

Leone Learning Systems, Inc.
Wonder. Create. Grow.

Leone Learning Systems, Inc. Phone 847 951 0127
237 Custer Ave Fax 847 733 8812
Evanston, IL 60202
Email tj@leonelearningsystems.com

LogoSpeak

How to talk to the little alien in the Logo
Command Center

TJ Leone
June 2005



Introduction

Programming in Logo is kind of like having a text messaging conversation with an alien. The alien Logo has learned some of our words, but Logo thinks quite differently, so not all of the words translate very well. And there are lots of words that Logo doesn't know.

So the conversation goes like this: you text message some request to Logo. If Logo understands you, Logo does what you ask. If Logo gets confused, Logo will do something weird or will send you a message trying to explain why she can't do what you ask.

Sometimes the weird thing that Logo does is cool, so you decide to use it. Other times, you decide to reword your request so Logo can understand it better. You can also teach Logo new words.

The conversation continues until you're happy with the result or decide to do something else.

This document will help you speak to Logo in a way that makes sense to her, and to understand the things that Logo tries to tell you.

•
•
•
•
•
•

The MicroWorlds EX Welcome Screen

To make contact with Logo, you need to start up the MicroWorlds application. Here's what you see when MicroWorlds first starts up:



This is the MicroWorlds EX Welcome Screen. On the left is a list of buttons that suggest what you can do next:

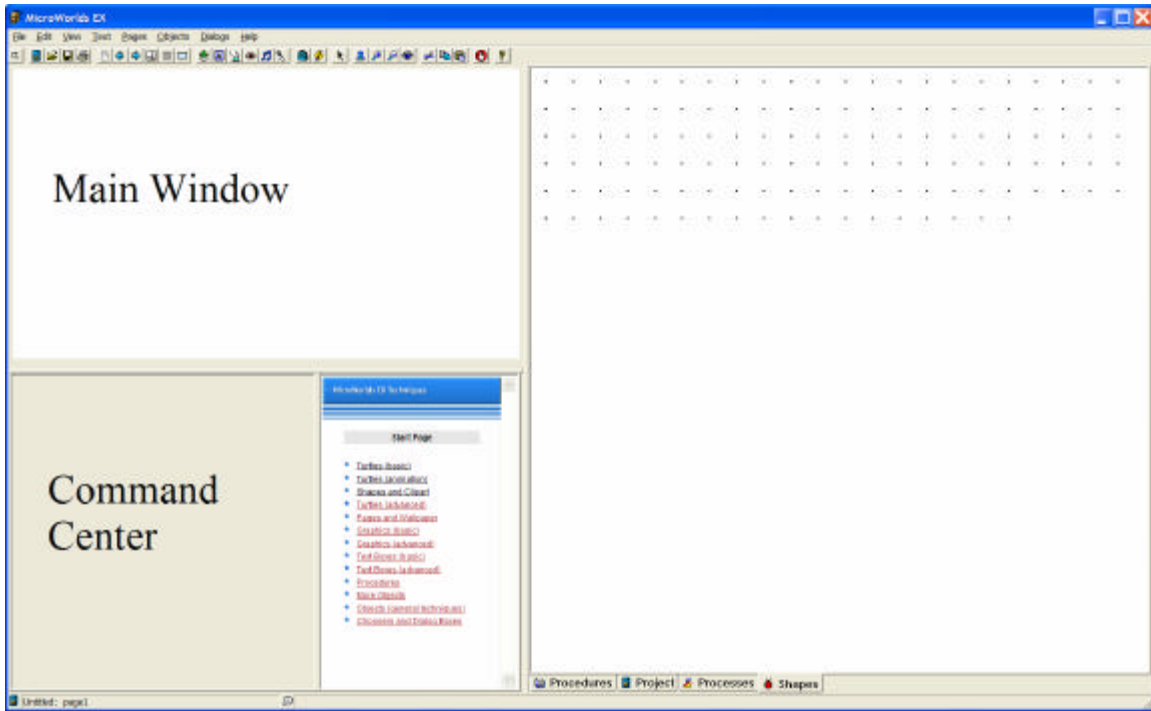
<i>Free mode</i>	Start a brand new MicroWorlds project
<i>Open Project</i>	Open up a MicroWorlds project you already started
<i>Tutorials</i>	Get step-by-step instructions in building different projects
<i>Samples</i>	Look at projects that others have created
<i>Quit</i>	Close the MicroWorlds application

For now, click on the *Free mode* button. This will bring up the main MicroWorlds EX window.

