1. On what line does the definition of the Ball class begin?

2. On what line does it end?

3. How many classes are defined in the file?

4. Which classes inherit, directly or indirectly, from pyglet.sprite.Sprite?

5. When the file is executed, a bunch of classes and functions are defined. Which is the first function that is *executed*?

6. Which class is the first to have an instance created?

7. Describe what the effect of line 355 is? Which Class's update method is being referred to here?

8. What methods are invoked, on what instances, when the user presses the W key on the keyboard? What happens as a result? (Hint: look in the GameWindow class, starting on line 333.)

9. When a new GameWindow is created, is a new Game object also created? If so, which line(s) of code do that?

10. On which lines of code are the two paddles created?

11. Write code that would create a Paddle instance that responds to the key Y to move up and H to move down, and uses a different image file, called 'curved_paddle1.png'.

12. On what line does the definition of the Game class's update method begin?

13. In the Game class's update method, what is the type of self.game_objects?

14. In that same method, what class is self.game_objects[-1] an instance of? (Hint: look at line 296 to see where the game_objects instance variable is set).

15. In the Ball class, the update() method invokes the move() method. On what lines do you find the code for that method?

16. If you want to change the ball's initial velocity to be 10 pixels, what line(s) of code would you change?

17. In the Ball class, what happens in the update() method other than moving the ball? Describe it in one or two sentences.

18. Which line(s) of code handle the event of the ball getting past player two and hitting the right side wall?

19. Which object instance is keeping track of the score in the game? (Hint: look at line 248)

20. Given your answers to the last two questions, what code would you add, and where, to make the first player's score increment by one when the ball hits the right side wall?

21. Right now, the paddle can move off the top of the screen. What might you do to prevent it from going off screen? Describe what methods you would add or change? You don't need to write the code.

The next few questions walk you through building a little maze into the game. wall_imgs[2] has name of an image file, 'brick.png'.

23. Notice that on line 295, we initialize the brick instance variable to be an empty list. Write code to add an extra brick-sized wall, at (x, y) coordinates (100, 100). Where would that code go?

24. Generalize what you just did, using a for loop to generate a stack of 3 bricks. All of them should be at x position 0, but the y coordinate will be increasing by the height of a brick (50 pixels) for each brick.

25. Similarly, make a stack of give bricks coming down from the top of the game, at x coordinate 400.

26. Suppose we want to change how paddles deflect balls to match the bowed out shape I showed on screen. When the ball hits the bottom of the paddle, it goes down. When it hits the top, it goes up. There is already a hit_position() method defined, starting on line 222, that returns a number between 0 and 1, with .5 if the ball hits directly in the middle, .9 if it hits near the top and .1 if it hits near the bottom. What method would you have to define or change in order to make the paddles call hit_position and set the angle of the ball accordingly?