

# CompSci 94

## Classwork: Camera Control

### September 9, 2021



Prof. Susan Rodger

CompSci 94 Fall 2021

# Classwork today

- We will setup a scene for today using the 2D camera scenes
- We will write one new procedure and use the tortoise from last time with its kick and write the story
- Then we will add in some camera markers and add different camera views to the story

# Brief outline of Story

- The tortoise kicks the skate into the hole. Then the tortoise kicks the gnome over and then kicks it out of the scene. The tortoise then jumps over the baby Yeti who charges it, and then the baby Yeti falls into the hole. The tortoise jumps over the Yeti, who falls into the hole. The tortoise faces front and says “The end”.
- **Note:** Follow the instructions and the more detailed story that follow
- **Note2:** Set up Camera views last!

# 1) Setting up the project and scene

- First load your Alice project from last time, and then select **File, Save As**, and name it **Classwork5Sept9**.
- It should already have the **desert** ground cover
- **Delete all the code in your myFirstMethod.**
- Then Delete all the objects **except the tortoise!**
- Drag in these objects so they are in positions similar to the pictures on the next 4 pages
  - Biped: tortoise (already have), yetiBaby, yeti
  - Props: gopherHole, gardenGnome, iceskate

# From the 2D FrontView

Don't move the camera (black object)!  
YetiBaby is at Yetis feet



# From the front view – note yeti and yetiBaby just offscreen



# Here is the side view

Make sure they are all lined up

Also make sure yeti and yeti baby are facing gopherHole



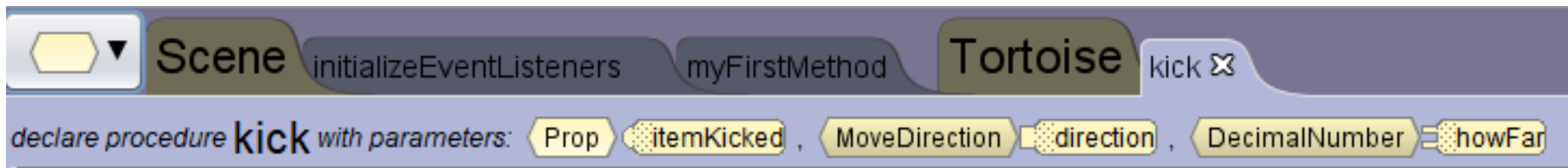
# Here is the Top view You can't see the yetiBaby





## 2) You should already have the **tortoise kick** procedure

- It has three parameters
  - **itemKicked** of type prop – the item to kick
  - **direction** of type moveDirection – the direction for the itemKicked object to move
  - **howFar** of type DecimalNumber – the distance for the item kicked to move



The screenshot shows a programming environment with a dark blue header bar. On the left, there is a yellow hexagon icon with a dropdown arrow. The header bar contains several tabs: 'Scene' (with a dropdown arrow), 'initializeEventListeners', 'myFirstMethod', 'Tortoise', and 'kick' (with a close icon). Below the header bar, the text 'declare procedure kick with parameters:' is followed by three parameter labels in yellow boxes: 'Prop' (with a dropdown arrow), 'itemKicked' (with a dotted pattern), and 'MoveDirection' (with a dropdown arrow). These are followed by two more parameter labels in yellow boxes: 'direction' (with a dotted pattern) and 'DecimalNumber' (with a dropdown arrow), and finally 'howFar' (with a dotted pattern).

```
declare procedure kick with parameters: Prop itemKicked , MoveDirection direction , DecimalNumber howFar
```

