

KAMIS 30 OKTOBER 2025

LAPORAN HASIL PRAKTIKUM
Jobsheet 8



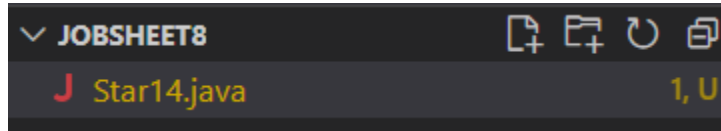
Disusun oleh:
Moch Dedy Triagwi
Kelas 1H/IT
254107020233

PROGRAM STUDI D-IV TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2025

1. Praktikum

1.1.1 Percobaan 1: Review Perulangan 1

1. Percobaan ini ditujukan untuk me-review kembali perulangan yang telah dibahas pada pertemuan sebelumnya. Pada percobaan 1 akan dibuat program untuk membuat tampilan * sebanyak n kali ke arah **samping**.
2. Buat file baru **StarNoAbsen.java**



3. Buat fungsi/method **main()** di dalamnya.
4. Karena program membutuhkan input dari keyboard, maka perlu import class Scanner. Jadi tambahkan sintaks import di baris atas sendiri program.
5. Di dalam fungsi **main()** yang telah dibuat, deklarasikan objek **Scanner** dengan nama **sc**.
6. Pada baris selanjutnya, tampilkan instruksi untuk memasukkan nilai yang akan disimpan ke variabel **n**.

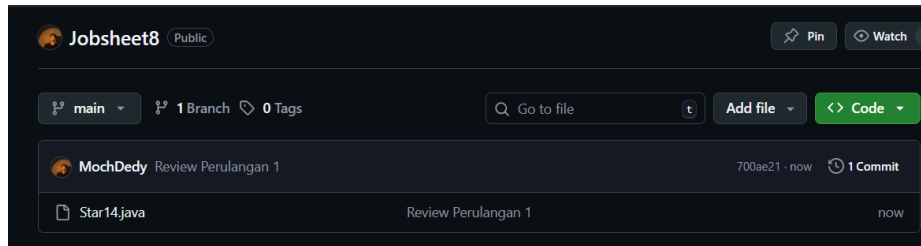
```
System.out.print(s:"Masukkan nilai n: ");  
n = sc.nextInt();
```

7. Pada baris selanjutnya, buat sintaks perulangan dengan for seperti di bawah ini.

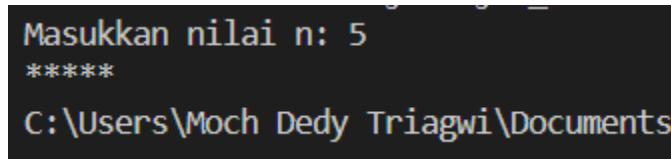
```
for (i = 1; i <= n; i++){  
    System.out.print(s:"*");  
}
```

Catatan: perlu diperhatikan, bahwa yang digunakan adalah perintah **print**, bukan **println** karena kita ingin menampilkan tanpa ada baris baru

8. Compile dan jalan program.



9. Amati hasilnya, maka hasilnya harusnya akan serupa dengan tampilan di bawah ini.



1.1.2 Pertanyaan

1. Jika pada perulangan for, inisialisasi $i=1$ diubah menjadi $i=0$, apa akibatnya? Mengapa bisa demikian?

Jawab:

Output akan bertambah satu bintang. Karena perulangannya jadi dari 0 sampai n (total $n+1$ kali). Terjadi karena kondisi $i \leq n$ masih benar saat $i=0$ sehingga loop berjalan satu iterasi lebih banyak.

2. Jika pada perulangan for, kondisi $i \leq n$ diubah menjadi $i < n$, bagaimana bentuk outputnya jika input $n = 5$? Mengapa hasilnya berbeda?

