

Book Series
of
jago{coding}
Learn & Share {Your Code}

www.jagocoding.com

Cara Mencari Selisih Waktu di Java

Oleh: Yudi Setiawan

Dalam bahasa pemrograman Java, ada library yang bisa Anda gunakan untuk mencari selisih waktu yakni JodaTime. JodaTime berfungsi untuk mencari selisih waktu di Java. Dalam beberapa kasus, JodaTime tidak selalu bisa Anda gunakan namun, kalau Anda hanya ingin mencari selisih waktu saya rasa JodaTime masih b...

Dalam bahasa pemrograman Java, ada library yang bisa Anda gunakan untuk mencari selisih waktu yakni JodaTime. JodaTime berfungsi untuk mencari selisih waktu di Java. Dalam beberapa kasus, JodaTime tidak selalu bisa Anda gunakan namun, kalau Anda hanya ingin mencari selisih waktu saya rasa JodaTime masih bisa Anda gunakan. Untuk contoh kasus, menghitung total jam kerja si Pegawai yang masuk pukul 08:00 dan pulang pukul 17:00 maka, JodaTime masih mampu menyelesaikannya namun, jika di perusahaan tersebut terdapat shift 24 jam dan si Pegawai masuk pukul 22:00 dan pulang pukul 07:00 maka, JodaTime akan menghasilkan output yang tidak sesuai pula dan outputnya ialah 15 jam. Lalu, bagaimana cara menyelesaikan contoh kasus yang si Pegawai masuk malam dan pulang pagi tersebut? Iya, jelas bisa. Anda harus bermain sedikit logika Anda. Berikut algoritma untuk menyelesaikannya:

1. Input Waktu masuk si pegawai dan waktu pulang si pegawai.
2. Kemudian, ubah Waktu masuk dan Waktu pulang menjadi 3 bagian yakni Jam, Menit dan Detik.
3. Buat sebuah variable untuk menghitung berapa selisih waktu masuk dengan waktu pulang. Nilai ini akan selalu bertambah 1 pada poin nomor 4 dan 5.
4. Setelah itu, lakukan secara berurutan dimulai dari detik dimana, Anda harus mengsamakan nilai detik waktu masuk dan detik waktu pulang. Caranya, cukup dengan cara menambahkan atau increment nilai detik waktu masuk. Lakukan ini dengan menggunakan perulangan. Dan jika nilai detik melewati 59 maka, nilai menit waktu masuk ditambah 1. Ingat, ketika perulangan maka, nilai variable yang Anda buat pada poin 3 tadi juga ikut Anda tambahkan atau increment 1.
5. Setelah selesai dengan detiknya maka, lakukanlah hal yang sama dengan menit waktu masuk dan menit waktu pulang dimana, Anda harus mengsamakan nilai waktu menit masuk dengan waktu menit pulang dengan cara increment dan lakukan dengan menggunakan perulangan. Jika nilai menit waktu masuk melewati menit 59 maka, nilai waktu jam ditambah 1. Lakukanlah hal yang sama dengan nilai jam waktu masuk. Ingat, pada poin ini juga Anda increment nilai variable pada poin nomor 3 tadi.
6. Setelah nilai waktu jam, menit dan detik masuk sama dengan waktu jam, menit dan detik pulang maka, saatnya mengolah nilai variable pada poin nomor 3 yang sudah Anda increment pada poin nomor 4 dan 5. Bagaimana cara mengolahnya? Anda bisa mengolahnya untuk mendapatkan nilai selisih dalam format HH:mm:ss atau Anda bisa juga mengolahnya menjadi satuan detik semua. Itu semua tergantung Anda sendiri gimana maunya.

Perlu diketahui, ini merupakan salah satu cara dari seribu satu cara yang ada untuk menyelesaikan masalah diatas tadi. Dan berikut ialah source code untuk menyelesaikan masalah diatas.

