Web Apps Development

• • •

Part II - Advanced Features
Canvas API

Homework Feedback

FORMAT YOUR CODE

Homework Feedback

- You cannot just name event handlers whatever you want (yet!).

```
window.addEventListener("deviceorientation", handleOrientation);
window.addEventListener("devicemotion", handleMotion);
htmlElement.addEventListener("click", handleClick);
```

Homework Feedback

- *processDeviceInformation* is a function of another object and not one of our classes!!!

```
(function() {
    var myStage = document.getElementById('main-stage');
    var colorizer = Object.create(Controller.Colorizer);
    myBall.init(myStage);
})();
```

```
var Model = {
    Colorizer: {
        view: null,
        init: function(stage) {
        },
    },
    DeviceMotionDemo: {
        view: null,
        init: function(stage) {
        },
        processDeviceInformation: function() {
        },
    },
};
```

```
    var Controller = {
        Colorizer: {
            model: null,
            init: function(stage) {
                this.model.Object.create(Model.Colorizer);
            },
        },
    };
```

Homework Feedback

- Folder Structure regarding the Dashboard:

```
|-- yourDashboard
|-- dashboard.html
|    |-- javascript
|        |-- appManager.js
|        |-- colorizer
|            |-- main.js
|            |-- model.js
|            |-- view.js
|            |-- controller.js
|        |-- screensaver
|            |-- main.js
|            |-- model.js
|            |-- view.js
|            |-- controller.js
```

Homework Feedback - App Manager

Instructions:

1. Copy the appManager.js file in your javascript folder; (https://github.com/leonardomra/web-apps-dev-course/blob/master/PE04-DeviceOrientation/html/appManager/appManager.js)

2. On your html, comment all the script files responsible for loading the "screensaver" and "colorizer", and other apps.

3. Load only the new file: <script type="text/javascript" src="javascript/appManager.js"></script>

4. Open the file, on its last line you need to type the path of the js files of both apps. Example: ...})
(["javascript/screensaver", "javascript/colorizer", "javascript/clock"]);

Canvas API

Canvas API - Introduction

Added in HTML5, the HTML <canvas> element can be used to draw graphics via scripting in JavaScript. For example, it can be used to draw graphs, make photo compositions, create animations, or even do real-time video processing or rendering.

Canvas API - Basics

```
<canvas></canvas>

this.element = document.createElement('canvas');

this.element.getContext('2d')
```

Canvas API - Context Examples
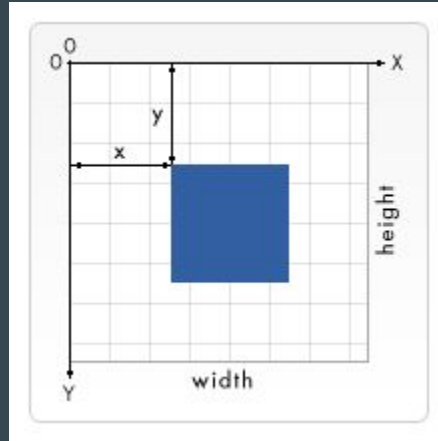
2D Context

http://codeincomplete.com/projects/racer/v4.final.html

http://breakout.enclavegames.com/lesson10.html

3D Context

http://stephaneginier.com/sculptgl/

https://playcanvas.com/

Canvas API - 2D Context Grid

Exercise

Implementing basic shapes.

Homework 1 - Clock App

1. Make a new folder called clock;
2. Inside this folder create 4 js files: main.js, model.js, view.js, controller.js;
3. Create a new object called Clock that draws an icon of a clock.
4. Display the clock icon within a widget of your dashboard.
5. Deadline: 18 December

Support

http://www.w3schools.com/html/html5_canvas.asp

https://developer.mozilla.org/en-US/docs/Web/API/Canvas_API/Tutorial/Drawing_shapes

https://developer.mozilla.org/en-US/docs/Web/API/Canvas_API/Tutorial/Applying_styles_and_colors

Homework 2 - Comment Puzzle

Based on the picture of the next slide and the following web page:
https://developer.mozilla.org/en-US/docs/Web/API/CanvasRenderingContext2D/drawImage
place the comments contained in the file
https://github.com/leonardomra/web-apps-dev-course/blob/master/PE05-
Canvas/html/animationHomework/javascript/model.js from line 136
inside of the function `drawOnCanvasContext` within `Model.Pacman`.

Deadline: 3 January.

Homework 3 - Implementation (Optional)

Based on the given code, implement the behaviour in red on the function pacmanFrameMap() of the file https://github.com/leonardomra/web-apps-dev-course/blob/master/PE05-Canvas/html/animationHomework/javascript/model.js

h: right, v: frozen
h: left, v: frozen
h: frozen, v: up
h: frozen, v: down
h: right, v: up
h: right, v: down
h: left, v: up
h: left, v: down

40px width

40px height

0   1   2   3   4   5   6   7   8   9   10   11   12   13   14   15   16

*40 = 0
*40 = 40
*40 = 80
*40 = 120
*40 = 160
*40 = 200
*40 = 240
*40 = 280
*40 = 320
...

Merry Christmas & Happy New Year!