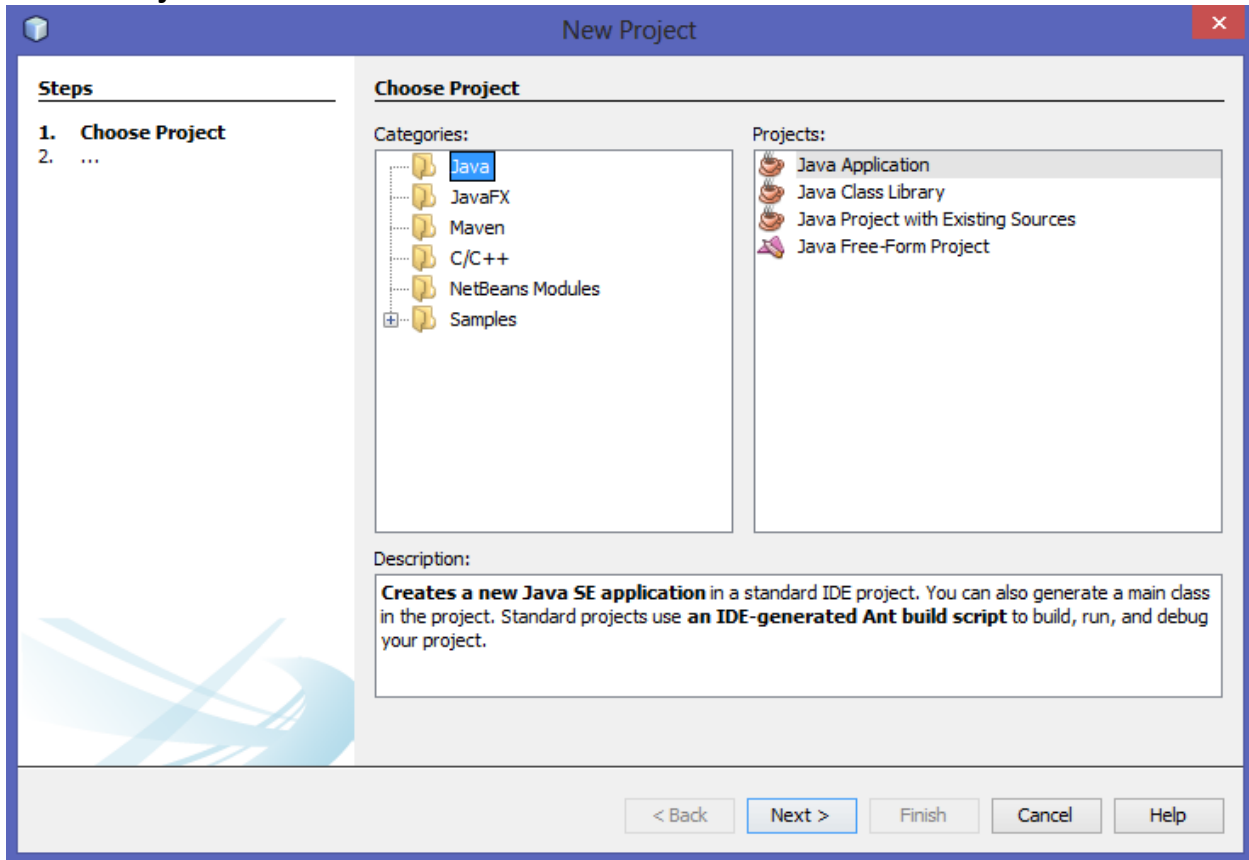


JOBSHEET NETBEANS

Jendela – jendela IDE NETBEANS



Gambar 1 : Tampilan IDE NetBeans ketika pertama kali dijalankan

Daftar Jendela – jendela pokok dari NetBeans

Jendela	Keterangan
Project	Jendela terpenting yang berisi daftar dari semua kandungan proyek, merupakan logical-view dari isi proyek, seperti daftar packet dan web page. Klik kanan pada sebuah node proyek akan menampilkan menu popup yang berisi operasi – operasi yang dapat dilakukan terhadap proyek
Files	Jendela ini menampilkan daftar struktural dari file - file dan folder yang tidak Nampak dalam jendela Project. Dari jendela ini sebuah file sumber dapat dibuka untuk diedit. File sumber dapat berupa file Java, file HTM, file Konfigurasi proyek. Selain file sumber , juga didaftar file – file output hasil proses build. Misalnya file “.class”, JAR,WAR, file Javadoc.
Navigator	Jendela ini menampilkan informasi mengenai file – file sumber yang berada dalam proyek yang sedang aktif. Ditampilkan elemen – elemen file dalam bentuk daftar atau bentuk pohon (inheritance-tree)
Source Editor	Editor teks untuk mengedit file – file sumber. Editor ini dapat dibuka dengan klik ganda pada sebuah node dalam jendela project.
Palette	Jendela Palette menampilkan daftar semua komponen yang terinstal di dalam NetBeans. Misalnya komponen yang berasal dari packet Swing, AWT, BEANS. Juga daftar layout yang dapat dipakai untuk mengatur format susunan komponen. Anda juga dapat menambahkan komponen atau paket sendiri
Inspector	Jendela Inspector menunjukkan struktur dari form. Daftar layout dan komponen yang ada dalam form ditampilkan dalam bentuk pohon (inheritance-tree)
Properties	Menampilkan daftar property dan tingkah laku (behavior) dari sebuah komponen,

	layout, atau form.
Debugging	NetBeans memiliki JDBA debugger. Jendela Debugger menampilkan informasi runtime dari program Java yang sedang dieksekusi. Kita juga dapat mengawasi, mendefinisikan atau melakukan breakpoint, watch, step-through, variable-examination, class examination, serta callstack, threads dan session.
Favorites	Jendela ini dapat dipakai untuk memudahkan pengaksesan sebuah lokasi dalam system computer, seperti file –file dan direktori yang berada di luar proyek.
Output	Jendela ini akan menampilkan output dari program jika ada. Misalnya program menuliskan teks ke standar output-stream. Maka tampilannya dapat dilihat di Jendela ini. Selain itu proses dari hasil build (atau kompilasi) juga akan ditampilkan disini. Kita dapat melihat pesan keberhasilan atau pesan error di sini.

Operasi Terhadap Proyek

Ketika membuat sebuah proyek, ada banyak fase yang dapat atau harus dilakukan. Mulai dari membuat sebuah proyek baru sampai dengan melakukan deployment. Operasi untuk melakukan fase atau menjalankan fitur tertentu memiliki langkah-langkah tertentu.

Daftar operasi terhadap proyek dan langkah-langkahnya.

Operasi	Langkah - langkah
Membuat Proyek	<ol style="list-style-type: none"> 1. Jalankan File New Project (Ctrl+Shift+N) 2. Pilih salah satu template 3. Ikuti wizard yang akan ditampilkan
Mengubah properti dari proyek	<ol style="list-style-type: none"> 1. Di dalam Jendela Project, lakukan klik kanan di atas node dari proyek. Pilih Properties. 2. Akan ditampilkan dialog Project Properties 3. Ubah property yang diinginkan
Menambahkan proyek-proyek lainnya sebagai library dari proyek yang aktif. (Compilation Dependencies)	<ol style="list-style-type: none"> 1. Di dalam jendela Project, klik kanan di atas node dari proyek yang aktif (biasanya main project). 2. Pilihlah Properties dari menu kontekstual. 3. Di dalam dialog Project Properties, pilihlah Categories="Libraries". 4. Di jendela sebelah kanan pilihlah tab Compile 5. Tekan tombol Add Project 6. Pilihlah Direktori lokasi file JAR dari proyek yang akan ditambahkan 7. Tekan tombol Add Project JAR Files.
Membuka Proyek yang diperlukan	<ol style="list-style-type: none"> 1. Di dalam Jendela Projects, klik kanan di atas node proyek. 2. Pilihlah menu open Required Projects
Melakukan build terhadap proyek	Jalankan menu Run Build Main Project. Atau tekan shortcut-key F11. Atau klik kanan di atas node proyek dan pilih menu Build.
Membersihkan (clean) proyek	Di dalam jendela Projects, klik kanan di atas node proyek dan pilih menu Clean.
Melakukan clean dan build terhadap proyek	Di dalam jendela Projects, klik kanan di atas node proyek dan pilih menu Clean and Build. Atau tekan shift-F11. Atau jalankan menu Run Clean and Build Main Project
Menjalankan (run) proyek	Jalankan menu Run Run Main Project. Atau tekan shortcut-key F6. Atau klik kanan di atas node proyek dan pilih menu Run.
Melakukan debug terhadap proyek	Jalankan menu Debug Debug Main Project. Atau tekan shortcut-key F5. Atau, klik kanan di atas node proyek dan pilih menu Debug
Melakukan kompilasi terhadap sebuah file	Jalankan menu Run Compile File . Atau tekan F9. Atau, klik kanan pada node file lalu pilihlah menu Compile File
Menambahkan file sumber JAR pada classpath dari proyek.	<ol style="list-style-type: none"> 1. Jalankan menu Tools Libraries. Akan tampil kotak dialog Library Manager. 2. Jika file JAR yang dimaksud belum terdaftar di dalam Library Manager, tekanlah tombol New Library untuk membuat library baru. 3. Isikan nama library.

