

Jurassic Fruit



In this game we have a very hungry dinosaur that needs to be fed. The only problem is this dinosaur is very picky about what he wants to eat. There is a food sprite located to the right of the dinosaur that changes depending on what fruit the dinosaur is in the mood for. The game also contains 3 different food sprites that will be continuously gliding across the screen. In order to score a point in this game, the player must click on the gliding fruit and the fruit that was clicked needs to be the same as the fruit that is showing in the sprite located on the floor next to the dinosaur. If the wrong fruit is clicked, then the player's score will go back to 0. Some important concepts this game covers include broadcasting, cloning, random number, the logical OR operator, as well as other concepts we have covered in previous projects. We will also be setting a time limit in our game. The player will have 30 seconds to score as many points as they can. In the Creature Seeker project we used the "timer" block provided by scratch to keep track of time in our game. We will be doing things a bit differently in this project. Instead of using the timer block we will create a "time" variable and use a loop to enable time keeping in our game. Now it's time to proceed onto step 1 and start creating your game!

1. The first thing we need to do is add the necessary sprites for our game. Navigate to the sprite area and click on "Choose a Sprite". Search for the sprites listed below and add them to our game.



Dinosaur4 x1



Dinosaur3 x1



Bananas x1

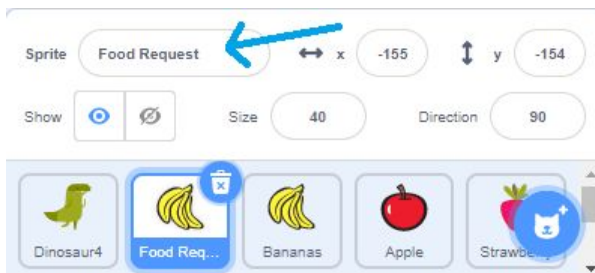


Apple x1

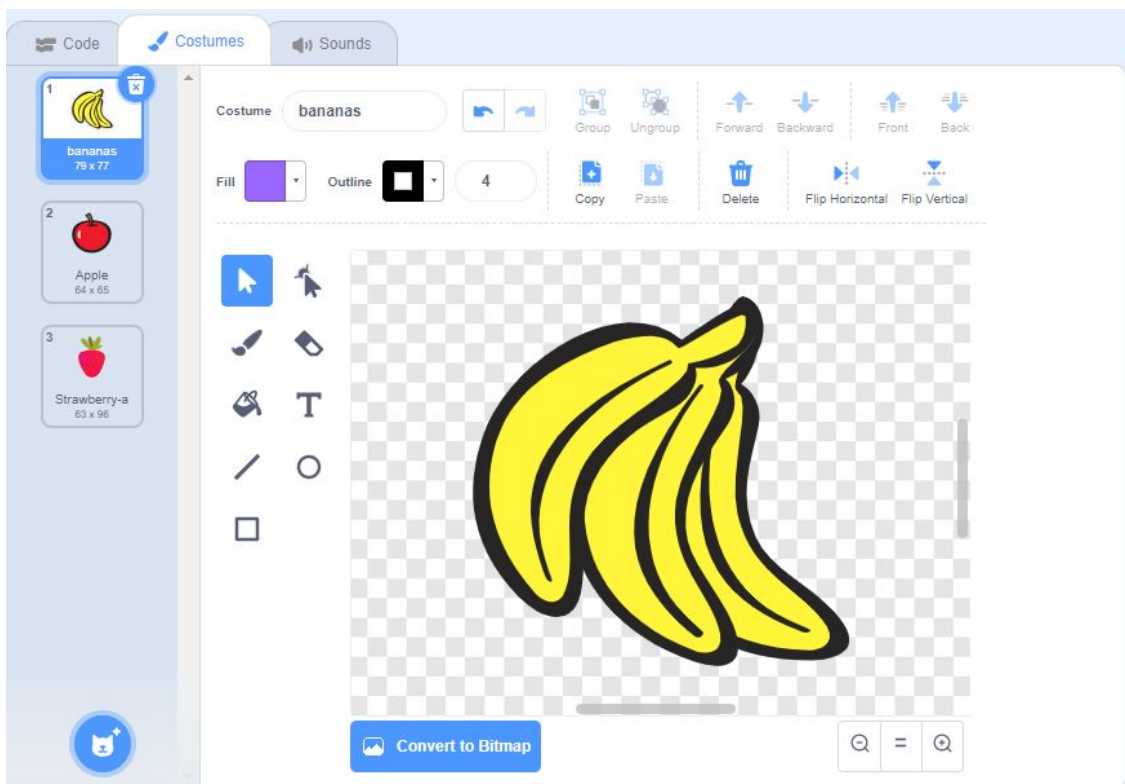


Strawberry x1

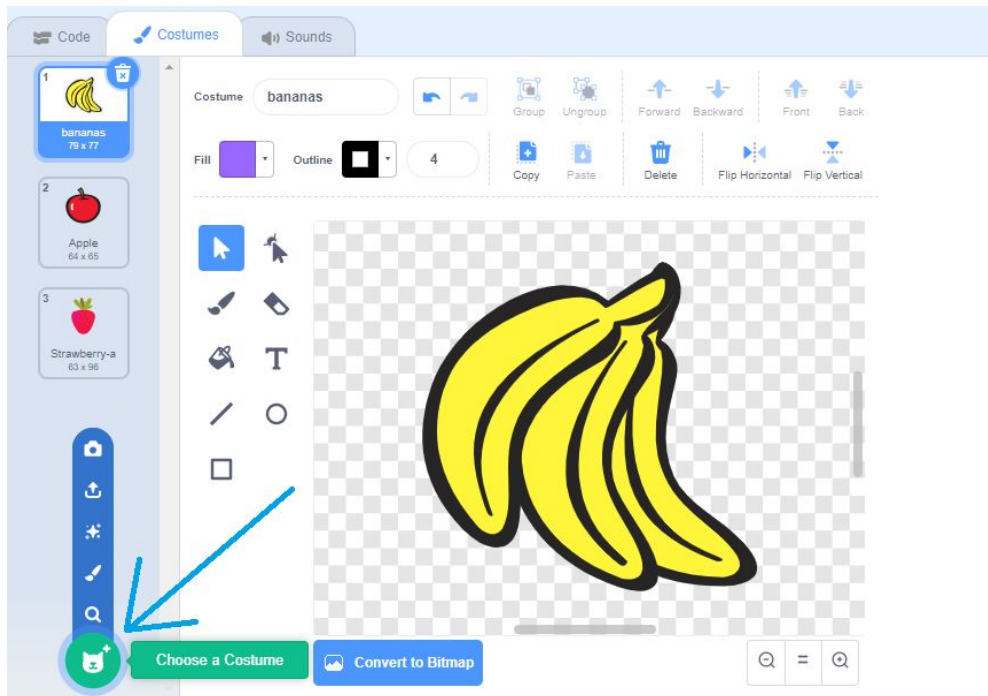
2. Next we will add one more sprite to our game. This sprite displays which food needs to be clicked in order to score a point. First we need to go to the sprite area again and click on “Choose a Sprite”. Search for the same banana sprite we added earlier and add it to our game.
3. Next we want to change the name of the sprite we just added so we do not get it confused with the other banana sprite. Change the name of this sprite to “Food Request”.



4. Now that we have our Food Request sprite, we need to add 2 more costumes to it. First thing we need to do is select the Food Request sprite and click on the costume tab.



5. Now we need to add 2 more costumes to our Food Request sprite. Remember, since this sprite has to display the fruit item that must be clicked on in order to score a point, we need it to be able to switch its costume and make it resemble the other fruit sprites in our game. Click on the “Choose a costume” button, search for the costumes shown below, and add them to the costume options for the Food Request sprite.



Apple

Apple



Strawberry-a

Strawberry-a

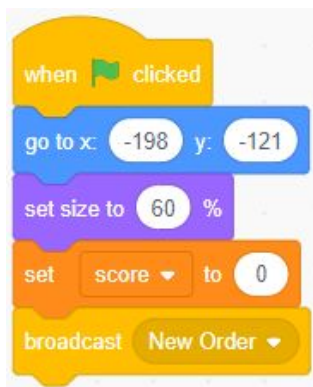
- Now that we have added all of our game sprites to our project we can now work on creating the variables needed in our game. Go to the variables menu and click on the “Make a Variable” button. Below is a list of variables that need to be created. Make sure “For all sprites” is selected when creating our variables.

