Test Cases

**Test Case 1:** **Purchasing an Item Test Scenario: The player attempts to purchase an item from the shop.**

Test Steps:

1. Start the game.
2. Check the initial amount of money.
   * Expected Result: The initial amount of money is 0.
3. Click on the "Buy Item" button.
4. Check the updated amount of money.
   * Expected Result: The amount of money is decreased by 10 if it was greater than or equal to 10.
5. Repeat steps 3 and 4, but with an insufficient amount of money (e.g., less than 10).
   * Expected Result: The amount of money remains the same, and a message is displayed indicating insufficient funds.

This test case verifies that the player can successfully purchase an item from the shop when they have enough money and that the amount of money is appropriately deducted. It also checks the behavior when the player has an insufficient amount of money.

**Test Case 2: Cat Disappearance Test Scenario: A cat's core values fully deplete, causing the cat to disappear from the screen.**

Test Steps:

1. Start the game.
2. Observe the cats on the screen.
   * Expected Result: There are initially some cats visible on the screen.
3. Wait for a cat's core values (happiness, hydration, and energy) to reach 0.
4. Observe the cats on the screen again.
   * Expected Result: The cat with depleted core values has disappeared from the screen.

This test case verifies that when a cat's core values fully deplete, the cat is correctly removed from the screen.

**Test Case 3: Cat Interaction with Item Test Scenario: A cat collides with an item, replenishing its corresponding core value.**

Test Steps:

1. Start the game.
2. Place an item (e.g., yarn) on the screen.
3. Observe the cats and their core values.
   * Expected Result: The cats' core values are decreasing over time.
4. Move a cat to collide with the placed item.
5. Observe the cat's core values after the collision.
   * Expected Result: The corresponding core value of the cat (e.g., happiness) increases, indicating successful interaction with the item.

This test case ensures that when a cat collides with an item, the corresponding core value of the cat is replenished.

**Test Case 4: Money Increment Test Scenario: The player’s money increases continuously.**

Test Steps:

1. Start the game.
2. Check the initial amount of money.
   * Expected Result: The initial amount of money is 0.
3. Wait for a certain period of time.
4. Check the updated amount of money.
   * Expected Result: The amount of money has increased since the initial check.

This test case validates that the player's money increases continuously over time as specified in the game's requirements.

**Test Case 5: Random Appearance of Cats Test Scenario: Cats appear randomly on the screen.**

Test Steps:

1. Start the game.
2. Observe the cats on the screen.
   * Expected Result: Cats appear randomly at different positions on the screen.
3. Wait for some time.
4. Observe the cats again.
   * Expected Result: New cats appear randomly on the screen while existing cats remain.

This test case ensures that cats appear randomly on the screen and new cats continue to appear over time.

**Test Case 6: Limited Money Increase Test Scenario: The player's money increases continuously, but up to a certain limit.**

Test Steps:

1. Start the game.
2. Check the initial amount of money.
   * Expected Result: The initial amount of money is 0.
3. Wait for a certain period of time.
4. Check the updated amount of money.
   * Expected Result: The amount of money has increased since the initial check, but it should not exceed a predefined limit.

This test case verifies that the player's money increases continuously but does not exceed the predefined limit as specified in the game's requirements.

**Test Case 7: Multiple Item Purchases Test Scenario: The player purchases multiple items from the shop.**

Test Steps:

1. Start the game.
2. Check the initial amount of money.
   * Expected Result: The initial amount of money is 0.
3. Click on the "Buy Item" button multiple times, spending the required amount of money each time.
4. Check the updated amount of money.
   * Expected Result: The amount of money is correctly decreased with each purchase, reflecting the cost of the items.

This test case validates that the player can successfully purchase multiple items from the shop, and the amount of money is appropriately deducted for each purchase.