F28WP Web Programming
Lab report
Lab 2

Student full name: Akshay Arunkumar Garg

Student HWU username: ag2006

Student's GitHub URL of the Lab: https://github.com/ag220502/Lab2BearGame

Demonstrated to Lab helper: Harshal Thachapully

Mode of demonstration: ☐ face-to-face ☐ online

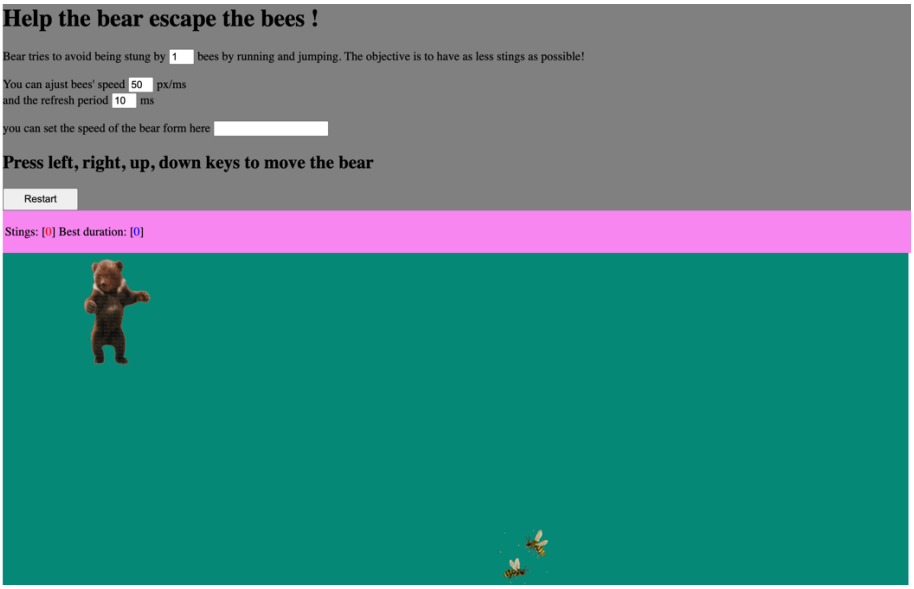
Date of demonstration: 2nd November 2021

Time of demonstration: 11:35 AM

The report should be about 3 pages long (excluding the title page) and submitted in PDF format. All explanations must be brief and supported by short code statements, and possibly illustrated by precise images captured from the screen (examples, the speed input field or the game over alert).

Part 1: **HTML/CSS** (**1 mark)**

Screen shot of the final web page of the game.



List any changes/additions/improvements you have done on HTML/CSS

I have added a restart button and an input field for the speed of the bear in the title div.
Also, I have changed text content of the best duration to 0 so that it can be a number

Part 2: **Bear motion (2 marks)**

Explain how you implemented the speed input functionality.

First, we are getting speed as a number on change means if the user changes value in input field we will get the value and then we are calling the setSpeed function for setting the speed of the bear and then we are setting bear variable to that value. So, the image will move accordingly.

Compare bear speed with bee speed. What is the difference in the implementation?

In bear speed, as there is only one bear we are not running for loop and then we are changing the position of the bear by the value in the variable. We are using offset property of CSS for which we are setting the value from left and the top and each time key is pressed we are adding the value to the previous value.
Whereas there are more bees, so we must run a for loop for setting their position. We are running for loop for all the bees present in array and then we are generating the random values according to the speed input so that we can move the bees to that position. And then we are calling the move function with the positions as parameters to move the bees from one place to another.

Part 3: **Bee creation (1 mark)**

Explain the use of DOM API to create <IMG> tag and add it to the game board.

We are creating the image of bees using the DOM API. In this function we are passing the number of the bees to be displayed. In this function, first we are getting details of the board like offset values so that bees cannot go beyond that value that is outside the board. Then we are creating the image element using DOM API. After that we are setting the attributes like source file of the image, its width and man more. Also, we are setting the position property of CSS as absolute and then appending the element to the board div inside the play area. After that we are generating the random values using the function for the width and height for random positioning of the image by setting the left and top values.

Explain the random coordinate generation.

In bee creation we are using random function for creating image bees. First, we are getting the width and height of the board (playing area) and then we are passing that value as max value for generating value between 0 and that value. Then we are setting the left and top properties of images using these randomly generated values.

Part 4: **Bee animation (2 marks)**

Explain the use of timer to periodically refresh the bee positions.

