X7: Refactoring Code

A re-design of the Progressive Break-It game is underway.

Some features have not yet been implemented, and some

features that are implemented need more work to make the

code clearer.

Step 1 (10 pts)

Your job is to improve something. Here are some ideas:

- Maybe the board could be a container?

It could hold three text objects.

- addMessageText seems messy.

- Messages probably should not be embedded in the code.

Should they be global constants?

- The keyboard input adds 2 constants and 3 variables

to the global namespace. That's undesirable.

- Multiple key actions are combined in the two keyboard

event handlers. Can this be organized better?

- Can BW and BH be moved into a smaller scope?

What about level? What about bricks? It seems

brick-related functions need more encapsulation.

Suggestion: Pressing the left or right arrow key changes

the behavior of the paddle. This is currently implemented

with two global variables. Move those variables into the

Paddle scope and provide what is necessary to retain the

original functionality.

Step 2 (10 pts)

Now add a new feature. Here are some ideas:

- New levels after a certain number of paddle hits.

- A "score" object.

- A "combo" feature.

- Scoring.

- A "lives" object.

- Allow puck to exit play area.

- game\_over and game\_restart.

- Sounds. (Avoid CORS issues.)

- A paddle with side areas that affect the puck angle.

Suggestion: An IIFE could be used to create a new scope

in which the current score is stored. Export a function

that can be used by bricks (when hit by the puck) to add

to the score.

Step 3 (1 pt) [BONUS]

Improve or add something else.

---------

The code is currently divided into the following files:

pb.htm - contains the HTML and loads the scripts

Puck.js - code for the puck

Paddle.js - code for the paddle

pb1.js - most of the program is here

pb2.js - code for the bricks

Here is the call-structure for the code in pb1 and \*pb2:

load()

init()

create stage

game\_build

buildPuck()

buildPaddle()

buildWalls()

new\_wall()

assign board

buildBricks()

\* addLevel()

\* moveBricks()

\* newLevel()

\* buildBrick()

setControls()

set key listers

addMessageText()

game\_start

set Ticker FPS

add Ticker tick listener

handleKeyDown(e)

handleKeyUp(e)

game\_step(e)

game\_update()

puck.update()

paddle.update

checkPaddle()

\* checkBricks()

game\_render()

puck.render()

paddle.render()

stage.update()

---------