## 2) Tortoise **kick** (continued)

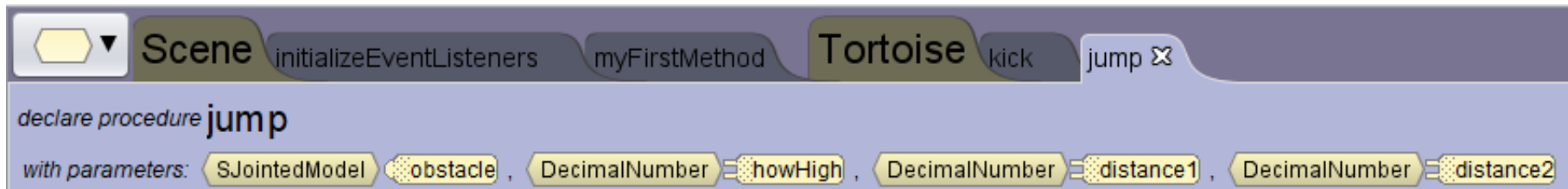
- The tortoise should kick its right leg once (at the knee – use `getRightKnee`) in three motions shown here from left to right



- Note with the last motion the object moves at the same time.
- You should already have this procedure

### 3) Write the **tortoise jump** procedure

- It has four parameters
  - **obstacle** of type **SJointedModel** – the object the tortoise will jump over
  - **howHigh** of type **DecimalNumber** – how high to move up
  - **distance1** of type **DecimalNumber** – how far for the obstacle to move the first time
  - **distance2** of type **DecimalNumber** – how far for the obstacle to move the second time



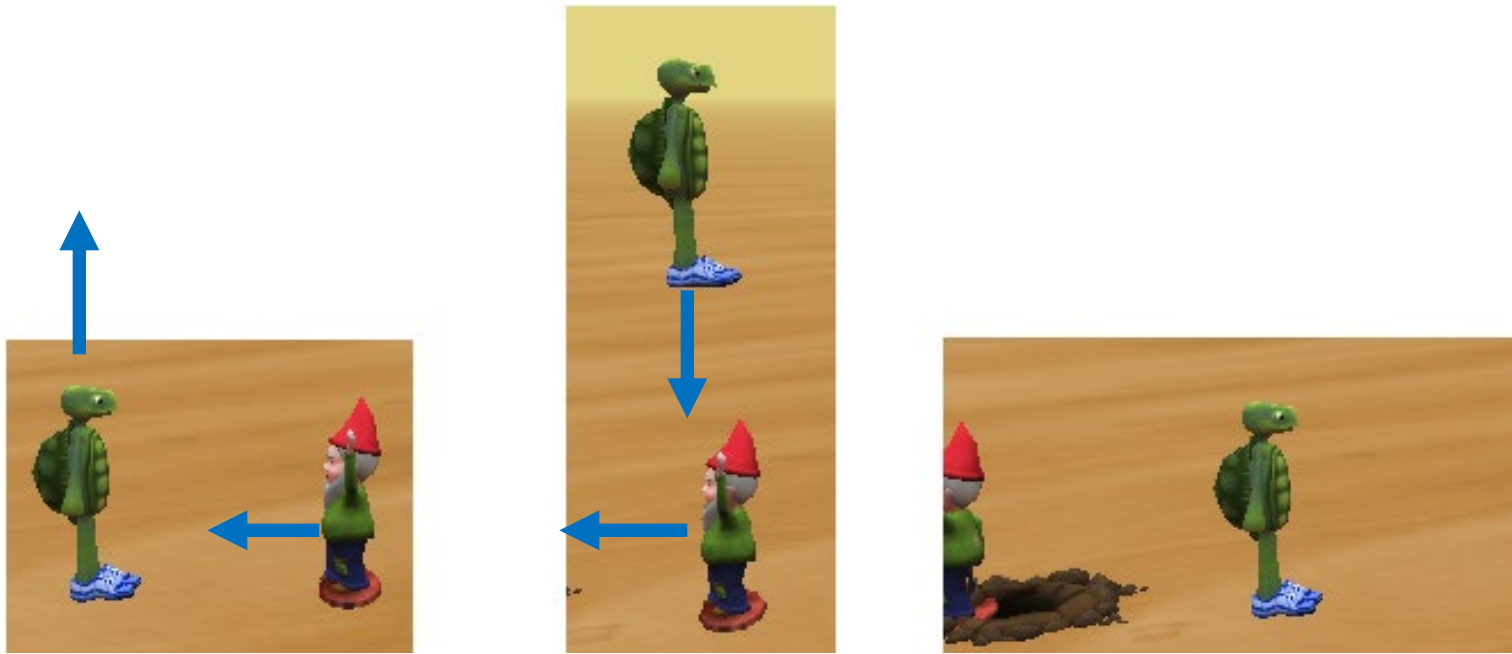
```
declare procedure jump
with parameters: SJointedModel obstacle , DecimalNumber howHigh , DecimalNumber distance1 , DecimalNumber distance2
```

