# Changes and Debugging in Tank Game Code

## Summary of Changes

The following changes were made between the original tank.py and the modified tank\_modified.py to enhance functionality, fix errors, and improve the user experience:

1. Background Integration:  
 - Added a desert-themed background (desert.png) for the main gameplay.  
 - Added a menu background (menu.jpg) for the control screen.  
 - These backgrounds are loaded from local files.  
   
2. Elevator Music:  
 - Integrated background music (elevator\_music.mp3) loaded from a local file.  
   
3. Button Text Rendering:  
 - Simplified and improved the text\_to\_button function to ensure text is rendered correctly on buttons, regardless of hover state.  
   
4. Font Adjustments:  
 - Ensured fonts are properly initialized and corrected inconsistencies in text rendering logic.  
  
5. Gameplay Adjustments:  
 - Modified tank wheel rendering using a loop to reduce redundancy.  
 - Simplified the logic for turret positioning and ensured consistent alignment with tank dimensions.  
  
6. Randomness Handling:  
 - Fixed errors in the random.randint and random.randrange functions by ensuring their bounds are integers.  
  
7. Game Over and Win Screens:  
 - Adjusted logic to properly display the desert background instead of filling the screen with a solid color.

## Debugging and Troubleshooting

Several issues were identified and resolved during the modification process. Below is a summary of the steps taken and debugging statements added to identify and fix the problems:

1. Float to Integer Conversion (Line 305):  
 - Issue: random.randint and random.randrange were being passed float bounds.  
 - Fix: Converted bounds to integers using int().  
 - Debugging: Added print statements to log the bounds and ensure correctness.

python *print(f"xlocation bounds: {int(-0.1 \* display\_width)}, {int(0.1 \* display\_width)}")*   
  
2. Menu Text Visibility:  
 - Issue: Text on the control screen was not visible due to blending with the background.  
 - Fix: Adjusted colors and added a semi-transparent overlay for better contrast.  
 - Debugging: Verified button rendering using debug prints.  
 python *print(f"Rendering button '{msg}'")*  
   
  
3. Undefined Variables:  
 - Issue: desert\_background and menu\_background were referenced before initialization.  
 - Fix: Ensured these variables are globally defined at the start of the script.  
  
4. Game Loop Logic:  
 - Issue: Misaligned or redundant code caused crashes or incorrect rendering.  
 - Fix: Reorganized code for better readability and corrected the order of operations.  
 - Debugging: Used print statements to log tank positions and game state changes.

## Additional Debugging and Troubleshooting

5. Projectile Trajectory Not Displaying:  
 - Issue: The trajectory of the fired projectile was not visible on screen.  
 - Fix: Added a pygame.display.update() call within the fireShell loop to refresh the screen after each frame.  
 - Debugging: Used print statements to log projectile coordinates and ensured they were drawn correctly.  
 python *print(f"Projectile position: {startingShell[0]}, {startingShell[1]}")*  
   
  
6. Computer Tank Collision Detection:  
 - Issue: The computer tank was not taking damage when hit by a projectile.  
 - Fix: Adjusted the collision detection logic to align with the tank's dimensions and ensure accurate hit detection.  
 - Debugging: Added print statements to log the projectile's impact coordinates and confirmed damage calculations.  
 python *if enemyTankX - (tankWidth // 2) <= hit\_x <= enemyTankX + (tankWidth // 2): print("Critical Hit!") damage = 25*  
   
  
7. Control Button Action on Win Screen:  
 - Issue: Selecting Controls from the You Win screen redirected to the main menu instead of the controls screen.  
 - Fix: Corrected the action assigned to the Controls button to invoke the game\_controls() function.  
 - Debugging: Verified button actions with debug logs.  
 python *print(f"Button '{text}' action: {action}")*