	<ol style="list-style-type: none"> 4. Pilihlah library baru pada pane sebelah kiri (pane Libraries). 5. Dalam pane sebelah kanan, pilihlah tab Classpath. 6. Tekan tombol Add JAR/Folder. 7. Tentukan lokasi direktori dari file JAR yang berisi file kelas yang diharapkan. 8. Buka tab Sources dalam pane sebelah kanan. 9. Tambahkan folder atau file archive yang berisikan kode sumber (source code).
Menambahkan Javadoc untuk file JAR pada Classpath dari proyek.	<ol style="list-style-type: none"> 1. Jalankan menu Tool Libraries. Akan tampil kotak dialog Library Manager. 2. Jika file JAR yang dimaksud belum terdaftar di dalam library Manager, daftarkan file JAR tersebut dengan langkah seperti yang dijelaskan di atas. 3. Dalam pane sebelah kanan, bukalah tab Javadoc. 4. Tekan tombol Add ZIP/Folder lalu tentukan lokasi dari file – file Javadoc.
Menentukan Main Project	Dalam jendela Projects, klik kanan di atas node proyek, lalu pilih menu Set as Main Project

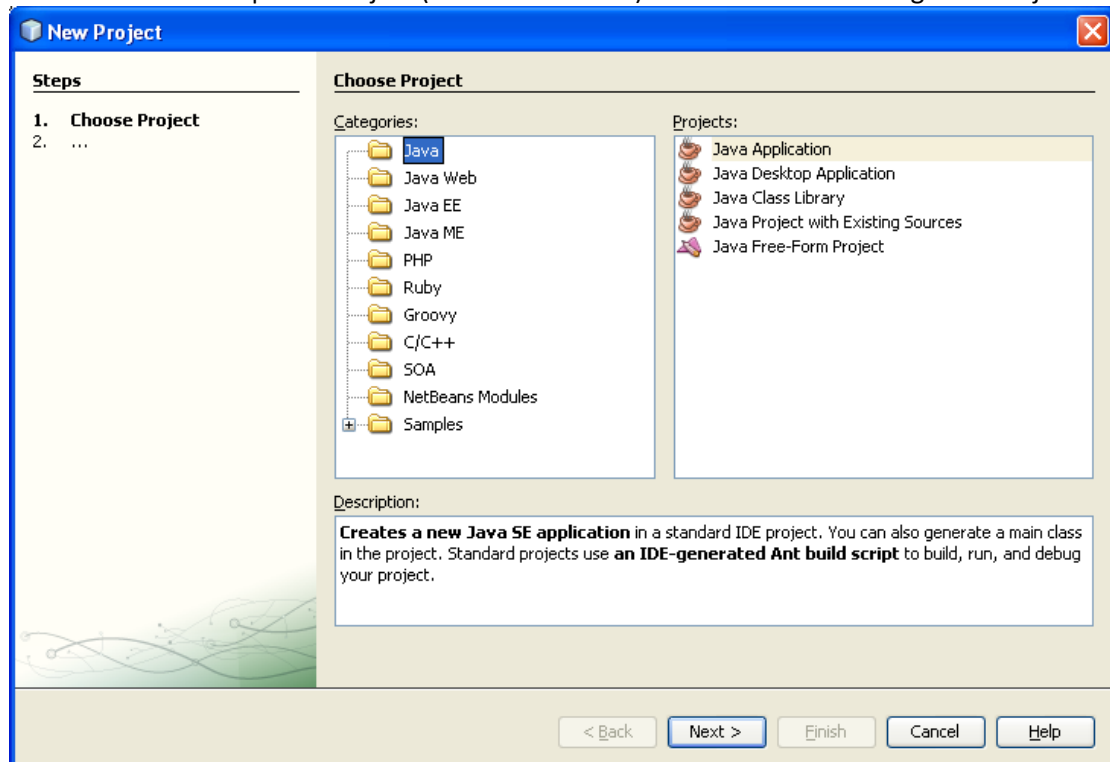
Membuat Proyek

Berikut ini kita akan membuat sebuah proyek yang berisi satu file program java sederhana. Program ini berupa aplikasi dengan tampilan sebatas teks (Program Java konsol). Tujuannya adalah untuk memperkenalkan garis besar pemrograman dengan NetBeans yang paling mendasar..

Proyek

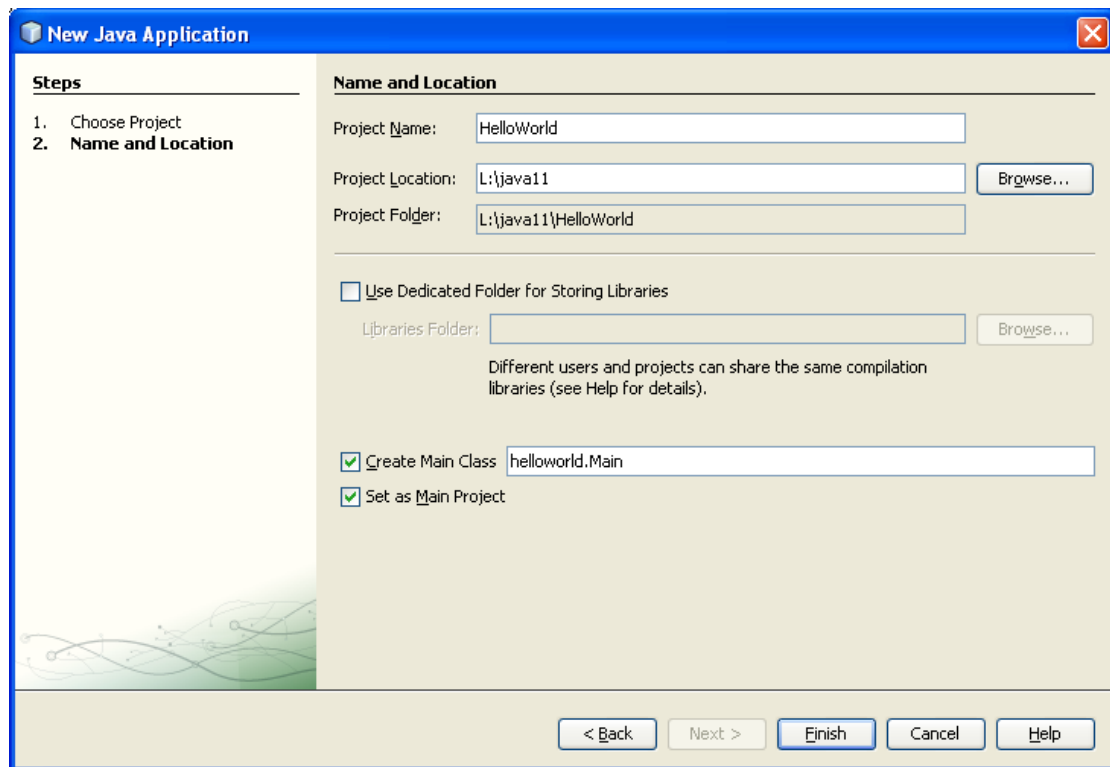
Sebuah program java dalam NetBeans selalu berada di dalam sebuah lingkup yang disebut proyek. Semua sumber program, baik berupa file, gambar, form, file web dan sebagainya berada di dalam sebuah proyek. Dengan demikian langkah pertama untuk membuat program Java adalah dengan membuat proyek terlebih dulu.

1. Jalankan menu File | New Project (atau Ctrl+Shift+N) untuk membuka dialog New Project



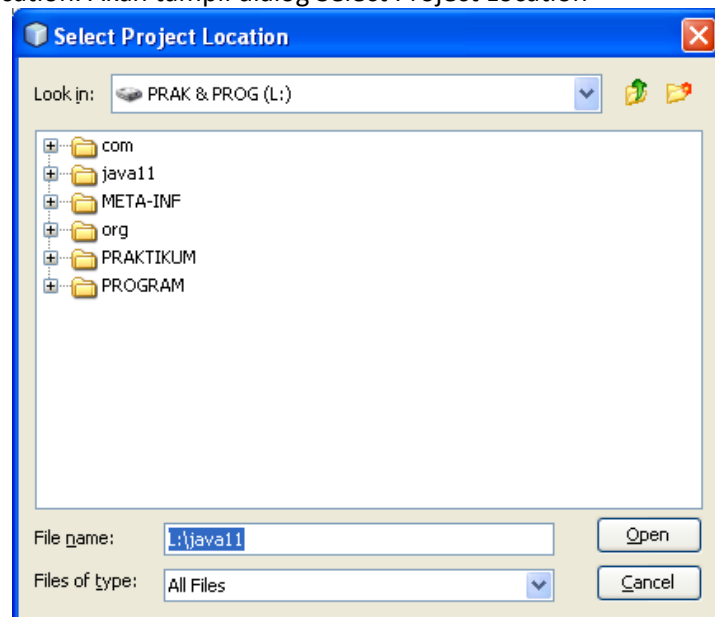
Gambar 2 : Kotak Dialog New Project step 1