Food Request

Score

Time

- It’s finally time to start adding some functionality to our game and add code to all of our sprites! We can begin by adding code to our dinosaur sprite. Make sure to select the green dinosaur(Dinosaur4) and add the code below.



This code sets some default values for our dinosaur sprite. After the green flag is clicked and the game begins, we use a “go to x:-198 y:-121” block to position our sprite where we want it. We then use a “set size to 60%” block from the looks menu to set the size of the dinosaur. The “set score to 0” block can be found in the variables menu. This block sets the score back to 0 every time a new game is started. The last block in the script, “broadcast New Order”, can be found in the events menu. In order for the block to say New Order we need to create a new message in the broadcast block. This broadcast block signals the game to generate a new order.

- With the green dinosaur sprite still selected, start a new script for the dinosaur and add the code below.

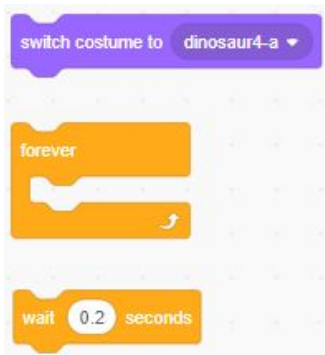


The above code basically receives the signal that was sent from the broadcast in step 7. We retrieve this broadcast using the “when I receive New Order” block. This block can be found in the events menu. The next block down basically sets the value of our Food Request variable to a random number between 1 & 3. The “set Food Request” block is located in the variables menu and the “pick random 1 to 3” can be found in the operators menu. After the variable is set we have 3 IF statements. This is because our Food Request variable can be set to 3 possible values(1, 2, or 3). Inside of our IF statements we have a broadcast block. What this broadcast block does is send a signal to the Food Request sprite. This signal is used to change the costume accordingly for our Food Request sprite.

9. Now it's time to add the code that will animate our dinosaur sprite while the game is running. With the green dinosaur sprite still selected add the code below.

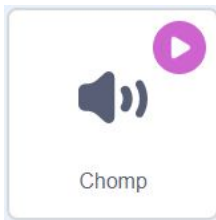


After the “when green flag clicked block”, we are basically only using 3 different blocks.

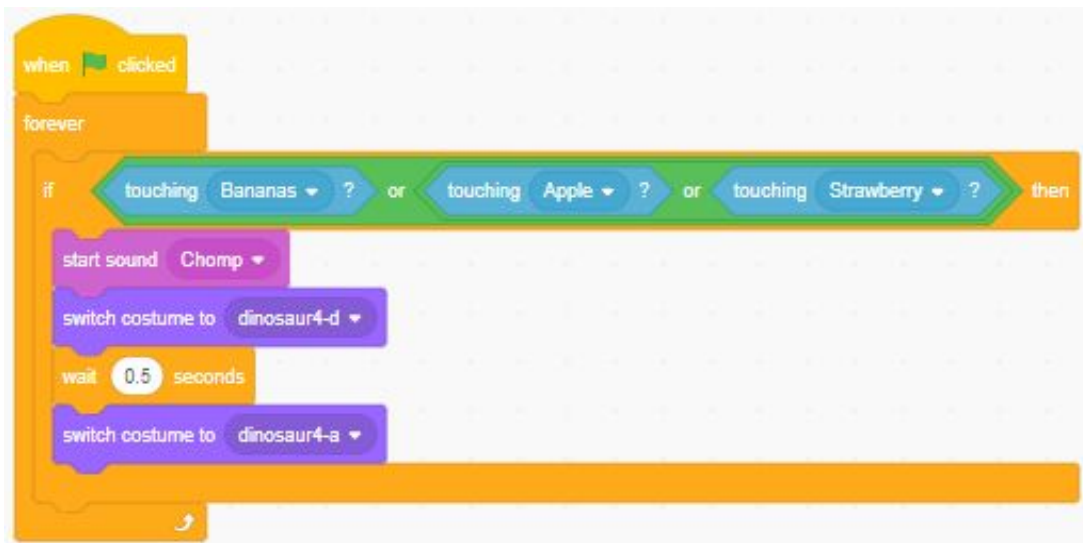


We have a “forever” loop so our animation can repeat, a “switch costume” block to change our sprites costume, and a “wait 0.2 seconds” block to slow the animation down. The reason we aren't using the “next costume” block like we've done in previous projects is because we want to prevent the dinosaur from switching to the 4th costume for this animation. We only want the 4th costume to be shown when our dinosaur is eating the fruit. You will see how this is done in the next step.

