

Unit I: Introduction: Installation and Alice Navigation

Chapter 2: Hello World



In this chapter, you will write your first program in **Alice 3**: a 3D character who says, "Hello World", creating your own 3D world. You will learn how to create a scene by selecting a template and adding characters and other objects to the scene. You will then create and edit code to control the actions of those objects. You will also learn how to add **comments** to your code and understand why comments are important for explaining and organizing a program. Finally, you will practice identifying and debugging simple errors so that your program works as intended.

Objectives

- Write your first Alice program.
- You will learn to add code.
- Create and modify a scene in the design view.
- Select and add 3D characters to the scene.
- Select, add and modify SIM people.
- Use the arrows to move, rotate, and resize the characters and other objects.
- Move the camera and change the angle.
- Add comments with your name and description of the program.
- Create code to have the character say "Hello World"
- Run the program, test, debug, and save it.

Select Project

When you start Alice3 the first thing you need to do is select a project. For your first project you should select one of the Blank Slates. Don't worry about making the wrong choice. You can change this later. Select one, and then click OK.

Notice that there are also tabs for

- Starters (complete scenes with buildings, trees and other objects).
- My Projects (to open projects from the My Projects folder.)
- Recent (to open recent projects)
- File System (to browse your files and search for a project.)

For your next project you can experiment with some of the other tabs.

Code View

After you select a starting template, you will be in the **Code Editor**. This is where you add code to the Alice project. Code controls what the movie **does**.

In the top left panel, you see a small view of the scene. There is also a button to **Setup Scene**. Click that button to add objects to the scene, setup the starting position, change the camera angle and do other design tasks. The screen view is where you control how the movie **looks**.

Naming Conventions

In the top left panel, you see a small view of the scene. There is also a button to Setup.

As you create your Alice virtual world you will need to make up names for the objects, variables, and methods that you add.

The following conventions help programmers to write code that is easy to read and understand.

- Classes start with an upper-case letter.
- Constants are all uppercase: PI for instance is a constant, it always has a value of 3.14...
- Everything else starts with a lower-case letter.
- After the initial letter, you may use more letters, digits and underscore.
- Names cannot have a space! Programmers use an uppercase letter to separate words instead of a space: for instance, bigRabbit or greenMushroom. This style is called **camel case**. It makes it easy to see where each word begins.
- It is very important to use meaningful names. Names of objects are usually a noun or an adjective and a noun: chicken or littleChicken and bigChicken. Sometimes a digit is added: pixie1, pixie2, pixie3 for instance if you are working with them as a group.
- When you add object to the scene you can select the default name or make your own name. For instance, instead of **thor** and **freya**, you could use the roles in your movie ,such as **father** and **mother**.
- Do not use single letters for variables except when they are used to count.
- Be consistent. If you develop a naming convention and stick to it, it will be much easier to remember what you have named things.

Preferences

if you want your code to look like that in the videos and illustrations, go to File, Preferences, programming language, and pick Java. You only need to do this once.

If you pick Alice, which is the default, you're going to see a variable described as **TextString**, instead of **String**, and that isn't going to match what you see in the lessons. Another advantage of using Java is that that's probably the language you will learn next,

In Alice, go to File, Preferences, programming language, and select Java.

Important Notes

When you go to the code editor you will be in a procedure called `myFirstMethod`. **DO NOT EVER DELETE `myFirstMethod`!**

Editing

If you make a mistake, you can undo it by clicking Ctrl+Z (Windows) or Command+Z (Mac).

If you add an object and want to delete it, right click on the object in the object list (where it says this.) and select delete.

Hello World

Traditionally, the first program you write is to display the words "Hello World." Usually, displaying text is the simplest program you can write. But it means a lot!:

It means that you have downloaded and installed the software or figured out where there is a computer lab with the software. You have figured out how to run the software, how to put in code, and how to run the program. You have probably also figured out how to **debug** (fix errors.)

