

SPIRIT NEWSLETTER

Volume 1, Issue 6
March 2010

Inside this issue:

Ask Alice	2
Riddles	3
Jokes	3
Ask Steve	4-5
Alice Lesson on Mendelian Genetics	6

Editorial By: Alka Harriger

The theme for this month's newsletter is Just Ask! Aligned with this theme, we begin with an *Ask Alice* article that shows you how you can create a "mad libs" Alice world that asks the user for several input values and produces a farm-based story using the provided input. If you'd like to try out the Mad Lib Alice world for yourself or your students, the article provides a web link for you to download your own copy.



Melissa continues the ask theme with a few challenging riddles. She's still keeping track of the people who submitted answers to any/all of her riddles and will award a prize to the person who answers the most riddles correctly.

At the last teacher in-service, several of you had asked questions about the future plans for Alice, so we decided that it would be a good idea to *Ask Steve* (Cooper). Jane Nawrocki compiled all of our questions for Steve and got us the answers. Fortunately, he offers reassurance about ongoing support for both Alice 2.2 and Alice 3.0. You might also recall that the Sims 2 characters were to be integrated within Alice. Read the article for information about this future enhancement. If you've enjoyed modifying Jane's list-based Alice world game templates for your own classes, you can learn about the future plans to address the bug related to dealing with a list with more items than what you really want. Steve also shares a work around Alice world (downloadable from the SPIRIT website) to address a limitation when Alice worlds include both timers and user input.

In lieu of our participant spotlight features, we decided to ask our participants to author articles related to their Alice work. Our first teacher to answer our call is Gary Cooper, a science teacher at Pike High School in Indianapolis, Indiana. At the last in-service meeting, Gary shared some amazing worlds created by his students, including one on ladybug genetics. In his article, Gary shares how he implemented the Alice-based activity to support his lesson on genetics as well as his students' response.

After you finish reading this Ask edition of the newsletter, I challenge you to submit an idea for a future article or a complete article regarding how you have used Alice and what impact (positive and negative) it has had on you and others.

Until next month, go out and enjoy the long-awaited sunshine!...Alka

Ask Alice By: Jessica Berger



Screen shots from
CowMadLibExample.a2w
Alice world

Dear Alice,

I am working on an Alice project trying to animate Mad Libs. I want the user to be able to answer a few simple questions and then use the answers to those questions in a simple story told in Alice. Can you please give me an example of how to make a simple Mad Lib this way?

-Mad Libby in Wonderland

Dear Libby,

This is a great idea for using the user input functionality in Alice. I have composed a basic Mad Lib featuring a cow, chicken, and horse. The story is as follows:

Once upon a time, there was a cow that lived in a barn. That cow's name was (name1). **Name1** the cow became friends with a chicken named (name2). They were (adjective1) friends. They liked to collect (pluralnoun1). and between the two of them, they had (number1) **pluralnoun1**. They decided to have a barn sale to sell their **number1 pluralnoun1** so they called the (adjective2) horse named (name3) and he said (**YesNo1**).

Before we begin the nitty gritty work of setting up the user input, we need to do two things. First, we need to get our characters in place and plan how the scene is going to proceed. I have added a cow, chicken, horse, and barn into my world. My cow is going to give the instructions for the mad lib, the questions will appear for the user to answer, and then the animals will tell the story using the user's answers. Secondly, we need to know what types of questions can be asked.

In your Alice world, while the world is selected in the object tree, click on functions. Scroll through the functions to the section entitled "ask user" and you will see three things you can ask a user; for a number, for yes or no, and for a string. I have incorporated these three types of questions into my example so you can see how they work.

I will begin my world with the cow giving some simple instructions using the cow.say method. Now for the fun part. We need to use those functions we saw earlier in the world functions, but we need a place to put them in our editor window. In the top right corner of the editor window, you will notice two buttons; create new parameter and create new variable. We need to create variables for each of the answers to the questions. Alice will store the answers as those variables and those can be used in the story that follows. In my story above, I named each of the variables in parentheses. The nouns, adjectives, and names are strings, and the other two are number and yes/no. Click on "create new variable" and name it, in this case, name1. Select "other" and be sure "string" is selected. Click OK. Do this for all your strings, be sure "number" is selected for any numbers and "Boolean" is selected for any yes or no variables.

Now that we have our variables created, let's get the questions to show up in our world. To do this, select the first variable that appears in your story and drag it into the editor window where you would like the question to be asked. The question will be asked in the form of a box that pops up in the world with a blank where the user can type their answer. When you drag the string variable into the editor window, click "set value" and select "default string". This default string is just a placeholder. With world selected in the object tree, click on functions and drag and drop "ask user for a string" into the box where "default string" appears. Select the question you would like to appear or click "other" to type your own question. For any numbers, select "set value" and set this to 1. Once again, this number is a placeholder for your variable that will be answered by your user. Drag and drop "ask user for a number" into this placeholder spot and

Ask Alice (continued from page 2) By: Jessica Berger

choose or type your question under “other”. For your yes/no question, drag the yes/no variable into the editor window and set the value to true (placeholder). Drag and drop the function “ask user for yes or no” into this placeholder and select the question to be asked.

