

PEMOGRAMAN APLIKASI BERGERAK (Java ME)

Yoannita





TEXTFIELD



TextField Constructor

- ❖ Konstruktor dari class TextField adalah:
`public TextField(String title, String text, int maxSize, int constraints)`
- ❖ Parameter-parameter pada konstruktor adalah:
 - **String title** : untuk memberikan title pada TextField
 - **String text** : untuk memberikan nilai awal pada TextField
 - **int maxsize** : untuk memberikan batasan maksimal jumlah karakter yang bisa diterima oleh TextField. Jadi Anda tidak bisa memasukkan karakter yang jumlahnya melebihi batas maksimal ini.
 - **int constraints**

TextField

TextField merupakan Item dimana user dapat memasukkan encode.

Beberapa batasan exclusive yang dapat di-set yaitu:

- ❖ TextField.ANY
- ❖ TextField.EMAILADDR
- ❖ TextField.NUMERIC
- ❖ TextField.PHONENUMBER
- ❖ TextField.URL
- ❖ TextField.DECIMAL

Masukan tersebut juga dapat memiliki macam-macam modifikasi:

- ❖ TextField.PASSWORD
- ❖ TextField.UNEDITABLE
- ❖ TextField.SENSITIVE
- ❖ TextField.NON_PREDICTIVE
- ❖ TextField.INITIAL_CAPS_WORD
- ❖ TextField.INITIAL_CAPS_SENTENCE

Keterangan TextField

Nilai Constraint (batasan)	Keterangan
TextField.ANY	Mengizinkan semua karakter ditulis pada TextField
TextField.EMAILADDR	Hanya mengizinkan masukan berupa alamat email pada TextField
TextField.NUMERIC	Hanya mengizinkan masukan berupa angka pada TextField
TextField.PHONENUMBER	Hanya mengizinkan masukan berupa nomor telepon pada TextField
TextField.URL	Hanya mengizinkan masukan berupa alamat URL



TextField.PASSWORD

- ❖ `TextField.PASSWORD` akan mengubah masukan menjadi karakter asterik dengan alasan keamanan

- ❖ Biasanya digunakan dengan `TextField.ANY` atau `TextField.NUMERIC` dengan penulisan :

- ✓ `TextField.PASSWORD` | `TextField.ANY`

atau :

- ✓ `TextField.PASSWORD` | `TextField.NUMERIC`

Macam-macam TextField

```
TextField ANY = new TextField("ANY", "", 64,  
    TextField.ANY);
```

```
TextField EMAILADDR =  
    new TextField("EMAILADDR", "", 64,  
    TextField.EMAILADDR);
```

```
TextField NUMERIC = new TextField("NUMERIC",  
    "", 64, TextField.NUMERIC);
```

```
TextField PHONENUMBER = new  
    TextField("PHONENUMBER", "", 64,  
    TextField.PHONENUMBER);
```

```
TextField URL = new TextField("URL", "", 64,  
    TextField.URL);
```

```
TextField DECIMAL = new TextField("DECIMAL",  
    "", 64, TextField.DECIMAL);
```



The screenshot shows a mobile application interface titled "TextField Types". It displays several text input fields with different types and example values:

- ANY**: test123#
- EMAILADDR**: any@mail.com
- NUMERIC**: 88888888
- PHONENUMBER**: 999 8888
- URL**: http://www.sun.com
- DECIMAL**: 88.88

At the bottom of the screen, there is a "Back" button.

Membuat sebuah TextField

TextField (**String label** , **String text** , **int maxSize** , **int constraints**)

❖ Contoh pendeklarasian :

```
txtphone    =  
new TextField("Nomor telepon tujuan",  
              "081812345678", 20,  
              TextField.PHONENUMBER);
```


Latihan(1)

- ❖ Buatlah form yang mempunyai textfield dengan karakteristik berikut ini :
- ❖ Isilah nilai textfield sehingga pada saat pemunculan form akan langsung ditampilkan nilai defaultnya



The screenshot shows a mobile application interface with a title bar containing a signal strength indicator, the number '123', and a battery icon. Below the title bar is a header labeled 'TextField Types'. The form contains several text input fields, each with a label above it and a default value inside the field:

- ANY**: test123#
- EMAILADDR**: any@mail.com
- NUMERIC**: 88888888
- PHONENUMBER**: 999 8888
- URL**: http://www.sun.com
- DECIMAL**: 88.88

At the bottom of the screen, there is a 'Back' button.

Metode-metode yang ada pada TextField(i)

Metode	Keterangan
TextField (String label, String text, int maxSize, int constraints)	Membuat sebuah TextField baru dengan label masukan, isinya text, ukuran maksimal maxSize, dan batasan constraints.
Void delete (int offset, int length)	Menghapus text mulai dari indeks offset sepanjang length.
Void insert (String src, int position)	Memasukkan String src pada posisi position.
Void insert (char[] data, int offset, int length, int position)	Memasukkan data pada indeks offset sepanjang length pada array ke posisi position pada TextField.
Void setChars (char[] data, int offset, int length)	Mengganti isi TextField dengan data pada indeks offset sepanjang length.

