ORACLE

Academy

Java Fundamentals

2-7
Functions

ORACLE Academy

Objectives

- This lesson covers the following objectives:
  - Using functions to control movement based on a return value

Functions

- Functions:
  - Are used to ask questions about properties of an object
  - Are similar to procedures except that they return a value of a particular type
  - Can be used to compute a value

Functions answer questions about an object, such as its height, width, depth, and even its distance to another object.

Functions are important as they can give you information about your object at any point in your animation. The information can then be used to carry out precise animations.

Functions Precisely Answer Questions

- Functions provide precise answers to questions, such as:
  - What is the distance between the Dalmatian and the bunny?
  - What is the height of the playingCard?
  - What is the width of the pocketWatch?
- A boolean function returns either a true or false value
- For example, if the isFacing function is called to determine if the Alice object is facing the bunny object, a true or false value will be returned

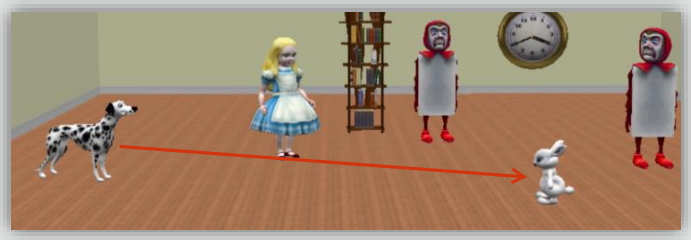
Functions Tab

- The Functions tab is in the methods panel
- Select the object from the Instance menu, and then view its functions
- This is where you can find information on joints that are specific to the type of object that you have added to your animation

The functions tab contains all of the general functions that an object inherits as well as access to the specific joints only available to that class. An example of this would be a rabbit's ears or a dragon's wings.

Functions Solve Distance Problems

- Suppose we want to move the Dalmatian directly to the bunny without having to manually determine, through trial and error, the distance between the Dalmatian and the Bunny
- We could guess the distance by specifying a placeholder value and testing the movement until we get close to the desired end result, but a more efficient way is to use a function to determine the exact distance to move

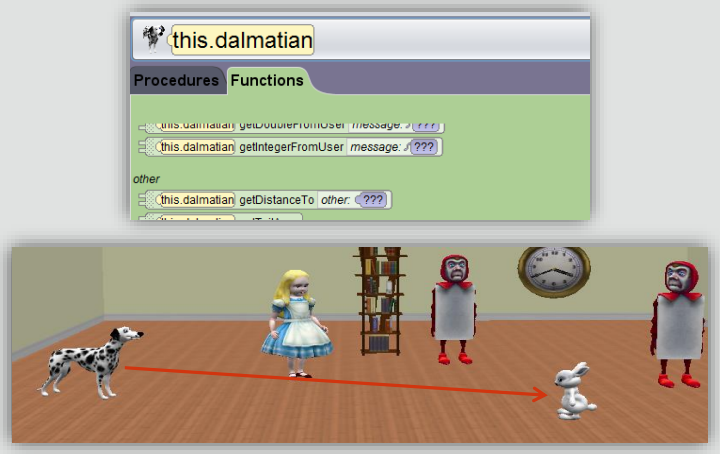


ORACLE Academy

Using functions to create dynamic animations that execute differently every time that you run them is one of the major strengths of Alice 3.

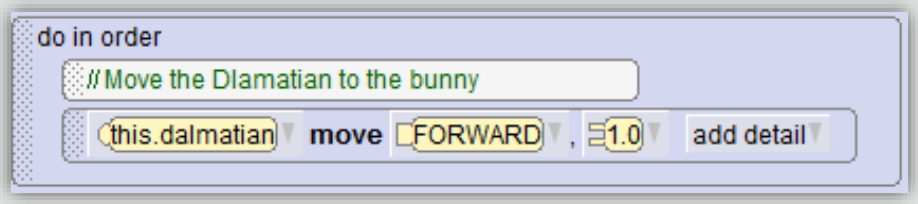
Use getDistanceTo Function

- Use the getDistanceTo function as part of a move procedure to solve this distance problem



