After finishing this chapter, you should be able to:

- Provide a brief definition of the following terms: algorithm, class, computer program, event, function, Integrated Development Environment (IDE), instance, instantiation, method, method parameter, object, object-oriented programming (OOP), programming language, property, state of an object.
- Run the Alice software and locate and describe the following components of the Alice interface: World window, Object tree, Details area, Editor area, Events area, menu bar, trash can, clipboard, Play button, Undo button, Redo button.
- Load and play an existing Alice world.
- Create a new Alice world by adding objects to a blank world, positioning them, and using simple methods to animate those objects.
- Print the code for Alice methods and events.
OBJECT-ORIENTED PROGRAMMING AND ALICE

An algorithm is a step-by-step process. A computer program is a set of instructions telling a computer how to perform a specific task. As such, every computer program is an algorithm. Early computers were far less complex than computers are today — their memories were smaller and their programs were much simpler. In order to help manage the growing complexity of computers, computer scientists have introduced the notion of objects and object-oriented programming. An object is anything that is manipulated by a computer program. It is possible for modern computers to manipulate many objects at the same time.

An object can be something in the physical world or even just an abstract idea. An airplane, for example, is a physical object that can be manipulated by a computer. Almost all commercial aircraft today, Boeing 777s, Airbus 330s, and so on, have autopilots — computers with programs that can fly the plane. The autopilot is a computer that manipulates an object in the physical world. To the computer, the airplane is an object.

Some objects, in fact most objects that computers manipulate, are not physical objects. A bank transaction is an example of an object that is not physical. There is a set of activities that can be called a transaction, there may be physical money that changes hands, and there is usually a paper record of the transaction; however, the transaction itself is simply a concept, or an idea. It is an object, but not a physical object.

Whether an object exists in the physical world doesn’t matter much in terms of what happens inside a computer. To a computer, an object is simply something that can be represented by data in the computer’s memory and manipulated by computer programs. The data that represents the object is organized into a set of properties. Each property describes the object in some way. For example, the weight of an airplane, its location, the direction in which it’s facing, and so on are all properties of the airplane. A computer manipulates an object by changing some of its properties or some of the properties of its sub-parts. For instance, the autopilot might change the angle of a wing flap (a sub-part), which in turn affects the entire airplane.

Sometimes the hardware in a computer can translate these changes in properties into actions that affect the physical world—as an airplane’s autopilot does—and sometimes the changes in an object only affect information in the computer’s memory and have no direct affect on the physical world. For example, when a bank deposit is recorded on a computer, the amount of money in the bank balance property of the bank account object is changed, but there is no other immediate effect on the physical world.

The programs that manipulate the properties of an object are called the object’s methods. We can think of an object as a collection of properties and the methods that are used to manipulate those properties. The values stored in the properties of the object at any one time are called the state of the object. This modern approach to computer programming is known as object-oriented programming, or OOP for short.

A computer programming language is a particular set of instructions for programming a computer, along with the grammar and syntax for using those instructions. Most modern computer programming languages are object-oriented languages, in which programs are organized into a set of methods that manipulate the properties of objects stored in a computer. In order to understand any object-oriented system of programming, you need to know something about how that system handles objects, and about the language that is used in methods to manipulate objects in that system.
Learning to program a computer is often a difficult task because of the need to learn about programming concepts and the language of programming at the same time. It’s also difficult because people find it hard to visualize all of the changes that are occurring as a computer program runs. Alice can make it easier to learn to program a computer by helping with both of these problems.

Alice is an object-oriented system of programming. The objects in Alice exist in a three-dimensional virtual world, much like a modern video game. In fact, the virtual world itself is an object in Alice—it has properties, and methods that can be used to manipulate those properties. Alice is somewhat like other modern object-oriented programming systems that use languages, such as Java, C++, or Visual Basic, but, as you will see, it is constructed so that you don’t need to memorize the grammar and syntax of the language in order to write computer programs. As you are learning Alice, you can concentrate on learning about the ideas of computer programming, such as the logic of your algorithms, instead of having to worry about the spelling and grammar of a new language at the same time.

The virtual world of Alice is one that you can see. Like the real world, it has three-dimensional space (and time), and each object has properties just like physical objects, such as color, size, position, the direction in which it’s facing, and so on. Alice has a camera that allows you to see its virtual world on a computer screen, just as you might view a movie or a video game. This ability to see what happens to objects in your virtual world makes it easier to learn computer programming with Alice than with almost any other system of programming. For instance, if you try to program a white rabbit to run around in a circle, and instead he simply stays in one spot and spins around, you can see that happening on the screen. You can get instant feedback from viewing the way Alice runs the programs you have created. Not every programming system is so easy to use. Often it is necessary to go through a process known as compiling before you can run a computer program.

For more information on compilers, interpreters and computer languages, see Appendix C of this text.

In summary, there are three things about Alice that make it easier to learn programming by using Alice than almost any other system of programming:

- Minimal memorization of syntax — Alice is constructed in such a way that you do not need to learn the grammar and syntax of a strange new language and can instead focus your attention on the concepts of computer programming;
- Visualization — Alice allows you to see the effects of your programs and any changes you make to them;
- Rapid feedback — Alice provides rapid feedback, which you may get at any time by simply starting your virtual world and watching what happens.

