

MENUS IN IAVADRAW VISUAL

FILE

PROJECT

EDIT

SEARCH

APPLET

STANDALONE

START

OPTIONS

WINDOW

TOOLS

HELP

FILE MENU

NEW
OPEN

SAVE
SAVE AS

CLOSE
CLOSE ALL

PRINT
SETUP PRINTER

EXIT

PROJECT MENU

NEW

OPEN

ADD

ADD FILE

REMOVE

SAVE

CLOSE

OVERVIEW

EDIT MENU

CUT

DELETE

COPY

PASTE

FORMAT SOURCECODE

SEARCH MENU

SEARCH
SEARCH AGAIN

SET BOOKMARK
GOTO BOOKMARK

APPLET MENU

COMPILE

VIEW

RUN IN BROWSER

STANDALONE MENU

COMPILE

RUN

START MENU

PARAMETERS

OPTIONS MENU

GENERAL
COMPILER
DIRECTORIES
COLORS
LANGUAGE
FONT

WINDOW MENU

OUTPUT

TOOLS MENU

BROWSER

ASCII TABLE

FIND PROCEDURE

GHOSTWRITER

APPLET ASSISTANT

HELP MENU

FILE | NEW

DESCRIPTION:

The File New command will create a normal java document: The new document will be referred to as "Noname.java" in the Document Bar.

FILE | OPEN

DESCRIPTION:

Open an existing file for editing. The file will appear in its own tab-window, and its name will be listed on the Document Bar at the top of the screen and in the editor itself.

FILE | SAVE

DESCRIPTION:

Saves the current document. If you have not saved the file before, this option has the same effect as choosing Save As from the File menu.

USAGE TIPS:

- 1 When you leave JavaDraw Visual, you will be asked if you want to save changes to any documents that have not been saved since the last alteration.
- 2 If you want to save an existing document under a new name, choose File SaveAs. When you save a document, the Most Recent Files list in the File menu will be updated.

FILE | SAVE AS

DESCRIPTION:

Saves the current document with a new name.

FILE | CLOSE

DESCRIPTION:

Closes the current document. If the document has been changed since the last time it was saved, you will be asked if you want to save the changes.

FILE | CLOSE ALL

DESCRIPTION:

Closes all open documents. If any document has been changed since the last time it was saved, you will be asked if you want to save the changes.

FILE | PRINT

DESCRIPTION:

Prints the current document to the default printer.

FILE | SETUP PRINTER

DESCRIPTION:

This option lets you change the default printer and change the setup of the printer. For example, you can choose to print documents in Landscape mode instead of Portrait.

FEATURE:

Gives access to the Windows printer dialog. Using this dialog you can change the,

- Printer properties, (Printers available etc.).
- Paper size & Source.
- Paper orientation, (Landscape or portrait.).

USAGE:

For more information, see the documentation for your printer.

FILE | EXIT

DESCRIPTION:

Using this option you quit out of lavaDraw Visual. If any documents have been changed since the last time they were saved, you will be asked if you want to save the changes.

PROJECT | NEW

DESCRIPTION:

The Project New command will create a new JavaDraw VisualVisual project.

FEATURES:

Creating a new project lets you use the visual components JavaDraw VisualVisual.

USAGE:

Set the name of the new project file and the location.

Further you can choose the type of your project.

Select between applet or standalone application.

By clicking the OK button, the visual form and visual inspector are coming up to your screen.

The JavaDraw Visual project files are coming with an *.jdp extension.

WARNING:

Don't create a new project in the directory, you specified as your temp-directory in the option | directories.

PROJECT | OPEN

DESCRIPTION:

This command will open an existing project.

PROJECT | ADD

DESCRIPTION:

With this command you can add a new file to your project.

FEATURES:

You can choose the baseclass of the new project file.

PROJECT | ADD FILE

DESCRIPTION:

With this command you can add an existing java-file to your project.

PROJECT | REMOVE

DESCRIPTION:

With this command you can remove a java-file from your project.

WARNING:

You can't remove the main project-file.

PROJECT | SAVE

DESCRIPTION:

The Project Save command lets you save your project file.

PROJECT | CLOSE

DESCRIPTION:

Closes the current document. If the project has been changed since the last time it was saved, you will be asked if you want to save the changes.

PROJECT | OVERVIEW

DESCRIPTION:

The project overview gives you important informations about your project.

FEATURES:

The project overview:

- shows you the structure of your project
- shows you the type of your project
- shows you how much time you are working with it
- lets you jump to the files by doubleclick

EDIT | CUT

HOTKEY: STRG+X

DESCRIPTION:

The Cut removes highlighted text from your document and puts it in the Clipboard.

USAGE:

- 1 The cut tool allows you to cut text you have highlighted with your mouse. You can highlight text by moving your mouse over the text while holding the left mouse button down.
- 2 The text is now stored in the clipboard. You can cut & paste slabs of text.
- 3 If you want to position the cut text,
 - Move your mouse to the desired position.
 - Then either use the key combination of CTRL-V
 - Or move your mouse to the Paste option in the Edit menu.
 - The cut text is now positioned on your page.

EDIT | DELETE

HOTKEY: STRG+DEL

DESCRIPTION:

The delete commands lets you delete text from your document.

EDIT | COPY

HOTKEY: STRG+C

DESCRIPTION:

Takes a copy of highlighted text in your document and puts it in the Clipboard . Text can then be pasted from the Clipboard into a document. or even another program.

USAGE:

If you want to copy something from the clipboard then,

- Move your mouse to the desired position.
- Then either use the key combination of CTRL-V
- Or move your mouse to the Paste option in the Edit menu.
- The cut text is now positioned on your page.

EDIT | PASTE

HOTKEY: STRG+V

DESCRIPTION:

Paste inserts the contents of the Clipboard in your document. The information remains in the clipboard. You can use Paste to insert the same information repeatedly.

USAGE:

If you want to paste something from the clipboard then,

- Move your mouse to the desired position.
- Then either use the key combination of CTRL-V
- Or move your mouse to the Paste option in the Edit menu.
- The cut text is now positioned on your page.

EDIT | FORMAT SOURCECODE

DESCRIPTION:

This tool formats your sourcecode. After that you can better read it.

SEARCH | SEARCH

HOTKEY:STRG+F

DESCRIPTION:

Find lets you search for text in your documents. When you choose this option you will be presented with a dialog box.

FEATURES:

There are a several options. The options can be used together or used in combination with each other. For example you can have Find & Case Sensitive together to do an exact search. The search options include,

Find: the text you want to search for.

Case Sensitive:If this option is enabled then a search for the will find “the” but not “The”.

