Symantec Visual Cafe Getting Started

Installation and Tour

The software described in this book is furnished under a license agreement and may be used only in accordance with the terms of the agreement.

Copyright Notice

Trademarks

Symantec Corporation
10201 Torre Avenue
Cupertino, CA 95014
408/253-9600

Printed in the United States of America.

10 9 8 7 6 5 4 3 2 1

SYMANTEC LICENSE AND WARRANTY

The software which accompanies this license (the "Software") is the property of Symantec or its licensors and is protected by copyright law. While Symantec continues to own the Software, you will have certain rights to use the Software after your acceptance of this license. Except as may be modified by a license addendum which accompanies this license, your rights and obligations with respect to the use of this Software are as follows:

• You may:

(i) use one copy of the Software on a single computer;

(ii) make one copy of the Software for archival purposes, or copy the software onto the hard disk of your computer and retain the original for archival purposes;

(iii) use the Software on a network, provided that you have a licensed copy of the Software for each computer that can access the Software over that network;

(iv) after written notice to Symantec, transfer the Software on a permanent basis to another person or entity, provided that you retain no copies of the Software and the transferee agrees to the terms of this agreement; and

(v) if a single person uses the computer on which the Software is installed at least 80% of the time, then after returning the completed product registration card which accompanies the Software, that person may also use the Software on a single home computer.

• You may not:

(i) copy the documentation which accompanies the Software;

(ii) sublicense, rent or lease any portion of the Software;

(iii) reverse engineer, decompile, disassemble, modify, translate, make any attempt to discover the source code of the Software, or create derivative works from the Software; or

(iv) use a previous version or copy of the Software after you have received a disk replacement set or an upgraded version as a replacement of the prior version, unless you donate a previous version of an upgraded version to a charity of your choice, and such charity agrees in writing that it will be the sole end user of the product, and that it will abide by the terms of this agreement. Unless you so donate a previous version of an upgraded version, upon upgrading the Software, all copies of the prior version must be destroyed.

• Sixty Day Money Back Guarantee:

If you are the original licensee of this copy of the Software and are dissatisfied with it for any reason, you may return the complete product, together with your receipt, to Symantec or an authorized dealer, postage prepaid, for a full refund at any time during the sixty day period following the delivery to you of the Software.

• Limited Warranty:

Symantec warrants that the media on which the Software is distributed will be free from defects for a period of sixty (60) days from the date of delivery of the Software to you. Your sole remedy in the event of a breach of this warranty will be that Symantec will, at its option, replace any defective media returned to Symantec within the warranty period or refund the money you paid for the Software. Symantec does not warrant that the Software will meet your requirements or that operation of the Software will be uninterrupted or that the Software will be error-free.

THE ABOVE WARRANTY IS EXCLUSIVE AND IN LIEU OF ALL OTHER WARRANTIES, WHETHER EXPRESS OR IMPLIED, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS. YOU MAY HAVE OTHER RIGHTS, WHICH VARY FROM STATE TO STATE.

• Disclaimer of Damages:

REGARDLESS OF WHETHER ANY REMEDY SET FORTH HEREIN FAILS OF ITS ESSENTIAL PURPOSE, IN NO EVENT WILL SYMANTEC BE LIABLE TO YOU FOR ANY SPECIAL, CONSEQUENTIAL, INDIRECT OR SIMILAR DAMAGES, INCLUDING ANY LOST PROFITS OR LOST DATA ARISING OUT OF THE USE OR INABILITY TO USE THE SOFTWARE EVEN IF SYMANTEC HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

SOME STATES DO NOT ALLOW THE LIMITATION OR EXCLUSION OF LIABILITY FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES SO THE ABOVE LIMITATION OR EXCLUSION MAY NOT APPLY TO YOU.

IN NO CASE SHALL SYMANTEC'S LIABILITY EXCEED THE PURCHASE PRICE FOR THE SOFTWARE. The disclaimers and limitations set forth above will apply regardless of whether you accept the Software.

• U.S. Government Restricted Rights:

RESTRICTED RIGHTS LEGEND. Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (c) (1) and (2) of the Commercial Computer Software-Restricted Rights clause at 48 CFR 52.227-19, as applicable, Symantec Corporation, 10201 Torre Avenue, Cupertino, CA 95014.

• General:

This Agreement will be governed by the laws of the State of California. This Agreement may only be modified by a license addendum which accompanies this license or by a written document which has been signed by both you and Symantec. Should you have any questions concerning this Agreement, or if you desire to contact Symantec for any reason, please write:

Symantec Customer Service, 175 W. Broadway,
Eugene, OR 97401.

- Addendum for Development Language Products

Because the Software is a Symantec development language product you have a royalty-free right to include object code derived from the libraries in programs that you develop using the Software and you also have the right to use, distribute, and license such programs to third parties without payment of any further fees, so long as a copyright notice sufficient to protect your copyright in the program is included in the graphic display of your program and on the labels affixed to the media on which your program is distributed.

- Copyright and License Information

Sun grants to you ("Licensee") a non-exclusive, non-transferable license to use the Java binary code version (hereafter, "Binary Software") without fee. Licensee may distribute the Binary Software to third parties provided that the copyright notice and this statement appear on all copies. Licensee agrees that the copyright notice and this statement will appear on all copies of the software, packaging, and documentation or portions thereof.

In the event Licensee creates additional classes or otherwise extends the Applet Application Programming Interface (AAPI), licensee will publish the specifications for such extensions to the AAPI for use by third party developers of Java-based software, in connection with licensee's commercial distribution of the Binary Software.

RESTRICTED RIGHTS: Use, duplication or disclosure by the government is subject to the restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software Clause as DFARS 252.227-7013 and FAR 52.227-19.

SUN MAKES NO REPRESENTATIONS OR WARRANTIES ABOUT THE SUITABILITY OF THE BINARY SOFTWARE, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. SUN SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE BINARY SOFTWARE OR ITS DERIVATIVES.

By downloading, using or copying this Binary Software, Licensee agrees to abide by the intellectual property laws, and all other applicable laws of the U.S., and the terms of this License. Ownership of the software shall remain solely in Sun Microsystems, Inc.

Sun shall have the right to terminate this license immediately by written notice upon Licensee's breach of, or non-compliance with, any of its terms. Licensee shall be liable for any infringement or damages resulting from Licensee's failure to abide by the terms of this License.

The JDK 1.0.2 binary release is based in part on the work of the Independent JPEG Group.

Contents

Chapter 5 Creating the Application and Debugging Code

Chapter 6 Support and More Information

Introducing Visual Cafe

1

Symantec Visual Cafe is the first Rapid Application Development (RAD) environment for creating Web-hosted Java applets and standalone Java applications. It offers a full suite of integrated tools for Java development, including a project manager, form designer, component library, code editor, class browser, interaction wizard, and an integrated graphical debugger coupled with a Java compiler and applet viewer. For many tasks, you don't have to work with raw source code at all.

Whether you are a novice user who wants to quickly create animations for a Web page without touching Java code or a high-powered Java programmer who wants to create sophisticated standalone applications and have control over every line of your code, Visual Cafe is for you.

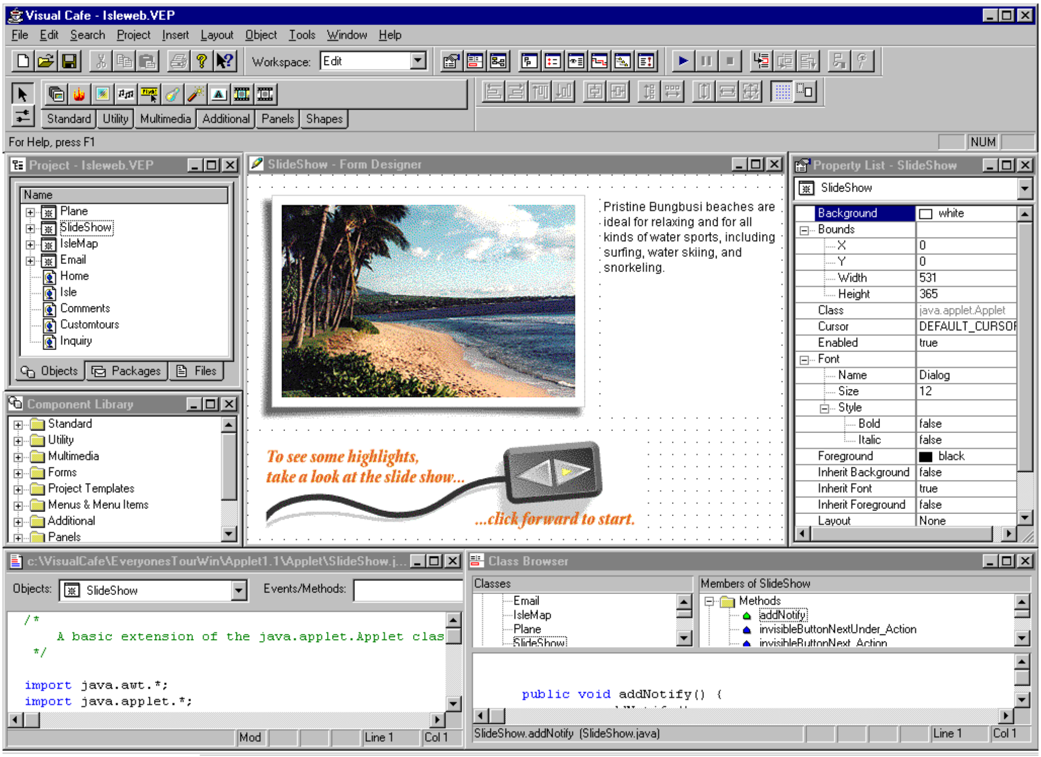
Java applets and Web browsers

If you've explored the World Wide Web (WWW) with a Java-enabled Web browser, chances are you've run into Java applets such as steaming coffee cups, fireworks, scrolling banners, hotel and airline reservation utilities, or even adventure games. Integrating an applet into a Web page is as simple as adding a graphic: you just add a line of Hypertext Markup Language (HTML) code. When you access a Web page with an applet, the applet code is transferred to your computer and executed by the browser, which reduces the load on the Web server and makes the applet run faster on the local computer.

Java is becoming very popular on the Internet because it makes Web pages much more appealing and adds more functionality than HTML alone can provide, so most new browser versions support Java. What's more, Java applets are secure since they can't access the file system on the local computer, but they can still read from the Web server. This means that applets are less likely to damage a user's system or send private information back to the Web server.

Java makes it possible for a Web site to include much richer capabilities, such as sound, animation, and the ability to interact real-time with data on the Web, than is currently possible with static text or graphics. You can include new types of data and provide associated Java bytecode programs that know how to process the data on the client computer. For example, if you wanted your applet to display an image that is stored in a format not supported by most browsers, you could incorporate the ability to handle this graphics format in the applet.

With Visual Cafe, it's easy to build Java applets that add dynamic, interactive functionality to your Internet or intranet Web site. You simply create a new project containing a "skeleton" applet, which you use as a template. You can display the applet in the Form Designer, which enables you to design the applet with graphical tools. To create your user interface, you can drag onto the form a variety of components, such as a button, text field, and even an animation or slide show, then position or resize the components as needed.You assign the component properties in a separate Property List window, and add interactions between components with the Interaction Wizard. Visual Cafe automatically creates the Java code for you during this design process. To test the applet, you can run it with Visual Cafe. The figure shows some elements of the Visual Cafe environment.


Visual Cafe - Isleweb.VEP
File Edit Search Project Insert Layout Object Tools Window Help
Workspace: Edit
Standard Utility Multimedia Additional Panels Shapes
For Help, press F1
NUM
Project - Isleweb.VEP
Name
Plane
SlideShow
IsleMap
Email
Home
Isle
Comments
Customtours
Inquiry
Objects Packages Files
Component Library
Standard
Utility
Multimedia
Forms
Project Templates
Menus & Menu Items
Additional
Panels
SlideShow - Form Designer
To see some highlights, take a look at the slide show...
...click forward to start.
Property List - SlideShow
SlideShow
c:\VisualCafe\EveryonesTourWin\Applet1.1\Applet\SlideShow.j...
Objects: SlideShow
Events/Methods:
Mod
Line 1
Col 1
Class Browser
Classes
Email
IsleMap
Plane
SlideShow
Members of SlideShow
Methods
addNotify
invisibleButtonNextUnder_Action
invisibleButtonNext_Action
SlideShow.addNotify (SlideShow.java)
Line 1
Col 1

Pristine Bungbusi beaches are ideal for relaxing and for all kinds of water sports, including surfing, water skiing, and snorkeling.

Background	□ white
⊟ Bounds	
X	0
Y	0
Width	531
Height	365
Class	java.applet.Applet
Cursor	DEFAULT_CURSOF
Enabled	true
⊟ Font	
Name	Dialog
Size	12
⊟ Style	
Bold	false
Italic	false
Foreground	■ black
Inherit Background	false
Inherit Font	true
Inherit Foreground	false
Layout	None

```
 /*
    A basic extension of the java.applet.Applet clas
  */

 import java.awt.*;
 import java.applet.*;
```

```
    public void addNotify() {
```

After your applet is finished, you can optionally add an HTML file to the project. Visual Cafe lets you create and view HTML files directly within its environment, or you can create the file in another environment and add it to a Visual Cafe project.

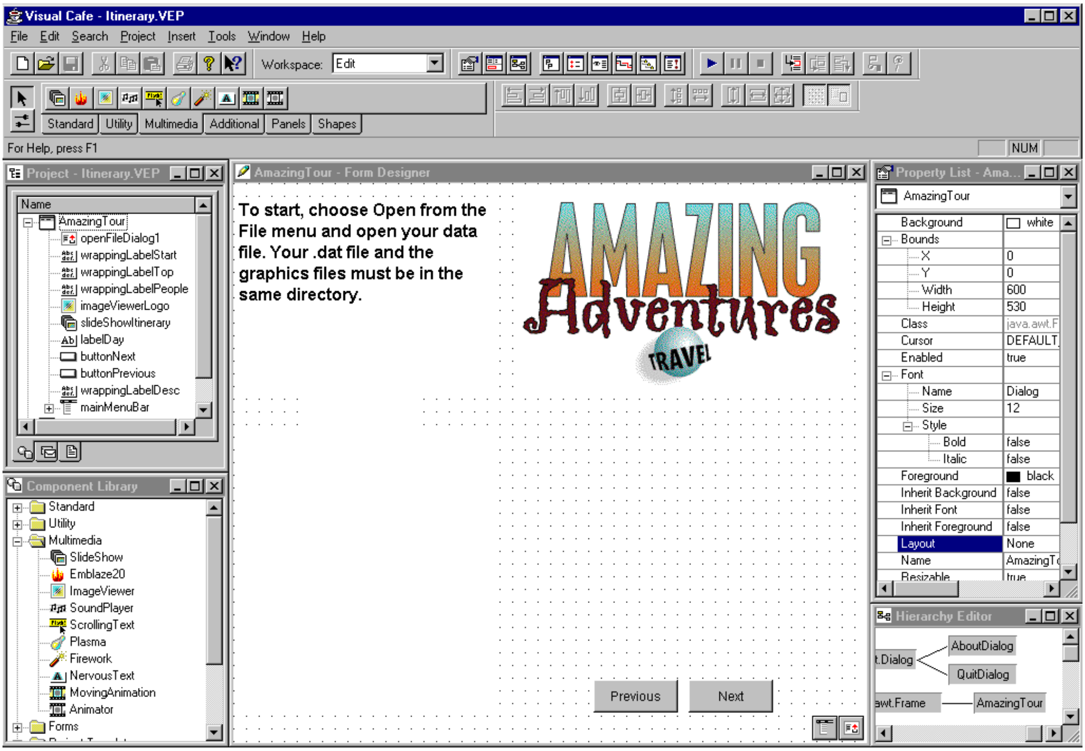
Because of Visual Cafe's extensive library of components, you can develop many applets without manually creating or editing Java code. To program other features, you can manually add Java code with Visual Cafe's Source Editor. For example, you can add more complex custom event handling that can't be created with the Interaction Wizard, add custom data processing, and write your own components then add them to the Visual Cafe Component Library.

Java standalone applications

Java applications are cross-platform, which makes them popular for intranets. A Java application can run on all computers that have Java, so separate applications for each environment aren't needed. Unlike Java applets, Java applications usually have menu bars and can access files on the local computer. This additional functionality lets you implement features in applications that you can't provide in an applet, but it also makes applications less secure than applets. You should be aware of this if you are going to write code to access local storage or any host other than a Web server.

Like other applications, Java applications run standalone without a Web browser. A Java application is like a C++ application, but it's cross-platform and you need a standalone Java virtual machine and supporting files on your computer to run it. For example, you could write a word processor program in Java. Java is free: the Sun Java Developers Kit (JDK), for instance, can be downloaded from the Sun Microsystems Web site, java.sun.com. In the future, Java will be integrated into many operating systems.

Developing a Java application with Visual Cafe is similar to developing an applet. First, you create a project and add all of the elements of your application to it. For your main application window, you can open an application template in the Form Designer, which enables you to design the Java frame with graphical tools. You can drag onto the form a variety of components, then position or resize the components as needed.You assign the component properties in a separate Property List window, including the menu items (the frame can have a menu bar); you can specify interactions between components with the Interaction Wizard. Visual Cafe automatically creates the Java code for you during this design process. The figure shows some of these elements of the Visual Cafe environment.



Workspace: Edit

To start, choose Open from the File menu and open your data file. Your .dat file and the graphics files must be in the same directory.

Background	white
Bounds	
X	0
Y	0
Width	600
Height	530
Class	java.awt.F
Cursor	DEFAULT_
Enabled	true
Font	
Name	Dialog
Size	12
Style	
Bold	false
Italic	false
Foreground	black
Inherit Background	false
Inherit Font	false
Inherit Foreground	false
Layout	None
Name	AmazingT(
Resizable	true

To test the application, you can run it with Visual Cafe.

What is Java?

Java, developed by Sun Microsystems, was designed for creating applets and applications for Internets, intranets, and any other heterogeneous, distributed network. This language offers the following powerful features, as described in the Java white papers published by Sun:

- **Simple**. Java is similar to C and C++, which many programmers are already familiar with. Some of the more difficult features of C++, including operator overloading, pointers and pointer arithmetic, multiple inheritance, and extensive automatic coercions, were omitted to make programming with Java easier. The Java automatic garbage collection feature reduces bugs by automatically freeing unused memory.

- **Small**. The Java virtual machine is relatively small in size, so it can be downloaded over the Internet and run on computers with little available memory. Many operating systems will include Java in the future.
- **Object-oriented**. Java mimics the object orientation of C++ and includes extensions from Objective C for dynamic method resolution. Some advantages of object-oriented programming include the following:
  - Code is encapsulated in objects, which have a public interface and a private implementation, so you can rapidly develop prototypes and group code into manageable chunks — even for very complex systems.
  - Objects can inherit the characteristics of other objects and override inherited characteristics, so you can easily reuse code, make your code more compact, and fix or update code in one place, which saves time and reduces bugs.
- **Network-ready**. Creating network connections is easier in Java than for C or C++ because Java has built-in routines for dealing with TCP/IP, including HTTP and FTP. These routines make it as easy to open and access objects over the network through URLs as it is to access a local file system.
- **Robust**. Java eliminates problems early by requiring declarations, using static typing, having the compiler perform type checks, and not supporting pointers, which can result in overwriting memory or corrupting data.
- **Secure**. Because there are no pointers, Java applications can't access data structures or private data that they don't have access to. This prevents most viruses from taking hold. Applets, when run within a Web browser on a local computer, can't read or write to the disk, execute programs on this computer, or connect to any other computers except the server they were downloaded from.
- **Architecture-neutral and portable**. The Java compiler generates an architecture-neutral object file format and bytecode instructions, so Java code can run on any computer that has a Java runtime system. *Bytecodes* are instructions that are similar to machine code, but are not

platform-specific. During execution, the Java virtual machine either interprets the bytecodes or converts them to machine code. Creating separate applications for different computer platforms is no longer an issue.

- **High-performance**. Java bytecodes can be translated on the fly to native machine instructions — for example, by a Java-enabled browser. Linking is faster than for C or C++. Once the Java bytecodes are converted to machine code by a Just-In-Time compiler in a Java virtual machine, the performance is comparable to that of C or C++.
- **Multithreaded**. Java code can deal with multiple things happening at once with sophisticated synchronization primitives that are integrated into the language, which makes them easier to use and more robust. Multithreading improves interactive responsiveness and real-time behavior, so is critical to high-performance Java applets because applet execution must continue while various image and binary files are being retrieved from one or more Web servers. In addition, the ability to control the execution of multiple concurrent threads is crucial for deploying real-world Web applications.
- **Dynamic**. New module plug-ins can be added to a Java application with minimal overhead. Java can look up a class definition at runtime from its name.

Features of the Visual Cafe environment

Visual Cafe gives you the easy-to-use Java language in a development environment designed to make you highly productive. It includes the following features, which you'll learn more about in the tour:

- Visual, form-based development
- Support for JDK 1.1, portable JavaBeans components, and Java Archive (JAR) files
- Quick creation of component connections with the Interaction Wizard
- Two-way development, which means the visual tools and Java code always match
- Integration with the Visual Page HTML editor

- Rapid turnaround with fast compilers
- An advanced debugger that includes expression evaluation

Installing Visual Cafe

2

This chapter describes how to install Visual Cafe. It has the following sections:

- Requirements for installation
- Installing Visual Cafe
- Starting Visual Cafe
- Exiting Visual Cafe
- Uninstalling Visual Cafe

Requirements for installation

To use Visual Cafe you need:

- Personal computer, Pentium equivalent, 90 MHz minimum
- 24 MB RAM (32 MB recommended)
- 30 to 100 MB free hard disk space (depending on configuration)
- VGA, 256 Color (SVGA recommended)
- Windows 95 or Windows NT 4.0

Installing Visual Cafe

Follow these steps to install Visual Cafe.

