Unit I: Introduction: Installation and Alice Navigation

Chapter 4: Start, Save, Reopen and Modify your Program

In this chapter, you will learn different ways to start a new project in Alice and become familiar with the basic steps for managing your work. You will learn how to modify, save, use Save As, and reopen an existing project so that you can continue working on it later. You will also explore how to set preferences to customize the Alice environment and make it better suited to the way you work.

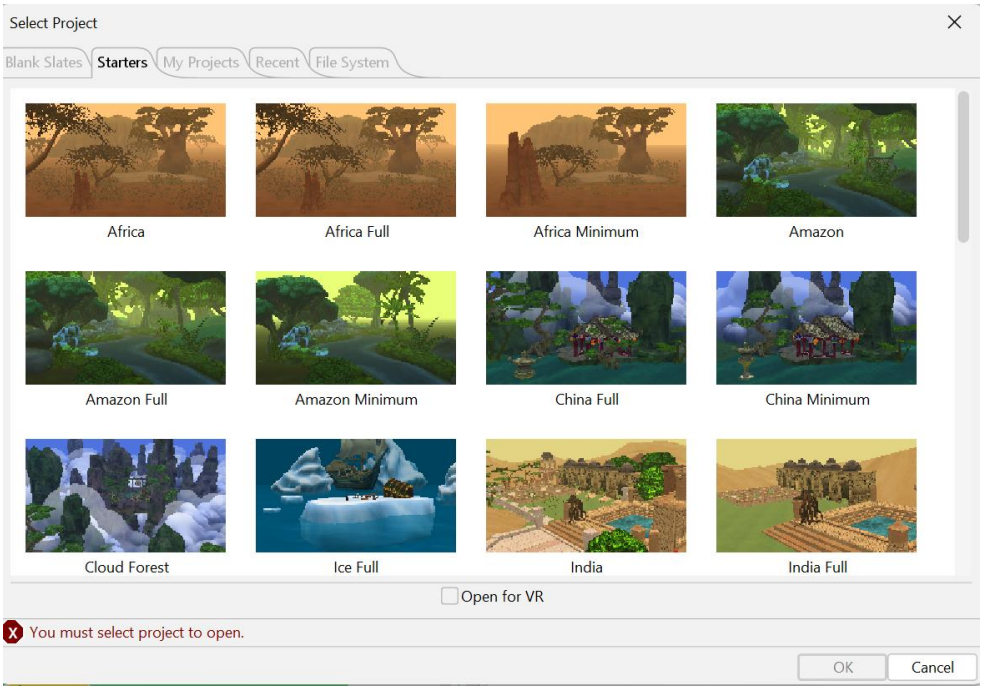
Objectives

- Instead of selecting a template choose other tabs: starters, my projects, recent, file.
- Use the arrows to move, rotate. and resize the characters and other objects.
- Move the camera and change the angle.
- Edit an existing project.
- Add arguments to commands

Starters

When Alice starts, instead of picking one of the blank templates, click on the starter tab: If you look at the scroll bar, you can see that there are many more choices.

