

# Getting Started With Alice



By Ruthie Tucker

under the direction of Prof. Susan Rodger

Duke University, July 2008

[www.cs.duke.edu/csed/alice/aliceInSchools/](http://www.cs.duke.edu/csed/alice/aliceInSchools/)



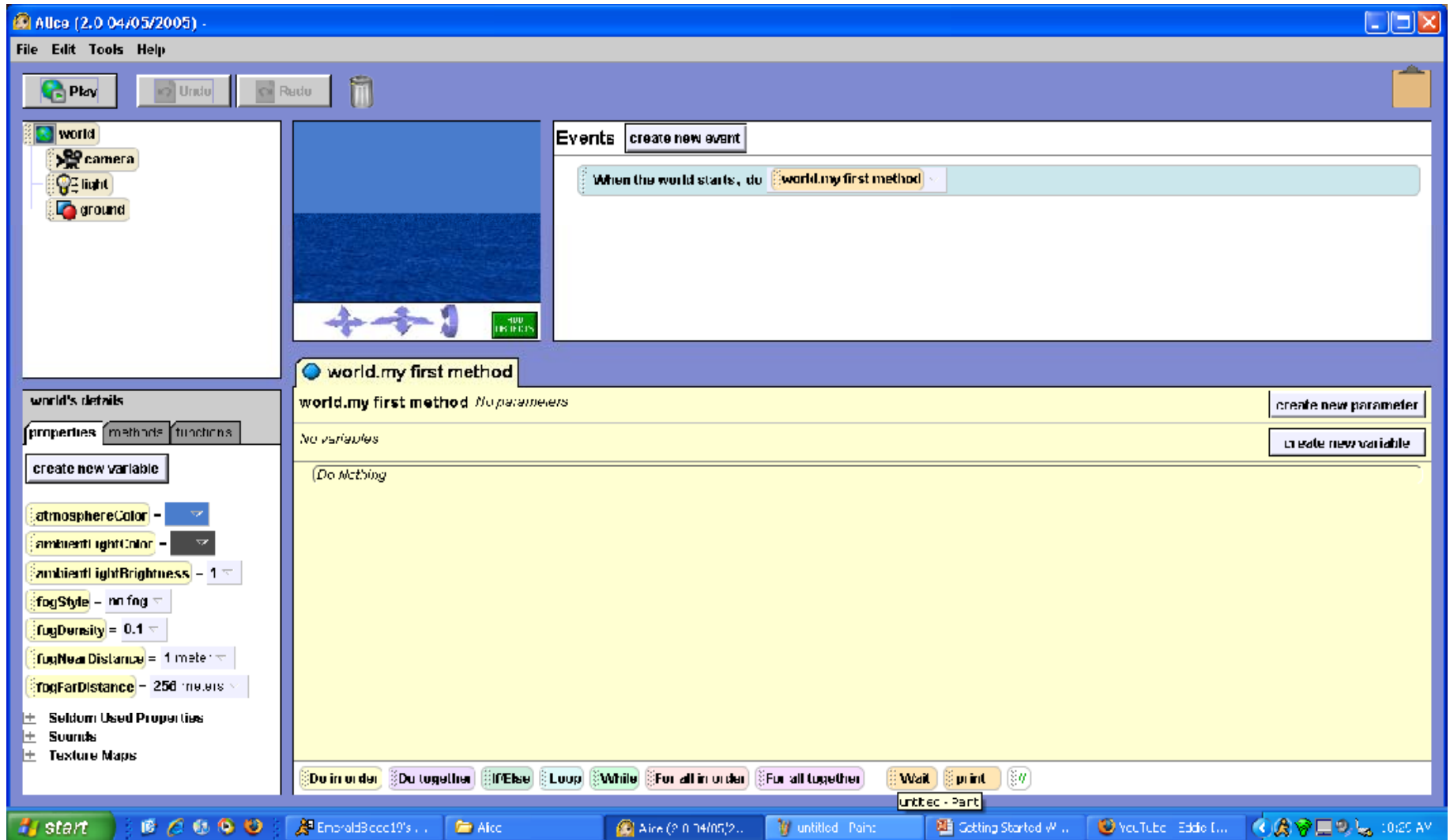
Hi! my name is Alice and this program was named after me. Alice will teach you how to make your own animations.

# Let's Get Started

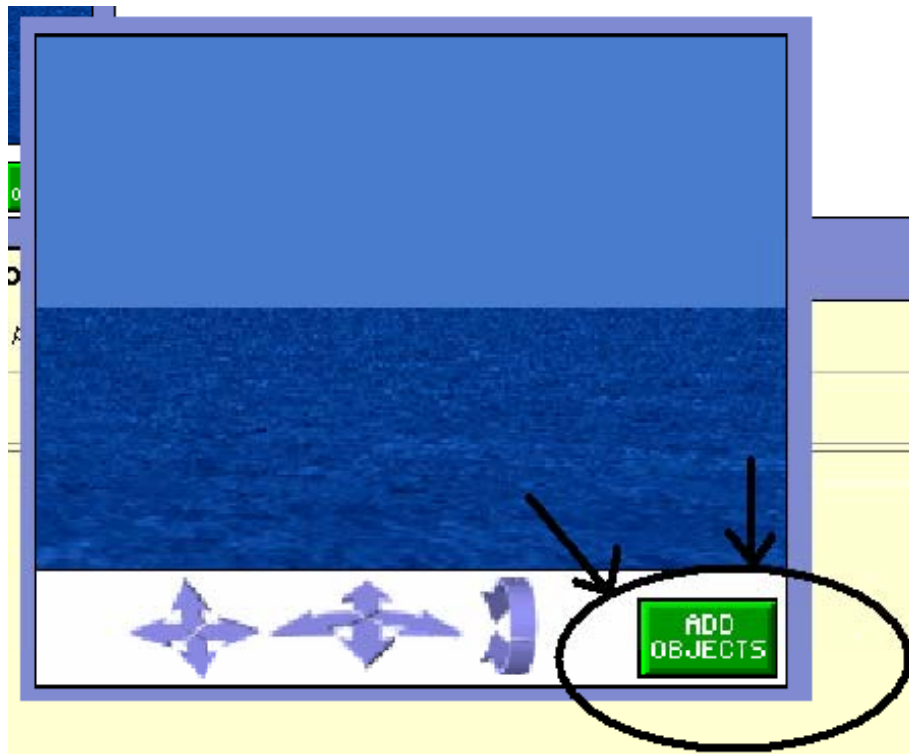


- The first step in making an Alice world is to pick a background.
- When you open Alice, this box to your left will automatically pop up. You may have to click on the “Templates” tab
- Today, we are going to select the water world.
- Click on “water” and then click on the “open” button.

Once you have chosen the water world this is how your screen should look.



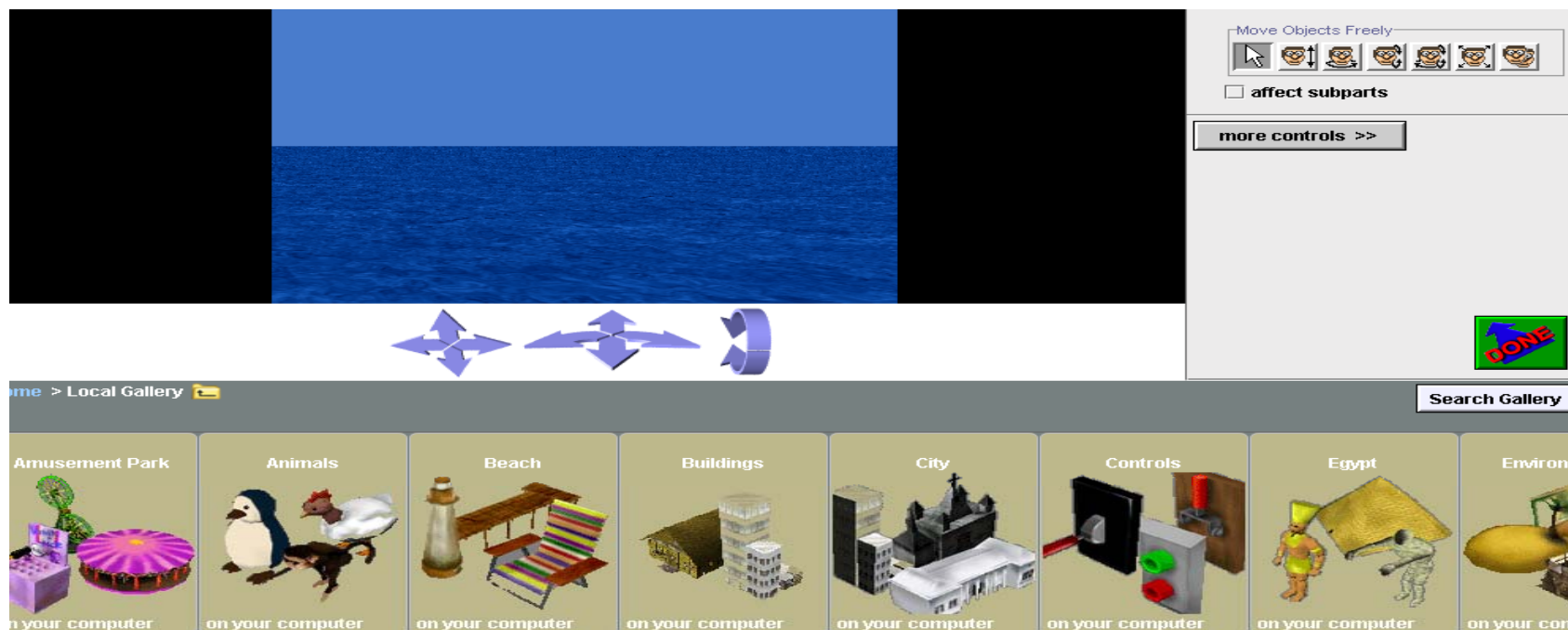
# Adding objects to your world



- Now lets add some objects to our world.
- To the right of the water world picture, there is a green button labeled “Add Objects.”
- Click on it.

# Adding objects to your world

- Once you have clicked on “add objects” a gallery of objects will appear, on your screen, below your water world.
- This is your local gallery of objects.
- There are lots of folders, of objects.



# Adding objects to your world



- Select the animals folder and scroll over until you find the Goldfish.



- Click on the Goldfish.
- And then click on “Add instance to world.”
- The Goldfish should appear in your world, on the water

# Adding objects to your world.



- Your goldfish will be very small. Don't worry about this yet.
- First we need to add another object to your world.

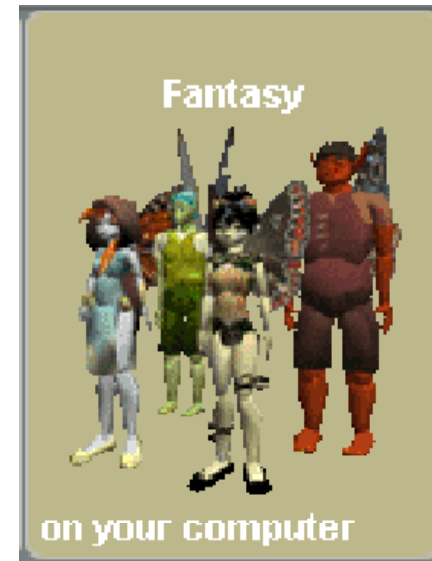
- Click on “Local Gallery” to go back to the starting point of the Local Gallery.





# Adding Objects to your world

- Now, scroll over and select the “Fantasy” folder.
- Then, select the fairies folder



# Adding objects to your world



- Add this fairy into your world, by clicking on her and then clicking on “Add Instance to World”.
- Your screen should now look like this.



# The Object Tree



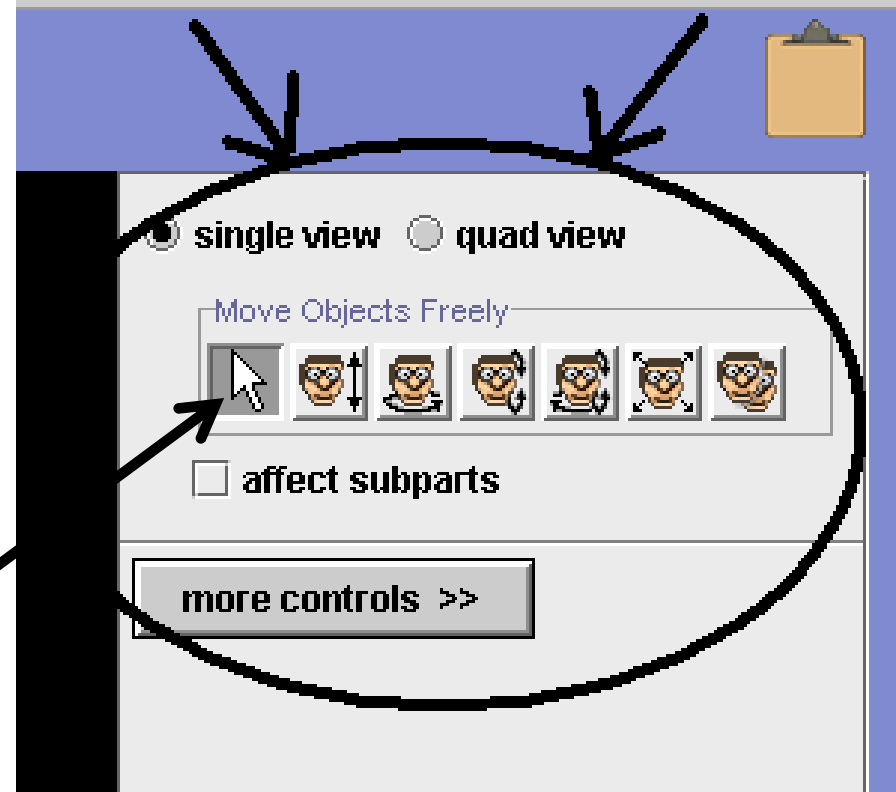
- When you add objects to the world they will appear in a list on the left of your screen.
- This list is called your “Object Tree.”



Now we have two objects in our world. Now we need to position them!

# Positioning your objects

- Now look at the right side of your screen.
- All of the buttons with the faces are designed to help you position your objects.
- Click on the white arrow, try clicking on your fairy and moving her around.