Match Whole Word Only:If this option is enabled then a search for the will find “the” but not “then”

SEARCH | SEARCH AGAIN

HOTKEY: F3

DESCRIPTION:

Search again lets you search again for text in your documents.

SEARCH | SET BOOKMARK

DESCRIPTION:

The set bookmark tool lets you set bookmarks in your source code.

USAGE:

Set the name of your bookmark in the bookmark dialog box.
You can jump to your bookmark by using goto bookmark

SEARCH | GOTO BOOKMARK

DESCRIPTION:

The goto bookmark command lets you jump to bookmarks in your source code. Specify bookmarks with the set bookmark command.

USAGE:

Select the bookmark you want to jump in the body of the goto bookmark dialog.

To jump to the bookmark click the goto button.

To add a new bookmark click the „+“ button.

To remove a bookmark, select the bookmark you want to remove and then press the - button.

APPLET | COMPILE

HOTKEY: STRG+F9

DESCRIPTION:

This command compiles your java file to an class file via the java compiler (javac.exe).

IMPORTANT: Make sure you set the directories properly in option | directories

Make sure the Java Developer Kit (<http://java.sun.com>) is properly installed on your system. Set in General Options | Directories the Location to your JDK directory (e.g. c:\java\jdk102) Further set directories for your temp(e.g. c:\temp) directory (this is the directory, where the compiler writes temporary files. Never save your working files in this directory, because your source will be overwritten from the compiler. Set a directory for the output (e.g. c:\output),this is the directory, where your compiled java files (classes) are located. Next tell javaDraw Visual Visual, where your HTML-browser is located (e.g. c:\netscape3\netscape.exe). You need this to test your java file at your webpage. At least please set the directory where the information portafilter is located on your hard drive. The information portafilter is one of the best documentation for java, written by Bill Bercik.

USAGE:

While the compiler is in process, a compile dialog box comes up. Please wait until the cancel button at this button turns to an ok button. This may take several moments:

3600 Lines of code takes 27 seconds (testmachine: Pentium90, 32MB RAM, WindowsNT4.0)

FEATURE:

If the compiler finds errors in your source code, javaDraw Visual will mark them blue in your source code.

To read more about the errors, open up the javac-output.

APPLET | VIEW

HOTKEY: F9

DESCRIPTION:

This command compiles your java file to a class file via the java compiler (javac.exe) and starts the applet in the appletviewer (appletviewer.exe).

IMPORTANT: Make sure you set the directories properly in option | directories

Make sure the Java Developer Kit (<http://java.sun.com>) is properly installed on your system. Set in General Options | Directories the Location to your JDK directory (e.g. c:\java\jdk102) Further set directories for your temp(e.g. c:\temp) directory (this is the directory, where the compiler writes temporary files. Never save your working files in this directory, because your source will be overwritten from the compiler. Set a directory for the output (e.g. c:\output),this is the directory, where your compiled java files (classes) are located. Next tell javaDraw Visual, where your HTML-browser is located (e.g. c:\netscape3\netscape.exe). You need this to test your java file at your webpage. At least please set the directory where the information portafilter is located on your hard drive. The information portafilter is one of the best documentation for java, written by Bill Bercik.

USAGE:

While the compiler is in process, a compile dialog box comes up. Please wait until the cancel button at this button turns to an ok button. This may take several moments:

3600 Lines of code takes 27 seconds (testmachine: Pentium90, 32MB RAM, WindowsNT4.0)

FEATURES:

If the compiler finds errors in your source code, javaDraw Visual will mark them blue in your source code.

To read more about the errors, open up the javac-output.

Press F9 to make compiling and viewing in one step.

APPLET | RUN IN BROWSER

DESCRIPTION:

This command compiles your java file to an class file via the java compiler (javac.exe) and starts the applet in your specified www-browser.

IMPORTANT: Make sure you set the directories properly in option | directories

Make sure the Java Developer Kit (<http://java.sun.com>) is properly installed on your system. Set in General Options | Directories the Location to your JDK directory (e.g. c:\java\jdk102) Further set directories for your temp(e.g. c:\temp) directory (this is the directory, where the compiler writes temporary files. Never save your working files in this directory, because your source will be overwritten from the compiler. Set a directory for the output (e.g. c:\output),this is the directory, where your compiled java files (classes) are located. Next tell javaDraw Visual Visual, where your HTML-browser is located (e.g. c:\netscape3\netscape.exe). You need this to test your java file at your webpage. At least please set the directory where the information portafilter is located on your hard drive. The information portafilter is one of the best documentation for java, written by Bill Bercik.

USAGE:

While the compiler is in process, a compile dialog box comes up. Please wait until the cancel button at this button turns to an ok button. This may take several moments:

3600 Lines of code takes 27 seconds (testmachine: Pentium90, 32MB RAM, WindowsNT4.0)

Make sure your webbrowser is java capable (for instance netscape navigator 3.0 or microsoft internet explorer 3.0)

FEATURES:

If the compiler finds errors in your source code, javaDraw Visual will mark them blue in your source code.

To read more about the errors, open up the javac-output.

Press F9 to make compiling and viewing in one step.

STANDALONE | COMPILE

DESCRIPTION:

This command compiles standalone java applications.

IMPORTANT: [Make sure you set the directories properly in option | directories](#)

Make sure the Java Developer Kit (<http://java.sun.com>) is properly installed on your system. Set in General Options | Directories the Location to your JDK directory (e.g. c:\java\jdk102) Further set directories for your temp(e.g. c:\temp) directory (this is the directory, where the compiler writes temporary files. Never save your working files in this directory, because your source will be overwritten from the compiler. Set a directory for the output (e.g. c:\output),this is the directory, where your compiled java files (classes) are located. Next tell javaDraw Visual Visual, where your HTML-browser is located (e.g. c:\netscape3\netscape.exe). You need this to test your java file at your webpage. At least please set the directory where the information portafilter is located on your hard drive. The information portafilter is one of the best documentation for java, written by Bill Bercik.

USAGE:

While the compiler is in process, a compile dialog box comes up. Please wait until the cancel button at this button turns to an ok button. This may take several moments:
3600 Lines of code takes 27 seconds (testmachine: Pentium90, 32MB RAM, WindowsNT4.0)

FEATURES:

If the compiler finds errors in your source code, javaDraw Visual will mark them blue in your source code.

To read more about the errors, open up the [javac-output](#).

STANDALONE | RUN

DESCRIPTION:

This command starts your standalone java applications.

