

The logo for Oracle Academy is centered on a light gray background. It features the word "ORACLE" in a bold, orange, sans-serif font. Below it, the word "Academy" is written in a smaller, dark gray, sans-serif font. The entire logo is framed by a thin black border, with dark gray horizontal bars at the top and bottom.

ORACLE

Academy

Yayasan Jawa

6-2

while dan do-while loop

ORACLE
Academy



Hak Cipta © 2022, Oracle dan/atau afiliasinya. Oracle, Java, dan MySQL adalah merek dagang terdaftar dan Oracle dan/atau afiliasinya. Nama lain mungkin merupakan merek dagang dari pemiliknya masing-masing.

Tujuan

- Pelajaran ini mencakup tujuan-tujuan berikut:
 - Gunakan while loop dalam program Java (pre-test)
 - Gunakan loop do-while dalam program Java (post-test)
 - Memahami ketika satu jenis loop mungkin lebih bermanfaat daripada lain



ORACLE
Academy

JFo 6-2
while dan do-while loop

Hak Cipta © 2022, Oracle dan/atau afiliasinya. Oracle, Java, dan MySQL adalah merek dagang terdaftar dari Oracle dan/atau afiliasinya. Nama lain mungkin merupakan merek dagang dari pemiliknya masing-masing.

3

Berapa Kali Diulang?

- Dalam beberapa situasi, Anda tidak tahu berapa kali harus mengulang sesuatu
- Artinya, Anda mungkin perlu mengulang beberapa kode hingga kondisi tertentu terjadi

Berapa Kali Diulang?

- Mari kita lihat sebuah contoh:

• Katakanlah Anda harus menulis sebuah program untuk memasukkan nilai ujian dan menemukan nilai rata-ratanya, tetapi Anda mungkin tidak tahu berapa banyak ujian yang terlibat

• Alih-alih memaksa pengguna untuk menghitung semuanya sebelumnya, Anda dapat mengizinkan mereka untuk memasukkan tanda satu per satu dan kemudian memasukkan -1 untuk menunjukkan penyelesaian entri

sementara Loop

- Dalam situasi seperti itu, Anda harus menggunakan perulangan **while** yang lebih mudah
- Ini bekerja seperti ini:

• Perulangan **while** secara terus-menerus mengeksekusi blok pernyataan sementara kondisi tertentu benar

while Loop Sintaks

- Pernyataan while mengevaluasi ekspresi boolean
- Pernyataan di dalam kurung kurawal dijalankan selama ekspresi boolean benar

```
while (<ekspresi boolean>)  
{ <pernyataan(s)> ; }//akhir sementara
```

Putaran Pra-Uji

- Sebuah pre-test loop mengevaluasi kondisi sebelum loop mengeksekusi
- Jika kondisi salah, perulangan berhenti atau mungkin tidak pernah menjalankan
- for dan while loop adalah pre-test loop

Skenario Hitung Mundur

- Mari kita menulis skenario Countdown yang dibahas di pelajaran sebelumnya dengan menggunakan while loop:

Apa yang kita tahu	Nama Teknis	Kode
Saat perulangan dimulai ... Ekspresi Inisialisasi <code>int i = 10;</code>		
Lanjutkan perulangan jika ...	Ekspresi Kondisi	<code>saya >= 0;</code>
Setelah setiap putaran...	Perbarui Ekspresi	<code>saya--;</code>
Kode untuk mengulang	Pernyataan Kode	<code>System.out.println(i);</code>





Skenario Hitung Mundur: while Loop

```
CountDownWhile kelas publik {  
  
    public static void main(String[] args) { int i = 10;  
        System.out.println("Hitungan Mundur untuk  
            Diluncurkan!");  
  
        while (i >= 0)  
            { System.out.println(i);  
              saya--; }//akhir sementara  
  
        System.out.println("Ledakan !"); }//akhir  
        metode main }//akhir kelas CountDownWhile
```



