**Documentation of SHOOTING GAME:**

1. Initialization on Window Load:

- The code begins with an event listener for the window.onload event.

- Displays a welcome message using the SweetAlert library, showing the username, last score, and selected level.

2. Timer and Start Button:

- Sets up a timer for the game with a duration of 120 seconds.

- Utilizes the setInterval function to update the timer every second.

- Listens for a click on an element with the class control-button, which triggers the removal of the button after a delay of 1000 milliseconds (1 second).

3. cavas Setup:

- Gets a id to the canvas element and its 2D context.

- Adjusts the canvas dimensions to match the inner dimensions of the window.

4. Score and Game Variables:

- Initializes variables related to the game state, including the score, time

- Defines the player's starting position and properties.

5. Player, Polit, and Enemy Classes (commented out):

- Contains class definitions for Player, Polit, and Enemy, each with properties and methods related to drawing and updating game elements. However, these classes are currently commented out and not in use in the code.

6. Object creation:

-creat a `Player` object named player1.

- Declares empty arrays for storing Polit and Enemy objects (arrpolit and arrenemies).

7. Update Timer Function:

- Defines a function (updateTimer) to update and display the game timer.

- If the timer reaches 0, it stops the timer and prompts the user to end or return to the home page.

8. Detect Enemy Function:

- Determines the level of difficulty based on the user's selected level.

- Periodically generates enemies and adds them to the arrenemies array.

9. Animation Function:

- Handles the animation loop for the game.

- Checks for victory conditions (score >= 300) and displays a victory message.

- Draws the player, updates and removes Polit objects, and checks for collisions between Polit and `Enemy` objects.

10. Event Listener for Keydown:

- Listens for keydown events and adjusts the player's position accordingly.

- If the spacebar is pressed, a Polit object is created and added to the arrpolit array.

11. Start Button Click Event:

- Listens for a click on the start button.

- Initiates the game by setting flags, starting the animation loop, detecting enemies, and updating the timer display.

Note:

- The code makes use of the SweetAlert library for displaying informative messages and confirmations to the user.

This documentation provides an overview of the code's structure and functionality.