You will also find that Alice is fun and interesting to use, which never hurts when one is trying to learn about something new.
TUTORIAL 1A — EXPLORING THE ALICE INTERFACE

In this tutorial, you will explore the Alice interface, and then load and play an Alice world. Before starting, you should have a computer system with the Alice software properly installed. Fortunately, installing Alice is easy. The software is available freely from The Stage Three Development Team at Carnegie Mellon University via their website at www.alice.org and is also on the CD that accompanies this book. See Appendix A for further instructions on acquiring, installing, and starting the Alice software.

Anyone attempting this exercise should have experience using a computer. You certainly don’t need to be an expert, but you should have some experience with things like word processing and accessing the Internet so that you are familiar with Windows, a mouse, a keyboard, and a printer.

TIP
A six-page laminated CourseCard for Alice version 2.0, which summarizes Alice features and commands, is available from Course Technology — ISBN 1-4188-4675-9. It will prove useful as you learn to use Alice, and later as a command reference.

THE OPENING SCREEN

1. Start the Alice software. You will see the Welcome to Alice! dialog box over the front of the Alice Integrated Development Environment (IDE), as shown in Figure 1-1. An IDE is a computer program that is used to write other computer programs. Most modern programming languages have IDEs with everything you need to create and run computer programs. Alice is no exception, but its IDE is simpler than most. The Alice IDE is often called the Alice interface.

FIGURE 1-1: THE ALICE INTERFACE WITH THE WELCOME TO ALICE! DIALOG BOX
2. Notice that the Welcome to Alice! dialog box has five tabs: Tutorial, Recent Worlds, Templates, Examples, and Open a world. You may get back to this dialog box at any time while using Alice by clicking File on the menu bar, and then clicking New World or Open World. Let's look at each of these tabs before continuing.

3. Click the Tutorial tab and you will see four Alice tutorials. You won't use the tutorials now, but you may want to come back to them later as an exercise on your own. When you are ready to use the tutorials, either click the tutorial you would like to run, or click the large Start the Tutorial button to follow them in order. They were created by the developers of Alice to help people learn the system. They are quite easy to follow.

4. Click the Recent Worlds tab. You will see thumbnail sketches of the most recently saved Alice worlds. If no worlds have been saved since the Alice software was installed on your system, this tab will be blank.

5. Click the Templates tab. Alice comes with six blank templates for starting a new virtual world—dirt, grass, sand, snow, space, and water. Each of the templates includes a texture for the surface, which is called the ground in Alice, and a background color for the sky.

6. Click the Examples tab. Several example worlds created by the Alice developers are provided with the Alice software. We'll come back to the examples tab later in this tutorial.

7. Click the Open a world tab. This tab is used to access other Alice worlds saved on your computer. In Figure 1-2, you can see this tab. It is very similar to the “Open File” dialog boxes seen in other programs, such as Microsoft Windows, with navigation icons across the top, a list of folders and Alice worlds from the current directory in the middle, and some controls to view and open files at the bottom. Notice that the Alice world files end with the extension .a2w. These files were created with version 2.0 of the Alice software, the most recent version. You may also notice that the interface looks slightly different than most other Windows interfaces. This is because Alice uses a generic interface that looks the same when using the Windows, Apple, or Unix operating systems.

8. Next, you will look at the Alice interface with an Alice world open. Click the Examples tab, click the lakeSkater thumbnail, and then click the Open button to open the lakeSkater Alice world. It will take a few seconds for Alice to load all of the elements of the world. You will see the names of the elements flash past in a small window in the center of the screen while this happens. When Alice is finished loading the world, your screen should resemble Figure 1-3.
FIGURE 1-2: THE OPEN A WORLD TAB IN THE WELCOME TO ALICE! DIALOG BOX SHOWING EIGHT FOLDERS AND THREE ALICE WORLD FILES

FIGURE 1-3: THE ALICE INTERFACE AFTER LOADING THE LAKESKATER EXAMPLE WORLD
THE MAIN WORK AREAS OF THE ALICE INTERFACE

The Alice interface has five main work areas, as shown in Figure 1-3: the World window, the Object tree, the Details area, the Editor area, and the Events area. There are also several elements across the top of the interface—a menu bar, three control buttons, a trash can, and a clipboard. Let’s look at each of these before playing with the lakeSkater world.

The World Window

The World window contains a view of the lakeSkater virtual world. There is a set of arrows below the window to control the Alice camera, which provides you with the view in the window. Next to the arrows is a large green ADD OBJECTS button. Click this button, and you will see a big change in the Alice world because you will have switched from the standard Alice interface to Scene Editor mode. Scene Editor mode is used to add and position objects in an Alice world before playing the world. In this mode, the Alice World window is larger and has additional controls. The Alice object galleries can be accessed at the bottom of the screen. You will work with the Scene Editor mode in a later tutorial. For now, click the large green DONE button to return to the standard Alice interface.

The Object Tree

The Object tree is to the left of the World window. It shows the objects in the current Alice world organized as a tree of tiles, with a tile for each object. The plus sign next to an object shows that it has sub-parts, which may be seen by clicking on the plus sign. Click the plus sign to see the sub-parts of the IceSkater, and then click the minus sign to hide its sub-parts. The top four tiles in the Object tree—world, camera, light, and ground—appear in every new Alice world.