•
•
•
•
•
•
•

MicroWorlds EX Window

The picture below is about what you should see when you start a new MicroWorlds project in *Free Mode*. I added labels for the Main Window and the Command Center.





Messages

The Command Center is the place where you and Logo can send and receive text message. Click in the Command and type something there. For example, try typing

```
Hello, Logo
```

Then press the Enter key. That sends the message. After you send Logo the message, Logo will respond somehow. Logo might respond by changing something in the Main Screen or by typing something in the Command Center. When I typed my message, Logo responded right below my message in the Command Center:

```
Hello, Logo  
I don't know how to Hello,
```

The messages that you send to Logo are called *instructions*. Whenever you send Logo a message, Logo assumes that the first word in the message is some kind of *command*. If it knows how to do the command, it tries to do it. If it doesn't know how to do the command, it tells you so with an *error message*.

Error messages are messages that Logo sends you when it's confused. These messages can be very useful when you need to figure out why Logo is having a hard time understanding you.

•
•
•
•
•
•

I don't know how to...

Why did Logo say she doesn't know how to `Hello`,? Can't she just have a normal conversation? And what exactly *does* she know what to do?

Well, as I said earlier, Logo is an alien, and conversations with her aren't normal. For one thing, Logo always expects that when you text message her, you're giving her some kind of instruction to do something. The things that Logo can do are Logo's *procedures*.

Logo instructions are made up of procedure names and other words that Logo understands. You can teach Logo new procedures, but Logo also knows a bunch of procedures already. The procedures that Logo knows without being taught are called *primitives*.

You can see the vocabulary of Logo primitives by choosing Vocabulary from the Help menu. These are the words you must use to communicate with Logo, along with any words that you teach Logo, and some other words we'll discuss later.

Put a turtle onto Logo's Main Window, someplace off in a corner. Try typing this into the Command Center:

```
home
```

Notice that the turtle moves to its home position, which is the center of the screen. The Logo primitive `home` is in the MicroWorlds Logo. Here's what the vocabulary entry looks like:

home	Moves the turtle to the center of the page coordinates [0 0], pointing up. See setpos .	Given that there is a turtle on the page: <pre>home pd repeat 50 [home setpos se random 100 random 100]</pre>
-------------	--	--

Since Logo is an alien, her grasp on the English is rather tenuous, so spelling is extremely important. If text messaged you with the word `hoam`, you could probably figure out that I meant `home`, but Logo would just text back with:

```
I don't know how to hoam
```



I don't know what to do with...

Even if you're talking to Logo with a perfectly good primitive, you can get into trouble. For example, you can find the primitive `xcor` in the Logo vocabulary:

<code>xcor</code>	Stands for x coordinate. Reports the x coordinate of the current turtle.	Given that there is a turtle on the page:
	See <code>pos</code> , <code>setx</code> and <code>ycor</code> .	<pre>repeat 360 [setx xcor + 5 sety 100 * sin xcor]</pre>

Make sure you have a turtle in your Main Window and then send Logo the message

```
xcor
```

Logo sends back

```
I don't know what to do with -59
```

You'll probably see some other number besides -59. The `xcor` procedure reports the information that you see in the Xcor text box of a turtle's backpack.

So what exactly is Logo telling us with this I don't know what to do with...? Remember, as we said earlier, when Logo gets a text message, Logo is expecting some kind of instruction. But `xcor` just gives Logo a piece of information—a number that corresponds to the x coordinate of the turtle on the screen.

Logo doesn't know what she's supposed to do with that number.

The important thing to understand here is that there are two kinds of procedures:

Commands are used to give Logo orders. They cause something to change. For example, the procedure `home` is a command. It causes the turtle to move to the center of the screen.

