



JOB SHEET 9

Perulangan 2

1. Tujuan

- Mahasiswa mampu memahami konsep perulangan bersarang
- Mahasiswa mampu menggambarkan flowchart perulangan bersarang
- Mahasiswa mampu membuat program perulangan bersarang dalam bahasa Java

Langkah Praktikum

Percobaan 1

1. Percobaan ini ditujukan untuk me-review kembali perulangan yang sudah dibahas pada pertemuan sebelumnya. Pada percobaan 1 akan dibuat program untuk membuat tampilan * sebanyak N kali ke arah samping.
2. Buat class baru dengan nama Star dan simpan dalam file **Star.java**

```
public class Star{  
}
```

3. Tambahkan main method di dalam class **Star**

```
public static void main(String[] args){  
}
```

4. Karena program membutuhkan input dari keyboard, maka diperlukan import class Scanner. Jadi tambahkan sintaks import di baris atas sendiri program.

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

5. Di dalam fungsi main() yang telah dibuat, deklarasikan objek Scanner dengan nama input.

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

6. Pada baris selanjutnya, tampilkan instruksi untuk memasukan nilai yang akan disimpan ke variabel N.

```
System.out.print("Masukkan Nilai N : ");  
int n = input.nextInt();
```



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

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

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

8. Compile dan jalankan program!
9. Amati hasilnya, maka hasilnya harusnya akan serupa dengan tampilan di bawah ini.

```
Masukkan Nilai N : 5
*****
```

Pertanyaan

1. Jika pada perulangan for, inisialisasi $i = 1$ diubah menjadi $i = 0$, apa akibatnya? Mengapa bisa demikian?
2. Jika pada perulangan for, kondisi $i \leq N$ diubah menjadi $i > N$, apa akibatnya? Mengapa bisa demikian?
3. Jika pada perulangan for, kondisi step $i++$ diubah menjadi $i--$ apa akibatnya? Mengapa bisa demikian?

Percobaan 2

1. Pada percobaan ke-2 akan dilakukan percobaan tentang loop bersarang. Kasus yang akan diselesaikan adalah untuk membuat tampilan bujursangkar *, dengan panjang sisi sebanyak N. Misalkan N dimasukan 5, maka hasilnya adalah:

```
Masukkan nilai n : 5
*****
*****
*****
*****
*****
```

2. Kalau diamati lebih lanjut, sebenarnya mirip dengan kasus percobaan 1 bukan? Jika di percobaan 1, misal input N bernilai 5, maka yang akan dihasilkan adalah *, maka untuk kasus percobaan 2 ini bukankah hasil dari percobaan 1 tersebut hanya perlu diulang lagi sebanyak N kali ?



3. Buatlah class Square dan simpan dengan nama file **Square.java**

```
public class Square{
}
```

4. Karena program membutuhkan input dari keyboard, maka perlu import class Scanner. Jadi tambahkan sintaks import di baris atas sendiri program.

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

5. Buat method main(), dan isikan kode program yang sama dengan isi method main() di percobaan 1.

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

System.out.print("Masukkan Nilai N : ");
int n = input.nextInt();

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

6. Perhatikan sintaks perulangan yang digunakan untuk mencetak * sebanyak N kali ke arah samping. Perulangan tersebut, harus diulang sebanyak N kali. Jadi perlu ditambahkan perulangan luar (outer loop).
7. Buatlah perulangan outer loop terlebih dahulu

```
for(int iOuter = 1; iOuter <= n; iOuter++){
    // taruh perulangan ke -n disini
}
```

Kemudian masukkan perulangan ke-n sebelumnya di dalam outer loop. Sehingga program akan menjadi seperti ini :

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

8. Compile dan jalankan
9. Setelah berhasil anda jalankan , maka hasilnya akan tampil seperti dibawah ini :



```
Masukkan nilai n : 5
```

```
*****
```

10. Untuk membuat hasil outputnya menjadi seperti dibawah ini,

```
Masukkan nilai n : 5
```

```
*****
```

maka tambahkan

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

Sehingga hasilnya akan menjadi seperti dibawah ini :

```
Masukkan nilai n : 5
```

```
*****
```

```
*****
```

```
*****
```

```
*****
```

```
*****
```

Pertanyaan

1. Perhatikan perulangan luar. Jika pada sintaks **for**, inialisasi **iOuter=1** diubah menjadi **iOuter=0**, apa yang akibatnya? Mengapa bisa demikian?
2. Kembalikan program semula dimana inialisasi **iOuter=1**. Kemudian perhatikan perulangan dalam, Jika pada sintaks **for**, inialisasi **i=1** diubah menjadi **i=0**, apa yang akibatnya? Mengapa bisa demikian?
3. Jadi, apakah perbedaan kegunaan antara perulangan luar dengan perulangan yang berada didalamnya?
4. Mengapa perlu ditambahkan sintaks **System.out.println();** di bawah perulangan dalam? Apa akibatnya jika sintaks tersebut dihilangkan?
5. Modifikasi program diatas sehingga menghasilkan output seperti dibawah ini :

```
*
```

```
* *
```

```
* * *
```

```
* * * *
```

```
* * * * *
```

**Percobaan 3**

1. Buat class baru dengan nama Quiz dan simpan dengan nama file Quiz.java

```
public class Quiz {  
  
}
```

2. Import class Random dan Scanner, di baris awal program.

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

3. Buatlah fungsi main()
4. Di dalam fungsi main() deklarasikan objek dari class Random dan Scanner. Class Random, pada kasus ini digunakan untuk mengacak angka.

```
Random random = new Random();  
Scanner input = new Scanner(System.in);
```

5. Kemudian pada baris selanjutnya, tambahkan sintaks seperti di bawah ini.



```
char menu='y';  
do{  
    int number = random.nextInt(10) + 1;  
    boolean success = false;  
    do {  
        System.out.print("Tebak angka (1-10): ");  
        int answer = input.nextInt();  
        input.nextLine();  
        success = (answer == number);  
    } while(!success);  
    System.out.print("Apakah Anda ingin mengulang permainan (Y/y)?");  
    menu = input.nextLine().charAt(0);  
} while(menu=='y' || menu=='Y');
```

Pertanyaan

1. Jelaskan alur program diatas.
2. Apa yang harus dilakukan untuk tidak melanjutkan (tidak mengulangi) permainan tersebut?



3. Modifikasi program di atas, sehingga bisa menampilkan informasi mengenai : input nilai tebakan yang dimasukan oleh user apakah lebih kecil atau lebih besar dari jawaban/*number* yang di random!

Tugas

1. 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
3 3 3  5      5
3  3  5      5
3 3 3  5 5 5 5
```

2. Buatlah program untuk mencetak tampilan segitiga angka seperti di bawah ini berdasarkan input N (nilai N minimal 3). Contoh N = 5

```
  1
 12
123
1234
12345
```