The Details Area

The Details area of the Alice interface is located below the Object tree. It has tabs to show properties, methods and functions for the currently selected Alice object. Properties contain information about an object, such as its color and position in the world. Methods are programs that manipulate an object, such as the method to make an IceSkater turn. A function is a method that returns a value, such as the distance between two objects.

You may select an object by clicking that object in the World window or by clicking its tile in the Object tree. Information about the currently selected object will be displayed in the Details area.

Click each of the following elements, and you will see the listed results:

- The world tile in the object tree ⇒ world’s details in the Details area
- The lake tile ⇒ lake’s details in the Details area
- The iceSkater tile in the Object tree and then the properties tab in the Details area ⇒ iceSkater’s properties
- The methods tab ⇒ iceSkater’s methods
- The functions tab ⇒ iceSkater’s functions
The Editor Area

The largest area of the Alice interface is the Editor area, which is to the right of the Details area. Here, methods are assembled and edited by clicking and dragging tiles from other parts of the interface. The bottom of the Editor area has a row of logic and control tiles that can be used to put branching, looping, and other logical structures into the algorithms that will make up an object’s methods. Most of the time that you spend working with Alice will be spent using the Editor area.

The Events Area

The Events area in Alice is above the Editor area. This part of the interface shows existing events and is used to create new events. An event consists of a condition, called an event trigger, and the name of a method, called an event handler. Whenever the event trigger occurs, the event handler is called into action. For example, you might want the sound of a splash to occur if an ice skater falls through the ice. Falling through the ice would be the event trigger, and the method that makes the splash sound would be the event handler.

Some events, such as causing a method to run when a key is pressed, provide user interaction for an Alice world. The flight simulator world that can be accessed through the examples tab is an interactive world that you might want to take a look at after finishing this chapter. Events are covered in detail in Chapter 3.

OTHER ELEMENTS OF THE ALICE INTERFACE

In addition to the main work areas that you have just explored, the Alice interface also has two icons, three buttons, and a menu bar near the top of the screen. These are identified in Figure 1-3 and are discussed in the following sections.

Alice Tools for Deleting and Copying

The Trash can icon and the Clipboard icon near the top of the screen are used for editing Alice worlds.

You can delete an item in an Alice world, such as an objects or instruction tile, by dragging and dropping it in the Alice trash can. You can also right-click on an object or tile, and select delete from the menu that appears.

You can copy an item by dragging and dropping it on to the Clipboard icon in the top-right corner of the interface, and then dragging it from the Clipboard icon and dropping it in its new location. You can also duplicate a method tile by right-clicking it and selecting make copy from the menu that appears, but this does not work with Alice objects.

The Undo and Redo buttons near the top left corner of the interface are also useful for editing an Alice world. You can undo the last change you made by clicking on the Undo button. The effects of the Undo button can be reversed by using the Redo button. Alice can remember the last several dozen changes that you made. The Ctrl+Z and Ctrl+Y shortcut keys can also be used for Undo and Redo, although there are no shortcut keys for cut, copy, and paste.

Alice Menus

The Alice interface has a menu bar at the top of the screen with four menus: File, Edit, Tools, and Help. The menus are used far less in Alice than in many other computer programs. For now, you will look at only a few of the items on these menus. All of the features on the Alice menus are listed in Appendix B.
**Tutorial 1B — Playing an Alice World**

**File Menu** The Alice File menu has commands for opening, closing, and saving Alice worlds, as well as options to export an Alice world as a movie file or as an HTML Web page. You will use these options in later tutorials throughout this book.

**Edit Menu** Currently the only option on the Alice Edit menu is Preferences, which is used to change settings for the Alice software. Appendix B of this book lists and describes these settings. The most important thing to know for now is that the Alice Edit menu is not used to edit Alice methods in the same way that the Edit menu can be used to edit elements, such as documents, in Microsoft Word. Instead, Alice emphasizes the use of a drop-and-drag interface, which uses the editing icons and buttons described in the previous sections of this chapter.

**Tools Menu** The Alice Tools menu contains three options: World Statistics, Text Output, and Error Console. The Text Output option allows you to see system messages generated as you play Alice worlds, and the Error Console can be used to look at detailed Alice error messages. Both of these are rather sophisticated, and are not very useful for novice programmers. The World Statistics allows you to see statistics, such as the number of objects in a world, the time the world has been open, and many other useful items. Only some of the information here will be meaningful to you until you learn more about Alice and computer graphics.

**Help Menu** The Help menu does not contain an option to look up the features of Alice as you might expect. By not providing a way to look up features, the developers of Alice were hoping to encourage people to learn about Alice by experimenting with it.

The Help menu does have three options: Tutorials, Example Worlds, and About Alice. Example Worlds and Tutorials will both take you back to the Welcome to Alice! dialog box. The About Alice option will give you general information about the development of Alice and has a link to the Alice Web site www.alice.org, where you can find out more about Alice.

From time to time you may encounter errors while using the Alice software. The Alice error message box contains a button to submit a bug report to the Alice development team in the Human Computer Interaction Lab at Carnegie Mellon University. Reporting errors will help to improve future versions of Alice, and in some cases a member of the team may contact you directly. You may also submit bugs and suggestions about Alice through the Alice web site. They want to hear from the users of Alice.