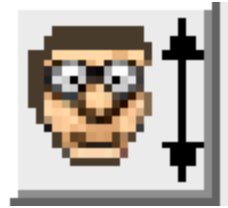


# Positioning your Objects

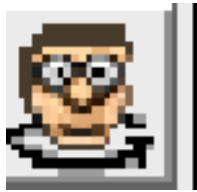


- Try moving your fairy so that she is to the right of your fish.
- Your screen should look like this

- The button with the up and down arrows will move your object up and down.
- Click on it and move your fairy so she is hovering over the water



# Positioning your objects

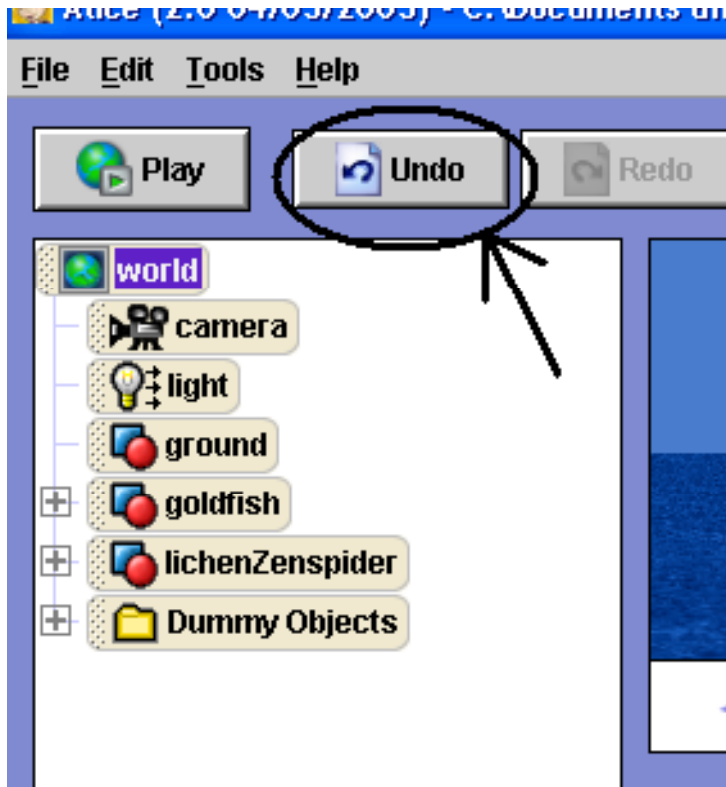


- This button is used to spin your objects around.



- Try spinning your goldfish around so he is directly parallel to the screen.
- Your screen should look like this.
- *Note: If you click on the water by mistake and move it, click the UNDO button to fix the water back.*

# The Undo Button is your Friend!



- Anytime you do something wrong in your Alice world you can click the undo button and Alice will undo the last command.
- Use it as much as you need!



# Positioning your objects

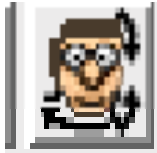


- The next button will turn your character either backwards or forwards. Use it to make your fairy tilt slightly forward.



- This is how your screen should look.

# Positioning your objects

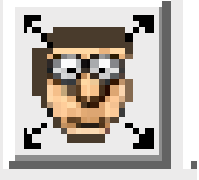


- This next button is a toggle button. It will move your object all over the place.
- Use it to move your fairy so that she is facing the fish

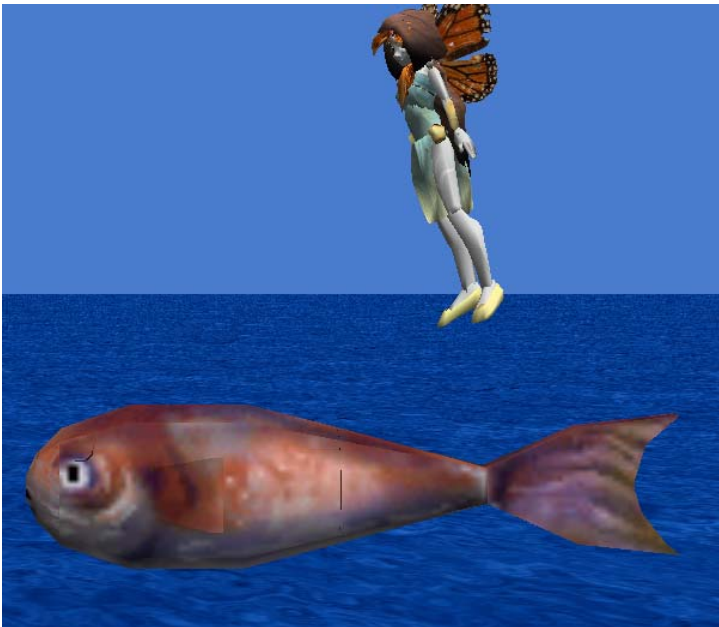


- Your screen should look like this.
- *Note: If you don't like her position, click on "undo" and try to move her again.*

# Positioning your objects



- This next button will make your objects bigger.
- Click on it and use it to make your goldfish bigger than the fairy.



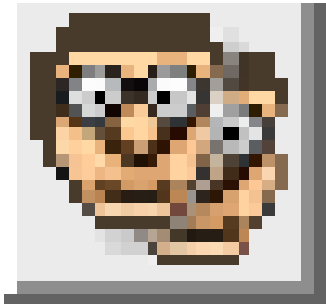
- Your screen should like this.

# Positioning your objects

- Now lets move our fish down so that he is slightly sticking out of the water.
- You may have to push the fish back further and resize him to get this picture.

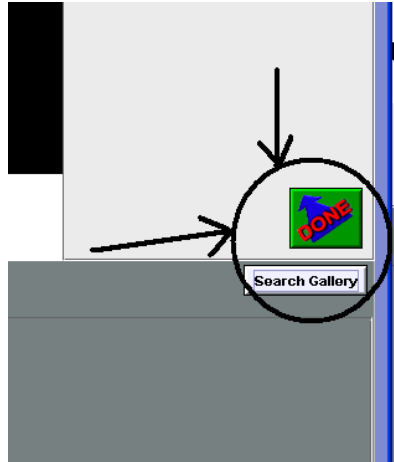


# Positioning your objects

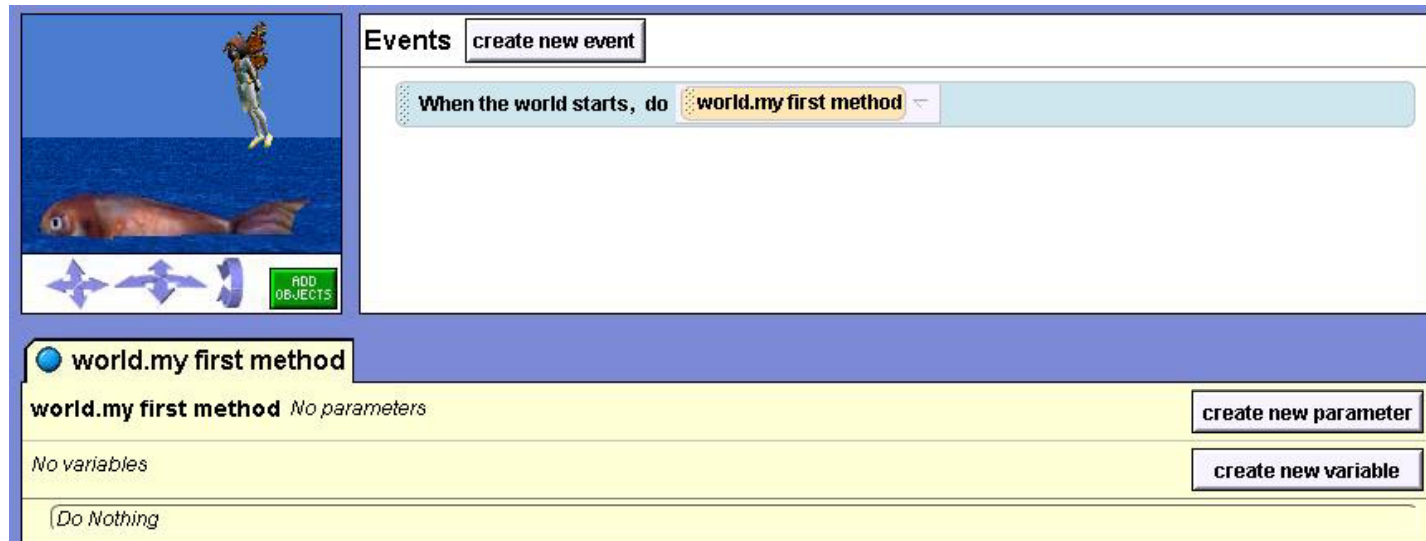


- There is one more button that we will NOT use.
- This button will copy your object, however it causes problems later SO DON'T USE IT!
- It's better to add another object from the gallery if you want two of the same thing.

# Positioning your objects



- Click on the “done” button to leave your objects gallery.
- Now your screen should look like this.

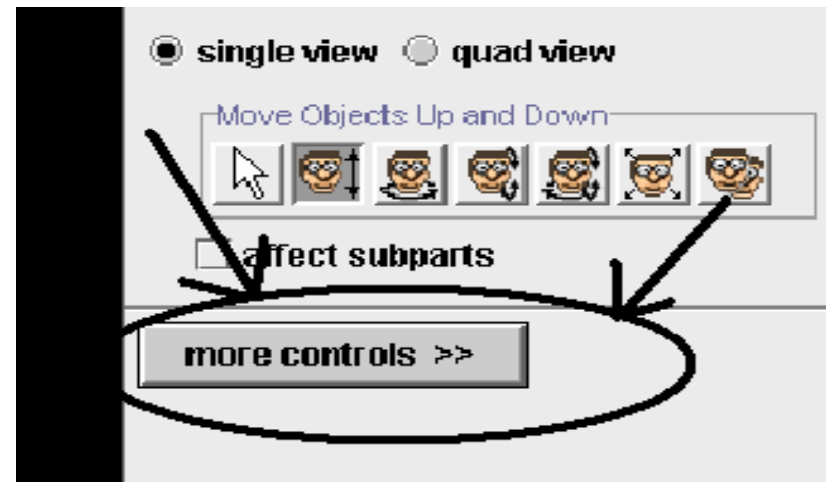




Now that our characters are positioned we can move around our camera!

# The Dummy Camera

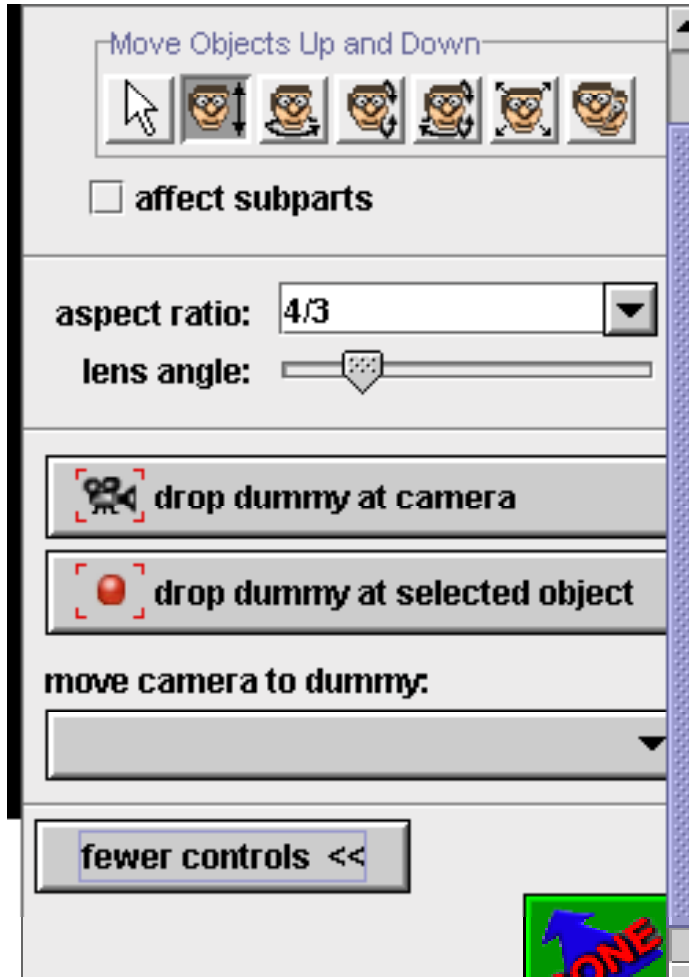
- The Dummy camera is designed to save camera positions as you move your camera around.
- Click on the “add objects” button again and look for a gray button under your object positioning buttons



- Click on the button labeled “more controls”.



# The Dummy Camera



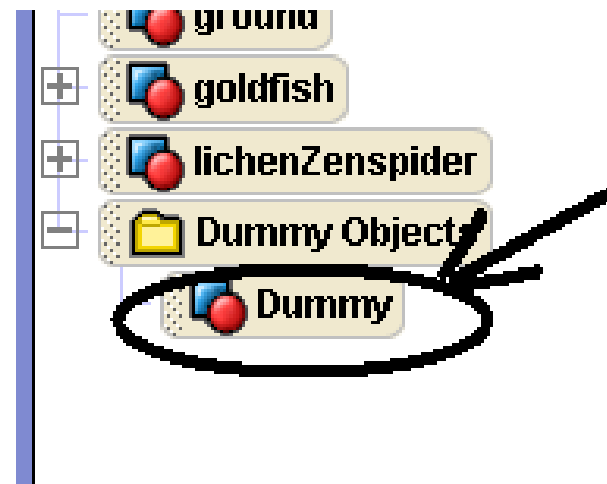
- Once you have clicked on the “more controls” button you will see a drop down menu.
- Click on the “drop dummy at camera” button.
- Just click once!

# The Dummy Camera

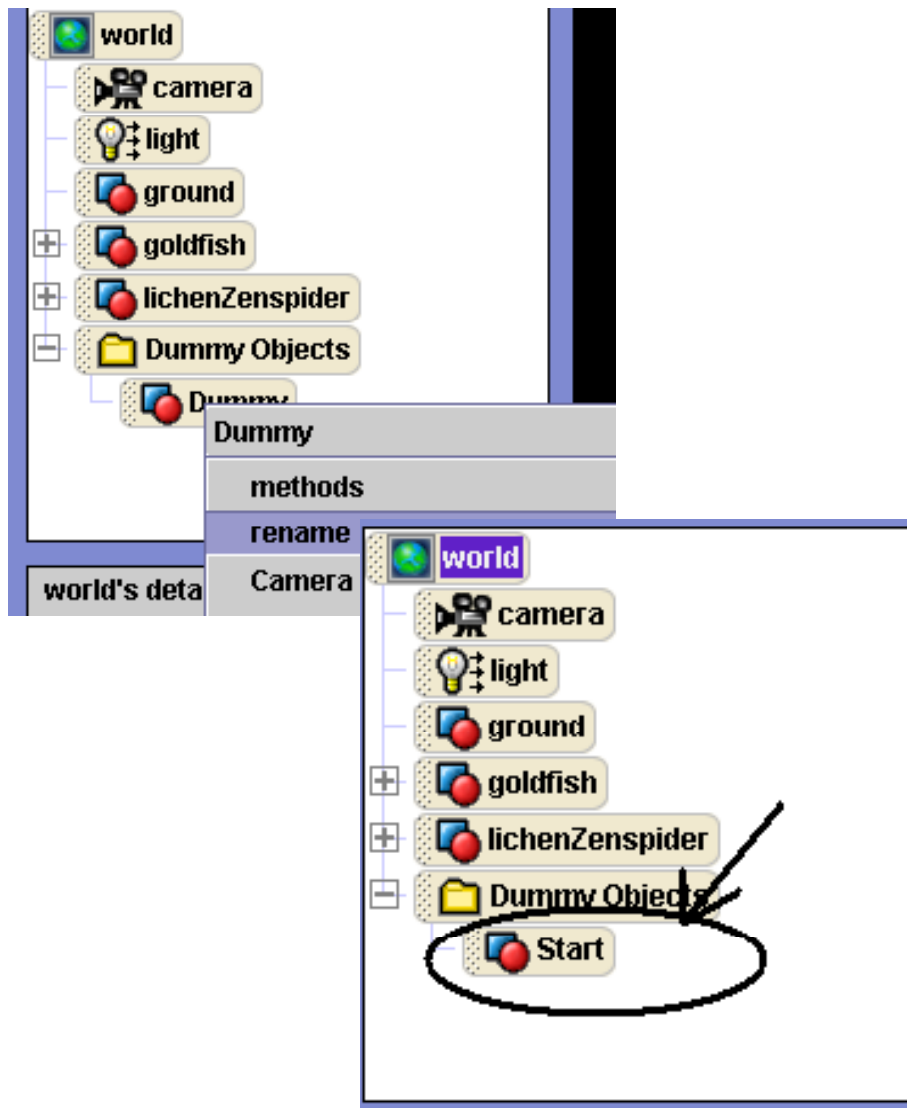


- Once you have dropped the dummy camera. A folder will appear on your object tree called “Dummy Objects.”