2. Dalam Dialog, pilih Categories=Java, Projects=Java Application. Lalu klik pada tombol Next.



Gambar 3 : Kotak Dialog New Project step 2

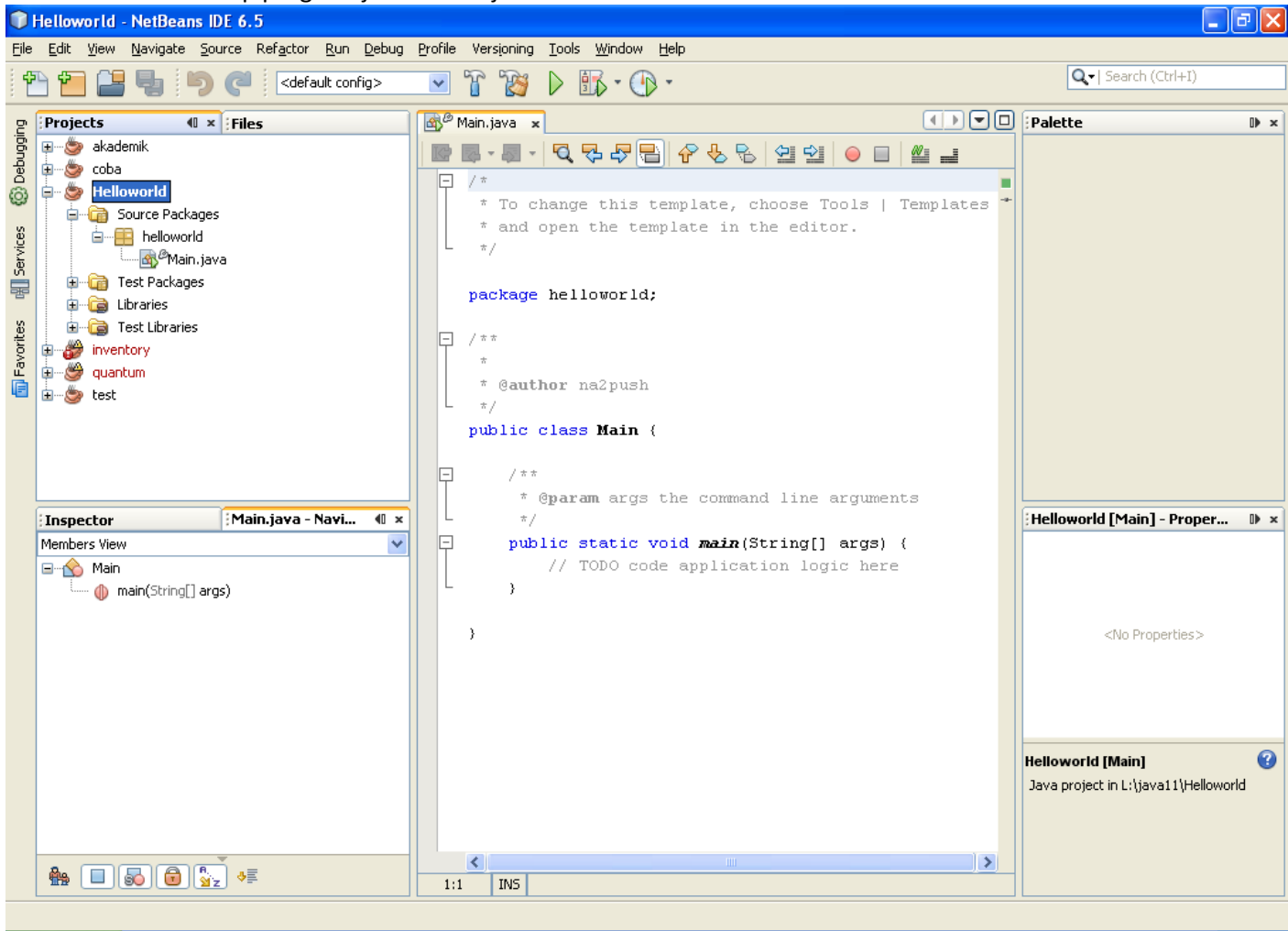
3. Pada langkah kedua, pilih terlebih dulu lokasi proyek. Klik pada tombol Browse di sebelah kanan isian Project Location. Akan tampil dialog Select Project Location



Gambar 4 : Kotak Dialog Select Project Location

4. Di dalam dialog tersebut, pilih lokasi direktori kerja default. Perhatikan bahwa direktori ini selanjutnya akan menjadi default. (misalnya dalam gambar di atas, direktori kerja default = D:\java11).
5. Setelah memilih sebuah direktori, klik tombol Open.
6. Kebali ke dialog New Project. Isikan Project Name=HelloWorld.
7. Perhatikan bahwa Project folder secara otomatis akan diset sesuai dengan nama proyek.
8. Centang pada pilihan Set as Main Project dan centang pada pilihan Create Main Class. Isikan nama main-class=helloworld.Main
9. Terakhir klik pada tombol Finish.

Di dalam NetBeans akan dibuka secara otomatis file utama java bernama Main.java. Perhatikan ekstensi dari setiap program java adalah java.



Gambar 5. Tampilan IDE setelah sebuah project baru dibuat melalui wizard.

Jendela Project dan Files

Perhatikan sebelah kiri atas dimana terdapat kelompok-kelompok jendela yang terdiri dari Project dan Files. Di dalam jendela Projects, terlihat struktur dari proyek. Struktur terdiri dari node – node. Dalam hirarki struktur proyek, yang paling atas adalah nama proyek HelloWorld. Di bawahnya terdapat beberapa node. Node yang pokok adalah Source Packages dan Libraries.

Node Source Packages berisi file – file sumber. Setiap node file sumber dapat terdiri dari node –node kelas, form dan lain –lain. Setiap node kelas dapat terdiri dari node –node anggota kelas seperti method, field, bean pattern dan sebagainya.

Node Libraries berisi daftar library yang dipakai oleh proyek. Library pada umumnya dalam bentuk JAR. Kita dapat menambahkan file baru, folder baru, atau kelas baru, ke dalam Source Packages. Kita juga dapat menambahkan library baru ke dalam Libraries.

Jendela Files menampilkan struktur dari file – file yang ada di dalam proyek. File – file ini berupa file yang dihasilkan oleh wizard, proses build, penambahan resource, dan sebagainya. Terhadap setiap node yang ada di dalam jendela – jendela tersebut, kita dapat menampilkan menu kontekstual. Menu ini dapat dibuka dengan melakukan klik kanan di atas node.

Package & Class

Node yang sering dipakai adalah Source Packages. Di dalamnya terdaftar file – file sumber yang dipakai oleh program. Setiap node file dapat memiliki node – node lagi di dalamnya secara hierarkis.

Di dalam node Source Packages, pada level paling atas terdapat daftar beberapa paket Java. Kita tadi telah mengisikan nama main-class = helloworld.Main. Ini berarti kita telah membuat sebuah paket Java bernama helloworld serta sebuah kelas Java bernama Main. Lihatlah di dalam jendela Source-Editor, yang di dalamnya dapat ditemukan deklarasi :

```
package helloworld;
public class Main {
    public Main() {
    }
    public static void Main(String[] args) {
    }
}
```

Dalam kode program terdapat beberapa blok komentar yang diawali `/**` dan diakhiri `*/`. Ini merupakan teks dokumentasi Javadoc. Perhatikan constructor `Main()` yang telah ditambahkan, serta method utama `main()`. Method `main()` hanya ditambahkan jika program merupakan aplikasi Java, seperti yang kita buat disini. Sebuah proyek bias saja terdiri lebih dari satu paket. Setiap paket dapat terdiri lebih dari satu kelas. Perhatikan aturan utama dari kelas dan file Java :

Setiap kelas Java (dengan modifier “public”) selalu disimpan dalam satu file dengan nama yang sama dengan nama kelas. Dalam contoh di atas, nama kelas adalah `Main` sehingga nama file java itu adalah `Main.java`. (Kelas dengan modifier non-public diperbolehkan berada dalam file yang sama dengan kelas public.)

Jendela Navigator

Jendela ini secara default berada di sebelah kiri bawah IDE. Klik sebuah node file di jendela Project maka akan tampil informasi mengenai file tersebut di dalam jendela Navigator. Di bagian bawah dari jendela ini terdapat beberapa tombol yang berguna untuk menentukan apa saja yang akan ditampilkan. Kita dapat memilih untuk menampilkan jenis – jenis anggota kelas : static, non-public, filed, dan anggota – anggota yang diturunkan (inherited) dari kelas induk.