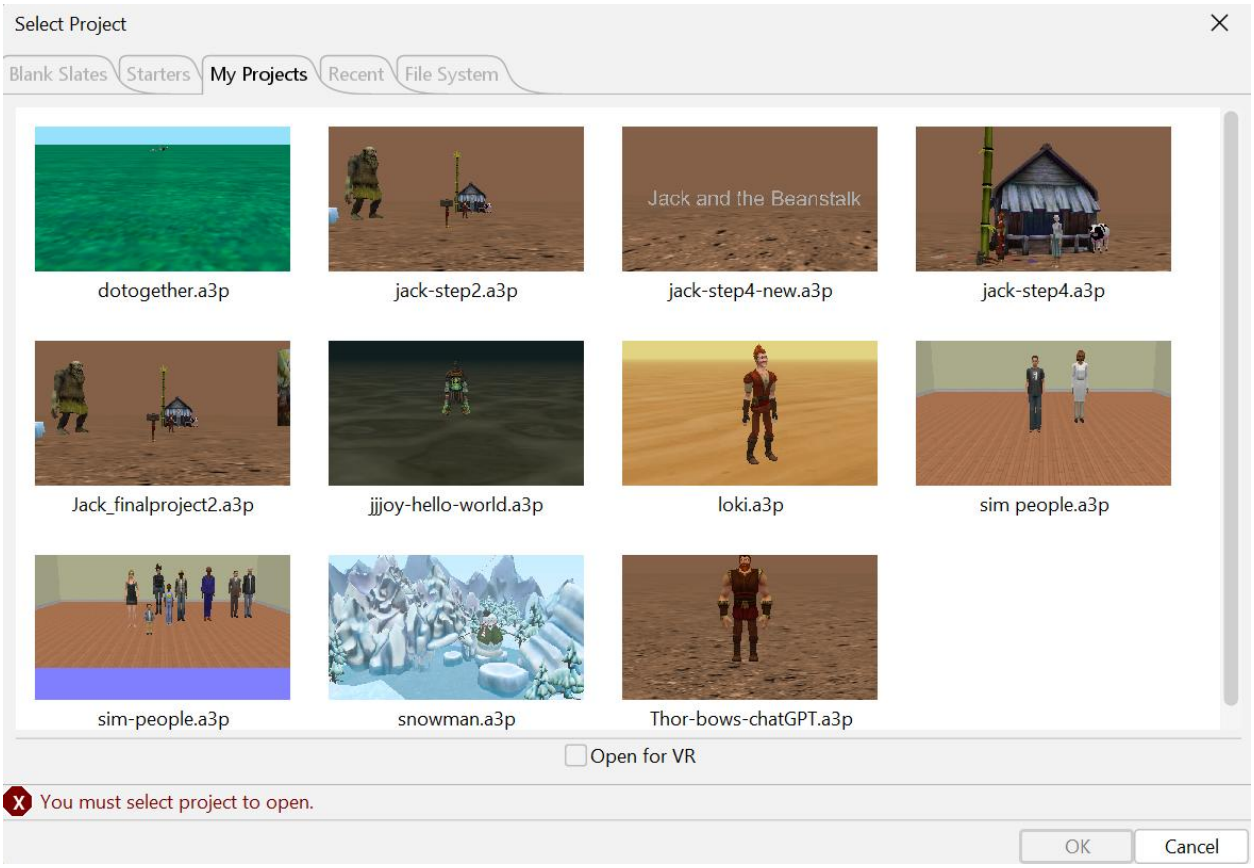
Each of these choices will give you a complete scene to start with.



There are two problems with these starters: first it can make it hard to find something you just added and want to move. You can solve this problem by adding something by dragging it into the scene instead of just clicking on it.

The second problem is that it takes up more memory. If crashing is a problem on your computer, you can delete some of the objects in the starter scene.