IMPORTANT: Make sure you set the directories properly in option | directories

Make sure the Java Developer Kit (<http://java.sun.com>) is properly installed on your system. Set in General Options | Directories the Location to your JDK directory (e.g. c:\java\jdk102) Further set directories for your temp(e.g. c:\temp) directory (this is the directory, where the compiler writes temporary files. Never save your working files in this directory, because your source will be overwritten from the compiler. Set a directory for the output (e.g. c:\output),this is the directory, where your compiled java files (classes) are located. Next tell javaDraw Visual Visual, where your HTML-browser is located (e.g. c:\netscape3\netscape.exe). You need this to test your java file at your webpage. At least please set the directory where the information portafilter is located on your hard drive. The information portafilter is one of the best documentation for java, written by Bill Bercik.

START | PARAMETERS

DESCRIPTION:

This command lets you specify the height and the width of your applet.

USAGE:

The parameters in this window, will take effect, when your applet don't have a resize command.

OPTIONS | GENERAL

DESCRIPTION:

You can set a lot of options to customize lavaDraw Visual for your needs in the general options.

FEATURES:

Shrink unused windows: This option lets you decide if the windows should be shrunk, if they are unused.

Show tab bar: if this option is checked, you will see the tab bar above the editor.

Show line numbers: if this option is checked, linenumbers are visible in your source code

Create Backup files: if this option is checked, lavaDraw Visual Visual will create backups of your files. The extension of the backup files will be *.bak

Show status line: if this option is checked, the status line at the top of lavaDraw Visual is visible

Auto associate extensions for class, java, jdp: if these options are checked you can start java, class or lavaDraw Visual project-files from the command, or via doubleclick from your explorer.

Autosave: if this option is checked lavaDraw Visual will save your files automatically in the interval you specify.

OPTIONS | COMPILER

DESCRIPTION:

Set here all options for the compiler.

FEATURE:

Debug:

Debug Info: Enables generation of debugging tables. Debugging tables contain information about line numbers and local variables - information used by Java debugging tools. By default, only line numbers are generated, unless optimization (-O) is turned on.

Depend:

Disable compiler warnings: Turns off warnings. If used the compiler does not print out any warnings.

Optimize output: Optimizes compiled code by inlining static, final and private methods. Note that your classes may get larger in size.

No write:

Verbose output: Causes the compiler and linker to print out messages about what source files are being compiled and what class files are being loaded.

Classpath: Specifies the path javac uses to look up classes. Overrides the default or the CLASSPATH environment variable if it is set. Directories are separated by semi-colons. Thus the general format for path is:

Compileorder: (enabled in projects only): set here in which order your project files should be compiled.

OPTIONS | DIRECTORIES

DESCRIPTION:

In the directories you can set all locations of the files javaDraw Visual needs for properly working.

USAGE:

Make sure the Java Developer Kit (<http://java.sun.com>) is properly installed on your system. Set in General Options | Directories the Location to your JDK directory (e.g. c:\java\jdk102). Please don't link anymore to the BIN-directory, but to the rootdirectory of the JDK. Further set directories for your temp(e.g. c:\temp) directory (this is the directory, where the compiler writes temporary files. Never save your working files in this directory, because your source will be overwritten from the compiler. Set a directory for the output (e.g. c:\output), this is the directory, where your compiled java files (classes) are located. Next tell javaDraw Visual, where your HTML-browser is located (e.g. c:\netscape3\netscape.exe). You need this to test your java file at your webpage. At least please set the directory where the information portafilter is located on your hard drive. The information portafilter is one of the best documentation for java, written by Bill Bercik.

OPTIONS | COLORS

DESCRIPTION:

Set colors for the editor here.

FEATURES:

Editor background: this is the color for the background of the editor

Editor font default: this is the color for the font

Editor font numbers: this is the color for the numbers

Editor font special: this is the color for the special characters like {}[]

Change button: with this command, you can change the colors

Default button: with this command, you can set the default color values

OPTIONS | LANGUAGE

DESCRIPTION:

You can set language in JavaDraw Visual. Choose between German or English language.

OPTIONS | FONT

DESCRIPTION:

Here you can set the parameters for the font, you want to use in the editor.

WINDOWS | OUTPUT

SHORTCUT: SHORTCUT BAR

DESCRIPTION:

The output window shows the messages from the javac compiler. Here you can get more informations about possible errors in your source codes.

TOOLS | BROWSER

DESCRIPTION:

This command starts the web browser, you specified in options | directories.

TOOLS | ASCII TABLE

SHORTCUT: SHORTCUT BAR

DESCRIPTION:

The ascii table is one of the must have's in a programmers life. This tools gives you the ascii value simply by pressing the key, you want the ascii value from.

TOOLS | FIND PROCEDURE

DESCRIPTION:

This command shows all procedures, that you use in your source code.

FEATURES:

Doubleclick to the procedures in the dialog box, lets you jump to it in your sourcecode.

TOOLS | GHOSTWRITER

DESCRIPTION:

The ghostwriter lets you program Java by using mouseclicks only.

FEATURES:

Select the procedures or methods you want to use in the above body of the ghostwriter. Parameters you can find in the lower half of the ghostwriter.

TOOLS | APPLET ASSISTANT

DESCRIPTION:

The Applet Assistant offers you the possibility to create your own applets by using a assistant. Further you can easily build an own assistant for your friends or for other purposes. Check out the following steps:

FEATURES:

Using the applet assistant script language, you can build your own assistants
The following steps will create an Assistant for your own applets:

- 1.) Add [Name of your applet]=[Filename of your applet] to the section 'Allfiles' in the file 'WIZARD.WIZ'.
You can either add the full path or only the filename.
When only the filename is specified the wizard assumes the file is located in ???\Javdraw(?)\Wizard
- 2.) Create a textfile named [Name of your applet].wiz which is in the following description just called: wizfile.
- 3.) Create the following sections in the wizfile:
'[Properties]'
'[Preset]'
'[Classtart]'
'[Description]'
'[Cut]'
- 4.) In the Properties-section of your wizfile add all names of the properties the user shall modify when running the Assistant and the type of the property. e.g.:
[Properties]
AText=String
MyFontColor=Color

The Assistant acknowledges the following types:

String
Integer
Color
Font

- 5.) Add in the Preset-section of the wizfile. The presets are wizard what the Assistant shows for your properties.
e.g.:
[Preset]
AText=Hello darling
MyFontColor=red
If you don't specify [Preset] then Assistant uses default.
- 6.) In the Classtart-section you should add the line which defines your applet class.
e.g.:
[Classtart]
Line=public class MyApplet extends Applet

This will be the line the Assistant places after the properties as variable definitions. If you don't specify [Classtart] or Line the Assistant uses default ('public class').