- If you click on the plus sign beside the Dummy Objects folder your list of dummy camera positions should appear.
- Right now there should only be one, called “Dummy”.



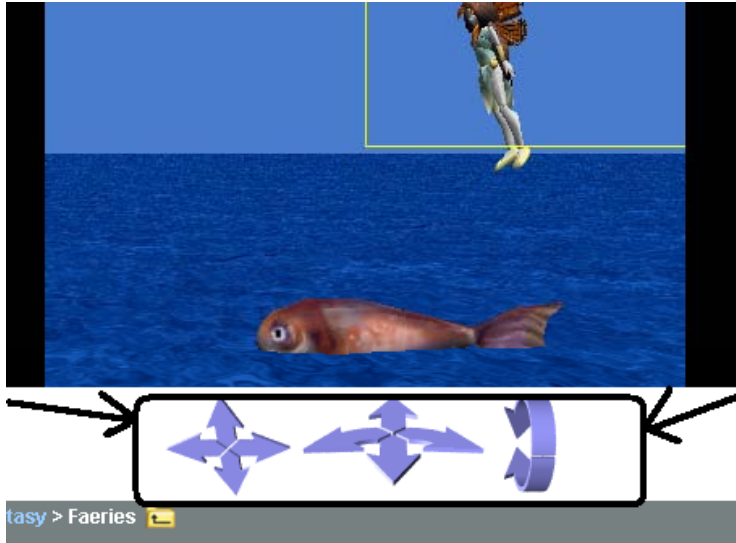
# The Dummy Camera



- You can rename your dummy camera to something that is easier to remember.
- Lets rename ours “Start.”

This will freeze your camera position for you to return to whenever you would like.

# The Dummy Camera

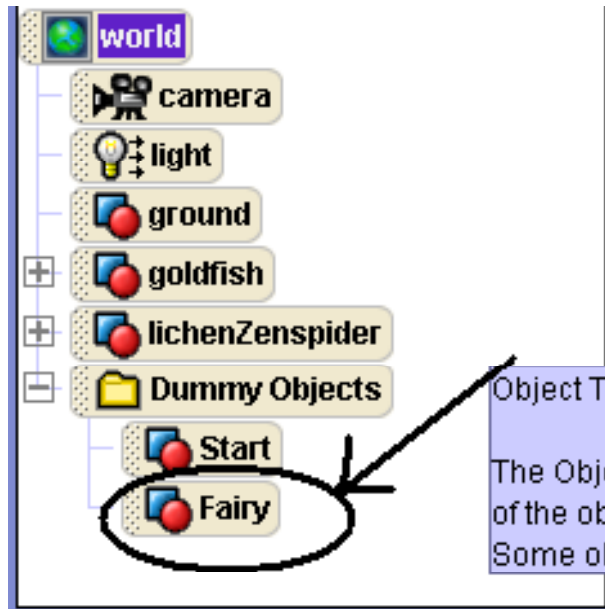


- Lets move the camera so that we can get a close up of the fairy.

- Now we can move our camera to another position using the purple arrows at the bottom of your screen.



# The Dummy Camera

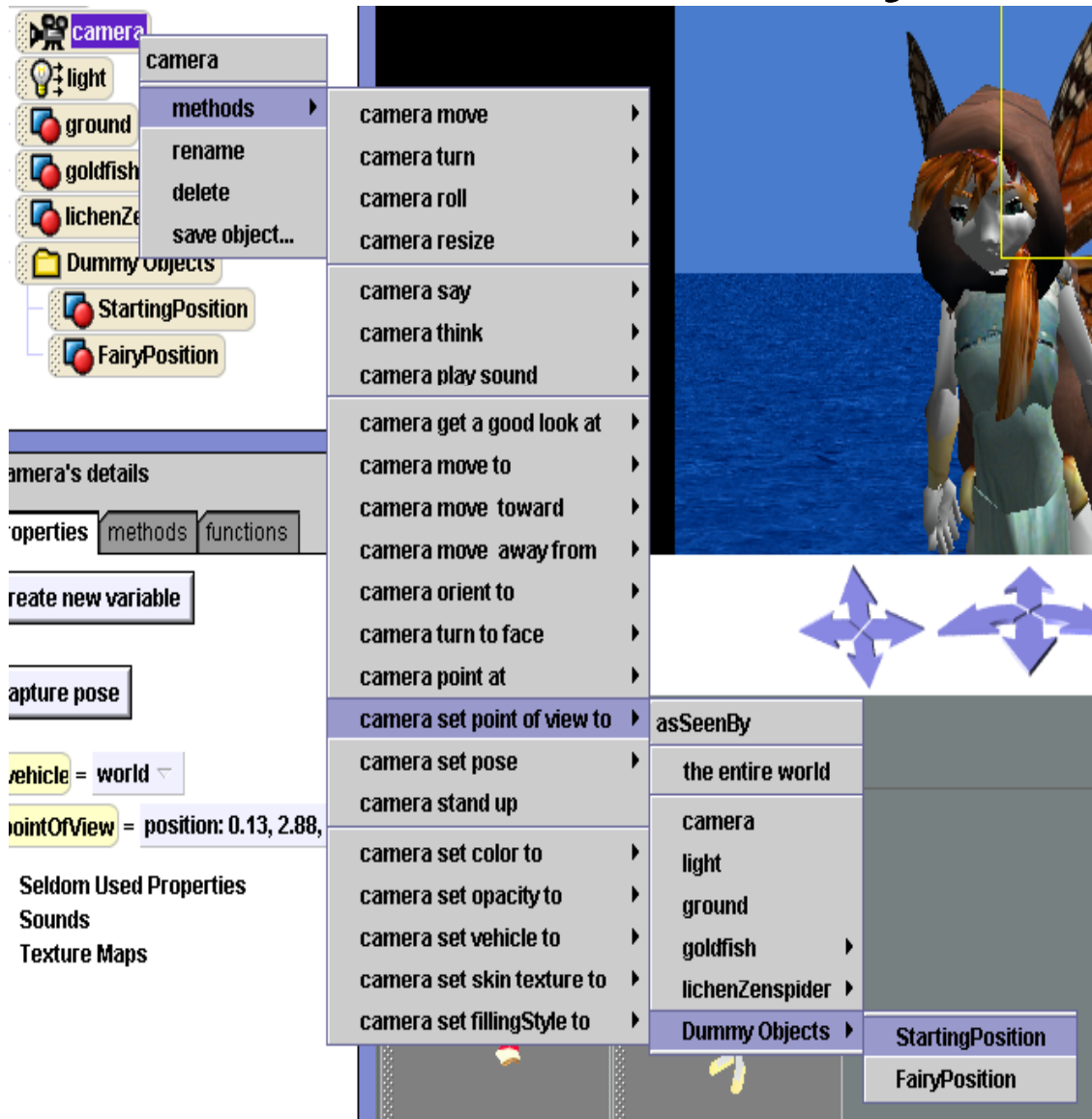


- Now what if we want to go back to our “Start” camera position?
- Right Click on camera in your object tree

- Lets drop another Dummy camera at our Fairy Position and label it “Fairy.”

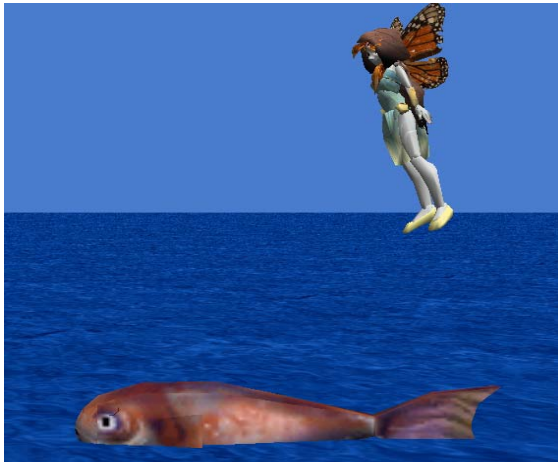


# The Dummy Camera



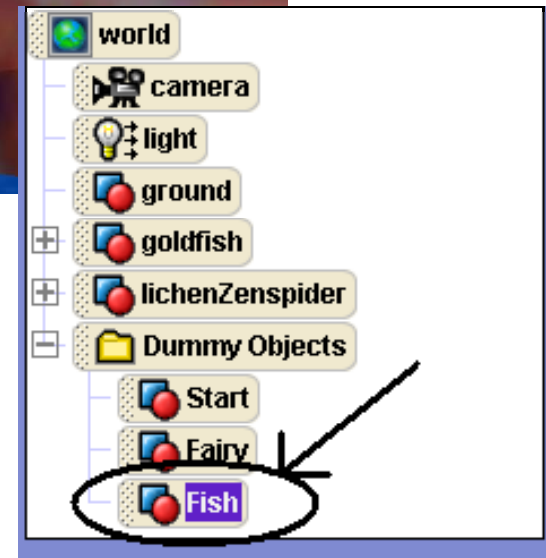
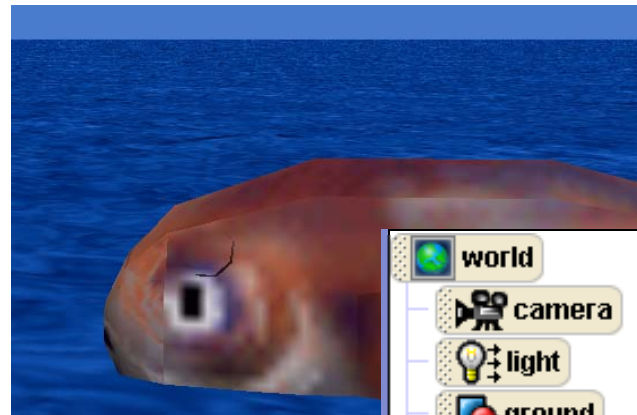
- Right click on Camera and choose “methods”, then “camera set point of view to” then “Dummy Objects”, then “Start”.

# The Dummy Camera

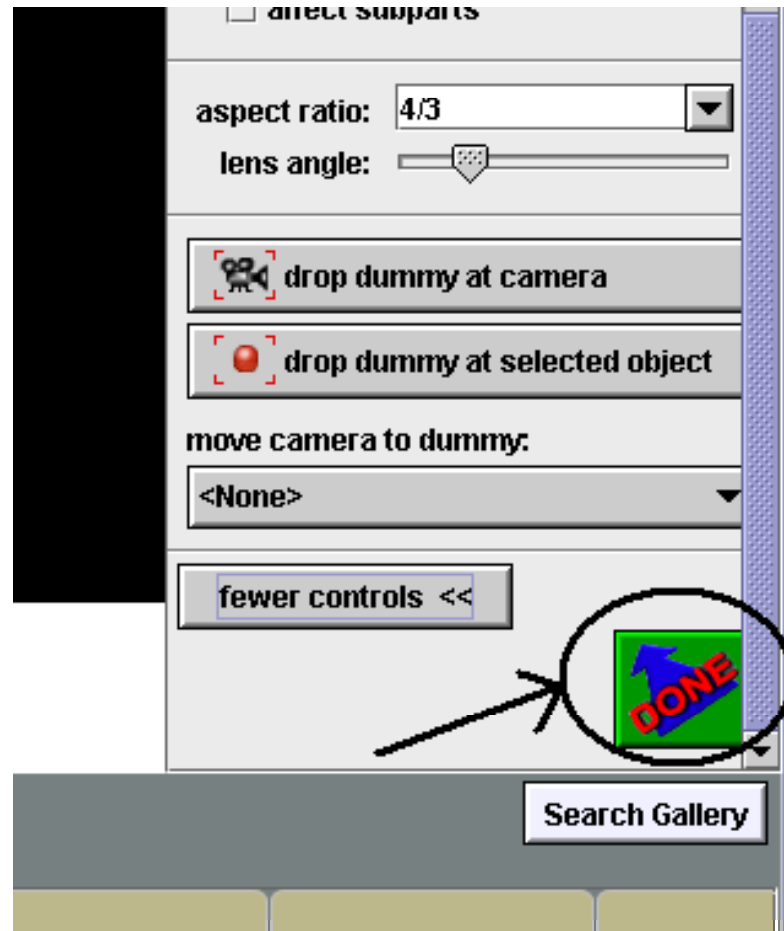


- Now go through these steps again to create a camera position for “Fish.”

- This should set your camera view back to its starting position.



# Dummy Camera Conclusion



- Finally, Set your camera position back to “Start” position.
- Now click on the “Done” button at the bottom right of your screen to exit out of your “add objects” section (you may have to scroll down to see the Done button).