Jawab:

Jika input $n = 5$ maka output yang dikeluarkan adalah 4 bintang karena loop berjalan dari $i=1$ hingga $i=4$ sehingga output yang dihasilkan adalah 4 kali

3. Jika pada perulangan for, kondisi $i \leq n$ diubah menjadi $i > n$, apa akibatnya? Mengapa bisa demikian?

Jawab:

Loop tidak akan berjalan karena inisialisasi variabel i adalah 1 sehingga otomatis keluar dari loop

4. Jika pada perulangan for, kondisi step $i++$ diubah menjadi $i--$ apa akibatnya? Mengapa bisa demikian?

Jawab:

Maka loop tidak akan berhenti (infinite loop) karena tidak sesuai dengan kondisi berhenti yaitu $i \leq n$

5. Jika pada perulangan for, step $i++$ diubah menjadi $i += 2$, bagaimana pola outputnya jika input $n = 6$? Apa yang menyebabkan perubahan tersebut?

Jawab:

$i += 2$ akan melangkahi 2 angka setelahnya jika input 5 maka, yang dibaca adalah 1,3,5 sehingga akan memunculkan 3 output

1.2.1 Percobaan 2: Bintang Persegi

1. Pada percobaan ke-2 akan dilakukan percobaan tentang **nested loop**. Kasus yang akan diselesaikan adalah untuk membuat tampilan bujursangkar *, dengan panjang sisi sebanyak n . Misalkan n dimasukan 5, maka hasilnya adalah:
2. Kalau diamati lebih lanjut, sebenarnya mirip dengan kasus percobaan 1. Jika di percobaan 1, misal input n bernilai 5, maka yang akan dihasilkan adalah ***** (bisa dianggap ini sebagai **inner loop** yang mencetak 5 bintang *****), maka untuk kasus percobaan 2 ini, hasil dari percobaan 1 tersebut hanya perlu diulang lagi sebanyak n kali. Yaitu dengan menambahkan **outer loop** untuk mengulangi proses **inner loop** sebanyak n kali.
3. Buat file baru **SquareNoAbsen.java**

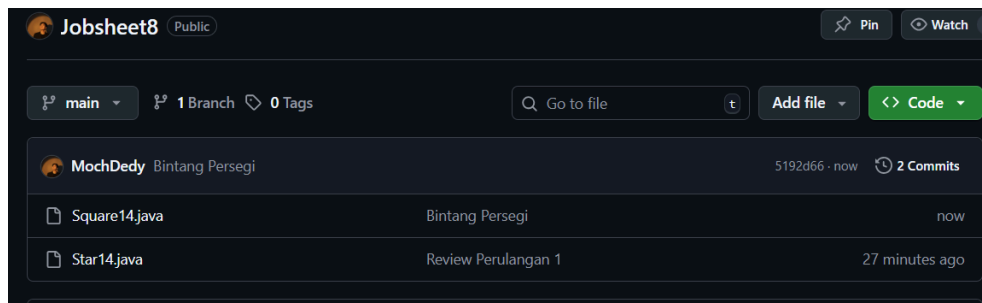


4. Karena program membutuhkan input dari keyboard, maka perlu import class Scanner. Jadi tambahkan sintaks import di baris atas sendiri program.
5. Buat method **main()**, dan isikan kode program yang sama dengan isi method **main()** di percobaan 1.

6. Compile dan jalankan program. Pastikan program berjalan seperti saat menjalankan percobaan 1.
7. Perhatikan sintaks perulangan yang digunakan untuk mencetak * sebanyak n kali ke arah samping. Di langkah 8 di atas kode **for** (kotak merah) dijadikan sebagai *inner loop*
8. Perulangan pada inner loop dilakukan sebanyak n kali untuk menghasilkan output seperti pada poin 1. Maka perlu ditambahkan perulangan luar (outer loop).

```
for (int iOuter = 1; iOuter <= n; iOuter++) {  
    for (i = 1; i <= n; i++) {  
        System.out.print(s: "*");  
    }  
    System.out.println();  
}
```

9. Simpan perubahan, compile dan jalankan program.



10. Amati hasilnya, maka hasilnya harusnya akan serupa dengan tampilan di bawah ini.

```
Masukkan nilai n: 5  
*****  
*****  
*****  
*****  
*****  
  
C:\Users\Moch Dedy Triagwi\Documents\Kuliah\DEDY NITIP\PRAKTEKDASPRO\Jobsheet8
```

1.2.2 Pertanyaan

1. Perhatikan perulangan luar. Jika pada sintaks for, inisialisasi `iOuter=1` diubah menjadi `iOuter=0`, apa yang akibatnya? Mengapa bisa demikian?