If you have a previous version of Visual Cafe, *do not* install to the same directory where it resides. You need to install to a new directory.

1. Insert the CD-ROM in the CD-ROM drive on your computer.
2. If the installer didn't automatically start, choose Run from the Windows Start menu to display the Run dialog box. Then type the following and click OK:

   ***drive*:\setup.exe**

   where *drive* is the CD-ROM drive identifier.

3. Select the name of the software you want to install.

   Look at each of the tabs in the installer to see what is available to you. The install wizard guides you through the installation process; you can select where to install Visual Cafe and which elements to install. Click Next to proceed through the steps.

4. After installing Visual Cafe, you are prompted to register your software.

5. Read the Readme.txt file *before* you use the software. It contains important last-minute information.

   Be sure to look at our Web site, http://cafe.symantec.com, so you can obtain technical support, software upgrades, and news about special promotions. Use the user ID and password included with your package.

Starting Visual Cafe

You can start the application in any of these ways:

- Double-click the application icon in the Program group.
- From the Start menu, choose the application from the Programs menu.
- Run the application (vcafe.exe) or a .vep file from the Run dialog.

The Visual Cafe environment appears. After you use Visual Cafe, it remembers what windows you had open when you last used the application and opens those windows the next time you run the product.

Exiting Visual Cafe

To exit from Visual Cafe, choose Exit from the File menu. You are prompted if you have unsaved changes.

Uninstalling Visual Cafe

To uninstall Visual Cafe:

1. Choose Settings from the Windows Start menu.
2. Choose Control Panel to display the Control Panel window.
3. Choose Add/Remove Programs from the Control Panel window.

4. Click the Install/Uninstall tab and highlight Visual Cafe.
5. Click Add/Remove and follow the instructions on the screen.

Uninstall does *not* remove any files created in the Visual Cafe directory tree after Visual Cafe is installed (for example, sample projects) or in the Visual Cafe group.

Starting the Tour

3

Imagine that you work for a travel agency, called Amazing Adventures Travel, that's ready to take advantage of the money-making opportunities that the Internet offers. You decide to add Java functionality to a series of Web pages that advertise your upcoming package tours and your custom tour services. To get started, you update the pages for your favorite destination, Isle de Cafe.

In this chapter, you'll look at the Isle de Cafe applets and application, some of which you'll create in the following chapters, plus learn some important Visual Cafe terms and concepts.

What you need

To work through the tour, you need to install the Visual Cafe software, as described in Chapter 2, "Installing Visual Cafe." You also need to install the version of Visual Page that is on your Visual Cafe CD, because you will view applets in Visual Page.

After you meet these requirements, you can start the tour with this chapter. Then work through the following chapters in order:

Chapter 4, "Creating an Applet for the Web Page," shows you how to create an island map applet including an animation and integrate it into a Web page, without needing to program in Java code.

Chapter 5, "Creating the Application and Debugging Code," describes how to create a simple application and how to perform advanced debugging operations. This chapter builds on the previous chapter.

Looking at the Isle de Cafe Web pages

Your main Isle de Cafe Web page is the most important page. On this page, you need to grab the initial interest of prospective clients, so they remember your company and want to spend some time at your Web site. A Web page with only HTML could potentially be a little boring. You decide to add Java applets so the page becomes dynamic and interactive.

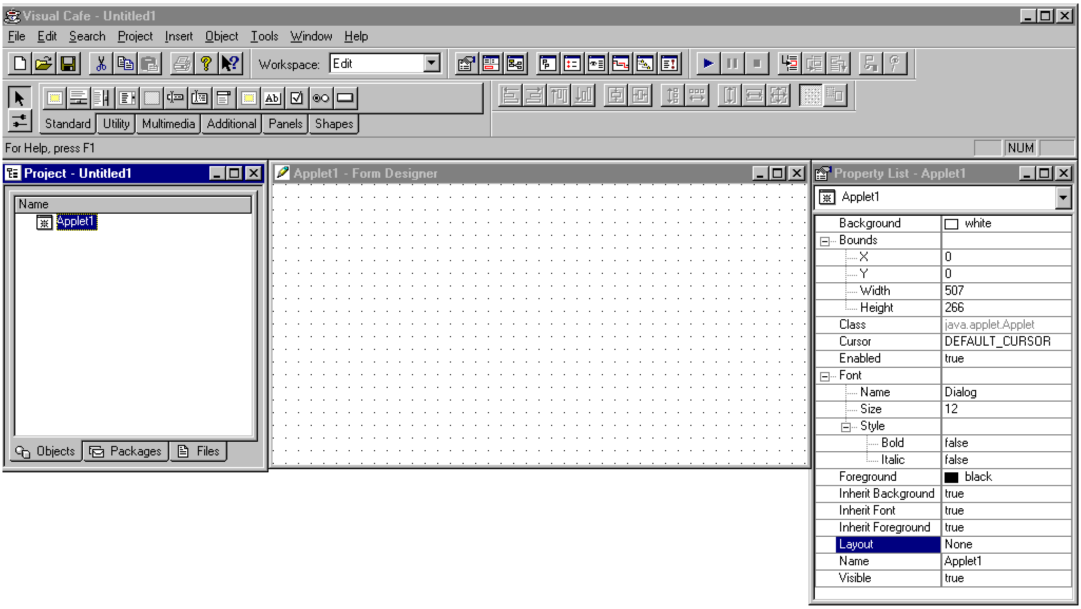
Let's look at the Web pages and identify the elements created with Visual Cafe.

Starting Visual Cafe

You can start the application in any of these ways:

- Double-click the application icon in the program group.
- In the Start menu of Windows 95, choose the application from the Programs menu.
- Run the application or a .vep file from the Run dialog.

When you first start Visual Cafe, the last project you opened or an unnamed project, based on the default project template, appears. By default, the Project, Form Designer, and Property List windows appear.



Background	□ white
Bounds	
X	0
Y	0
Width	507
Height	266
Class	java.applet.Applet
Cursor	DEFAULT_CURSOR
Enabled	true
Font	
Name	Dialog
Size	12
Style	
Bold	false
Italic	false
Foreground	■ black
Inherit Background	true
Inherit Font	true
Inherit Foreground	true
Layout	None
Name	Applet1
Visible	true

You'll work with the Visual Cafe applet and application templates later in this tour.

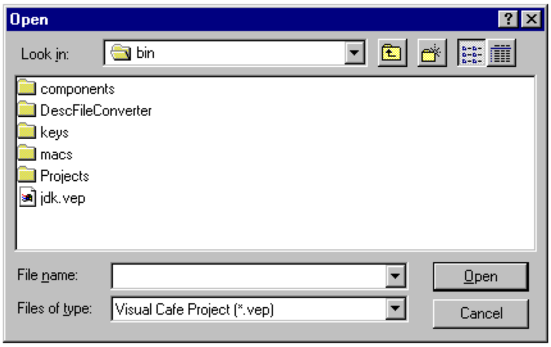
The windows shown in this tour are from Visual Cafe Professional Development Edition. The tour also applies to Visual Cafe Web Development Edition, but some of the windows will look slightly different.

Opening the project

All of the applets and HTML files that make up the Isle de Cafe Web pages are contained within the IsleWeb.vep project. To look at the project, follow these steps:

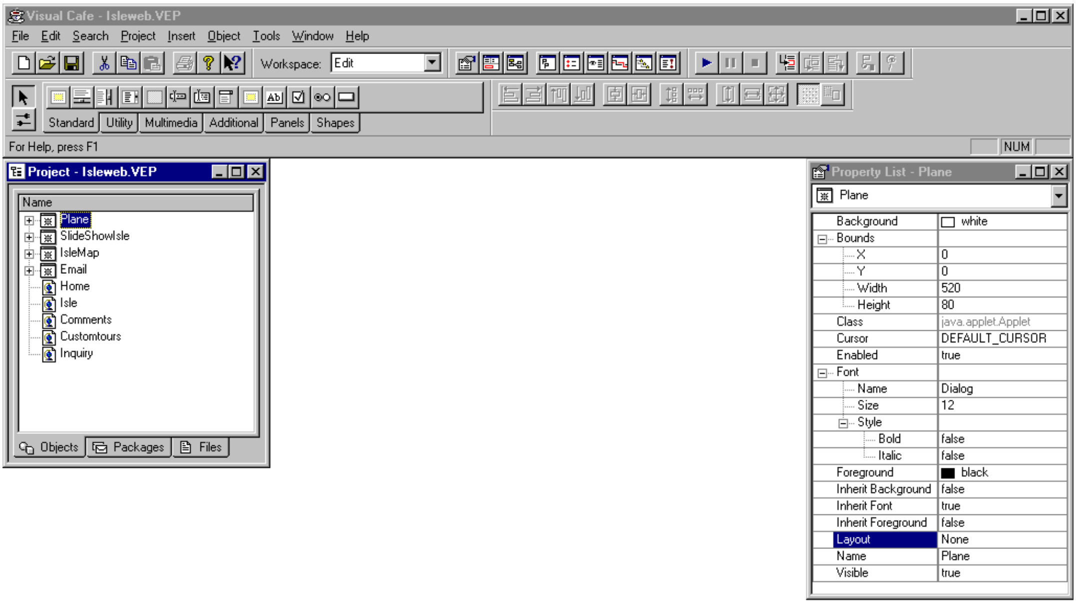
1. Close any open projects by clicking the Project window, then choosing Close Project in the File menu.
2. Choose Open from the File menu.

   The Open dialog appears.



3 Select the IsleWeb.vep project in the \Tour\Applet folder, then click Open.

The project appears.



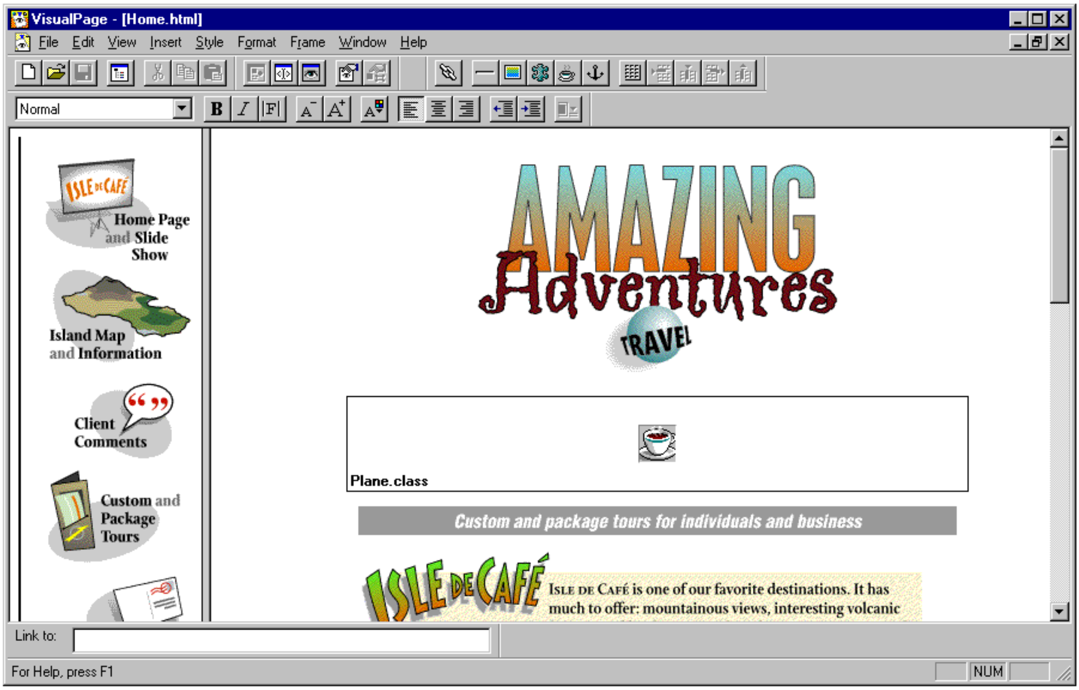
Background	□ white
⊟ Bounds	
X	0
Y	0
Width	520
Height	80
Class	java.applet.Applet
Cursor	DEFAULT_CURSOR
Enabled	true
⊟ Font	
Name	Dialog
Size	12
⊟ Style	
Bold	false
Italic	false
Foreground	■ black
Inherit Background	false
Inherit Font	true
Inherit Foreground	false
Layout	None
Name	Plane
Visible	true

Running the applets in the Web page

Visual Cafe lets you run the applet directly within the Web page by using Visual Page or your default Web browser. The advantage of using Visual Page is you can also modify your HTML files as needed. Follow these steps to open the Web page in Visual Page:

1 In the Objects tab of the Project window, double-click Home.html.

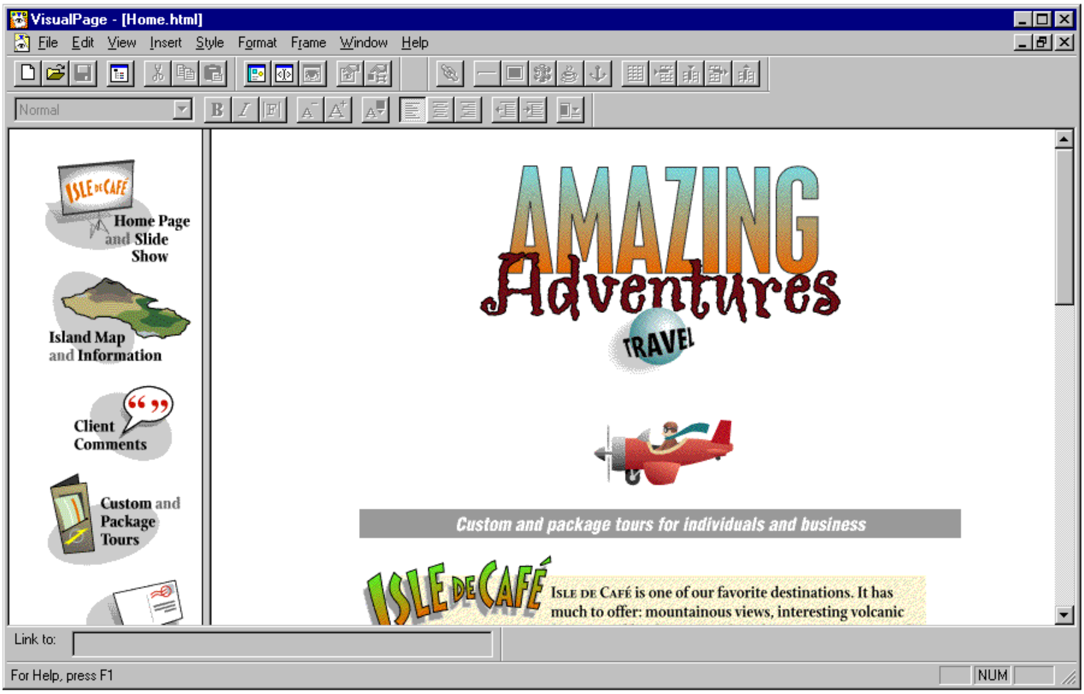
If you have properly installed Visual Page from the Visual Cafe CD, Visual Page starts and displays the Isle de Cafe home page.


VisualPage - [Home.html]
File Edit View Insert Style Format Frame Window Help
Normal
Home Page and Slide Show
Island Map and Information
Client Comments
Custom and Package Tours
AMAZING Adventures TRAVEL
Plane.class
Custom and package tours for individuals and business
ISLE DE CAFÉ
Link to:
For Help, press F1
NUM

ISLE DE CAFÉ is one of our favorite destinations. It has much to offer: mountainous views, interesting volcanic

2 Choose Preview from the View menu.

The applets run in Visual Page.


VisualPage - [Home.html]
File Edit View Insert Style Format Frame Window Help
Normal
Home Page and Slide Show
Island Map and Information
Client Comments
Custom and Package Tours
AMAZING Adventures TRAVEL
Custom and package tours for individuals and business
ISLE DE CAFÉ
Link to:
For Help, press F1
NUM

Isle de Café is one of our favorite destinations. It has much to offer: mountainous views, interesting volcanic

The Web pages contain elements that were created with HTML and with Java using Visual Cafe. The Visual Cafe applets on this page are the plane animation created with the Visual Cafe Animator component and the slide show created with the Visual Cafe SlideShow component.

On the left of the page are links to the other pages.

You use the plane animation to attract the attention of your Web site guests. You also want your potential customers to get an idea of what Isle de Cafe offers them, so you include the slide show. The slide show contains photos of Isle de Cafe that you've collected over the years. Your customers can move back and forward between the photos of the slide show. Next to each photo is a description of the location so your customers know what they're looking at.

3 Click the Forward button (the right triangle) until you've viewed the entire slide show. Also try clicking the Previous button (the left triangle).

4 Explore the other Web pages by clicking a link on the left of the page.

While the main Isle de Cafe Web page is important to capture people's initial interest, once they're interested they may want more in-depth information about the island.

You decide that the best way to present this detailed information is through an island map. On the map, customers can click a location to jump to information about it. This feature was implemented with the ImageViewer and InvisibleHTMLLink components provided with Visual Cafe. You also decide to add an animation, implemented with the Animator component, that draws attention to a main attraction on the island — the ability to safely view an active volcano.

Some additional features you'd like on the Web site include a list of your upcoming package tours, information on the custom tours you can design for clients, an e-mail request form (an applet created with Visual Cafe) so clients can get additional information, and some quotes from clients about how much they enjoyed your tours.

It's time to look at these Web page elements in Visual Cafe directly. You can exit Visual Page or keep it running so you can refer back to the completed Web pages.

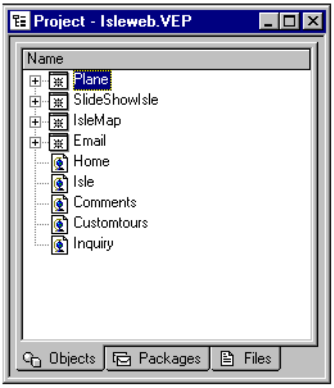
Looking at the applets in Visual Cafe

Now that you've seen how the applets look in your Web pages, let's see how they were created with Visual Cafe.

Looking at the project

The Visual Cafe full-featured project management system and flexible editing and browsing tools let you develop and maintain even the most complex Java applications:

1 Look at the items appearing in the Project window.



Project - Isleweb.VEP

Name

Plane
SlideShowIsle
IsleMap
Email
Home
Isle
Comments
Customtours
Inquiry

You'll notice the following applets in the Project window:

- Plane — plane animation
- SlideShowIsle — slide show
- IsleMap — Isle de Cafe map with volcano eruption
- Email — e-mail information form

And the following HTML files:

- Home — first page
- Isle — page with island map
- Comments — page containing customer comments
- Customtours — page describing custom and package tours
- Inquiry — page with e-mail questionnaire

The Objects view of the Project window shows the components of a project:

- A *visual component* is a user interface element, such as a window, menu, button, and so on, that is visible at run time and appears in the Visual Cafe Project window and Form Designer. It extends from the Java Component class and has a screen position, a size, and a foreground and background color. It is often called a control in C++.
- A *top-level component*, also called a Visual Cafe *form*, can only appear at the top level in the Objects view of the Project window. Applet, Frame, Window, and some dialog components are forms. A form

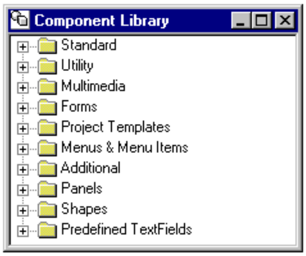
has an associated Java source file and can contain other components, which are represented by code in the source file.

The Objects view also shows HTML files, if present.

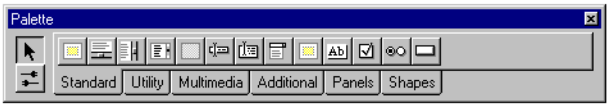
2 In the Project window, click the + next to a form to view the components contained by it.

3 Choose Component Library from the Window menu.

The Component Library appears.



The Component Library is a repository of components that comply with the JavaBeans standard. Visual Cafe provides an extensive library of components, which you can supplement with third-party JavaBeans components and your own components. In addition to supporting the complete set of standard Java components, Visual Cafe provides many additional components. For fast access, configure the Component Palette, which is at the top of the main window or floating, to contain your favorite components:



To add a component to a Visual Cafe form, drag a component from the Component Library to the Project window or Form Designer, then position and resize it as needed. Or, you can click a component on the Component Palette and draw it on the form. In addition, the Component Library contains forms, so you can drag a form into a Project window to add it to the project. To add a component to the Component Palette so you can get to it more quickly, drag the component from the Component Library to the Palette.

4 Close the Component Library window.

5 In the Project window, click the Packages tab.

The Packages view shows the source files in a project, grouped as *packages*. A Java package is a group of related Java classes and interfaces.

6 In the Project window, click the Files tab.

The Files view lists the files in your projects. The Imports folder contains the classes the Java programs need.