Menambahkan Program

Di dalam jendela Source-Editor, tambahkan satu baris program ke dalam method `main()` sehingga program keseluruhan akan menjadi seperti berikut :

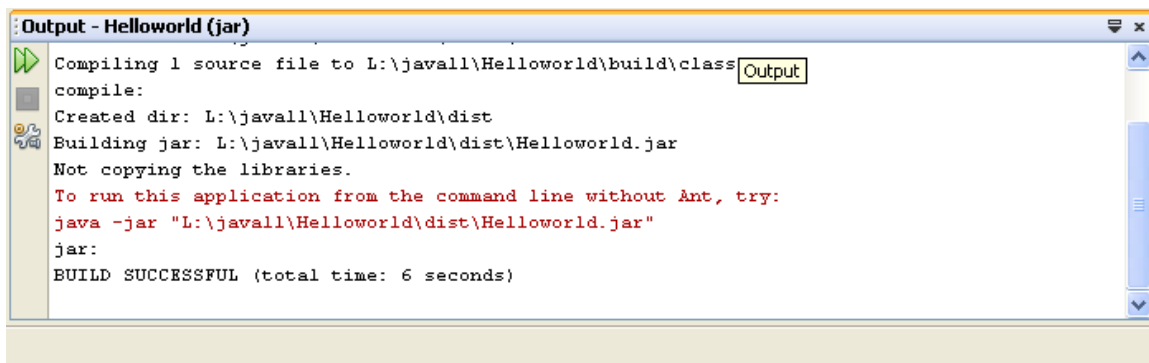
```
package helloworld;
public class Main {
    public Main() {
    }
    public static void Main(String[] args) {
        System.out.println("Hello World !!!");
    }
}
```

Setelah mengetikkan program, jangan lupa untuk menyimpannya. Jalankan menu `File | Save` atau `File | Save All` jika perubahan dilakukan terhadap lebih dari satu file. Perintah `println()` di dalam method `main()` berfungsi untuk menampilkan teks pesan. Dalam hal ini pesan akan ditampilkan dalam standar output dari system (`System.out`). Sedangkan dalam constructor `Main()` (berbeda dengan `main()` !), kita tidak menambahkan kode program apapun.

Melakukan Build terhadap Proyek

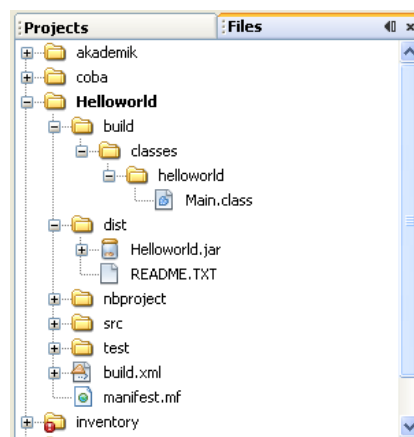
Setelah membuat program Java, maka dapat dilakukan proses build terhadap proyek. Proses Build berarti juga melakukan kompilasi terhadap file atau file – file Java yang ada di dalam proyek.

1. Jalankan menu `Run | Build Main Project`. Proses build juga dapat dilakukan di dalam Jendela Projects. Caranya , klik kanan di atas node proyek (`HelloWorld`). Akan tampil menu popup yang disebut dengan menu kontekstual. Pilih menu `Build`.
2. Perhatikan jendela Output yang ditampilkan di bagian bawah dari IDE, yang akan menampilkan pesan berupa hasil dari proses build.



Gambar 6 : Tampilan hasil proses build dalam jendela Output

Pesan “BUILD SUCCESSFUL” menandakan proses build berjalan dengan baik. Ini berarti tidak ada kesalahan dalam program. Jika ada kesalahan maka akan ditampilkan pesan error yang berarti Anda harus meneliti dan membetulkan terlebih dulu kesalahan dalam kode program. Bukalah jendela Files. Buka node build | classes | helloworld. Akan terlihat sebuah file bernama Main.class. File ini merupakan file binary Java, hasil output dari proses kompilasi terhadap file Main.java. File Main.class juga dihasilkan dalam bentuk JAR (Java Archive). Dalam contoh ini, file JAR berada di dalam folder dist bernama HelloWorld.jar



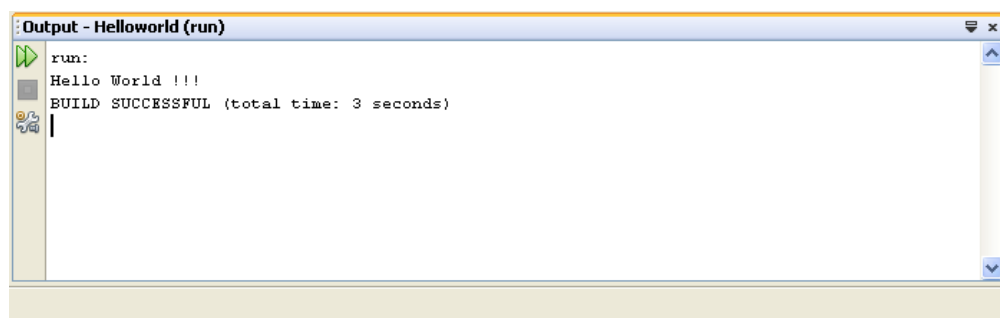
Gambar 7 : File .class hasil proses build ditampilkan dalam jendela Files

Jika Anda ingin melakukan distribusi program, maka file JAR ini dapat didistribusikan.

Menjalankan Proyek

Setelah proses build selesai, proyek atau program dapat dijalankan di dalam IDE.

1. Jalankan menu Run | Run Main Project. Dapat juga dilakukan dari dalam jendela Project. Caranya lakukan klik kanan dan pilih menu kontekstual Run.
2. Teks Output dari program akan ditampilkan di jendela Output.



Gambar 8 : Output program java konsol berupa teks ditampilkan dalam jendela Output

Perhatikan teks Hello world !!! yang ditampilkan. Teks ini berasal dari perintah yang tadi telah kita tambahkan :

```
System.out.println("Hello World !!!");
```

Keluaran dari program java konsol ini hanya sebatas teks. Sesungguhnya program java sangat luas cakupannya. Pada tingkat paling dasar, ia dapat dipakai untuk membuat program yang tampilannya sebatas teks seperti contoh ini. Namun Java dan NetBeans juga dapat dipakai untuk membuat program yang menampilkan grafik atau gambar dengan warna true-color. Bahkan dengan Library dari Java 2D dan Java 3D kita dapat memakai gambar 2 dimensi dan 3 dimensi. Bukansekedar gambar, multimedia (audio dan video) juga dapat dijalankan oleh program java.

Java dan NetBeans juga menyediakan palet lengkap yang berisi bermacam – macam komponendari paket AWT dan Swing/JFC. Paket –paket ini berisi macam –macam komponen seperti line-edit,combo-box,check-box,radio-button sampai dialog – dialog standar, seperti dialog pencetakan, dialog untuk memilih warna , dan sebagainya.

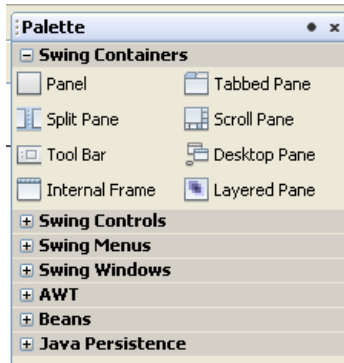
Mengenal GUI Builder

DAFTAR PALETTE

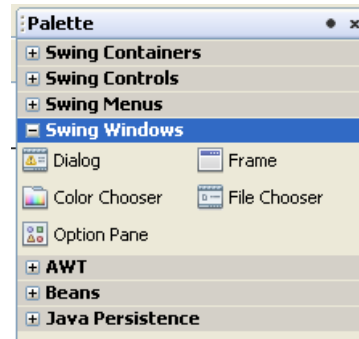
Palette	Komponen	Keterangan
Swing Container	Panel (JPanel)	Kontainer yang paling sederhana. Menyediakan space untuk meletakkan komponen ataupun panel lainnya.
	SplitPanel (JSplitPanel)	Membagi dua ruang komponen
	Tool Bar (JToolBar)	Untuk menampilkan daftar toolbar sebagai shortcut-button bagi menu – menu yang sering dipakai
	Internal Frame (JInternalFrame)	Sebuah frame yang menyediakan fungsi – fungsi dragging, closing, icon, resize, title dan mendukung menu bar. Umumnya sebuah JInternalFrame ditambahkan ke dalam sebuah JDesktopPane. Ke dalam JInternalFrame kita biasanya menambahkan komponen atau container sebagai children
	Tabbed Pane (JTabbedPane)	Mengelompokkan komponen – komponen dalam satu box, dengan judul berupa teks dan / atau icon
	Scroll Pane (JScrollPane)	Menyediakan scrollable-view dari sebuah lightweight-component
	Desktop Pane (JDesktopPane)	Kontainer yang dipakai untuk mengimplementasikan MDI (multiple document interface) atau desktop virtual. Kelas ini diturunkan dari JLayeredPane
	Layered Pane (JLayeredPane)	Merupakan kontainer yang membuat komponen dapat overlapping satu sama lain dalam bentuk layer – layer.
Swing Controls	Label (JLabel)	Untuk meletakkan teks statis, atau image atau keduanya.
	Button (JButton)	Push-button atau tombol yang jika ditekan akan mengerjakan instruksi tertentu.
	Toggle Button (JToggleButton)	Button dengan dua state (keadaan), merupakan superclass bagi JRadioButton

