Cse121b

Final

1. \*\*Constants\*\*: These are variables that store values that remain constant throughout the game. In this case, `NUM\_MOLES` represents the total number of moles in the game, `GAME\_DURATION` represents the duration of the game in seconds, and `PENALTY\_SCORE` represents the score penalty for clicking on an empty space.

2. \*\*Variables\*\*: These are mutable values that store various states and data during the game. `score` keeps track of the player's score, `molesWhacked` tracks the number of moles successfully whacked, `gameStarted` indicates if the game has started, `gameIntervalId` stores the interval ID for the mole placement, and `timeLeft` keeps track of the remaining time in the game.

3. \*\*DOM Elements\*\*: These variables store references to HTML elements that are used in the game. `scoreElement` refers to the HTML element that displays the score, `timeLeftElement` refers to the element that displays the time left, and `moleContainer` represents the game container element.

4. \*\*startGame()\*\*: This function is called when the player clicks the "Start Game" button. It initializes or resets the game state by setting the score, molesWhacked, and timeLeft to their initial values, updating the score and timer display, clearing the moleContainer, starting the interval to place random moles, and scheduling the end of the game.

5. \*\*moleClickHandler()\*\*: This function handles the click event on moles. It first checks if the game is in progress (`gameStarted` is true), the clicked element has the class `mole`, and it has not already been whacked. If these conditions are met, the mole is considered successfully whacked, and the score is incremented, molesWhacked is increased, and the display is updated. The `whacked` class is added to the mole element, triggering the animation, and after a delay, the mole is removed from the container. Additionally, if the clicked element does not have the class `mole`, it means the player clicked on an empty space, and the score is decreased as a penalty.

6. Other functions like `updateScore()`, `updateTimer()`, and `endGame()` are used to update the score and timer display and handle the end of the game, respectively.

These modifications and explanations enhance the game by introducing a penalty for clicking on empty spaces, providing a more engaging gameplay experience.