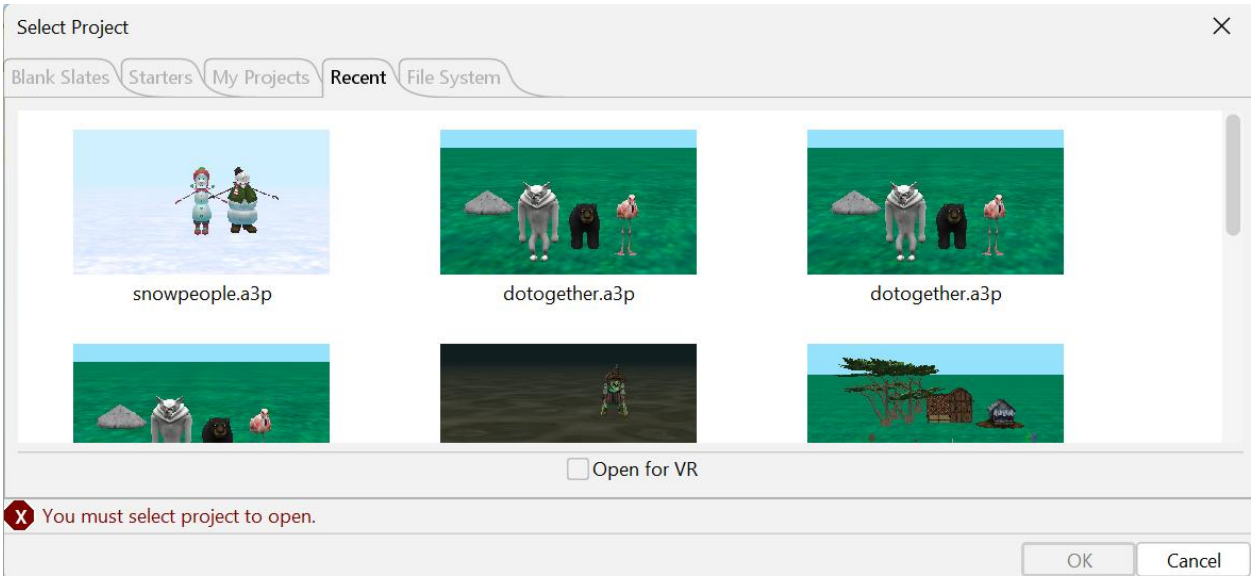
My Projects



The third tab, **My Projects**, shows project that you have saved in your projects folder. You need to create a projects folder.

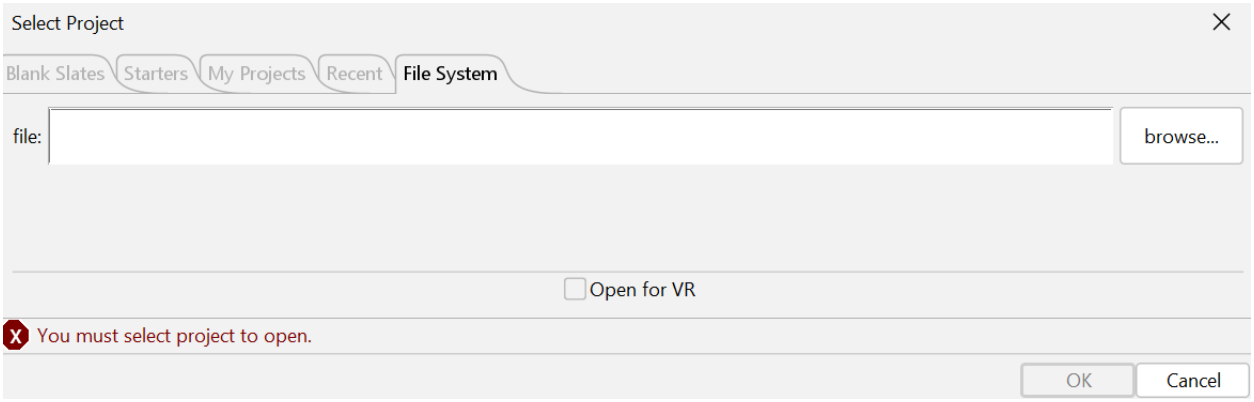
1. Create a folder in your storage device called **AliceProjects**, or other easily recognizable name.
2. In Alice, when you save your first program, locate this folder and save your program.
3. The next time you want to open or save, it will start with this folder.

Recent



The **Recent** tab shows the projects that you have opened recently. This might not be the same as the ones in **MyProjects**. For instance, you may have opened one of the projects from zebra0.com, or some other safe source.

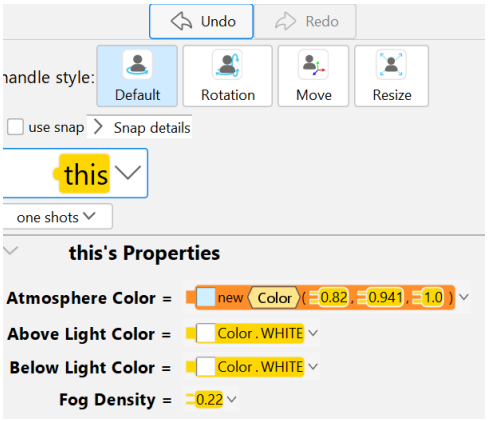
File System



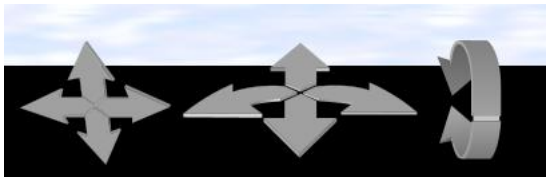
The last tab, **File System**, lets you browse for a project. This is especially useful if you have the project stored on a removable drive that you used on a different computer.