Reporters are used to ask questions. They supply information, but they don't tell Logo what to do with the information. For example, the procedure `xcor` is a reporter we would use to ask Logo the x coordinate of the turtle. The information supplied by a reporter is called the *output* of the reporter.

So if we can't use reporters the way we use commands, then how *can* we use them? To answer that, we first need to look at the Logo message in the next section.

⋮

...needs more inputs

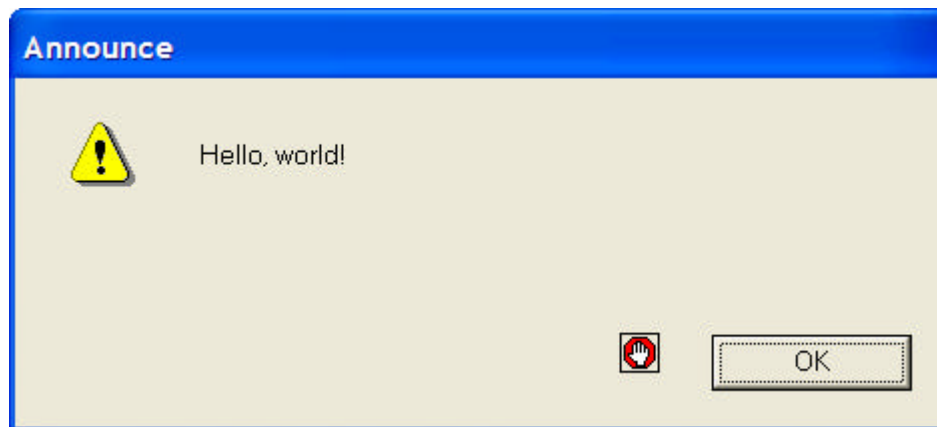
Suppose we want to use the `announce` command to make an announcement. If we send Logo the text message `announce`, Logo sends back:

```
announce needs more inputs
```

What's an input? Some procedures need information in order to do their jobs. That information is the input for that procedure. The `announce` command needs input so Logo knows what it's supposed to announce. Here's a text message for Logo that includes `announce` along with input:

```
announce [Hello, world!]
```

Type this message into the Command Center and press the enter key to send the message. What happens? You should see a window like this:



We can tell that `announce` is a Logo primitive because we can find it in the MicroWorlds Logo Vocabulary (Go ahead, you can look it up if you want. I'll wait).

But `[Hello, world!]` *not* a primitive. How does Logo understand that?

⋮

...needs more inputs (continued)

To see how Logo understands `[Hello, world!]` in the instruction

```
announce [Hello, world!]
```

we need to take another look at the entry for `announce` in the Logo Vocabulary. In case you haven't seen it yet, here it is:

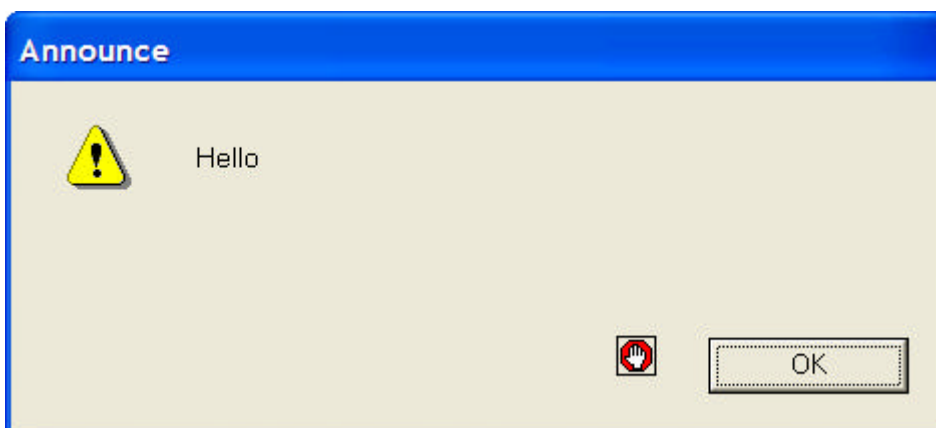
<code>announce</code>	Displays the <i>word-or-list</i> message in an alert box. Clicking OK closes the box. If you drag the alert box to a new position while it is being displayed, this is the position where the next alert box is opened in the project.	<code>announce [Guess what's in the package...]</code>
<i>word-or-list</i>	See question and answer .	