10. Before we add the last script to our dinosaur, we need to add some sounds to our sprite. Make sure the dinosaur is selected and click on the sounds tab. Click on the “choose a sound button”, search for the sound shown below, and add it to our sprite.



11. This is the final piece of code that is needed for our green dinosaur sprite. Double check to make sure the green dinosaur sprite is still selected and add the code below.

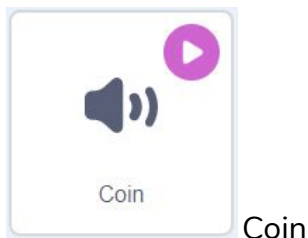


In this code, we are using a “forever” loop with an IF statement nested inside it. This IF statement basically checks if the dinosaur touches the banana, apple, or strawberry. Notice how we are able to check if it’s touching all 3 of the fruit sprites using just 1 IF statement. This is possible because we made use of the logical OR operator that can be found in the operators menu. Also keep in mind that we are using 2 “or” blocks inside of our conditional statement. After you drag 2 “or” blocks on the screen, you will also need to grab 3 “touching” blocks

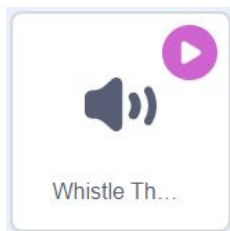
from the sensing menu. Make sure each touching block has a different fruit selected from the dropdown menu.

If the condition in our IF statement is true and the dinosaur touches any of the food sprites then the code nested inside the IF statement will get executed. The “start sound chomp” block can be found in the sounds menu. This enables the dinosaur to make a chomp sound so it sounds like he is eating the fruit. After that block, we have a “switch costume to dinosaur4-d” block. This switches the costume to the image that shows the dinosaur’s mouth open. We then use a “wait 0.5 seconds” block to create a brief pause before our dinosaur switches back to the first costume.

12. Now that we are done coding our green dinosaur sprite, we can begin adding code to our fruit sprites. But before we do this, let’s add 2 different sounds to each of our food sprites. Let’s start with the banana sprite. Make sure the banana sprite is selected and navigate to the sounds tab. When the sounds tab is open, click on the “Choose a Sound” button. Search for the sounds shown below and add them to the banana sprite.



Coin



Whistle Thump

13. Now, repeat what we just did in step 12 for the strawberry and apple sprites. All of our fruit sprites need to be able to play these sounds as well. We will play the “Coin” sound when the right fruit is clicked on and the “Whistle Thump” sound will be played when the wrong fruit is clicked on.

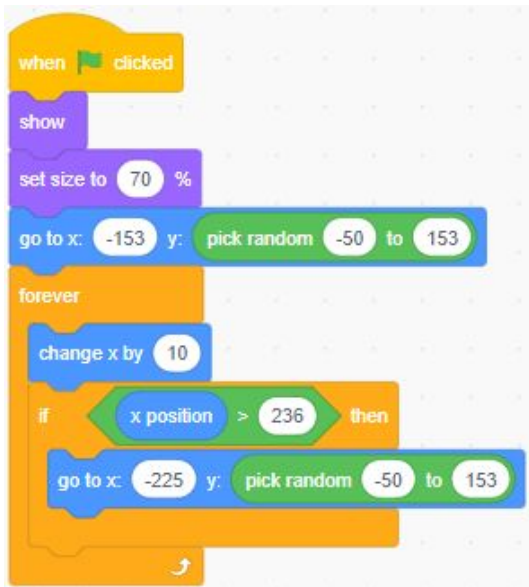
14. It is now time to begin adding code to our fruit sprites. Select the banana sprite and add the code below.



The first thing we want to do right after the green flag is clicked is make our banana visible using a “show block”. When then use a “set size to 70%” block to Shrink our sprite. The “go to x:-244 y:pick random -50 to 153” block sets the position of our sprite on the screen. We set the y-position to a random number because we want it to spawn randomly along the y-axis before it starts moving. The code inside of our forever loop is what makes our fruit sprite move from left to right. We use a “change x by 10” to make our sprite move along the x-axis. The “IF x position > 236” statement checks to see when our sprite touches the right edge of our game screen. If our sprite is touching the right edge, then we reset the sprite’s position to go back to the left side of the screen, and as I mentioned earlier, we want it to spawn randomly along the y-axis. This makes the game more interesting and less predictable.