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

```

/**
 *
 *    @author Yudi Setiawan
 *
 */

public class Main {
    public static void main(String[] args) {

        //          Scanner
        Scanner scan = new Scanner(System.in);

        //          Input Waktu Awal
        System.out.println("Waktu Awal");
        System.out.print("Input Jam   : ");           String
str_jam_awal = scan.nextLine();
        System.out.print("Input Menit : ");           String
str_menit_awal = scan.nextLine();
        System.out.print("Input Detik : ");           String
str_detik_awal = scan.nextLine();

        //          Input Waktu Akhir
        System.out.println("\nWaktu Akhir");
        System.out.print("Input Jam   : ");           String
str_jam_akhir = scan.nextLine();
        System.out.print("Input Menit : ");           String
str_menit_akhir = scan.nextLine();
        System.out.print("Input Detik : ");           String
str_detik_akhir = scan.nextLine();

        //          Tampilkan Waktu Awal dan Waktu Akhir
        String str_waktu_awal =
str_jam_awal+":"+str_menit_awal+":"+str_detik_awal;
        String str_waktu_akhir =
str_jam_akhir+":"+str_menit_akhir+":"+str_detik_akhir;
        System.out.println("Waktu Awal  : "+str_waktu_awal);
        System.out.println("Waktu Akhir  : "+str_waktu_akhir);

        //          Konversi variable str menjadi integer
        int jam_awal = Integer.parseInt(str_jam_awal);
        int jam_akhir = Integer.parseInt(str_jam_akhir);
        int menit_awal = Integer.parseInt(str_menit_awal);
        int menit_akhir = Integer.parseInt(str_menit_akhir);
        int detik_awal = Integer.parseInt(str_detik_awal);
        int detik_akhir = Integer.parseInt(str_detik_akhir);

        //          Proses pencarian selisih Waktu Awal dan Waktu
        Akhir
        //          Langkah Ketiga, cari selisih detik awal dan
        detik akhir
        int selisih_detik = 0;
    }
}

```

```

//      Jika detik awal > detik akhir
//      Contohnya, detik awal = 40 dan detik akhir = 30
if(detik_awal > detik_akhir) {

    while(detik_awal != detik_akhir) {

        if(detik_awal == 60) {
            detik_awal = 0;
            menit_awal++;
            continue;
        }
        detik_awal++;
        selisih_detik++;
    }
}

//      Jika detik awal < detik akhir
//      Contohnya, detik awal = 30 dan detik akhir = 40
else if(detik_awal < detik_akhir) {
    selisih_detik = detik_akhir - detik_awal;
}

//      Masukkan ke variable selisih waktu
int selisih_waktu = selisih_detik;

//-----
//      Langkah Kedua, cari selisih menit awal dan menit
akhir

int selisih_menit = 0;

//      Jika menit awal > menit akhir.
//      Contohnya, menit awal = 50 dan menit akhir = 10
if(menit_awal > menit_akhir) {

    while(menit_awal != menit_akhir) {

        if(menit_awal == 60) {
            menit_awal = 0;
            jam_awal++;
            continue;
        }

        menit_awal++;
        selisih_menit++;
    }
}

//      Jika menit awal < menit akhir
//      Contohnya, menit awal = 10 dan menit akhir = 50
else if(menit_awal < menit_akhir) {

```

```

        selisih_menit = menit_akhir - menit_akhir;
    }

    //      Ubah selisih menit menjadi satuan detik (1 menit
= 60 detik)
    selisih_waktu += selisih_menit * 60;

    //      Langkah Pertama, cari selisih jam awal dan jam
akhir
    int selisih_jam = 0;

    //      Jika jam awal > jam akhir. Contohnya, Jam Awal =
22 (10 malam) dan Jam Akhir = 05 (5 pagi)
    if(jam_awal > jam_akhir) {

        //      Cocokkan antara jam awal dan jam akhir
        while(jam_awal != jam_akhir) {
            if(jam_awal == 24) {
                jam_awal = 0;
                continue;
            }

            jam_awal++;
            selisih_jam++;
        }
    }

    //      Jika jam awal < jam akhir. Contohnya, Jam Awal =
05 (5 pagi) dan Jam Akhir = 22 (10 malam)
    else if(jam_awal < jam_akhir)
        selisih_jam = jam_akhir - jam_awal;

    //      Ubah selisih jam menjadi satuan detik (1 jam =
3600 detik)
    selisih_waktu += selisih_jam * 3600;

    //      Tampilkan selisih antara waktu awal dan waktu
akhir dalam satuan detik
    System.out.println("Selisih antara waktu awal dan waktu
akhir (detik) : "+selisih_waktu);

    //      Ubah selisih_waktu menjadi format waktu HH:mm:ss
    //      Ubah selisih waktu menjadi satuan jam
    int jam = selisih_waktu / 3600;
    selisih_waktu %= 3600;

    //      Ubah selisih waktu menjadi satuan menit
    int menit = 0;
    int detik = 0;
    if(selisih_waktu >= 60) {