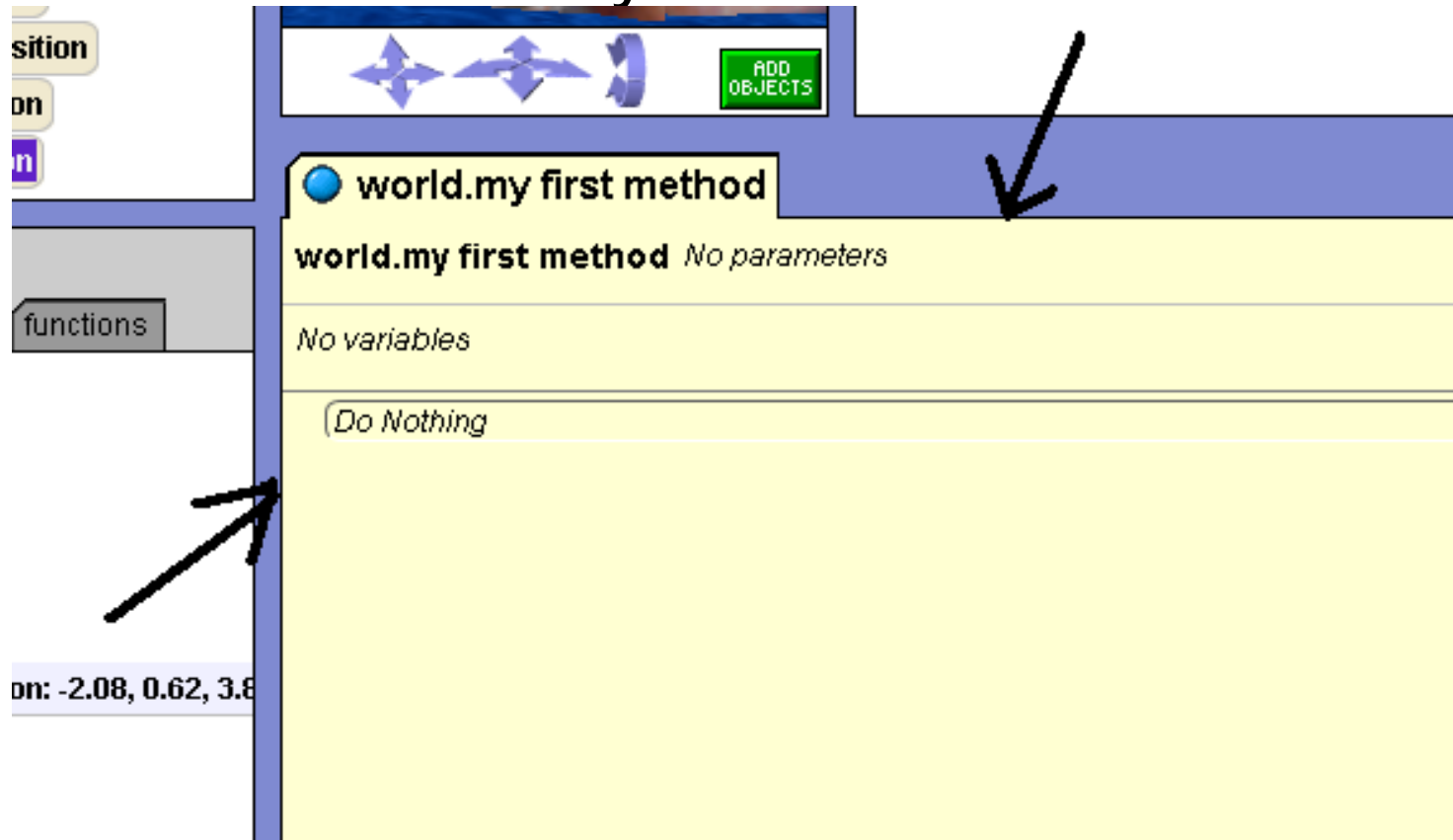




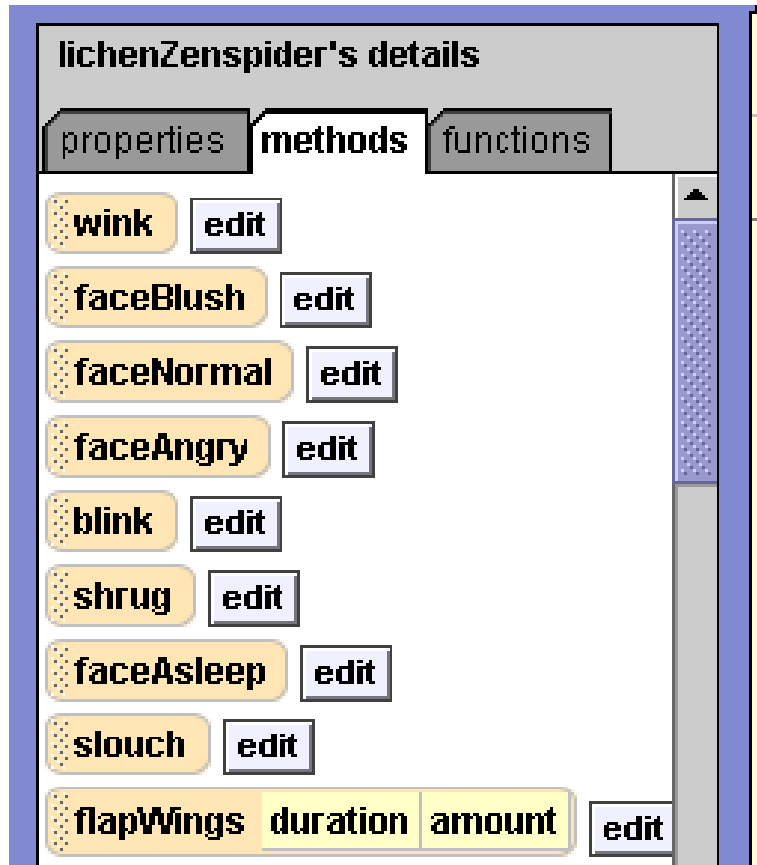
Now that we have done some cool things with the camera, we can animate our characters!

# Writing Methods

- The large tan part in the center of your screen is called your “Method Editor.”

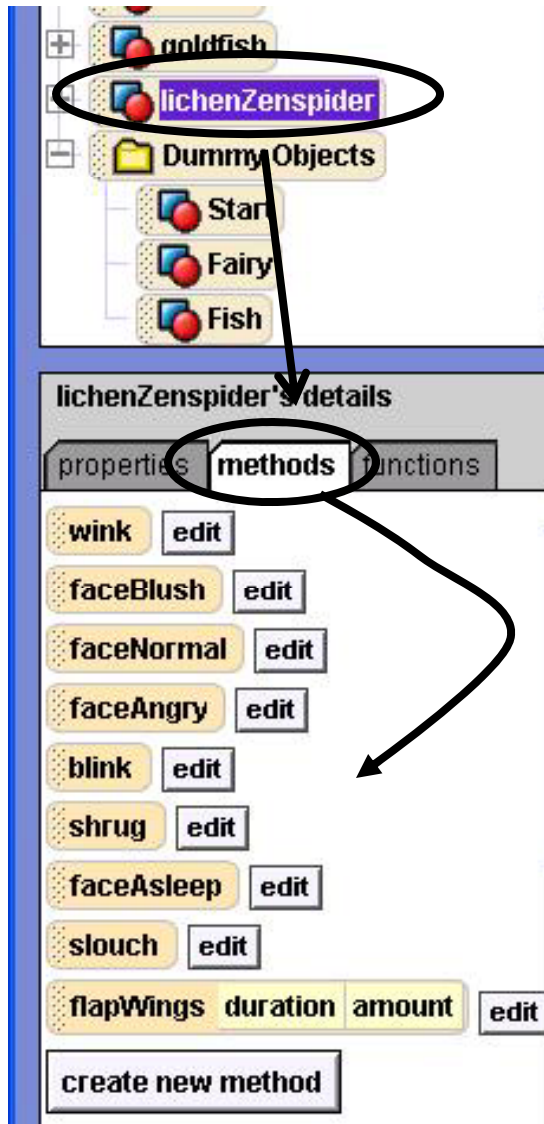


# Methods



- The method editor is for making your characters do something.
- Your characters already know how to do several things.
- These are all of the things that your fairy knows how to do.

# Methods



- You can find these by clicking on your fairy in your object tree and choosing methods.
- Lets try animating our fairy with some of these methods.

# Methods

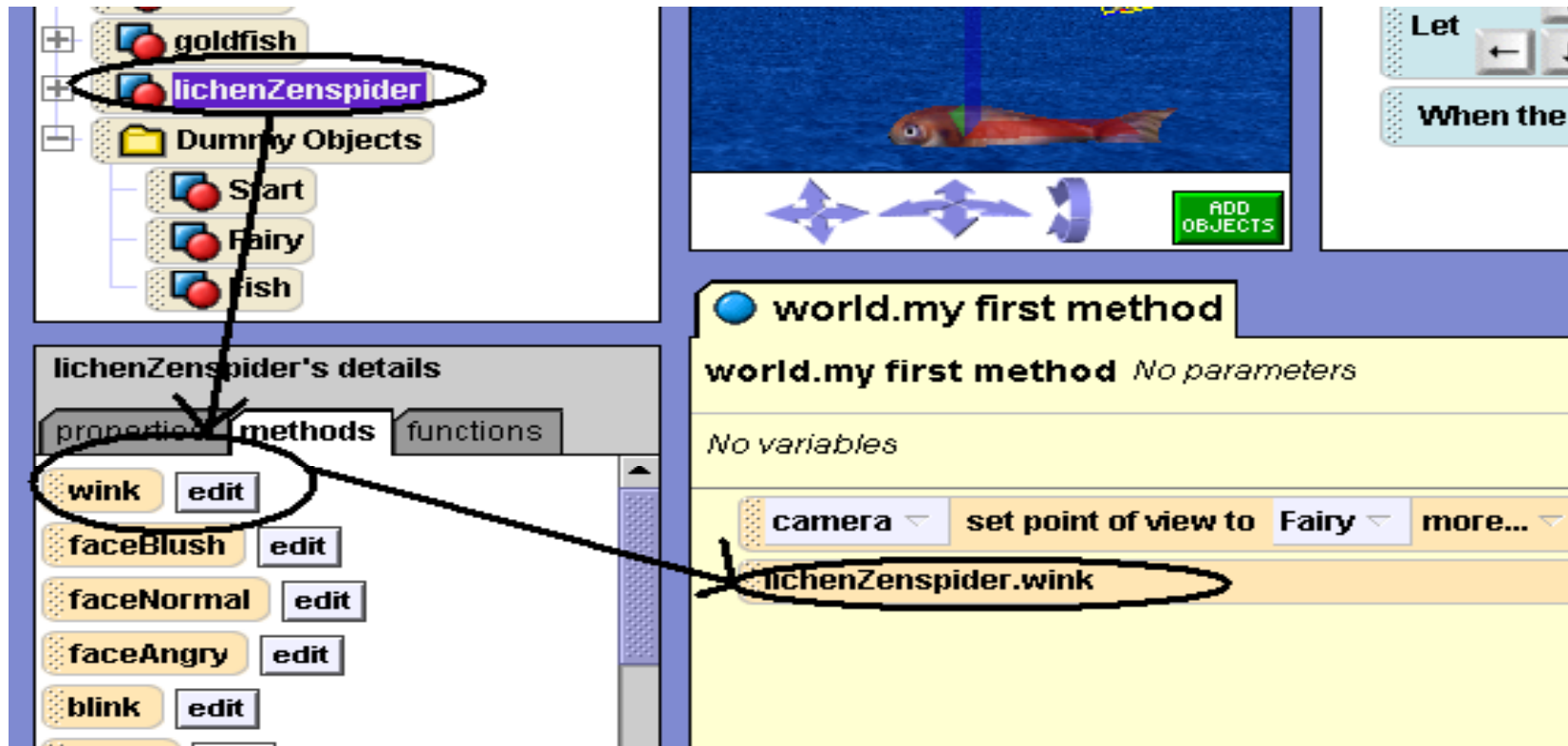
The screenshot shows a game engine interface with several panels:

- Scene Tree:** A list of objects in the world, including 'camera', 'light', 'ground', 'goldfish', 'lichenZenspider', 'Dummy Objects', 'Start', 'Fairy', and 'Fish'. The 'camera' object is circled.
- 3D View:** A 3D rendering of the world showing a fairy flying above a goldfish in a blue environment.
- Events Editor:** A panel for creating and editing events, showing a sequence of actions like 'When Space is typed', 'Let move goldfish', and 'When the world starts, do'.
- Method Editor:** A panel for editing methods. The method 'world.my first method' is selected. It has no parameters and no variables. The method body contains a 'Do Nothing' block. A dropdown menu is open for the 'asSeenBy' property, showing a list of objects: 'the entire world', 'camera', 'light', 'ground', 'goldfish', 'lichenZenspider', 'Dummy Objects', 'Start', 'Fairy', and 'Fish'. The 'Fairy' option is circled.

- This time instead of right clicking, drag your camera view change into your method editor.
- Set your camera view to “Fairy.”

# Methods

- Click on the method called “Wink,” drag it over and drop it in your method editor.

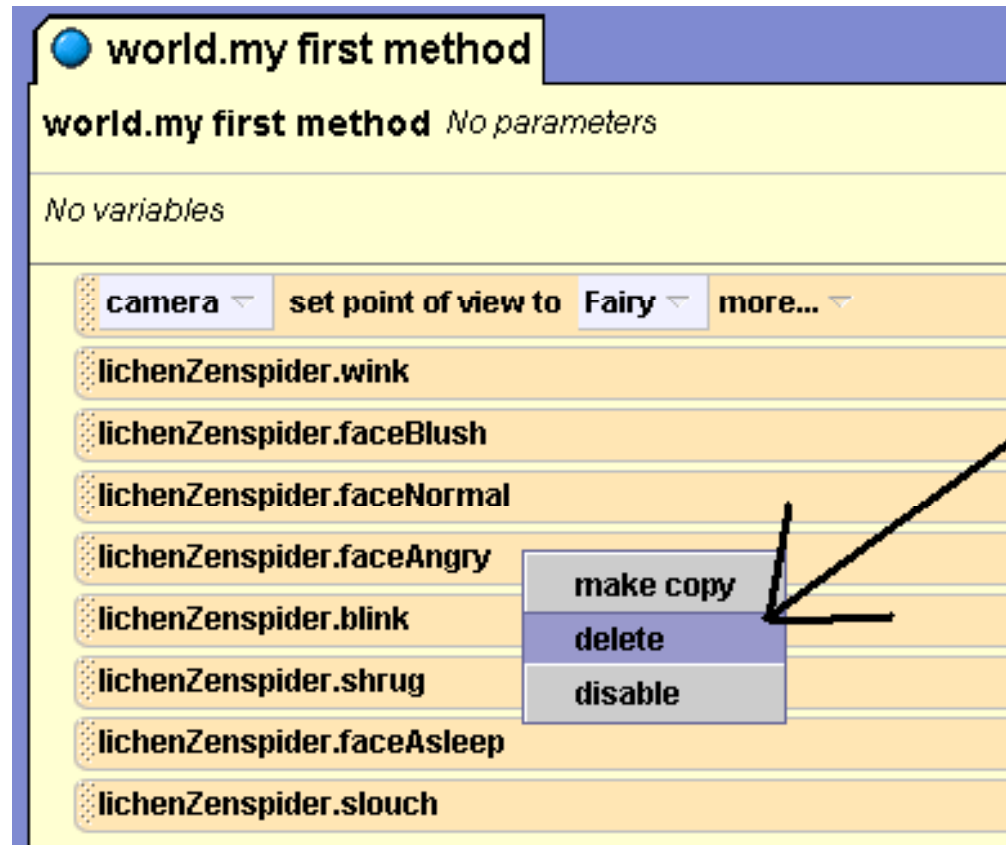


# Methods

- When you press play the fairy should wink with her right eye.
- Try dragging in each of the methods to see what she does.
- Note that the “flap wing” method asks you for a duration and an amount. Type in numbers for both and see what happens.