Jawab:

Output yang dihasilkan menjadi 6 baris bintang ($n+1$ baris). Karena loop terjadi dari 0 sampai n .

2. Kembalikan program semula dimana inisialisasi `iOuter=1`. Kemudian perhatikan perulangan dalam. Jika pada sintaks for, inisialisasi `i=1` diubah menjadi `i=0`, apa yang akibatnya? Mengapa bisa demikian?

Jawab:

Output yang dihasilkan setiap baris adalah 6 bintang karena inisialisasi `i` adalah 0 ($n+1$ bintang) karena inner loop terjadi dari 0 hingga n .

3. Apakah perbedaan kegunaan antara perulangan luar dengan perulangan yang berada di dalamnya?

Jawab:

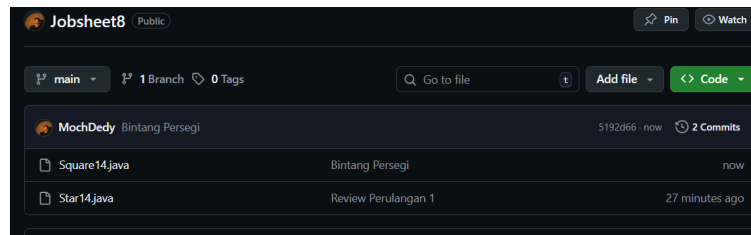
- Outer loop untuk mengontrol jumlah baris yang diinginkan
- Inner loop untuk mengontrol jumlah bintang dalam satu baris

4. Mengapa perlu ditambahkan sintaks `System.out.println()`; di bawah perulangan dalam? Apa akibatnya jika sintaks tersebut dihilangkan?

Jawab:

Untuk memindahkan ke baris baru setelah mencetak satu baris bintang. Jika dihapus maka semua bintang muncul dalam satu baris panjang.

5. Silakan commit dan push ke repository Anda.



1.3.1 Percobaan 3: Bintang Segitiga

1. Pada percobaan ke-3 akan dilakukan percobaan segitiga *, dengan sama siku dengan tinggi sebesar n. Misalkan n dimasukan 5, maka hasilnya adalah:
2. Buat file baru **TriangleNoAbsen.java**
3. Karena program membutuhkan input dari keyboard, maka perlu import class Scanner
4. Buat method **main()**, dan isikan kode program berikut kedalam method **main()**.

```
while (i <= n) {  
    j = 0;  
    while (j < i) {  
        System.out.print(s:"*");  
        j++;  
    }  
    i++;  
}
```

5. Compile dan jalankan program. Amati apa yang terjadi.

Hasil Program

```
Masukkan nilai n: 5  
*****  
c:\Users\Moch Dedy Triagwi\Documents\
```

1.3.2 Pertanyaan

1. Perhatikan, apakah output yang dihasilkan dengan nilai $n = 5$ sesuai dengan tampilan berikut?

```
  *
 * *
* * *
* * * *
* * * * *
```

Jawab:

Karena tidak ada perintah **println()** yang berguna untuk memindahkan baris

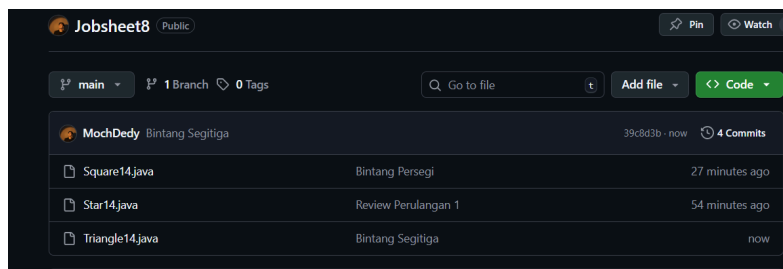
2. Jika tidak sesuai, bagian mana saja yang harus diperbaiki/ditambahkan? Jelaskan setiap bagian yang perlu diperbaiki/ditambahkan.