Now the cow is going to tell the story and use the answers from the user to fill in the story. When the user answers the questions, Alice stores these answers as the values for these answers. Now when these variables are recalled in the story, the user’s answers will appear in their place. So, I am now going to have the cow start the story by using the cow.say method. I will have the cow say the first sentence. (Remember, when using text boxes, change the duration to be an appropriate amount of time for the user to read the text.) Then we need to, in the second sentence, recall the user’s answer for the first name. Use the same cow.say method and just select a placeholder phrase. Select the world in the object tree and go back to the functions. Scroll down to the “string” heading and drag and drop “a joined with b” into the say method where you selected the placeholder. In “a,” I type “That cow’s name was” and I am sure to put a space after the word “was” so that there will be a space before the second part of the string. For “b” I drag and drop the variable for “name1” into this spot so that the sentence will read in its entirety. The cool part about this function is that they can be stacked if your variable appears in the middle of a sentence or if you have more than one variable in a sentence.



Yours truly,
Alice

Riddles By: Melissa Weddle

1. I can always go up, never down, I can always turn left, never right, I am always hot when I'm cold
2. Alive without breath, As cold as death, Clad in mail never clinking, Never thirsty, ever drinking
3. Remove six letters from this sequence to reveal a familiar English word.

BSAINXLEATNTEARS

4. The more you take, the more you leave behind.

Jokes By: Melissa Weddle

In Computer Heaven:

The management is from Intel, The design and construction is done by Apple, The marketing is done by Microsoft, IBM provides the support, Gateway determines the pricing.

In Computer Hell:

The management is from Apple, Microsoft does design and construction, IBM handles the marketing, The support is from Gateway, Intel sets the price.



Ask Steve By: Jane Nawrocki



Dr. Stephen Cooper, currently a professor in the Computer Graphics Technology Department of Purdue University and one of the world's foremost experts of the Alice programming language, graciously agreed to answer questions asked by the 2009 teacher cohort of the SPIRIT program. Below are Dr. Cooper's answers (in **bold**) to the questions posed by the teachers.

What are the long-term plans for Alice 2.2 and Alice 3.0?

We believe that Alice 2.2 and Alice 3 serve different audiences, so it is our intention to stay with both. Alice 2.2 works with a pre-CS1/pre-AP audience (or a programming class for non-majors), and such an environment doesn't need the ability to go back (and forth) to typing Java. Alice 3 will be appropriate for use in a CS1 (or AP) course, where one of the goals of the class is typically to learn the syntax of Java. That is the driving force behind Alice 3. Additionally, the Sims 2 characters are only going to be part of Alice 3; so if you want to build worlds with the Sims 2 characters, this would be another reason for moving to Alice 3.

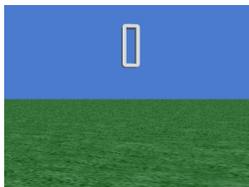
If I add one too many items to a list when I'm creating it, how can I delete it without using code? I'd like to drag the extra list item to the trash. The only solution I've come up with is: 1) add a dummy/duplicate item in that extra list location or 2) delete the entire list and begin again.

Unfortunately, this is a known bug in Alice. It is on our list to be corrected, but as we don't frequently hear the request for this to be fixed, it is not that high on the priority list.

Describe your favorite assignment that you have given to your college Alice programming students.

I love Pop-A-Mole (Chapter 9 of textbook). I also like the Frogger game (Chapter 9), as well as the Bumper Cars game (Chapter 10) and the Midas touch and its variations (Chapter 8). The truth is that my favorite assignments are the ones that students design on their own (based on my open assignments). Most of the questions and assignments from our text come from students and their projects.

Are there any plans to add a feature to Alice 2.2 that allows the Alice worlds to be made into stand-alone executable files?



Screen shot of
UserInput_timer.a2w
Alice world

This is a great question. When work was first being done on Alice 2, Alice actually got installed into the operating system, so that the .a2w extension could be associated with Alice files, and could automatically launch the world when an .a2w got double-clicked on. Unfortunately, many IT people didn't (and don't) trust academic software and wouldn't allow Alice to be installed in their computer labs. So, we went with a separate approach where Alice worlds were intentionally non-executable (outside of the Alice application). In other words, XP or 7 or Vista has no idea what an .a2w or an .a3p is, and won't run them. This made the IT people happier, but made students and teachers less happy.