```

```

        menit = selisih_waktu / 60;
        detik = selisih_waktu % 60;

    } else {
        menit = 0;
        detik = selisih_waktu;
    }

    //      Tampilkan selisih antara waktu awal dan waktu
akhir dalam format waktu HH:mm:ss
    String waktu = jam+":"+menit+":"+detik;
    System.out.println("Selisih antara waktu awal dan waktu
akhir (HH:mm:ss) : "+waktu);
    }
}

```

Dan berikut ialah yang menggunakan library JodaTime. Jangan lupa download terlebih dahulu library nya di internet.

```

import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Scanner;

import org.joda.time.DateTime;
import org.joda.time.Seconds;

/**
 *
 * @author Yudi Setiawan
 *
 */

public class Main2 {
    public static void main(String[] args) throws ParseException {

        //      SimpleDateFormat
        SimpleDateFormat sdfTime = new
SimpleDateFormat("HH:mm:ss");

        //      Scanner
        Scanner scan = new Scanner(System.in);

        //      Input Waktu Awal
        System.out.println("Waktu Awal");
        System.out.print("Input Jam   : ");      String
waktu_awal = scan.nextLine()+" : ";
        System.out.print("Input Menit : ");      waktu_awal +=
scan.nextLine()+" : ";

```

```

        System.out.print("Input Detik : ");        waktu_awal +=
scan.nextLine();

        //      Input Waktu Akhir
        System.out.println("\nWaktu Akhir");
        System.out.print("Input Jam   : ");        String
waktu_akhir = scan.nextLine()+":";
        System.out.print("Input Menit : ");        waktu_akhir +=
scan.nextLine()+":";
        System.out.print("Input Detik : ");        waktu_akhir +=
scan.nextLine();

        //      Ubah waktu awal dan waktu akhir menjadi DateTime
        DateTime dtAwal = new DateTime(sdfTime.parse(waktu_awal));
        DateTime dtAakhir = new
DateTime(sdfTime.parse(waktu_akhir));

        //      Tampilkan waktu awal dan waktu akhir
        System.out.println();
        System.out.println("Waktu Awal   : "+waktu_awal);
        System.out.println("Waktu Akhir  : "+waktu_akhir);