Beberapa while Loops Never Run

- Kemungkinan loop body tidak akan pernah berjalan jika kondisi sedemikian rupa sehingga ekspresi boolean sudah salah, misalnya:

```

kelas publik WhileLoopExample { public static
void main(String args[]) { int num = 0; System.out.println("Ayo
hitung sampai 10!"); while (bil > 10) { bil = bil + 1;
System.out.println("Nomor: " )//end while

+ bilangan);

System.out.println("Kami telah menghitung sampai 10! Hore!");
} //akhiri metode utama } //
akhiri kelas WhileLoopExample
  
```

ORACLE
Academy

JFo 6-2
while dan do-while loop

Hak Cipta © 2022, Oracle dan/atau afiliasinya. Oracle, Java, dan MySQL adalah merek dagang terdaftar dari Oracle dan/atau afiliasinya. Nama lain mungkin merupakan merek dagang dari pemiliknya masing-masing.

11

Pada contoh slide, nilai awal num adalah 0 dan ekspresi boolean adalah num > 10, bukan num < 10. Ini sudah salah sejak awal, karena 0 tidak akan pernah lebih besar dari 10. Perulangan while mengevaluasi ekspresi boolean, num > 10, menemukan bahwa itu salah, dan mencetak:

Ayo hitung sampai 10!

Kami telah menghitung sampai 10! Hore!

Terjebak dalam Putaran Tak Terbatas

- Anda akan terjebak dalam perulangan while jika Anda menulis kondisi boolean yang tidak akan pernah bernilai false
- Kami menyebutnya **loop tak terbatas** karena tidak pernah berhenti mengeksekusi
- Jika ini terjadi, perulangan Anda akan dieksekusi selamanya atau sampai Anda mengirimkan perintah interupsi
- Anda harus menghindari penulisan perulangan tak terbatas dan selalu memverifikasi ekspresi boolean untuk memastikan bahwa perulangan berhenti secara normal

Mari Kembali ke Skenario Hitung Mundur

- Bagaimana jika kita secara tidak sengaja menulis **i++** daripada **i--** di dalam while loop?

```
int i = 10;  
System.out.println("Hitungan Mundur untuk  
Diluncurkan!"); while (i >= 0) { System.out.println(i);  
    saya++; } //akhiri while System.out.println("Ledakan !");
```

- Itu akan terus menambahkan 1 ke i, mempertahankan nilainya lebih dari 10 selamanya
- Ini adalah perulangan tak terbatas karena kondisi boolean selalu benar, dan program ini terus dijalankan

Menggunakan while Loop dan Kelas Pemindai

- while loop sering digunakan dengan input dengan menggunakan the Kelas pemindai

```
public static void main(String[] args) {
    Konsol pemindai = Pemindai baru (System.in); int
    jumlah = 0;

    System.out.println("Masukkan angka (-1 untuk keluar): "); int num =
    console.nextInt(); while (bil != -1) { jumlah = jumlah + jumlah;
    System.out.println("Masukkan angka (-1 untuk keluar): "); num =
    console.nextInt(); }//akhir sementara

    System.out.println(" Jumlahnya adalah " + jumlah); }//
    akhir metode utama
```

Lihat slide berikutnya untuk hasil dari kode ini.

Menggunakan while Loop dan Kelas Pemindai

- **Contoh:**

Sebuah program yang meminta angka dari pengguna hingga mereka mengetik -1, dan kemudian menampilkan jumlah mereka

```
public static void main(String[] args) {
    Konsol pemindai = Pemindai baru (System.in); int
    jumlah = 0;

    System.out.println("Masukkan angka (-1 untuk keluar): "); int num =
    console.nextInt(); while (bil != -1) { jumlah = jumlah + jumlah;
    System.out.println("Masukkan angka (-1 untuk keluar): "); num =
    console.nextInt(); }//akhir sementara

    System.out.println(" Jumlahnya adalah " + jumlah); }//
    akhir metode utama
```

ORACLE
Academy

JFo 6-2
while dan do-while loop

Hak Cipta © 2022, Oracle dan/atau afiliasinya. Oracle, Java, dan MySQL adalah merek dagang terdaftar dari Oracle dan/atau afiliasinya. Nama lain mungkin merupakan merek dagang dari pemiliknya masing-masing.