Jawab:

```
while (i <= n) {
    j = 0;
    while (j < i) {
        System.out.print(s:"*");
        j++;
    }
    System.out.println();
    i++;
}
```

Kita perlu menambahkan **println()** karena setiap mencetak kita perlu memindahkan ke baris yang baru sehingga akan mencetak seperti segitiga, jika kita tidak menggunakan **println()** maka output hanya satu baris saja

3. Silakan commit dan push ke repository Anda.



4. Jelaskan peran masing-masing variabel *i* dan *j* dalam program ini. Mengapa *j* di-set ulang ke 0 di awal setiap iterasi outer loop? Apa yang akan terjadi jika *j* tidak di-reset?

Jawab:

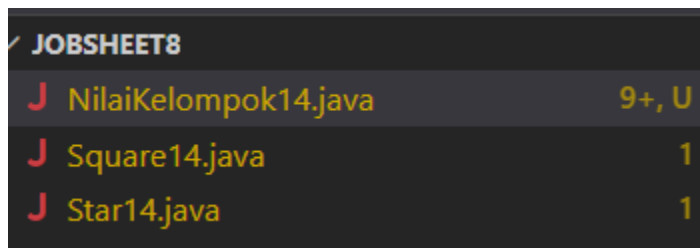
- a. *i* untuk mengontrol seberapa banyak baris
- b. *j* untuk mengontrol seberapa banyak bintang setiap baris

j harus di-reset ke 1 setiap iterasi outer loop agar inner loop dimulai ulang dari awal. Jika tidak di-reset maka *j* akan terus meningkat dan inner loop tidak berjalan lagi.

1.4.1 Percobaan 4: Studi Kasus Nilai Tugas Proyek Kelompok

Dalam pembelajaran kolaboratif, dosen menerapkan sistem peer review untuk menilai tugas kelompok. Setiap kelompok akan dinilai oleh beberapa kelompok lain sebagai bentuk umpan balik sejawat. Untuk setiap kelompok akan dinilai oleh beberapa kelompok lainnya. Pada percobaan ini, program akan memproses nilai untuk kelas yang berjumlah 6 kelompok. Masing-masing kelompok akan dinilai oleh 5 kelompok yang lain, dalam skala 1-100. Program akan menghitung nilai akhir setiap kelompok sebagai rata-rata dari semua penilai. Berikut adalah flowchart yang menunjukkan algoritma dari penyelesaian masalah tersebut.

1. Buat file baru **NilaiKelompokNoAbsen.java**



- 2. Buatlah struktur dasar program Java yang terdiri dari fungsi **main()**.
- 3. Tambahkan library Scanner di bagian atas (luar) class
- 4. Buat deklarasi Scanner dengan nama variabel *sc* di dalam fungsi **main()**
- 5. Deklarasikan variabel sesuai pada flowchart

6. Buat struktur perulangan untuk **inner loop** menggunakan **for** untuk memasukkan 5 nilai dari masing-masing kelompok penilai

```
for (j = 1; j <= 5; j++){  
      
}
```

7. Di dalam perulangan **for** tersebut, tambahkan perintah untuk memasukkan 5 nilai dari kelompok penilai dan menghitung total nilainya. Jangan lupa untuk memberikan nilai awal 0 pada totalNilai sebelum perulangan.

```
totalNilai = 0;  
for (j = 1; j <= 5; j++){  
    System.out.print("Nilai dari kelompok Penilai " + j + ": ");  
    nilai = sc.nextInt();  
    totalNilai += nilai;  
}
```

8. Setelah proses perulangan memasukkan 5 nilai dan diperoleh total nilai, maka hitung rata-rata nilai dengan rumus

```
totalNilai = 0;  
for (j = 1; j <= 5; j++){  
    System.out.print("Nilai dari kelompok Penilai " + j + ": ");  
    nilai = sc.nextInt();  
    totalNilai += nilai;  
}  
rataNilai = totalNilai / 5;
```

