Objects in the world

- Ground – any ground
- Quadruped
  - Lots of pandas, one alienRobot
- Props
  - Bell

Click-a-panda Game Description

- The game has 3 or more pandas, spread out on the ground and a bell in the top right corner.
- The player is prompted for how many pandas to click on. (The player must type in a number between 2 and the number of pandas)

Click-a-panda Game Description (2)

- The bell welcomes the player to the game and tells them they must click on a specific number of pandas (the number they entered)
- The bell says to click on it to start the game
Click-a-panda Game Description (3)
• The player clicks on the bell.
• The pandas all drop down below ground and the number 0 appears in the top left corner.

Click-a-panda Game Description (4)
• The **panda heads** then randomly pop up to the ground (only one at a time) and then drop below ground.
• The player tries to click on them. If they get a panda, 1 is added to the number displayed.

Click-a-panda Game Description (5)
• When the player clicks on the specified number of pandas, the pandas stop popping up.
• An AlienRobot comes down from the sky to the ground and tells the player that they did great!
Group Work

• In a group of 3 or 4 discuss how to set up this game by discussing the following questions.
• Don’t write any code yet! Just discuss how you would design this game

Question 1

• What things would be good to keep track of? These could be variables, properties, arrays?

Question 2

• Any functions or procedures that would be helpful to write?

Question 3

• Outline which events would be useful to have and the flow of the program? Which parts of the program would be in myFirstMethod vs in an event and which event? Consider the timing…
Does your design consider these?

- If you want to add more pandas in, does your design still work?
- If the user types in the wrong number (-3 or 20) what happens?

Discussion Time

- We will discuss in class and then you can start coding the game.
- Slides from here on are notes taken in class!!
Variables/properties

- answerNumber – whole number (number of pandas total to click on) – **setup as a scene property**
- textString object – to display the score
- scoreNumber - start at 0 and increment everytime click on a panda – this should be a **textString property**
- arrayOfPandas – make this a scene property, since just pandas in it, make it a PANDA array

Variables/properties (cont)

- Boolean variable – gameOn
  - True when we are playing the game
  - False when not playing
  - Start false, set to true when ready to play
  - Turn back to false when game is over
- OR instead use TextString gameState
  - Set to “setup” when in setup mode
  - Set to “run” when game is running
  - Set to “finish” when game is over
Procedures

• Procedure move pandas up and down
  – Just Move one panda up/down that is a parameter
  – Another loop somewhere else determining random panda to send as an argument
• Update counter – add one to counter and redisplay new value
• Initialize counter – all the setup for counter
• Setup – other kind of setup

Flow of the game

• Event – when you click on bell, game starts
  – Setup
  – Reset gameOn variable to true
• Event – click on any panda in the panda array
  – Update score
  – check if hit max score

MyFirstMethod

• Mostly run the game here
• Go through instructions first
• Ask the user how many pandas to click on and set that variable
• While user input no good
  – Ask them again
• More instructions – tell them to click on the bell to start

myFirstMethod (cont)

• While not Gameon
  – Do nothing
  – // This pauses while it waits for you to click on bell
• While gameon - play the game
  – Randomly pick a panda to move and down
    • Generate random number that is an array position
    • Pass the panda in that position to move procedure
• Now game is over – do whatever when game is over