		dan JCheckBox
	Check Box (JCheckBox)	Tombol berbentuk kotak yang dapat di check atau uncheck untuk menyatakan keadaan on atau off.
	Radio Button (JRadioButton)	Button untuk menampilkan keadaan select atau deselect.
	Button Group (JButtonGroup)	Untuk mengelompokkan beberapa button, biasanya radio-button atau check-box
	Combo Box (JComboBox)	Daftar yang ditampilkan dropdown. Dapat juga ditambah editable-field.
	List (JList)	Menampilkan pilihan obyek dalam daftar.
	Text Field (JTextField)	Untuk mengedit satu baris teks
	Text Area (JTextArea)	Untuk menampilkan teks multi-line. Teks dapat diedit oleh user
	Scroll Bar (JScrollBar)	ScrollBar, untuk memilih nilai dalam range tertentu
	Slider (JSlider)	Memilih nilai tertentu dengan menggeser kenop di dalam range tertentu. Memiliki tick-mark pada kotaknya.
	Progress Bar (JProgressBar)	Menampilkan kemajuan dari suatu proses secara grafis.
	Formatted Field (JFormattedTextField)	Perluasan dari JTextField dengan kemampuan melakukan format terhadap teks.
	Password Field (JPasswordField)	Untuk mengedit satu baris teks berupa password. Tampilan teks akan disembunyikan.
	Spinner (JSpinner)	Untuk memilih nilai atau obyek tertentu dengan tombol up dan down.
	Separator (JSeparator)	Komponen untuk mengimplementasikan garis pembagi. Terutama dipakai untuk membagi item –item menu ke dalam kelompok –kelompok logikal.
	Text Pane (JTextPane)	Komponen teks yang dapat ditandai dengan attribute yang ditampilkan secara grafis.
	Editor Pane (JEditorPane)	Komponen teks untuk mengedit bermacam – macam content.
	Tree (JTree)	Menampilkan satu set data dalam bentuk hierarkis sebagai suatu outline.
	Table (JTable)	Menampilkan dan mengedit tabel dalam bentuk sel dua dimensi.
Swing Menus	Menu Bar (JMenuBar)	Menampilkan menu dalam frame
	Menu (JMenu)	Menampilkan menu yang ditempatkan di dalam JMenuBar
	Menu Item (JMenuItem)	Menampilkan menu item
	Menu Item / Check Box (JCheckBoxMenuItem)	Menampilkan menu item yang berupa check-box
	Menu Item / Radio Button (JRadioButtonMenuItem)	Menampilkan menu item yang berupa radio-button
	Popup Menu (JPopupMenu)	Menampilkan menu pop-up yang dapat ditampilkan dalam posisi dinamis tertentu.
Swing windows	Dialog (JDialog)	Kelas utama untuk membuat jendela custom dialog

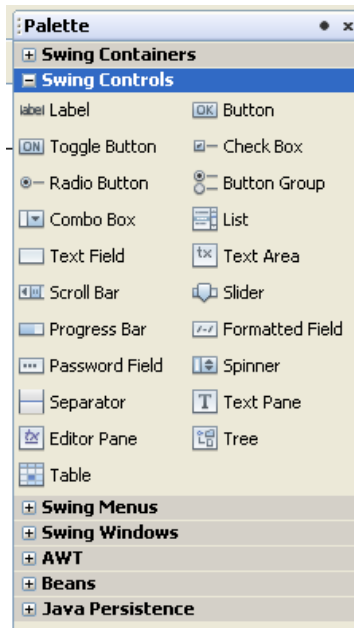
	Color Chooser (JColorChooser)	Menampilkan dialog box untuk memilih warna.
	Frame (JFrame)	Perluasan dari java.awt.Frame. Sebuah container frame yang mendukung arsitektur komponen swing.
	File Chooser (JFileChooser)	Menampilkan dialog box untuk memilih file
	OptionPane (JOptionPane)	Popup dialog box untuk menampilkan pesan atau meminta masukan dari user.
AWT	Label	Komponen untuk meletakkan teks statis di dalam container. Teks satu baris tidak dapat diedit.
	Button	Push-button atau tombol yang jika ditekan akan mengerjakan instruksi tertentu.
	Text Field	Untuk editing satu baris teks
	Text Area	Untuk menampilkan teks multi-line. Teks dapat diedit oleh user
	Choice	Menu popup yang terdiri dari beberapa pilihan.
	List	Untuk menyajikan beberapa pilihan teks dalam satu daftar yang dapat di-scroll (gulung)
	Scroll Bar	Scroll Bar, untuk memilih nilai dalam range tertentu.
	Scroll Pane	Kontainer dengan Scroll Bar horizontal dan vertical yang otomatis diset sesuai ukurannya.
	Panel	Kontainer yang paling sederhana. Menyediakan space untuk meletakkan komponen ataupun panel lainnya.
	Canvas	Kontainer yang diletakkan di dalam Panel
	Menu Bar	Menampilkan menu dalam frame. Dipakai bersama obyek Frame. Memakai setMenuBar().
	Popup Menu	Menampilkan menu pop-up yang dapat ditampilkan dalam posisi dinamis tertentu.
Beans	Choose Bean	Untuk menggunakan kelas dari Java Bean
Java Persistence	Entity Manager	
	Query	
	Query Result	



