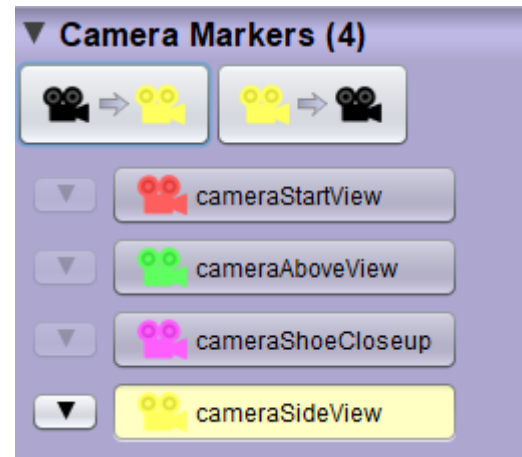


CompSci 94

Camera Controls

September 9, 2021



Prof. Susan Rodger

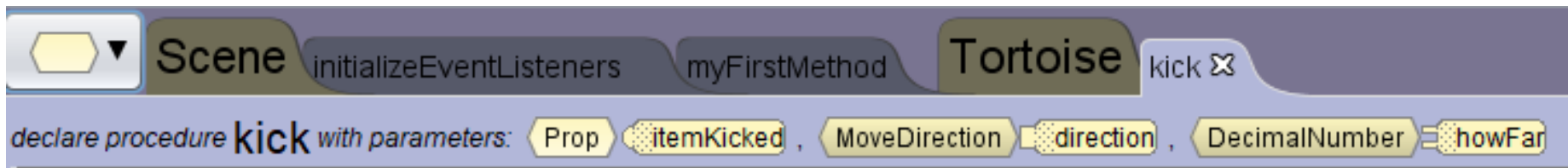
Announcements

- Assignment 2 due on Tuesday, Sept 14
 - Storyboard
 - Alice program
 - Make a Video – talk about code, run
- How to Submit Assignment 2
 - Storyboard, Alice program, video → Sakai
 - URL for Video → Reflect form
- Coming– procedures for classes of objects, and properties.
- QZ06 due Tuesday by class time

Review – Parameters/Arguments

Write 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

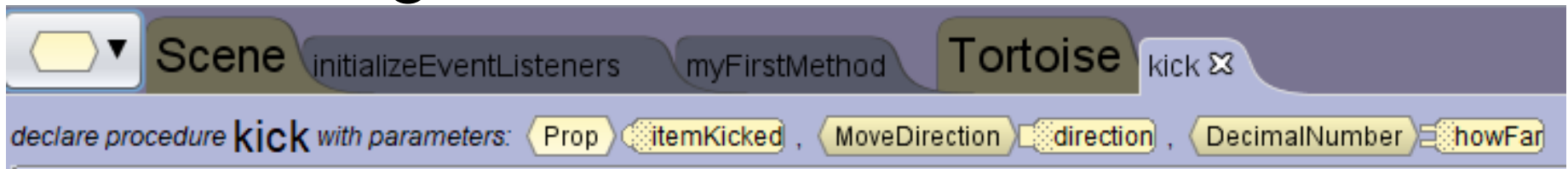


```
declare procedure kick with parameters: Prop itemKicked , MoveDirection direction , DecimalNumber howFar
```