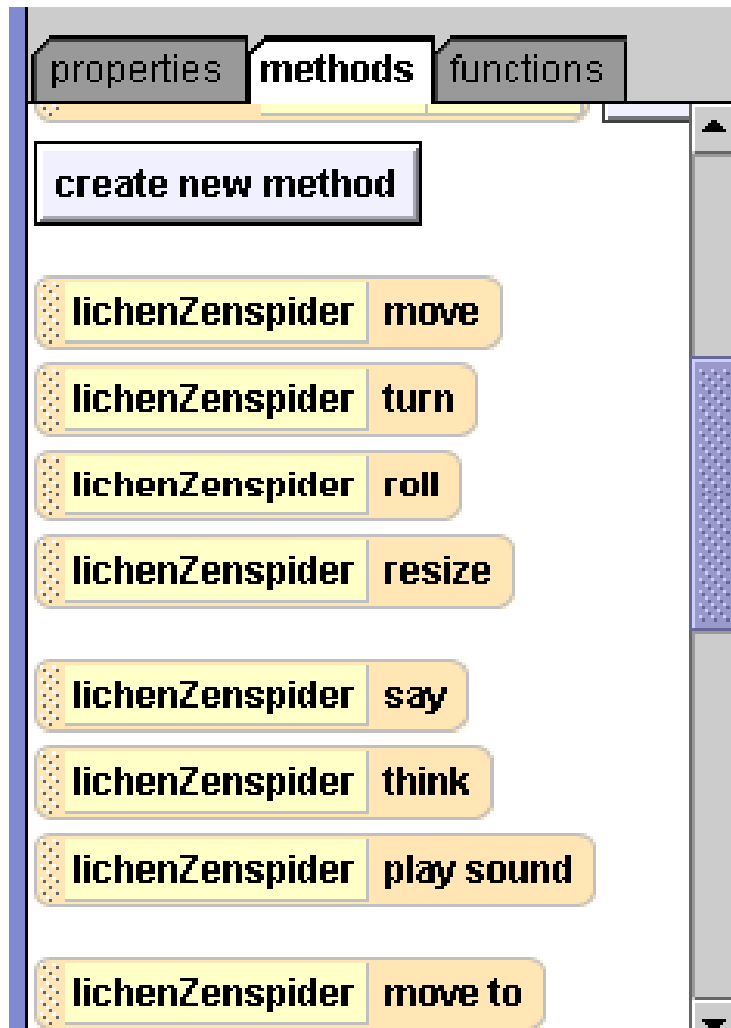
# Methods

- If you don't want any of this code you can right click on the sections that you don't want and choose "delete."



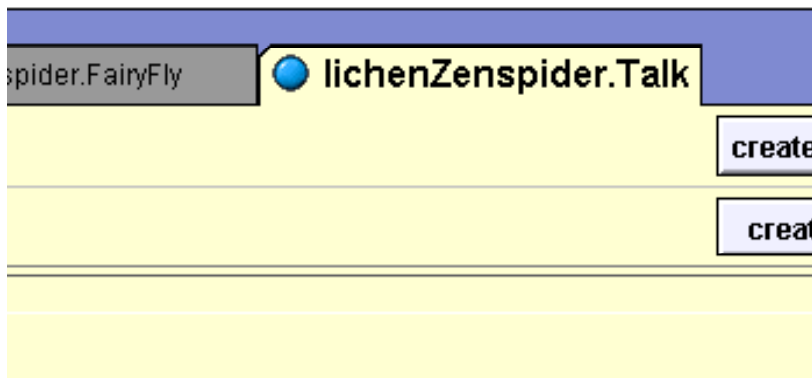
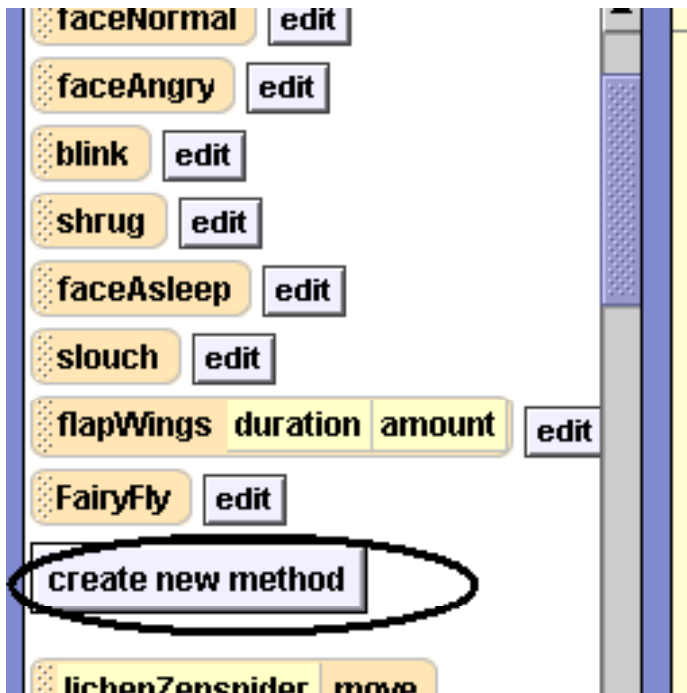


# Methods



- What if we want to teach our fairy to do something that she doesn't know how to do?
- If you look below the methods that you just tried there are more simple methods.
- These are the methods that we will use to teach our fairy to do new things.

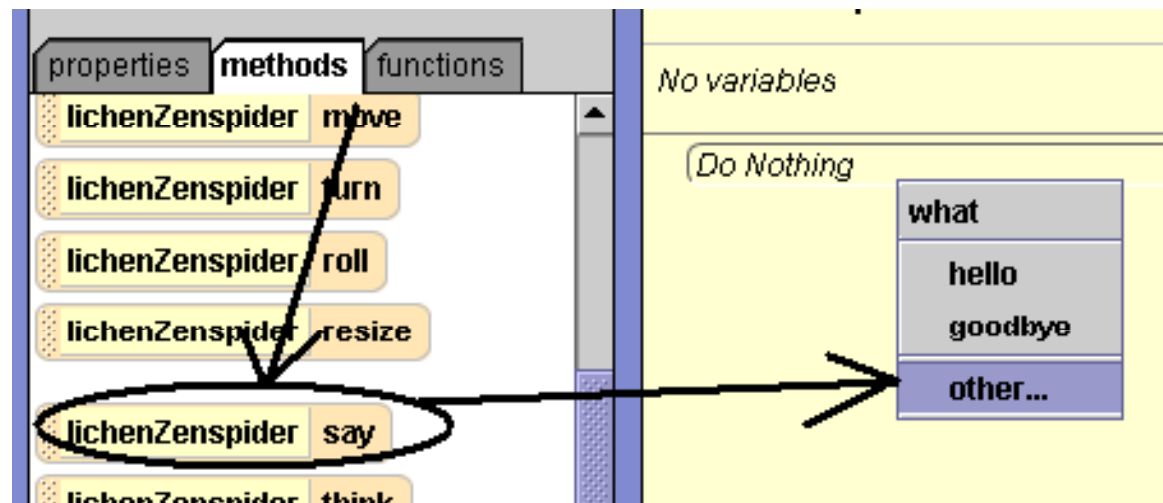
# Talk



- Lets start with something simple
- Go to the methods for your fairy and select the gray button labeled “Create new method.”
- Name it “Talk.”
- This box should appear

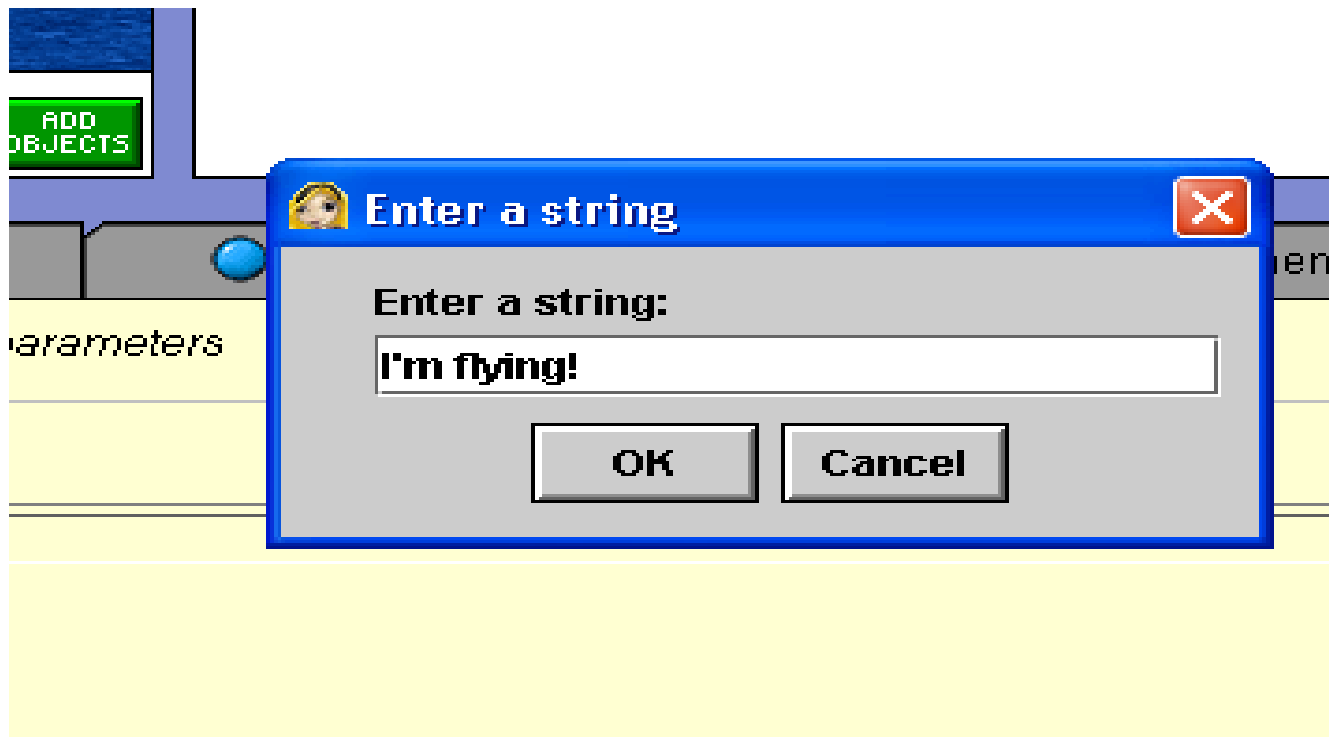
# Methods

- Now look in the simple methods of your fairy and find the “Say” method
- Drag it over and drop it into your method editor.



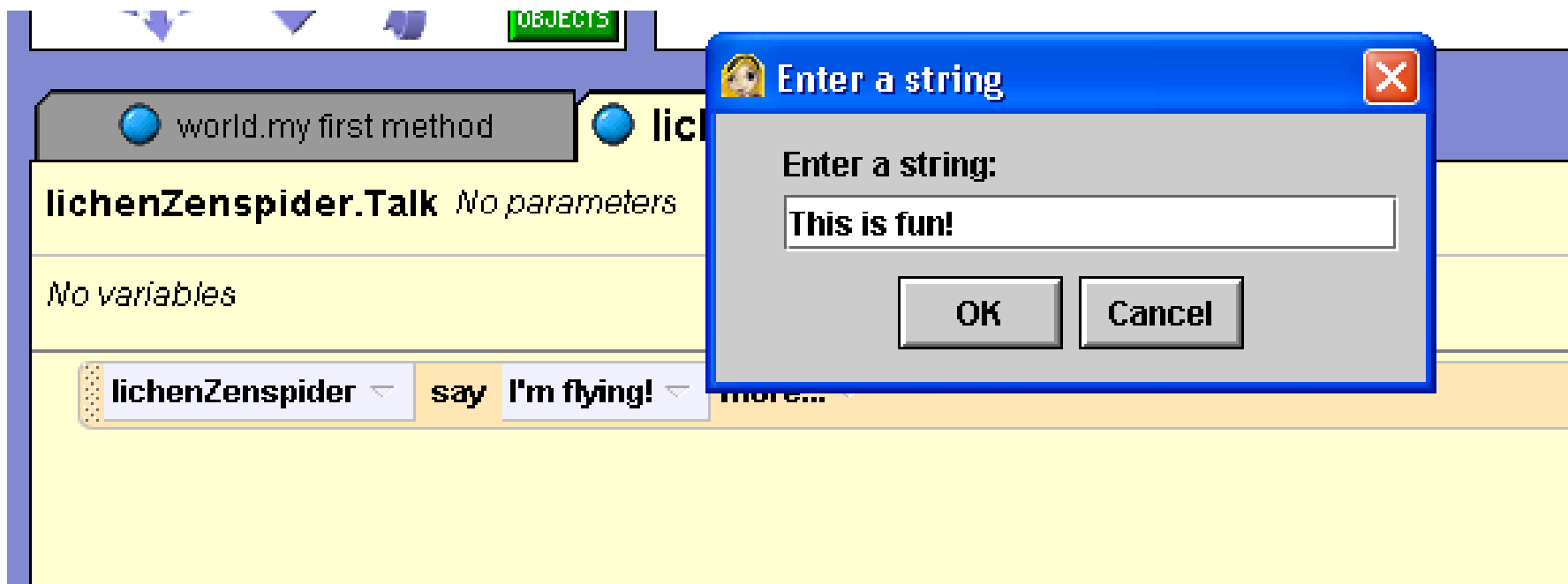
# Methods

- Select “other”
- This should give you a box for you to type something.
- Type “I’m flying!”



# Methods

- Try this again but have her say something else.



# Methods

- Now your method should look like this.
- Now how can we play this method?

The screenshot shows a software interface with a blue header bar. On the left, a grey tab labeled 'world.my first method' is visible. On the right, a white tab labeled 'lichenZenspider.Talk' is active. Below the tabs, the main area is yellow and contains the following text:

**lichenZenspider.Talk** *No parameters*

*No variables*

Below this, there are two rows of method call blocks. Each row consists of four parts: a dropdown menu with 'lichenZenspider', a 'say' button, a text input field, and a 'more...' dropdown menu. The first row has the text 'I'm flying!' in the input field. The second row has the text 'This is fun!' in the input field.

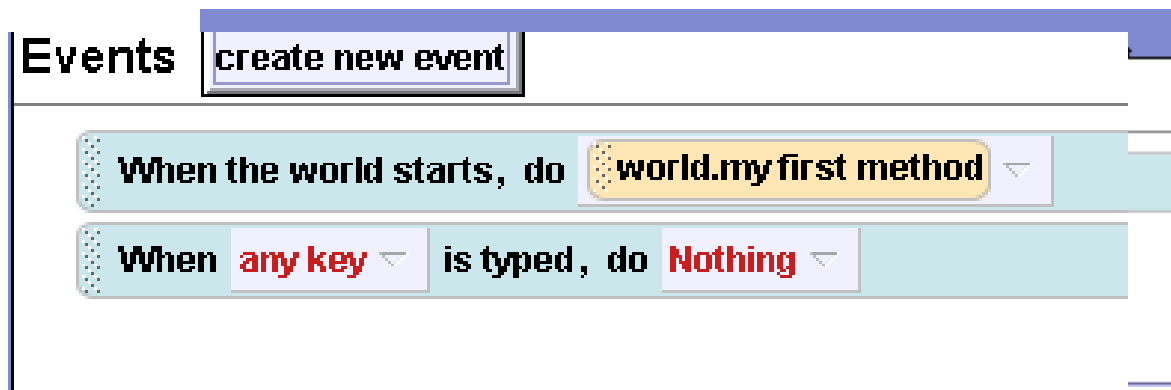
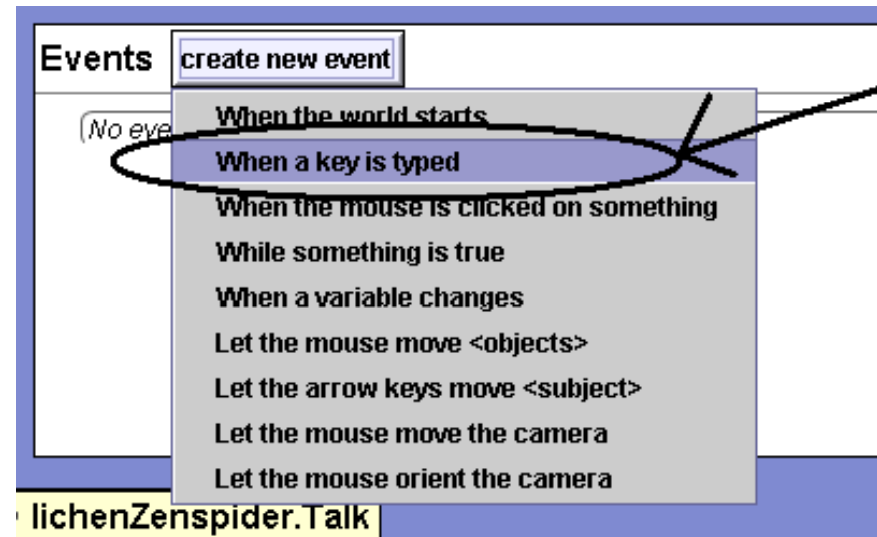
# Events



- We want the fairy to say “I’m flying” and the other things that you typed whenever we press the space bar.
  - To do this, we will add an event and call the Talk method we wrote.
- An event is a way to call the methods you create when you play your world
  - The event editor is in the top right hand corner of your screen.