Scene Setup

If you add the snowwoman by clicking, then the snowman, they will be on top of each other. Click on one of them, and you will see a yellow ring at the base that indicates that it is selected. Then you can drag it to another position. As you drag it you will see some arrows, called “handles”. You can move the object backward, forward, and drag the ring to rotate the objects. Try rotating both figures so that they are facing each other.



The Camera

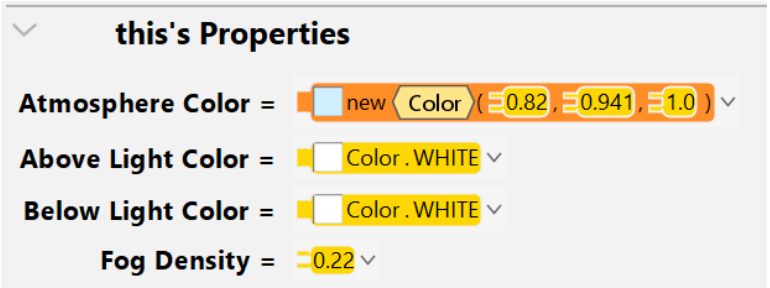


At the bottom of the stage in scene view there are 3 sets of arrows to adjust the camera. The arrows on the left **move the stage** to the left, right, up and down. The middle arrows change the camera angle by **rotating the stage** to the left or right, The up and down arrows in the middle zoom in and out. The 2 arrows on the right move the camera up (looking down on the stage) or move the camera down (eye level). Experiment with all the arrows to see for yourself.

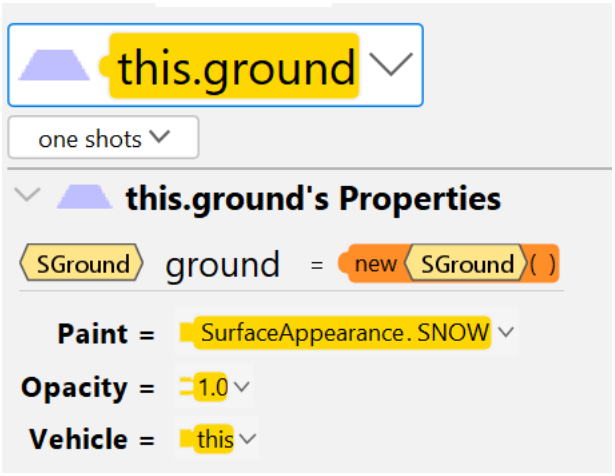
Properties

Objects have several properties.

Click on the stage itself. In Alice, the stage itself is referred to as **“this.”** On the right you will see a panel where you can change the Atmosphere Color, the above and below lights color, and fog density. Experiment!



The Ground



Select ground from the drop-down box. You can change the **Paint** for the ground, choosing from Snow, Ocean, Swamp, etc. The **Opacity** sets the visibility of an object: 1 is fully visible; 0 is invisible and 0.5 is partially visible or transparent.

In Alice, **Vehicle** refers to an object that is followed. For instance, if we want Thor to hold a sword, we move the sword to Thor's hand, then make the vehicle for the sword Thor's hand. Then, if Thor's hand moves, the sword moves with it.

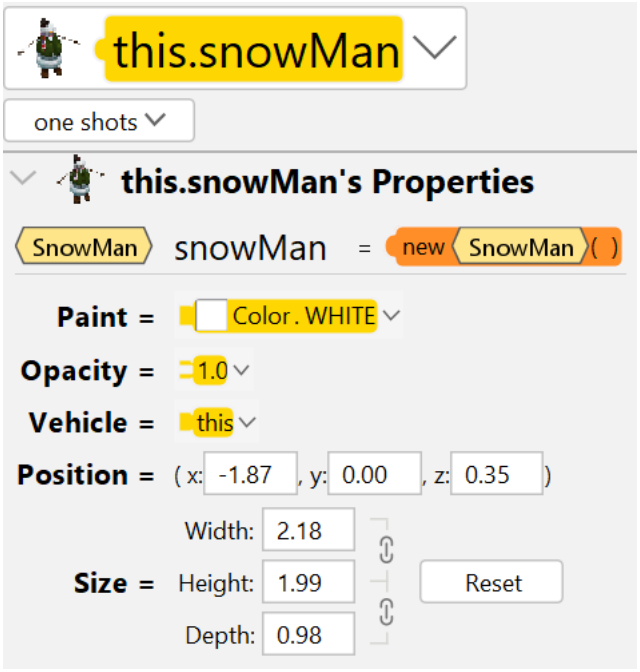
The Snowman

Click on the snowman to select him.