**TIP**

In this tutorial you will experiment with playing an Alice world. Alice worlds fit into two different categories — some Alice worlds are interactive the way a video game is, while others are simply run and viewed like a video tape. In either case, experienced Alice users refer to “playing” an Alice world the way most software developers talk about “running” a computer program.

The Alice world that you will play in this tutorial is the lakeSkater world used in Tutorial 1A. It is not an interactive world, but is more like watching a video of an ice skater’s performance. If you have just finished Tutorial 1A and still have the lakeSkater world open, then continue with the steps below. If you do not have the lakeSkater world open, then before starting this tutorial you need to run the Alice software and open the lakeSkater example world. The lakeSkater example world can be found on the Examples tab of the Welcome to Alice! dialog box that appears when you start the Alice software.
PLAYING THE LAKESKATER WORLD

1. There are three buttons near the top of the Alice interface, labeled Play, Undo, and Redo. Undo and Redo are used for editing, as described in Tutorial 1A. The Play button is used to play the current Alice world. When this button is clicked, the world will play in a larger version of the World window, with player controls at the top of the window, as shown in Figure 1-4. Click the Play button now and watch the show unfold. Let the world play through to the end at least once before proceeding.

2. Notice that the new window has a speed slider control and five buttons across the top of the window in which the Alice world plays. The buttons are labeled Pause, Resume, Restart, Stop and Take Picture.

3. The Restart button is used to begin playing the current world again from the beginning. The Pause and Resume buttons work like the pause and play buttons on a VCR or DVD. Click the Restart button now to restart the lakeSkater world, and then experiment with the Pause and Resume buttons.

4. The speed slider is used to change the speed of the world while it is playing. Restart the world, and experiment with the speed slider control.

5. The Take Picture button captures an image of the currently playing world and saves it in a data file. Restart the world and click the Take Picture button to take a picture of the world. An Image captured and stored dialog box will appear, showing you the full path name of the file that was saved. The stored image file can be viewed and used as any other computer image file can be. Appendix C has more information on changing the settings for Alice’s screen capture function.
6. The Stop button stops the world that is currently playing and returns you to the standard Alice interface. Once the Stop button is pressed, you will need to click the standard interface’s Play button to replay the world. Try this now. After you have finished experimenting, click the Stop button one last time to return to the standard Alice interface.

**EXITING ALICE**

Alice is a graphics intensive program that can use a lot of a computer’s memory, so you should exit the Alice program when it is not in use. Now that you are finished with this tutorial, you should exit the Alice program.

To exit Alice:

1. Click File on the menu bar, and then click Exit.
2. If you have made any changes to the current world since it was last changed, a Save World? dialog box will appear, asking you if you want to save the world first. If this happens when you attempt to exit Alice after viewing the lakeSkater world, click No so that you do not change the saved example world.

**TIP**

While you are viewing or editing an Alice world, a dialog box will appear every 15 minutes, warning you that you have not saved your Alice world. If this happens while you are playing an Alice world, such as in Tutorial 1B, then it’s probably safe to ignore the warning. If it happens while you are creating or editing your own Alice world, then it’s probably a good idea to save your world.

**TUTORIAL 1C — CREATING YOUR FIRST ALICE WORLD**

In this tutorial, you will create, play, and save a new Alice world. You should have finished Tutorials 1A and 1B before starting. You will create an Alice world, in which a bunny will move from the right side of the screen to the center, turn to face the camera, and then say “Hello, World!” This is an Alice equivalent of the “Hello, World!” program that students traditionally write as their first program in a new programming language.

**TIP**

The tutorial begins with the Alice software closed. If you have an Alice world open, then exit Alice before continuing.

To start the new world:

1. Start the Alice software.
2. Click File, and then click New World. You should now see the Templates tab of the Welcome to Alice! dialog box, as shown in Figure 1-5.
3. Thumbnail sketches for six new world templates are now available—dirt, grass, sand, snow, space, and water. The templates appear to be very simple with a texture for the ground and a background color for the sky, but looks can be deceiving. There is actually a great deal of computer programming behind a new Alice world, with a camera, ambient light, and other elements already in place. Click the grass thumbnail, and then click the Open button.

A new Alice world based on the grass template is now open, and you can see the standard Alice interface that you used earlier in the chapter. Notice that the Object tree in the upper-left part of the interface contains the four tiles that appear in every Alice world: world, camera, light, and ground, as shown in Figure 1-6. You can see from the way the tree is organized that the other objects are sub-objects of the world.

The new world also has the default event—When the world starts, do world.my first method event—in the Events area, and a blank default method—world.my first method—in the Editor area.

**FIGURE 1-6: THE OBJECT TREE AFTER STARTING A NEW WORLD**

ADDING OBJECTS TO AN ALICE WORLD

The next several steps will introduce you to the Alice object galleries and the process of adding objects to an Alice world. Many people get carried away with creating big Alice worlds with many objects when they first start to use Alice. In this tutorial, you will start with a very small Alice world with a minimum number of objects. Objects are added to an Alice world in Scene Editor mode.
To enter Scene Editor mode:

1. Click the large green **ADD OBJECTS** button in the bottom-right corner of the World window to enter Alice’s Scene Editor mode, which is used to add objects to an Alice world and position them.