# Events

- Click on “create new event”
- You will see the list of events in the events editor. Select “when a key is typed”
- After clicking it you should get this.





# Events

- Change the red boxes that say “any key” to the “space bar” and the method “Talk.”  
Use the white arrows

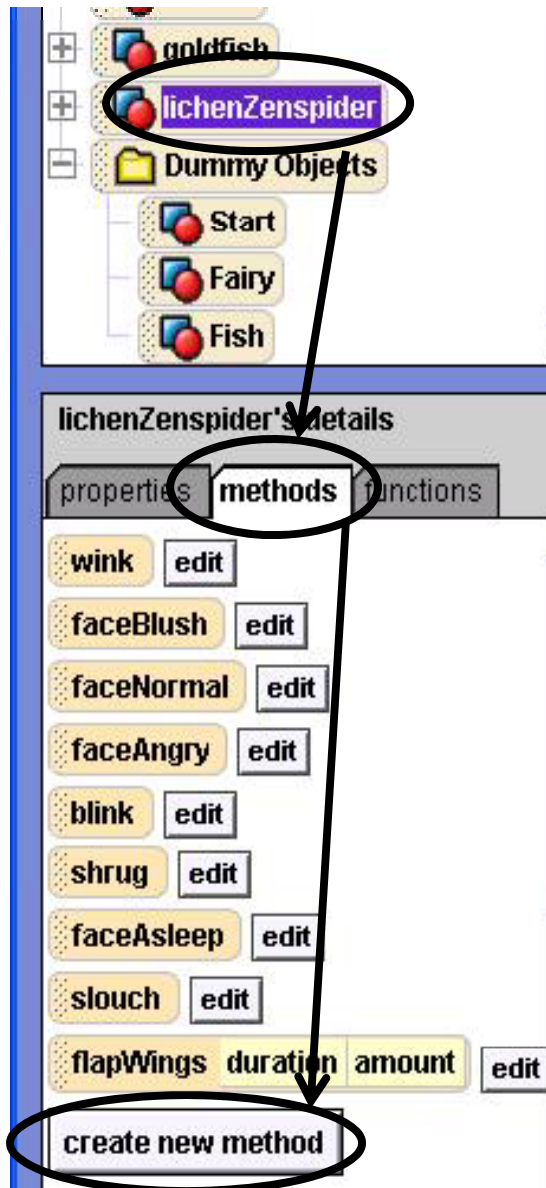
The screenshot shows the Scratch Events palette. At the top, there is a header "Events" and a button "create new event". Below the header, there are two event blocks. The first block is "When the world starts, do" followed by the method "world.my first method". The second block is "When Space is typed, do" followed by the method "lichenZenspider.Talk". Two white arrows originate from the text "Use the white arrows" in the list above. One arrow points to the "Space" dropdown menu in the second block, and the other arrow points to the "Talk" dropdown menu in the second block.

# Events

- Try playing your world
- Each time you press the space bar, your fairy should say “I’m flying”, then “This is fun!”

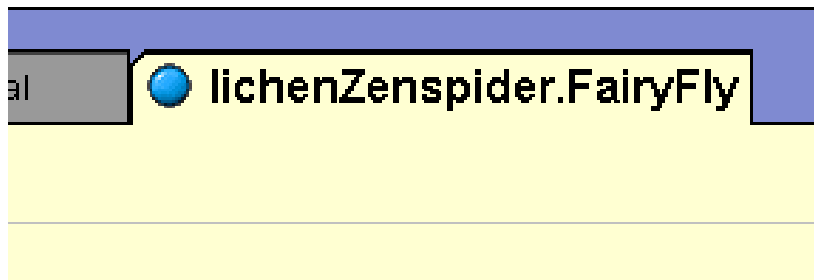
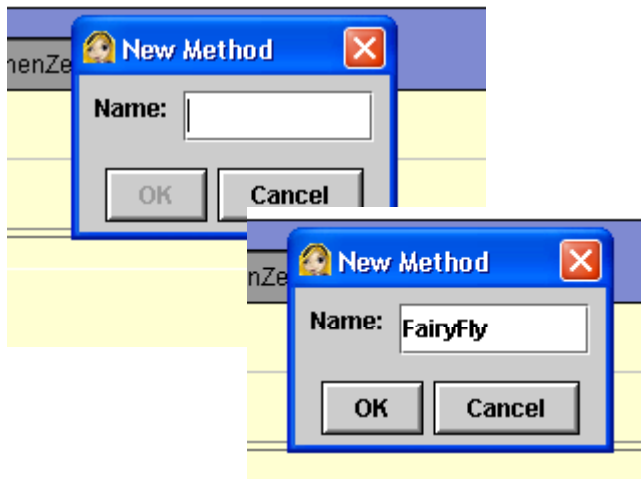


# Methods



- Now, lets have the fairy fly around the fish.
- Click on your fairy in your object tree and go to methods.
- Select the gray button that's labeled "create new method."

# Methods



- After clicking on it a gray box will appear that asks you to name your new method.
- Let's name ours "FairyFly".
- Now a new method will appear on your method editor.

# FairyFly

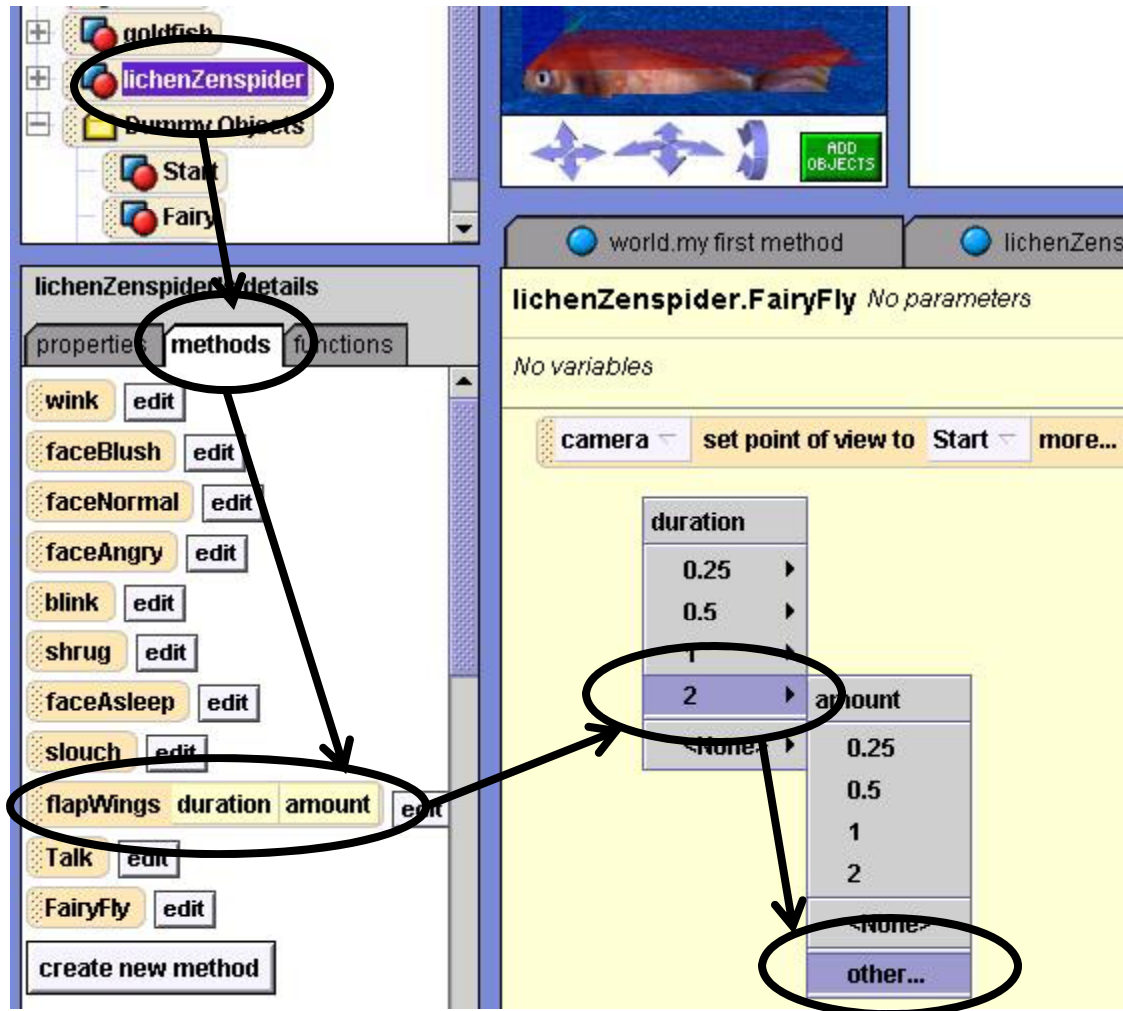
- First we will set our camera to “Start” position.”

The screenshot displays a game engine interface with several panels:

- World Hierarchy:** A tree view on the left showing the scene's structure. The **camera** object is circled in black.
- 3D Viewport:** A central window showing a 3D scene with a character, a red fish, and a ground plane. Navigation arrows and an **ADD OBJECTS** button are visible.
- Events:** A panel on the right showing a single event: "When the world starts, do world.my first method".
- Method Editor:** A panel at the bottom showing the configuration for the **lichenZenspider.FairyFly** method. It includes tabs for **properties**, **methods**, and **functions**. The **methods** tab is active, and the **set point of view to** method is selected and circled in black. A dropdown menu for this method is open, showing a list of objects: **the entire world**, **camera**, **light**, **ground**, **goldfish**, **lichenZenspider**, and **Dummy Objects**. The **Dummy Objects** category is expanded, and the **Start** object is selected and circled in black.

# Methods

- Now lets have our fairy flap her wings.
- Click on lichenZenspider in the object tree and then “methods” tab.
- Drag in the flapWings method.
- Have her flap 6 times for 2 seconds.
- If the number you want is not there, select “other” and then type it in.



Resulting code is:



# Methods



- Now before we play this we need to change our events editor.
- The events editor is at the top right of your screen.

- We need to change the “when the world starts” event to our “FairyFly” method. So that we can test it out.



# Methods

- Now try playing your method. Your camera should move back to the starting position and your fairy should flap her wings.

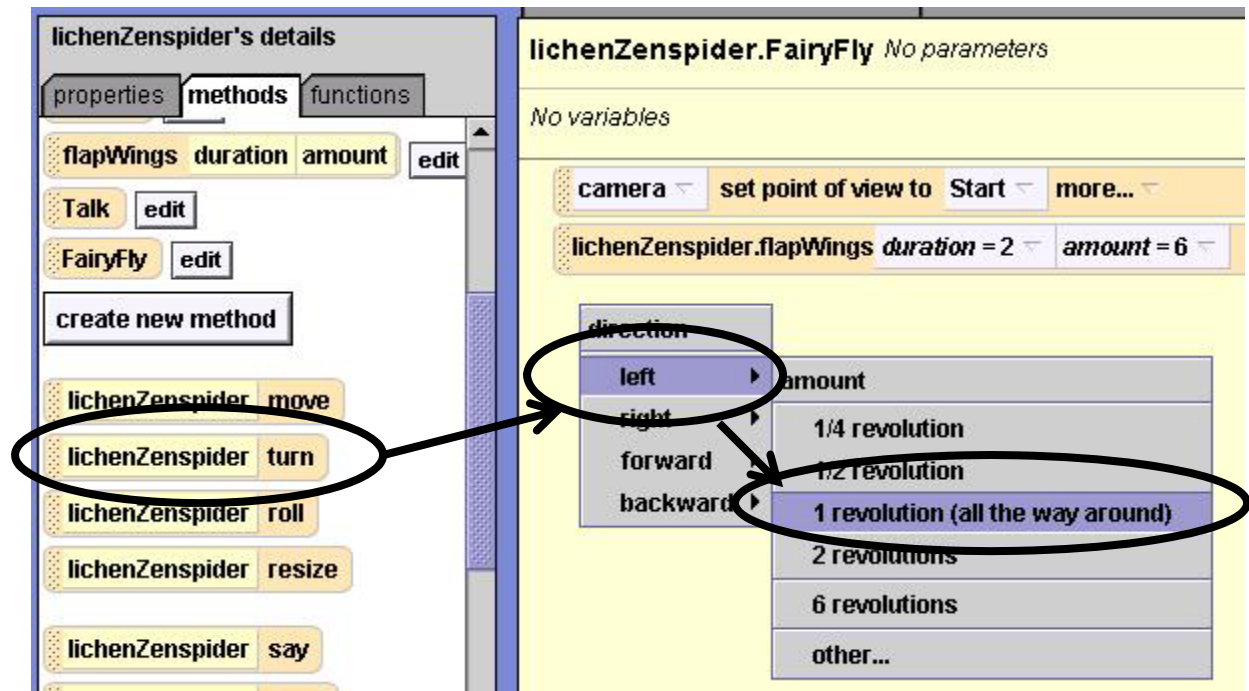
When the world starts, do `lichenZenspider.FairyFly` ▾





# Methods

- Now we need to have our fairy fly around the fish instead of flying in place.
- Go back to your simple methods for your fairy and select “Turn” and drag it into your code.
- Have your fairy turn left 1 revolution.



# Methods

- Your code should now look like this.