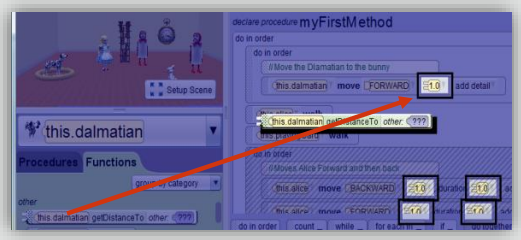
Steps to Use the getDistanceTo Function

- Determine the moving object and target object
- In the Code editor, select the moving object from the Instance menu
- Drag the move procedure into the Code editor
- Select the direction and a placeholder argument for distance (the distance argument will be modified in the next step)



Steps to Use the getDistanceTo Function

- From the Functions tab, drag the getDistanceTo tile onto the highlighted distance value

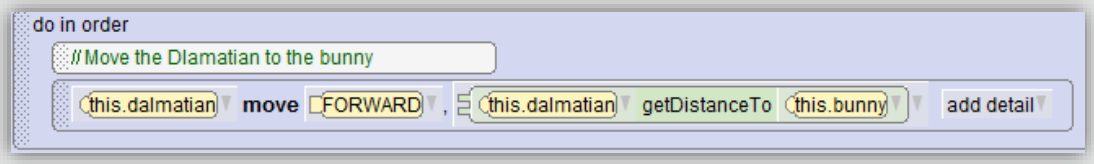


- Select the target object

ORACLE Academy

Test the Function

- In the example below, the Dalmatian moves to the center of the Bunny at run-time
- This could be read out loud as "determine the distance from the center of the Dalmatian to the center of the Bunny and then move the Dalmatian forward that amount"



- Click the Run button to test the programming statement

It is good practice to read your Alice code aloud. It gives you a clearer understanding of what the code does and is useful for when you move on to other Integrated Development Environments (IDE's) that are not drag and drop based.

Test the Function

- The Dalmatian moves to the middle of the Bunny
- This is because the getDistanceTo function calculates the distance between the centers of both objects
- The function calculated the distance from the center of the Dalmatian to the center of the bunny, and moved the object using that value

ORACLE Academy

Sometimes using functions has unanticipated results. This tends to come from inexperience, as you use them more you will be more comfortable with the exact information that they return.

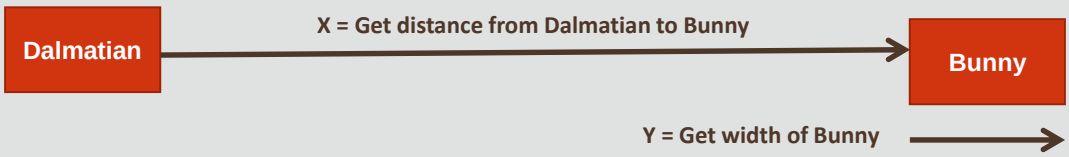
Avoid Collisions

- You can enhance function calls using the math operators:
  - (+) addition
  - (-) subtraction
  - (*) multiplication
  - (/) division
- For example, you can reduce the distance an object will move to avoid a collision

The math operators are used within Alice exactly as they are anywhere else.

Using Math Operators

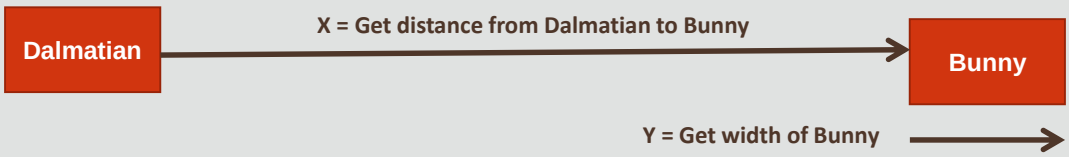
- A function determines the distance between the Dalmatian and the Bunny
  - To reduce the value returned by the getDistance function, the subtraction operator subtracts a specified value
  - The specified value is determined by calling the getWidth function and dividing that value in half
    - Z = X – (Y / 2)



Remember an object's co-ordinates specify the center point of an object. This is why you need to take into account the width of the objects as well.