7 Click the Objects tab to return to Objects view.

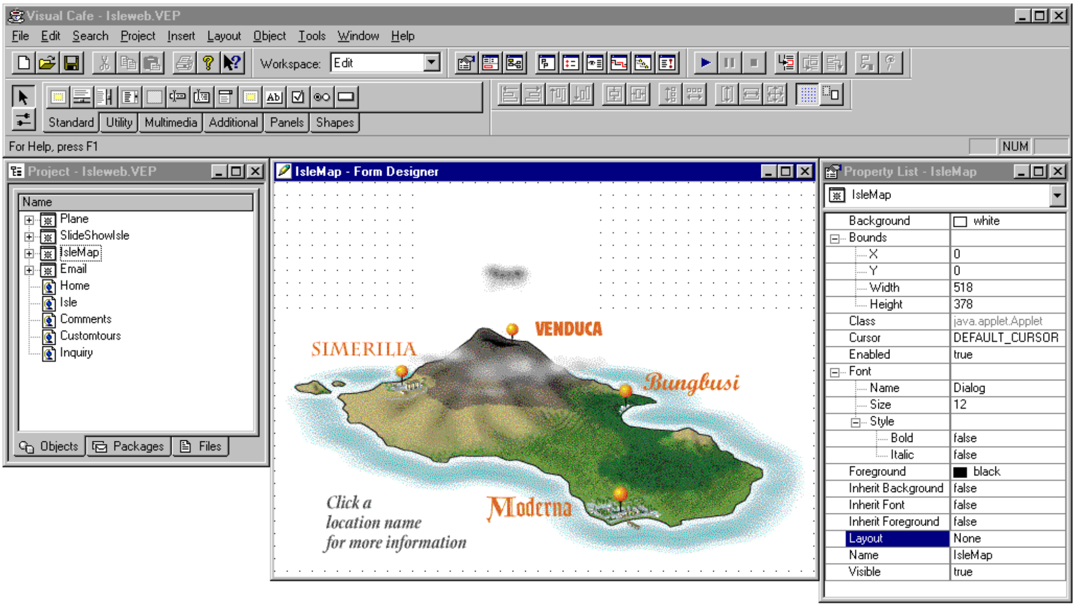
You can double-click a form or file in the Project window to open it. In the following sections, you'll look at the applets and an HTML file in the project.

Looking at the applets

Once the Project is opened, you can look at the applets it contains:

1 Double-click the IsleMap applet in the Project window.

The Form Designer appears.



Background	□ white
Bounds	
X	0
Y	0
Width	518
Height	378
Class	java.applet.Applet
Cursor	DEFAULT_CURSOR
Enabled	true
Font	
Name	Dialog
Size	12
Style	
Bold	false
Italic	false
Foreground	■ black
Inherit Background	false
Inherit Font	false
Inherit Foreground	false
Layout	None
Name	IsleMap
Visible	true

The Form Designer is a true What-You-See-Is-What-You-Get (WYSIWYG) form layout tool: it lets you edit all types of forms, including applets, windows, message boxes, and dialog boxes. To design the visual interface of your Java applications and applets, you can drag and drop components from the Component Palette or Library directly onto your forms. Visual Cafe automatically generates the Java code and changes the code as you make modifications. Several layout managers are available, so you can pick the one that best suits your form, or don't use one at all.

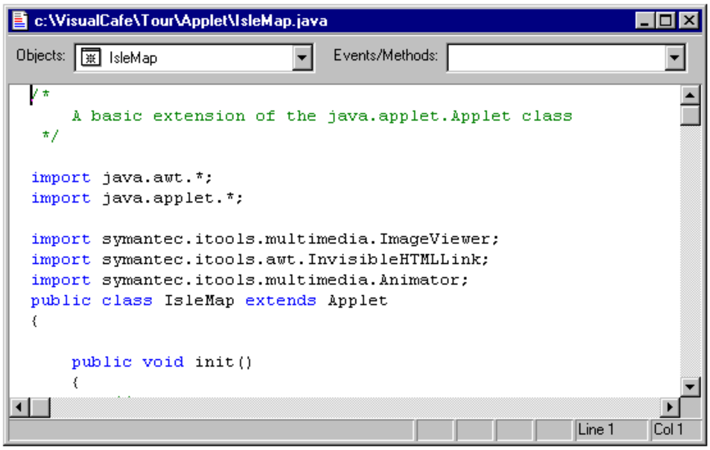
The properties of the components are displayed in the Property List window. To specify the properties of components on a form, you simply fill in fields in the Property List. Visual Cafe updates the component appearance on the form and the Java code for you.

2 To look at the properties for the Animator component, choose the component (animatorErupt) from the pull-down menu at the top of the Property List or click the component on the form.

The properties for the Animator component appear in the Property List window.

3 To look at the underlying Java source code for the applet, double-click the Form Designer. Or right-click over the IsleMap object in the Project window, then choose Edit Source from the pop-up menu.

The applet source code appears in the Source window.



Objects: IsleMap

Events/Methods:

```
/*
    A basic extension of the java.applet.Applet class
 */

import java.awt.*;
import java.applet.*;

import symantec.itools.multimedia.ImageViewer;
import symantec.itools.awt.InvisibleHTMLLink;
import symantec.itools.multimedia.Animator;
public class IsleMap extends Applet
{

    public void init()
    {
```

Visual Cafe supports two-way development by translating Java code into a visual representation and by translating the visual representation in the Form Designer to Java code. Your code and your visual model always match.

The Visual Cafe Source Editor simplifies development with full-color Java syntax and keyword highlighting and an integrated macro language for extending the editor. A single mouse click in a pull-down menu navigates you to a declaration inside the Java project. The Form Designer ensures full synchronization between source code and the visual tools. During debugging, you can use the Source window to monitor program execution.

4 Close the Source window.

5 You can also look at the plane animation by double-clicking Plane in the Project window, the slide show by double-clicking SlideShowIsle, and the e-mail questionnaire applet by double-clicking Email.

Because these are detailed applets, you might want to close each Form Designer window before opening another one.

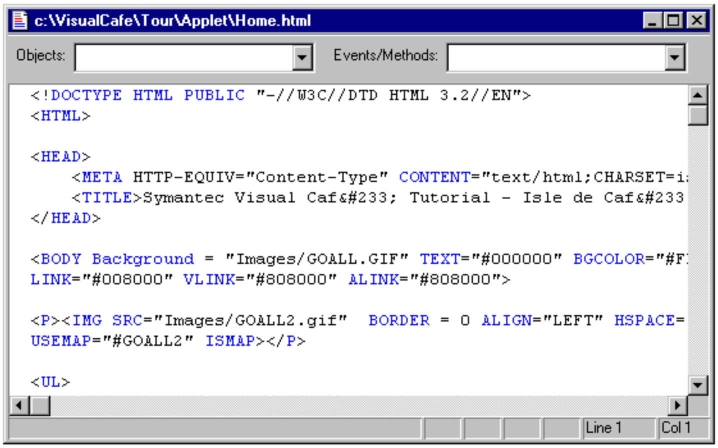
To test your applet, you can run it in the Visual Cafe Applet Viewer. Visual Cafe also lets you run your applets within a Web browser by launching the browser for you. You'll try the Applet Viewer later in the tour.

Opening an HTML file

An applet runs from an HTML file. Although your project doesn't have to contain the HTML files that the applets run in, it's convenient to include the HTML files in the project. Let's look at an HTML file in this project:

1 Click the Files tab in the Project window, then double-click Home.html.

The Source window appears.

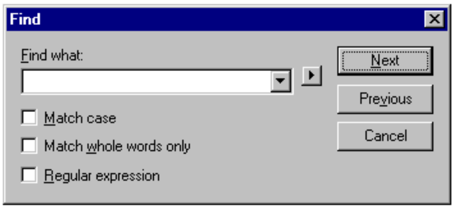


```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">
<HTML>

<HEAD>
    <META HTTP-EQUIV="Content-Type" CONTENT="text/html;CHARSET=i
    <TITLE>Symantec Visual Caf&#233; Tutorial - Isle de Caf&#233
</HEAD>

<BODY Background = "Images/GOALL.GIF" TEXT="#000000" BGCOLOR="#F
LINK="#008000" VLINK="#808000" ALINK="#808000">

<P><IMG SRC="Images/GOALL2.gif"  BORDER = 0 ALIGN="LEFT" HSPACE=
USEMAP="#GOALL2" ISMAP></P>

<UL>
```

The HTML file is displayed in this window. You can create and edit HTML code here. If you've installed Visual Page from the Visual Cafe CD, you can double-click an HTML file in the Project window to launch Visual Page and edit the file there.

2 Choose Find from the Search menu.

The Find dialog appears.



3 Type **applet code**, then click Next.

The words are highlighted in the HTML code. This is part of the applet tag that causes the Visual Cafe applet to appear on the Web page. An applet tag has the following basic format:

```
<APPLET code="applet.class" width=pixw height=pixh></APPLET>
```

`applet` is the name of the applet, *`pixw`* is the number of pixels for the width, and *`pixh`* is the number of pixels for the height.

4 Choose Find Again from the Search menu.

Another applet tag is highlighted in the HTML file.

5 Choose Close Home.html from the File menu.

The Source window closes.

Now let's look at the application. You can close your project by closing the Project window.

Running the custom itinerary application

A large part of your business revenue is generated from designing custom, escorted tours to Isle de Cafe. As you expand your operations onto the WWW, you look at ways to take full advantage of the time-saving aspects of electronic communications.

You decide that a cross-platform Java application would be ideal for distributing custom itineraries to clients for approval. Instead of spending time and money putting together an itinerary on paper — especially one that looks good enough for people purchasing a premium tour — you can put together a polished itinerary in the form of an application. This itinerary is easily modified as changes are needed and includes photos of the locations the clients will visit.

To run the itinerary application, follow these steps:

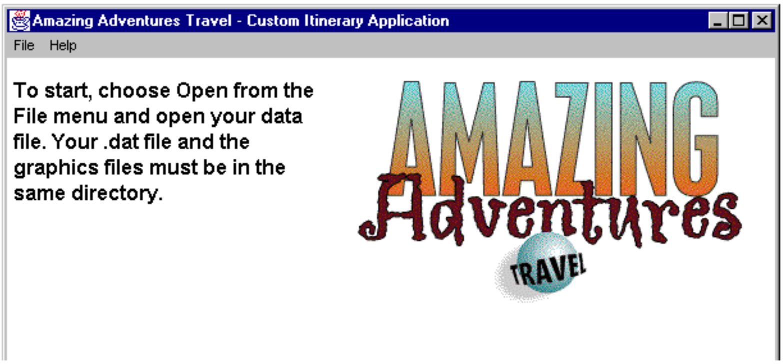
1 In a DOS window, navigate to the directory containing AmazingTour.class.

  If you installed in the default location, it should be in the \Tour\Application directory.

2 At the DOS prompt, enter the full path to java.exe followed by the class. For example:

  \visualcafe\java\bin\java AmazingTour

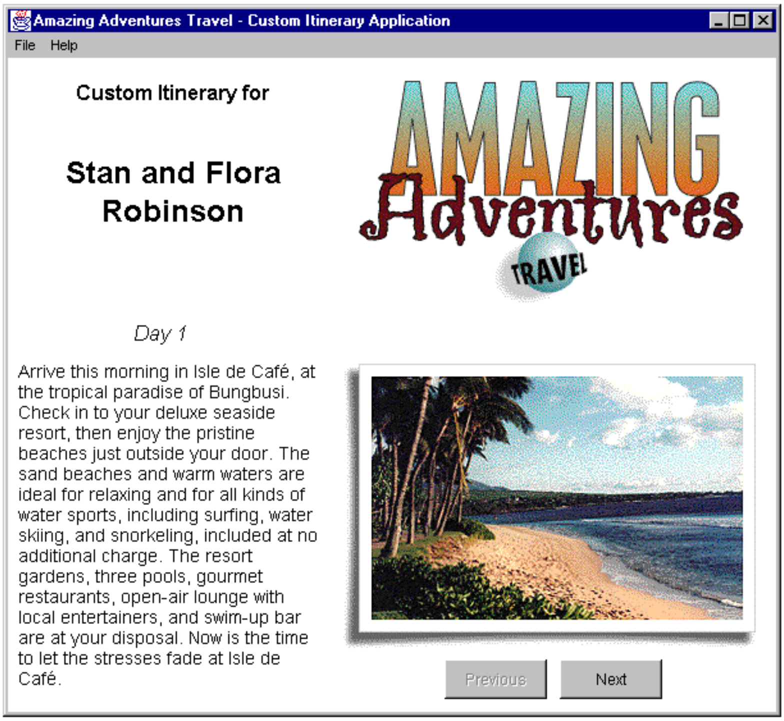
  The application appears.



To start, choose Open from the File menu and open your data file. Your .dat file and the graphics files must be in the same directory.

3 From the application, open Tour.dat, which is located in the Images directory.

The itinerary appears.



Custom Itinerary for

Stan and Flora Robinson

Day 1

Arrive this morning in Isle de Café, at the tropical paradise of Bungbusi. Check in to your deluxe seaside resort, then enjoy the pristine beaches just outside your door. The sand beaches and warm waters are ideal for relaxing and for all kinds of water sports, including surfing, water skiing, and snorkeling, included at no additional charge. The resort gardens, three pools, gourmet restaurants, open-air lounge with local entertainers, and swim-up bar are at your disposal. Now is the time to let the stresses fade at Isle de Café.

4 Press the Next and Previous buttons to look through the itinerary.

5 To exit the application, choose Exit from the File menu. Then click Yes in the dialog.

Now let's look at the application in Visual Cafe.

Looking at the application in Visual Cafe

Now that you've seen how the application works, let's see how it was created with Visual Cafe.

To look at the itinerary application in Visual Cafe, follow these steps:

1 If you don't have Visual Cafe running, start it now.

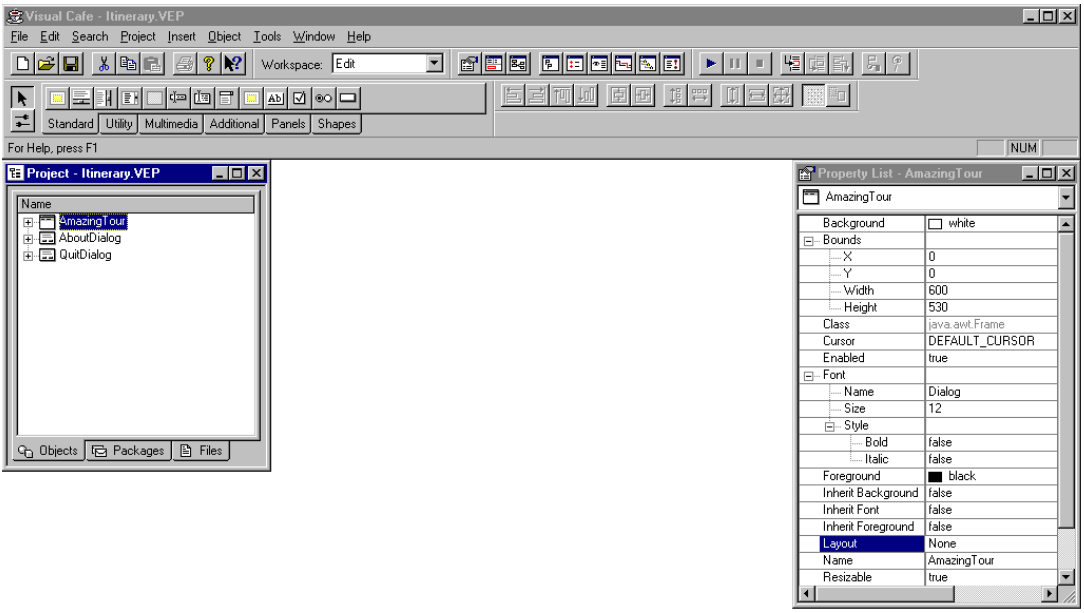
2 Close any open projects.

3 Choose Open from the File menu.

The Open dialog appears.

4 Select the Itinerary.vep project in the \Tour\Application folder, then click Open.

The project appears.



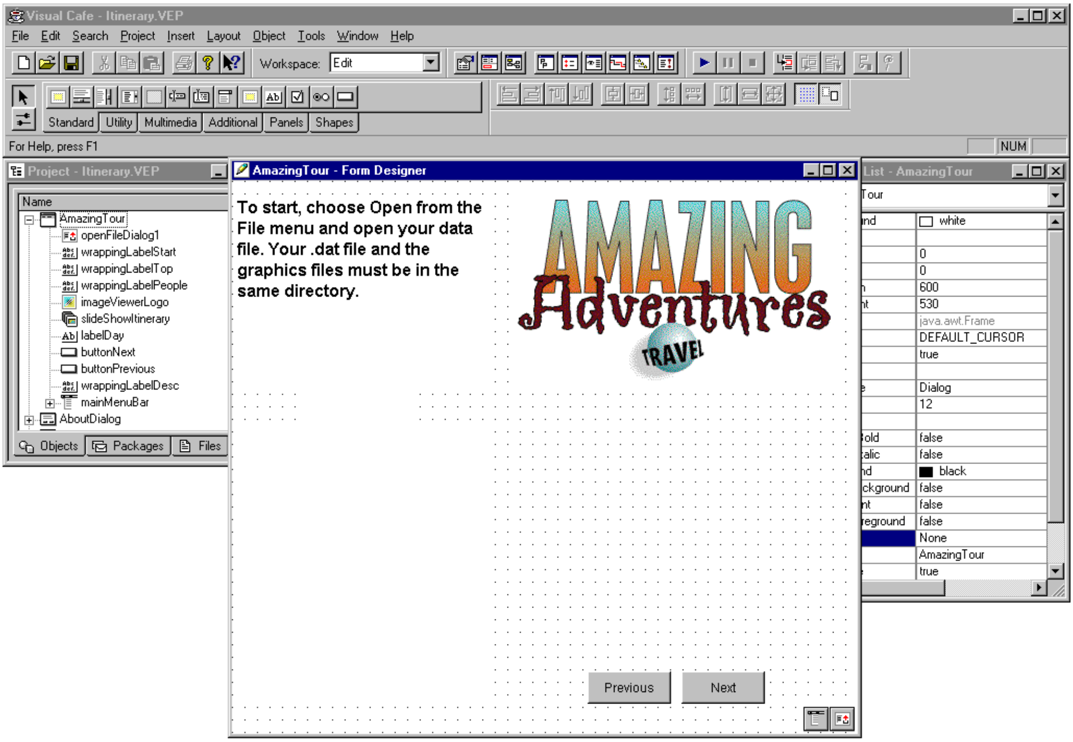
Background	□ white
Bounds	
X	0
Y	0
Width	600
Height	530
Class	java.awt.Frame
Cursor	DEFAULT_CURSOR
Enabled	true
Font	
Name	Dialog
Size	12
Style	
Bold	false
Italic	false
Foreground	■ black
Inherit Background	false
Inherit Font	false
Inherit Foreground	false
Layout	None
Name	AmazingTour
Resizable	true

You'll notice the following components in the Project window:

- AmazingTour — main application window
- AboutDialog — About dialog
- QuitDialog — Quit dialog

5 In the Project window, double-click AmazingTour.

The Form Designer appears.



To start, choose Open from the File menu and open your data file. Your .dat file and the graphics files must be in the same directory.

nd	□ white
	0
	0
h	600
ht	530
	java.awt.Frame
	DEFAULT_CURSOR
	true
e	Dialog
	12
old	false
alic	false
nd	■ black
ckground	false
nt	false
eground	false
	None
	AmazingTour
	true

The Project window lists the names of the components on the form. To see the components within AmazingTour, you can click the + next to AmazingTour.

6 Close your project by closing the Project window

Visual Cafe lets you launch your application from within its environment so you can test it. You'll try this feature later in the tour.

Now you can continue with the next chapter to build and test one of the applets.

Creating an Applet for the Web Page

4

This chapter quickly acquaints you with the main features of Visual Cafe by showing you how to create an applet for your Isle de Cafe Web site. This portion of the tour doesn't involve manually creating new Java code; instead, you use the visual design capabilities of Visual Cafe to create your complete applet. In the next chapter, you do work directly with Java code.

To learn how to add applets to your HTML page, you work through the following sections in this chapter:

- Creating and naming the project
- Creating the island map applet
- Displaying the applet in a Web page

Before working through this chapter, you should have completed the previous chapter "Starting the Tour."

When you first install Visual Cafe, you have the default environment options. If you've changed your environment options since you've installed the product, you might run into variations because of your settings. In particular, make sure the keyboard is set to vcafe. Choose Environment Options from the Tools menu to view your environment option settings.

Creating and naming the project

The project contains the elements needed in a Web page, a group of related Web pages, or an application, including applets, HTML files, application windows, and so on. To get started quickly, you can use project templates with skeleton applets and applications that you extend with additional functionality. You can also create custom templates and add them to the Component Library, and add your favorite templates to the Component Palette.

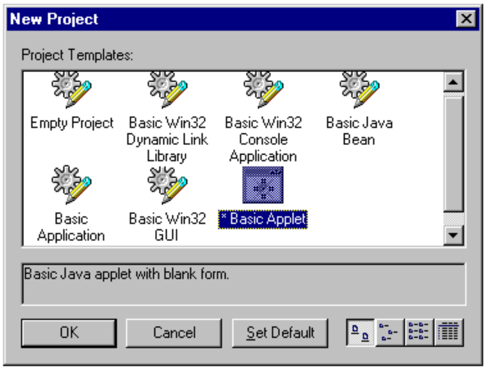
To create the project that will the hold the applets and HTML file for your Isle de Cafe Web page, follow these steps:

1 Close any open projects by clicking the Project window, then choosing Close Project in the File menu.

When you first start Visual Cafe, the last project you opened or an unnamed project, based on the default project template, appears. Here you'll create a new project so you can see how to choose a template.