# 3) Tortoise **jump** (continued)

- Here is how the jump works
  - There are just two movements
  - At the same time, the tortoise goes straight up and the obstacle moves towards the tortoise the first distance, likely stopping directly below the tortoise
  - At the same time, the tortoise goes back down and the obstacle moves the second distance
  - See pictures on the next page

### 3) Tortoise **jump** (continued)

- From left to right here is the jump
- Be sure to test the jump out in myFirstMethod with the gardenGnome



## 4) Put the story together

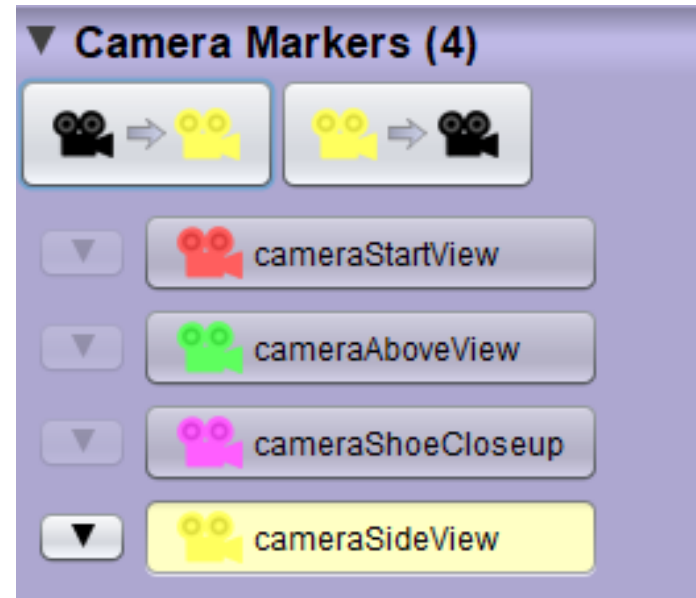
- Delete any testing code you have in `myFirstMethod`
- Put in a `do` in order tile
- Here is the story with more detail:
  - The tortoise turns to face the skate, moves over to it and kicks it, it moves to the hole and then drops down into the hole.
  - The tortoise turns to face the `gardenGnome`, moves over to it and kicks it. The `gardenGnome` moves and then it falls over backward. The tortoise moves up to it again, kicks it and this time it moves off screen.
- (continued)

## 4) Put the story together (continued)

- The baby yeti approaches. The tortoise jumps over the baby Yeti as it passes by the tortoise.
- The baby Yeti reaches the hole and drops underground.
- Next the yeti approaches. The tortoise says something like “Whoa, another”
- The tortoise jumps over the Yeti as it passes by. The yeti drops through the hole.
- The tortoise faces front and says “The end”

## 5) Add four camera markers

- It is important to add camera markers last, or your setup scene may freeze and you cannot click on objects and move them.
- Drop a camera marker before moving the camera for the cameraStartView
- Then add three more camera markers described on the next few pages.





# *Tips on Camera Markers*

- Once you put in camera markers, you shouldn't be moving objects around. Camera markers should go in last!
- If you do need to move an object after Camera markers are added, you may have to use a one shot to do so.
- We will see another way to do camera markers later.