**lichenZenspider.FairyFly** *No parameters*

*No variables*

camera ▾ set point of view to Start ▾ more... ▾

lichenZenspider.flapWings *duration = 2* ▾ *amount = 6* ▾

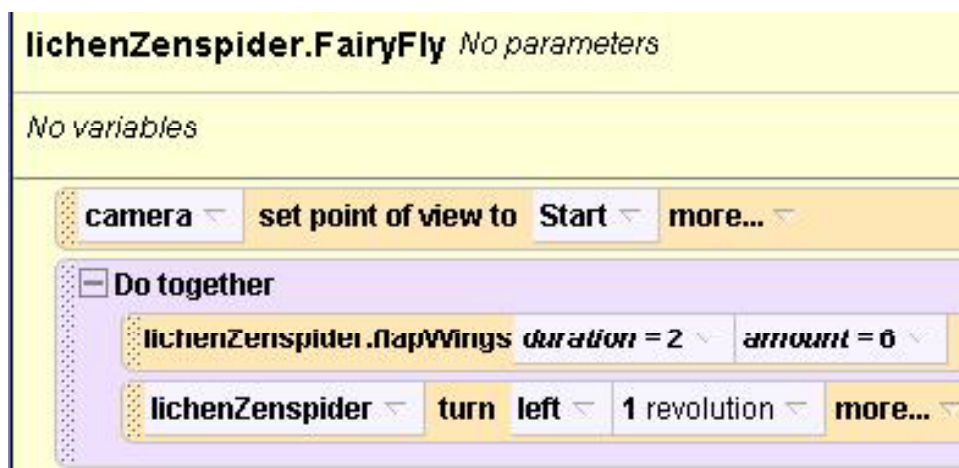
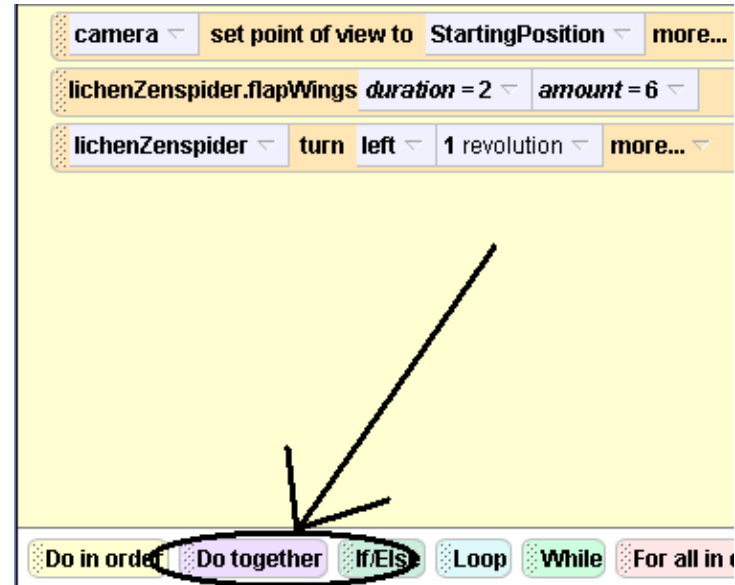
lichenZenspider ▾ turn left ▾ 1 revolution ▾ more... ▾

# Methods

- Now try playing this.
- Your Fairy should flap her wings and then spin in place.
- Now we know that we need to fix two things.
  - We need her to flap while she turns
  - And we need her to turn around the fish instead of turning in place.

# Methods

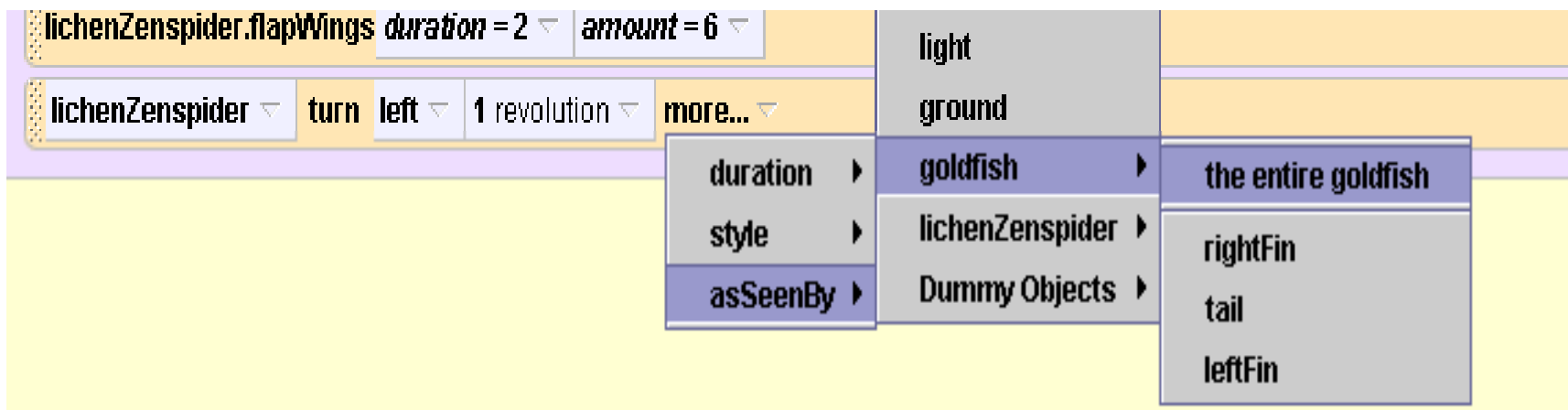
- First lets have her flap and turn at the same time.
- Look at the bottom of your screen for a purple button labeled “Do Together.”



- Drag this up and drop it under your code. Then place your “flap wings” code and your “turn” code inside it.
- Try playing it

# Methods

- In order to make the fairy fly around the fish we need to select the “more” button beside the “Turn” code.
- After clicking on the arrow a pop up menu will appear.
- Select “As seen by”, “goldfish”, “the entire goldfish”.



# Methods

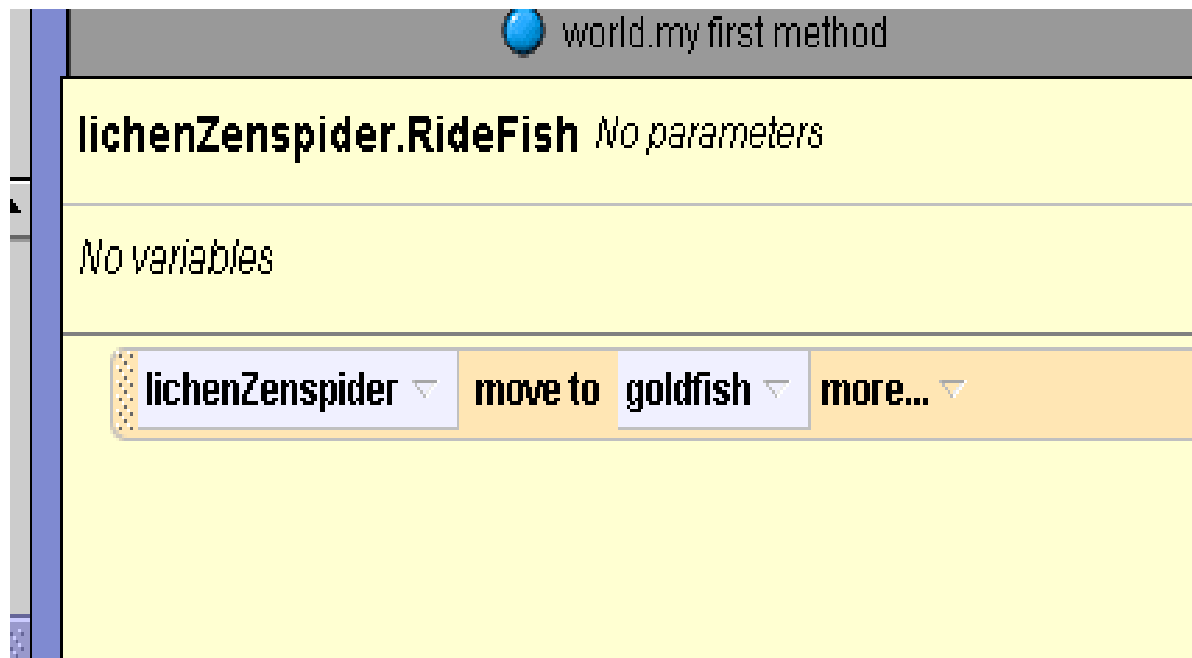
- This should make your fairy turn around in a circle from the point of view of the fish.
- This means it will rotate around the fish.
- Press play to try it.

# Methods

- Lets write one more method
- Lets make the fairy ride the fish after she has flown around him.

# RideFish

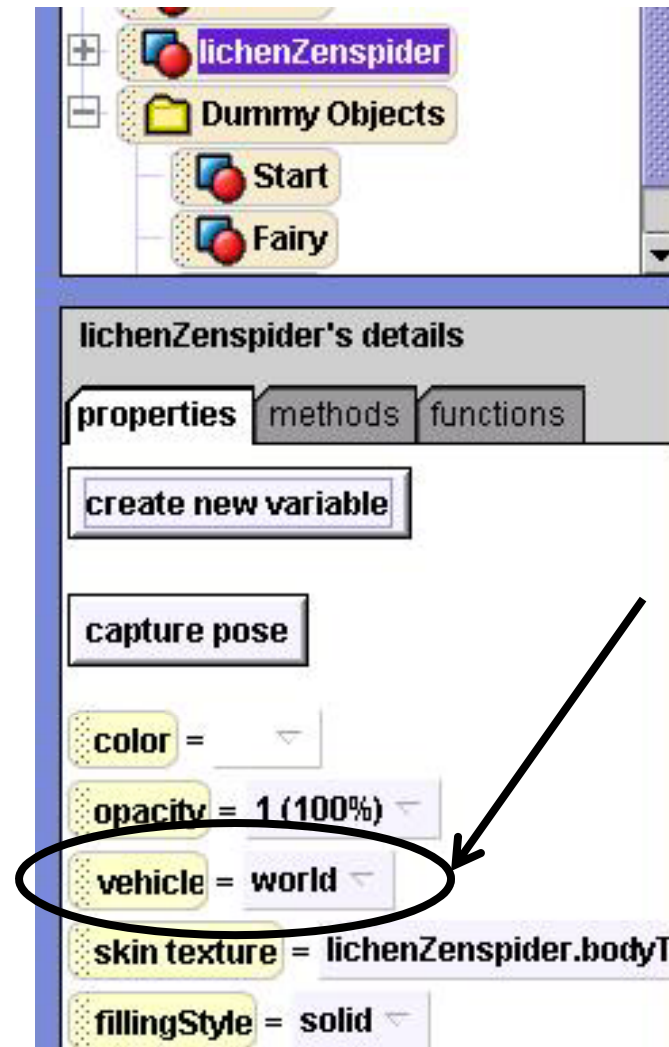
- First you will want to create a new method labeled “RideFish.”
- Click on “lichenZenspider/ “methods”tab/then “create new method”/ and name it “RideFish.”
- Inside your method editor drag the “move to” command and select goldfish/ the entire goldfish.
- Your code should look like this.





# Glue Fairy to Fish

- Now we need a way to glue the fairy to the goldfish so that when the goldfish moves, the fairy will move with it.
- This is called the vehicle property.
- You can find it under the properties tab of your fairy.



# Methods

**lichenZenspider.RideFish** *No parameters*

*No variables*

lichenZenspider ▾ move to goldfish ▾ more... ▾

lichenZenspider ▾ set vehicle to goldfish ▾ more... ▾

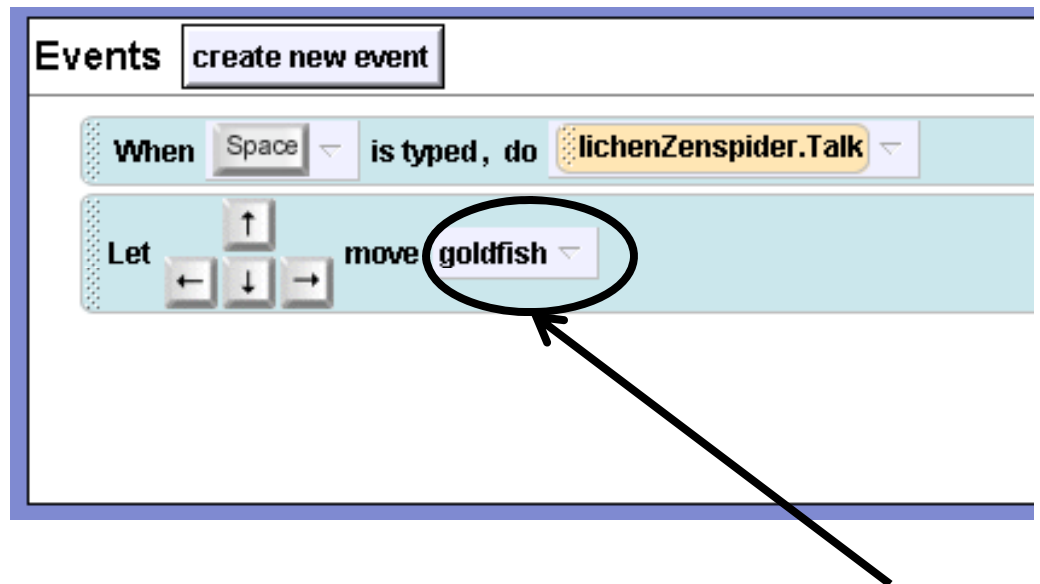
- Drag the vehicle property over to your method editor and select goldfish/entire goldfish.
- This should set your goldfish as a vehicle to your fairy.
- When the fish moves the fairy should move with it.

# Events



- Lets make an event for this method.
- Go to your events editor and select “let the arrow keys move subject.”

- Right now it is set as a default to the camera.
- Lets change this so it is set to the goldfish/the entire goldfish.