15

Contoh slide menghasilkan output berikut:

Masukkan angka (-1 untuk keluar):

20

Masukkan angka (-1 untuk keluar):

40

Masukkan angka (-1 untuk keluar):

-1

Jumlahnya 60

Latihan 1

- Buat proyek baru dan tambahkan file SquareRootWhile.java ke proyek
- Modifikasi SquareRootWhile.java untuk menggunakan while loop untuk berulang kali meminta pengguna mengetikkan angka hingga mereka mengetikkan angka non-negatif, lalu menghitung akar kuadrat

- Keluaran yang diharapkan:

Ketik bilangan bulat bukan
negatif: -5 Angka salah, coba
lagi: -1 Angka salah, coba lagi:
11 Akar kuadrat dari 11 adalah 3,166

Putaran Pasca Tes

- Sebuah post-test loop mengevaluasi kondisinya di bagian bawah loop, bukan di atas
- Do-while loop adalah post-test loop

do-while Loop

- Perulangan do-while adalah perulangan while yang dimodifikasi yang memungkinkan Anda menjalankan perulangan satu kali, sebelum menguji kondisi boolean
- Sintaks:

```
do{ <pernyataan(s)> }  
sementara(<kondisi>;
```



Perulangan do-while membutuhkan titik koma setelah kondisi di akhir perulangan

Jika kondisinya salah, perulangan masih dijalankan setidaknya sekali, tetapi berhenti di akhir perulangan. Oleh karena itu, pernyataan-pernyataan di dalam blok do selalu dieksekusi setidaknya satu kali.



Skenario Hitung Mundur: do-while Loop

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

```
    int i = 10;
```

```
    System.out.println("Hitungan Mundur untuk Diluncurkan!");
```

```
    do
```

```
    { System.out.println(i); saya--; }
    sementara (i >= 0);
```

} Dieksekusi sekali sebelum
mengevaluasi kondisi

```
    System.out.println("Ledakan !"); } // akhir metode utama
```



ORACLE
Academy

JFo 6-2
while dan do-while loop

Hak Cipta © 2022, Oracle dan/atau afiliasinya. Oracle, Java, dan MySQL adalah merek dagang terdaftar dari Oracle dan/atau afiliasinya. Nama lain mungkin merupakan merek dagang dari pemiliknya masing-masing.

19

Keluaran:

Hitung mundur untuk Meluncurkan!

10

9

8

7

6

5 4

3

2

1

0 Ledakan!

Latihan 2

- Tambahkan file SumofNums.java ke proyek yang Anda buat untuk latihan 1
- Periksa SumofNums.java, yang merangkum urutan 10 bilangan bulat yang dimasukkan oleh pengguna • Bisakah Anda menerapkan hal yang sama dengan menggunakan do-while loop?

Standar untuk Loop Dibandingkan dengan while Loop

- Perbedaan antara kedua perulangan ini: •

Dalam perulangan for:

- Inisialisasi, kondisi, dan pernyataan kenaikan semuanya disatukan dalam satu baris, yang membuat perulangan lebih mudah dipahami dan diterapkan

Standar untuk Loop Dibandingkan dengan while Loop

- Perbedaan antara kedua perulangan ini:
- Dalam perulangan while:
 - Inisialisasi dilakukan sebelum permulaan loop
 - Pernyataan bersyarat selalu diletakkan di awal lingkaran
 - Pernyataan kenaikan dapat dikombinasikan dengan kondisi atau disematkan ke dalam tubuh loop

Dalam tiga slide berikutnya, Anda akan melihat contoh while loop di bagian atas slide. Di bagian bawah, Anda melihat logika yang sama diimplementasikan dengan menggunakan loop for standar.