9. Selanjutnya adalah mengulang proses memasukkan nilai tersebut untuk 6 kelompok. Buat struktur perulangan untuk **outer loop** menggunakan **while**.

```
i = 1;
while (i <= 6) {
    System.out.println("Kelompok " + i);

    totalNilai = 0;
    for (j = 1; j <= 5; j++) {
        System.out.print("Nilai dari kelompok Penilai " + j + ": ");
        nilai = sc.nextInt();
        totalNilai += nilai;
    }
    rataNilai = totalNilai / 5;
    i++;
}
```

10. Tambahkan narasi keterangan untuk kebutuhan masukan dan keluaran
11. Compile dan run program.

Kode Program

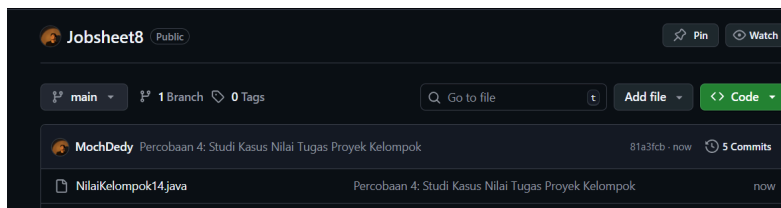
```
J NilaiKelompok14.java > ...
1  import java.util.Scanner;
2
3  public class NilaiKelompok14 {
4      Run | Debug | Run main | Debug main
5      public static void main(String[] args) {
6          Scanner sc = new Scanner(System.in);
7
8          int i, j, nilai;
9          float totalNilai, rataNilai;
10
11          System.out.println(x:"==== Program Penilaian Tugas Kelompok ===");
12          System.out.println(x:"Terdapat 6 kelompok dalam 1 kelas");
13          System.out.println(x:"Setiap kelompok akan dinilai oleh 5 kelompok lainnya!");
14          System.out.println(x:"Silahkan masukkan nilai dengan rentan 1 - 100.\n");
15
16          i = 1;
17          while (i <= 6) {
18              System.out.println("Kelompok " + i);
19
20              totalNilai = 0;
21              for (j = 1; j <= 5; j++) {
22                  System.out.print("Nilai dari kelompok Penilai " + j + ": ");
23                  nilai = sc.nextInt();
24                  totalNilai += nilai;
25              }
26              rataNilai = totalNilai / 5;
27              System.out.println("Rata-rata nilai kelompok " + i + " = " + rataNilai);
28              i++;
29          }
30      }
```

12. Amati hasilnya, apakah program telah menjalankan perintah memasukkan 5 nilai untuk 6 kelompok dengan tampilan serupa dengan di bawah ini:

```
Kelompok 1
Nilai dari kelompok Penilai 1: 88
Nilai dari kelompok Penilai 2: 79
Nilai dari kelompok Penilai 3: 90
Nilai dari kelompok Penilai 4: 82
Nilai dari kelompok Penilai 5: 85
Rata-rata nilai Kelompok1= 84.8
Kelompok 2
Nilai dari kelompok Penilai 1: 89
Nilai dari kelompok Penilai 2: 85
Nilai dari kelompok Penilai 3: 90
Nilai dari kelompok Penilai 4: 85
Nilai dari kelompok Penilai 5: 82
Rata-rata nilai Kelompok2= 86.2
Kelompok 3
Nilai dari kelompok Penilai 1: 90
Nilai dari kelompok Penilai 2: 91
Nilai dari kelompok Penilai 3: 86
Nilai dari kelompok Penilai 4: 84
Nilai dari kelompok Penilai 5: 90
Rata-rata nilai Kelompok3= 88.2
Kelompok 4
Nilai dari kelompok Penilai 1: 77
Nilai dari kelompok Penilai 2: 75
Nilai dari kelompok Penilai 3: 80
Nilai dari kelompok Penilai 4: 79
Nilai dari kelompok Penilai 5: 76
Rata-rata nilai Kelompok4= 77.4
Kelompok 5
Nilai dari kelompok Penilai 1: 80
Nilai dari kelompok Penilai 2: 82
Nilai dari kelompok Penilai 3: 81
Nilai dari kelompok Penilai 4: 77
Nilai dari kelompok Penilai 5: 83
Rata-rata nilai Kelompok5= 80.6
Kelompok 6
Nilai dari kelompok Penilai 1: 91
Nilai dari kelompok Penilai 2: 90
Nilai dari kelompok Penilai 3: 85
Nilai dari kelompok Penilai 4: 88
Nilai dari kelompok Penilai 5: 90
Rata-rata nilai Kelompok6= 88.8

c:\Users\Moch Dedy Triagwi\Documents\Kuliah\DEDY N
```