# Add a close up view of the skate – cameraShoeCloseup

- Be sure to see the hole and also the tortoise's legs



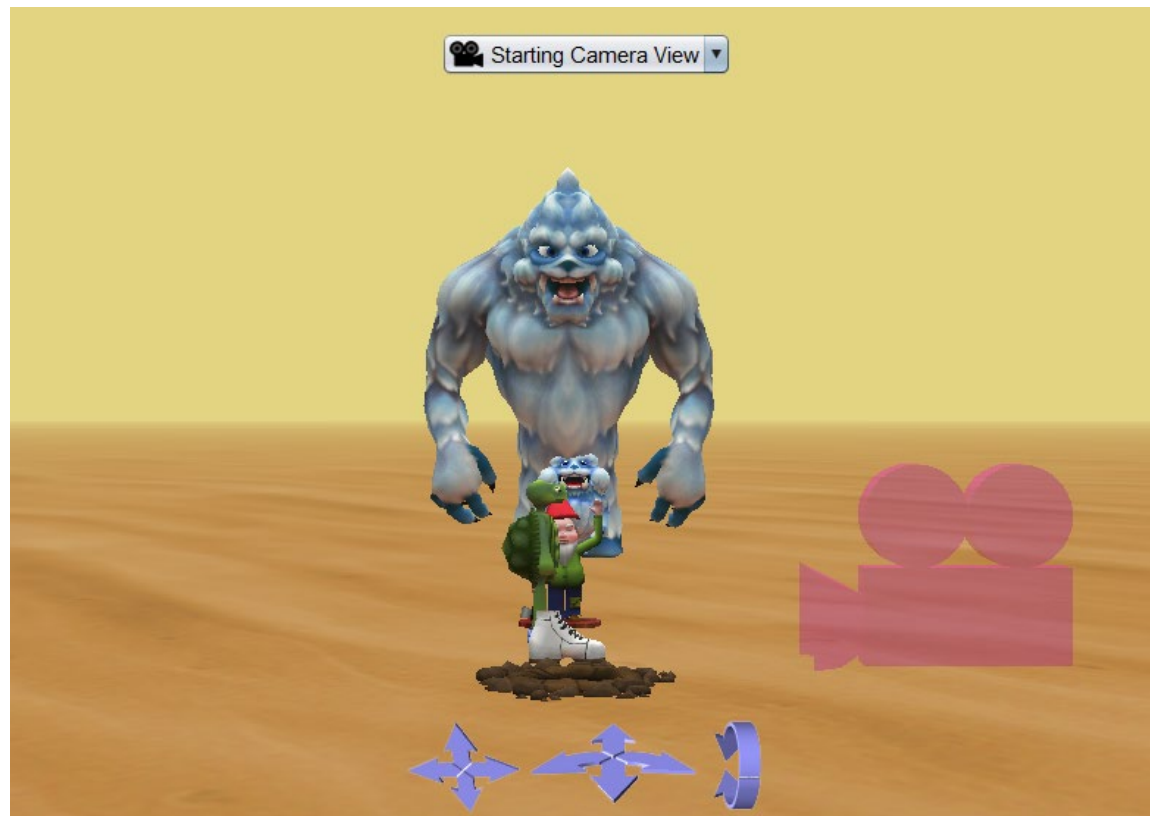
# Add a top view – cameraAboveView

- Try moving the camera to the tortoise, upright it, then move the camera up and turn it to look down. Try this with one shots.
- You can see the shoe cameraMarker



# Add the side view camera marker

- Put the camera back to the cameraStartView and start from there with oneShots



## 6) Add camera views to the story.

- Start in the cameraStartView
- Change to the cameraShoeCloseup when the tortoise faces the shoe
- Change back to the cameraStartView after the shoe disappears
- Change to the cameraAboveView after the gardenGnome falls over
- Change to the cameraStartView after the gardenGnome goes off screen.
- (continued)

## 6) Add camera views (cont.).

- Change to the `cameraSideView` after the `yetiBaby` falls through the hole
- Change to the `cameraStartView` after the `yeti` falls through the hole