Change the Paint to RED:

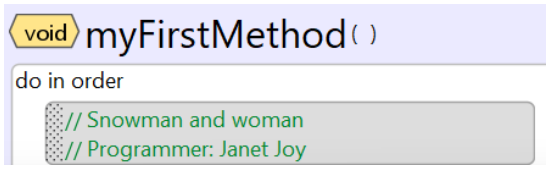
Then Click the UNDO button at the top right, or Ctrl+Z to undo.

Change the opacity to 0.2 to make him somewhat transparent, then UNDO.



Next, click **Vehicle** and select snowwoman as the vehicle. Then drag the snowwoman around and the snowman follows. If you drag the snowman, the woman doesn't move. UNDO until the Vehicle is back to "this."

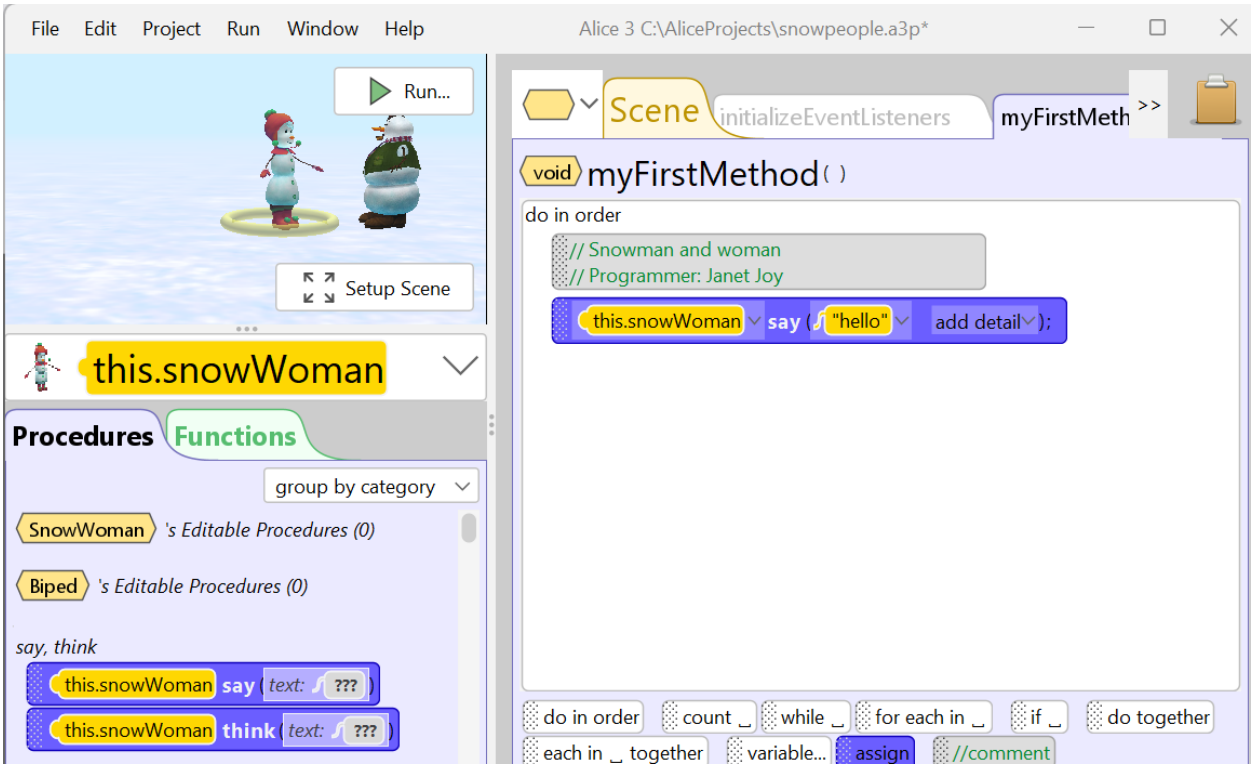
Code



Click on the Edit Code button to go back to Code view. From the bottom of the code view, drag a comment tile into the code and drop it beneath **do in order** in **myFirstMethod**. Type in the name of the program and the programmer.