15. Next we want to make sure the same thing we did on the banana is done for our apple and strawberry sprites. The code is very similar to our banana sprite, except there are differences to where on the x-axis we want the fruits to be placed. Steps 16 & 17 will show the code needed for our apple and strawberry sprites.

16. Select the Apple sprite and add the code shown below.



```
when clicked
show
set size to 70 %
go to x: -153 y: pick random -50 to 153
forever
change x by 10
if x position > 236 then
go to x: -225 y: pick random -50 to 153
```

The image shows a Scratch code block for an Apple sprite. It starts with a yellow 'when clicked' block, followed by a purple 'show' block, a purple 'set size to 70 %' block, and a blue 'go to x: -153 y: pick random -50 to 153' block. Below these is an orange 'forever' loop containing a blue 'change x by 10' block, an orange 'if x position > 236 then' block, and a blue 'go to x: -225 y: pick random -50 to 153' block.

17. Now, select the strawberry sprite and add the code shown below.



```
when clicked
show
set size to 70 %
go to x: -56 y: pick random -50 to 153
forever
change x by 10
if x position > 236 then
go to x: -225 y: pick random -50 to 153
```

The image shows a Scratch code block for a strawberry sprite. It starts with a yellow 'when clicked' block, followed by a purple 'show' block, a purple 'set size to 70 %' block, and a blue 'go to x: -56 y: pick random -50 to 153' block. Below these is an orange 'forever' loop containing a blue 'change x by 10' block, an orange 'if x position > 236 then' block, and a blue 'go to x: -225 y: pick random -50 to 153' block.

18. The next thing we need to do is add the code that checks to see when our fruit sprites are clicked on. There are 3 different scripts shown below. Make note of the event block that starts the script. We want to make sure we use the “when this sprite is clicked” block. The first thing we do after the sprite is clicked on is create a clone of that sprite. We then use an IF/ELSE statement to check if the fruit that was clicked on matches the fruit that is showing in the Food Request icon. Remember back in step 8 when we set the Food Request variable to a random number between 1 and 3? Well now we are checking if the Food Request variable matches the number of the fruit that was clicked on. If the right fruit was clicked on, then we add 1 to the score and play the coin sound. The code in the ELSE statement only runs if the wrong fruit was clicked on. If the wrong fruit was clicked then we set the score back to 0 and play the whistle thump sound.

Select the Banana and add following code



Select the Apple and add the following code



Select the Strawberry and add the following code



19. Now it is time to add the code that controls our fruit once it has been cloned. Below is the code needed for our 3 fruit sprites.

Select the Banana sprite and add the code shown below.



Select the Apple sprite and add the code shown below.

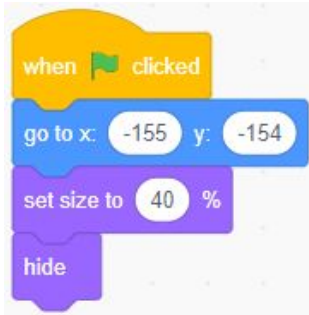


Select the Strawberry sprite and add the code shown below.



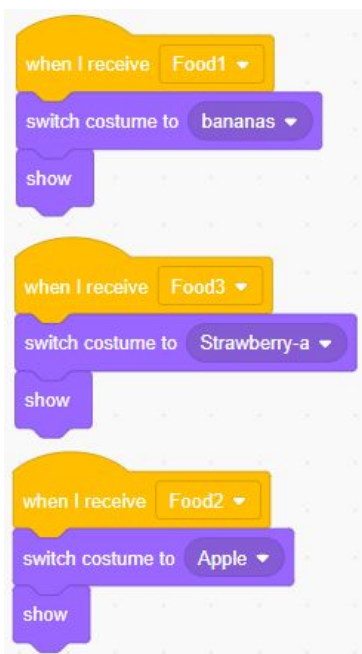
Whenever a fruit is cloned we want it to glide to our dinosaur sprite. This is done using a “glide 1 secs to dinosaur4” block. Once it touches the dinosaur, we then want to hide our clone and broadcast a new order. Every time a new order is broadcasted, the process of randomly choosing a new fruit to feed the dinosaur begins. The last block, “delete this clone”, is needed because we want to delete clones whenever they are not of use to us anymore. If we don’t delete the clones, our game may start lagging and running slower.

