1)
```
rect_x = 50
pygame.draw.rect(screen, WHITE, [rect_x, 50, 50, 50])
rect_x += 1
```

Correction: rect_x += 1 should be put under rect_x so that it can work with the main loop.

2) If the screen is 400 pixels tall, and the shape is 20 pixels high, at what point should the code check to see if the shape is in contact with the bottom of the screen.
The point would be at 3,80 so that when it bounces it'll be able to reach the top of the screen.

3) When drawing a starry background, explain why it doesn't work to put code like this in the main program loop:

```
for i in range(50):
x = random.randrange(0, 400)

y = random.randrange(0, 400)

pygame.draw.circle(screen, WHITE, [x, y], 2)
```

Explanation: It doesn't work when it's on the main program loop because it wouldn't keep the snowflakes in the same spot instead it would generate new positions every second, and instead of keeping a list of where the are the main loop would add more snowflakes to the list. You need to put it before the main program loop.

4) If you have a list of coordinates like the following, what code would be required to print out the array location that holds the number 33?

```
stars = [[ 3, 4],
[33, 94],

[ 0, 0]]
```

Explanation: print(stars[1][0])

5) Process each snow flake in the list
```
for i in range(len(snow_list)):
# Draw the snow flake
pygame.draw.circle(screen, WHITE, snow_list[i], 2)
# Move the snow flake down one pixel
snow_list[i][1] += 1
```

Explanation: Instead of requiring the variable of X and Y its already inputted into the code which gets rid of the need of needed and X and Y variable for snow_ist.

6. What's wrong with this code that uses a function to draw a stick figure? Assume the colors are already defined and the rest of the program is ok. What is wrong with the code in the function?

```
def draw_stick_figure(screen, x, y):

# Head

pygame.draw.ellipse(screen, BLACK, [96,83,10,10],0)

# Legs

pygame.draw.line(screen, BLACK, [100,100], [105,110], 2)

pygame.draw.line(screen, BLACK, [100,100], [95,110], 2)

# Body

pygame.draw.line(screen, RED, [100,100], [100,90],2)
```

```
# Arms

pygame.draw.line(screen, RED, [100,90], [104,100],2)

pygame.draw.line(screen, RED, [100,90], [96,100],2)
```

Explanation: The X/Y variables on the shapes aren't inputted onto the shapes.

7. Show how to grab the mouse coordinates, and then grab just the x coordinate of where the mouse is.

```
os = pygame.mouse.get_pos()
x = pos[0]
y = pos[1]
If you just wanted an x or y coordinate you have to put a list inside
the list, the main list has all the coordinates and inside of that
list each coordinate is a list of an x and a y.
```

8. When we created a bouncing rectangle, we multiplied the speed times -1 when the rectangle hit the edge of the screen. Explain why that technique won't work for moving an object with the keyboard.
It wouldn't prevent a character from moving off the edge of the screen. To do this you need to set a set of if statements that would be needed to check the x coordinate and y coordinates values, if they are outside the boundaries of the screen, then reset the coordinates to the edge.