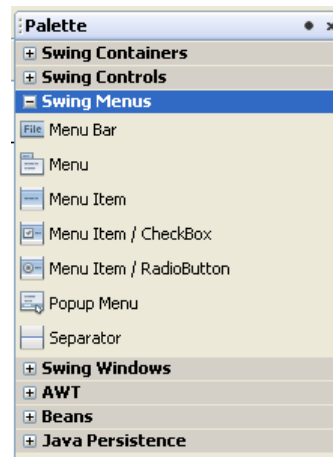
Gambar 9 : Palette Swing Containers



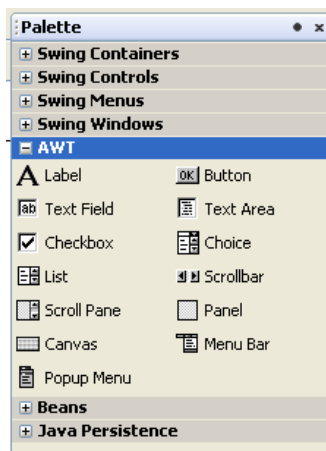
Gambar 10 : Palette Swing Windows



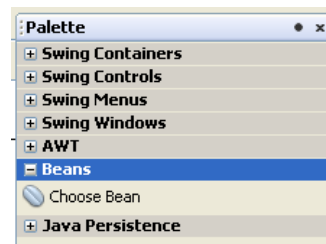
Gambar 11 : Palette Swing Controls



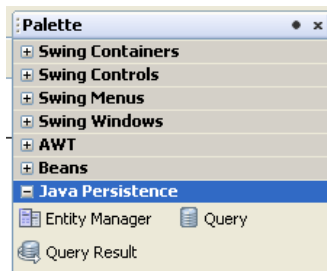
Gambar 12 : Palette Swing Menus




Gambar 13 : Palette AWT

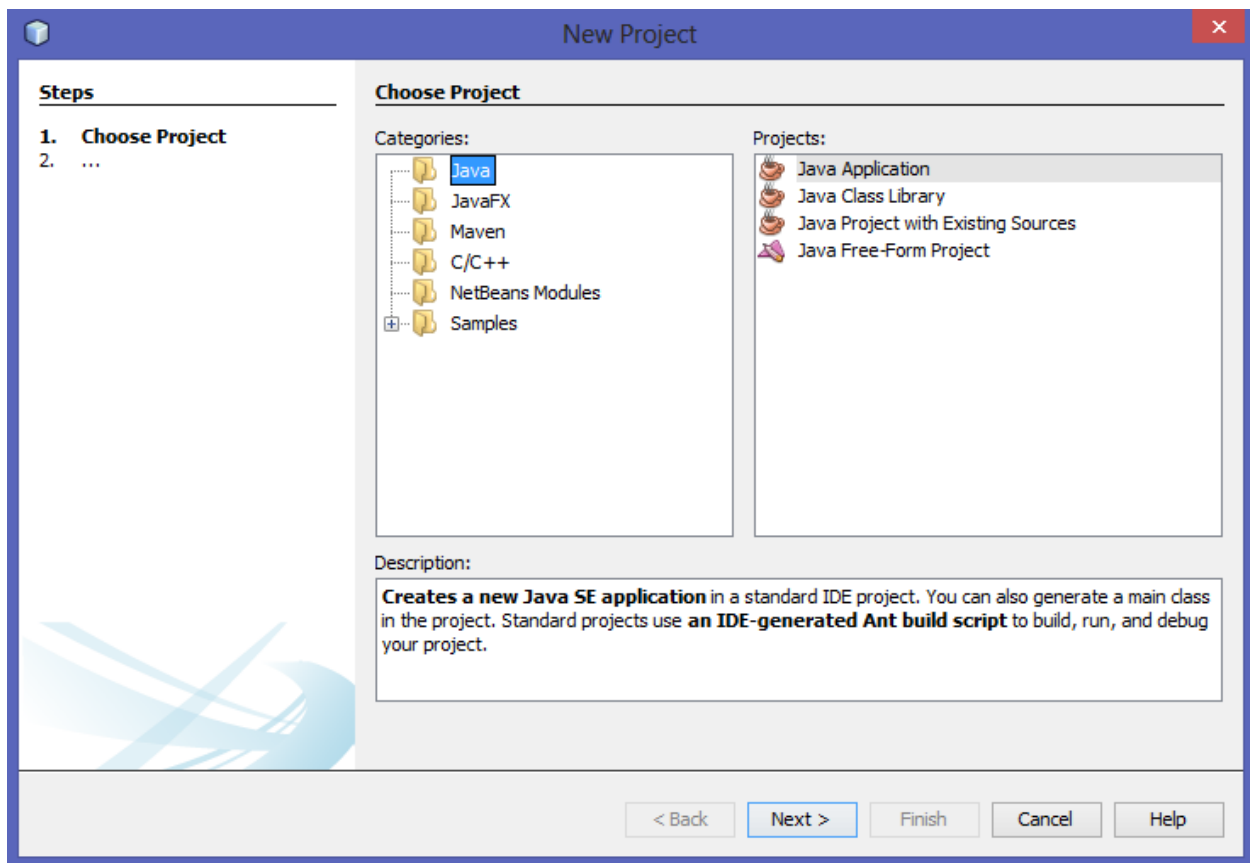


Gambar 14 : Palette Beans

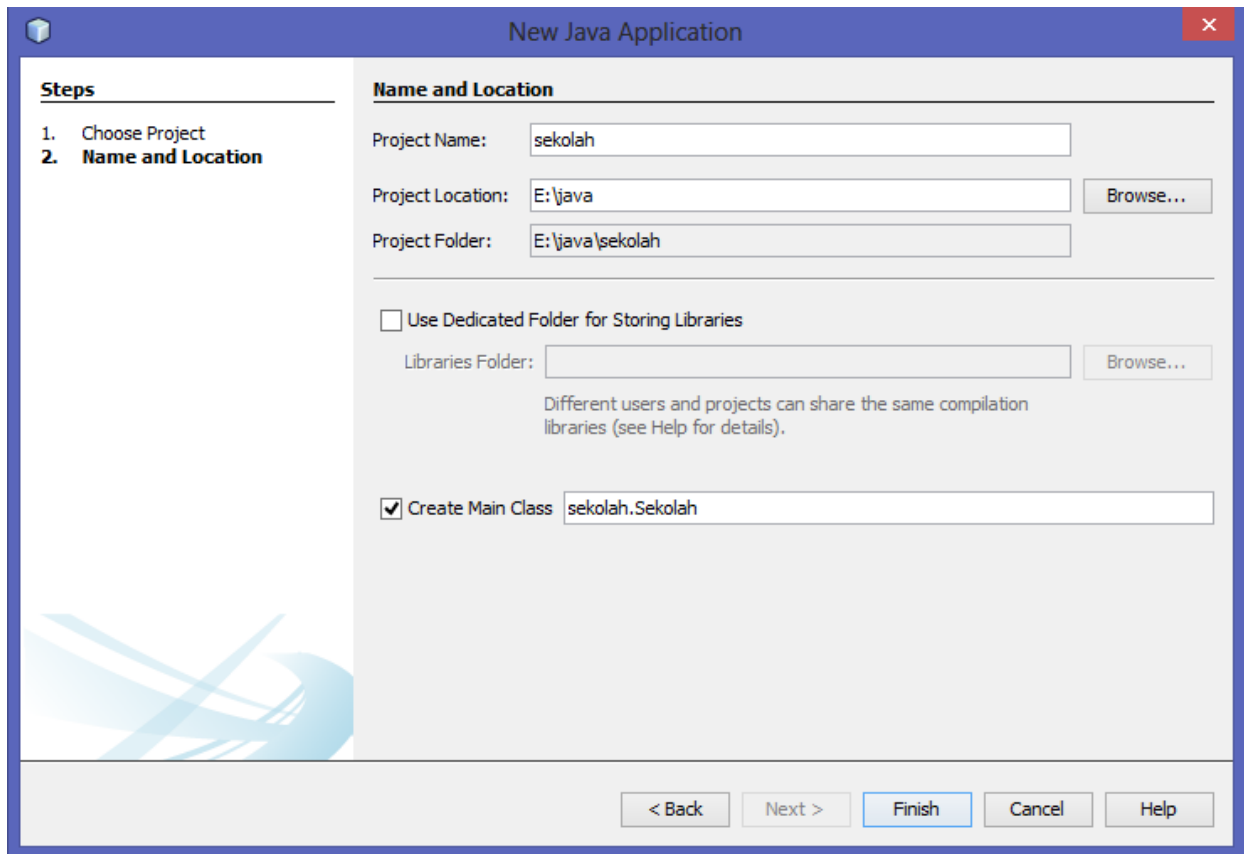


Gambar 15 : Palette Java Persistence

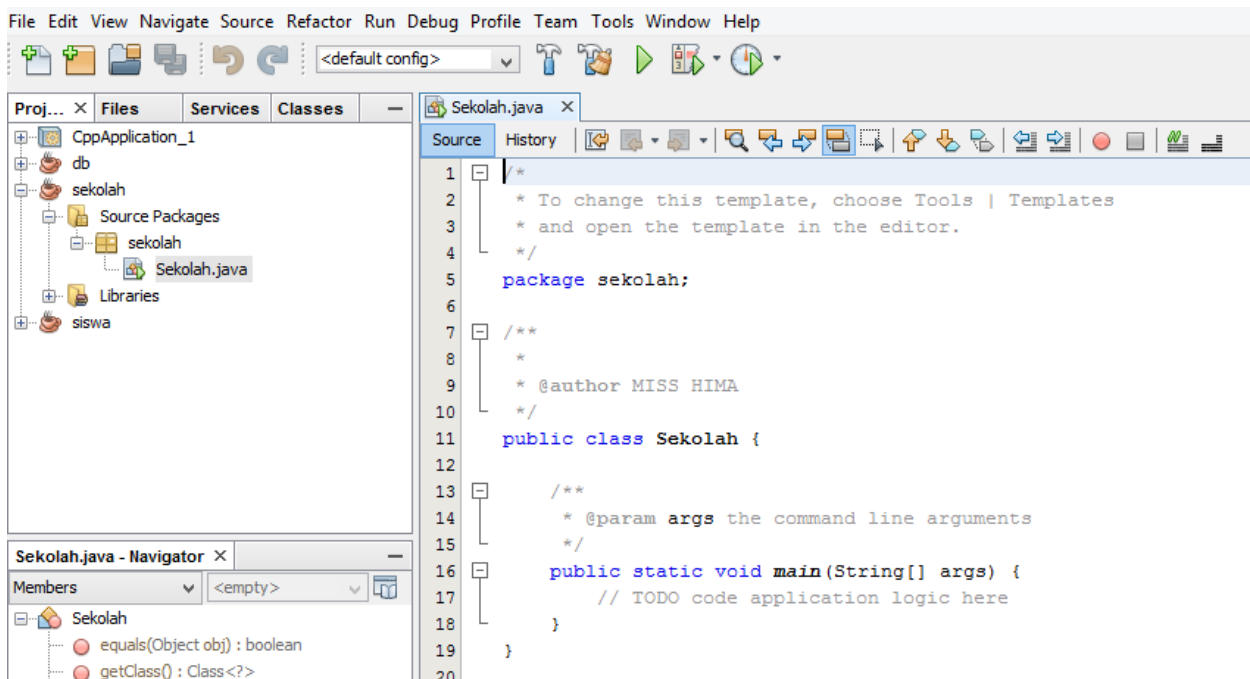
Cara membuat project pada netbeans 7.3.1 bisa dilakukan dengan memilih menu file→new project, atau dengan memilih icon , akan muncul jendela seperti dibawah ini :



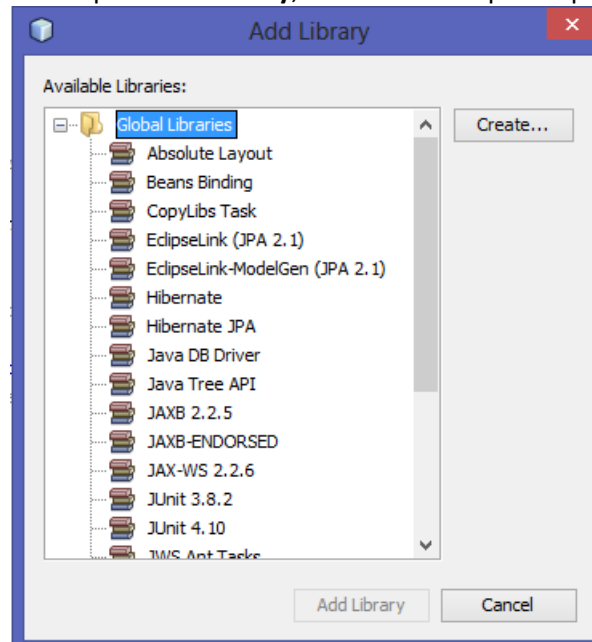
Pilih java application pada jendela projects dan klik next, setelah itu isikan 'sekolah' pada project name dan klik finish



