

Tutorial Aplikasi Web JavaServer Faces(JSF) dengan NetBeans 6.0

Bagian 1 : Perkenalan

Yudi Tri Wibowo

yuditriwibowo@yahoo.com

Lisensi Dokumen:

Copyright © 2003-2006 IlmuKomputer.Com

Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin

Java adalah bahasa pemrograman ataupun bisa disebut sebagai platform yang paling banyak digunakan untuk mengembangkan aplikasi Enterprise. Kebutuhan SDM yang mahir Java sangat tinggi. Sayangnya banyak orang yang beranggapan bahwa membangun aplikasi dengan Java itu tidak mudah. Pada seri tutorial ini akan ditunjukkan bahwa membangun aplikasi web dengan menggunakan IDE NetBeans dan framework JavaServer Faces(JSF) itu sangat mudah.

Tutorial ini menunjukkan bagaimana membuat aplikasi web Java dengan framework JavaServer Faces menggunakan IDE NetBeans 6.0. Pada bagian perkenalan ini, tutorial dimulai dengan contoh “Hello” sederhana. Selanjutnya aplikasi ini dikembangkan untuk mengambil nama dari database.

Catatan : Tulisan ini merupakan terjemahan dari tutorial yang ada di www.netbeans.org. Menerapkan petuah bijak “Don't reinvent the wheel!”, penulis merasa bahwa tutorial yang ada sudah sangat bagus dan tidak ingin melakukan pekerjaan “reinventing the wheel” dengan membuat tutorial baru dari awal.

Software yang Dibutuhkan pada Tutorial Ini

Sebelum mulai, anda perlu menginstal software berikut pada komputer anda.

- NetBeans IDE 6.0 versi Web & Java EE atau versi All. (Anda bisa mendapatkan NetBeans 6.0 di <http://www.netbeans.org> atau silakan kirim email ke penulis di yuditriwibowo@yahoo.com

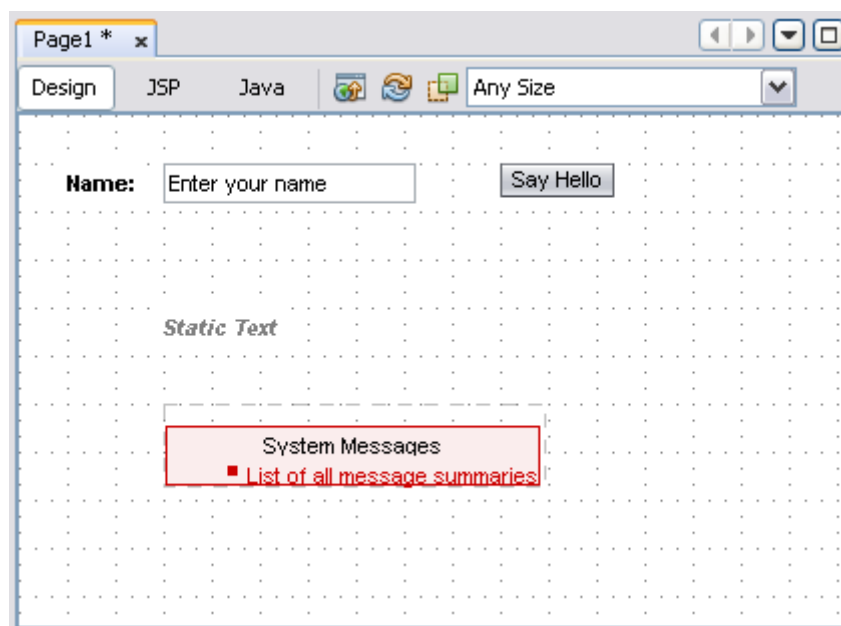
Membuat Project

1. Dari menu utama, pilih File > New Project.
2. Pada New Project Wizard, pilih Web dari list Categories dan pilih Web Application dari list Projects. Kemudian klik Next
3. Isikan HelloWeb sebagai nama project dan kemudian klik Next.
4. Pilih Visual Web JavaServer Faces untuk pilihan framework dan klik Finish.

