

CompSci 4
Chap 5 Sec 2
Oct 16, 2007

Prof. Susan Rodger



Announcements

- Read Chapter 6 Tips and tech for next time
 - Randomness
- Assignment 5
 - Part 1 and 2, due next week
- Prof Rodger no office hours rest of this week
- Today - Lecture on Chap 5, Sec 2
 - Event-handlers with parameters
 - Classwork

Mouse Clicks

- Interactive programs – allow user to mouse click an object
 - Buttons in an interface
 - Targets in a game
 - Checklist of items on a form
- Will see how to pass information about a mouse clicked object to an event handler

Example

- People are trapped in a burning building
- Select which person will be rescued



Storyboard

- Three people are to be rescued
- Could write 3 different methods

Event: click on guy1

Responding Method:
Save guy on first floor

Event: click on girl2

Responding Method:
Save girl on second floor

Event: click on girl3

Responding Method:
Save girl on third floor

A Better Solution

- Write one event handler
- Send in information needed for action

firetruck.savePerson:

parameters: *whichFloor, whichPerson, howFar*

Do in order

point ladder at *whichFloor*

extend ladder *howFar* meters

whichPerson slides down ladder to fire truck

pull ladder back *howFar* meters

What type are the parameters?

Demo

- Demonstration of burning building and setting up events for *firetruck.savePerson*

Three Events


- The argument sent to parameters depends on which person is mouse clicked

Events

☐ World

When  is clicked on ,

do

When  is clicked on ,

do

When  is clicked on ,

do

- Note - we positioned fire truck so distance from floor X is X meters (to floor 3 is 3 meters)

Example 2 – put events in

- Zeus was a powerful god in Greek mythology. When Zeus was angry, he would shoot a thunderbolt out of the heavens to strike anyone who got in the way
- The user will choose the philosopher who will be the next target of Zeus' anger.



Storyboard

- Possible design – method with Object parameter named *who*, for object clicked

Event: an object is mouse-clicked

Event handler: *shootBolt*

Parameter: *who* – object clicked

Do in order

prepare to strike object that was clicked

thunder plays and lightning strikes object clicked

lightning is repositioned for next strike

- The actions in storyboard are complex
- Break actions into simpler steps using stepwise refinement

Event: an object is mouse-clicked

Event handler: *shootBolt*

Parameter: *who* – object clicked

Do in order

prepare to strike object that was clicked

thunder plays and lightning strikes object clicked

lightning is repositioned for next strike

prepareToShoot

Parameter: *target*

Do together

turn Zeus to face the *target*

make the lightning bolt visible

lightning and Thunder:

Parameter: *target*

Do together

play sound

call *specialEffects* method

- send *target*



specialEffects:

parameter: *target*

Do in order

Do together

lightning bolt move to *target*

smoke move to *target*

Do together

set smoke to visible

set lightning to invisible

call smoke cycle – built-in method

set *target* color to black

move *target* up and down

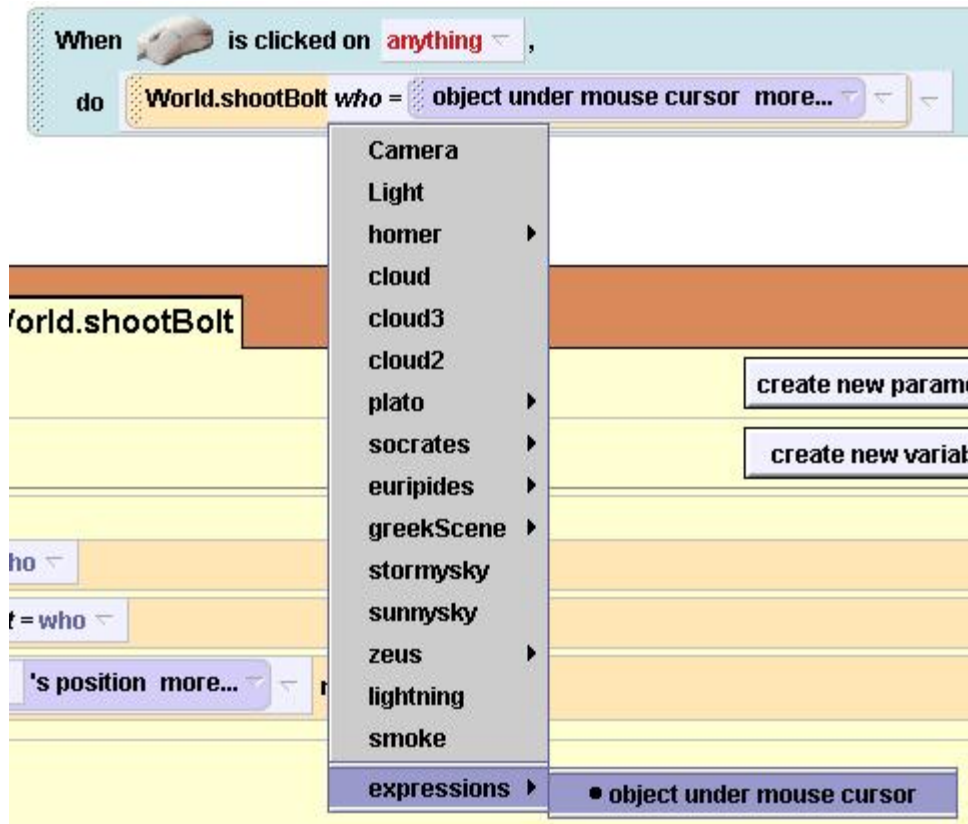
A Driver

- shootBolt method - top level of our design
- It calls other methods and controls the overall action of the program – we call this a **driver**



One Link

- In the fire rescue example, we used three links – one for each person in the burning building.
In this example, we use only one link by selecting “object under mouse cursor” as the argument.



Demo

- Test run of Zeus world – (this version doesn't have the if statements from Chap 6 added in)
- When parameters are used in interactive programming – especially important to test that all possible parameter values work as expected
 - What happens if you click on each philosopher, one at a time?
- Also try things that shouldn't work
 - What happens if you click on a column?
 - What happens if you click on a philosopher twice?
 - What happens if you click on Zeus?

Classwork today

- Create 2 worlds (or can combine them)
 - Problem 14, page 140
 - Problem 15, page 141 (can use any person)