Comments

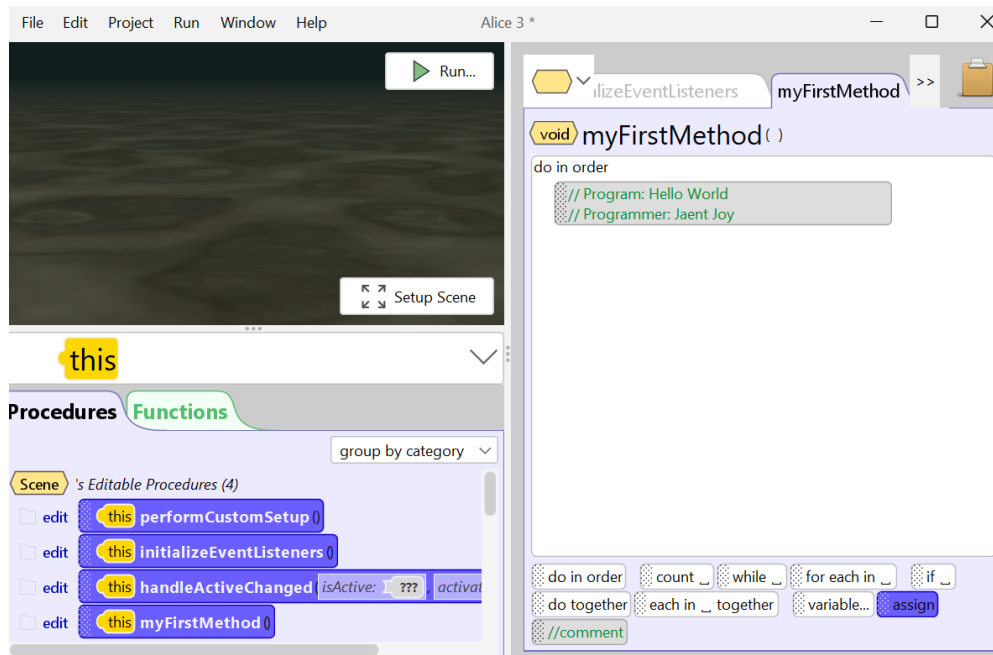
Comments are important in programming because they make code easier to understand, maintain, and collaborate on. They are ignored by the computer but read by people. Here are the main reasons comments matter:

- **Improve readability:** Comments explain what a section of code does, making it easier for others (and your future self) to understand.
- **Simplify maintenance:** When you revisit code months later, comments help you remember why certain decisions were made.
- **Help teamwork:** In collaborative projects, comments make it easier for team members to understand and modify each other's code.
- **Aid debugging:** Comments can describe the intended behavior, making it easier to spot where the code isn't working as expected.

- **Document complex logic:** If an algorithm or calculation is difficult to understand, a comment can explain it without requiring someone to analyze every line.

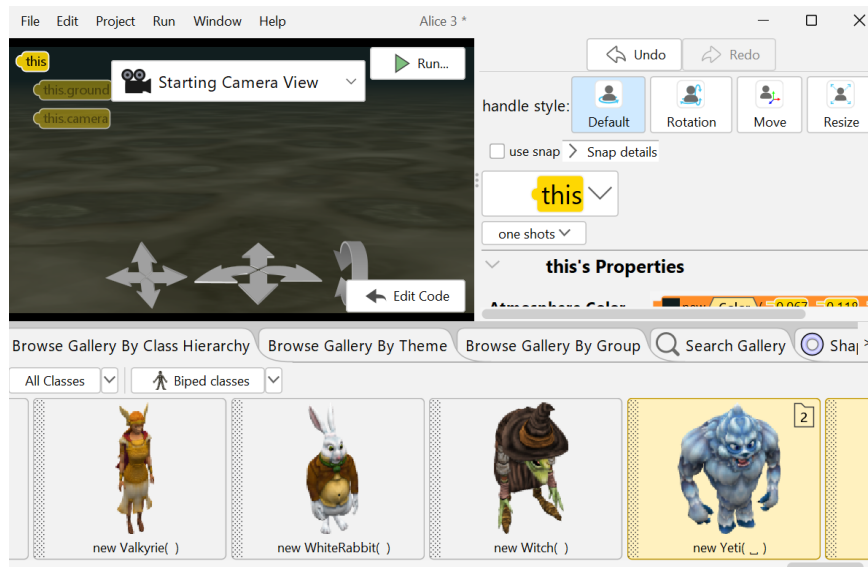
The Hello World Program: Add a comment

The first thing to do is in the code view window, drag the comment tile from the bottom on the right into the code for **myFirstMethod** and drop it beneath the line that says, “do in order.” Type in the name of the program and the name of the programmer (you.)



The Hello World Program: Add a character

The next thing we need to do is go to the scene editor and add someone to say, “Hello World.” Click the **setup scene** button that is in the lower right of the scene window and then select the **biped class**: characters that stand on two legs. That’s usually people and monsters, but in Alice there are a few others, such as the rabbit as you can see in the illustration.