Project anda akan terbuat dengan halaman awal (Page1) terbuka di Visual Designer.

Mendesain halaman

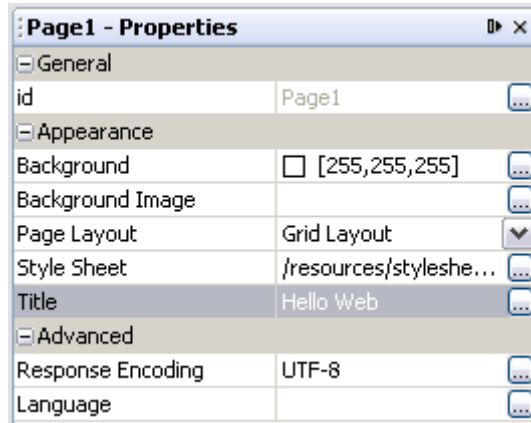
Untuk memulai, anda akan mendesain satu halaman seperti gambar berikut.



Gambar 1: Desain Page1

1. Pada window Properties, ketikkan Hello Web di text box untuk property Title seperti ditunjukkan gambar berikut.

Nilai dari property Title akan muncul di title bar dari browser ketika halaman (project) ini dideploy.



Gambar 2: Property-property page di Window Properties

Tip: Anda dapat mengakses property dari sebuah komponen di window Properties dengan cara memilih komponen tersebut di Visual Designer atau window Navigator. Untuk membuka window Navigator, pilih Window > Navigating > Navigator.

Sedangkan untuk mengakses property dari halaman, anda dapat mengklik tempat kosong di halaman.

2. Jika node Basic di window Palette tidak terekspan, ekspansikanlah sekarang. Semua komponen yang akan kita gunakan di contoh ini berada di bagian Basic dari palette. Jika palette belum tampak, pilihlah Window > Palette untuk memunculkannya.
3. Drag satu komponen Label dari Palette bagian Basic ke bagian kiri dari halaman yang ada di Visual Designer, ketikkan Name : dan tekan Enter.
4. Perhatikan bahwa komponen label tersebut akan menempel ke grid di halaman. Perhatikan juga bahwa nilai dari property text di window Properties berubah menjadi Name :.

Tip: Anda dapat mengubah suatu komponen menjadi ke mode edit dengan cara melakukan klik kanan pada komponen dan memilih Edit Label Text dari pop-up menu.

5. Drag satu komponen Text Field dari Palette bagian Basic ke Visual Designer, ketikkan Enter Your Name, dan tekan Enter.
6. Di window Property, ubahlah property id milik Text Field dari textField1 menjadi nameField.
7. Pilih komponen Label Ctrl-Shift-Drag dari komponen Label ke komponen Text Field.

Perhatikan bahwa property for komponen Label sekarang diset menjadi nameField.

8. Drag satu komponen Button ke sebelah kanan komponen Text Field, ketikkan Say Hello, dan tekan Enter.

- **Catatan Penting:** Telah ditemukan bahwa ada masalah berkaitan dengan property width dari komponen Button pada JSF 1.2 jika dijalankan di IE7. Untuk mengatasinya, dengan meletakkan komponen Button di sebuah komponen layout (Grid Panel, Group Panel, atau Layout Panel). Mengubah

ukuran dari komponen layout secara otomatis akan mengubah ukuran komponen Button.

9. Di window Properties, ubahlah property id komponen Button dari `button1` menjadi `helloButton`.
10. Drag komponen Static Text ke daerah di bawah komponen Label.
11. Ubah property id komponen Static Text dari `staticText1` menjadi `helloText`.
12. Drag satu komponen Message Group di bawah komponen Static Text.

Menambahkan komponen Message Group, yang bisa dipakai untuk menampilkan runtime error akan berguna untuk mendiagnosa kesalahan pemrograman.

13. Di toolbar editing, klik JSP untuk mengubah tampilan desain visual editor menjadi mode tampilan editor source JavaServer Pages (JSP).