Underneath the vocabulary word `announce` appears *word-or-list*. This tells you that Logo needs a word or a list along with `announce` so that it can carry out the `announce` instruction.

Now, when we think of words and lists, we might think of words in a sentence or lists of things we want from the store. But Logo doesn't think like we do. To Logo, the thing that makes something a word or a list is the way you type it. For example, try this:

```
announce "Hello
```

If you spelled `announce` right, put a space after the word `announce`, and no space between the quotation mark and the H in `Hello`, you should see this:



⋮

...needs more inputs (continued)

We're used to using quotation marks at the beginning and end of a word or a phrase. But Logo is an alien creature. For Logo, a single quotation mark at the beginning means, "whatever comes next is a word". So,

```
"Hello "t1 7 "false "42 "banana
```

are all words. Oh yeah, notice that the number 7 above doesn't have a quote in front of it. Numbers are the exception to the rule for words—quotation marks in front of a number are optional. Any number, with or without a quote in front, is considered a word.

Just as Logo has her own weird definition of a word, she has her own idea of what a list is. According to Logo, a list is anything that begins with a left square bracket and ends with a right square bracket. So,

```
[fd 30] [t1 t2 t3] [How now brown cow?] [7 happy cg]
```

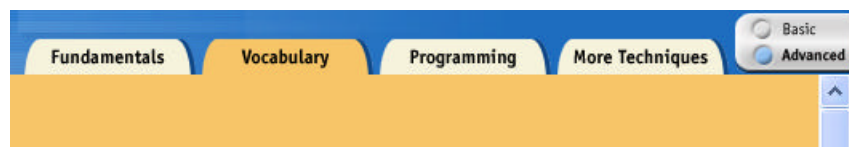
are all lists.

Now we can explain how reporters are used. The *output* of a reporter is used as the *input* to other procedures. When speaking to Logo, you use reporters the same way you use words or lists.

For example, since `xcor` reports a number, you can use `xcor` anywhere you could use a number. Try the instructions below. Make sure you have a turtle on your screen first, and move the turtle away from the screen so the `xcor` is not 0.

```
show 7
show xcor
fd 12
fd xcor
setc 19
setc xcor
rt xcor
```

By the way, some reporters need input, too. Look up the `sum` reporter and the first reporter in the vocabulary and try the examples. You'll have to select the "Advanced" radio button at the top right of the help window and then select the Vocabulary tab:





...does not like...as input

Some procedures are picky about the kind of inputs they like. For example, the `fd` command only likes numbers as input. Try this:

```
fd "hello
```

See what I mean? Here are some particular kinds of words or lists that you'll come across in the Vocabulary, with examples:

- | | |
|-------------------------|---|
| <i>number</i> | Any number (1, 2, 3, etc) or any reporter that outputs a number (color, xcor, sum, size, pensize, etc.) |
| <i>true-or-false</i> | The word "true, the word "false, or any reporter that outputs the word "true or the word "false (equal?, touching?, member?, etc.). Reporters that output "true or "false are called predicates. They usually end with a question mark. |
| <i>char</i> | A word that only has one letter ("a "x, etc.), or any reporter that outputs a word with only one letter (e.g., the reporter char). |
| <i>turtle-name</i> | The name of a turtle. Turtles are automatically given names like "t1, "t2, and so on, but you can pick names for them in the backpack or with the newturtle command. |
| <i>instruction-list</i> | A list with instructions in it, for example, |

```
[fd 50]  
[rt 90 bk 20 lt 40]
```

You'll also see [list-to-run](#) instead of [instruction-list](#)



The Author

TJ Leone owns and operates Leone Learning Systems, Inc., a private corporation that offers tutoring and educational software. He has a BA in Math and an MS in Computer Science, both from the City College of New York. He spent two years in graduate studies in education and computer science at Northwestern University, and six years developing educational software there. He is a former Montessori teacher and currently teaches gifted children on a part time basis at the Center for Talent Development at Northwestern University in addition to his tutoring and software development work. His web site is <http://www.leonelearningsystems.com>.