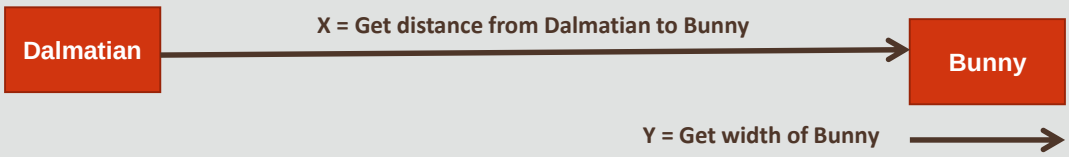
Examine the Math Calculation

- Let's examine the math calculation Z = X - (Y / 2):
  - Z represents the total distance the Dalmatian will move
  - X represents the distance between the Dalmatian and Bunny
  - Y represents the width of the Bunny
  - Y / 2 represents the width of the Bunny divided by 2
  - ( ) represent the order of precedence



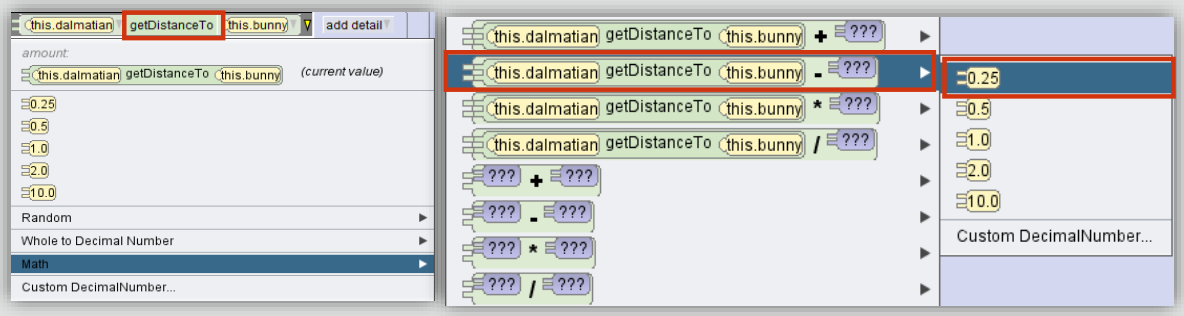
Math Operator Tip

- Why did we divide the width of the Bunny in our calculation?
  - Because we want the animation to appear as though the Dalmatian is moving to the very edge of the Bunny
  - If we used the entire width of the Bunny, the Dalmatian would stop further from the Bunny than desired



Steps to Use Math Operator to Avoid Collision

- Click the function name between the two objects
- Select Math
- Select the getDistanceTo subtraction option
- Select a fixed distance amount

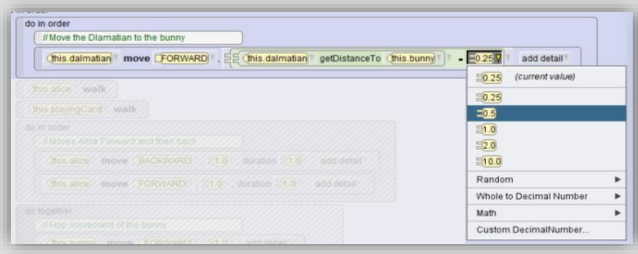


It can seem quite daunting when you first start to build statements that contain math operators. As with everything else it gets easier with practice.

Steps to Use Math Operator to Avoid Collision

- Run the animation to test how the object moves at run-time

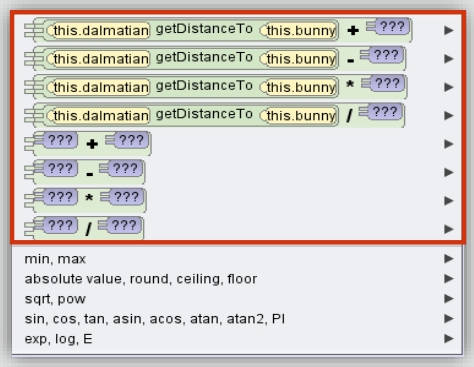
- Adjust the fixed value to get the desired result



This is another case when testing your animation is crucial. Use the required distance value and test your program. Adjust the value as required and re-test. You should continue this process until you are happy with the results.

Understanding the Math Menus Example 1

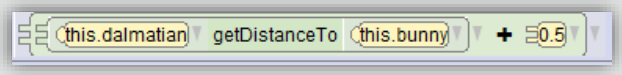
- The following image displays the math operators (+ - * /) requiring one or two arguments
- Each option will provide one or two cascading menus to specify the argument values



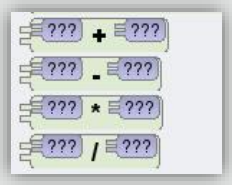
The top four menu options allow you to choose a single value that will be used as part of the equation on the function value. The next four allow you to choose the value on both sides of the equation.