- 7.) The description-section gives the Assistant-user additional information, which property is to be specified and what kind it is. Each property may have two lines of description. Add in the following way: First property e.g. AText, line 1:

[Description]

11=Please enter a text!

second line

12=...or let it be...

e.g. third property:

31=Welcome...

32=...nice to see you

- 8.) In the cut-section you may add two definitions:

e.g.: [Cut]

From=//Wizard cuts from this line...

To=//...this line.

The wizard then will cut all lines between From and To (including From and To).

The purpose for cutting is in the event the properties defined in the Assistant already exists in the original applet. This would result in a compile error.

If you don't specify From or to the Assistant uses defaults:

From=//JAVADRAW Wizard.Cut.Start

To=//JAVADRAW Wizard.Cut.End

- 9.) Make sure your applets behavior relates to the specified properties.

HELP | HELP TOPICS

DESCRIPTION:

This will start the information portafilter written by Bill Bercik.

IAVADRAW VISUAL VISUAL - THE COMPONENTS

DESCRIPTION:

The use of the **lavaDraw VisualVisual components is possible in projects only.**

Components are the building blocks of every lavaDraw Visual Application, and the base of the lavaDraw Visual Visual tools. The Visual-page tab displays a group of icons representing the components used to design your application interface.

Components can be either visual or non-visual. Each component has specific attributes that enable you to control your application. These attributes are Properties, Events and Methods.

The Component palette provides Help Hints. Help Hints display a small pop-up window containing the name or brief description of the button when your cursor is over the button for longer than one second.



The Visual Components:

- 1 ... Label
- 2 ... Button
- 3 ... Checkbox
- 4 ... CheckboxGroup
- 5 ... TextField
- 6 ... TextArea
- 7 ... FileDialog
- 8 ... List
- 9 ... MessageFrame
- 10 ... Image
- 11 ... PopUpMenu
- 12 ... Frame
- 13 ... MenuBar
- 14 ... Polygon
- 15 ... Rectangle
- 16 ... ScrollBar
- 17 ... Line
- 18 ... FindDialog

VISUAL FORM

DESCRIPTION:

The Form component is at the center of JavaDraw Visual applications. You design your application by putting other components on a form.

Select the component that you like to place, at the visual component Menu and click to the Visual Form. Now you can set the properties and the events for the specified component in the inspector.

The look of the visual form mirrors the look of your applet or application, showed in your HTML-browser or Appletviewer, when you compile the java-file.

Every project file can have its own visual form. For that choose add in the project menu. Take care that the base class of your applications is frame, windows or dialog.

VISUAL INSPECTOR

DESCRIPTION:

With the inspector you can set the properties and the events of the visual components.

The inspector is your tool for creating your applications. You

The lavaDraw Visual Visual Inspector is the gateway between your application's visual appearance and the code that makes your application run. The Inspector enables you to set design-time properties for the visual components you have placed on the visual Form (or for the form itself), and create and help you navigate through event handlers.

In the General Options Menu you can set that the inspector is shrinking, when you don't need it.

The inspector has two pages:

Values:

The Values page of the lavaDraw Visual Inspector enables you to set design-time properties for components on your form, and for the form itself. You can set run-time properties by writing source code inside event handlers.

The Values page displays only the properties of the component that is selected on the form.

By setting properties at design time you are defining the initial state of a component.

Events:

The Events page of the Object Inspector enables you to connect forms and components to program events. When you double-click an event from the Events page, lavaDraw Visual creates an event-handler and switches focus to the Editor. In the Editor, you write the code inside event-handlers that specifies how a component or form responds to the a particular event.

The Events page displays only the events of the component that is selected in the form.

You can set events by doubleclick to the event you want.

LABEL COMPONENT

DESCRIPTION:

Displays text that the user cannot select or manipulate, such as title text.

Purpose:

Use a label to display text that the user cannot edit, for example, to display a name for components that do not have their own Caption property.

available Properties:

Color: the color of the label
Caption: the caption of the label
Name: the name of the label, set by javaDraw Visual
Left: the distance to the left side of the visual form
FontColor: the color of the font
Top: the distance to the top of the visual form
Fontname: the name of the font; you can use TimesRoman, Helvetica, System
Fontsize: the size of the font
Fontstyle: the style of the font

see also:

[visual components](#)

BUTTON COMPONENT

DESCRIPTION:

Creates a pushbutton control that users choose to initiate actions.

Purpose

Use the Button component to provide buttons users can choose to carry out commands.

available Properties:

Caption: the caption of the button
Height: the height of the button
Name: the name of the button, set by javaDraw Visual
Width: the width of the button
Left: the distance to the left side of the visual form
Parent:
Top: the distance to the top of the visual form

available Events:

onClickEvent
onKeyPressEvent
onKeyReleaseEvent

see also:

[visual components](#)

CHECKBOX COMPONENT

Purpose

Use the CheckBox component to present Yes/No or True/False options to the user, particularly where more than one choice at a time is available from a group of choices.

Presents an option that a user can toggle between Yes/No or True/False. You can use check boxes to display a group of choices that are not mutually exclusive. Users can select more than one check box in a group.

See these topics for more information: [CheckBoxGroups](#)

available Properties:

Caption: the caption of the checkBox
Height: the height of the checkBox
Name: the name of the checkBox, set by lavaDraw Visual
Width: the width of the checkBox
Left: the distance to the left side of the visual form
Parent:
Top: the distance to the top of the visual form
Checked: you can set true or false by doubleclick; false is unchecked true is checked

available Events:

onClickEvent
onKeyPressEvent
onKeyReleaseEvent

see also:

[visual components](#)

THE CHECKBOXGROUP COMPONENT

DESCRIPTION:

The CheckBoxGroup component lets you easily create a checkboxgroup at the Visual Form. See these topics for more informations: [CheckBox](#)

available Properties:

ChildCount:

Height: the height of the CheckBoxGroup

Name: the name of the checkBoxGroup, set by lavaDraw Visual

Width: the width of the checkBoxGroup

Left: the distance to the left side of the visual form

Top: the distance to the top of the visual form

see also:

[visual components](#)

TEXTFIELD COMPONENT

DESCRIPTION:

Displays an editing area where the user can enter or modify a single line of data.
See these topics for more information: [TextArea](#)

Purpose

Use an textField to read or write a single line of text (the TextField component does not recognize end-of-line characters).

To read and write multiple lines of text, use the [TextArea](#) component. To display text that the user cannot modify, use the [Label](#) component.

available Properties:

Height: the height of the TextField
Left: the distance to the left side of the visual form
Name: the name of the TextField, set by javaDraw Visual
Width: the width of the TextField
Parent:
Text: the Text to display in the TextField
Top: the distance to the top of the visual form

available Events:

onKeyPress
onKeyRelease

see also:

[visual components](#)

TEXTAREA COMPONENT

DESCRIPTION:

Displays an editing area where the user can enter or modify multiple lines of data.