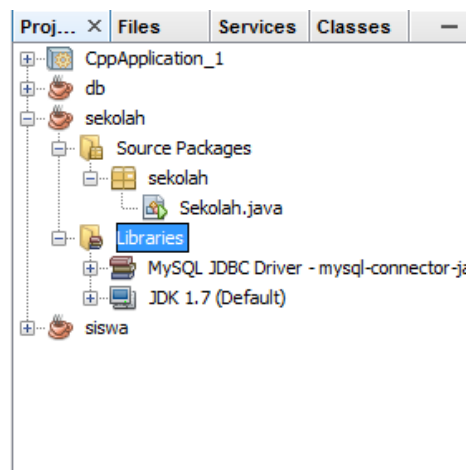
Akan muncul jendela awal seperti gambar di bawah ini



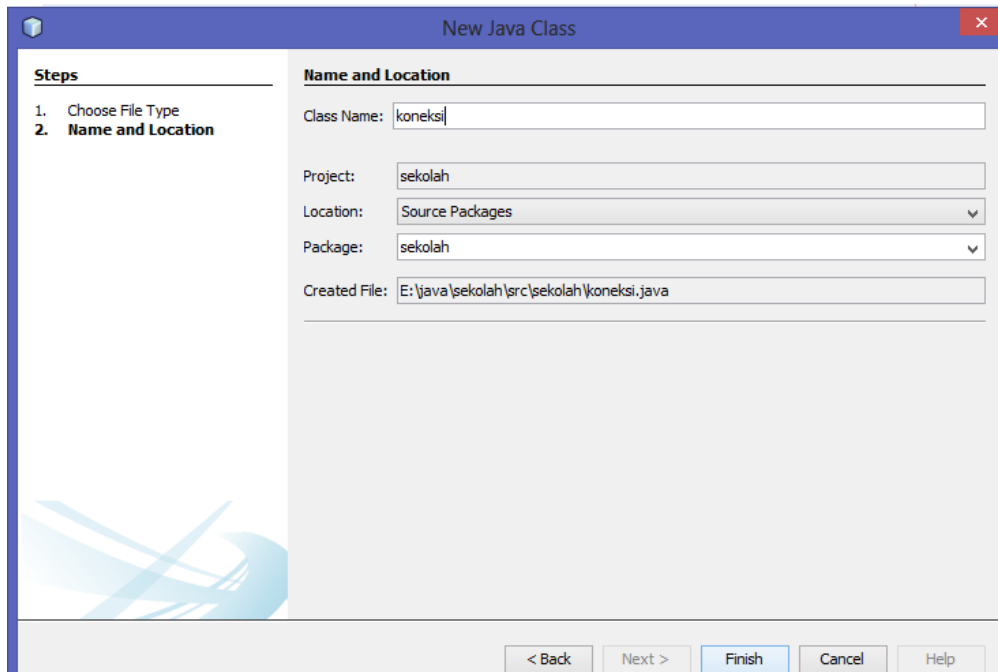
Terdapat file default bernama sekolah.java sebagai file utama dalam project sekolah dimana function main akan dijalankan pada saat project sekolah dirun/eksekusi. Sekarang kita akan membuat sebuah file untuk membuat koneksi antara java dengan mysql, sebelum itu kita harus menambahkan library **mysql jdbc driver** pada package library untuk pendukung koneksi, caranya dengan mengklik kanan folder libraries pada project sekolah dan pilih **Add Library**, maka akan tampil tampilan berikut



Lalu pilih Mysql JDBC Driver → Add Library, sampai muncul tampilan berikut pada folder libraries



Setelah itu, kita sekarang akan membuat file untuk koneksi mysql dengan java. Buatlah file baru dengan klik kanan pada folder/package sekolah, pilih **New → Java Class**. Isi nama Class Name nya dengan 'koneksi' lalu tekan finish

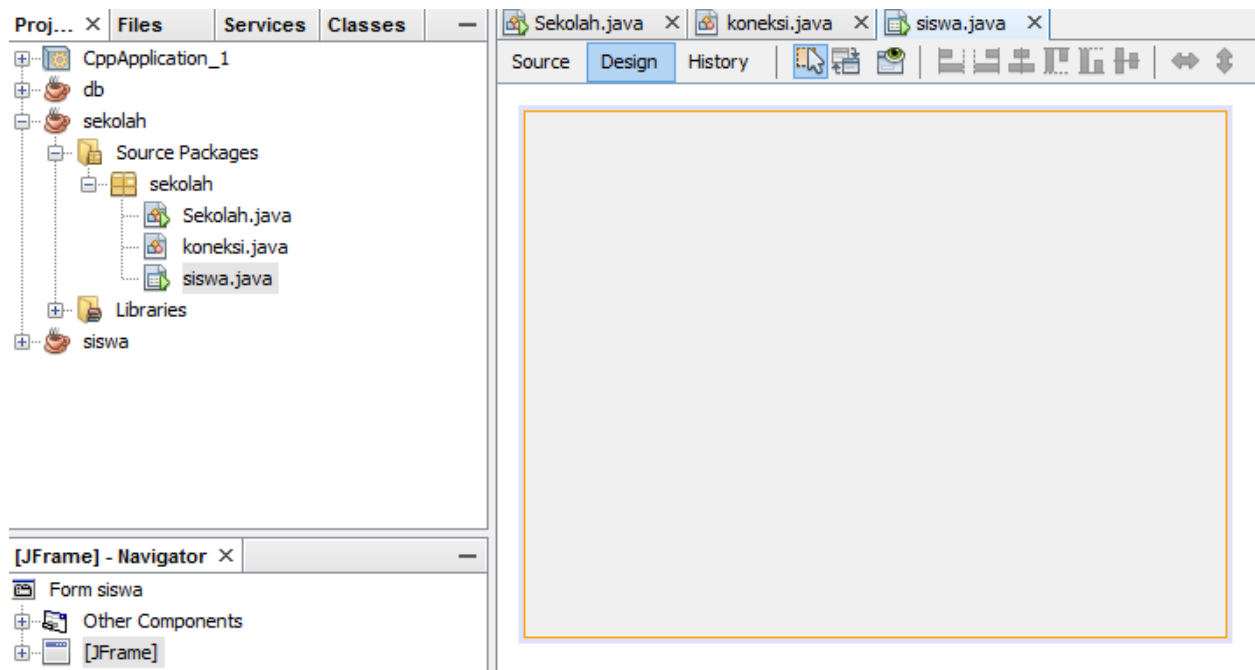


Lalu masukan coding ini pada file koneksi

```
package sekolah;
import java.sql.*;

public class koneksi {
    private static Connection koneksi;
    static String status="";
    public static Connection getKoneksi() throws SQLException{
        try{
            Class.forName("com.mysql.jdbc.Driver");
            Connection con=DriverManager.getConnection("jdbc:mysql://localhost/test","root","");
            return con;
        }
        catch(ClassNotFoundException | SQLException ex){
            return null;
        }
    }
}
```

Simpan filenya dan lakukan proses build tanpa eksekusi karena pada file ini kita tidak menaruh function main. Setelah membuat file koneksi kita akan membuat form inputan untuk siswa pada file yang berbeda, pilih New kembali pada package sekolah dan pilih **JFrame Form**, Isikan 'siswa' pada Class Namanya maka akan tampil tampilan berikut



Buatlah JLabel pada jendela swing containers sebanyak 5 buah dengan letak seperti dibawah ini



Untuk mengubah tampilan text pada JLabel bisa dilakukan dengan 2 cara, yaitu dengan mengklik kanan pada JLabel dan pilih **edit text** atau pilih **text** pada jendela properties di kanan bawah, Untuk JLabel1 edit text menjadi 'MASTER DATA SISWA' dan ubah ukuran dan jenis huruf pada properties font dengan memilih Arial, bold, ukurannya 18. Untuk JLabel2 edit text menjadi 'NIS', JLabel3=nama, JLabel4=jenis kelamin, JLabel5=alamat sehingga tampilannya jadi seperti ini

MASTER DATA SISWA

NIS

Nama

Jenis Kelamin

Alamat

Lalu tambahkan JTextField untuk masing-masing NIS, Nama dan Jenis Kelamin serta JTextArea untuk alamat. Untuk Menghilangkan tulisan JTextField1, dst lakukan edit text. Dan beri nama variable untuk masing masing textField dan textArea dengan mengklik kanan dan pilih **Change variable Name**. Beri nama untuk masing-masing textField dan textarea : **TxtNis, TxtNama, TxtJenkel, AreaAlamat**. Sehingga tampilannya akan jadi seperti berikut :

MASTER DATA SISWA

NIS

Nama

Jenis Kelamin

Alamat

^

v

<

>

Setelah itu kita akan tambahkan JTable untuk menampilkan data yang ada pada database. Letakan JTable di bawah textarea, namun sebelumnya tambah ukuran panjang JFrame untuk menampungnya, baru setelah itu dikembalikan ukuran ke bentuk semula setelah JFrame dikurangi panjangnya, sehingga tampilannya menjadi seperti berikut

MASTER DATA SISWA

NIS

Nama

Jenis Kelamin

Alamat

Title 1	Title 2	Title 3	Title 4

Lalu ubah variable name nya menjadi **jTableSiswa**, setelah itu ubah nama kolom jTable bisa degan klik kanan pada jTable pilih 'table contents' atau dengan memilih tablemodel pada jendela properties. Pilih tab Columns. Ubah Title1..dst menjadi Nis, Nama, Alamat, Jenis Kelamin

Table Model
Columns
Rows

Title	Type	Resizable	Editable
NIS	Object	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Nama	Object	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Alamat	Object	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Jenis Kelamin	Object	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Count:

Insert

Delete

Move Up

Move Down

Title:

☒ Resizable
☒ Editable

Type: Object

Pref. Width: Default

Editor: <none>

Min. Width: Default

Renderer: <none>

Max. Width: Default

Selection Model: Not Allowed

☒ Allow to reorder columns by drag and drop

Close

Lalu klik close, maka tampilannya akan jadi seperti dibawah ini

MASTER DATA SISWA

NIS

Nama

Jenis Kelamin

Alamat

NIS	Nama	Alamat	Jenis Kelamin

Setelah tampilan formnya terbuat, kita akan membuat tombol-tombol untuk mengoperasikan form. Buatlah JButton sebanyak 4 buah dibawah jTable. Masing-masing dengan text **simpan, ubah, hapus, tambah/clear** dengan nama variable **BtnSimpan, BtnUbah, BtnHapus, BtnClear**.