2. Note that the Alice interface now looks different, as shown in Figure 1-7. The Object tree and the Details area are still visible on the left, but there is a new area on the right side of the screen. This new area is called the Scene Editor.

The Scene Editor has new controls, a larger world window, and object galleries on the bottom. The Scene Editor replaces the Events area and the Editor area when you are in Scene Editor mode.

Let’s look at the Alice object galleries. There are two object galleries: a **Local Gallery** provided with the Alice software, and a **Web Gallery** maintained by the Stage Three Development Team at Carnegie Mellon University. (You need an active Internet connection to use the Web gallery.) The Local Gallery is visible at the bottom of the screen in Scene Editor mode, as shown in Figure 1-7. The galleries are organized as a tree of folders containing related objects. You can navigate the tree of galleries by clicking a gallery folder to enter that gallery, or by using the gallery navigation bar, which is just above the galleries.

![FIGURE 1-7: THE ALICE INTERFACE IN SCENE EDITOR MODE](image)

To view the Alice galleries:

1. You should already be able to see the Local Gallery. You are going to explore the galleries a bit before preparing to add objects to your new world. Click the folder icon in the gallery navigation
bar to move up one level in the tree of galleries, so that you can see the top level in the tree, as seen in Figure 1-8. Two icons are visible: one for the Local Gallery and one for the Web Gallery.

2. Click the **Local Gallery** icon to go back to the Local Gallery. Scroll left and right through the Local Gallery using the controls below the gallery folders, and you will see some of the many categories of objects available in Alice.

3. Find and click the **Animals folder** icon to open the folder. Scroll left and right through this gallery to see some of the animal objects available in Alice.

**OBJECT CLASSES AND INSTANCES IN ALICE**

Each of the tiles in the Animals folder represents a **class** of objects. A class is a group of objects with the same properties and the same methods. Objects in the same class are virtually identical to each other, except that the values stored in some of their properties may be different. For example, you could have an Alice world with two Penguin objects. They would both have the same methods and the same properties, but the values of some of those properties, such as location or color, might be different.

Each copy of an object from a particular class is called an **instance** of the object. The two penguins described in the last paragraph are two instances of the penguin class of objects. As you use Alice, you will notice that the object class tiles in the object galleries have the word class in their title and each begin with a capital letter, such as Class Bunny or Class Chicken, but once an instance of an object is placed in a particular Alice world, its name begins with a lowercase letter. Of course, it is possible to rename objects, so this distinction is not always maintained.

The act of adding an instance of an object class to an Alice world is called **instantiation**. The same terminology — classes, instances and instantiation—is used in most object-oriented programming languages.

You are going to add an instance of the first object in the first object folder in the Local Gallery to your new Alice world. You are going to instantiate a Bunny class object.

To add an instance of the bunny class of objects to your world:

1. Click the **Class Bunny** icon. A window with information about Bunny class objects, like the one in Figure 1-9, should appear.
2. Click the Add instance to world button to put a bunny into the world. This is sometimes called dropping an object into the world. You should see a bunny appear in the center of the World window.

3. There is a second way to add an object to an Alice world. You can click an object class tile and drag it into place in the World window. Try this now: drag and drop a chicken into your Alice world. This approach lets you place the new object wherever you would like on the ground in the world window, but does not show you the object’s information window first.

4. You should now have an Alice world with two objects—a bunny and a chicken. Notice that tiles for the new objects have also been added to the object tree. You really don’t need the chicken for the rest of this exercise. To delete the chicken, right-click the object or the object’s tile in the Object tree and select delete from the menu that appears.

POSITIONING OBJECTS

The layout tools to the right of the World window in Scene Editor mode can be used to manipulate objects. This area contains the seven standard tools listed in Table 1-1.

<table>
<thead>
<tr>
<th>Button</th>
<th>Name</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Pointer tool</td>
<td>Selects an object and moves the object parallel to the ground.</td>
</tr>
<tr>
<td></td>
<td>Vertical tool</td>
<td>Moves an object up or down.</td>
</tr>
<tr>
<td></td>
<td>Turn tool</td>
<td>Turns an object along its X-Y plane parallel to the ground.</td>
</tr>
<tr>
<td></td>
<td>Rotate tool</td>
<td>Rotates an object forward or backward (Z-axis rotate).</td>
</tr>
<tr>
<td></td>
<td>Tumble tool</td>
<td>Freely turns and rotates an object in any direction.</td>
</tr>
<tr>
<td></td>
<td>Resize tool</td>
<td>Changes the size of an object.</td>
</tr>
<tr>
<td></td>
<td>Duplicate tool</td>
<td>Creates a new instance of the same object.</td>
</tr>
</tbody>
</table>
To position the bunny:

1. The Pointer tool is already selected, so experiment a bit by using the pointer to click the bunny and move it around on the screen. Notice that you can move the bunny around on the ground with the pointer, but you cannot use the pointer to turn the bunny, rotate it, or move it up and down.

2. Click the Rotate tool and try turning the bunny a few times. You may be tempted to use the other tools, but please wait — for now they’ll only confuse things. You can come back and experiment with them after you’ve finished this chapter.