Purpose

The TextArea component provides an area for text manipulation. Use the TextArea component to read in multiple lines of text, whether entered by the user or programmatically. (Contrast with the TextField component, which handles only single lines of text.)

available Properties:

Height:	the height of the TextArea
Left:	the distance to the left side of the visual form
Name:	the name of the TextArea, set by javaDraw Visual
Width:	the width of the TextArea
Text:	the Text to display in the TextArea
Top:	the distance to the top of the visual form

see also:

[visual components](#)

FILEOPEN DIALOG COMPONENT

Purpose

Displays files and enables the user to select them in the current directory.

available Properties:

Caption: the caption of the Filedialog

Name: the name of the FileDialog, set by lavaDraw Visual

see also:

[visual components](#)

MESSAGEFRAME COMPONENT

DESCRIPTION:

The MessageBox method displays a generic dialog box that displays a message and one or more buttons.

available Properties:

Caption:the caption of the MessageFrame

Name:the name of the MessageFrame, set by javaDraw Visual

CancelButton:you can set false or true by doubleclick ; false=set no cancel button, true=set a cancelbutton

Text:the Text to display in the MessageFrame

NoButton:you can set false or true by doubleclick ; false=don't set nobutton, true=set a nobutton

OkButton:you can set false or true by doubleclick ; false=don't set Okbutton, true=set a Okbutton

YesButton: you can set false or true by doubleclick ; false=don't set Yesbutton, true=set a Yesbutton

Documentation:

function

public int getResult

Value 0 = OkButton

Value 1 = YesButton

Value 2 = NoButton

Value 3 = CancelButton

see also:

[visual components](#)

LISTBOX COMPONENT

Purpose

Use the ListBox component to display a scrollable list of items that users can select but cannot directly modify. You need to populate the list box programmatically, as users cannot enter or edit items in the list.

Available Properties

Width:	the width of the listbox component
Height:	the height of the listox component
Top:	the distance to the top of the visual form
Items:	the items of the listbox component
Name:	the name of the listbox component
Left:	the distance to the left side of the visual form

see also:

[visual components](#)

MENUBAR COMPONENT

WARNING:

The menubar is available in standalone java application projects only

Purpose

Use the MenuBar component to create a menu bar at the top of your application.

Available Properties

Top:	the distance to the top of the visual form
Items:	the items of the menu bar component
Name:	the name of the menu bar component
Left:	the distance to the left side of the visual form

see also:

[visual components](#)

IMAGE COMPONENT

DESCRIPTION:

Displays a gif or jpg.

Purpose:

Use the Image component to import a graphical image into your form. javaDraw VisualVisual supports the following picture formats:

Available Properties

Filename: set the filename of the image-file

Path: set the path to the image; by doubleclick you can select the image-file from the filedialog

Name: the name of the image

Left: the distance to the left side of the visual form

Top: the distance to the top of the visual form

Note: Image component needs import `java.net`

see also:

[visual components](#)

RECTANGLE COMPONENT

DESCRIPTION:

The Rectangle component lets you easily create a Rectangle.

available Properties:

Height:	the height of the Rectangle
Name:	the name of the Rectangle
Left:	the distance to the left side of the visual form
Width:	the width of the Rectangle
Color:	the color of the rectangle
Top:	the distance to the top of the visual form

see also:

[visual components](#)

FIND DIALOG COMPONENT

DESCRIPTION:

The FindDialog component lets you easily create a FindDialog at the Visual Form.

Available Properties

Name: the name of the FindDialog

Text: the text to display in the FileDialog

Documentation:

function

public string getSearchtext = The text you want to search for.

public boolean getWholeword = Searches for words only

public boolean getCasesensitive = Differentiates uppercase from lowercase when performing a search.

public boolean getDown = From the current position to the end of the file.

public boolean getEntiretext = The search covers either the entire block of selected text or the entire file (no matter where the cursor is in the file).

see also:

[visual components](#)

VISUAL COMPONENTS

DESCRIPTION:

Components are the building blocks of lavaDraw VisualVisual applications. Although most components represent visible parts of a user interface, components can also represent nonvisual elements in a program.

From the end user's perspective, a component is something to choose from the palette and use in an application by manipulating it in the Forms Designer or in code. From the component writer's perspective, however, a component is an object in code. Although there are few real restrictions on what you can do when writing a component, it's good to keep in mind what the end user expects when using the components you write.

At a very practical level, a component is any element that can "plug into" the lavaDraw VisualVisual development environment. It can represent almost any level of complexity, from a simple addition to one of the standard components to a vast, complex interface to another hardware or software system. In short, a component can do or be anything you can create in code, as long as it fits into the component framework.

The definition of a component, then, is essentially an interface specification.

Defining the limits of "component" is therefore like defining the limits of programming. We can't tell you every kind of component you can create, any more than we can tell you all the programs you can write in a given language. What we can do is tell you how to write your code so that it fits well in the lavaDraw Visual Visual environment. The integration of your own components or that you can build your own components is coming in the next lavaDraw Visual Visual releases. Keep your eyes wide open.

EVENT DOCUMENTATION

[OnClick](#)

An OnClick event occurs when the user clicks the component.

[OnKeyPress](#)

The OnKeyPress event occurs when a user presses a single character key. Use the OnKeyPress event handler when you want something to happen as a result of a single character key press.

[OnMouseDown](#)

The OnMouseDown event occurs when the user presses a mouse button with the mouse pointer over a control. Use the OnMouseDown event handler when you want some processing to occur as a result of pressing a mouse button.

The Button parameter of the OnMouseDown event identifies which mouse button was pressed. By using the Shift parameter of the OnMouseDown event handler, you can respond to the state of the mouse buttons and shift keys. Shift keys are the Shift, Ctrl, and Alt keys.

[OnMouseMove](#)

The OnMouseMove event occurs when the user moves the mouse pointer while the mouse pointer is over a control. Use the OnMouseMove event handler when you want something to happen when the mouse pointer moves within the control.

By using the Shift parameter of the OnMouseDown event handler, you can respond to the state of the mouse buttons and shift keys. Shift keys are the Shift, Ctrl, and Alt keys.

[OnMouseUp](#)

The OnMouseUp event occurs when the user releases a mouse button that was pressed with the mouse pointer over a component. Use the OnMouseUp event handler when you want processing to occur when the user releases a mouse button.

The OnMouseUp event handler can respond to left, right, or center mouse button presses and shift key plus mouse-button combinations. Shift keys are the Shift, Ctrl, and Alt keys.

