <=50){
    System.out.println("nilai mahasiswa" + (i+1) + "adalah d");
}

else {
    System.out.println("nilai mahasiswa" + (i+1) + "adalah e");
}

```

9. Cocokkan hasil compile kode program Anda dengan gambar berikut ini.

```

Masukkan nilai mahasiswa ke-1: 85
Nilai mahasiswa ke-1 adalah A
Masukkan nilai mahasiswa ke-2: 63
Nilai mahasiswa ke-2 adalah C+
Masukkan nilai mahasiswa ke-3: 101
Nilai tidak valid. Masukkan lagi nilai yang valid!
Masukkan nilai mahasiswa ke-3: 23
Nilai mahasiswa ke-3 adalah E
Masukkan nilai mahasiswa ke-4: -15
Nilai tidak valid. Masukkan lagi nilai yang valid!
Masukkan nilai mahasiswa ke-4: 70
Nilai mahasiswa ke-4 adalah B
Masukkan nilai mahasiswa ke-5: 55
Nilai mahasiswa ke-5 adalah C

```

```

masukan jumlah mahasiswa : 5
masukan nilai mahasiswa ke 1 :85
nilai mahasiswa1adalah a
masukan nilai mahasiswa ke 2 :63
nilai mahasiswa2adalah c+
masukan nilai mahasiswa ke 3 :101
nilai tidak valid masukan nilai yang valid
masukan nilai mahasiswa ke 3 :23
nilai mahasiswa3adalah e
masukan nilai mahasiswa ke 4 :-15
nilai tidak valid masukan nilai yang valid
masukan nilai mahasiswa ke 4 :70
nilai mahasiswa4adalah b
masukan nilai mahasiswa ke 5 :55
nilai mahasiswa5adalah c

```

Pertanyaan

1. Pada potongan kode berikut, tentukan maksud dan kegunaan dari sintaks berikut:

```

if (nilai < 0 || nilai > 100) {
    System.out.println(x:"Nilai tidak valid. Masukkan lagi nilai yang valid!");
    continue;
}

```

a. `nilai < 0 || nilai > 100`

Jawab : untuk memprogram nilai yang lebih dari 100 atau kurang dari 0 ,jika salah satu nya benar maka statement akan di jalankan

b. `continue`

Jawab : `continue` berfungsi untuk mengabaikan statement jika nilai tidak lebih dari seratus,dan tidak kurang dari nol

2. Mengapa sintaks `i++` dituliskan di akhir perulangan WHILE? Apa yang terjadi jika posisinya dituliskan di awal perulangan WHILE?  
Jawab : agar perulangan dilakukan setelah semua statement dalam while dijalankan, jika di tulis di awal maka perulangan akan dilakukan sebelum statement di jalankan.
3. Apabila jumlah mahasiswa yang dimasukkan adalah 19, berapa kali perulangan WHILE akan berjalan?  
Jawab : perulangan while berjalan 19 kali
4. Modifikasi kode program sehingga apabila terdapat mahasiswa yang mendapat nilai A, program menampilkan pesan tambahan "Bagus, pertahankan nilainya"!
5. Commit dan push kode program ke Github  
Jawaban pertanyaan nomor 4&5 telah di push ke github

Percobaan 3

1. Buat file baru, beri nama `KafeDoWhileNoAbsen.java`

## KafeDoWhile20.java 1

2. Buatlah struktur dasar program Java yang terdiri dari fungsi main().

```
public class KafeDoWhile20 {  
    Run | Debug  
    public static void main(String[] args) {
```

3. Tambahkan library Scanner di bagian atas (luar) class

```
import java.util.Scanner;
```

4. Buat deklarasi Scanner dengan nama variabel sc di dalam fungsi main()

```
Scanner sc = new Scanner (System.in);
```

5. Deklarasikan variabel kopi, teh, dan roti bertipe integer untuk menampung banyaknya item yang dibeli pelanggan, serta namaPelanggan bertipe String. Deklarasi dan inisialisasi hargaKopi dengan 12000, hargaTeh dengan 7000, hargaRoti dengan 20000.

```
double hkopi = 12000, hteh = 7000, hroti = 20000, jkopi, jteh, jroti, totalharga;  
String namapelanggan;
```

6. Buat struktur perulangan DO-WHILE dengan kondisi true

```
do {  
    } while (true);
```

7. Di dalam perulangan DO-WHILE tersebut, tambahkan perintah untuk memasukkan namaPelanggan. Kemudian tambahkan kondisi IF untuk mengecek isi variabel namaPelanggan. Selanjutnya, tambahkan perintah untuk memasukkan banyaknya item yang dibeli pelanggan untuk setiap menu, apabila masukan nama pelanggan bukan "batal". Hitung total harga pembelian dan tampilkan hasilnya.

