Description of the "Super Mario"

Table of Contents

General description of the product

Product "Super Mario" - a realization on JavaScript, all well-known game from company Nintendo. In developing the library was used *JQuery*, information about library you can read on the site jquery.com [http://jquery.com/]. Image, for convenience, divided by type on the sprite-map.

Control is performed using the buttons on the keyboard and is given below in the table.

Table 1. Controls

Left	Arrow **Left**
Up	Arrow **Up**
Right	Arrow **Right**
Down	Arrow **Down**
Shot	Arrow **CTRL**

Implementation and screenshots of work

The whole logic of the product is implemented in a single file `main.js` , other information such as constants and levels for the game are stored in files: `constants.js` and `testlevels.js`.

About build-level in this game :

```
var definedLevels = [
{
 width: 10,
 height: 11,
 id: 0,
 background: 1,
 data:
 [
  ['' , '' , '' , '' , '' , '' , '' , '' , '' , 'grass_top' , 'soil'],
  ['' , '' , '' , '' , '' , '' , '' , '' , '' , 'grass_top' , 'soil'],
  ['' , '' , '' , '' , '' , '' , '' , '' , '' , 'grass_top' , 'soil'],
  ['' , '' , '' , '' , '' , '' , '' , '' , 'mario' , 'grass_top' , 'soil'],
  ['' , '' , '' , '' , '' , '' , '' , '' , '' , 'grass_top' , 'soil'],
  ['' , '' , '' , '' , '' , '' , '' , '' , '' , 'grass_top' , 'soil'],
  ['' , '' , '' , '' , '' , '' , '' , '' , '' , 'grass_top' , 'soil'],
  ['' , '' , '' , '' , '' , '' , '' , '' , 'gumpa' , 'grass_top' , 'soil'],
  ['' , '' , '' , '' , '' , '' , '' , '' , '' , 'grass_top' , 'soil'],
  ['' , '' , '' , '' , '' , '' , '' , '' , '' , 'grass_top' , 'soil']
 ]
```

All objects in the game have their own characteristics and look similar to:

```
var BrownBlock = Ground.extend({
    /**
     * @method init
     * @param x {number} coordinate X
     * @param y {number} coordinate Y
     * @param level
     */
    init: function(x, y, level) {
        var blocking = ground_blocking.all;
        this._super(x, y, blocking, level);
        this.setImage(images.objects, 514, 194);
    }
}, 'brown_block');
```

Below is a listing of file `keys.js` which contains a keyboard handler, without him it would be impossible to control your character.

```
/**
 * Keyboard
 * @module Keyboard
 * @type {{bind: Function, reset: Function, unbind: Function,
 * handler: Function, accelerate: boolean,
 * left: boolean, up: boolean, right: boolean, down: boolean}}
 * @class keys
 */
var keys = {
    /**
     * Connection with a Keyboard
     * @method bind
     * @return {handler} Keyboard handler
     */
    bind : function() {
        $(document).on('keydown', function(event) {
            return keys.handler(event, true);
        });
        $(document).on('keyup', function(event) {
            return keys.handler(event, false);
        });
    },
    /**
     * Reset
     * @method reset
     */
    reset : function() {
        keys.left = false;
        keys.right = false;
        keys.accelerate = false;
        keys.up = false;
        keys.down = false;
    },
    /**
     * Unbind
     * @method unbind
     */
    unbind : function() {
        $(document).off('keydown');
        $(document).off('keyup');
```

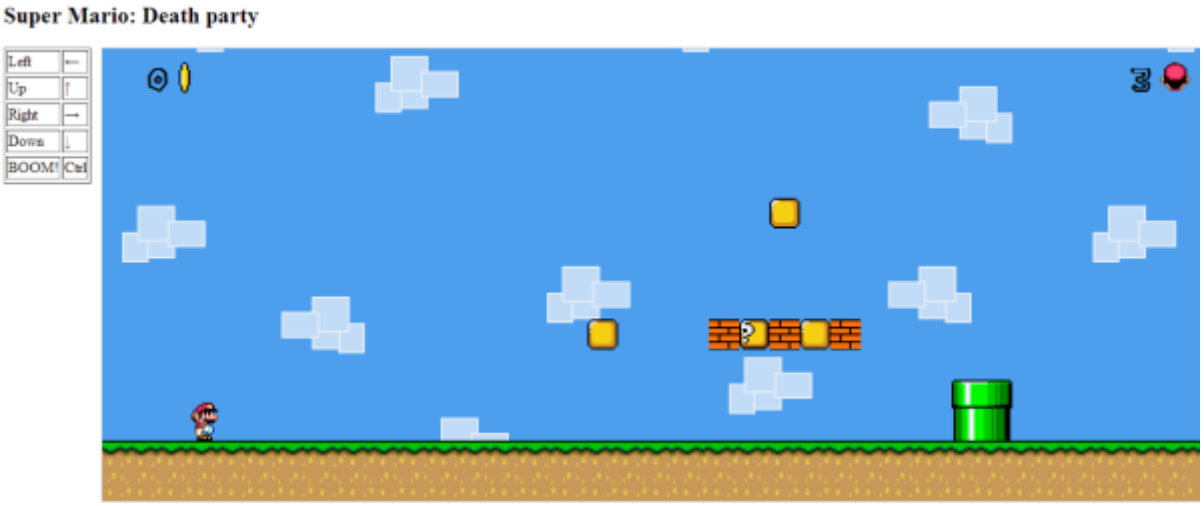
```
    },
    /**
     * @method handler
     * @param event
     * @param status
     * @returns {boolean}
     */
    handler : function(event, status) {
        switch(event.keyCode) {
            case 17://CTRL
                keys.accelerate = status;
                break;
            case 40://DOWN ARROW
                keys.down = status;
                break;
            case 39://RIGHT ARROW
                keys.right = status;
                break;
            case 37://LEFT ARROW
                keys.left = status;
                break;
            case 38://UP ARROW
                keys.up = status;
                break;
            default:
                return true;
        }

        event.preventDefault();
        return false;
    },
    accelerate : false,
    left : false,
    up : false,
    right : false,
    down : false
};
```

Main file `index.html` is pretty simple. Below is a screenshot of part of the code responsible for the connection of all files containing the game logic and handling the keyboard.

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>
<script>window.jQuery || document.write('<script src="Scripts/jQuery.js"><\/script>')</script>
<script src="Scripts/testlevels.js"></script>
<script src="Scripts/oop.js"></script>
<script src="Scripts/keys.js"></script>
<script src="Scripts/constants.js"></script>
<script src="Scripts/main.js"></script>
```

Here is a screenshot showing the first level of a running game:



Conclusions

Within a few weeks have developed a software product "Super Mario" using JavaScript, CSS, HTML. This is a great game for those who want to recollect the youth or just kill time. GLHF.