13. Commit dan push kode program ke Github



1.4.2 Pertanyaan

1. Jelaskan apa yang terjadi pada variabel totalNilai di setiap iterasi outer loop dan mengapa inisialisasinya (total = 0) berada di dalam outer loop, bukan di luar.

Jawab:

Variabel totalNilai digunakan untuk menampung jumlah nilai dari setiap input, variabel tersebut diinisialisasi di dalam outer loop karena harus di set ke-0 dari setiap outer loop, jika berada di luar maka nilai input dari kelompok sebelumnya akan ikut terhitung sehingga akan salah

2. Modifikasi program di atas, sehingga dapat mencari kelompok dengan rata-rata nilai tertinggi dan tampilkan nomor kelompok tersebut.

Jawab:

Kode Program

```
J NilaiKelompok14.java > Java > NilaiKelompok14 > main(String[] args)
1  import java.util.Scanner;
2
3  public class NilaiKelompok14 {
4      Run | Debug | Run main | Debug main
5      public static void main(String[] args) {
6          Scanner sc = new Scanner(System.in);
7
8          int i, j, nilai, kelompokTertinggi = 0;
9          float totalNilai, rataNilai, maxRata = 0;
10
11          System.out.println(x:"==== Program Penilaian Tugas Kelompok ====");
12          System.out.println(x:"Terdapat 6 kelompok dalam 1 kelas");
13          System.out.println(x:"Setiap kelompok akan dinilai oleh 5 kelompok lainnya!");
14          System.out.println(x:"Silahkan masukkan nilai dengan rentan 1 - 100.\n");
15
16          i = 1;
17          while (i <= 6) {
18              System.out.println("Kelompok " + i);
19
20              totalNilai = 0;
21              for (j = 1; j <= 5; j++) {
22                  System.out.print("Nilai dari kelompok Penilai " + j + ": ");
23                  nilai = sc.nextInt();
24                  totalNilai += nilai;
25              }
26              rataNilai = totalNilai / 5;
27              System.out.println("Rata-rata nilai kelompok" + i + " = " + rataNilai);
28
29              if (rataNilai > maxRata) {
30                  maxRata = rataNilai;
31                  kelompokTertinggi = i;
32              }
33              i++;
34          }
35
36          System.out.println("\nKelompok dengan rata-rata tertinggi adalah Kelompok: " + kelompokTertinggi);
37          System.out.println("Dengan rata-rata nilai: " + maxRata);
38      }
```

