Memory - Code Clinic tips

This page include helpful tips based on our experience in helping students in the "Code Clinic" (interactivepython@online.rice.edu). Be sure and take a look at these tips if you get stuck.

1. Many of you are tempted to make conditionals with 16 cases to handle every possible card position on the canvas. Please don't. Learn how to use loops. For example, if `cards` contains the list of card values, the following code is a good start towards drawing cards

   ```
   for card_index in range(len(cards)):
       card_pos = 50 * card_index
       canvas.draw_text(str(cards[card_index]), card_pos, ....)
   ```

   Note that we could also iterate directly over the cards via `for card in cards:`. However, iterating over the indices of the cards will make adding code to expose cards based on mouse clicks easier.

2. In your mouse click handler, you need to determine which card you have clicked on. Don't use a long if/elif/else statement here. Compute the index of the card via `pos[0] // 50`. You can then use this index when manipulating the list exposed that control whether a card is drawn or not.

3. As you get close to the end of this mini-project, you will need to determine if the two cards match when you transition from state two to state one. The key here is to keep track of the indices of two cards that were clicked on during the last turn (in two separate variables) so you can test whether their values are equal during the next turn.

4. Bonus tip for images
   If you intend to use images in your mini-project, please remember to test your finished mini-project on a separate computer to make sure that your images will load successfully for your peers when they assess your project. It's your responsibility to make sure that the images load successfully.

   Also, I would suggest that you hard code the size of your images into your mini-project. If you compute `my_image.get_width()` and `my_image.get_height()` before `my_image` has finished loading, these calls will return size zero. If you use a source rectangle of size zero in subsequent calls to `draw_image(my_image, ...)`, the call will throw a DOM error.

5. Visit the discussion forum for Memory for more helpful tips on coding Memory by CTAs.

Created Wed 28 Jan 2015 3:44 AM CST

Last Modified Tue 19 May 2015 3:51 AM CST