3. Before closing Scene Editor mode, you need to properly position the bunny in its starting position for the new Alice world you are creating. Remember, in this world, the bunny will move from the right side of the screen to the center, turn to face the camera, and then say “Hello, World!” Position the bunny using the pointer and then the rotate tool, so that it is near the right side of the window, facing toward the viewer’s left, as shown in Figure 1-10.

4. After the bunny is in position, click the large green DONE button to close the Scene Editor and go back to the standard Alice interface.

**ADDING SOME ACTION**

The next step is to add some motion to your world. You can start with something simple — making the bunny move across the screen — and then add a little more action. To make things happen in your world, you need to use methods for the objects. In the Editor area, you already have the default method, which is `world.my first method`. The full name of every method has two parts: the name of the object associated with the method (which comes before the period), and the name of the method itself (which comes after the period).
To add some action to your world:

1. Click the world tile in the Object tree in the upper-left corner of the interface.
2. Click the methods tab in the Details area. You will see that the name of the method in the Details area is simply my first method, but in the Editor, you see the full name of world.my first method.

The default event can be seen in the Event area on the top-right side of the interface. In Figure 1-11, you can see the tile for the default event, which shows that the event trigger is When the world starts, do, and the event handler is world.my first method. Any instructions you add to world.my first method will be executed when the world starts to play.

3. Click the bunny tile in the Object tree. Now you can see information about the bunny in the Details area. You should be able to see the bunny’s methods. You create new code for Alice objects by dragging tiles for objects, other methods, and control structures into the method you are currently editing. You get object tiles from the Object tree, method tiles from the methods tab in the Details area, and logic and control tiles from the bottom of the Editor area.
4. Make sure that the methods tab is selected in the Details area, and then find and drag the bunny move tile into the middle of world.my first method in the Editor area.
5. A short menu will appear asking you to choose the direction and amount you wish the bunny to move. Select forward for the direction, and then 1 meter for the amount.

Information that you must give to a method whenever you use the method is called a method parameter. Direction and amount are two parameters for the move method.

6. To test your world, click the Play button near the upper-left corner of the Alice interface. You will see the bunny move. It’s not much, but it’s a start. Click the Restart button to play the world again, and then click the Stop button to return to the standard Alice interface.

In this short tutorial, you are going to play the new world several times before saving it. However, when you are working on your own worlds, it is best to save the world before playing it.

7. To change the amount the bunny moves, click the 1 meter parameter in the move tile in your new method, and then choose one of the values in the drop-down menu that appears. You can also click other and then enter a number on the calculator-style keypad that appears. Try changing the amount a few times and then playing the world after each change until you can make the bunny move approximately to the middle of the screen.
8. Let’s add a few more instructions. First, find and click the **bunny turn to face** method tile. It’s in the Details area about 11 or 12 tiles below the bunny move tile that you just used. You will probably need to scroll down to see it. The parameter for this tile will be the object you want the bunny to face. Drag the tile into the Editor area below the **bunny move** tile, and choose **camera** as the object you want the bunny to face.

9. Next you are going to add two tiles to make the bunny speak, and then save the method. First, find and drag the **bunny say** tile into the Editor area below your other two instructions. The parameter for this method is the phrase you want the bunny to say. This parameter is a string parameter, which contains a string of characters from the keyboard. When the parameter menu appears, click **other**, and then type **Hello, World!** in the **Enter a string** input box that appears.

Why “Hello, World!”? One of the most useful and popular programming languages ever created was the C programming language developed at AT&T labs in the 1970s. The “Hello, World!” program first appeared in a book for the C language by two AT&T software developers, Brian Kernighan and Dennis Ritchie. Dr. Kernighan, who is now a professor at Princeton University, actually wrote the first “Hello World!” program for a tutorial on an earlier programming language, BCPL. Ever since then, it has been the custom for someone programming in a new language to write a version of the “Hello, World!” program as his or her first program. Congratulations, you’ve just done that in a new language. Let’s add one more instruction before saving the program.

To add additional methods and save your world:

1. After the **say Hello, World!** tile in your program, add another bunny say tile to make the bunny say “Hello, Dr. Kerninghan!”

2. Now play your world again (several times if you’d like), and when you are finished, click the **Stop** button to return to the Alice interface.

3. To save the world you created, click the **File** menu and then click **Save World**. Notice that Alice has File menu options to Save World and to Save World As. The first time you try to save a new world, you will see the Save World As dialog box, as shown in Figure 1-12. This dialog box looks like similar “Save As” dialog boxes in other Windows programs, such as Microsoft Word, with a navigation bar and other controls.
4. You should decide where you want to save the world, and then navigate the computer’s tree of
directories to find the right directory.

**TIP**

The default location for saving Alice files is the Windows desktop, but you may save files
wherever you wish. If you are in a course using Alice, please find out where your instruc-
tor would like you to save your files, as this may depend on how your classroom comput-
ers are configured. If you are working on your own, it is suggested that you save them
someplace where they will be easy to find, and that you remember (or write down) where
you saved each file. Appendix C contains more information about changing the default
settings for the save command.

5. Type the name **hello world**, with no punctuation, in the **File name** input box. Notice that the file
type shown below the **File name** input box is an **A2W** (Alice World File). This indicates that the file
you save will end in the extension .a2w, for Alice version 2 World. You should not change this.