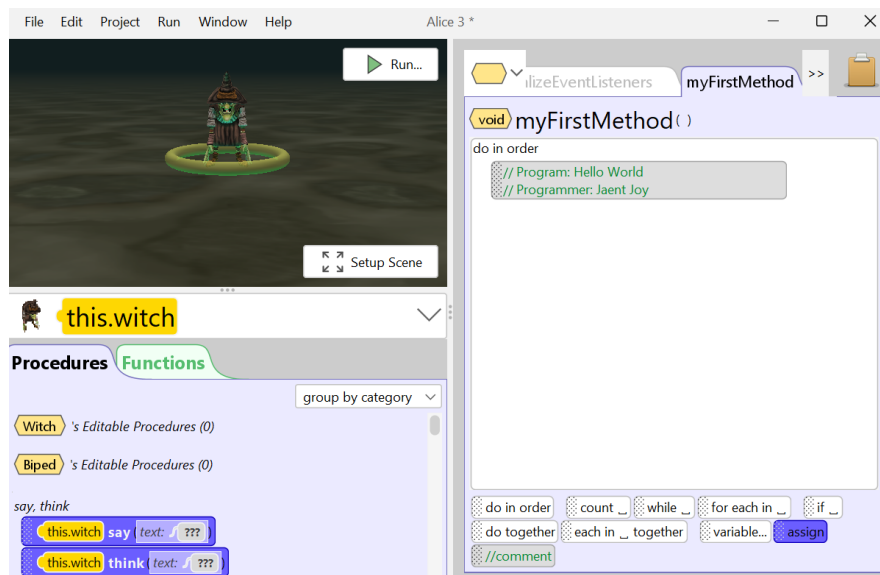


Click whatever biped you want. In this example, the witch is added. When you click, you must give it a name (according to the naming conventions. In the example the name is witch.

The Hello World Program: Say

Next, click on the **Edit Code** button.

With the witch (or whatever character you added) selected (if the witch isn't selected, you can select it with the drop-down window that says “THIS”)

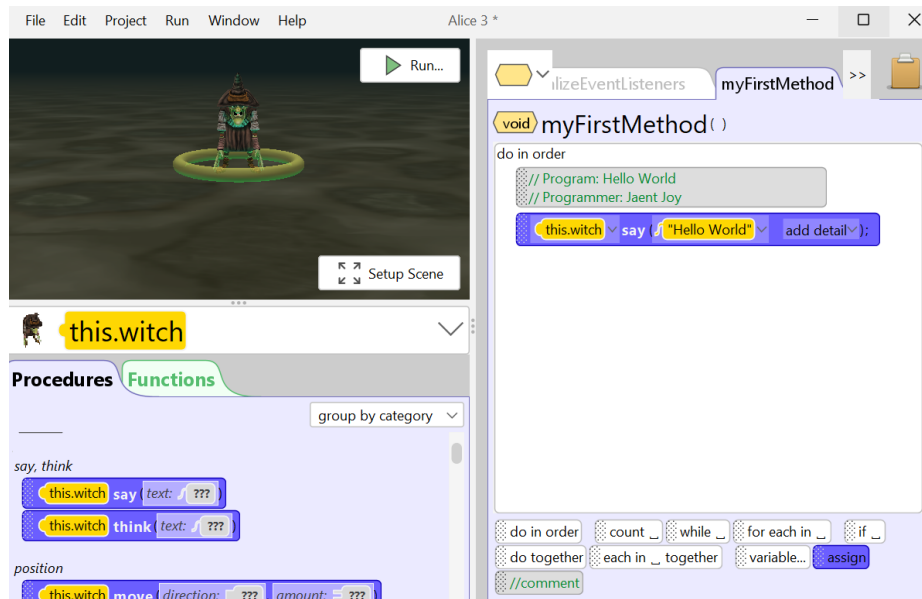


In the bottom left panel, there are two tabs: **Procedures** and **Functions**.

Procedures perform actions.

Functions return values.

Look at the list of procedures that the witch can perform: the first group is **say, think**. Drag the tile that says **this.witch.say** to the code window and drop it under the comment. In the pop-up window select **Custom TextString** and type **Hello World** as the value, then click OK.



The Hello World Program: Run

Next, click on the **Run** button.