Hasil Program

```
Silahkan masukkan nilai dengan Rentan 1 - 100.

Kelompok 1
Nilai dari kelompok Penilai 1: 90
Nilai dari kelompok Penilai 2: 80
Nilai dari kelompok Penilai 3: 80
Nilai dari kelompok Penilai 4: 60
Nilai dari kelompok Penilai 5: 53
Rata-rata nilai Kelompok1= 72.6
Kelompok 2
Nilai dari kelompok Penilai 1: 90
Nilai dari kelompok Penilai 2: 58
Nilai dari kelompok Penilai 3: 77
Nilai dari kelompok Penilai 4: 90
Nilai dari kelompok Penilai 5: 100
Rata-rata nilai Kelompok2= 83.0
Kelompok 3
Nilai dari kelompok Penilai 1: 48
Nilai dari kelompok Penilai 2: 55
Nilai dari kelompok Penilai 3: 79
Nilai dari kelompok Penilai 4: 90
Nilai dari kelompok Penilai 5:
100
Rata-rata nilai Kelompok3= 74.4
Kelompok 4
Nilai dari kelompok Penilai 1: 90
Nilai dari kelompok Penilai 2: 88
Nilai dari kelompok Penilai 3: 38
Nilai dari kelompok Penilai 4: 51
Nilai dari kelompok Penilai 5: 99
Rata-rata nilai Kelompok4= 73.2
Kelompok 5
Nilai dari kelompok Penilai 1: 100
Nilai dari kelompok Penilai 2: 100
Nilai dari kelompok Penilai 3: 100
Nilai dari kelompok Penilai 4: 100
Nilai dari kelompok Penilai 5: 100
Rata-rata nilai Kelompok5= 100.0
Kelompok 6
Nilai dari kelompok Penilai 1: 87
Nilai dari kelompok Penilai 2: 88
Nilai dari kelompok Penilai 3: 90
Nilai dari kelompok Penilai 4: 33
Nilai dari kelompok Penilai 5: 45
Rata-rata nilai Kelompok6= 68.6

Kelompok dengan rata-rata tertinggi adalah Kelompok: 5
Dengan rata-rata nilai: 100.0
```

Tugas

1. Buatlah program untuk menghitung dan menampilkan jumlah kuadrat bilangan 1 s.d n. Gunakan perulangan bersarang. Berikut output yang diharapkan jika n pada rentang 1 s.d 5.

```
n = 1 → jumlah kuadrat = 1
n = 2 → jumlah kuadrat = 1 + 4 = 5
n = 3 → jumlah kuadrat = 1 + 4 + 9 = 14
n = 4 → jumlah kuadrat = 1 + 4 + 9 + 16 = 30
n = 5 → jumlah kuadrat = 1 + 4 + 9 + 16 + 25 = 55
```

Jawab:

Kode Program

```
Tugas1Jobsheet8.java > Java > Tugas1Jobsheet8 > main(String[] args)
1  import java.util.Scanner;
2
3  public class Tugas1Jobsheet8 {
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6
7          System.out.print(s:"Masukkan nilai n: ");
8          int n = sc.nextInt();
9
10         int total = 0;
11
12         for(int i = 1; i <= n; i++){
13             int kuadrat = i * i;
14             System.out.println(i + "^2 = " + kuadrat);
15             total += kuadrat;
16         }
17
18         System.out.println("Jumlah = " + total);
19     }
20 }
```

Hasil Program

```
Masukkan nilai n: 5
1^2 = 1
2^2 = 4
3^2 = 9
4^2 = 16
5^2 = 25
Jumlah = 55
```

2. Buatlah program untuk mencetak tampilan persegi angka seperti di bawah ini berdasarkan input n (nilai n minimal 3). Contoh n = 3, dan n = 5

```

      5 5 5 5 5
      5      5
      5      5
      5      5
      5 5 5 5 5

3 3 3
3 3
3 3 3

      5 5 5 5 5
      5      5
      5      5
      5      5
      5 5 5 5 5
```

Jawab:

Kode Program

```
J Tugas2Jobsheet8.java > Language Support for Java(TM) by Red Hat > Tugas2Jobsheet8
1  import java.util.Scanner;
2
3  public class Tugas2Jobsheet8{
    Run | Debug | Run main | Debug main
4  public static void main(String[] args) {
5      Scanner sc = new Scanner(System.in);
6
7      System.out.print(s:"Masukkan nilai n (minimal 3): ");
8      int n = sc.nextInt();
9
10     if (n < 3) {
11         System.out.println(x:"Input minimal 3!");
12         return;
13     }
14
15     for (int i = 1; i <= n; i++) {
16         for (int j = 1; j <= n; j++) {
17
18             if (i == 1 || i == n || j == 1 || j == n) {
19                 System.out.print(n + " ");
20             } else {
21                 System.out.print(s:" ");
22             }
23         }
24         System.out.println();
25     }
26 }
27 }
```

Hasil Program

```
Masukkan nilai n (minimal 3): 3
3 3 3
3 3
3 3 3
C:\Users\Moch Dedy Triagwi\Document

Masukkan nilai n (minimal 3): 5
5 5 5 5 5
5      5
5      5
5      5
5 5 5 5 5
c:\Users\Moch Dedy Triagwi\Documents\Kul
```


