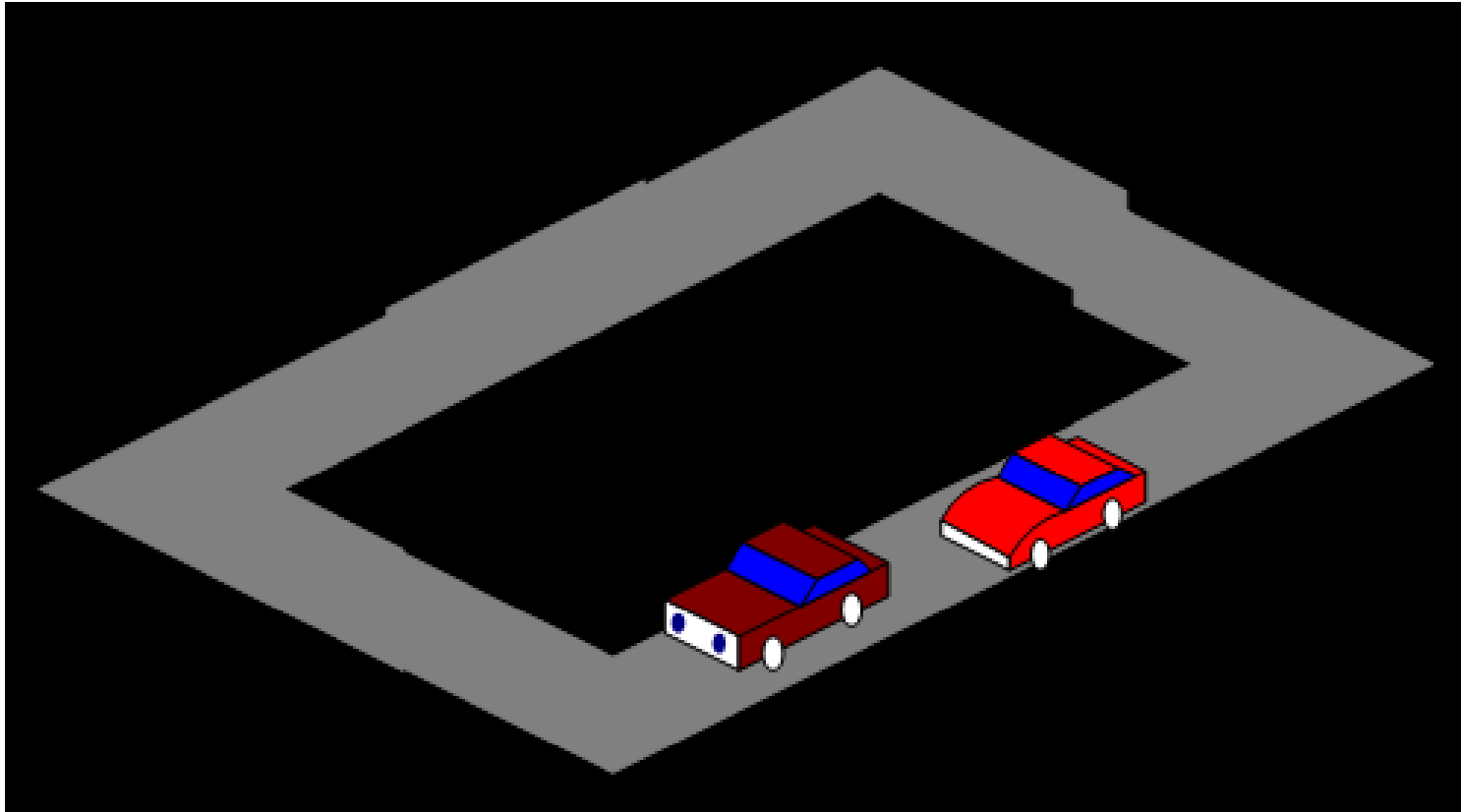


TPI4202  
e-tp.ub.ac.id

# Perulangan (*Looping*)

Lecture 6

# Perulangan



# Pengulangan

- Satu atau sekumpulan instruksi kadang harus diulang sejumlah kali untuk tujuan tertentu
- Hal yang harus ditentukan sebelum pengulangan dilakukan:
  - Mengapa harus diulang?
  - Berapa kali harus diulang?
  - Kapan berhenti?
  - Bagaimana dengan nilai awal?