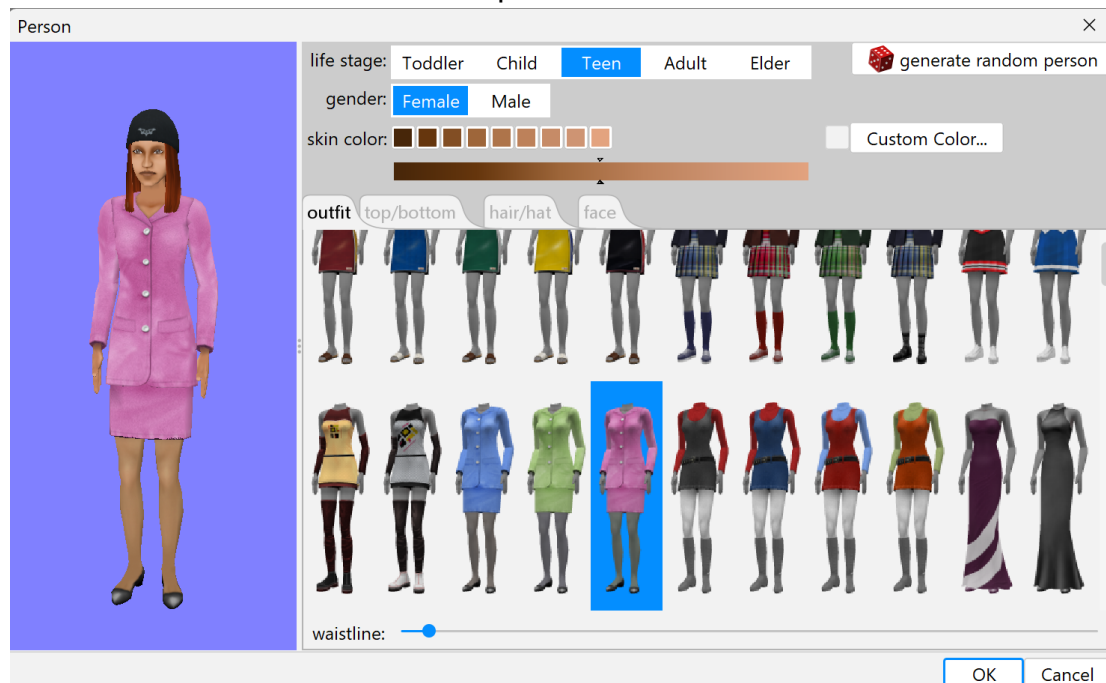


Sim People

In **Alice 3**, **Sim People** are customizable human characters that come with the software. Unlike many other characters in the gallery, you can change their appearance before adding them to your scene, including their age, gender, clothing, hairstyle, and facial features.

How to create a Sim Person in Alice 3

1. Open your Alice 3 scene view.
2. In the **Gallery**, click the **Class Hierarchy** tab.
3. Select the **Biped** category.
4. Choose one of the **Sims People** (such as Adult, Teen, Child, Toddler, or Elder).
5. The **Person Builder** window will open.



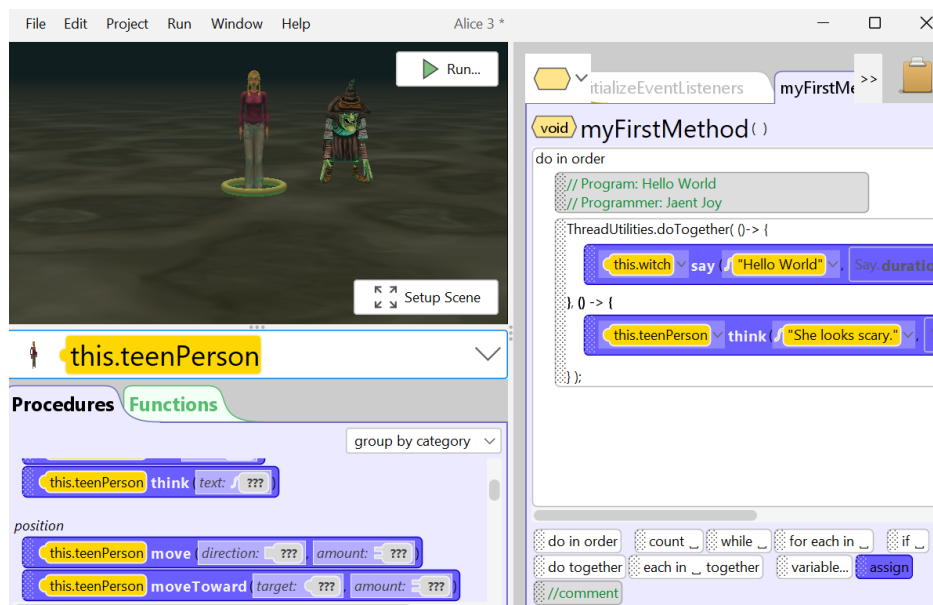
6. Customize your character by selecting options such as:
 - Life stage (adult, teen, child, etc.)
 - Gender
 - Skin tone
 - Hair style and hat
 - Facial features
 - Clothing (full outfit or separate top and bottom)
 - Body shape (waistline slider) ([Alice](#))
7. Click **OK** when you're finished.
8. Give your Sim Person a name (for example, waiter, mary, or coach) and click **OK** again. The character is added to your scene. ([Alice](#))

Why use Sim People?

Sim People are useful because they:

- Look more realistic than many of the cartoon-style characters.
- Can be customized to fit your story.
- Include many built-in animations such as walking, turning, sitting, and gestures.
- Are ideal for creating animations involving people, such as conversations, dances, or bows.

DoTogether



The **doTogether** control structure lets you make several procedures happen at the same time. This is very useful when you start doing animation. In animation, you might need to have the character bend the elbow and shoulder at the same time to wave.

In the illustration, a Sim person has been added. Without the **doTogether**, the witch says “Hello World”, then the teen thinks “She looks scary.” Drag the **doTogether** into the code window, then drag the **say** and **think** procedures into the **doTogether** block. Now they do the two procedures at the same time.