2 Choose New Project from the File menu.

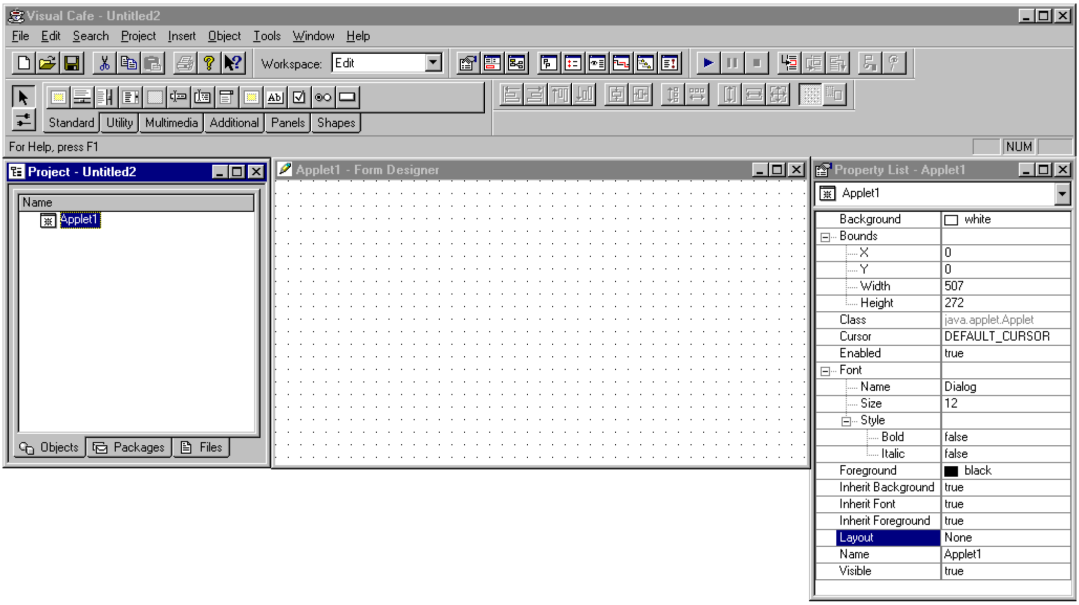
The New Project dialog box appears.



Basic Java applet with blank form.

3 Select the Basic Applet project template. Then click OK.

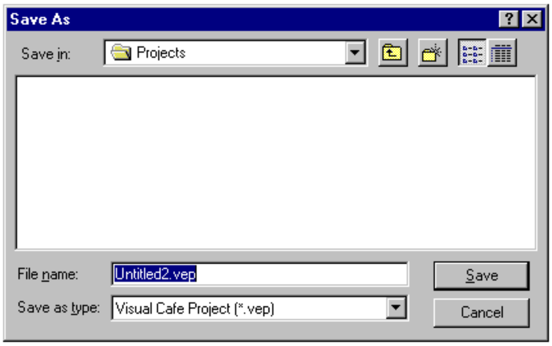
The Project and Form Designer windows appear.



Background	□ white
Bounds	
X	0
Y	0
Width	507
Height	272
Class	java.applet.Applet
Cursor	DEFAULT_CURSOR
Enabled	true
Font	
Name	Dialog
Size	12
Style	
Bold	false
Italic	false
Foreground	■ black
Inherit Background	true
Inherit Font	true
Inherit Foreground	true
Layout	None
Name	Applet1
Visible	true

4 Click the Project window, then choose Save As from the File menu.

The Save As dialog box appears.



Save in: Projects

File name: Untitled2.vep

Save as type: Visual Cafe Project (*.vep)

5 Create a new directory structure in the Visual Cafe directory, such as \MyApps\Applet, to hold your applet files.

The project and all of the files contained within it should be in the same directory. For easier project management and to prevent filename conflicts, you should save each project in its own directory.

6 Type MyIslePg.vep in the File name field, then click Save.

The new name appears in the titlebar of the Project window. Visual Cafe Project files have a .vep extension.

The first time you save a project, all of the files it contains are also saved to the new directory. After you save it once, saving just the project does not save other files, such as applets. Save All does save all files.

Creating the island map applet

To create the applet, you work through the following tasks:

- Gathering the files you need
- Setting the applet properties and adding the map
- Adding the HTML links
- Adding the volcano eruption animation
- Running the applet

Gathering the files you need

To construct your Web page, you need an HTML file that your applet will run within and the graphics files that your applet uses. You will copy an HTML file into your project directory. You will copy the graphics files into an Images directory subordinate to the project directory.

Visual Cafe lets you specify URLs that are relative to the applet directory. That way you can move your directory to another location and avoid the problems associated with hardcoded file specifications.

Now you can copy the files by using Windows operating system commands:

1 Copy Isle.html from the \Tour\Applet directory to your project directory.

2 Create an Images directory in your project directory.

3 Copy these files from the \Tour\Applet\Images directory to your new Images directory:

   - Bungbusi.gif
   - Erupt01.gif through Erupt16.gif
   - Goall.gif

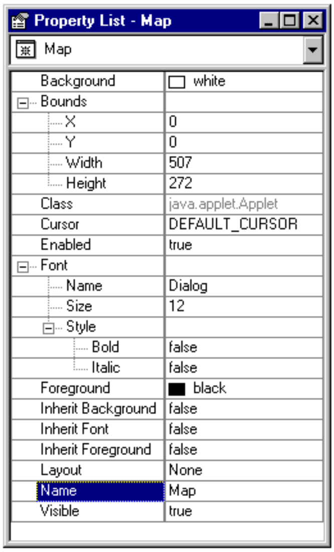
- Goall2.gif
- Map.gif
- Moderna.gif
- Photo_1.gif through Photo_8.gif
- Simerili.gif
- Venduca.gif

Setting the applet properties and adding the map

To start, you need to customize the properties of your form. Then you can add a picture of the island to the form. Follow these steps:

1. In the Property List, for Applet1, set the Name property to **Map** and the Inherit Background, Font, and Foreground properties to **false**.

   When you are finished, the Property List will look like this:

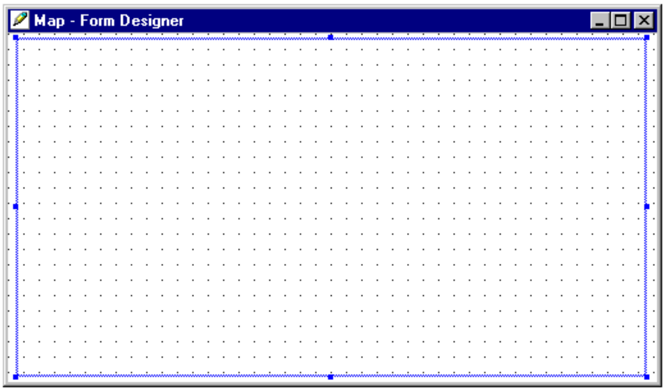


Background	□ white
⊟ Bounds	
X	0
Y	0
Width	507
Height	272
Class	java.applet.Applet
Cursor	DEFAULT_CURSOR
Enabled	true
⊟ Font	
Name	Dialog
Size	12
⊟ Style	
Bold	false
Italic	false
Foreground	■ black
Inherit Background	false
Inherit Font	false
Inherit Foreground	false
Layout	None
Name	Map
Visible	true

2. In the Component Palette, click the Multimedia tab, then click ImageViewer.

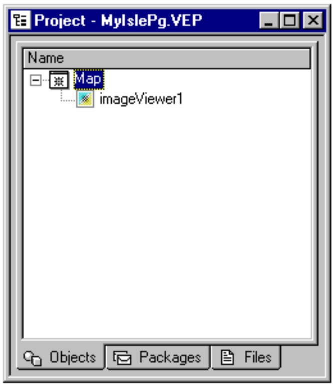
   ToolTips shows you which button is for the ImageViewer component. Move the cursor over a button until a pop-up message appears.

3 Draw a rectangle (about as large as the island map) on the applet form by clicking and dragging the cursor.



You can make the Form Designer window bigger by dragging the right side or bottom of the window. The components are positioned on the window relative to the top and left sides, so this spacing remains the same unless you move the components in the window.

The component you just added to the applet appears in the Project window.

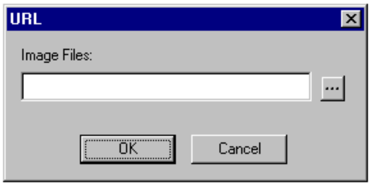


While the component is selected on the form, the Property List displays the properties of this component.

4 In the Property List, change the Name property to **imageViewerMap**.

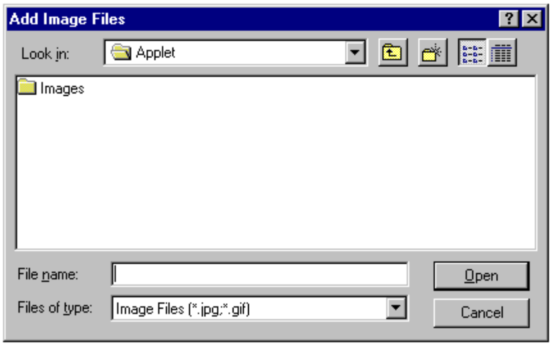
5 Double-click the URL property.

A URL dialog box appears.



6 Click the ... button.

An Add Image Files dialog box appears.


Look in: Applet

Files of type: Image Files (*.jpg;*.gif)

7 Navigate to your Images directory, select Map.gif, and click Open.

The complete path appears in the URL dialog box.

8 Delete the first part of the URL, so the only path remaining is Images/Map.gif.

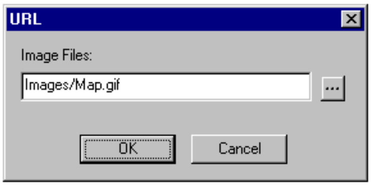
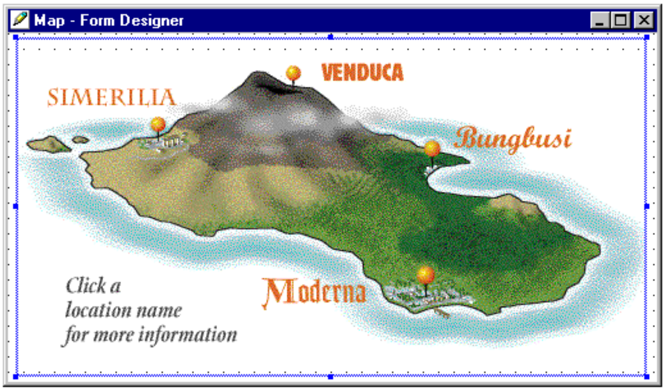


Image Files: Images/Map.gif

This creates the relative URL.

9 Click OK.

The map appears in the Form Designer.



10 Resize the form and the component as needed, so you can see the entire image.

When a component is selected and you move the cursor over a square (called a handle) on the edge of a component, the cursor changes to a two-way arrow, which means you can click and drag to resize the component.

↔

When a component is selected and you move the cursor over it, the cursor changes to a four-way arrow, which means you can click and drag the component to another location.

✥

The Form Designer window size represents the applet size. So when you run the applet, it will appear the same size as the window.

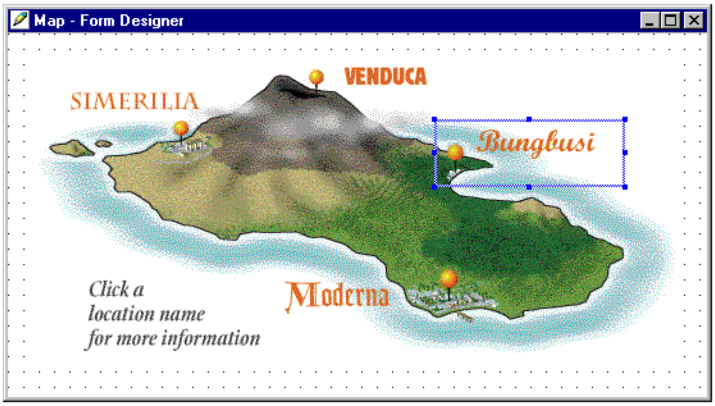
11 Click the Project or Form Designer window, then choose Save All from the File menu.

The project and all included files are saved.

Adding the HTML links

Visual Cafe lets you specify clickable areas on the map by drawing InvisibleHTMLLink components on it. These areas are associated with a location in an HTML file, so when the applet runs clicking the area will cause a jump to another location. Follow these steps to add these components:

1. In the Component Palette, click Additional, then click InvisibleHTMLLink.
2. Draw the component over the image, by clicking and dragging, so it contains the word Bungbusi and the location on the map.



3. In the Property List, change the Name property to **invisibleHTMLLinkB**.
4. In the Property List, double-click the HTML Link URL property.

   An HTML Link URL dialog box appears.
5. Type **Isle.html#Bungbusi** in the dialog box, then click OK.
6. Add InvisibleHTMLLink components for the remaining three locations. Use the following values for the Name and HTML Link URL properties:
   - Venduca — invisibleHTMLLinkV and Isle.html#Venduca
   - Simerilia — invisibleHTMLLinkS and Isle.html#Simerilia
   - Moderna — invisibleHTMLLinkM and Isle.html#Moderna

7 Click the Project or Form Designer window, then choose Save All from the File menu.

Now you can add the final component to the applet: an eruption animation.

Adding the volcano eruption animation

The volcano eruption is an added feature of the island map that should attract attention. Here are the steps to create the animation:

1 Select the island map on the form, then choose Send to Back from the Layout menu.

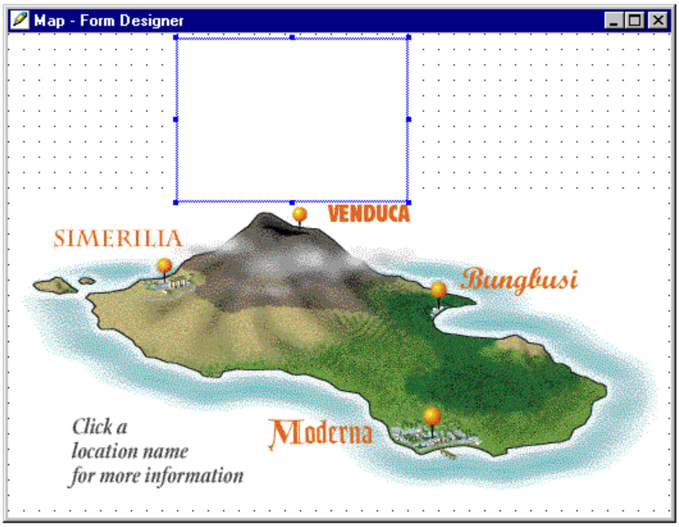
This puts the InvisibleHTMLLinks on top of the image. When you add components to a form, they are added after other components in the Project window. The order of components in the Project window determines how components overlap. After you choose Send to Back, the ImageViewer changes position in the Project window, indicating that it is now underneath the other components on the form.

2 Resize the form and move the components as needed so you can put the animation above the map.

Shift-click the image and each of the links, then drag to move them.

3 On the Component Palette, click Multimedia, then click Animator.

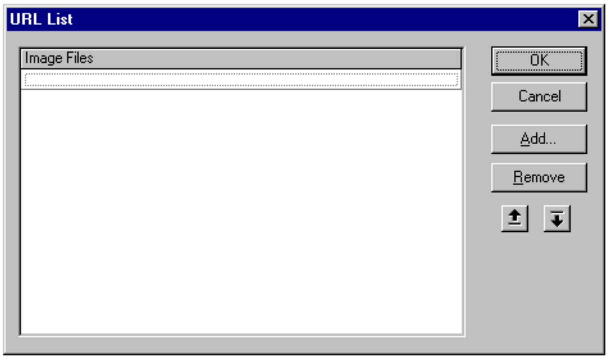
4 Draw a rectangle (about as large as the eruption) on the applet form by clicking and dragging the cursor.



5 In the Property List, set the Name property to **animatorErupt**.

6 In the Property List, double-click the URL List property of the Animator component.

A URL List dialog box appears.



7 Click Add, select the Erupt01.gif file in the File dialog box, then click Open.

A dialog box, asking if you want to add the files Erupt*.gif, appears.

8 Click Yes to add the sequence of files.

The files appear in the URL List dialog box.

9 For each file specification, double-click the row then remove the first part of each file specification up to the Images directory.

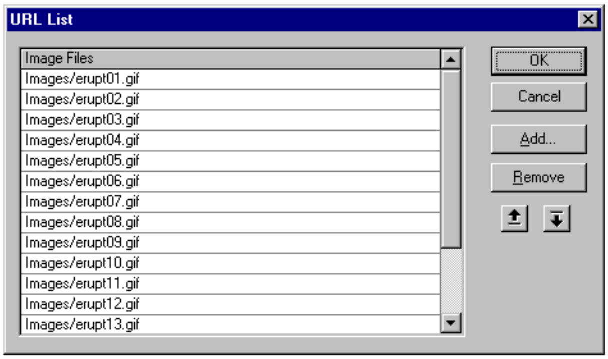
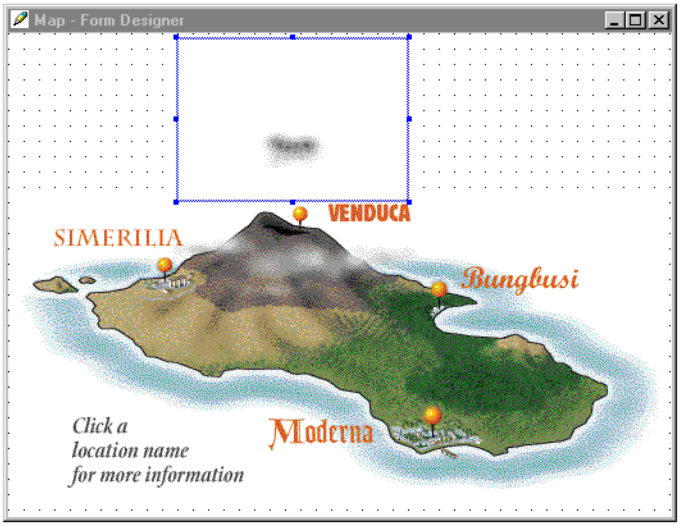


Image Files
Images/erupt01.gif
Images/erupt02.gif
Images/erupt03.gif
Images/erupt04.gif
Images/erupt05.gif
Images/erupt06.gif
Images/erupt07.gif
Images/erupt08.gif
Images/erupt09.gif
Images/erupt10.gif
Images/erupt11.gif
Images/erupt12.gif
Images/erupt13.gif

10 Click OK to close the dialog box.

An animation picture appears in the Form Designer.



11 Set Preview Component to **true** to view the animation.

12 Resize the Animator component as needed. Move the component until it's positioned where you want it in the applet.

13 Set Preview Component to **false**.

14 Click the Project or Form Designer window, then choose Save All from the File menu.

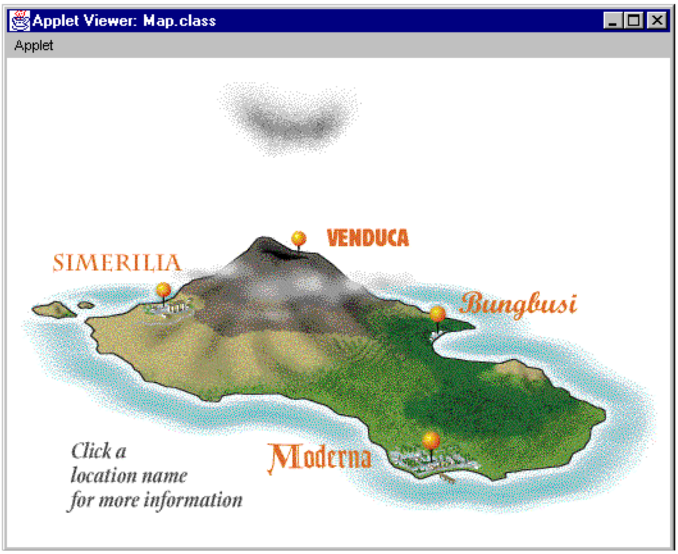
You have completed the animation and the applet, so you can close the Form Designer.

Running the applet

The integrated Applet Viewer enables you to view and test your working Java applets within the Visual Cafe environment. Now that you've created your applet, you can compile it and look at it while it's running:

1 Choose Execute from the Project menu to run the applet.

The applet appears in the Applet Viewer. Visual Cafe saves your source code and compiles your applet before displaying it.



When you choose Execute from the Project menu, the Visual Cafe bytecode compiler creates machine-independent bytecode instructions. Then the Symantec Just-In-Time (JIT) compiler translates these instructions into native machine instructions, which Visual Cafe runs for you. The machine instructions are stored temporarily in memory — no native executable is generated.

Visual Cafe's native bytecode compiler currently builds Java programs several times faster than the standard Sun Java compiler. The compiler also provides improved code optimizations and shows detailed error information in the Messages window. You can double-click an error message in this window to go to the line in the code where the error was generated.

The enhanced JIT compiler has superior execution speed, too. To start, it provides immediate applet and application execution by converting bytecodes on-the-fly to native code as needed. When compilation is complete, the JIT compiler seamlessly starts executing the native code.

2 Choose Quit from the Applet menu.

The Applet Viewer window closes.

Displaying the applet in a Web page

To test your applet, you need to view it in a Web page. This is the only way to try your HTML links. You will work through the following procedures to do so:

- Adding the HTML file to the project
- Adding the applet tag to the HTML file
- Executing the applet within the Web page

To be able to run your applet in a Web browser, your default Web browser must support JDK 1.1 and AWT 1.1.