```

System.out.print(s:" masukan nama pelanggan (ketik batal untuk keluar) :");
namapelanggan = sc.nextLine();

if (namapelanggan.equalsIgnoreCase (anotherString:"batal")){
    System.out.println(x:"pesanan dibatalkan");
    break;
}
System.out.print(s:"jumlah kopi : ");
jkopi = sc.nextInt();
System.out.print(s:"jumlah teh : ");
jteh = sc.nextInt();
System.out.print(s:"jumlah roti : ");
jroti = sc.nextInt();
totalharga = (jkopi*hkopi) + (jteh*hteh) + (jroti*hroti) ;
System.out.println( "total harga yang harus di bayar : " + totalharga);
sc.nextLine();
while (true);
System.out.println(x:"semua transaksi selesai");

```

8. Cocokkan hasil compile kode program Anda dengan gambar berikut ini.

```

Masukkan nama pelanggan (ketik 'batal' untuk keluar): Rena
Jumlah kopi: 3
Jumlah teh: 0
Jumlah roti: 1
Total yang harus dibayar: Rp 56000
Masukkan nama pelanggan (ketik 'batal' untuk keluar): Yuni
Jumlah kopi: 1
Jumlah teh: 4
Jumlah roti: 2
Total yang harus dibayar: Rp 80000
Masukkan nama pelanggan (ketik 'batal' untuk keluar): BATAL
Transaksi dibatalkan.
Semua transaksi selesai.

```

```

masukan nama pelanggan (ketik batal untuk keluar) :rena
jumlah kopi : 3
jumlah teh : 0
jumlah roti : 1
total harga yang harus di bayar : 56000.0
masukan nama pelanggan (ketik batal untuk keluar) :yuni
jumlah kopi : 1
jumlah teh : 4
jumlah roti : 2
total harga yang harus di bayar : 80000.0
masukan nama pelanggan (ketik batal untuk keluar) :BATAL
pesanan dibatalkan
semua transaksi selesai
PS C:\Users\muham\Desktop\ngoding\pekan 7\jobsheet-7>

```

Pertanyaan :



1. Pada penggunaan DO-WHILE ini, apabila nama pelanggan yang dimasukkan pertama kali adalah “batal”, maka berapa kali perulangan dilakukan?

Jawab : karena do while akan melakukan perulangan setidaknya 1 kali bagaimanapun kondisinya

2. Sebutkan kondisi berhenti yang digunakan pada perulangan DO-WHILE tersebut!

Jawab : kondisi berhenti adalah ketika input yang dimasukan adalah “batal”

3. Apa fungsi dari penggunaan nilai true pada kondisi DO-WHILE?

Jawab : agar kode dalam blok do akan di eksekusi berulang kali sampai ada kondisi yang mengharuskan loop itu berhenti.

4. Mengapa perulangan DO-WHILE tersebut tetap berjalan meskipun tidak ada komponen inisialisasi dan update?

Jawab : karena do while akan melakukan perulangan setidaknya 1 kali bagaimanapun kondisinya

#### Tugas :

1. Seorang pengelola bioskop ingin membuat program untuk menghitung total penjualan tiket dalam satu hari. Tiket dijual dengan harga Rp 50.000 per tiket. Program harus menghitung total tiket yang terjual dan total harga penjualan tiket selama satu hari dengan ketentuan sebagai berikut:
  - Jika pelanggan membeli lebih dari 4 tiket, pelanggan mendapatkan diskon 10%.
  - Jika pelanggan membeli lebih dari 10 tiket, pelanggan mendapatkan diskon 15%.
  - Jika input jumlah tiket tidak valid (negatif), program akan mengabaikan input tersebut dan meminta input ulang.wss
2. Sebuah tempat parkir ingin membuat program untuk menghitung total pembayaran parkir dari beberapa kendaraan. Tarif parkir adalah Rp 3.000 per jam untuk mobil dan Rp 2.000 per jam untuk motor. Namun, jika durasi parkir lebih dari 5 jam, diberikan tarif tetap sebesar Rp 12.500 untuk semua kendaraan. Program akan terus meminta masukan selama input bukan 0. Implementasikan flowchart tersebut ke dalam bentuk kode program Java!

Tugas 1 & 2 sudah di push ke github