Perhatikan kode yang ditampilkan editor pada mode source JSP ini. Perhatikan bahwa perubahan yang anda lakukan melalui window properties berdampak pada perubahan kode source JSP. Ketika satu halaman pertama kali ditampilkan di browser, halaman itu akan ditampilkan sesuai dengan tag-tag di halaman JSP. Ketika bean halaman anda mempunyai kode yang mengubah nilai dari property-property, perubahan seperti itu hanya akan terjadi pada request di mana halaman itu disubmit dan kemudian ditampilkan lagi.

Menambahkan Perilaku

Di bagian ini anda akan menambahkan program untuk membuat halaman akan ditampilkan kembali dengan pesan `Hello nama-yang-dimasukkan`. Anda melakukan ini dengan cara menambahkan event handler yang dipanggil aplikasi ketika button diklik. Event handler ini mengeset property `text` dari komponen Static Text menjadi “pesan hello” dan kemudian menyebabkan halaman itu ditampilkan kembali dengan text pesan `hello` ditampilkan.

1. Di toolbar editing, klik Design untuk berubah ke Visual Designer.
2. Klik dua kali komponen Button.

Editing Area berubah menjadi Editor Java dan menampilkan bean dari halaman Page1. Event handler dari button, `helloButton_action`, telah ditambahkan pada bean dari halaman Page1.

3. Gantilah badan dari method `helloButton_action` dengan beberapa baris kode berikut (bagian yang dicetak **tebal**). Dan kemudian tekan Alt-Shift-F untuk memformat ulang kode itu.

Contoh kode 1: Kode `helloButton_action()`

```
public String helloButton_action() {  
    String name = (String)nameField.getText();  
    helloText.setText("Hello, " + name + "!");  
    return null;  
}
```

Baris pertama yang dicetak **tebal** mengambil nilai dari property `text` komponen `TextField` `nameField` menggunakan method `getText`. Nilai itu adalah sebuah object dengan type `Object`, sedangkan yang diinginkan adalah string, sehingga dalam kode itu diubah (cast) menjadi object `String`. Dan kemudian object itu diberikan ke variabel `name`.


Baris kedua yang dicetak **tebal** mengeset nilai dari property `text` komponen `Static Text` `helloText`. Nilai ini berisi nama yang dimasukkan user ke komponen `Text Field` `nameField`. Sebagai contoh, jika user memasukkan `Fred`, baris ini mengeset property `text` dari komponen `Static Text` menjadi `Hello, Fred!`

Menjalankan Aplikasi

1. Di Editor Java, pastikan bahwa tidak terdapat error pada kode Java anda.

Jika terdapat error akan ditandai dengan garis bawah merah atau kotak-kotak merah di sebelah kiri kode. Project anda tidak akan mem-build jika masih terdapat error pada kode.

Tip: Letakkan cursor di atas kotak merah di sebelah kiri kode untuk melihat deskripsi dari error.

2. Klik Button `Run Main Project` . 

IDE akan menyimpan file-file anda dan mem-build project anda. Window output build akan tampak di bagian bawah screen dan menampilkan pesan-pesan tentang status yang relevan. ketika aplikasi anda sudah di-build dan dideploy. IDE akan membuka browser web default anda (jika belum terbuka) dan aplikasi web anda akan muncul.

3. Masukkan nama anda di text field dan klik `Say Hello`.

"Hello *nama anda*" akan ditampilkan.

Browser men-submit form ke web server, yang akan memanggil aplikasi web anda. Aplikasi akan mengeksekusi method `button action`, mengupdate elemen-elemen halaman, merender ulang halaman yang sama dengan data yang telah berubah dan mengirim kembali halaman itu ke browser.



Gambar 3: Hasil Hello Web

Mengganti Text Field dengan Drop Down List

Bagian selanjutnya dari tutorial ini akan menunjukkan bagaimana menggunakan komponen Drop Down List sebagai ganti dari Text Field untuk mendapatkan masukan user, seperti ditampilkan dalam gambar berikut. Komponen Drop Down List ini mendapatkan daftar pilihannya dari tabel `PERSON` yang terdapat di database contoh yang telah diinstal otomatis saat menginstal NetBeans.