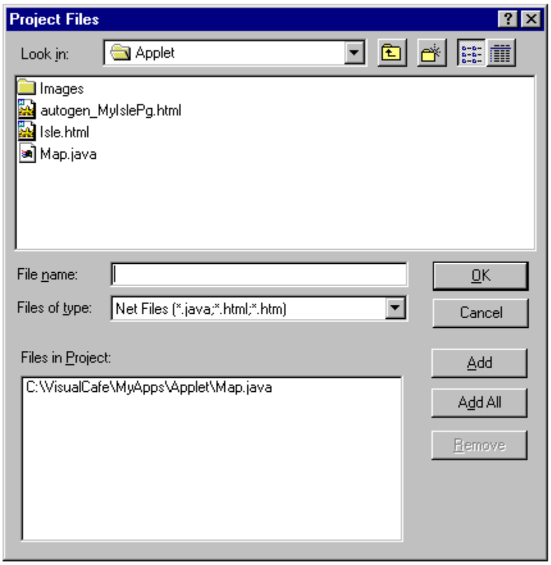
Adding the HTML file to the project

You can add HTML files to your project as an organizational tool, but it isn’t required to run an applet in a Web page. Visual Cafe lets you create HTML files in the Source window, or you can import existing HTML files into a project. You already have your HTML file, so you can just add it.

Follow these steps to add the file to the project:

1 From the Insert menu, choose Files into Project.

The Project Files dialog box appears.



Project Files

Look in: Applet

Images
autogen_MyIslePg.html
Isle.html
Map.java

File name:

Files of type: Net Files (*.java;*.html;*.htm)

Files in Project:

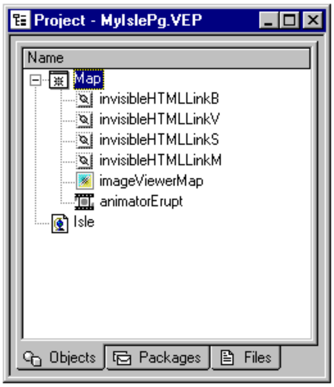
C:\VisualCafe\MyApps\Applet\Map.java

2 In the Project Files dialog box, select Isle.html in your project directory, then click Add.

The file appears in the Files in Project list.

3 Click OK to add the file to the project.

The file appears in the Project window.



Project - MyIslePg.VEP

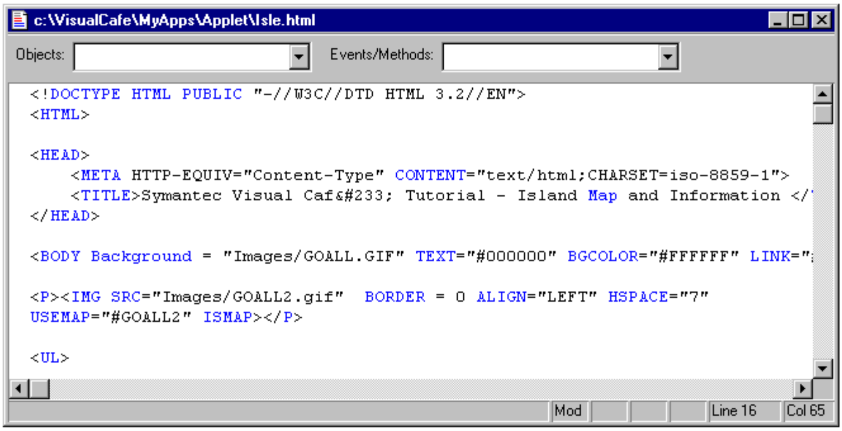
Name
Map
invisibleHTMLLinkB
invisibleHTMLLinkV
invisibleHTMLLinkS
invisibleHTMLLinkM
imageViewerMap
animatorErupt
Isle

Adding the applet tag to the HTML file

Now you're ready to add the applet you created in Visual Cafe to the HTML file:

1 To look at Isle.html, click the Files tab in the Project window, then double-click Isle.html.

The Source window appears. It contains the contents of Isle.html. You can edit the code in this window.



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">
<HTML>

<HEAD>
    <META HTTP-EQUIV="Content-Type" CONTENT="text/html;CHARSET=iso-8859-1">
    <TITLE>Symantec Visual Caf&#233; Tutorial - Island Map and Information </
</HEAD>

<BODY Background = "Images/GOALL.GIF" TEXT="#000000" BGCOLOR="#FFFFFF" LINK="

<P><IMG SRC="Images/GOALL2.gif"  BORDER = 0 ALIGN="LEFT" HSPACE="7"
USEMAP="#GOALL2" ISMAP></P>

<UL>
```

2 Find the following applet tag in the file:

```
<APPLET CODE="IsleMap.class" ARCHIVE="tour.jar" WIDTH=594
 HEIGHT=349 ALIGN=center>
   <blockquote>
   <hr>
   If you were using a Java-enabled browser, you would see an
   island map and volcano eruption instead of this paragraph.
   <hr>
   </blockquote>
</APPLET>
```

This tag causes the applet to run within the Web page. If the browser isn't Java-enabled, the text displays instead.

Whenever you create an HTML document and want to include an applet, you need to add an applet tag.

3 Change the name IsleMap.class to **Map.class**. If the applet tag contains the `ARCHIVE="tour.jar"` variable and value, remove it.

4 If different, change the width and height in the applet tag so it matches the width and height in the Property List for your Map applet.

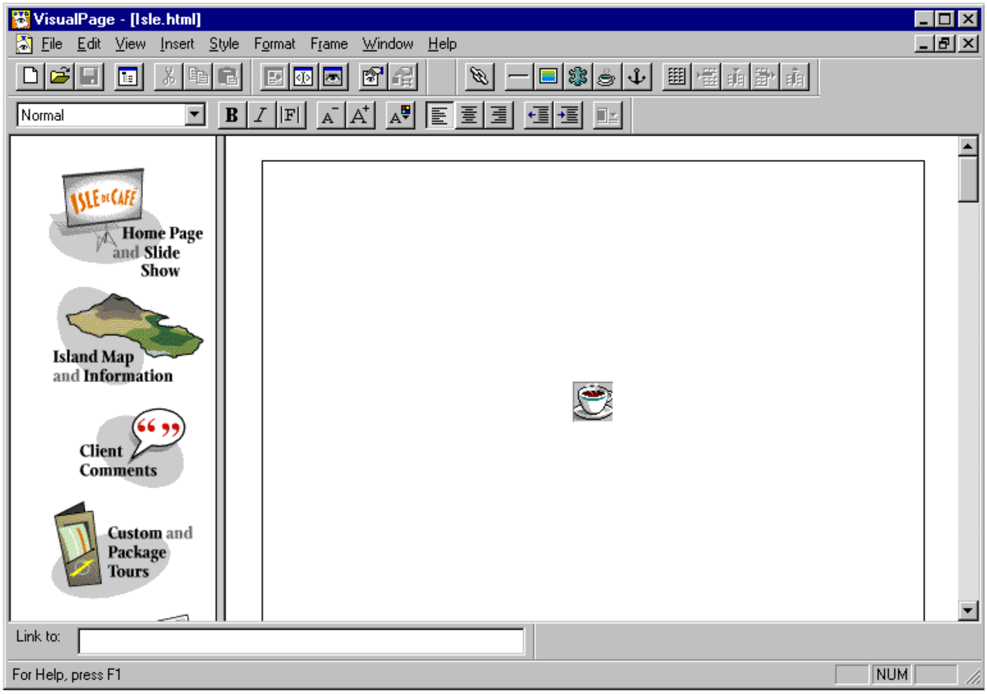
5 Close the Source window, then save any changes.

Executing the applet within the Web page

Visual Cafe lets you run the applet directly within the Web page by using Visual Page or your default Web browser. The advantage of using Visual Page is you can also modify your HTML files as needed. Follow these steps to open your Web page in Visual Page:

1 In the Objects view of the Project window, double-click Isle.html.

If you have properly installed Visual Page from the Visual Cafe CD, Visual Page starts and displays your Web page.



2 Choose Preview from the View menu.

The applet runs in Visual Page.

3 Try out your applet.

Some browsers use a different z-order when determining how components overlap in an applet. When run in a browser, the InvisibleHTMLLink must be on top for users to be able to click it. To ensure compatibility with different browsers, it is a good idea to "sandwich" InvisibleHTMLLinks on top and beneath the image they overlap. For now, if you are unable to use the InvisibleHTMLLinks, close the Web page, select the island map on the form, then choose Send to Back from the Layout menu and view the Web page in Visual Page again.

4 Exit Visual Page.

The window closes.

If your applet works properly in Visual Page, you have completed the applet. Continue with the next chapter to create the application.

Creating the Application and Debugging Code

This chapter builds on the previous chapter, "Creating an Applet for the Web Page." It teaches you some advanced features that you may want to use particularly if you have more programming experience and want to take advantage of all the diverse capabilities of Visual Cafe. In this chapter, you create the custom itinerary application that your clients can use to view their custom tour itineraries. You also try some of Visual Cafe's debugging features.

Creating the itinerary application

To create the itinerary application, work through the following tasks:

- Opening and naming the new project
- Gathering the files you need
- Setting the properties of Frame1 and adding the SlideShow component
- Adding the Button components to the frame
- Adding the Amazing Adventures Travel logo
- Adding the labels
- Specifying component interactions
- Adding custom Java code
- Customizing the menu bar
- Customizing the About dialog
- Customizing the Quit dialog
- Running the application

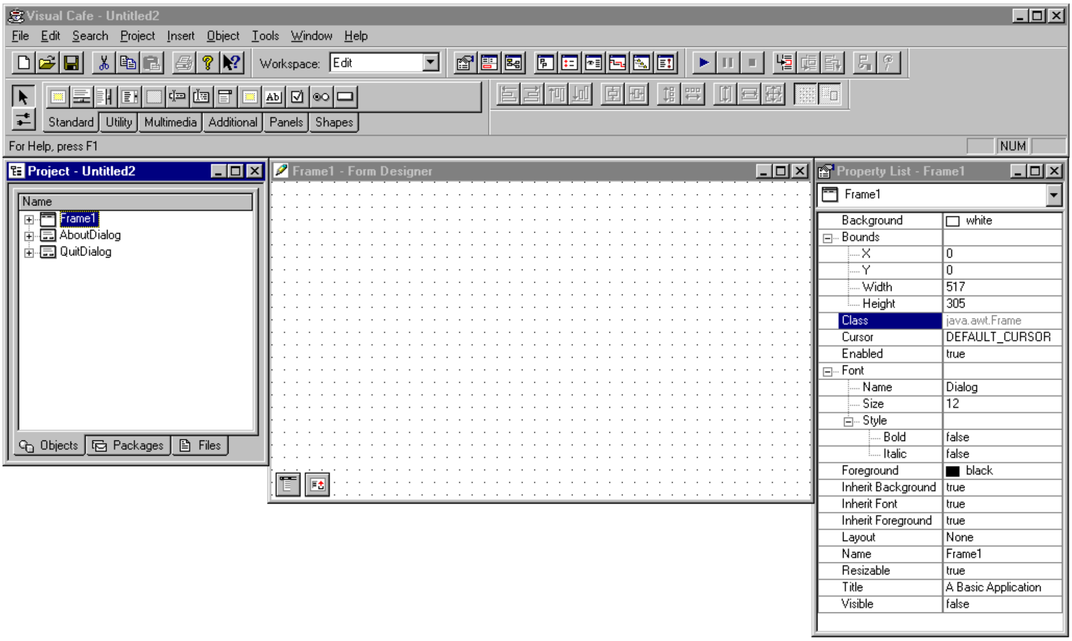
Opening and naming the new project

You should create a separate project for your application and save it in its own directory. Follow these steps:

1. Close any open projects.
2. Choose New Project from the File menu.

   The New Project dialog box appears.
3. Select the Basic Application project template. Then click OK.

   The Project and Form Designer windows appear.



Background	□ white
Bounds	
X	0
Y	0
Width	517
Height	305
Class	java.awt.Frame
Cursor	DEFAULT_CURSOR
Enabled	true
Font	
Name	Dialog
Size	12
Style	
Bold	false
Italic	false
Foreground	■ black
Inherit Background	true
Inherit Font	true
Inherit Foreground	true
Layout	None
Name	Frame1
Resizable	true
Title	A Basic Application
Visible	false

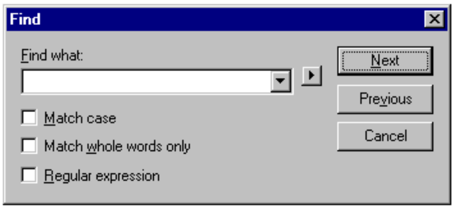
A standard application project contains a Java frame (a special kind of window) with a predefined menu bar, an About dialog, and a Quit dialog. Two objects already included in Frame1 are mainMenuBar and openFileDialog1, which are represented by icons on the Form Designer. If you select an icon on the Form Designer, it is identified in the Property List.

4. Double-click the Form Designer.

   The Source window appears.

5 Choose Find from the Search menu.

The Find dialog box appears.



6 Type **void main**, then click Next.

The word is highlighted in the Java code. When you use the frame in the application template, it's already set to be the main application window. If you do not use the frame in the template, you need to add the main and show methods to your main application window.

7 Close the Source window.

8 Click the Project window, then choose Save As from the File menu.

The Save As dialog box appears.

9 Create a new directory structure, such as \MyApps\Application, to hold your application files.

The project and all of the files contained within it should be in the same directory. For easier project management and to prevent filename conflicts, you should save each project in its own directory.

10 Navigate to your project directory, type MyIslePr.vep in the File name field, then click Save.

The new name appears in the titlebar of the Project window, and all the files in the project are saved to this directory.

Gathering the files you need

To use relative URLs, you will store your graphics files in a directory subordinate to the project directory as follows:

1. Using Windows operating system commands, copy the \Tour\Application\Images directory.
2. Paste the directory in your project directory.

   The Images directory contains the following files:

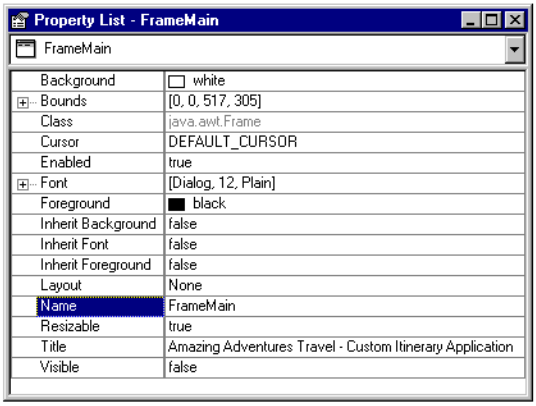
   - Logo.gif
   - Photo_1.gif through Photo_8.gif
   - Tour.dat

Setting the properties of Frame1 and adding the SlideShow component

The user interface of your application can be put together quickly with the SlideShow component, which associates a picture with text and allows you to navigate through sequential photos with button clicks.

Follow these steps to add the SlideShow component to the frame and set its properties:

1. Set the Frame1 properties, including Name, Title, and Inherit Background, Font, and Foreground, so it appears as follows:



Property List - FrameMain

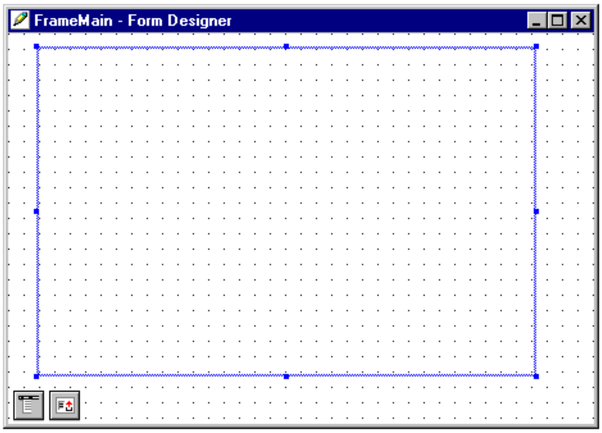
FrameMain

Background	□ white
⊞ Bounds	[0, 0, 517, 305]
Class	java.awt.Frame
Cursor	DEFAULT_CURSOR
Enabled	true
⊞ Font	[Dialog, 12, Plain]
Foreground	■ black
Inherit Background	false
Inherit Font	false
Inherit Foreground	false
Layout	None
Name	FrameMain
Resizable	true
Title	Amazing Adventures Travel - Custom Itinerary Application
Visible	false

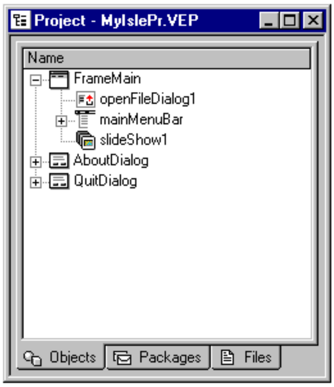
2 On the Component Palette, click Multimedia, then click SlideShow.

ToolTips shows you which button is for the SlideShow component. Move the cursor over a button until a pop-up message appears.

3 Draw a rectangle (about as large as the photos) on the form by clicking and dragging the cursor.



While the component is selected on the form, the Property List displays the properties of this component. The Project window shows the new component.

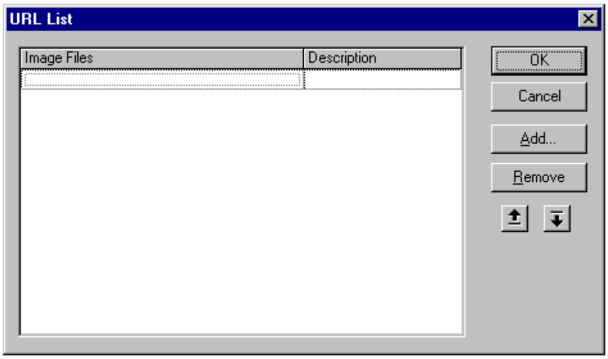


4 In the Property List, change the Name property to **slideShowItinerary**.

5 Resize the slide show until it's approximately 340 pixels in width by 235 pixels in height by typing in the Property List.

6 In the Property List, double-click the URL List property of the SlideShow component.

The URL List dialog box appears. Normally, you would enter photo names and text descriptions here, but for your application you will read in a file that sets these properties at runtime.



7 Click Cancel.

8 Click the Project or Form Designer window, then choose Save All from the File menu.

Adding the Button components to the frame

The slide show is controlled by button clicks. You need to add the buttons and set their properties:

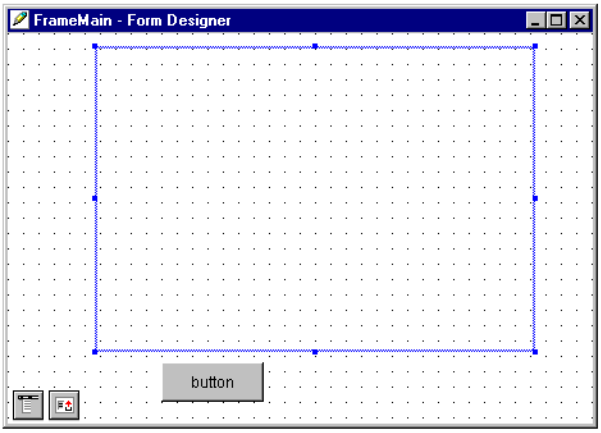
1 Resize the Form Designer window so two buttons can fit under the SlideShow component. Move the component until it's positioned where you want it in the frame.

You can make the Form Designer window bigger by dragging the right side or bottom of the window. The components are positioned on the window relative to the top and left sides, so this spacing remains the same unless you move the components in the window.

The Form Designer window size represents the Java frame container size. The frame size is the same as the window size. So when you run the application, it will appear the same size as the window.

2 While the Form Designer is the active window, click the Standard tab on the Component Palette, then click Button.

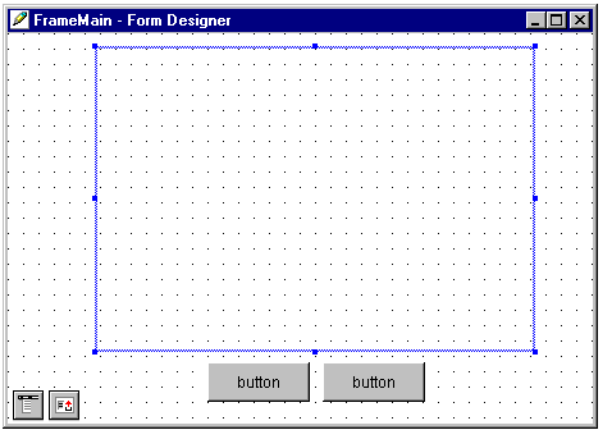
3 Draw a button under the SlideShow component by clicking and dragging.



4 To copy and paste this button, select the button and choose Copy from the Edit menu, then choose Paste from the Edit menu. You can also choose these menu items from the pop-up menu that appears when you right-click.

A second button appears.

5 Position the buttons under the SlideShow component as follows:



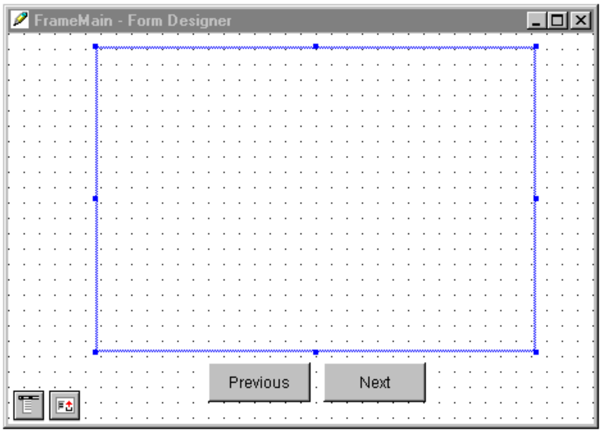
6 Select the button on the left, then set the following properties:

- Name — buttonPrevious
- Label — Previous
- Visible — false (because you don't want the button to appear until you load a data file)

7 Select the button on the right, then set the following properties:

- Name — buttonNext
- Label — Next
- Visible — false

The buttons now both have names.



8 Click the Project or Form Designer window, then choose Save All from the File menu.

Adding the Amazing Adventures Travel logo

Now you can add your company logo:

1 In the Property List, resize FrameMain to 600 pixels in width and 530 pixels in height.

The form size changes. If you have a small screen, you can make FrameMain smaller.

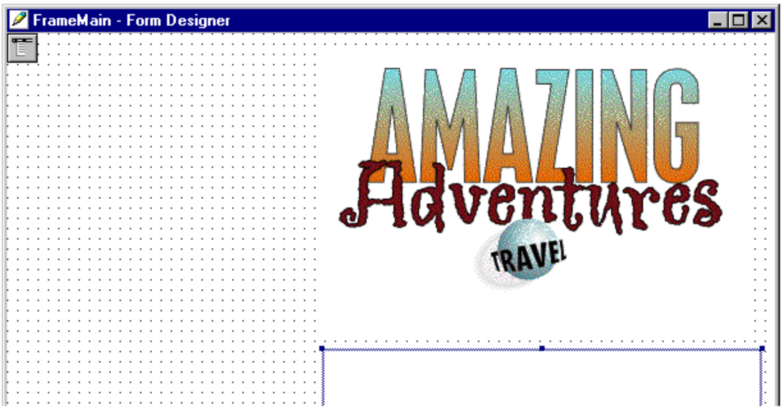
2 Shift-click the SlideShow component, then each button, to select them.

3. Drag the components to the bottom right corner of the form.
4. In the Component Palette, click Multimedia, then click ImageViewer.
5. On the form, draw a rectangle above the SlideShow component (a little less wide than the slide show) by clicking and dragging the cursor.
6. In the Property List, change the Name property to **imageViewerLogo**.
7. Double-click the URL property.

   A URL dialog box appears.
8. Click the ... button.

   An Add Image Files dialog box appears.
9. Navigate to your Images directory, select Logo.gif, and click Open.
10. To create a relative URL, delete the file specification so it displays just Images/Logo.gif. Then click OK in the URL dialog box.

    The logo appears in the Form Designer.



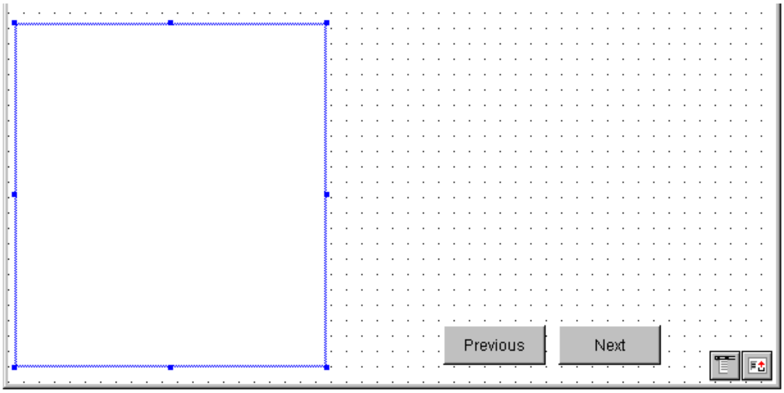
11. Click the Project or Form Designer window, then choose Save All from the File menu.

Adding the labels

Your application displays text, so you need to add text fields to the form. Follow these steps:

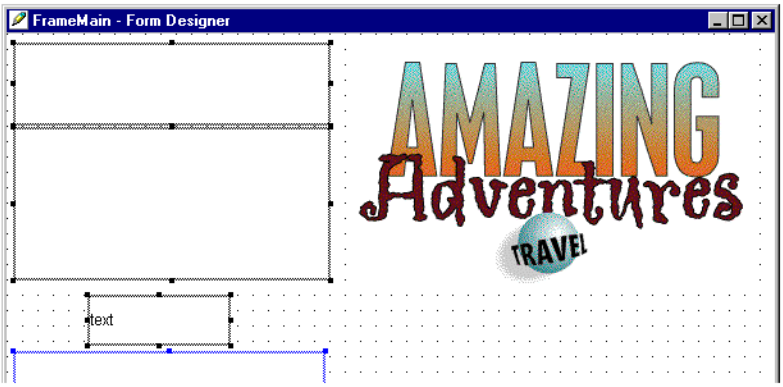
1. In the Component Palette, click the Additional tab, then click WrappingLabel.
2. Draw a text box to the left of the SlideShow component.

   The dimensions can be about 240 pixels in width by 265 pixels in height.

   You can choose Grid Options from the Layout menu to turn Snap to grid on or off.
3. In the Property List, set the following properties:
   - Name — wrappingLabelDesc
   - Font Size — 14
   - Font Style Bold — false

4 To the left of the picture, draw two WrappingLabel components and underneath a Label component (in the Standard tab of the Component Palette) as shown.



5 In the Property List, set the properties of the top WrappingLabel component as follows:

- Name — wrappingLabelTop
- Font Size — 16
- Font Style Bold — true
- Text Alignment — ALIGN_CENTERED
- Text — Custom Itinerary for

To display the properties in the Property List for this component, either select the component on the form or choose it from the drop-down list in the Property List window.

6 Set the properties of the middle WrappingLabel component as follows:

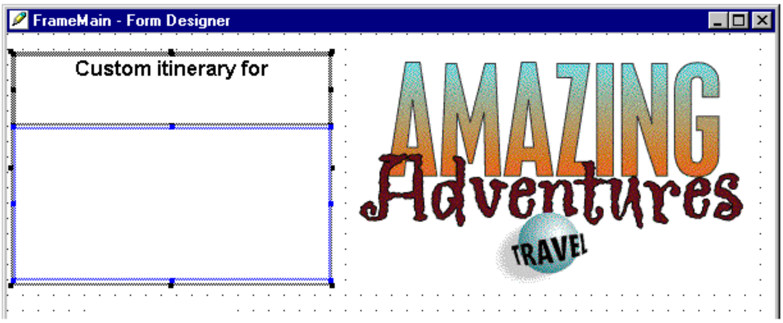
- Name — wrappingLabelPeople
- Font Size — 24
- Font Style Bold — true
- Text Alignment — ALIGN_CENTERED

7 Set the properties of the Label component as follows:

- Name — labelDay
- Font Size — 16
- Font Style Bold — false
- Font Style Italic — true
- Alignment — CENTER

Also, delete the text in the Text field.

8 Now draw a WrappingLabel component over the top two labels as follows:



9 Set the properties of the new WrappingLabel component as follows:

- Name — wrappingLabelStart
- Font Size — 16
- Font Style Bold — true
- Text — To start, choose Open from the File menu and open your data file. Your .dat file and the graphics files must be in the same directory.

10 Select wrappingLabelStart on the Form Designer, then choose Bring to front from the Layout menu.

This ensures that the label is on top of the other labels. Because the labels overlap, an easy way to select wrappingLabelStart is to choose it from the Property List drop-down menu.

11 Click the Project or Form Designer window, then choose Save All from the File menu.

Specifying component interactions

One of the most powerful features of Visual Cafe is the ability to create interactions between components. The Interaction Wizard lets you graphically build relationships between components, or between a component and itself (for example, double-clicking an item in a list box could remove the item from the list box and add it to another list box). These relationships specify the actions to take when an event is triggered on a component. For example, in a slide show, a button click could cause the next photo to be displayed. Visual Cafe automatically generates the necessary Java code for the specified relationship.

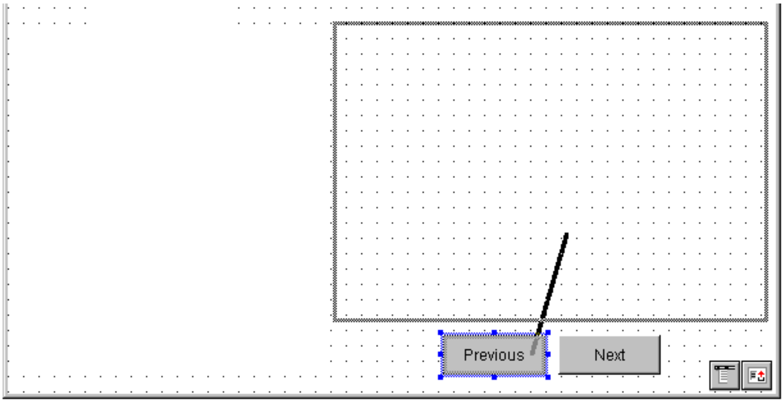
In the following sections you specify interactions between components on the application form:

- Setting up the Previous button to display the previous image
- Setting up the Next button to display the next image
- Disabling the Next button at the end of the slide show
- Disabling the Previous button at the start of the slide show
- Making the slide show descriptions appear in the wrapping labels
- Setting the Day label

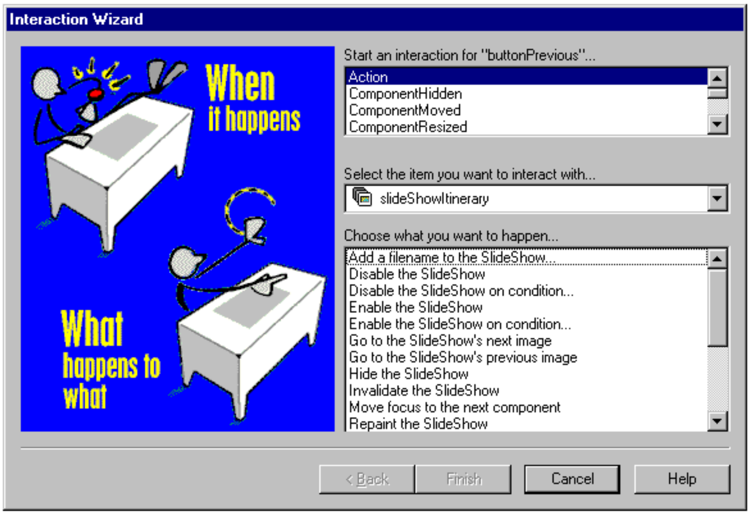
Setting up the Previous button to display the previous image

Here you set up an interaction between the Previous button and the SlideShow component so that a button click causes the previous image to be displayed. Follow these steps:

1 Click the Interaction Wizard button, then click the Previous button and drag a line to the SlideShow component.



The Interaction Wizard appears.



Interaction Wizard

Start an interaction for "buttonPrevious"...

Action
ComponentHidden
ComponentMoved
ComponentResized

Select the item you want to interact with...

slideShowItinerary

Choose what you want to happen...

Add a filename to the SlideShow...
Disable the SlideShow
Disable the SlideShow on condition...
Enable the SlideShow
Enable the SlideShow on condition...
Go to the SlideShow's next image
Go to the SlideShow's previous image
Hide the SlideShow
Invalidate the SlideShow
Move focus to the next component
Repaint the SlideShow

2 Select the following items:

Start an interaction for “buttonPrevious”: Action
Select the item you want to interact with: slideShowItinerary
Choose what you want to happen: Go to the SlideShow’s previous image

3 Click Finish.

Setting up the Next button to display the next image

Here you set up an interaction between the Next button and the SlideShow component so a button click causes the next image to be displayed:

1 Click the Interaction Wizard button, then click the Next button and drag a line to the SlideShow component.

The Interaction Wizard appears.

2 Select the following items:

Start an interaction for “buttonNext”: Action
Select the item you want to interact with: slideShowItinerary
Choose what you want to happen: Go to the SlideShow’s next image

3 Click Finish.

Disabling the Next button at the end of the slide show

When you’re at the last picture, you want to indicate that by disabling the Next button:

1 Click the Interaction Wizard button, then click the SlideShow component and drag a line to the Next button.

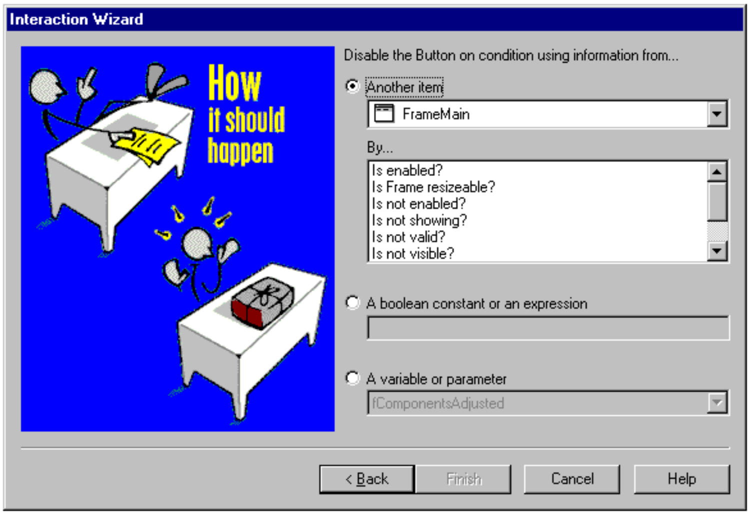
The Interaction Wizard appears.

2 Select the following items:

Start an interaction for “slideShowItinerary”: SlideChanged
Select the item you want to interact with: buttonNext
Choose what you want to happen: Disable the Button on condition

3 Click Next.

The second page of the dialog box appears.



Interaction Wizard

Disable the Button on condition using information from...

● Another item

FrameMain

By...

Is enabled?
Is Frame resizeable?
Is not enabled?
Is not showing?
Is not valid?
Is not visible?

○ A boolean constant or an expression

○ A variable or parameter

fComponentsAdjusted

4 While **Another item** is selected, choose slideShowItinerary and select Is at the last image?

5 Click Finish.

Disabling the Previous button at the start of the slide show

When you're at the first picture, you want to indicate that by disabling the Previous button:

1 Click the Interaction Wizard button, then click the SlideShow component and drag a line to the Previous button.

The Interaction Wizard appears.

2 Select the following items:

Start an interaction for "slideShowItinerary": SlideChanged
Select the item you want to interact with: buttonPrevious
Choose what you want to happen: Disable the Button on condition

3 Click Next.

The second page of the dialog box appears.

4 While **Another item** is selected, choose slideShowItinerary and select Is at the first image?

5. Click Finish.
6. Click the Project or Form Designer window, then choose Save All from the File menu.

Making the slide show descriptions appear in the wrapping labels

You need to set up an interaction between the SlideShow component and a WrappingLabel component so the descriptive text associated with a picture displays in the text field. Follow these steps:

1. Click the Interaction Wizard button, then click the SlideShow component and drag a line to the wrappingLabelDesc component.

   The Interaction Wizard appears.

2. Select the following items:

   Start an interaction for “slideShowItinerary”: SlideChanged
   Select the item you want to interact with: wrappingLabelDesc
   Choose what you want to happen: Set the text for WrappingLabel

3. Click Next.

   The second page of the dialog box appears.

4. While **Another item** is selected, choose slideShowItinerary and select Get current description.
5. Click Finish.

Setting the Day label

Now you’ll set up the incrementing and decrementing Day field:

1. Click the Interaction Wizard button, then click the Next button and drag a line to the labelDay component.

   The Interaction Wizard appears.

2. Select the following items:

   Start an interaction for “buttonNext”: Action
   Select the item you want to interact with: labelDay
   Choose what you want to happen: Set the text for Label

3. Click Next.

   The second page of the dialog box appears.

4. Select A String constant or an expression.

5. Deselect Add quotes.
6. Type this expression (don’t forget the blank spaces):

   ```
   "Day " + (++dayCount)
   ```

7. Click Finish.
8. Click the Interaction Wizard button, then click the Previous button and drag a line to the labelDay component.

   The Interaction Wizard appears.

9. Select the following items:

   Start an interaction for “buttonPrevious”: Action
   Select the item you want to interact with: labelDay
   Choose what you want to happen: Set the text for Label

10. Click Next.

    The second page of the dialog box appears.

11. Select A String constant or an expression.
12. Deselect Add quotes.
13. Type this expression:

    ```
    "Day " + (--dayCount)
    ```

14. Click Finish.
15. Click the Project or Form Designer window, then choose Save All from the File menu.

Now it’s time to add the custom code. You can close the Form Designer if you want.

Adding custom Java code

Java code, including complete applets and applications, that was not created in Visual Cafe can be added to a project. If the code contains visual aspects, Visual Cafe translates the code into a visual representation that’s displayed in the Form Designer.

Visual Cafe has many features to assist you with creating new Java code. For example, when you want to handle an event for a component, the Source Editor helps you. Select the component in the Object pull-down menu, then select the event you want to handle from the Events/Methods pull-down menu. Visual Cafe creates a method for that event and puts a call to that method in the event handler. You fill in the body of the method.

Now that Visual Cafe has created the main portion of your program, you can add custom code to read data from a file at runtime and to increment the Day label.

The data file is a text file in the following format:

client_names
number_of_pictures
picture1_filename
picture1_text
picture2_filename
picture2_text
...

Here's an easy way to add the code:

1 Choose Open from the File menu, and open \Tour\Application \AmazingTour.java.

The file appears in the Source window. The custom code is identified by comments directly in the code.

2 In the Project window, click FrameMain, then right-click. Choose Edit Source from the pop-up menu.

FrameMain.java appears in another Source window.

3 Copy the custom code from AmazingTour.java into your application.

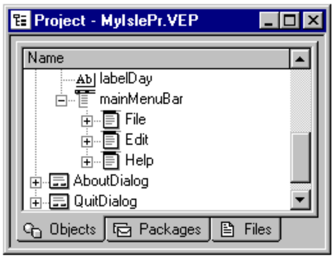
You can search for the words "Custom code" in AmazingTour.java to find the code quickly. There are several places where you need to copy the code. Make sure you insert the code into your application directly under the same line that appeared above it in AmazingTour.java.

4 When you're finished, click the FrameMain Source window and choose Save All. Then close the Source windows.

Customizing the menu bar

Follow these steps to modify the default menu bar in the application:

1 In the Project window, display the contents of mainMenuBar (subordinate to FrameMain) by clicking the +.



2 In the Project window, select Edit. Then choose Delete from the Edit menu.

3 In the Project window, display the contents of the File menu, then delete New, Save, and SaveAs.

4 Click the Project window, then choose Save All from the File menu.

Customizing the About dialog

The About dialog is included as part of the application template. Follow these steps to modify it:

1 Double-click the AboutDialog object in your project.

The Form Designer appears.

2 Change the properties as follows:

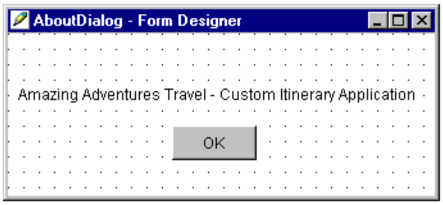
- Inherit Font — false
- Inherit Background — false
- Inherit Foreground — false

3 Click the Label component in the Form Designer.

4 In the Property List, click the Text property value to highlight it, then type **Amazing Adventures Travel - Custom Itinerary Application**.

The text appears in the Label.

5 Click the Form Designer, then resize and reposition the label, button, and form as needed.


AboutDialog - Form Designer
Amazing Adventures Travel - Custom Itinerary Application
OK

6 Click the Project or Form Designer window, then choose Save All from the File menu.

The About dialog is saved in the project.

7 Close the Form Designer for the About dialog.

Customizing the Quit dialog

The Quit dialog is also included as part of the application project template. Follow these steps to modify it:

1 Double-click the QuitDialog object in your project.

The Form Designer appears.

2 Change the properties as follows:

- Inherit Font — false
- Inherit Background — false
- Inherit Foreground — false

3 Click the Project or Form Designer window, then choose Save All from the File menu.

The Quit dialog is saved in the project.

4 Close the Form Designer for the Quit dialog.

Running the application

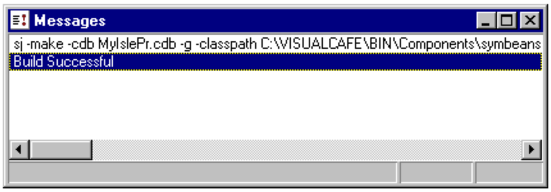
Now that you've specified the user interface and other aspects of your application, you're ready to run it. This is how you run the application from within Visual Cafe:

1. Choose Messages from the Window menu.

   The Messages window appears.

2. Choose Execute from the Project menu to run the application.

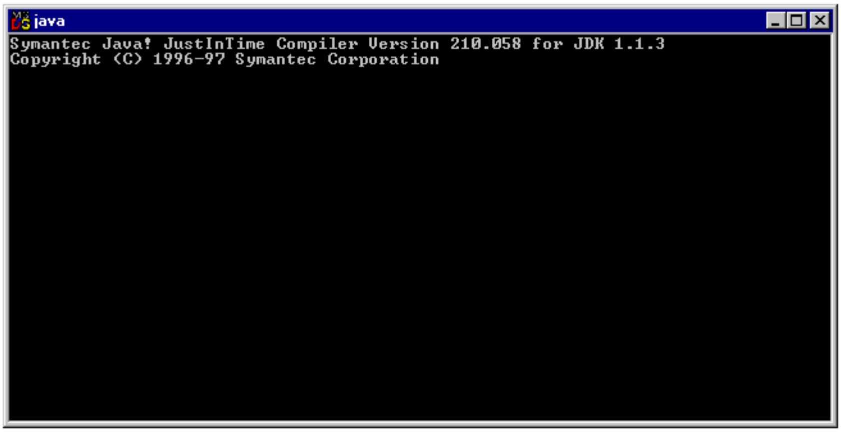
   Visual Cafe compiles your application before displaying it. Any warning or error messages appear in the Messages window.