Tiga elemen penting dari perulangan while juga ada di perulangan for, tetapi di tempat yang berbeda.

1. Pencacah (i) dideklarasikan dan diinisialisasi di luar while loop pada baris 1.
2. Penghitung bertambah dalam while loop pada baris 4.
3. Ekspresi boolean yang menentukan jumlah iterasi perulangan berada di dalam tanda kurung untuk perulangan while pada baris 2.

Dalam perulangan for, ketiga elemen terjadi di dalam tanda kurung, seperti yang ditunjukkan dalam slide. Output untuk setiap pernyataan adalah sama.

Membandingkan Penghitung Inisialisasi

Ulangi sementara

```
int i = 10; while  
(i >= 0)  
    { System.out.println(i); saya--; }//  
    akhiri while  
System.out.println("Ledakan !");
```

Inisialisasi
menangkal

Ulangi for

```
( int i = 10; i >= 0; i--)  
    { System.out.println(i); }//akhir  
untuk System.out.println("Ledakan !");
```

Membandingkan Ekspresi Boolean

Ulangi sementara

```
int i = 10;  
while (i >= 0)  
{ System.out.println(i);  
  saya--; } //akhiri while  
System.out.println("Ledakan !");
```

boolean
ekspresi

Ulangi for

```
( int i = 10; i >= 0; i--) { System.out.println(i); } //  
akhir untuk  
System.out.println("Ledakan !");
```


Membandingkan Penghitung Kenaikan

Ulangi sementara

```
int i = 10; while  
(i >= 0)  
{ System.out.println(i);  
  saya--; } // akhiri while  
System.out.println("Ledakan !");
```

Kenaikan
menangkal

Ulangi for

```
( int i = 10; i >= 0; i--) { System.out.println(i); } //  
akhir untuk System.out.println("Ledakan !");
```

Loop mana yang saya gunakan?

Tipe Lingkaran	Definisi	Kapan Menggunakan
ketika	Pre-test loop yang berulang sampai kondisi tertentu salah	Gunakan ketika Anda tidak yakin berapa kali loop harus dieksekusi atau bahkan jika harus sama sekali
lakukan-sementara	Perulangan post-test yang mengeksekusi perulangan sebelum menguji kondisi, kemudian berulang sampai kondisi salah	Gunakan ketika Anda mengetahui bahwa kode harus dieksekusi setidaknya sekali dan mungkin lebih banyak tergantung pada kondisinya
untuk	Loop yang berisi penghitung yang diinisialisasi, dan menambah penghitung dengan setiap putaran melalui loop. Ulangi sampai kondisi salah	Gunakan saat Anda perlu menjalankan loop beberapa kali, atau saat Anda perlu menambah satu set data. Penghitung juga dapat digunakan sebagai indeks untuk mengakses data satu item pada satu waktu

Ringkasan

- Dalam pelajaran ini, Anda seharusnya telah mempelajari cara untuk:
 - Gunakan while loop dalam program Java (pre-test)
 - Gunakan loop do-while dalam program Java (post-test)
 - Memahami ketika satu jenis loop mungkin lebih bermanfaat daripada lain



ORACLE
Academy

JFo 6-2
while dan do-while loop

Hak Cipta © 2022, Oracle dan/atau afiliasinya. Oracle, Java, dan MySQL adalah merek dagang terdaftar dari Oracle dan/atau afiliasinya. Nama lain mungkin merupakan merek dagang dari pemiliknya masing-masing.

27

The logo for Oracle Academy is centered on a light gray background. It features the word "ORACLE" in a bold, orange, sans-serif font. Below it, the word "Academy" is written in a smaller, dark gray, sans-serif font. The entire logo is framed by a thin black border, with dark gray horizontal bars at the top and bottom.

ORACLE

Academy