# Jenis-jenis struktur perulangan

- Terdapat tiga macam statemen perulangan (*looping/repetition*) yang dapat digunakan:
  - Pengulangan-for
  - Pengulangan-while
  - Pengulangan-do-while
- Masing-masing digunakan pada jenis perulangan yang berbeda, meski untuk kasus sebuah perulangan dapat diganti dengan statemen perulangan yang lain.

# Perulangan FOR

- Digunakan untuk mengeksekusi code yang bernilai sama, berulang-ulang

- **Bentuk umum:**

```
for (InisialisasiEkspresi ; KondisiPengulangan ; StepEkspresi)
{
    statement1;
    statement2;
    . . .
}
```

- **dimana**

InisialisasiEkspresi –meninisialisasi variabel pengulangan.

KondisiPengulangan – membandingkan variabel pengulangan dengan nilai limit.

StepEkspresi – memperbarui variabel pengulangan.

# Contoh FOR

```
int i;  
for( i = 0; i < 10; i++ ){  
    System.out.println(i);  
}
```

# Perulangan WHILE

- Pernyataan while digunakan untuk perulangan yang banyaknya perulangan tidak diketahui. Pernyataan while mirip dengan pernyataan if yang melakukan pemeriksaan ekspresi boolean sebelum sebuah atau serangkaian pernyataan dilakukan.
- Statement atau blok dari statement diulang selama kondisinya sesuai
- **Bentuk umum:**

```
while( boolean_ekspresi ){  
    statement1;  
    statement2;  
    . . .  
}
```
- Jika ekspresi bernilai true, statement dijalankan dan diperiksa kembali dan keluar dari perulangan jika bernilai false.

# Contoh WHILE

```
int i = 0;
while( i < 10 ){
    System.out.print(i);
    i++; }
```

- code di atas sama dengan pengulangan-for berikut

```
int i;
for( i = 0; i < 10; i++ ){
    System.out.println(i);
}
```



# Perulangan WHILE

//Pengulangan tanpa batas

**while(true)**

**System.out.println("hello");**

//Tanpa pengulangan

// statement yang tidak pernah dieksekusi

**while (false)**

**System.out.println("hello");**

# Perulangan DO-WHILE

- Sama dengan pengulangan-while
- Statement di dalam pengulangan do-while akan dieksekusi beberapa kali selama kondisinya sesuai dengan ekspresi yang diberikan.
- Hal utama yang membedakan antara pengulangan while dan do-while:
  - Statement di dalam pengulangan do-while setidaknya dieksekusi satu kali.
- Form pengulangan-do-while:

```
do {  
    statement1;  
    statement2;  
}while( boolean_ekspresi );
```

# Contoh DO-WHILE

```
int x = 0;  
  
do {  
    System.out.println(x);  
    x++;  
}while (x<10);
```

# Perulangan DO-WHILE

```
//pengulangan tanpa batas  
do{  
    System.out.println("hello");  
} while (true);
```

```
//satu kali pengulangan  
// statement dieksekusi satu kali  
do  
    System.out.println("hello");  
while (false);
```

# Contoh2 Lain

- Program deret huruf
- Program tampil bilangan ganjil sesuai batasan
- Program tampil bilangan prima sesuai batasan
- Dll

# Perulangan FOR → Basic Array

```
int [] a = {1,2,3,4};  
for(int x = 0; x < a.length; x++) // basic for loop  
System.out.print(a[x]);  
for(int n : a) // enhanced for loop  
System.out.print(n);
```

# Kesimpulan

- Struktur perulangan di Java baik FOR, WHILE, maupun DO-WHILE membutuhkan 3 poin penting yang harus dipersiapkan, yaitu:
  - Nilai awal
  - Nilai akhir
  - Statement pengubah nilai sehingga perulangan bisa berhenti

# Review

1. Tampilkan bilangan genap antara 1 dan masukan user dengan menggunakan struktur perulangan FOR, WHILE dan DO-WHILE
2. Buat program Java untuk menghitung rata-rata nilai ujian sebuah kelas
3. Buat program Java untuk menghitung nilai faktorial sebuah bilangan