```
sj -make -cdb MyIslePr.cdb -g -classpath C:\VISUALCAFE\BIN\Components\symbeans
Build Successful
```

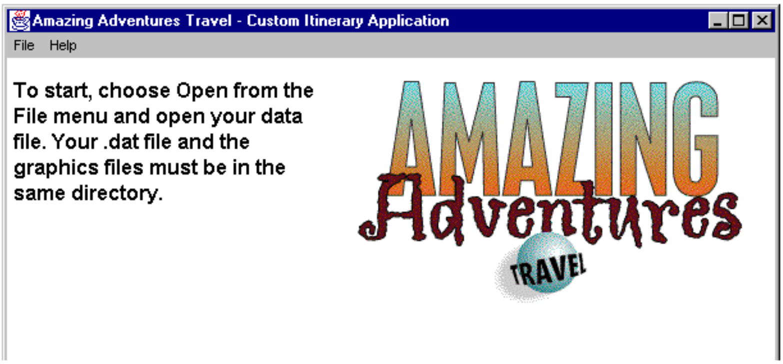
You can double-click an error message in this window to go to the line in the code where the error was generated.

Println statements and certain errors are displayed in the java window.



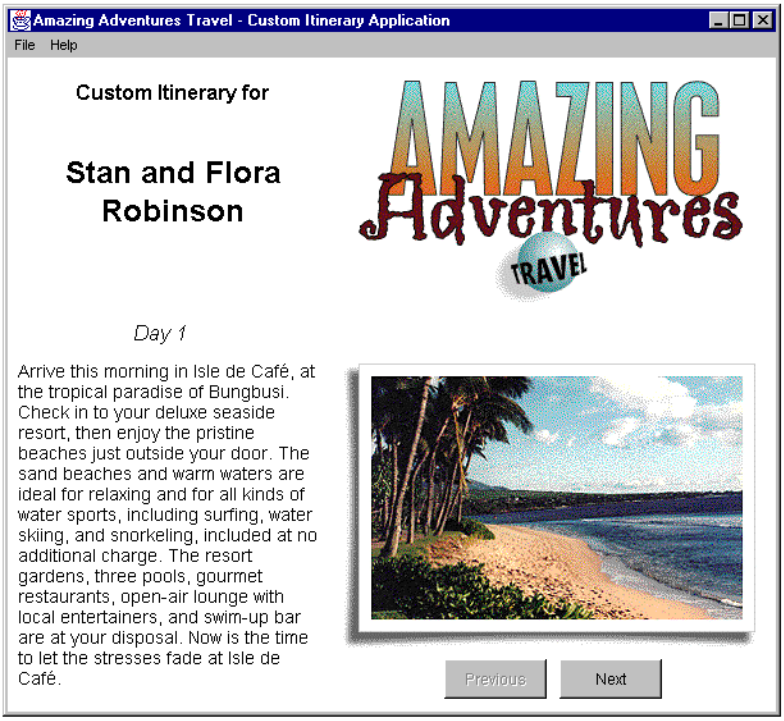
```
Symantec Java! JustInTime Compiler Version 210.058 for JDK 1.1.3
Copyright (C) 1996-97 Symantec Corporation
```

Then the application appears.


Amazing Adventures Travel - Custom Itinerary Application
File Help
AMAZING Adventures TRAVEL

To start, choose Open from the File menu and open your data file. Your .dat file and the graphics files must be in the same directory.

3 Open the data file in the Images directory, and click the Next and Previous buttons to verify your application works as planned.


Amazing Adventures Travel - Custom Itinerary Application
File Help
AMAZING Adventures TRAVEL
Previous
Next

Custom Itinerary for

Stan and Flora Robinson

Day 1

Arrive this morning in Isle de Café, at the tropical paradise of Bungbusi. Check in to your deluxe seaside resort, then enjoy the pristine beaches just outside your door. The sand beaches and warm waters are ideal for relaxing and for all kinds of water sports, including surfing, water skiing, and snorkeling, included at no additional charge. The resort gardens, three pools, gourmet restaurants, open-air lounge with local entertainers, and swim-up bar are at your disposal. Now is the time to let the stresses fade at Isle de Café.

4 Close the application.

5 Make adjustments in the Form Designer as needed. Then execute the application again. Repeat until the application looks the way you want it.

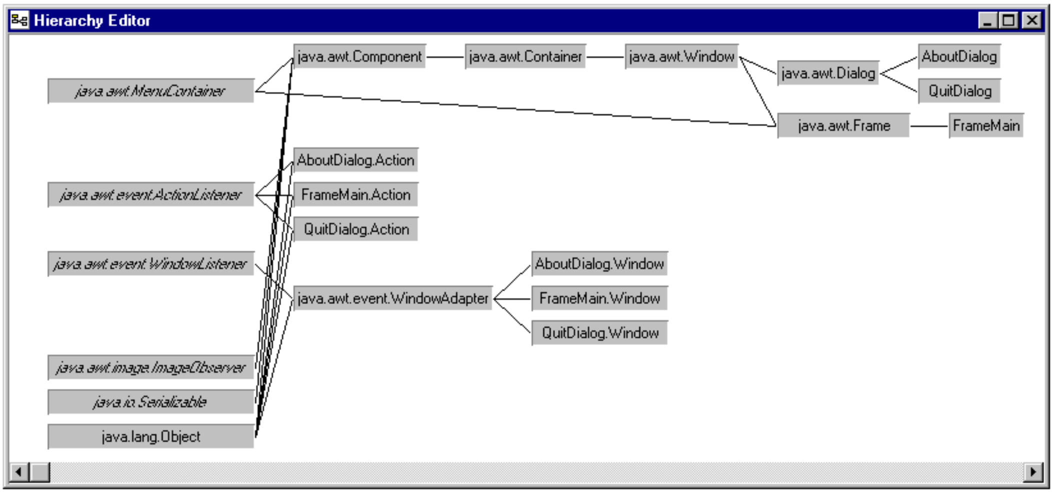
If the application works as planned, you have completed it.

Looking at the application in the Hierarchy Editor and the Class Browser

Visual Cafe has special object programming tools that make it easy to understand and manage Java classes. Let’s take a look at your application in the Hierarchy Editor and Class Browser:

1 While your application project is open, choose Hierarchy Editor from the Window menu.

Your application appears in the Hierarchy Editor.



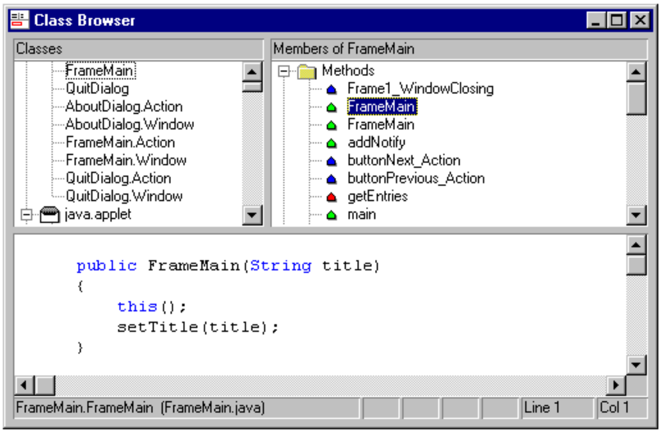
The Hierarchy Editor lets you visually learn, modify, and extend the Java class hierarchy for an applet or application. Drag and drop to instantly edit the hierarchy structure — both the visual model and the source code.

2 Right-click, then choose View Imports from the pop-up menu.

The Hierarchy Editor now shows all Java classes.

3 In the Hierarchy Editor, double-click a class to navigate to a class definition in the Class Browser. Then select a member in the upper right list.

The Class Browser shows the code associated with the class member.



```
    public FrameMain(String title)
    {
        this();
        setTitle(title);
    }
```

The three-pane Class Browser lists the classes in your project, and shows each method and data member contained within a class, all of which are individually editable to make you more productive. For example, when you click a class in the upper left pane, its methods and data members appear in the upper right pane. When you click a method, that implementation of the method is extracted from the source file and displayed in the lower pane where you can edit it. This makes it easy to navigate to the areas you want to edit — no matter how large the source file — and reduces the possibility of accidentally editing code outside the scope of the selected object.

Color syntax highlighting makes your Java source more readable. The Class Browser also has an incremental search feature when you have a class selected: as you type the name of a class, the list of matching objects is refined until you have automatically selected the class you want.

Visual Cafe reads your Java source code so the Hierarchy Editor and Class Browser always contain the most up-to-date information.

4 Close the Class Browser and Hierarchy Editor.

Debugging

The integrated graphical debugger provides source-level debugging and lets you browse data and manipulate calls and threads. You can modify a variable value and continue debugging without reloading. The Source window shows ValueTips so you can quickly view variable values.

While debugging, you can either step through each line of source code one at a time (including sublevel method code, when available) or step over the execution of method calls, which causes the debugger to execute all method code then pause on the line following the method call.

The debugger includes the following windows:

- Breakpoints window — The breakpoints for the entire project appear in this window. You can set simple breakpoints that stop execution at a certain line or method, or conditional breakpoints based on an expression.
- Threads window — You can pause individual threads, which causes their execution to cease temporarily while all other threads continue to execute, and resume them. This helps you check for and resolve thread synchronization errors, where more than one thread is in contention for the execution of a method. Double-clicking a thread in this window updates the Calls window, so it reflects the chain of execution for the thread, and the Variables window, so it reflects the scoping level of the thread.
- Calls window — All code modules and methods executed by a thread, and the current execution point and return points for each called method, appear in this window. This view is useful for following the flow of the code, for example.
- Watch window — In this window, you can specify variables and expressions that you want to watch continuously while debugging your program. The values update when you pause execution or step through code. You can also examine the contents of a class member or variable. To watch all the variables accessible to a method, drag the variable from the Variables window to the Watch window. To modify a variable value, you can enter it and continue debugging without having to stop and restart the debug session.

- Variables window — This window incorporates a tree view so you can collapse and expand the various scopes and to make it easy to examine the values of both simple and complex data types, such as arrays. It automatically displays the local and global variables and objects that are local to the current method, and the variables that are members of the current object. You can edit variable values.

All of the debugger windows have resizable columns so you can create the most readable display.

Visual Cafe also lets you debug an applet or application that's running on another computer. One computer displays the debug windows while an applet or application runs on a remote computer.

Working with the debugger is divided into the following sections:

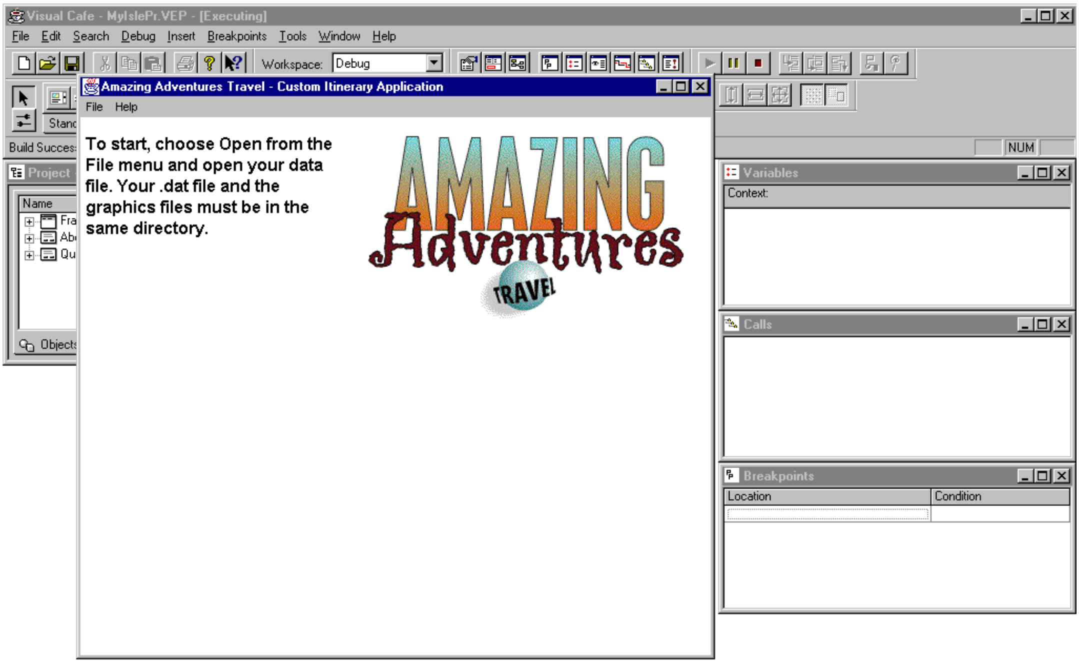
- Debugging the About dialog
- Debugging the Day label

Debugging the About dialog

Although the About dialog works properly in your application, here you can explore some debugging features by working with the About dialog code:

1. While your application project is open, choose Run in Debugger from the Project menu.

   Visual Cafe compiles your program and displays several debugger windows. Then it displays your application.


Visual Cafe - MyIslePr.VEP - [Executing]
File Edit Search Debug Insert Breakpoints Tools Window Help
Workspace: Debug
Amazing Adventures Travel - Custom Itinerary Application
File Help
AMAZING Adventures TRAVEL
Build Succes
Project
Name
Fra
Abo
Qu
Objects
Stanc
NUM
Variables
Context:
Calls
Breakpoints
Location
Condition

To start, choose Open from the File menu and open your data file. Your .dat file and the graphics files must be in the same directory.

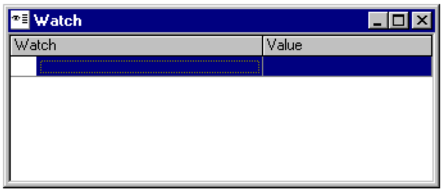
When your application is running, “Executing” appears in the titlebar of the Visual Cafe main window. Realize that when your application is paused, it cannot redraw its window, so the application may pick up the contents of other windows when you move them.

The Variables, Calls, and Breakpoints windows should be displayed. If they aren’t, open the window by choosing it in the Window menu.

2 If it’s not already open, open FrameMain in the Source window.

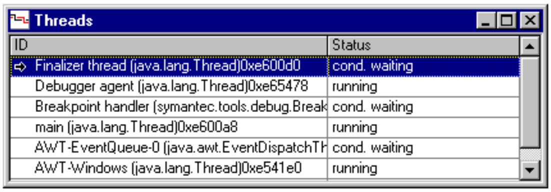
3 Choose Watch from the Window menu.

The Watch window appears.



4 Choose Threads from the Window menu.

The Threads window appears.



ID	Status
⇨ Finalizer thread (java.lang.Thread)0xe600d0	cond. waiting
Debugger agent (java.lang.Thread)0xe65478	running
Breakpoint handler (symantec.tools.debug.Break	cond. waiting
main (java.lang.Thread)0xe600a8	running
AWT-EventQueue-0 (java.awt.EventDispatchTh	cond. waiting
AWT-Windows (java.lang.Thread)0xe541e0	running

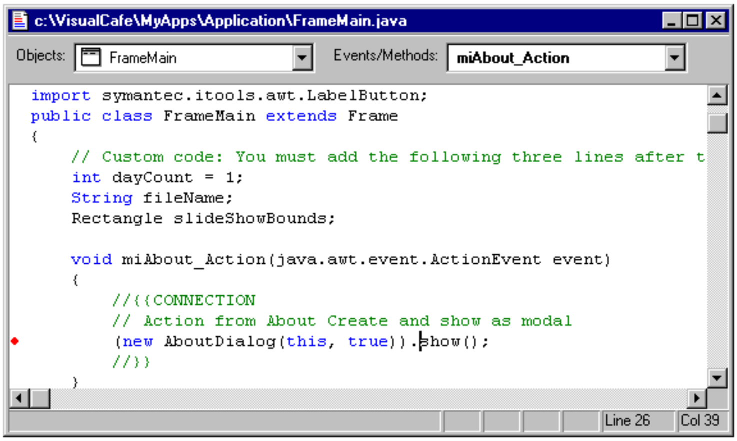
The window shows the standard Java AWT threads and other threads, including the debugger threads.

5 In the Source window, choose FrameMain in the Objects drop-down menu, then choose miAbout_Action from the Events/Methods drop-down menu.

The miAbout_Action method appears in the window.

6 Click the line with the show method, right-click, then choose Set Breakpoint from the pop-up menu.

A red diamond indicates a breakpoint was set.



Objects: FrameMain

Events/Methods: miAbout_Action

```
import symantec.itools.awt.LabelButton;
public class FrameMain extends Frame
{
    // Custom code: You must add the following three lines after t
    int dayCount = 1;
    String fileName;
    Rectangle slideShowBounds;

