2D BREAKOUT GAME

Author : Safa Bouhlel

Web developper Student

DCI Institut Hamburg,Germany

Email: **safa.bouhlel91@gmail.com**

General Info:

2D BREAKOUT GAME is a game where you should destroy all the bricks with a ball without letting this ball touch the ground.

Every time you start new game you will have three 3 lives where you must not let the ball touch the ground otherwise you will lose one life evrytime, once you got no life left you lose the game.

Time Management:

Start- Day: 20.04.2020

End-Day : 30.04.2020

Technologies

I have build this game using HTML5, canvas, CSS3, VANILLA JAVASCRIPT.

The design of the game consists of 4 pages:

- Start page where you can choose to start the game, get highest score or exit the game.
- Second page is basically the main page where you play and try to not let the ball touch the ground while destroying the bricks.
- Third page : it will show up if you win or lose, if you win you can save your name and will show your score , if you lose you get different window.
- Final page: this is the page that appears if you choose in first page to see the highest scores saved

Description of steps:

Start & end Page : **1Day**

- First Page is a start page where you will find the 5 highest scores and the start button to log to the game.

- End Page is to choose if you want to save your name with your score on the local storage and compare it to the others score on the lists of scores.

- Highscores Page is to show the 5 best scores with ascendant order

Game Page : **6 Days**

1. **Drawing with canvas:** 2/3 Days

   There is 2 types :normal image and animated image using setInterval.

   - using layout on making many canvas to clear without affecting others canvas elements.

   - *drawing the paddle*: I made a simple function to draw an image and added to it an event with the up and down key.

   - *drawing the ball:* For the ball there is a function used to draw a filled circle and this function is called each 20ms using setTimeout and cleared each time with changing the x and y by adding a step.

   - *drawing Bricks:* An array is used to store in differents rows mutiple underwater animals. It is 4 rows by 12 columns array.

   To draw each sort of animals multiple functions has been defined and implemented to be able for example to draw each 300ms a delphin with differents position and inclinisation which will create the illusion of having a moving Delphin and this approach is used also to create the shark.

2. **Making collision:** 3 Days

   *ballWallCollision

   *ballPaddleCollision

   *ballBricksCollision

The collision of the ball is done either with the wall , bricks or paddle.On this step we need to compare the x and y coordinate of the ball with the x and y coordinates of the paddle or the wall or the bricks.

The collision between the ball and the bricks was not easy especially with the animated bricks so i needed to stop the setInterval and clear the context.

3. **Score and lives:** 1 Day

   * need to make a global variables for lives and score.

   * add 10 to the score each time you destroye a ball.

   * loose 1 life from 3 lives each time the ball touch the ground.

Save the data : **1 Day**

- using local storage to save the data.
- get the last data of the local storage and compare it to the list on the highscore Page and add it to the highscores list .

Design and Style: **1 Day**

- Choose the same background for the 3 Pages: startPage, game and endPage.
- Add a simple effect to the buttons.
- Make the undermarine moving down on a click event.

Setup:

If you want to try this code on your computer, follow those steps:

```
Installation
1.open the terminal on your computer
2. select where you want to put your project using  cd ..
3. from the github repository slect the link and write it in this line:
git clone ../safa.js
4.when every thing is done pull the file with the command : git pull origin
master.
5.then open this folder on your computer and run the index.html file.
```