Alice Yeti Program

0) If you are annoyed by the *this* prefixes to scene objects, you can defeat this by selecting
   `Window -> Preferences -> Display "this." -> Display "this." except for Scene Properties`
   from the Alice cascading menus (at the top of the Alice window).

1) Begin by reloading the scene you created in the previous lab, and repeated in the picture below. If for
   some reason you are unable to reload the scene, then rebuild it.

2) Be certain the Alice interface is in edit mode so you can write a program. Create an initial program
   consisting of one instruction - a procedure that moves the snowboard forward by 2 units. Run this program
   to see how it behaves.

3) You might notice a problem with the instruction from Step 2 in that the snowboard moves in the wrong
   direction. (This seems to be corrected in later releases of Alice.) If this is a problem replace the
   FORWARD in the instruction by clicking it.

4) One difficulty you will notice is that when the snowboard moves the Yeti does not. We could add
   instructions to make the Yeti move at the same time as the snowboard, but a better solution is to attach
   the Yeti to the board and then just move the board (and the Yeti will move along with). To accomplish
   this, insert a Yeti setVehicle instruction as the new first instruction of your program. Now check out
   your program's behavior.

5) Next, add instructions to your program in order to make the Yeti on his snowboard slalom around the
   two boulders, then ski though the right gate opening, turn and ski out the left gate back toward the
   front of the scene. Be patient, this may take a few experiments of writing instructions, testing and
   modifying instructions.

6) Next, insert an instruction at the beginning of this program to cause the Yeti to say "Wee!" To do this
   you will need to use the Custom Textstring option for the parameter of the say procedure.

7) Now modify your program by removing the instruction that caused the Yeti to say "Wee!" that was
   created in Step 5. Instead, you should replace this instruction with one that causes the Yeti to think
   (not say) "I can slalom around these boulders."

8) Modify your program further so that after the Yeti is finished slaloming the yak moves forward by a
   distance so that it appears to bump into the right boulder. When the yak reaches the boulder, your
   program should then cause the penguin to jump up in the air, perform a complete forward summersault
   and land back down on the boulder.

9) Return to setup mode and insert a second baby yeti object. Note that Alice doesn't allow a single
   program to contain two objects with the same name. Position this second yeti so that it appears to
   share the snowboard with the first yeti. Change the program so that it causes both Yetis to ride on the
   slalom. (This can be accomplished by inserting a single instruction.)
10) If time permits, this is a good time to learn about program variables. Virtually, every programming language, including Alice, supports the use of variables. A variable is really just a name for a small piece of program data. You may think of a variable as storing one thing. To use a variable, it must first be declared.

In Alice, you declare a variable by dragging the "variable..." tile from the bottom of the window and dropping it into your program at the desired position (generally at or near the top of your program). Every variable must have a type. The type restricts the kind of things that the variable can store. For example, a variable of type WholeNumber can store only integers and a variable of type Quadruped stores a Quadruped kind of object.

Begin by creating a variable to store a baby Yeti. Your variable declaration should be dropped at the top of your program and be defined according to the window below:

```
Clicking OK declares a variable called "chillyBooBoo" that can store Biped kinds of objects. You should notice that the instruction that is inserted into your program looks like this.

The arrow in this instruction indicates that it performs an assignment. The assignment assigns the value that the variable stores. In this case chillyBooBoo is assigned yetiBaby; in other words, chillyBooBoo has just been assigned as another name for the yetiBaby object. To verify this, select chillyBooBoo from the menu of objects just below the scene picture in the upper left corner, then add an instruction to the end of your program to cause chillyBooBoo to move toward the camera.

You have just discovered that an object can have more than one name. chillyBooBoo and yetiBaby are really just two different names for the same object. The next thing to investigate is what occurs when you assign a variable a different object. To do this you need to insert an assignment instruction. In Alice the "assign" tile at the bottom of the window is used to insert an assignment instruction. Drag and drop this tile at the end of your program. Choose the name of your second baby yeti object to complete the "???" part of the assignment. Now add another instruction after the assignment to cause chillyBooBoo to roll to the right by 2. Did the first baby yeti, the second baby yeti or both yetis roll?

You may want to experiment more with assignment instructions because we will use them in the future. How many variables can store the same object? What happens each time you assign a new object to a variable?