20. Remember the sprite we created that has 3 different costumes which represent the 3 fruits we have in our game? Well now it’s time to add code to that sprite. Make sure the Food Request sprite is selected and add the code shown below.



All we are doing in this script is positioning our sprite so it is placed on the floor to the right of our dinosaur. We then use the “set size to 40%” block to shrink our sprite. Finally we want to make sure this sprite is hidden when the game starts.

21. We still have one last piece of code to add to the Food Request sprite. Make sure the Food Request sprite is still selected and add the code shown below.



The code we just added is what we need to receive the broadcast signals we sent in step 8. Depending on the broadcast that was sent (Food1, Food2, or Food3), we want to change the costume to show which type of fruit needs to be clicked on in order to score a point.

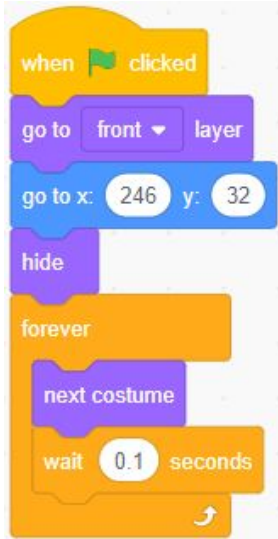
22. How about we add a background to our game. Go to the backdrop area and click on “Choose a Backdrop”. The backdrop that I decided to use for this game is called “Jurassic”. Although, feel free to use whatever backdrop you like.

23. Now that we have a background for our game, let's add some code to our backdrop. The code that is added to our backdrop will basically keep track of time. Make sure the backdrop is selected and add the code shown below.



After the green flag is clicked, we want to make sure we set the “time” variable to 30. This means that when the game begins, the timer will start at 30 seconds. Inside of our forever loop we have a “wait 1 second” block followed by a “change time by -1” block. This basically subtracts 1 from our “time” variable every second. You will also notice an “if time = 0” block. Once the “time” variable is equal to 0 we send an End Game broadcast. This broadcast will be received by our 2nd dinosaur sprite(Dinosaur3). We will add the code for that sprite in the next step.

24. Hang in there, we are almost finished with our project! Select our second dinosaur sprite(dinosaur3) and add the code shown below.



After the green flag is clicked we have a “go to front layer block”. This block makes our sprite appear in front of the other sprites in our game. Then we have a “go to x:246 y:32” block. Just like we’ve done in other projects, we use this block to position our sprite where we want them when the game begins. After that we want to make sure our 2nd dinosaur sprite is hidden using a “hide” block. Finally, we have a forever loop, and inside of our forever loop we are animating the sprite to make it look like it’s flying.

25. Now, make sure the second dinosaur sprite is still selected. We will add one more script that gets executed when the End Game broadcast is sent to us. Add the code below to the 2nd dinosaur sprite.



In this script, we are using a “when I receive end game ” block which can be found in the Events menu. This is the block needed whenever we want to receive a broadcast that was signaled from somewhere else in our program. The

first thing that happens when we receive the End Game broadcast is make sure the sprite is visible using a “show” block. After our sprite appears on the screen, we use a “glide 1 secd to x: -8 80” to make our sprite glide to the center of the screen. We then use a say block to let the player know how many points he scored within 30 seconds. Remember, in order for us to get our sprite to say how many points the player scored, we need to combine 3 different blocks: The “say” block from the looks menu; A “join” block from the operators menu; The variable “Score” which can be found in the variables menu.

GOOD JOB!

You just finished creating Jurassic Fruit! Give it a try and see how many points you can score before the 30 second timer runs out. Don't forget! If you click on the wrong fruit then the score gets set back to 0. The fruit that should be clicked on will be shown in the Food Request sprite located next to the dinosaur. If the game doesn't seem to be working properly, then it's time for you to switch your brain into debugging mode. Double check all of the code to make sure every block is correctly placed. As a challenge, try customizing the game. Maybe even make the timer start at 60 seconds. See if you can increase the speed of the fruit after a certain number of seconds has passed. Maybe you can try adding a super fruit that gives extra points when clicked.