Metode-metode yang ada pada TextField(ii)

Metode	Keterangan
Void setString (String Text)	Mengganti isi TextField dengan text.
Int getChars (char[] data)	Memasukkan isi TextField pada data.
String getString ()	Mengembalikan isi TextField.
Int getConstraints ()	Mengembalikan batasan (<i>constraint</i>) TextField.
Void setConstraints (int constraints)	Mengeset batasan (<i>constraint</i>) pada TextField.
Int getMaxSize ()	Mengembalikan nilai kapasitas maksimum dari TextField.
Int setMaxSize (int maxSize)	Mengeset nilai kapasitas maksimum TextField.
Int getCaretPosition ()	Mengembalikan nilai posisi kursor pada TextField sekarang.
Int size ()	Mengembalikan nilai jumlah karakter yang ada pada TextField.



.getString()

❖ Method `.getString()` digunakan untuk mengambil isi dari `TextField`.

❖ Contoh:

```
frmHitung.append(txtnama.getString());
```

Latihan (2)



The screenshot shows a mobile application interface with a dark grey header and footer. The header contains the Sun Microsystems logo and a power button icon. The main content area is titled "Pengisian Data" and contains several input fields: "Nama" (empty), "NPM" (containing "2008"), "Kelas" (empty), "Hobi" (empty), and "Cita-cita" (empty). A "Qwerty" keyboard icon is visible next to the "Nama" and "Cita-cita" fields. The footer contains two buttons: "Keluar" and "Menu".

Pengisian Data

Nama

NPM

2008

Kelas

Hobi

Cita-cita

Keluar Menu



The screenshot shows a mobile application interface with a dark grey header and footer. The header contains the Sun Microsystems logo and a power button icon. The main content area is titled "Informasi" and contains a text block with placeholder text. The footer contains a "Back" button.

Informasi

Namaku xxxxxxx, dengan Nomor Pokok Mahasiswa 2008xxxxxxx dan kelas xxxx, hobiku adalah xxxxxxxxxxx dan cita-cita gue ingin jadi xxxxxxxxxxx.

Back

Konversi String ke Integer

❖ `Integer.parseInt(<string>)`

❖ Contoh :

```
❖ int h = Integer.parseInt("20");  
// akan menghasilkan nilai h = 20
```

```
❖ int i =  
    Integer.parseInt(txtHarga.getString());  
// akan menghasilkan nilai i sesuai dengan  
    isi textField txtHarga
```



Konversi Integer ke String

`String.valueOf(<integer>)`

❖ Contoh penggunaan :

❖ `int total = 1200;`

❖ `String t = String.valueOf(total);`

// akan menghasilkan nilai t = "1200"

❖ `frmHitung.append(t);`

Latihan (3)

The screenshot shows a mobile application interface with a dark grey header containing the 'Sun' logo and 'microsystems' text. The main content area has a blue background and is titled 'Pengisian Data'. It contains two input fields: 'Angka pertama' with a value of '123' and 'Angka kedua' which is empty. At the bottom, there are two buttons: 'Keluar' and 'Menu'.

Pengisian Data

Angka pertama
123

Angka kedua

Keluar Menu

The screenshot shows the same mobile application interface, now displaying the results of a calculation. The main content area is titled 'Hasil Kalkulasi' and shows the following text: 'Nilai angka pertama = 30', 'Nilai angka kedua = 10', a separator line of equals signs, 'Hasil penjumlahan = 40', and 'Hasil pengurangan = 20'. At the bottom, there is a single button labeled 'Back'.

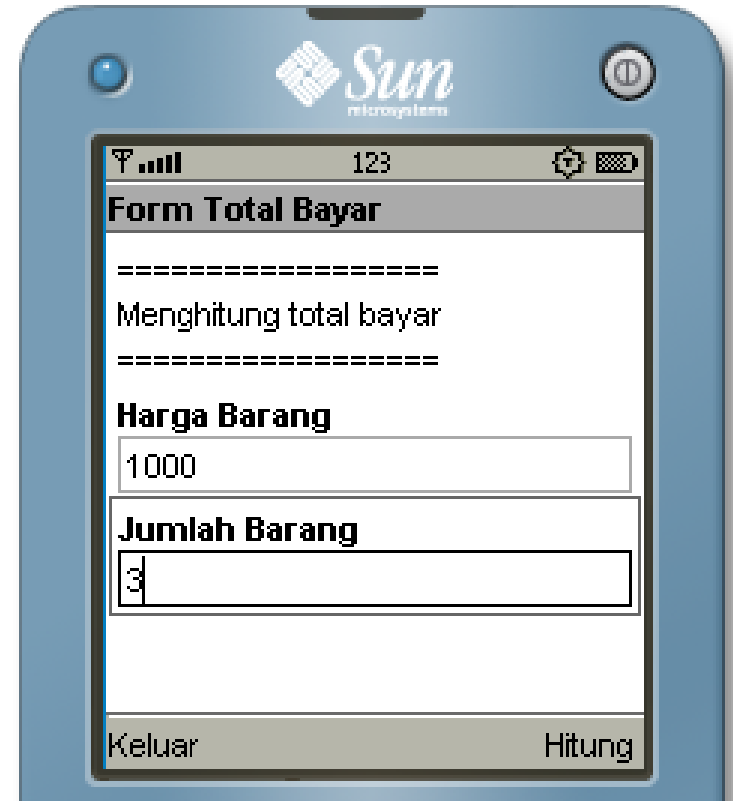
Hasil Kalkulasi

Nilai angka pertama = 30
Nilai angka kedua = 10
=====
Hasil penjumlahan = 40
Hasil pengurangan = 20