[OnKeyUp](#)

The OnKeyUp event occurs when the user releases a key that has been pressed. Use the OnKeyUp event handler when you want special processing to occur when a key is released. The OnKeyUp handler can respond to all keyboard keys, including function keys and keys combined with the Shift, Alt, and Ctrl keys.

CREDITS & COPYRIGHTS

lavaDraw Visual - the visual development platform for Java(tm), and lava are registered trademarks of SFS SOFTWARE , GERMANY

1. The copyright of the information portafilter is by Bill Bercik. It was very kindly licensed for lavaDraw Visual by Bill Bercik. Thanx Bill.

2. Microsoft, Windows, Windows NT, Windows 95, Internet Explorer, are registered trademarks of Microsoft Corporation.

3. Netscape and the Netscape Navigator are registered trademarks of Netscape Communications.

4. SFS Software would like to thank everyone involved in our test program for their feedback and participation.

1. Java(tm) is an registered trademark of sun microsystems.

HOW TO CONTACT SFS SOFTWARE

Contacting SFS Software varies depending on what you want to contact us for! Check below for the various contact routes and select the most appropriate method for you.

I want to contact SFS Software:

- **to register / order**
- **to make a general query**
- **to contact technical support**
- **to contact sales**
- **by postal mail**
- **by fax**
- **to report a possible software bug**
- **to make some technical suggestions for software**

I WANT TO ORDER NOW !

There are a lot of ways to order/register our software products from us from us. Whatever method you choose, your order must be accompanied by our order form. When we receive your payment we will then e-mail you the serial number that officially registers your copy of the SFS Software product.

I want to order using

secure server of registration service ShareIt!

Ordering this preferred method, you can fill out the ORDERFORM online.

ShareIt! Offers many other methods for you to purchase and license our products to you.

Please jump to the following URL: <http://www.schlenzig.de/sfs/regist.htm> to order our software NOW !

- email, fax modem

using

- Credit card
- Cheque
- International Money Order/Bank Draft
- Cash

SFS Software Gesellschaft bürgerlichen Rechts

Order Form

product	unit price	quantity	Total US\$
lavaDraw Visual 2.1 _____	US\$60,00	_____	_____

Total Price (in US\$) \$ _____

Please select method of payment:

Check (enclosed) OR Money Order (enclosed)
Credit card: VISA Mastercard Eurocard Diners Club AMEX
(we regret that we cannot accept any other credit cards at this time)

Card number: |_|_|_|_| |_|_|_|_| |_|_|_|_| |_|_|_|_|
Expiry Date: ___/___

Card account name:.....

Signature (if possible):

Registration name:.....

E-mail Address:.....

!! PLEASE WRITE E-MAIL CLEARLY !!

Alternative contact:

FAX:.....

PHONE:.....

Address:.....

.....
.....
.....

Thank You for choosing a SFS Software Product.

WHERE IS THE NEXT SFS SOFTWARE OUTLET?

To get a copy of any SFS Software products you can download from our servers at <http://www.schlenzig.de/sfs> .

WHAT'S SFS SOFTWARE'S WWW ADDRESS ?

- <http://www.schlenzig.de/sfs>

I WANT TO REPORT A POSSIBLE BUG !

The e-mail address for bugs is,

sfsware@aol.com

**Remember,
make sure you supply the following
information for bug reports !**

I WANT TO MAKE A TECH. SUPPORT QUERY!

The e-mail address for the Technical
Support crew is,

sfsware@aol.com

**Remember,
make sure you supply the following information
for technical support queries
Technical support is unlimited for registered
users only !**

I WANT TO MAKE A SALES QUERY!

The e-mail address for the Sales
Support crew is,

sfsware@aol.com

**Remember,
make sure you supply the following
information for sales query.**

I WANT TO MAKE A GENERAL QUERY!

If you want to download software, check out the [SFS Software World Wide Web](#) site. If you want to make a general inquiry to SFS Software then e-mail the Sales Support crew,

e-mail: sfsware@aol.com

I WANT TO CONTACT SFS SOFTWARE BY POSTAL MAIL!

The postal mail address for
SFS Software is,

SFS Software
Martin-Luther-Ring 31
98574 Schmalkalden
Germany

Remember,

Postail mail is snail mail and is just that.
If you're mailing from overseas it will take some
time to respond. Print your details clearly.

I WANT TO CONTACT SFS SOFTWARE BY FAX

Our fax number is,

+49 3681 39 40 15

Remember,

Print your details clearly

In the United States, dial 011 instead of „+“

I HAVE SOME GREAT TECHNICAL SUGGESTIONS !

The e-mail address for any technical suggestions is,

sfsware@aol.com

**Remember,
make sure you supply to the following
information for tech. suggestions!**

ORDER BY MONEY ORDER/BANK DRAFT!

These are normally available from banks or Post Offices.
The draft should be payable to SFS Software. The currency of the draft should be Deutsche Mark, and the amount of the draft should be the equivalent in DM of the price in US\$.

ORDER BY EMAIL OR FAX MODEM ?

If you are using e-mail or a fax modem, a personal signature is not required on the order form (although we do recommend that you attach a PGP signature to any e-mail registrations).

ORDER BY CREDIT CARD!

We accept VISA, Mastercard, Eurocard and Diners Club. You can send us your credit card details by fax, post, or PGP - encrypted e-mail.

Do not send your credit card number over the Internet without proper encryption. We will not be held responsible for any financial consequences if you do. Having said that, we have received a large number of our orders via un-encrypted e-mail, and have not heard of any problems so far. If you're feeling particularly lucky, you can ignore this advice... But don't say that we didn't warn you!

ORDER BY CHEQUE!

You can post us a cheque drawn on any bank in the world, in that country's currency. The amount should be equivalent to the price in US\$. For example, if you are in Great Britain you could order by sending a cheque made out in Pounds Sterling, for an amount in pounds equivalent to US\$60.00. The cheque should be made out to SFS Software.

ORDER BY CASH!

We do accept cash payment in US dollars or Deutsche Mark. However, we will take no responsibility for any cash lost in transit. If you are paying in cash, we strongly recommend that your payment is sent to us by registered mail for maximum security.

SALES QUERY INFORMATION!

Name

(full name of person, company the software is registered to)

Registration name

(full name of person, company the software is registered to)

Which software version

(which version and what's the name!)

PRODUCT AND PAYMENT DETAILS

[Purchase information](#)

Please note that if you purchase by credit card, the price you are actually charged may vary slightly from the one listed above. We process all credit card charges at the current exchange rate, but due to daily fluctuations in the value of the dollar we cannot guarantee that this exchange rate will be exactly the same when the transaction is processed by your bank. Purchasing entitles you to unlimited technical support by e-mail.