Click the **Save** button to save your world.

### CLOSING AND RELOADING A SAVED WORLD

Next, let’s close the Alice program and then try to open your saved world.

To close and reload a saved world:

1. Click **File** on the menu bar, and then click **Exit**. The Alice program will close.
2. Re-open the Alice software. Click the **Recent Worlds** tab in the **Welcome to Alice!** dialog box and then
   open your “Hello world!” program. If the name of a world you wish to open does not appear on the
   **Recent Worlds** tab, you can click the **Open a world** tab and look for your world in the tree of directories.
TUTORIAL 1D — PRINTING CODE FROM AN ALICE WORLD

Before you finish this chapter, let’s try printing the code from your “Hello World!” program. Alice code is saved as an HTML web page, which you may then print on a printer, send to someone as an e-mail attachment, or use like any other Web page. You can also cut and paste items from the resulting Web page to other programs, such as Microsoft Word or PowerPoint.

This feature of Alice is not in the standard Windows format that is familiar to most people. Thus, it can be confusing, so go slowly through the steps below and pay careful attention as you do. You are going to save the HTML file to the root directory on the C: disk drive. If you are a student in a course using Alice, it is best to ask your teacher whether you should use the C: disk drive or another location.

To print the code from an Alice world:

1. Click File on the menu bar, and then click the Export Code For Printing option. The Export to HTML dialog box, as shown in Figure 1-13, should appear.

2. Notice that you need to tell the computer what to print. In Figure 1-10 you can see that you only have two items in your world that you can print. The first is the default event — when the world starts, do world.my first method, and the second is the default method, world.my first method. The code for an Alice world consists of the code for all of its events, and for all of its methods, but Alice lets you decide what parts of that code you wish to print. In this exercise, you will print everything, since you only have a small amount of code, but with large Alice worlds you may choose to print just a few items at a time. Make sure that the boxes in front of both items are checked.

3. You now need to tell Alice where you want to save the HTML file. This is the tricky part. There is a browse button, but it does not work as you might expect, so avoid it for now. You are going to save the file in the root directory of the C: disk drive (or in another location if so directed by your instructor). Enter the full path name of the new HTML page in the Export to: input box, as shown in Figure 1-13. Use the full path name, such as “C:\hello world.html” (or another name if directed to do so by your instructor).
4. You also need to add your name as the author of the code. Type your name in the *Author's name* input box.

5. Once you have entered the full path name for your new file and your name as the author of the file, click **Export Code** to create the new HTML document. Now you can find the document where you saved it and open it to look at your code. You can also copy the code to another location, such as on a USB memory chip or other device. You simply open and print the HTML page to print your code, just as you would for other HTML documents.
CHAPTER SUMMARY

This chapter consisted of an introduction followed by four hands-on tutorials. The introduction discussed the following:

- An algorithm is a step-by-step process; computer programs are algorithms.
- Most modern computer programming languages are object-oriented languages, in which programs are organized as a set of methods that manipulate the properties of objects stored in a computer.
- An object can be anything that is manipulated by a computer and consists of properties that contain information about the object and methods that can be used to manipulate an object's properties.
- The values stored in the properties of the object at any one time are called the state of the object.
- A class of objects is a collection of all objects that have the same properties and methods.
- Each individual object in a class is called an instance of that class.
- Alice is an object-oriented system of programming in which objects exist in a three-dimensional virtual world, which can be seen on a computer screen.
- Alice makes it easier to learn programming because of minimal memorization of syntax, visualization, and rapid feedback.

In Tutorial 1A, you explored the Alice interface, which has five main work areas — the World window, the Object tree, the Details area, the Editor area, and the Events area, as well as a menu bar, a Play button, an Undo button, a Redo button, a trash can, and a clipboard.

In Tutorial 1B, you learned to load and play an Alice world, and to use the speed slider control and the Pause, Resume, Restart, Stop and Take Picture buttons that appear when a world is playing.

In Tutorial 1C, you learned to create your own simple Alice world. You learned how to add objects from the object gallery, position them in the virtual world with the Scene Editor layout tools, and add instructions to the default method that is initiated by the default event.

In Tutorial 1D, you learned that Alice code may be exported to an HTML Web page and then viewed or printed from the Web page.

REVIEW QUESTIONS

1. Define the following terms:

   - algorithm
   - class
   - computer program
   - event
   - function
   - IDE
   - instance
   - instantiation
   - method
   - method parameter
   - object
   - OOP
   - programming language
   - property
   - state of an object
2. List and describe the five tabs in the Welcome to Alice! dialog box that appear when the Alice software is first started.

3. Describe the role of each of the five main areas of the Alice interface: the World window, the Object tree, the Details area, the Editor area, and the Events area.

4. What is the difference between a method and a function?

5. Briefly describe how to do each of the following:
   a. Add an object to an Alice world.
   b. Delete an object from an Alice world.
   c. Add an existing method to an Alice world.
   d. Change the value of a method parameter.
   e. Capture and store an image of an Alice world while it is playing.
   f. Save an Alice world.
   g. Print the code from an Alice world.

6. What is the difference between the Pause and Stop buttons in the window for a playing Alice world?

7. What is the function of the speed slider control in Alice?

8. What is the difference between the standard Alice interface and Scene Editor mode?

9. List and describe the function of the following Scene Editor layout tools: the Pointer tool, the Vertical tool, the Turn tool, the Rotate tool, and the Tumble tool.