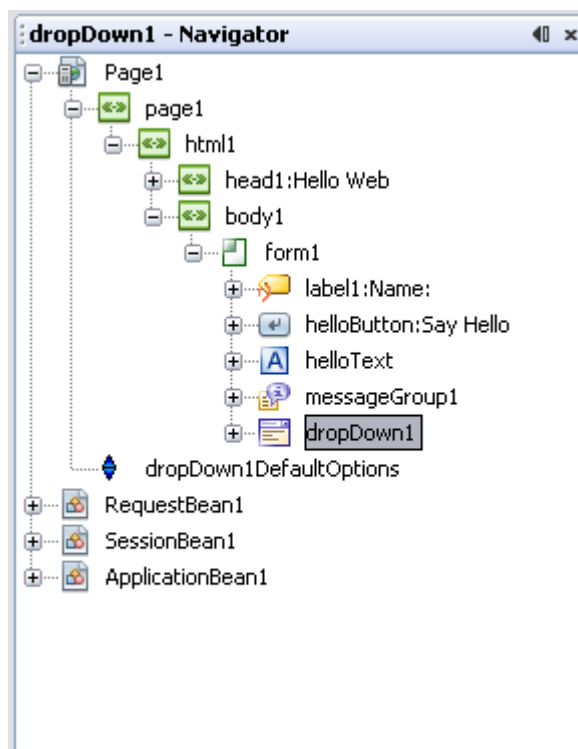


Gambar 4: Hello Web, versi akhir

1. Di toolbar Editing, klik Design untuk berpindah ke Visual Designer.
2. Di Visual Designer, klik kanan komponen Text Field `nameField` dan pilih Delete dari menu pop-up.
3. Drag satu komponen Drop Down List dari Palette bagian Basic ke halaman anda di Visual Designer. Pindahkan komponen itu ke tempat di mana sebelumnya ada Text

Field.

Perhatikan, seperti ditampilkan di gambar berikut bahwa window Navigator memperlihatkan komponen `dropDown1` dan object `dropDown1DefaultOptions`. Property `items` komponen Drop Down List mengidentifikasi object yang berisi pilihan di list. Ketika anda menambahkan satu komponen Drop Down List ke halaman, IDE membuat object Default Options (`dropDown1DefaultOptions`) dan mengeset object ini sebagai nilai dari property `items` komponen Drop Down List. Hanya komponen Drop Down List yang terlihat di Visual Designer, sedangkan Object Default Options hanya menyuplai pilihan yang terlihat di list. Di bagian berikutnya dari tutorial ini, anda akan memodifikasi komponen Drop Down List untuk mendapatkan pilihannya dari sumber yang berbeda-beda.



Gambar 5: Komponen-komponen di Window Navigator

4. Di window Properties, ubah property `id` komponen Drop Down List menjadi `nameDropDown`.
5. Pilih komponen Label dan **Ctrl-Shift-Drag** dari komponen Label ke komponen Drop down List.

Perhatikan bahwa property `for` dari komponen Label sekarang menjadi `nameDropDown`.