3. Sebuah jaringan kafe “Kopi Senja” memiliki beberapa cabang di berbagai lokasi kota. Untuk memantau kinerja operasional harian, manajemen membutuhkan data sederhana mengenai aktivitas penjualan di setiap cabang. Setiap hari, setiap cabang melayani sejumlah pelanggan, dan setiap pelanggan memesan satu atau lebih item, baik berupa makanan maupun minuman. Buatlah program untuk mencatat dan menghitung banyak pelanggan dan total item yang terjual pada masing-masing cabang. Karena sistem yang digunakan masih sederhana dan tidak menyimpan data historis, program hanya perlu mencatat secara langsung berapa banyak pelanggan yang dilayani dan berapa total item yang terjual di masing-masing cabang, tanpa perlu menyimpan nama pelanggan, jenis menu, atau detail lainnya. Program akan meminta input jumlah cabang terlebih dahulu. Lalu untuk setiap cabang, operator memasukkan jumlah pelanggan yang datang hari ini. Selanjutnya, untuk setiap pelanggan, operator memasukkan jumlah item yang dipesan. Program kemudian menghitung dan menampilkan ringkasan penjualan per cabang, serta total keseluruhan dari seluruh cabang pada akhir sesi. Berikut adalah contoh input dan output program:

Jawab:

Kode Program

```
Tugas3Jobsheet8.java > Java > Tugas3Jobsheet8 > main(String[] args)
1  import java.util.Scanner;
2
3  public class Tugas3Jobsheet8 {
    Run | Debug | Run main | Debug main
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6
7          int i, j, cabang, totSemuPel, totSemItem, pelanggan, totItemCabang, item;
8
9          System.out.print(s:"Masukkan jumlah cabang Kafe: ");
10         cabang = sc.nextInt();
11
12         totSemuPel = 0;
13         totSemItem = 0;
14
15         System.out.println(x:"\n==== Input Penjualan Setiap Cabang ====");
16
17         for (i = 1; i <= cabang; i++) {
18             System.out.println("\n--- Cabang " + i + " ---");
19             System.out.print(s:"Masukkan jumlah pelanggan: ");
20             pelanggan = sc.nextInt();
21             totSemuPel += pelanggan;
22
23             totItemCabang = 0;
24
25             for (j = 1; j <= pelanggan; j++) {
26                 System.out.print("- Pelanggan " + j + " memesan berapa item: ");
27                 item = sc.nextInt();
28                 totItemCabang += item;
29             }
30             System.out.println("Cabang " + i + ": ");
31             System.out.println("- Pelanggan: " + pelanggan + " orang");
32             System.out.println("- Item terjual: " + totItemCabang);
33             totSemItem += totItemCabang;
34         }
35
36         System.out.println("\nTotal seluruh cabang: " + cabang + " Cabang");
37         System.out.println("Pelanggan " + totSemuPel + " orang");
38         System.out.println("Item terjual: " + totSemItem + " item");
39     }
40 }
```

Hasil Program

```
==== Input Penjualan Setiap Cabang ====

--- Cabang 1 ---
Masukkan jumlah pelanggan: 3
- Pelanggan 1 memesan berapa item: 2
- Pelanggan 2 memesan berapa item: 4
- Pelanggan 3 memesan berapa item: 1
Cabang 1:
- Pelanggan: 3 orang
- Item terjual: 7

--- Cabang 2 ---
Masukkan jumlah pelanggan: 4
- Pelanggan 1 memesan berapa item: 3
- Pelanggan 2 memesan berapa item: 5
- Pelanggan 3 memesan berapa item: 1
- Pelanggan 4 memesan berapa item: 2
Cabang 2:
- Pelanggan: 4 orang
- Item terjual: 11

Total seluruh cabang: 2 Cabang
Pelanggan 7 orang
Item terjual: 18 item

c:\Users\Moch Dedy Triagwi\Documents\Kuliah\DEDY NITTI
```