We are using timer in updateBees function to update the bee's position and calling the same function again and again according to the value entered for bee's speed by the user. And if the score reaches 1000, we are stopping the timer so that bees does not move and the update function also won't be called.

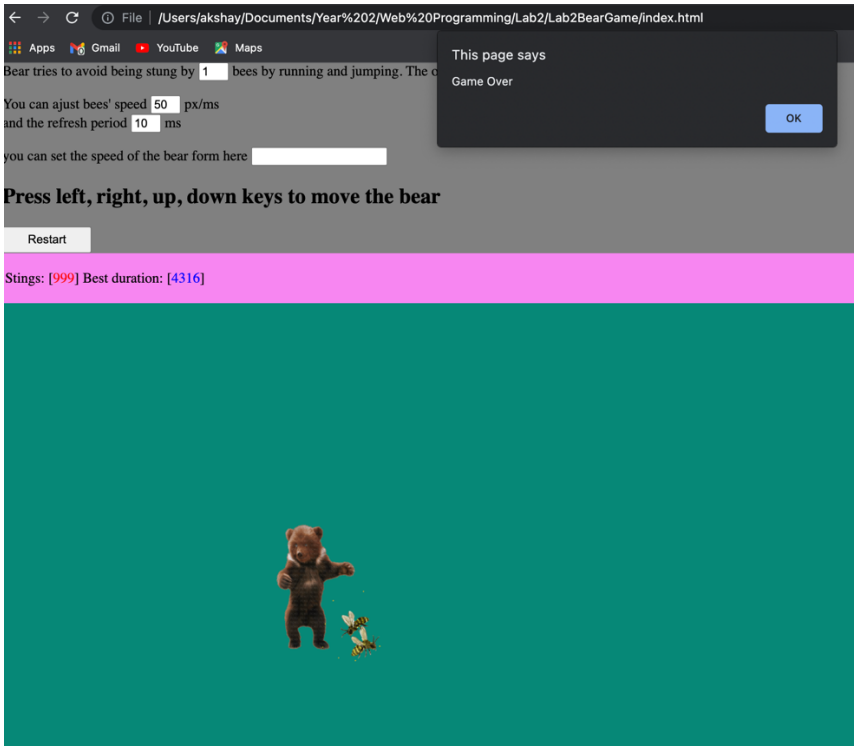
Part 5: **Scores (1 mark)**

Explain the method used to detect a sting and counting stings

We are doing this in isHit function by using overlap function to detect whether the image of bear and the bees are overlapped or has any same position. Then in overlap function we are getting the values of all four properties which are left, top, right and bottom of both the elements and then checking if they are intersecting or not. If they are not intersecting, then we are returning false and if they are intersecting then we are returning true. And if it returns true then we are incrementing the score by one and displaying it using DOM API.

Explain your implementation of the game over feature and insert a screen shot

For this, in the update bees function we are checking the score every time and if the score is 1000, then we are stopping the timer by giving an alert of Game Over. After this, bees stop moving as we are not calling the update function again and functionality of the game stops.



Part 6: **Best duration (1 mark)**

Explain your implementation of calculating the longest period without stings.

For calculating the longest period, first we are taking one temporary variable to check if the key is pressed or not. If the key is pressed, then we are we are initializing the last sting time with the value of time the key is pressed. And we are constantly updating the new Sting time when the bee stings the bear. After that we are calculating the difference and taking longest duration as the value stored in html content. After this, we are checking if the value if greater or not. If it is greater, then we are changing the longest duration as the new duration which is longer than the previous one. If it is not greater, then we are keeping the previous one as longest duration.

Part 7: **Additional features (2 marks)**

You need to describe only the features you implemented. Implementing correctly one feature is sufficient to get two marks).

- **Restarting the game**: *explain your implementation of the restart feature*
- ***Add a bee**: explain how you implemented adding a bee without stopping the game*
- **jQuery**: *give three examples of DOM API statements ad how you changed them to become jQuery operations*

Restarting the game:

For the additional feature, I have selected to add “Restart the Game” feature. For this, I have created one global variable and one restart function. In restart function, first we are setting global variable as true and then we are changing all the game parameters to default. For example, we are setting length of array of bees to 0 so that there will be no element in it, and they can be initialized again. Also, we are setting values to 0 for stings and longest time duration. Then, we are clearing time so that the previous bees stop, and then new bee can be initialised. Also, we are resetting the position of bear in the start of the game. And then we are calling the start function to start the game again.