Understanding the Math Menus Example 2

- The following image displays an (+) addition operator requiring a single argument



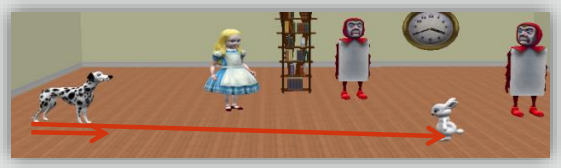
- The following image displays math operators requiring two arguments

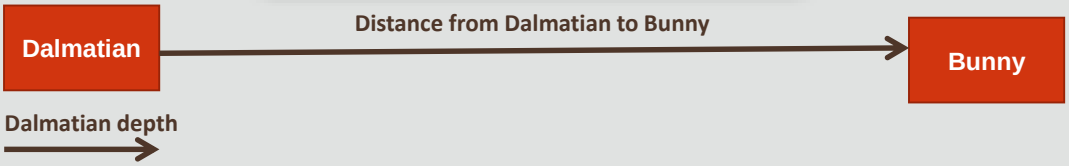


- Remember, you can select placeholder values for the arguments
- Placeholder values can always be edited

Remove Object's Depth from Function

- Another precise way to avoid collisions is to remove the depth (length) of the moving object from the function
- In the example below, the Dalmatian will move the distance to the Bunny, minus the depth of the Dalmatian



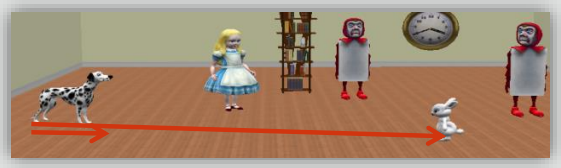


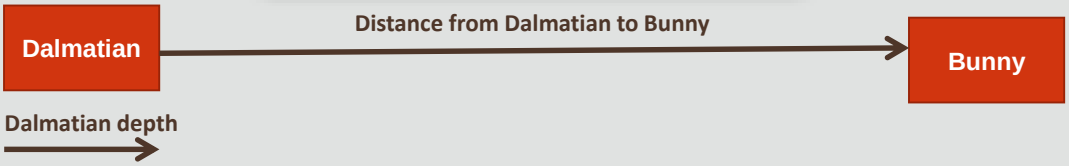
ORACLE Academy

Be careful not to use the objects width in this equation. That is a common mistake and leads to incorrect results when you execute the code.

Depth is Measured from Object's Center

- When a distance value is calculated, it is measured from one object's center to another object's center
- The same is true for math calculations
- When the depth of the Dalmatian is subtracted from the Bunny, it is actually subtracted from the center of the Bunny



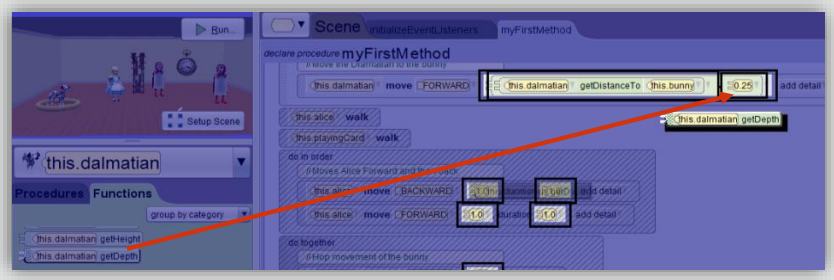


Dalmatian depth

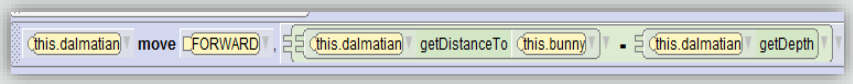
ORACLE Academy

Steps to Remove Depth from Function

- In the Functions tab, drag the moving object's getDepth function onto the highlighted distance value



- Run the animation to test how the object moves at run-time
- Adjust with additional math calculations if necessary



ORACLE Academy

By clicking on the math operator you can edit the statement to suit your needs.

Terminology

- Key terms used in this lesson included:
  - Functions
  - Math operators

Summary

- In this lesson, you should have learned how to:
  - Use functions to control movement based on a return value


ORACLE
Academy