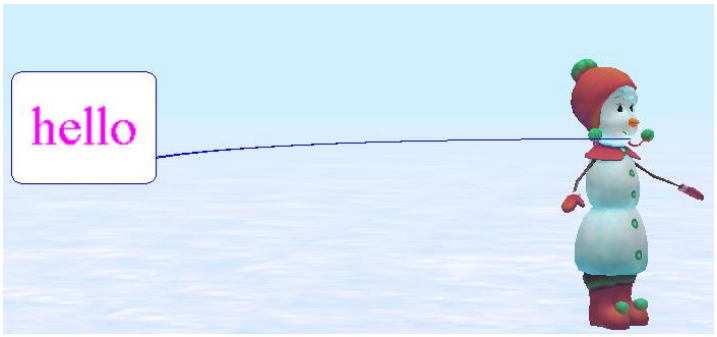
Say Procedure

Select the snowwoman, then drag the say procedure into the code and drop it below the comment.



Run the program. The snowwoman says hello, but the words disappear quickly.

Notice that the **say** procedure forces you to put in the words (a string) that the character says, but that there is a drop-down button after the words that says “add detail”. Look at the choices and select duration of 10. That will leave the words up for a while. Experiment with some of the other choices and see if you can duplicate this:

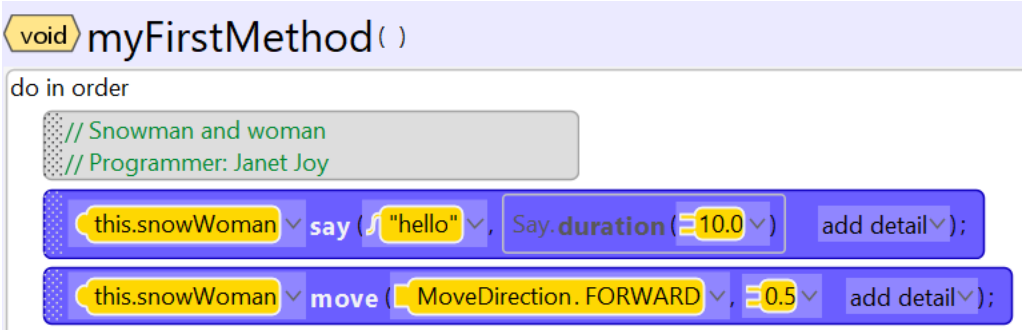


The rules for the values that a procedure gets (text, duration, position, etc.). are called **parameters**. The values you send, such as “hello”, 10, or left are called **arguments**.

Take some time to try a few other procedures. Don’t worry, you can always undo.

The “say” command **requires** one string for the words that the character will say. The other values are **optional**.

Try the move procedure: it requires two arguments, the direction, and the amount.



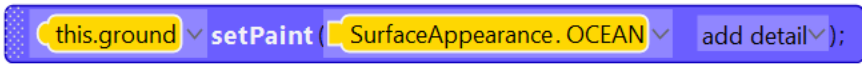
Order Matters

Note that the statements above are under the words “do in order”. They are executed in **sequence**. The snowwoman doesn’t move forward until the **say** command is finished. Drag the **say** command down and drop it after the **move** command. Run it. Now she moves forward first and then says “hello.”

Change Properties with Code

The properties we changed in setup scene can also be changed in code. Try each of these:

Select the ground, then drag the **setPaint** tile into the code:



Select the snowman and add the code shown below to make him invisible, then he says “goodbye:”

Notice that he gradually fades away (setting the opacity to 0 with a delay of 2.0), then you see the words “Goodbye” after the snowman is invisible. Experiment! Rearrange the statements and make up your own story.