TECHNICAL SUGGESTION INFORMATION!

To help out technical suggestion please have the following information ready,

Name

(full name of person, company the software is registered to)

Registration name

(full name of person, company the software is registered to)

Which software

(which version what's the name!)

System configuration

(operating system, browser & hardware)

E-mail address

(Please make sure it is legible)

Other software

(what other software was running at the time)

Suggestions

(List any suggestions you think necessary.)

Technical support information!

To help out technical support please have the following information ready,

Name

(full name of person, company s/w is registered to)

Registration name

(full name of person, company s/w is registered to)

Which software

(which version what's the name!)

System configuration

(operating system, browser & hardware)

Send a COPY of any JAVA you are using

E-mail address

(Please make sure it is legible)

If an error happens

(write down exactly what happened or better print it!)

Other software

(what other software was running at the time)

What's PGP?

PGP encryption is the a secure method SFS Software uses to encrypt sensitive data. If you don't know how to use PGP please check the file PGPKEY.TXT included. Following is the Key to be used.

```
Type Bits/KeyID   Date       User ID
pub 1024/D3C82E15 1996/07/04 SFS SOFTWARE <sfsware@aol.com>
```

-----BEGIN PGP PUBLIC KEY BLOCK-----

Version: 2.6.3i

```
mQCNAzHb6TsAAAEANHIAT9qIA0YdqhOloKKISt6JIDPHzZ3I8jO3Xg/LQSwwXaV
hwG/i3G8eYuqQdkchqxHZ5fCP0Cr4K6cqOGrrGfFLrFP3BLref3FUtukgVfzXBN9
iYvsAy3HtYM26YA2H++2D3BvOD9neqUNHI4YYkYhkyT+Fzy9IMIak/bTyC4VAAUR
tB9TRIMgU09GVFdBuKUGPGphdmFkcmF3QGFvbC5jb20+iQCVAwUQMdvpPMIAk/bT
yC4VAQFXHAQAIb9HAVbWtu5QRi2JEazT/lp/xJOvx+JO/ObGocqE55f5nMLsl2LO
Ve9Qj+cPfpGeltGXgDQ1G6iyGXED/5GPVmFixizTjIAVt0eyoQB/wUHKFEh372zi
HSDJFal5R8WYBFTALsR2bbDqkZvW15Cq07DXvF16/6t7bdChzmJf+IU=
=dKOh
```

-----END PGP PUBLIC KEY BLOCK-----

What happens when I purchase the software ?

We don't ship anything physical - when you register/order we will e-mail you a serial number. This registers the program in your name and removes the nag screen, time limit or provides access to any functions that have been disabled in the evaluation copy you have been using.

Report Bug Information

To help out technical support, please have the following information ready:

Name

(full name of person or company,
software is registered to)

Registration name

(full name of person or company,
software is registered to)

Which software

(which version and what's the name)

System configuration

(operating system & browser, etc.)

Send a copy of any Java you are using

E-mail address

If an error happens, write down, what happens

Other software

(write down, what other software was
running at this time)

FIRSTSTEPS - LESSON FOR BEST START IN WORKING WITH IAVADRAW VISUAL

For best starting we created a little first steps tutorial. In it, you can find some little lessons. If you have some ideas or wishes we should add, please let us know.

Lesson1 - Compile and View an existing Applet

Lesson2 - Compile an existing applet and start it in HTML-browser

Lesson3 - Build an applet using javaDraw Visual!'s Visual components

Lesson4 - Build an standalone Java application* using visual components

Lesson5 - Start an applet from the appletassistant

Lesson6 - Learn about the Use of Development Tools

Lesson7 - Your first little applet „HelloWorld.java“

Lesson 1: Compile and View an existing Applet

Step1: Set all directories

Step2: Choose an applet

Click to the appletbar and open applet „tic tac toe“

Step3: Compile applet

The applet „tic tac toe“ is loaded in the editor now. To compile it please choose from the menubar the command `applet | compile`. `lavaDraw Visual Visual` compiles your applet „tic tac toe“ this will take several moments.

Step4: View applet

After compiling the button of the compile dialog box will turn from „cancel“ to „ok“. Press the button now and choose from the menubar the command `applet | View` .

Congratulations.

Lesson2: Compile an existing applet and start it in HTML-browser

Step1: Set all directories

Step2: Choose applet

Click to the appletbar and open applet „tic tac toe“

Step3: Compile applet

The applet „tic tac toe“ is loaded in the editor now. To compile it choose from the menubar the command applet | compile. **lavaDraw Visual Visual** compiles your applet „tic tac toe“ this will take several moments.

Step4: Start applet in browser

After compiling, the button of the compile dialog box will turn from „cancel“ to „ok“. Press the button now and choose from the menubar the command applet | Run (in browser). This command will start applet „tic tac toe“ in your specified HTML-browser (make sure that it is java enabled) .

Congratulations.

Lesson3: Build an applet using javaDraw Visual!’s Visual components

Step1: Set all directories

Step2: Start a new project

To use the visual components of javaDraw Visual Visual You have to start a project. To do that please choose from the menubar the command Project | New. Please insert a name for your project (e.g. Firststeps)
Further set the location for the file by using the browse button. Set the type of the project to applet. Further you can choose which import classes the project file should contain. Press the ok-button.

Step3: Inspector, Visual Forms, Editor

A lot of things will happen now. Your project file will be loaded in the editor, the inspector and the visual form are coming up. They will be open when you are moving your mouse over it. To disable this effect please choose Options | General options.

Step4: Place a button to the visual form

Open up the visual bar. Click to the visual button component. Move your mouse to the visual form now and click on it. The button is on your form now.

Step5: Change the properties of the button

Look at the button on your form. There you can see a blinking edge around it. That means that it is activated now and you can set properties for it. To change the properties of the button, you need the inspector.

The inspector consists of two pages. The Value page (consists the properties) and the event page (consists events, that means what should happen, when the button is clicked, or released,..) .
Open up the value page and change the caption of the button. Change it from „Abutton“ to „OK button“. Do that by changing the text in the textfield. After modifying that press return. The caption of the button is changed now.

Step6: Change height and width of the button

Modify the properties height and width for the button now. Insert 40 for height and 100 for width. Please insert the label component now and make the same steps. Change fontcolor,font or size and look what’s happened.

Step7: Compile and view the applet

You can compile and view your applet by simply press to the F9 key.

Congratulations.