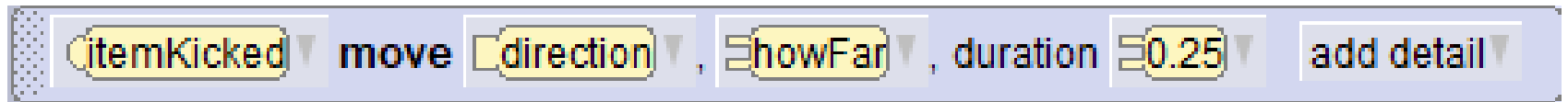
Review

Defining instruction, using parameters

- Defining the instruction



- Using the parameters in the kick code



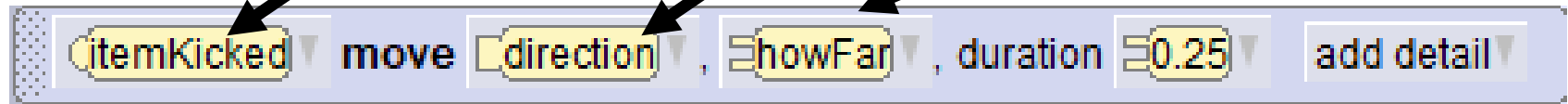
Review

Defining instruction, using parameters

- Defining the instruction



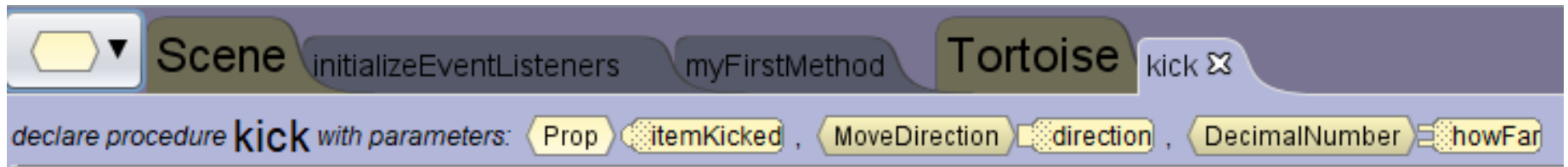
- Using the parameters in the kick code



Review

Defining instruction, Calling instruction

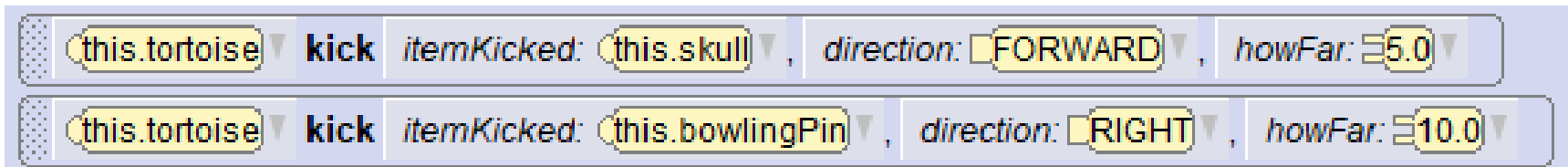
- Defining the instruction



The screenshot shows a code editor with two tabs: 'Scene' and 'Tortoise'. The 'Tortoise' tab is active and contains the following code:

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

- Calling the instruction **in myFirstMethod**
pass arguments to parameters



The screenshot shows two lines of code in a code editor, each representing a call to the 'kick' procedure. The first line is:

```
this.tortoise kick itemKicked: this.skull, direction: FORWARD, howFar: 5.0
```

The second line is:

```
this.tortoise kick itemKicked: this.bowlingPin, direction: RIGHT, howFar: 10.0
```

Review

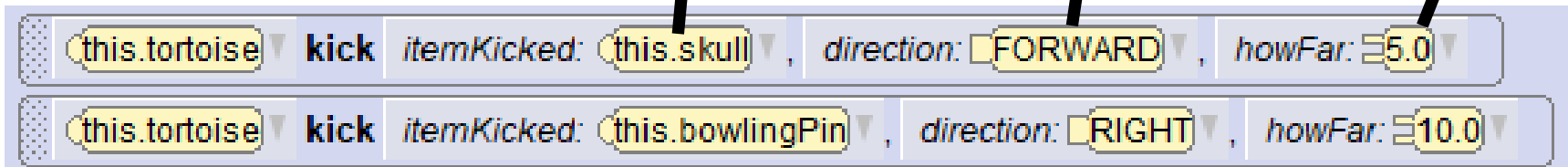
Defining instruction, Calling instruction

- Defining the instruction



The screenshot shows the Scratch IDE interface. At the top, there are tabs for 'Scene', 'initializeEventListeners', 'myFirstMethod', and 'Tortoise'. The 'Tortoise' tab is active, and a 'kick' instruction is selected. Below the tabs, the code editor shows the definition of the 'kick' procedure: `declare procedure kick with parameters: Prop itemKicked, MoveDirection direction, DecimalNumber howFar`. Three arrows point from the parameter names in the code editor to the corresponding arguments in the procedure calls shown in the next screenshot.