MASTER DATA SISWA

NIS

Nama

Jenis Kelamin

Alamat

NIS	Nama	Alamat	Jenkel

Sekarang kita akan menampilkan data yang ada pada table siswa di mysql kedalam jTable, jika data belum ada tambahkan minimal 1 record agar terlihat tampilan datanya. Langkah membuat tampilan datanya adalah sebagai berikut

1. Buatlah sebuah method dengan nama tampil siswa dengan coding sebagai berikut :

```

public void tampil_siswa(){
    try{
        Connection con=koneksi.getKoneksi();
        Statement st=con.createStatement();

        String sql="select * from siswa";
        ResultSet rs=st.executeQuery(sql);
        while(rs.next()){
            Object obj[]=new Object[4];
            obj[0]=rs.getString("nis");
            obj[1]=rs.getString("nama");
            obj[2]=rs.getString("alamat");
            obj[3]=rs.getString("jenkel");
        }
        rs.close();
        st.close();
    }
    catch(SQLException ex){
        System.out.println("ada error");
    }
}

```

2. Buat defaulttable model dengan mengimportnya **import javax.swing.table.DefaultTableModel;** Setelah itu buat objectnya yang bernama model. Ketik **private DefaultTableModel model;** sebelum konstruktor. Setelah itu sempurnakan method tampil siswa menjadi seperti ini

```

public void tampil_siswa(){
    model.getDataVector().removeAllElements();
    model.fireTableDataChanged();
    try{
        Connection con=koneksi.getKoneksi();
        Statement st=con.createStatement();
        String sql="select * from siswa";
        ResultSet rs=st.executeQuery(sql);
        while(rs.next()){
            Object obj[]=new Object[4];
            obj[0]=rs.getString("nis");
            obj[1]=rs.getString("nama");
            obj[2]=rs.getString("alamat");
            obj[3]=rs.getString("jenkel");
            model.addRow(obj);
        }
        rs.close();
        st.close();
    }
    catch(SQLException ex)System.out.println("ada error");
}

```

3. Tambahkan coding berikut pada konstruktor

```
public sis() {  
    initComponents();  
    model=new DefaultTableModel();  
  
    jTableSiswa.setModel(model);  
    model.addColumn("Nis");  
    model.addColumn("Nama");  
    model.addColumn("Alamat");  
    model.addColumn("Jenkel");  
  
    tampil_siswa();  
}
```

4. Setelah itu kita panggil tampil siswa(); pada file utamanya yaitu sekolah.java, pada main functionnya ketik perintah **new siswa().setVisible(true);** setelah itu coba build dan jalankan dari file projectnya, tampilan normal akan seperti ini

The screenshot shows a Java Swing window titled "MASTER DATA SISWA". The window has a light gray background and a blue border. It contains the following elements:

- Four input fields with labels: "NIS", "Nama", "Jenis Kelamin", and "Alamat".
- A table with the following data:

Nis	Nama	Alamat	Jenkel
1001	Anggi	Cicadas 25	P
1002	Beno	Cihanjuang 78	L

Below the table are four buttons: "Simpan", "Ubah", "Hapus", and "Tambah/Clear".

Nis	Nama	Alamat	Jenkel
1001	Anggi	Cicadas 25	P
1002	Beno	Cihanjuang 78	L
1003	Chika	Cibabat 14	P

Setelah menampilkan data dari database, selanjutnya kita akan membuat coding untuk menambah data yang aksinya ada pada buton simpan, maka klik 2 kali pada button simpan maka akan terbuka function BtnSimpanActionPerformed. Berikan perintah ini pada function tersebut

```
private void BtnSimpanActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String nis=TxtNis.getText();
    String nama=TxtNama.getText();
    String jenkel=TxtJenkel.getText();
    String alamat=AreaAlamat.getText();
    try{
        Connection con=koneksi.getKoneksi();
        String sql="insert into siswa(nis,nama,alamat,jenkel) values(?,?,?,?)";
        PreparedStatement ps=con.prepareStatement(sql);
        ps.setString(1, nis);
        ps.setString(2, nama);
        ps.setString(3, alamat);
        ps.setString(4, jenkel);
        ps.executeUpdate();
        ps.close();
    }
    catch(SQLException ex){
        System.out.println("Terjadi error");
    }finally{
        tampil_siswa();
    }
}
```

MASTER DATA SISWA

NIS:

Nama:

Jenis Kelamin:

Alamat:

Nis	Nama	Alamat	Jenkel
1001	Anggi	Cicadas 25	P
1002	Beno	Cihanjuang 78	L

Simpan Ubah Hapus Tambah/Clear

MASTER DATA SISWA

NIS:

Nama:

Jenis Kelamin:

Alamat:

Nis	Nama	Alamat	Jenkel
1001	Anggi	Cicadas 25	P
1002	Beno	Cihanjuang 78	L
1003	Chika	Cibabat 14	P

Simpan Ubah Hapus Tambah/Clear

Sekarang kita akan membuat proses mengubah data yang aksinya berada pada button ubah. Sebelum itu kita harus memberikan aksi pada jTable, jika data pada jTable di klik maka data tersebut akan berpindah ke form untuk dilakukan perubahan data. Klik kanan pada jTable dan pilih **Event→Mouse→MouseClicked**, lalu ketik perintah ini pada functionnya

```

private void jTableSiswaMouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    try {
        int row = jTableSiswa.rowAtPoint(evt.getPoint());

        String nis = jTableSiswa.getValueAt(row, 0).toString();
        String nama = jTableSiswa.getValueAt(row, 1).toString();
        String jenkel = jTableSiswa.getValueAt(row, 3).toString();
        String alamat = jTableSiswa.getValueAt(row, 2).toString();

        TxtNis.setText(String.valueOf(nis));
        TxtNama.setText(String.valueOf(nama));
        TxtJenkel.setText(String.valueOf(jenkel));
        AreaAlamat.setText(String.valueOf(alamat));
    } catch (Exception e) {
    }
}

```

Setelah itu kita akan membuat coding untuk melakukan perubahan data, klik 2 kali pada button ubah, dan berikan tuliskan perintah di bawah ini

```

private void BtnUbahActionPerformed(java.awt.event.ActionEvent evt) {
    String nis=TxtNis.getText();
    String nama=TxtNama.getText();
    String jenkel=TxtJenkel.getText();
    String alamat=AreaAlamat.getText();
    try{
        Connection con=koneksi.getKoneksi();
        String sql="update siswa set nama='"+nama+"',alamat='"+alamat+"',jenkel='"+jenkel+"
where nis='"+nis+"'";
        PreparedStatement ps=con.prepareStatement(sql);

        ps.executeUpdate();
        ps.close();
    }
    catch(SQLException ex){
        System.out.println("Terjadi error");
    }finally{
        tampil_siswa();
    }
}

```

Coba jalankan programnya, dan lihat perubahan datanya. Untuk button hapus ketiklah perintah seperti di bawah ini


```

private void BtnHapusActionPerformed(java.awt.event.ActionEvent evt) {
    String nis=TxtNis.getText();
    try{
        Connection con=koneksi.getKoneksi();
        String sql="delete from siswa where nis='"+nis+"'";
        PreparedStatement ps=con.prepareStatement(sql);

        ps.executeUpdate();
        ps.close();
    }
    catch(SQLException ex){
        System.out.println("Terjadi error");
    }finally{
        tampil_siswa();
    }
}

```

Setelah save dan build coba jalankan proses hapusnya. Untuk menjalankan proses hapus caranya seperti proses edit, pilih dulu data yang akan dihapus pada JTable baru selanjutnya tekan tombol hapus. Untuk selanjutnya kita membuat aksi untuk button clear. Klik 2 kali pada button tersebut dan ketik perintah berikut

```

private void BtnClearActionPerformed(java.awt.event.ActionEvent evt) {
    TxtNis.setText("");
    TxtNama.setText("");
    TxtJenkel.setText("");
    AreaAlamat.setText("");
}

```