Back

Latihan(4)

- ❖ Buatlah sebuah form yang mempunyai dua textfield dan dua command seperti tampak pada gambar di samping! Tampilkan total bayar berdasarkan harga dan jumlah barang yang diinput.



Latihan (4)

Sun
microsystems

Form Total Bayar

=====
Menghitung total bayar
=====

Harga Barang

1000

Jumlah Barang

3

Keluar Hitung

Sun
microsystems

Form Total Bayar

=====
Menghitung total bayar
=====

Harga barang : Rp. 1000
Jumlah barang : 3 buah
Total yang harus dibayar Rp. 3000

Keluar Hitung

IF ELSE

- ❖ Buatlah program yang menginput angka antara 1-9 lalu menampilkan terbilanganya, seperti contoh berikut ini :



Contoh IF ELSE

```
public class ProgramPertama extends MIDlet implements CommandListener{
    Display display;
    Form frmHitung;
    TextField txtAngka;
    Command cmdOK, cmdKembali;

    public ProgramPertama()
    {
        display = Display.getDisplay(this);
        frmHitung = new Form("Contoh IF ELSE");
        cmdOK = new Command("OK",Command.OK,1);
        cmdKembali = new Command("Kembali", Command.BACK,1);
        txtAngka = new TextField("Input Angka", "", 1, TextField.NUMERIC);
        frmHitung.append(txtAngka);
        frmHitung.addCommand(cmdOK);
        frmHitung.setCommandListener(this);
    }
}
```

Contoh IF ELSE

```
public void startApp() {
    display.setCurrent(frmHitung);
}
public void pauseApp() {
}
public void destroyApp(boolean unconditional) {
}

public void commandAction(Command c, Displayable d) {
    if (c == cmdOK)
    {
        frmHitung.deleteAll();
        frmHitung.removeCommand(cmdOK);
        frmHitung.addCommand(cmdKembali);
        int bil = Integer.parseInt(txtAngka.getString());
        String ket="";
```

Contoh IF ELSE

```
if(bil == 1)
{
    ket = "SATU";
}
else if (bil == 2)
{
    ket = "DUA";
}
else if (bil == 3)
{
    ket = "TIGA";
}
frmHitung.append("Angka
yang diinput adalah angka " + ket);
}
```

```
else if (c == cmdKembali)
{
    frmHitung.deleteAll();
    frmHitung.append(txtAngka);

    frmHitung.removeCommand(cmdKembali);
    frmHitung.addCommand(cmdOK);
}
}
```

Latihan (5)

- ❖ Buatlah program yang menginput kode bulan lalu menampilkan keterangannya sesuai dengan kode bulan yang diinputkan
- ❖ Tampilkan tulisan “angka/kode bulan salah. Angka harus antara 1-12” jika pengguna salah menginputkan kode bulan
- ❖ Tombol Back digunakan untuk kembali ke layar sebelumnya



Membandingkan String

equals dan **equalsIgnoreCase**

Contoh (potongan) kode program

```
...
if (teks.equals("HaLLo"))
{
    System.out.println ("teks berisi huruf HaLLo");
}
...
if (teks.equalsIgnoreCase("HaLLo"))
{
    System.out.println ("teks berisi huruf hallo (huruf besar maupun kecil)");
}
...
```

```
if (teks ==("HaLLo")) → SEBAIKNYA JANGAN DIGUNAKAN!
{
    System.out.println ("teks berisi huruf HaLLo");
}
```




Equals dan EqualsIgnoreCase

- ❖ Method **equals** akan membandingkan suatu string persis sama dengan string yang hendak dibandingkan,
String a = "Teknik Komputer";
a.equals("teknik komputer") → FALSE
- ❖ Method **equalsIgnoreCase** mengabaikan huruf besar dan kecil, sehingga huruf besar maupun kecil dianggap sama
String a = "Teknik Komputer";
a.equalsIgnoreCase("teknik komputer") → TRUE

Latihan(6)

- ❖ Buatlah program Java ME yang menginputkan kode jurusan lalu menampilkan nama jurusannya sesuai ketentuan berikut ini :

Kode Jurusan	Nama Jurusan
TI	Teknik Informatika
SI	Sistem Informasi
MI	Manajemen Informatika
TK	Teknik Komputer
KA	Komputerisasi Akuntansi



Thanks!

