# JavaScript Basics – Canvas

The goal of this lab is to practice **canvas techniques**. Your task is to use canvas functions to draw the images below. Submit your solutions as .zip/.rar as homework:

## Write your name with glow

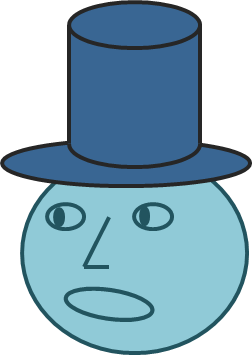
Your first task is to write your name with text glow as shown in the picture below.



## Draw a face using paths in canvas

Using beginPath(), moveTo(), lineTo(), arc() and stroke()/fill() try to visualize the following image:

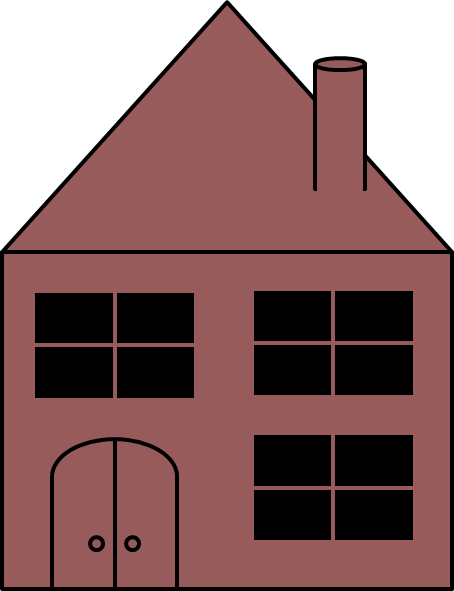
*Note:100% accuracy is not required.*



## Draw a house using canvas

Using beginPath(), moveTo(), lineTo(), fillRect(), arc() and stroke()/fill() try to visualize the following image:

*Note:100% accuracy is not required.*



# Animation:

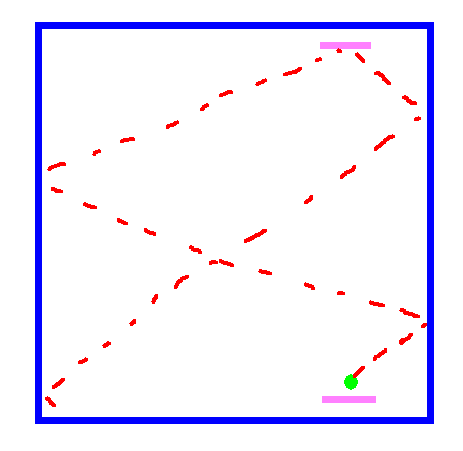
You are given a framework.js file to use plus an image for the paddle and the ball.

## Create Ball class that holds a ball object

Create a ball object that has the following properties:

* Position -> x, y
* Dimensions -> width, height
* Speed -> velocity
* Image -> a ball image
* Bounding box -> a rectangle

You should implement a simple ball movement logic that moves the ball diagonally. The ball should bounce off the canvas’ right and left border ( x == 0 and x == canvas.width). The ball should move in the canvas as the picture below:



## Create a Player paddle class that holds a rectangle object

Create a paddle object that has the following properties:

* Position -> x, y
* Dimensions -> width, height
* Speed -> velocity
* Image -> an image
* Bounding box -> a rectangle

## Create a second player paddle class

Create a paddle object that has the following properties. The second player’s position (x,y) are linked to the first player’s position. Meaning that by moving the first player’s paddle you move at the same time the second player’s paddle:

* Position -> x, y
* Dimensions -> width, height
* Speed -> velocity
* Image -> an image
* Bounding box -> a rectangle

## Create a main class

Create a main class that holds:

* The canvas object
* The canvas context
* Input handler

1. Create an instance of the first player class
2. Create an instance of the second player class
3. Create an instance of the ball
4. Link the input handler with the first and second player class. Meaning that by pressing ‘left’ and ‘right’ arrows as well as ‘d’ and ‘a’ buttons the first and second paddle should move in the same time.
5. Write intersection event logic. Meaning that when the first paddle hits the ball. The ball should bounce off the paddle.
6. Write a scoring logic that is triggered when the ball goes of the y coordinate. Meaning that when the ball’s ‘y’ is lower than 0 (goes out the canvas from the top), the player on the bottom scores a point. If the ball’s ‘y’ is bigger than the canvas’ height (goes out the canvas from the bottom), the player on the top scores a point.