10. Alice methods have full method names, such as robot.dance. Describe the meaning of the two different parts of the full method name.

EXERCISES

1. It can be very difficult for people to write clear and complete algorithms, such as a set of directions. We often take things for granted when writing directions and use our intelligence to interpret poorly written directions. For example, directions often contain clauses like “turn left at the third red light.” But what if one of the lights is green? Does it count? Would a person even ask this question, or just make an assumption about what the writer meant? How would a computer-ized robot handle such a problem? Try writing a detailed set of directions for a simple everyday process, such as making a pot of coffee, then exchange your directions with another student. Critique each other’s directions to see if they are clear and complete. Did the writer make assumptions that caused steps to be left out of the algorithm?

2. E-mail the HTML Web page that you saved with the code for your world to someone, such as your teacher or another person who will be impressed that you are beginning to learn three-dimensional, interactive, virtual reality programming with modern high-speed digital electronic computers. It might be best to send it as an attachment to a message.
3. Open the “hello world” Alice world that you saved as part of Tutorial 1C, and add some additional animation to the world. You may want to experiment with the methods to make the bunny move, turn, and roll. See if you can do the following:
   a. Make the bunny jump up and down.
   b. Make the bunny jump up, turn 1 revolution, and then land.
   c. Make the bunny jump up, roll 1 revolution, and then land.
   d. Make the bunny move and turn several times to go around in a full circle (or polygon).

   What is the difference between turn and roll? What difference does it make if you change the order of instructions in a particular world? When you are finished, click Tools on the menu bar, and then click World Statistics to see how long your Alice world has been open.

4. The methods available for the Bunny class of objects are called “primitive methods” and are available for all Alice objects. Certain classes of objects, such as the Penguin class, have additional methods available. Try starting a new world with a penguin and experiment with some of its user-created methods. These methods include wing_flap, glide, jump, jumping, walk, and walking.

5. Try creating, playing, and saving another Alice world on your own. While doing so, follow these two pieces of advice:
   a. Follow McGinley’s Rule for New Programmers: K.I.S.S. — Keep it Small and Simple. You should be encouraged to experiment, but be careful about getting in over your head. Try a few simple things with only a few objects to get started.
   b. Try to plan what you will do in the world before you start working on it. Keep in mind the Rule of the Six P’s: Proper Prior Planning Prevents Poor Performance. Many developers of Alice worlds like to outline or storyboard their work first. They draw a series of a few simple sketches of what they would like to try to make the objects in the world do. Professional programmers also use pseudo-code and flowcharts, which you will learn about in later chapters, to design the algorithms that methods will follow.

6. Try planning and creating a simple Alice world as part of a team of students. How does this experience differ from working on your own?

7. Alice has tool tips that appear if you place the mouse pointer on one of the tools, buttons, or tabs on the Alice interface and leave it still for more than two seconds. Table 1-1 in Tutorial 1C shows how the Scene Editor tools can be used to manipulate objects, but the table doesn’t tell you everything. The tool tip for the pointer tool in the Scene Editor mode tells you several additional ways to use the Alice pointer. See if you can find out what they are.

8. There are many Web sites that contain useful information about computer technology. Here are two for you to try: www.webopedia.com and www.wikipedia.com. Both are free online encyclopedias. Webopedia focuses on computer technology and provides a brief definition of terms and links to other sites. Wikipedia is more general. A Wikipedia is a free Web-based encyclopedia written collaboratively by volunteers. Pick a few of the terms from this lesson, such as algorithm, object-oriented programming, or IDE, and see what you can find out. You can also look up people, like Brian Kerninghan. The Wikipedia page at http://en.wikipedia.org/wiki/Hello_world_program has the “Hello, World!” program in dozens of different computer languages. The ACM “Hello World” project
page on Louisiana Tech's Website at http://www2.latech.edu/~acm/HelloWorld.shtml has 204 different examples of the “Hello, World!” programs.

9. *Building Virtual Worlds* is a course taught in the Entertainment Technology Department at Carnegie Mellon University. A Website for the course can be accessed at http://www.etc.cmu.edu/bvw/. The site includes sample worlds created with different software, including Alice. Visit the site if you would like to learn more about creating virtual worlds or to see some of the worlds that more experienced students majoring in Computing and Entertainment Technology have created.

10. In Lewis Carroll's original story about Alice in Wonderland, why did Alice follow the rabbit down the hole? What does her motivation have to do with creating successful virtual worlds? How is this related to one's education? The name Alice comes from the Lewis Carroll novels *Alice's Adventures in Wonderland* and *Through the Looking Glass and What Alice Found There*. Electronic editions of both, with the original text and the original illustrations by John Tenniel can be found in the Electronic Text Center of the University of Virginia Library at http://etext.lib.virginia.edu/toc/modeng/public/CarAlic.html and http://etext.lib.virginia.edu/toc/modeng/public/CarGlas.html. The Electronic Text Center also contains thousands of other works of literature that are available online, including the complete works of Shakespeare, the King James version of the Bible, the Koran, and the Book of Mormon. Their main page is on the Web at http://etext.lib.virginia.edu.