Ask Steve (continued) By: Jane Nawrocki

Along this same line of thinking, have you heard of Mama by Eytam? Their web site states “Mama IDE is an improvement of Carnegie Mellon's open source *Alice* development environment, supporting unicode, build and edit of 3D objects, uploading animations to YouTube, and much more.” Their web URL is: <http://www.eytam.com/mama>



I am unfamiliar with it. I'm not sure that it's worth \$10 (not to mention the \$34.80 original price). Building your own tutorials using stencils is pretty easy to do on your own. There is mention of adding your own 3D models using a tool called ArtOfIllusion. I don't know how easy ArtOfIllusion is to use, as compared to Maya or 3DS Max, which are the tools we've been using to build custom 3D models (See: <http://www.aliceprogramming.net/models/tutorial/tutorial.html> for details). They allow you to export an Alice world as an exe (which could be a nice feature – see my response to that question above). The limited forum comments seem to indicate it was built off of Alice 2 so does not have the bug fixes included in Alice 2.2.

A colleague is writing an Alice game that has a timer and also asks for user input. He would like the timer to continue running while waiting for the user input. However, it appears that when the request for user input is encountered, the timer stops running. (The point of the game is to answer the questions is less time than the next user. So the timer becomes very important.) What do you suggest?

The truth is that we haven't done a lot with user input. We provided the 3 questions to let the user enter various types of information, but didn't seriously consider the environment in which these features would be used. Thus, the behavior (which is probably wrong – at a minimum, there should be an option allowing the world to wait or not) is that you see results. A “not so great” workaround is to define a behavior in response to the user typing a key sequence. See my sample world for one way to do it. (Note: The workaround world is attached to this article.) Refer to the following link to see the Alice world.

http://www.itpossibilities.org/Data/Files/UserInput_timer.a2w

What has been the most unexpected thing about your move to Purdue?

Two things:

The number and quality of folks to work with (good).

The coldness of winter (bad).



As a participant in the 2008 SPIRIT, I sincerely thank Dr. Cooper for sharing his expert, “in-the-know” information and ideas. I look forward to working and speaking with him in the future. Dr. Cooper may be reached at: coopersc@purdue.edu

Alice Lesson in Mendelian Genetics By: Gary Cooper

Purpose:

Teachers are always on the lookout for lesson ideas that will enhance and enrich the students' experience in addition to increasing achievement. In this lesson, students are instructed to create an ALICE world presenting a concept in Mendelian Genetics. (**Biology Core Standard 6; B.6.1**) Concepts such as Incomplete Dominance, Co-dominance and Multiple Alleles were assigned to small groups. Students research the concepts and then use the ALICE world they create to illustrate the concept. When complete, ALICE worlds will be on display for all to see. The ultimate goal is that the ALICE worlds created by students might be used to teach the concept to other students in the class.

The target class for this project is my first year Biology students. The class is comprised of upperclassmen with the majority being sophomores. This group is rather heterogeneous in terms of academic interest and achievement. We are a "majority-minority" school so there are many African-Americans in the class along with several Hispanic students and a few Caucasians. There are 14 boys and 9 girls in the class.



Screen shots from Alice world created by students to explain ladybug genetics

General Procedure:

I created an ALICE world covering the concept of Dominance and presented it to the students in my class. I then reviewed the ALICE world with them two times. The first time I showed them how I programmed the events in my ALICE world. The second time through, I demonstrated how my program described the concept of dominance in, "Lady Bug Genetics." I gave them the rest of the period to "play!"

On day two, I required students to produce storyboards for their ALICE worlds. Many completed their storyboards and began to create objects in their worlds.

On day three, I prepared handouts from the manual we received during our summer workshop. I was especially interested in them manipulating objects, using billboards and proper use of properties and methods. During the second 50 minute period, many groups were able to create objects in their worlds and begin the programming. As usual, there was a fair amount of difference between those who "got it" and those who didn't. Most of my time was spent with students who needed encouragement and those whose storyboards were too vague.

Day four was used to complete projects. I insisted that each person in the group had time on the computer programming the world their group was creating. I used a format similar to the one we used at the March 1, 2010 in-service day.

Results and Preliminary Analysis:

About 33% of the students actually finished their ALICE world during the class time given to them. I have given students a due date of March 24. They have a computer lab in the media center to use during study hall or during our "after school" program. I will also try to get them some extra time during class.

While the project is still incomplete, I feel it was successful because students were actively engaged in researching the concepts in order to present them accurately. (Though they take some prodding at times to stay on task) They thoroughly enjoy working with ALICE! It sure is an upgrade from the old lecture I used to deliver on these topics. (Though I still had to fill in some gaps.) I should also add that I had only one student who did not stay on "ALICE task" during the time we worked with ALICE. This is a significant upgrade from a general lab scenario for this set of topics.

While time is always a consideration, I garnered time from deleting some activities that were redundant and I had them do a little more homework than usual. No one has complained about the extra work.

Gary can be reached at Pike High School, Indianapolis, Indiana, glcooper@pike.k12.in.us, 317-387-2771