    void miAbout_Action(java.awt.event.ActionEvent event)
    {
        //{{CONNECTION
        // Action from About Create and show as modal
◆       (new AboutDialog(this, true)).show();
        //}}
    }
```

Line 26 Col 39

Notice the breakpoint in the Breakpoints window.

7 In your application, choose About from the Help menu.

The breakpoint is triggered. The Source window appears and the breakpoint line is highlighted. Variables and their values at this point in the code appear in the Variables window; the current call stack appears in the Calls window. The titlebar of the Visual Cafe main window displays [Paused].

8 Choose Step Into from the Debug menu.

Another Source window appears. It displays AboutDialog.java.

9 Choose Step Over from the Debug menu.

The next call is highlighted.

10 Choose Step Out from the Debug menu.

You return to the FrameMain.java file.

11 Right-click the highlighted line and choose Clear Breakpoint from the pop-up menu.

The breakpoint is removed.

12 Choose Continue from the Debug menu.

13 Now you can click OK in the About dialog.

14 Choose Open from the File menu and open Tour.dat.

The data appears in the application.

15 Click the FrameMain Source window.

16 Choose Find from the Search menu, and search for **class Action**.

The class appears. ActionListener receives events such as mouse clicks and determines what event handler each event goes to.

17 Set a breakpoint at **Object object**.

The breakpoint appears.

18 Click a button in the application window.

The breakpoint is triggered.

The Visual Cafe debugger lets you dynamically set breakpoints while the program runs for more powerful debugging. A listener or adapter is a useful place to put a breakpoint, because in the Breakpoints window you can disable the breakpoint by deselecting the checkbox and enable it with a condition for particular types of events.

19 Clear the breakpoint, then choose Continue from the Debug menu.

The application appears.

20 Close the Source window displaying AboutDialog.java.

Debugging the Day label

Continue exploring the debugger with these steps that examine the Day label code:

1. If not already open, open the Source window for your FrameMain application.
2. While the Objects field shows FrameMain, choose buttonNext_Action from the Events/Methods menu.

   Under buttonNext_Action, you'll see this method: slideShowItinerary.nextImage. The method is activated when the Next button is clicked.
3. Set a breakpoint at nextImage.

   A breakpoint symbol appears next to the code.
4. In your application window, click the Next button.

   The breakpoint is activated. The Source window appears with the breakpoint highlighted.
5. Look at the variables in the Variables window.

   The Variables window shows the variables and their values at this breakpoint.
6. Click the + next to **this**.

   The variables associated with the object appear.
7. Type **this.day**.

   The dayCount variable row is highlighted. This is the variable that sets the Day value.
8. Drag dayCount from the Variables window to the Watch window. (Or right-click and choose Evaluate Expression; in the Evaluate Expression dialog box, click Add Watch.)

   dayCount appears in the Watch window. You can drag the variables you want to watch closely, no matter where they are in the Variables window hierarchy, to this window.
9. Drag fileName (under dayCount) to the Watch window.

   fileName appears in the Watch window.
10. Scroll down in the Variables window, and click the + next to wrappingLabelDesc.

The variables associated with this component appear. You can resize columns for a more readable display.

11 Drag one of the wrappingLabelDesc variables, text, to the Watch window.

The variable appears in the Watch window.

12 Choose Continue in the Debug menu.

13 Click the Next button in the application.

The breakpoint is triggered. The Day and text values change in the Watch window.

14 In the Calls window, double-click actionPerformed.

The portion of code called in the action listener appears in the Source window.

15 In the Calls window, double-click buttonNext_Action.

You return to the original breakpoint position.

16 Deselect the checkbox in the Breakpoints window to deactivate the breakpoint.

17 In the Watch window, click the dayCount row, then click the Value field and change the value to 2 less than its current value.

18 In the Source window, highlight the dayCount variable under the breakpoint, and leave the cursor there.

ValueTips show that the value has changed.

19 Choose Continue from the Debug menu.

20 Press the Previous and Next buttons.

Notice what has happened to the Day value in your application.

21 In the Breakpoints window, select the breakpoint checkbox, then click the Next button in the application.

The breakpoint is triggered.

22 In the Breakpoints window, enter the following in the Condition field:

dayCount == 0

Now the breakpoint triggers only when dayCount is zero.

23 Choose Continue from the Debug menu.

24 Press the Previous and Next buttons until the breakpoint is triggered.

25 Clear the breakpoint, then choose Stop from the Debug menu.

The debugger windows close.

You have now completed the Visual Cafe tour and learned about many important Visual Cafe features. Now you're ready to build your own applets and applications. For more in-depth information, explore the online documentation and help provided with Visual Cafe.

6

Support and More Information

This chapter describes where to get more information on Java and Visual Cafe. It has the following sections:

- Where to get information
- Product updates
- Symantec service and support solutions
- Registering your Symantec product
- Technical support
- Electronic support
- Customer service
- Service and support headquarters
- Worldwide service and support

The information in this chapter, including URLs, is subject to change.

Where to get information

Visual Cafe includes the following documentation:

Title	To find it …
Visual Cafe help	Choose **Help Topics** from the **Help** menu. Choose a menu command, then press F1 to see a description. Press F1 in any dialog box or window to display help for that option. Press Shift+F1, then choose the item for which you need help.

Title	To find it …
Visual Cafe Samples	Go to the Samples directory. You need to use the custom install to install these samples.
Java API Specifications	Choose **Java API Reference** from the **Help** menu. Highlight a keyword in a Source window and press F1 for a description.
Java Language Specifications	Choose **Java Language Reference** from the **Help** menu. Highlight a keyword in a Source window and press F1 for a description.

To find the most up-to-date information about Sun Java and Visual Cafe:

For information about…	See…
Sun Microsystem's Java	http://www.javasoft.com
Symantec Internet Tools	http://cafe.symantec.com
Symantec Technical Support	http://service.symantec.com See the section "Symantec service and support solutions" in this chapter. For technical support telephone numbers, see the section "Service and support headquarters" in this chapter.

You can also purchase books on Java. Some books for Java and HTML are listed at http://service.symantec.com/amazon/index.html.

Product updates

Check the Symantec Web site (cafe.symantec.com) for updates to Visual Cafe. After you have registered your copy of Visual Cafe, you can use your login name and password to download patches and minor updates to your current version. If you bought your product in a box, the user ID and password are included with your package. If you bought your product online, the user ID and password were displayed to you on the screen.

Symantec service and support solutions

Symantec is committed to excellent service worldwide. Our goal is to provide you with professional assistance in the use of our software, wherever you are located.

Technical Support and Customer Service solutions vary by country. If you are outside the United States or Canada, please refer to the Worldwide Service and Support section at the end of this chapter.

Registering your Symantec product

If you have ordered the product online, you are automatically registered through Customer Service. In addition, you can use the toll-free fax number listed below to register your product, or register on the Customer Service Home Page under Service and Support at www.symantec.com.

If your address changes, you can mail or fax your new address to Customer Service. Please send it to the attention of the Registration Department.

Symantec Corporation
Attn: Registration Dept.
175 W. Broadway
Eugene, OR 97401

(800) 800-1438 Fax

You can also change your address on the Customer Service Home Page under Service and Support at www.symantec.com.

Technical support

Symantec's Technical Support department offers expanded support options designed for your individual needs and to help you get the most out of your software investment.

The phone numbers listed in the “Service and support headquarters” section are for support in North America. If you are outside the United States or Canada, please call the local Symantec office or distributor in your area, or refer to the information provided at the end of this chapter.

Symantec now offers different types of technical support services for you to choose from, which are described below. You are given StandardCare Support by purchasing the product and then can choose from Symantec’s PriorityCare and PremiumCare services to extend your level of support.

For the most current information on Symantec Support Solutions, please call our automated fax retrieval service, located in the United States, at (800) 554-4403 or (541) 984-2490, and request document 070. Alternatively, visit Symantec on the World Wide Web at www.symantec.com.

StandardCare support

All registered users of Symantec products are entitled to these services at no charge:

- Unlimited calls for 90 days (from the date of the first call) for installation assistance, configuration, and general usage questions. The phone number is (541) 465-8470.
- Unlimited technical assistance through CompuServe and America Online. These forums offer electronic access to our technical support staff, libraries of sample files, technical notes, and bulletins. You will also find a rich interaction and information exchange with other users of Symantec software.
- Unlimited use of the Symantec Discussion Groups available from the Service and Support section on www.symantec.com. These forums offer electronic access to our professional technical support staff, the Symantec knowledge base, FAQs, and announcements. The news server is at service.symantec.com.
- Unlimited access to company information through the Internet. With a Web browser such as Netscape, you get the latest company news by entering the URL for Symantec’s home page: www.symantec.com.

  The Web site include links to our anonymous FTP site, where you can download the latest software patches and drivers for Symantec products. You can also FTP directly to the site by entering: ftp.symantec.com.

- Unlimited use of Symantec's automated fax retrieval system for instant printouts of technical notes, bulletins, product literature, and general information by fax.
- StandardCare Support is available Monday through Friday, 7:00 a.m. to 4:00 p.m. Pacific Time.

For your first 90 days of free technical support, please refer to the StandardCare Support (541) phone number in the "Service and support headquarters" section.

PriorityCare support

All registered users of Symantec products are entitled to these services on a "pay-as-you-go" basis:

- The PriorityCare 800-number is charged to your VISA, MasterCard, or American Express on a per incident basis.
- The PriorityCare 900-number is charged to your telephone bill on a per minute basis. (As of this writing, the charge is $2 per minute, and an equivalent 900-number service is not available outside the United States.)
- Average hold time will be kept to a minimum.
- PriorityCare Support is available Monday through Friday, 6:00 a.m. to 5:00 p.m. Pacific Time.

To use the PriorityCare 800- and 900-number services, please refer to those numbers in the "Service and support headquarters" section.

PremiumCare support

All registered users of Symantec products are entitled to these services on an annual subscription basis:

PremiumCare Gold support

- Support for one year for a set number of incidents, with an option to add more incidents.
- Priority support on a a toll-free 800 line.
- Average hold time will be kept to a minimum.

- Electronic mail support is provided with a 24-hour response time on working days. The PremiumCare Gold Support is available for Visual Cafe for Windows owners at support_javawin@symantec.com and for Visual Cafe for Macintosh owners at support@devtools.symantec.com. Please include your subscription ID with any e-mail.
- PremiumCare Gold Support is available Monday through Friday, 6:00 a.m. to 5:00 p.m. Pacific Time.

PremiumCare Platinum support

- Unlimited calls on a toll-free 800 line.
- Average hold time will be kept to a minimum.
- A Support Center Manual with troubleshooting, installation, configuration, and usage information.
- Quarterly updates of technical notes and bulletins.
- Instant access to senior support staff.
- Automatic updates of inline software revisions. (Inline software revisions do not include version upgrades.)
- After hours and weekend support is also available to PremiumCare Platinum customers for an additional fee.
- PremiumCare Platinum Support is charged on an annual subscription basis per product family. The annual fee is for two subscribers; other subscribers can be added on a per person basis.
- PremiumCare Platinum Support is available Monday through Friday, 6:00 a.m. to 5:00 p.m. Pacific Time.
- Electronic support for Visual Cafe for Windows owners at support_javawin@symantec.com and for Visual Cafe for Macintosh owners at support@devtools.symantec.com.

To order PremiumCare Gold or Platinum support, please contact Customer Service or your Symantec sales representative.

Electronic support

Technical information is available electronically 24 hours a day. Symantec maintains the Symantec forums on CompuServe and America Online.

CompuServe

You can exchange information and ideas with Symantec representatives and with other users of Symantec products on the CompuServe bulletin board.

To access the Symantec forums on CompuServe, type:

`GO SYMANTEC` at any ! prompt or
`GO SYMDEVTOOL`

For additional information, or to subscribe in the United States and Canada, please call CompuServe at (800) 848-8199. Outside the United States and Canada, please call (1) (614) 718-2800. Check with CompuServe for data communications settings.

America Online

To access the Symantec bulletin board on America Online, type keyword:

SYMANTEC

For additional information, or to subscribe in the United States and Canada, please call America Online at (800) 227-6364. Check with America Online for data communications settings.

Symantec discussion groups

Use your Web browser to go to http://www.symantec.com and go to the Service and Support section. Here you can review the Frequently Asked Questions, search the Symantec Knowledge Base for known solutions to problems previously encountered, or post your own query to a support newsgroup.

Symantec representatives are ready to answer your questions via our support discussion groups. All messages posted will receive a response from a Symantec representative within 48 hours. The discussion groups are similar to electronic bulletin boards where you post a message and then return later to find an answer. These support forums are in Usenet newsgroup (Internet news) format and require that you have a newsreader available.

You can FTP directly to this site to download technical notes and software patches at: ftp.symantec.com. You can also use the news server, service.symantec.com.

Automated fax retrieval system

Symantec's automated fax retrieval system can be used 24 hours a day to receive product information on your fax machine. You can call from any touch tone phone to receive an index listing of both Technical Support and Customer Service documents available, then have any of these specific documents faxed to you.

To receive technical application notes and samples of "how tos," please call our Technical Support fax retrieval number, and choose Option 2.

You can receive general product information, data sheets, and product upgrade order forms from our Customer Service fax retrieval number.

- Technical Support: (541) 984-2490
- Customer Service: (800) 554-4403

In addition, you can receive a listing of Symantec offices and worldwide service and support partners by calling the Technical Support fax retrieval number, choosing Option 2, and requesting Document 1400.

Customer service

Symantec's Customer Service department builds and maintains long-lasting customer relations through consistent, expert service. Our Customer Service department is available to help you:

- Order an upgrade.
- Subscribe to the technical support solution of your choice.
- Fulfill your request for product literature or demonstration disks.
- Find out about dealers and consultants in your area.
- Replace missing or defective pieces (disks, manuals, and so on) from your package.
- Get status on an order or a return.

- Update your product registration with address or name changes.
- Replace lost passwords or user IDs.

You can reach Customer Service at (800) 441-7234 or under the Service and Support section at www.symantec.com. For specific questions about how to use your Symantec software, please contact Technical Support.

Replacing a CD-ROM

If the Visual Cafe CD-ROM is damaged or unusable, you can obtain a replacement.

To receive a replacement or refund for a product purchased through a reseller, please visit or contact the authorized dealer from whom you purchased the product.

If you ordered the product or upgrade directly from Symantec and wish to receive an replacement or refund in accordance with our 60-day money back guarantee, please return your CD-ROM and a letter including a brief explanation for the return. Please send by traceable means, such as UPS or FEDEX, to:

Symantec Corporation
ATTN: RMA Dept.
175 West Broadway
Eugene, OR 97401

Recovering user IDs and passwords

If your user ID and password become inoperative, send a message to Customer Service at custserv@symantec.com with your name, address, daytime phone number, ID, and password and we will resolve the situation for you within 24 to 48 hours. Type **Cafe 1.5 DL Reset** in the subject field of the message.

If you have lost your Visual Cafe ID, send a message to Customer Service at custserv@symantec.com. If your information does not appear in our registration database, we will provide additional avenues so you can verify ownership of the product.

Old version support

When a new version of this software is released, registered users may receive upgrade information through electronic mail or the postal service. After the release of the new version, telephone support will be provided for the previous version for 6 months. Technical information for the older versions may still be available on the electronic services and automated fax retrieval system.

Discontinued product support

When Symantec announces that a product will no longer be marketed or sold, telephone support will be discontinued 60 days later. Support will only be available for discontinued products through services such as our automated fax retrieval system or through documentation posted on electronic services such as the World Wide Web, Symantec BBS, CompuServe or America Online.

Service and support headquarters

Symantec's service and support headquarters for North America is at the following location.

Symantec Corporation 175 W. Broadway Eugene, OR 97401	(800) 441-7234 (USA & Canada Customer Service) (541) 334-6054 (all other locations) Fax (541) 984-8020
	For Visual Cafe technical support, please call the following numbers: StandardCare Support (541) 465-8470 PriorityCare 800 Support (800) 927-4014 PriorityCare 900 Support (900) 646-0004 or post a message in the Visual Cafe Discussion Group at http://www.symantec.com/techsupp/index.html.

Worldwide service and support

Symantec provides technical support and customer service worldwide. Services vary by country and include International Partners (IPs) who represent Symantec in regions where there is no Symantec office. Most IPs provide customer service and technical support for Symantec products in your local language, as close to your home or office as possible.

If your country is not listed in the International Locations section below, please call our Technical Support automated fax retrieval service, located in the United States, at (541) 984-2490, choose Option 2, and request Document 1400.

International locations

European headquarters

Symantec Europe Ltd.	Tel. (31) (71) 535 3111
Kanaalpark 145	Fax (31) (71) 535 3150
2321 JV Leiden	
The Netherlands	

Customer Service	Tel. (31) (71) 535 3294

Technical Support -	Dutch	Tel. (31) (71) 579 4407
	French PC/Mac	Tel. (33) (1) 41 38 69 80
	French Mac	Tel. (33) (1) 41 38 69 81
	German	Tel. (49) (211) 9917 110
	English	Tel. (44) (1628) 788 580
	Other Countries	Tel. (31) (71) 579 4425
		Fax (31) (71) 535 3153

International BBS (up to 14.4 baud)	Tel. (31) (71) 535 3169
Automated fax retrieval (24 hrs.)	Tel. (31) (71) 535 3255

Asia/Pacific Rim region

Symantec Australia Pty. Ltd.	Tel. (61) (2) 9850 1000
408 Victoria Road	Fax (61) (2) 9850 1001
Gladesville, NSW 2111	
Australia	
Technical Support	Tel. (61) (2) 9879 6577
	Fax (61) (2) 9879 6594
BBS	Tel. (61) (2) 9879 6322
Automated fax retrieval (24 hrs.)	Tel. (61) (2) 9817 4550

Mexico

Symantec Mexico	Tel. (52) (5) 545 1234
Rubén Dario No. 36, Piso 2, OFNA 6	Fax (52) (5) 531 2252
Colonia Chapultapec Polanco	
11560 México, D.F.	
Technical Support	For Technical Support call the automated fax retrieval service, located in the United States, at (541) 984-2490, and request Document 1400.

Japan

Symantec Japan, Inc.
Pine Bldg. 6F
3-1-2, Shibuya, Shibuya-ku
Tokyo 150 JAPAN
Phone: (81)03-3498-1118
Fax: (81)03-3498-1124

Every effort has been made to ensure the accuracy of this information. However, the information contained herein is subject to change without notice. Symantec Corporation reserves the right for such change without prior notice.

Glossary

adapter

Classes that implement an interface. The java.awt.event package provides, as a convenience, a series of listener classes that can be implemented by classes

applet

A program that can be added to a Web page and run by Java-enabled Web browsers (or the Applet Viewer). Java applets are secure because they can't access the file system on the local computer, but they can still read from the Web server. This means that applets are less likely to damage a user's system or send private information back to the Web server. (While an applet usually doesn't have a menu bar, it can create a Java frame that has a menu bar.)

applet tag

HTML code that causes an applet to appear in a Web page. It has the following basic format:

<APPLET code="*applet*.class" width=*pixw* height=*pixh*></APPLET>

applet is the name of the applet.

pixw is the number of pixels for the width.

pixh is the number of pixels for the height.

Consult an HTML book for more information on the applet tag.

application

A Java application is standalone and cross-platform: it can run on all computers with a standalone Java virtual machine and supporting files (which, for instance, you can obtain from the Sun JDK at java.sun.com), so separate applications for each environment aren't needed. Java applications can have menu bars and access files on the local computer. A Java application is like a C++ application, but it's cross-platform and you need a Java virtual machine to run it. For example, you could write a word processor program in Java. A Java frame is a good base class for a main application window: it can exist by itself, contain components and menus, and can show dialogs and windows.

bean

A component complying with the JavaBeans standard. Also called a JavaBean or a JavaBeans component. A bean is a reusable component that can be visually manipulated in a builder tool. The minimum requirements are that it can be instantiated (it is not an abstract class or interface) and has a class constructor method that takes zero parameters (it has a null constructor). It can also have properties and events, implement the serializable or externalizable interface, and follow method signature rules so it can be introspected.

bytecodes

Instructions that are similar to machine code, but are machine-independent. During execution, the Java virtual machine either interprets the bytecodes with a Java interpreter or converts them to machine code (stored temporarily in memory) with a Just-In-Time compiler.

class

A collection of attributes and methods that defines the implementation of a particular kind of object. A class definition defines instance variables and methods, and specifies the interfaces the class implements and the immediate superclass of the class. See also *inner class*.

component

A JavaBeans component.

- A *visual component* is a user interface element, such as a window, menu, button, and so on, that is visible at run time and appears in the Visual Cafe Project window and Form Designer. It extends from the Java Component class and has a screen position, a size, and a foreground and background color.
- An *invisible component* is not visible at run time, such as a Timer, or displays in a different way in the Form Designer and at run time, such as a MenuBar. It does not extend from the Java Component class. In the Form Designer, an invisible component is represented by an icon that does not effect the form layout.
- Some components can contain other components, such as an application window containing a button; these components are called *containers*. In the Objects view of the Project window, the components in a container appear subordinate to the container, like a file system display. The containers at the top level are separate Java files in your project (called Visual Cafe *forms*), while the components in the containers are Java code within the container Java file.
- To create your user interface, you can display a form in the Form Designer, then drag onto the form a variety of components from the Component Library or Palette; you assign the component properties in a separate Property List window, and add interactions between components with the Interaction Wizard. Visual Cafe automatically creates the Java code for you during this design process.
- Components are like controls in C++.

container

A component that can contain other components. There are two general categories of Java containers: windows and panels. Some types of windows are a frame (which can have a menu bar) and a dialog. A panel is a defined area that can appear in a window, such as an applet that appears in a browser. A top-level container, also called a Visual Cafe *form*, is at the top level in the Objects view of the Project window. It has a corresponding Java file that appears in the Packages and Files views.

dialog

A Java Dialog component is a simple window with a title bar that can receive and process input from a user. Dialogs cannot be used by applets, and are not suitable for use as a main window for an application. Dialogs can contain components, but not menus. Use dialogs for temporary windows. See also *component* and *container*.

event

An action, initiated by a user or a program, to which an object can respond. Events are typically user actions that the program can capture and respond to. Examples of events include mouse clicks, key presses, and mouse movements.

event handler

A method that is called when a certain type of event is triggered. Visual Cafe automatically generates the code needed to bind the occurrence of an event to an event handler when you create an interaction with the Interaction Wizard or from the Events/Methods pull-down menu of the Source window. An event binding is made of three parts: the event handler, the listener or adapter implementation, and the code to register the listener or adapter to the object triggering the event. The default name of an event handler is the object name, followed by an underscore then the name of the action that triggers the event.

form

A component that can only appear at the top level in the Objects view of the Project window. Applet, Frame, Window, and some dialog components are forms. In the Component Library, the Visual Cafe forms are all in the Forms group. When you open a form in the Form Designer, the form boundaries are the bounds of the window. You can position other components, such as text, buttons, and graphics, on the form; these components are contained by the form.

frame

A Java Frame component is a general-purpose application window to which you can add components and menus. See also *component* and *container*.

inner class

A class that is included within the body of another class, even within a method (called a local class). An inner class is also called a nested class. This feature is new for JDK 1.1 and is useful for creating adapter classes. After compilation, the inner class ends up in its own class file, which has a dollar sign ($) in its name.

interaction

A relationship that you define between components, or a component and itself, resulting in Visual Cafe–generated code. You can create an interaction between components on one form, or between a component on one form and a top-level component (such as having a button click open another form). An interaction consists of one or more components, a trigger event, and an action. For example, you can connect a button (the trigger component) to a text box (the action component) so that when the user clicks on the button (the trigger event), the associated text box is enabled for user input (the action).

interface

A set of methods and constants to be implemented by another object. It defines the behavior, or certain characteristics, that another object implements. An interface can define abstract methods and final fields, but not the implementation of them.

introspection

The ability to read JavaBeans classes directly with the Core Reflection API using the Introspector class. This information is stored in a BeanInfo object and includes data such as properties, events, and all the accessible methods.

invisible component

See *component*.

Java Archive (JAR) file

Compressed archive file that complies with the JavaBeans standard. It is the primary method for delivering JavaBeans components. For example, a JAR file can contain one or more related beans, and any support files, including classes, icons, graphics, sounds, HTML documentation, serialization files, and internationalization files. You can deploy applets and applications from a JAR file. A JAR tool, called jar.exe on Windows computers, archives and extracts JAR files and is provided with JDK 1.1. In Visual Cafe, to use the JavaBeans components in a JAR file, you must first add the file to the Component Library.

Java Virtual Machine

The Java Virtual Machine (Java VM) contains a bytecode translator that converts the downloaded binary Java files into instructions to be executed by the client machine. The Java VM also contains the library routines that are called by a Java applet.

JavaBeans

A standard for creating portable, cross-platform components.

JavaBeans component

A component complying with the JavaBeans standard. Also called a bean or JavaBean. A bean is a reusable component that can be visually manipulated in a builder tool. The minimum requirements are that it can be instantiated (it is not an abstract class or interface) and has a class constructor method that takes zero parameters (it has a null constructor). It can also have properties and events, implement the serializable or externalizable interface, and follow method signature rules so it can be introspected.

layout manager

The layout is a property of container components that automatically arranges components within the container so that they display well on different platforms and screens, and at different resolutions.

listener

An event listener is an object that has defined the listener interface for a specific event. After this interface has been implemented in a class, an instance of this class may be registered as an event listener. When an event is generated, the event is sent to the object, as well as all other registered listeners.

method

A procedure that is defined as part of an object and acts on an object that is based on that object. It is a collection of statements that perform a complex operation. When a method is called, the instructions in the method are performed and the result is returned to the statement that made the call.

nested class

See *inner class*.

package

A Java package is a group of related Java classes and interfaces. It is both a way of organizing code and a library of code. You can view the source files and packages of a project in the Package view of the Project window.

panel

A Java Panel component is a general-purpose container that can be added to another container. A panel has no visible border and can be used to break up windows into logical regions. See also *component* and *container*.

project

For an application, a Visual Cafe project is a collection of components and code that make up the application. For applets, a Visual Cafe project is a collection of components and code that make up an applet or a group of related applets; it optionally contains one or more HTML files that the applets run in.

property

A named attribute of a component. Properties define component characteristics such as the size, color, label, and state — for example, enabled or disabled.

top-level component

A component that can only appear at the top level in the Objects view of the Project window. Also called a *form*. Applet, Frame, Window, and some dialog components are top-level components. In the Component Library, the Visual Cafe top-level components are all in the Forms group. When you open a top-level component in the Form Designer, the component boundaries are the bounds of the window. You can position other components, such as text, buttons, and graphics, on the top-level component in the Form Designer; these components are contained by the top-level component.

visual component

See *component*.

window

A Java Window component, which is based on the Window class, is an area that has no borders and no menu bar. See also *component* and *container*.