Mengaitkan Drop Down List ke Tabel Database

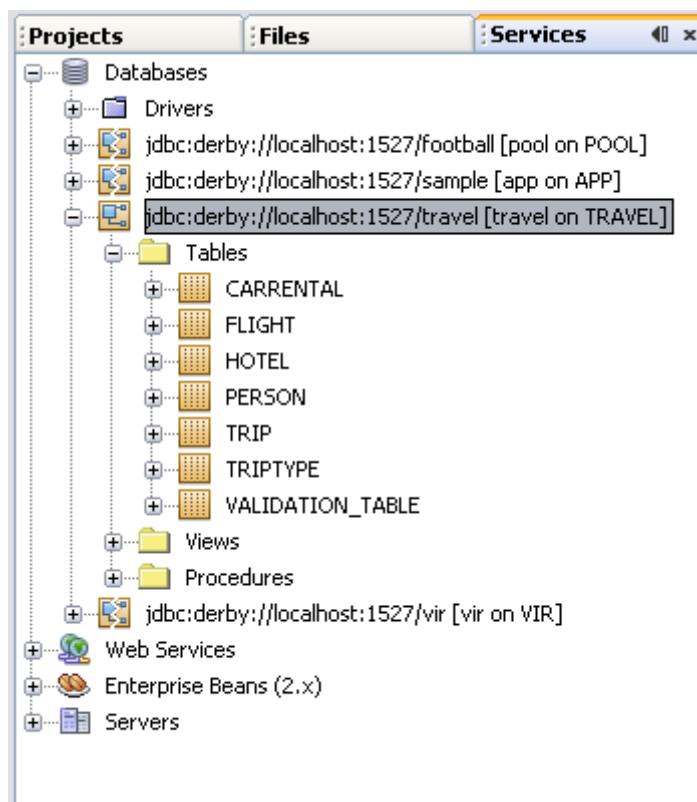
Di window Services, yang terlihat di sebelah kiri workspace IDE, terdapat node Databases.

Node Databases memperlihatkan semua driver Database dan koneksi yang telah ditambahkan ke IDE. IDE NetBeans dilengkapi dengan Database contoh yaitu database Travel. Database Travel terlihat di bawah node Databases. Pada tutorial bagian ini anda akan menggunakan tabel PERSON dari database Travel untuk mensupply pilihan untuk komponen Drop Down List.

1. Pada window Services, ekspansikanlah node Databases dan periksa apakah database Travel terhubung. Jika ikon badge untuk node jdbc database TRAVEL terlihat pecah dan anda tidak bisa mengekspansikan node, artinya IDE belum terhubung dengan database. Untuk menghubungkan database TRAVEL, klik kanan di node jdbc untuk database TRAVEL dan pilih Connect dari menu pop-up. Jika diminta, masukkan travel sebagai password, pilih Remember Password During This Session, dan klik OK. Jika anda tidak melihat node jdbc untuk database TRAVEL, pelajari petunjuk instalasi NetBeans (NetBeans Installation Instructions) untuk mendapatkan informasi tentang bagaimana membuat database tersedia untuk IDE.

2. Ekspansikan node Tables database TRAVEL.

Di bawah node Tables, anda bisa melihat setiap tabel yang ada di database, seperti CARRENTAL dan FLIGHT. Gambar berikut memperlihatkan window Services dengan node Tables terekspan.