- Calling the instruction in **myFirstMethod**, pass arguments to parameters




The screenshot shows two calls to the 'kick' instruction. The first call is: `this.tortoise kick itemKicked: this.skull, direction: FORWARD, howFar: 5.0`. The second call is: `this.tortoise kick itemKicked: this.bowlingPin, direction: RIGHT, howFar: 10.0`. Three arrows point from the parameter names in the code editor above to the arguments in these calls.

Review

Defining instruction, Calling instruction

- Defining the instruction



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

The screenshot shows the Scratch IDE interface. At the top, there are tabs for 'Scene', 'initializeEventListeners', 'myFirstMethod', and 'Tortoise'. The 'Tortoise' tab is active, and a 'kick' instruction is selected. Below the tabs, the procedure definition is shown: 'declare procedure kick with parameters: Prop itemKicked, MoveDirection direction, DecimalNumber howFar'. Three black arrows point from the parameter names 'itemKicked', 'direction', and 'howFar' in the code to their corresponding values in the two call examples below.

- Calling the instruction in myFirstMethod, pass arguments to parameters



```
this.tortoise kick itemKicked: this.skull, direction: FORWARD, howFar: 5.0  
this.tortoise kick itemKicked: this.bowlingPin, direction: RIGHT, howFar: 10.0
```

The screenshot shows two instances of the 'kick' instruction being called. The first call is: 'this.tortoise kick itemKicked: this.skull, direction: FORWARD, howFar: 5.0'. The second call is: 'this.tortoise kick itemKicked: this.bowlingPin, direction: RIGHT, howFar: 10.0'. Three black arrows point from the parameter names 'itemKicked', 'direction', and 'howFar' in the code above to their corresponding values in the two call examples.

Call second time with different arguments!

Q1. Camera Views

- How do you line up the animals in the front in a line?
- How do I make sure the animal behind the ogre is directly behind it?



Q2. Setup Scene

- What other views can you use beside Starting Camera View?

Q3. Camera Markers

- How does one create a camera marker?

- How does one use a Camera Marker during animation?

Q4. More on Camera markers

- When do you add Camera markers?
- What do these buttons mean?

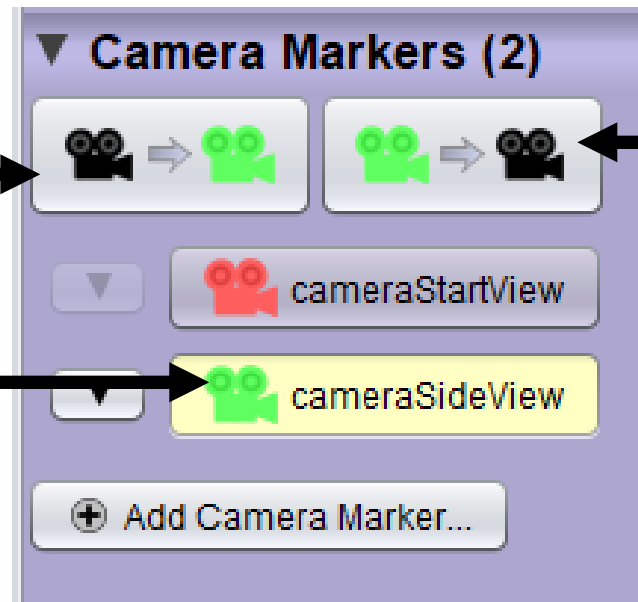
A)



B)



C)



Q5. Setting up a CameraMarker

- How does one setup camera for side view?
 - (give all the steps you would do)



Class Today

- Continue writing procedures with parameters
- Moving between camera views