Lesson4: Build an standalone Java application* using visual components

*standalone applications can run without appletviewer or Html browser

Step1: Set all directories

Step2: Start a new project

To use the visual components of lavaDraw Visual Visual You have to start a project. To do that please choose from the menubar the command Project | New. Please insert a name for your project (e.g. Firststeps)

Further set the location for the file by using the browse button. Set the type of the project to standalone. Further you can choose which import classes the project file should contain. Press the ok-button.

Step3: Inspector, Visual Forms, Editor

A lot of things will happen now. Your project file will be loaded in the editor, the inspector and the visual form are coming up. They will be wide open when you are moving your mouse over it. To disable this effect please choose Options | General options.

Step4: Place a menubar to the visual form

Open up the visual bar. Click to the visual menubar component. Move your mouse to the visual form now and click on it. The icon for the menubar is on your form now.

Step5: Change the properties of the menubar

Look at the icon for the menubar on your form. There you can see a blinking edge around it. That means that it is activated now and you can set properties for it. To change the properties of the menubar, you need the inspector.

Open up the value page and change the Items of the menubar, by simply clicking to the button (...). A new textarea comes up now. Click in the area and write the following words, press the return key after each word: „Working|with|lavaDraw Visual Visual“ Please press the ok button after insert the words and take a look to your visual form. You created a menubar on it.

Step7: Compile and view the applet

You can compile and view your standalone application by simply press to the F9 key. Watch that it doesn't run in the appletviewer, but in it's own window.

Congratulations.

Lesson 5: Start an applet from the appletassistant

Step1: Set all directories

Step2: Choose appletassistant

Choose an assistant from the assistant menu by clicking to Tools | Applet Assistant. Please choose the „Javex“ applet assistant. Press the next button now.

Step3: Make your choices

You will be guided now by the assistant. Make your choices. Choose colors, set fonts and keep in mind that you can build the same assistant for your applet too. More information you can find [here](#):

When you reach the apply- button, click it.

Step4: The file in the editor

The file will be loaded to the editor now. To compile and view it press simply the F9 key.

Congratulations.

Lesson 6: Learn about the Use of Development Tools

Step1: Set all directories

Step2: Choose an applet

Click to the appletbar and open applet „tic tac toe“

Step3: Development tool Find Procedure

Start the development tool find procedure with the command Tools|Find procedure. This tool find all the procedures, that are used in the source code. So it is easy to jump between them. See how it works: Doubleclick to procedure in the Find procedure tool.

Step4: Development tool Ascii table

The ascii table is the bible for a programmer. Everytime he needs to know the ascii value for a key. The development tool ascii tool can help fast here. Start this tool with the command tools|ascii table. Click to it and press the key from which you want to know the ascii value. You can find the value for it in the bottom of the table.

Step5: Development tool Ghostwriter

The development tool ghostwriter can help you create software without on fingertip, but creating software using mouseclicks. Click to the commands at the top of the ghostwriter and set the parameters to the commands by using the buttons at the bottom. Software means often that, routines repeat itself and a lot of commands and parameters are often the same. For that you can use the ghostwriter.

Step5: Development tool Bookmarks

With this tool you can set bookmarks in your source code. Set bookmarks with the command search|set bookmarks. Insert a word for your bookmark (e.g. Declarations). Using the command Goto bookmarks you can jump to your bookmarks. Highlight it in the listbox and press the goto button.

Step6: Development tool Format Source code

This tool formats your source code automatically. After using that, it is more easier to read the code.

Start it with Edit|Format Sourcecode.

Congratulations.

Set all directories

[Make sure you set the directories properly in option | directories](#)

Make sure the Java Developer Kit (<http://java.sun.com>) is properly installed on your system. Set in General Options | Directories the Location to your JDK directory (e.g. c:\java\jdk102) Further set directories for your temp(e.g. c:\temp) directory (this is the directory, where the compiler writes temporary files. Never save your working files in this directory, because your source will be overwritten from the compiler. Set a directory for the output (e.g. c:\output),this is the directory, where your compiled java files (classes) are located. Next tell javaDraw Visual, where your HTML-browser is located (e.g. c:\netscape3\netscape.exe). You need this to test your java file at your webpage. At least please set the directory where the information portafilter is located on your hard drive. The information portafilter is one of the best documentation for java, written by Bill Bercik.

Lesson 7: Your first little applet „HelloWorld.java“

[Step1: Set all directories](#)

[Step2: Create a new project](#)

Create a new project using the command `project | new`. Choose the name for the project (e.g. helloworld). Make sure that the import for the classes AWT and applet is enabled. The type of the project should be 'applet'. Press ok - for keep on going.

[Step3: Place a button and a label to your visual form](#)

Open up the visual bar. Click to the visual button component. Move your mouse to the visual form now and click on it. The button is placed on your visual form now. Place the button via drag and drop to the bottom of the form. Place now the visual component label on your form. Please align it at the top of the form.

[Step4: Set properties for the component 'button'](#)

You can set the properties for the components in the visual inspector's values page. So let's set the properties for the button at first. Please the button component at your form now, so you can change the values in the inspector.

Value: caption - change the value caption from „Abutton“ to „Show Hello world!“ - press enter
Value: width - change the value width from 75 to 175 - press enter

[Step5: Set properties for the component 'label'](#)

Please use the same procedure, which you have done with the button-component.

Value: caption- change the caption from „Alabel“ to „Click the button!“ - press enter
Value: fontcolor - change the caption from „black“ to (e.g.) „red“ - press enter
Value: Fontsize - please change the fontsize from „8“ to „40“

[Step6: Set the mouseclick event for the button](#)

The target of this little lesson is it, that when anybody click the button, the caption of the label changes from „click the button“ to „Hello World“. So let's tell javaDraw now, what is to do when anybody clicks the button.

Mark the button on your form and move to the visual inspectors event page. Move your mouse to the onclick - field and doubleclick it.

Please look to the top of the editor, there you can see the following java code:

```
private boolean Button1onClick (Event e)
{
    //Add your code here
    return false;
}
```

All you have to do now, is to insert the following lines to the place where „//add your code here“ is

placed:

```
Label1="HELLO WORLD"; //this will change the caption of the label to 'HELLO WORLD'  
repaint(); //this will paint the screen new for taking the changes effect
```

the whole section for the mouseclick event should be:

```
private boolean Button1onClick (Event e)  
{  
    //Add your code here  
    Label1 = "HELLO WORLD";  
    repaint();  
    return false;  
}
```

Step7: Compile and view your applet

All you have to do now is to compile and view your little java applet. Press F9.
When the applet is coming up in the appletviewer or in your webbrowser, try out if the applet do what it is created to. Press the button !

When the screen shows you HELLO WORLD! Then we congratulate very much to your applet !