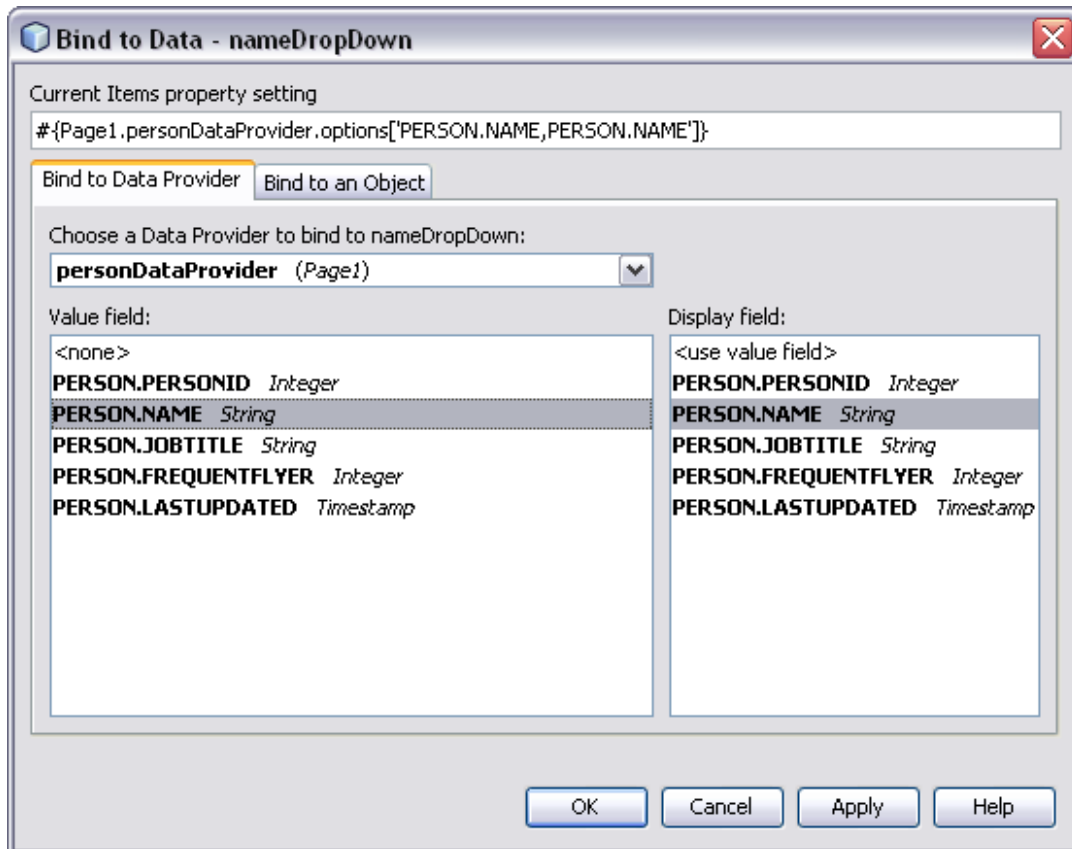
Gambar 6: Window Services

Drag PERSON dari window Services dan drop pada Drop Down List.

Tampilan pada list berubah dari item 1 menjadi abc, mengindikasikan bahwa list sedang menampilkan data yang dikaitkan dan data yang ditampilkan adalah data bertipe String.

IDE menambahkan satu komponen nonvisual `personDataProvider` untuk tabel database `PERSON`. Komponen `personDataProvider` terlihat di window Navigator. IDE juga menambahkan satu property `personRowSet` pada `SessionBean1`.

3. Klik kanan Drop Down List dan pilih Bind to Data dari menu pop-up. Kotak dialog Bind to Data akan ditampilkan, seperti ditunjukkan pada gambar berikut.



Gambar 7: Mengaitkan data ke Drop Down List

Ketika anda mengaitkan data ke komponen Drop Down List, anda harus menetapkan apa yang akan tampil di list (Display Field) dan anda juga harus menetapkan nilai yang mana yang akan digunakan sebagai penentu keputusan pada program yang dibuat (Value Field). Umumnya anda akan menampilkan nilai yang deskriptif dari tabel database seperti nama orang, tapi anda akan lebih memilih menggunakan identifier unik untuk keperluan program, seperti ID dari seseorang. Walaupun demikian, pada contoh aplikasi ini anda akan mengaitkan kedua field, Value field dan Display field dengan kolom yang sama dari database, yaitu kolom `PERSON.NAME`, seperti dijelaskan dalam dua langkah berikut.

4. Set Value field di kotak dialog menjadi `PERSON.NAME`, biarkan Display field diset ke `PERSON.NAME`, dan klik OK.

Menambahkan Beberapa Perilaku

1. Di Visual Designer, klik dua kali komponen Button.

Editing Area berubah menjadi Editor Java dan mengarahkan anda ke method

helloButton_action.

2. Gantilah badan dari method `helloButton_action` dengan kode berikut (ditunjukkan dengan dicetak **tebal**).

Contoh Kode 2: Kode pengganti `helloButton_action`

```
public String helloButton_action() {  
    String name = (String)nameDropDown.getSelected();  
    String splitnames[] = name.split(",");  
    helloText.setText("Hello, " + splitnames[1] + "!");  
    return null;  
}
```

Baris pertama menggunakan method `getSelected` untuk mendapatkan nilai saat ini dari Drop Down List, yang tidak lain adalah nama yang dipilih user di list.