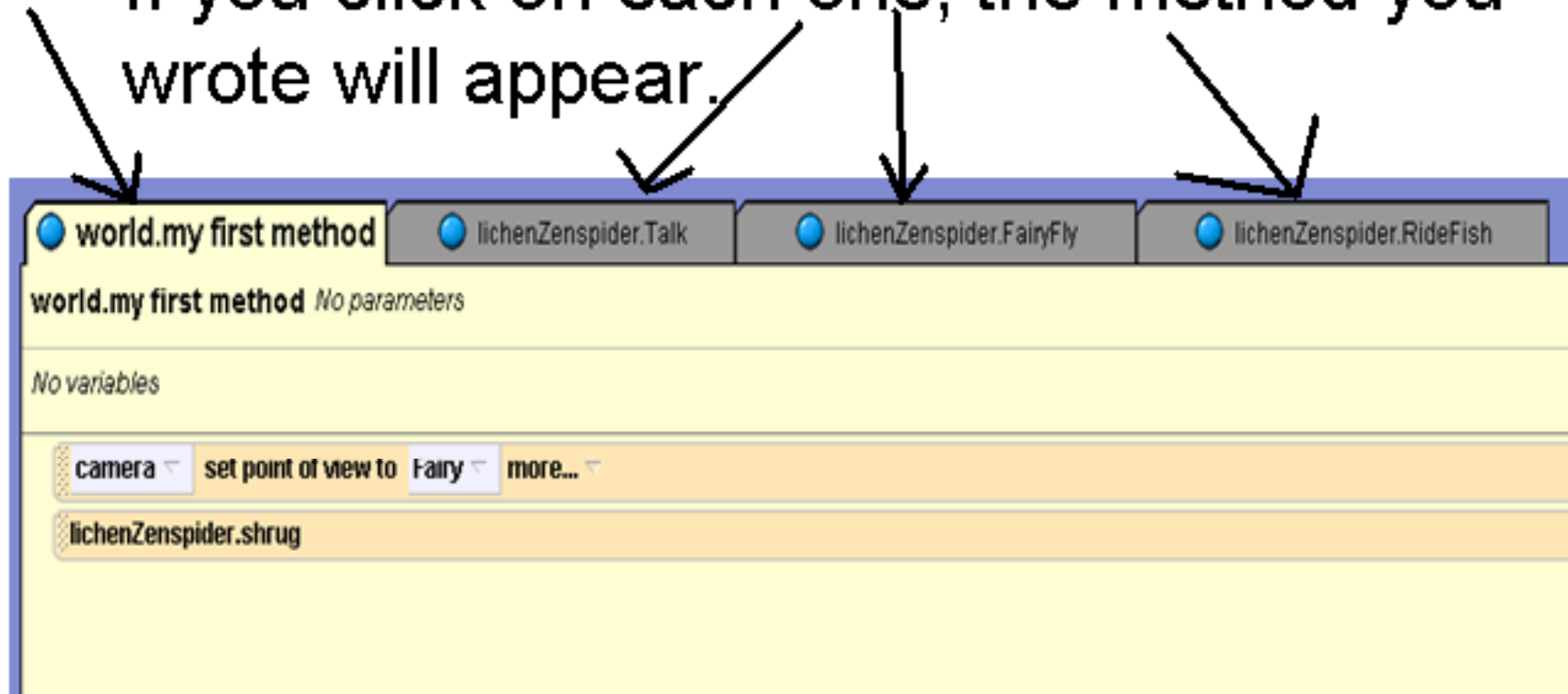
# Events

- Now we can put it all together.
- If you noticed at the beginning of your method editor there was a method that said “World.myfirstmethod.”
- This method is for telling our story.
- You may have more code in there than I do, that is fine.

The screenshot shows a method editor interface with a blue header bar. The header bar contains four tabs, each with a blue circular icon and text: "world.my first method", "lichenZenspider.Talk", "lichenZenspider.FairyFly", and "lichenZenspider.RideFish". The "world.my first method" tab is selected and highlighted in yellow. Below the header bar, the method name "world.my first method" is displayed in bold black text, followed by the text "No parameters" in a smaller, italicized font. Below this, the text "No variables" is displayed in a smaller, italicized font. The main area of the editor is a light yellow background. It contains two lines of code in a light orange box. The first line is "camera set point of view to Fairy more..." and the second line is "lichenZenspider.shrug".

# Events

- Note that there are a lot of tabs here.
- If you click on each one, the method you wrote will appear.

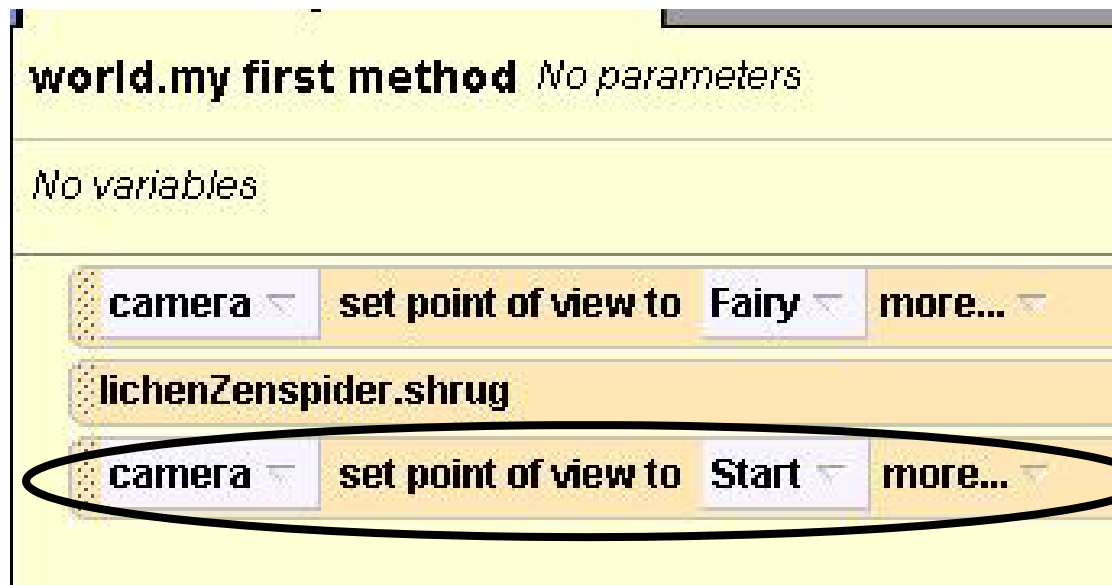


# Events

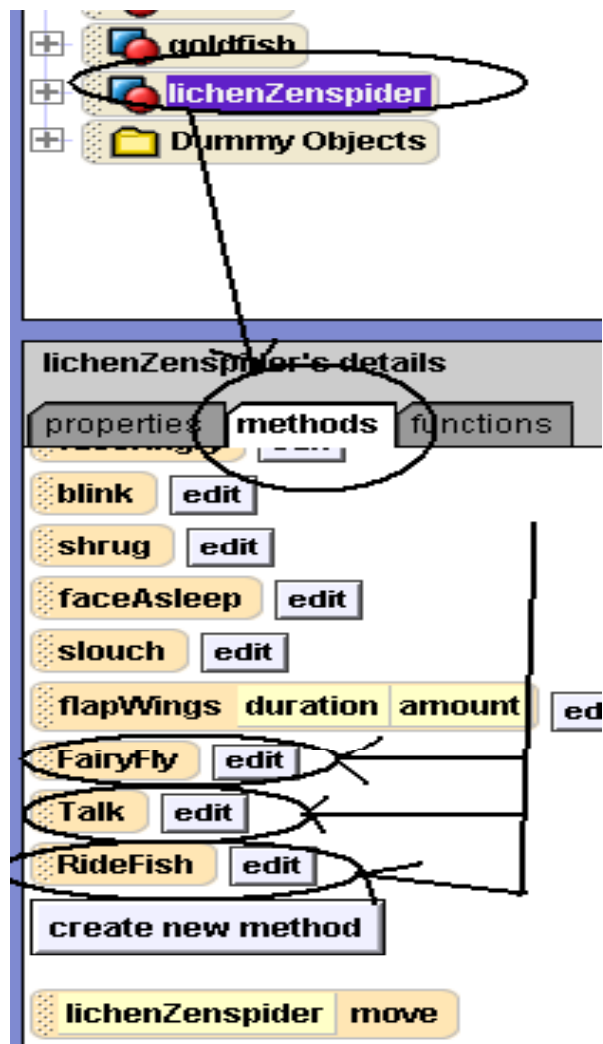
- First click on the tab that says “World.MyFirstMethod”
- It is the tab that we put all of our original code in.

# Events

- First we need to set our camera position in “world.myfirstmethod” to “Start” right after the code you have.
- Drag in “camera set point of view to” and set to “Dummy objects”, then “Start”



# Events and Methods



- If you look under the methods of your fairy you will now see the methods that you wrote.
- We can put these together in world.myfirstmethod to tell a story.



# Events and Methods

**world.my first method** *No parameters*

*No variables*

- camera ▾ set point of view to Fairy ▾ more... ▾
- lichenZenspider.shrug
- camera ▾ set point of view to Start ▾ more... ▾
- lichenZenspider.FairyFly
- lichenZenspider.RideFish

- Lets have the fairy fly around the fish and then get on the fish and ride
- When can do this by dragging “FairyFly” and “Ridefish” into World.myfirstmethod.

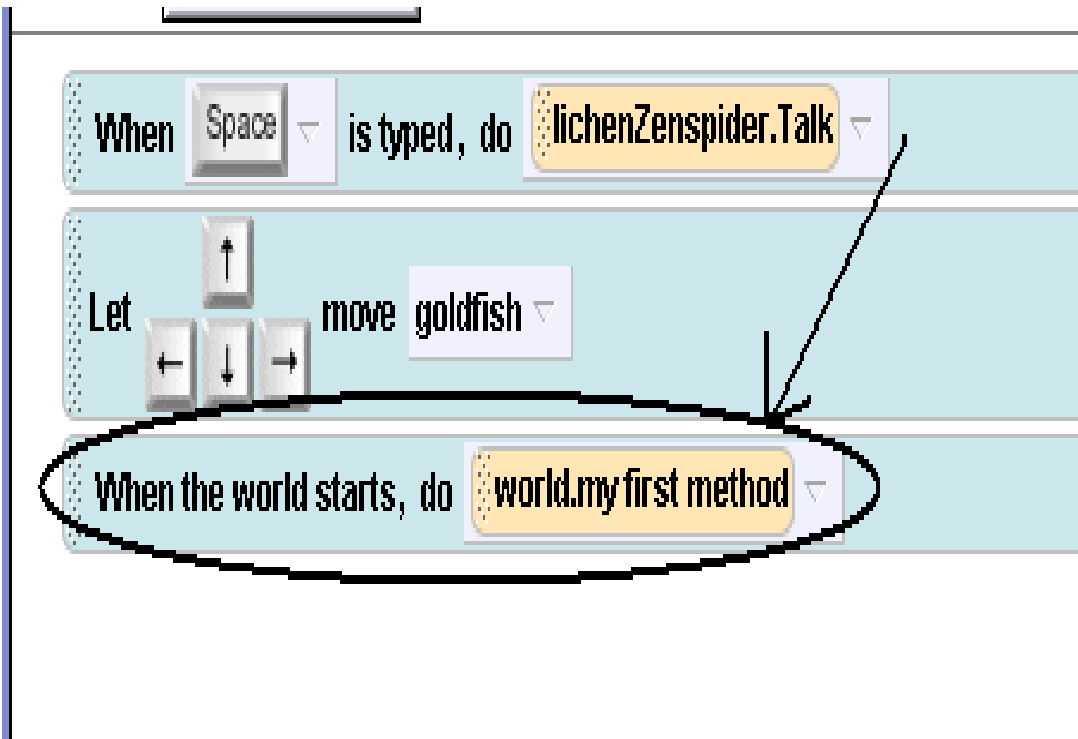
- Now we can use the event “When the world starts” to call it.

When Space ▾ is typed, do lichenZenspider.Talk ▾

Let ↑ ↓ ← → move goldfish ▾

When the world starts, do world.my first method ▾

# Events and Methods



- If you look at your events you will see that when you play your world the fairy will fly around the fish and then get on its back.
- You can have her talk by pressing the space bar
- And move the fish with the arrow keys.
- Try it!

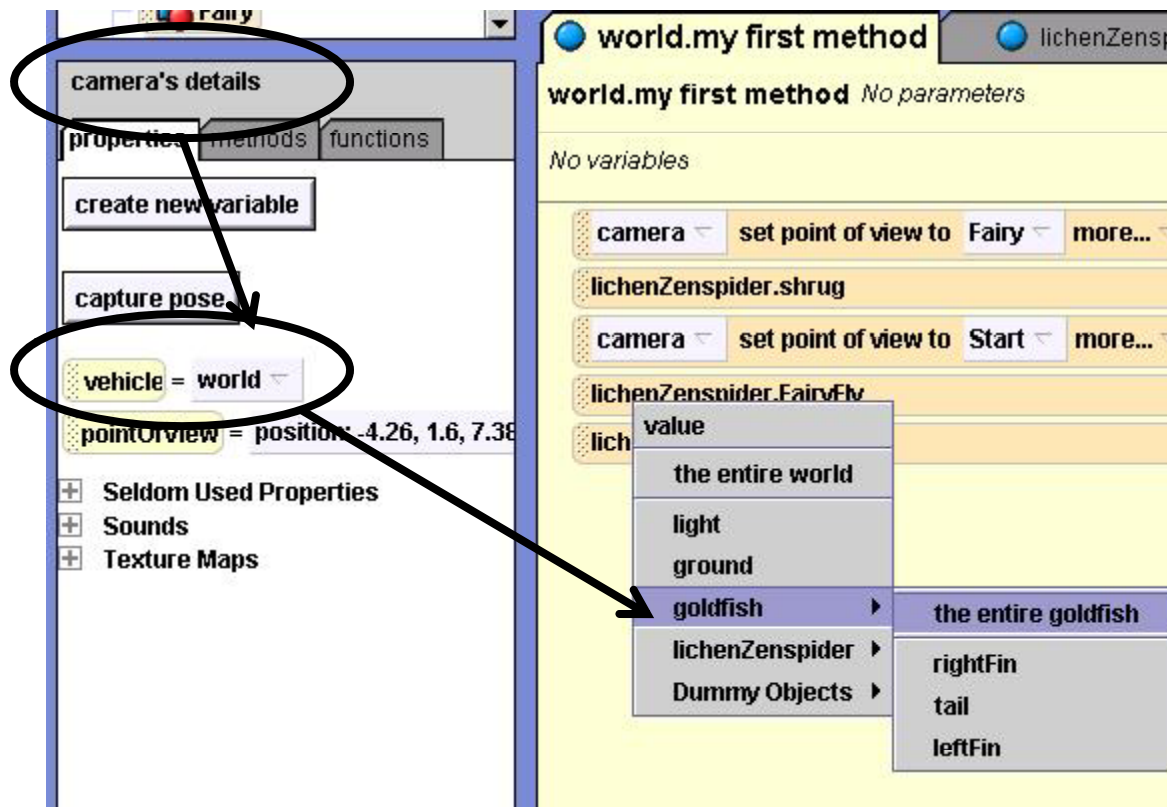
# Events and Methods

- Wait! Our fish swam away!
- Hmm how can we fix it so that we can follow the fish?

Let's set the camera as a vehicle to the fish!


# Events and Methods

- Click on “Camera”, “Properties”
- Drag the vehicle property inbetween “FairyFly” and “RideFish” and set it to “goldfish”, “the entire goldfish”.



# Events and Methods

- Once you have done this try playing your world again.



The image shows a screenshot of a Scratch script editor. At the top, there are two tabs: 'world.my first method' (active) and 'lichenZenspider.'. Below the tabs, the script editor shows a script for 'world.my first method' with the following code blocks:

- world.my first method** *No parameters*
- No variables*
- camera ▾ set point of view to Fairy ▾ more... ▾
- lichenZenspider.shrug
- camera ▾ set point of view to Start ▾ more... ▾
- lichenZenspider.FairyFly
- camera ▾ set vehicle to goldfish ▾ more... ▾
- lichenZenspider.RideFish

The block 'camera ▾ set vehicle to goldfish ▾ more... ▾' is circled in black.

# Notes on “Vehicle Property”

- If you later want to unglue the fairy from the fish or the unglue the camera, see these instructions.
- To unglue the fairy from the fish, set the vehicle property for the fairy back to “world”.
- To unglue the Camera from the fish, also set its vehicle property back to “world.”

# Now try to do these!

- Drop an island from the environments folder.
- Steer the fish around the island.
- Unglue the fairy (by setting her vehicle back to world) and have her move over to the island.
- Try gluing the camera to the fairy instead of the fish.



**Congratulations! You have just  
made your first Alice world. Come  
back and see me soon!!!!**