        //      proses cari selisih waktu dalam satuan detik
        //      10:50:20 - 11:30:50
        int selisih_detik = Math.abs(Seconds.secondsBetween(dtAwal,
dtAakhir).getSeconds());
        System.out.println("Selisih waktu awal dan waktu akhir
(detik) : "+selisih_detik);

        //      Ubah selisih detik menjadi format HH:mm:ss
        int jam = selisih_detik / 3600;
        selisih_detik %= 3600;

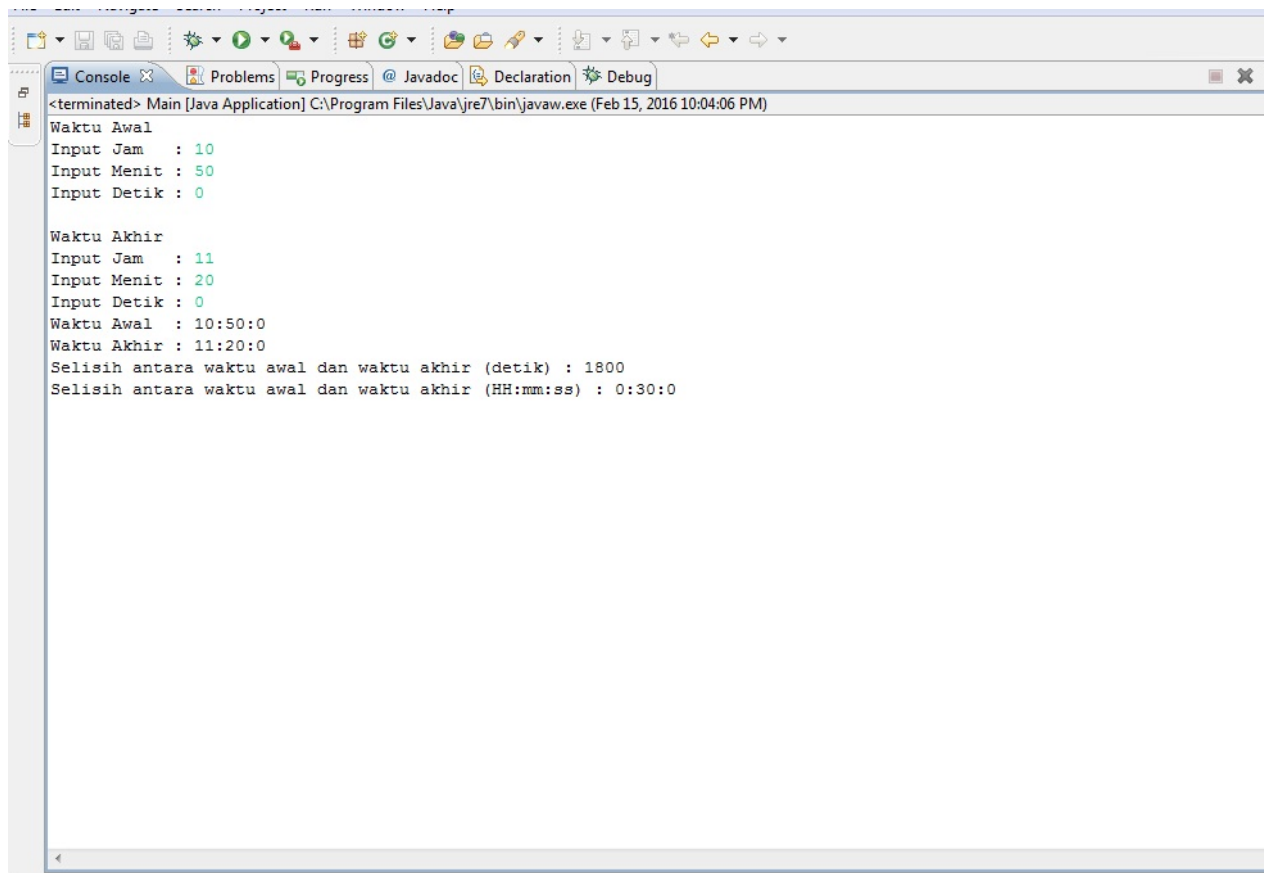
        int menit = 0;
        int detik = 0;
        if(selisih_detik >= 60) {
            menit = selisih_detik / 60;
            detik = selisih_detik % 60;
        } else {
            detik = selisih_detik;
        }

        //      Tampilkan selisih waktu dalam format HH:mm:ss
        String selisih_waktu = jam+": "+menit+": "+detik;
        System.out.println("Selisih waktu awal dan waktu akhir
(HH:mm:ss) : "+selisih_waktu);

    }
}

```

Dan berikut output untuk kedua program diatas.



The image shows a screenshot of an IDE's console window. The window title is "<terminated> Main [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe (Feb 15, 2016 10:04:06 PM)". The console output is as follows:

```
Waktu Awal
Input Jam : 10
Input Menit : 50
Input Detik : 0

Waktu Akhir
Input Jam : 11
Input Menit : 20
Input Detik : 0
Waktu Awal : 10:50:0
Waktu Akhir : 11:20:0
Selisih antara waktu awal dan waktu akhir (detik) : 1800
Selisih antara waktu awal dan waktu akhir (HH:mm:ss) : 0:30:0
```



```
<terminated> main2 [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe (Feb 15, 2016 10:06:42 PM)
Waktu Awal
Input Jam : 10
Input Menit : 10
Input Detik : 0

Waktu Akhir
Input Jam : 11
Input Menit : 0
Input Detik : 0

Waktu Awal : 10:10:0
Waktu Akhir : 11:0:0
Selisih waktu awal dan waktu akhir (detik) : 3000
Selisih waktu awal dan waktu akhir (HH:mm:ss) : 0:50:0
```

Tentang Penulis



Yudi Setiawan

Saat ini aktif sebagai Mahasiswa di salah satu Universitas di kota Medan dengan mengambil bidang Fakultas Teknik dan Ilmu Komputer. Sangat senang dengan bahasa pemrograman Java dan Android.