Karena data di simpan di database dalam bentuk *lastname*, *firstname*, string dari database harus dimodifikasi terlebih dahulu sebelum ditampilkan. Jika tidak, aplikasi akan menampilkan "Hello, *lastname*, *firstname*!" Baris kedua menggunakan method `split` untuk men-split string menjadi array, menggunakan koma sebagai penanda pemisah. Item pertama di array (di posisi 0) berisi nama belakang, dan posisi 1 berisi nama depan.

Di baris ketiga, property `text` untuk komponen Static Text component di set ke nilai yang menyertakan nama depan.


Catatan: method ini mengasumsikan bahwa semua nilai di tabel berformat *lastname*, *firstname*. Kode ini tidak menghandle dengan benar string yang tidak mengikuti format tersebut.

3. Tambahkan kode berikut ke method `prerender`. Kode ini mengeset item pertama di list sebagai pilihan default.

Contoh Kode 3: Kode Method `prerender`

```
public void prerender() {  
    // If no selection, set default selection  
    if (nameDropDown.getSelected() == null) {  
        personDataProvider.cursorFirst();  
        nameDropDown.setSelected  
            ((String)personDataProvider.getValue("person.name"));  
    }  
}
```

Menjalankan Aplikasi

1. Klik button Run Main Project . 

IDE akan mem-build dan mendeploy aplikasi dan menampilkan halaman pada web browser.

2. Pilih satu nama dari list dan klik Say Hello.

Browser mengirimkan nilai terpilih dari komponen Drop Down List ke server, dan server mengeksekusi method `helloButton_action` yang dimiliki button.

Pengembangan

Cobalah. Komponen Listbox serupa dengan komponen Drop Down List. Cobalah mengganti komponen Drop Down List dengan komponen Listbox. Di aplikasi ini, property `multiple` komponen Listbox harus tidak dicek, karena hanya satu item yang dapat dipilih dalam satu waktu. Jangan lupa untuk mengaitkan Listbox ke tabel database dan mengubah method `helloButton_action` untuk mendapatkan nilai terpilih dari Listbox.

Cobalah. Menggunakan langkah-langkah yang serupa dengan yang anda pelajari di tutorial ini, cobalah membangun aplikasi web yang mempunyai satu komponen Drop Down List yang memperlihatkan semua nilai `DESCRIPTION` di tabel `TRIPTYPE`. Ketika pengguna meng-klik button `Show Type Id`, buatlah halaman menampilkan tipe dari perjalanan(trip) `TRIPTYPEID`. Untuk melakukan ini, anda harus mengaitkan `Display field` komponen Drop Down List ke `TRIPTYPE.DESCRPTION` dan `Value field` Drop Down List ke `TRIPTYPE.TRIPTYPEID`.

Ringkasan

Secara umum workflow untuk mendesain suatu halaman web menggunakan IDE NetBeans terdiri dari langkah-langkah berikut :

1. Membuat suatu halaman.
2. Meletakkan komponen, seperti komponen Text field dan komponen button pada halaman.
3. Mengedit property-property dari komponen untuk mengubah penampilan dan perilakunya.
4. Mengaitkan komponen ke koneksi datanya, jika memang ada.
5. Mengedit kode Java untuk mengatur perilaku server-side, seperti penanganan event(event handling).
6. Mem-build dan mengetes aplikasi anda.

Biografi Penulis



Yudi Tri Wibowo. Menyelesaikan S1 di Institut Teknologi Bandung jurusan Matematika pada tahun 2001. Setelah lulus S1 sampai tahun 2007 bekerja sebagai Dosen di Politeknik Caltex Riau Pekanbaru. Periode 2004-2005 menempuh dan menyelesaikan S2 di Magister Manajemen Teknologi Informasi Institut Teknologi Sepuluh Nopember Surabaya (ITS). Periode 2005-2007 menjabat sebagai Ketua Jurusan Teknik Komputer, Teknik Informatika dan Sistem Informasi di Politeknik Caltex Riau. Sejak tahun 2007 hingga sekarang bekerja di PT. Sigma Karya Sempurna (Balicamp) sebagai Project Manager untuk proyek-proyek pengembangan software.