

When updating the BreakAPlate game to use PNG images instead of GIFs I encountered several key issues that required debugging and fixing.

1. Images Not Displaying

At first, none of the images (for the plates and the prizes) appeared in the game window. This was because the image files were either not in the correct format or weren't located in the expected directory. I needed to ensure all images were converted to PNG format and that the file paths in the code correctly referenced these PNGs.

Error Example:

```
// Incorrect image format and file path
placeholderIcon = new ImageIcon("placeholder.gif"); // Wrong file
type
```

```
// Corrected to:
placeholderIcon = new ImageIcon("path_to_images/placeholder.png");
```

This fixed the issue, allowing the images to display correctly in the game window.

2. Path Errors for Image Files

Another issue came from incorrect file paths. Even after converting the images to PNG format, the program couldn't find them because the paths I had specified in the code were either relative or incorrect. I corrected this by making sure the paths pointed to the correct location where the PNG files were stored.

Error Example:

```
// Incorrect path leading to failure in loading images
tigerIcon = new ImageIcon("images/tiger_plush.gif"); // Wrong file
location
```

```
// Corrected to:
tigerIcon = new ImageIcon("path_to_images/tiger_plush.png"); //
Correct path
```

With this change, the images loaded properly and appeared when the plates were broken.

3. Random Prize Selection Not Working

Initially, the random prize selection wasn't functioning correctly, and the same prize kept appearing after each round. The issue stemmed from an error in the random number generation logic. I corrected the random selection mechanism to ensure that the tiger plush or sticker prize appeared randomly each time.

Error Example:

```
// Incorrect random logic causing the same prize to appear each time  
int prize = random.nextInt(1); // Only generates 0
```

```
// Corrected to:  
int prize = random.nextInt(2); // Generates 0 or 1 for two different prizes
```

After fixing this, the game became more unpredictable and fun as users could now win either of the two prizes randomly.