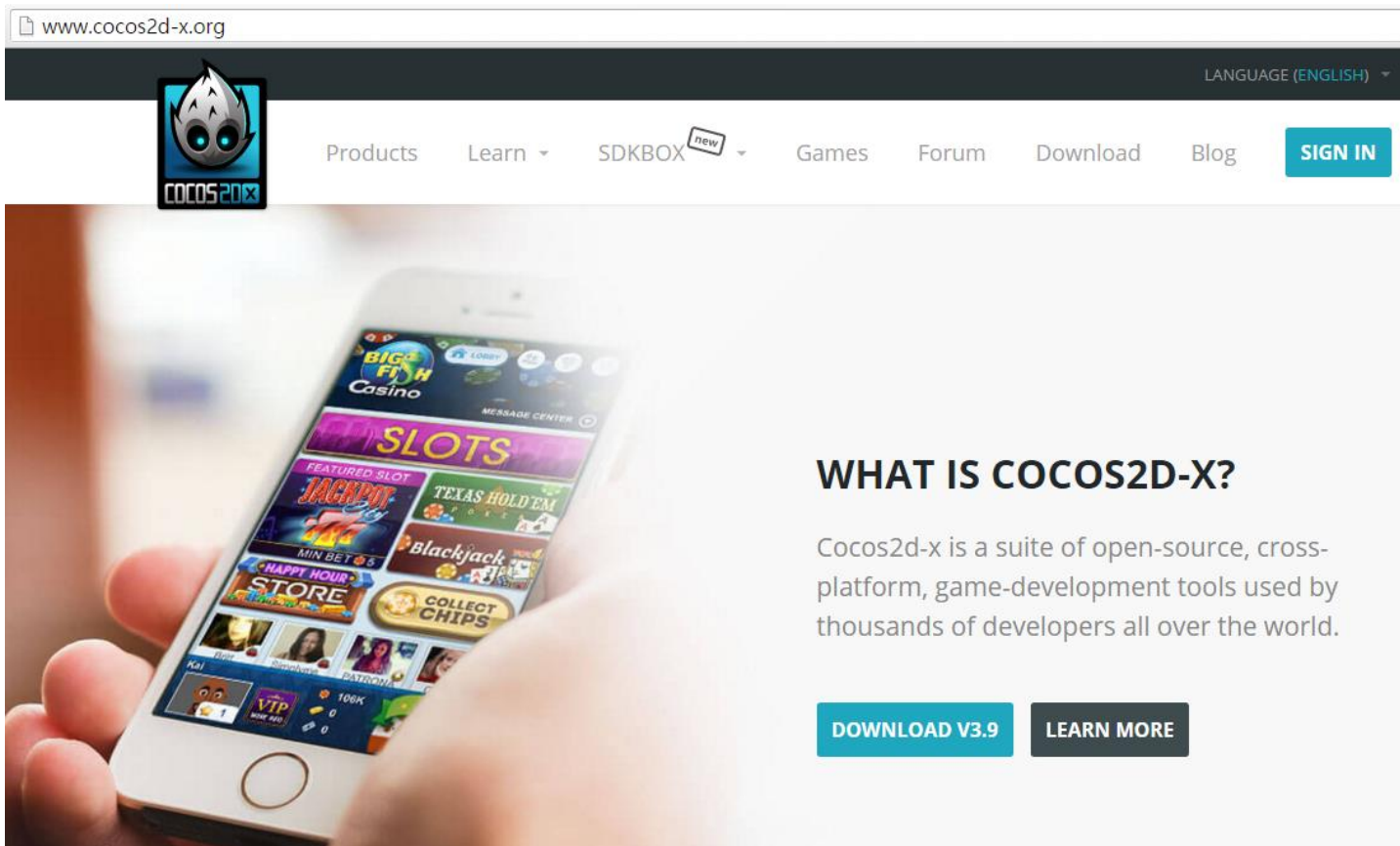


COCOS2D-JS INTRODUCTION

By Dannis Mok

Cocos2D-X

A well-known cross platform game engine



The screenshot shows the Cocos2D-X website. At the top is a browser address bar with 'www.cocos2d-x.org'. Below it is a dark navigation bar containing the Cocos2D-X logo, a 'LANGUAGE (ENGLISH)' dropdown, and links for 'Products', 'Learn', 'SDKBOX' (with a 'new' badge), 'Games', 'Forum', 'Download', 'Blog', and a 'SIGN IN' button. The main content area features a large image of a hand holding a smartphone displaying a colorful casino game interface with various slot and card game options. To the right of the image, the text 'WHAT IS COCOS2D-X?' is followed by a paragraph describing it as an open-source, cross-platform game-development tool. Below this text are two buttons: 'DOWNLOAD V3.9' and 'LEARN MORE'.

www.cocos2d-x.org

LANGUAGE (ENGLISH) ▾

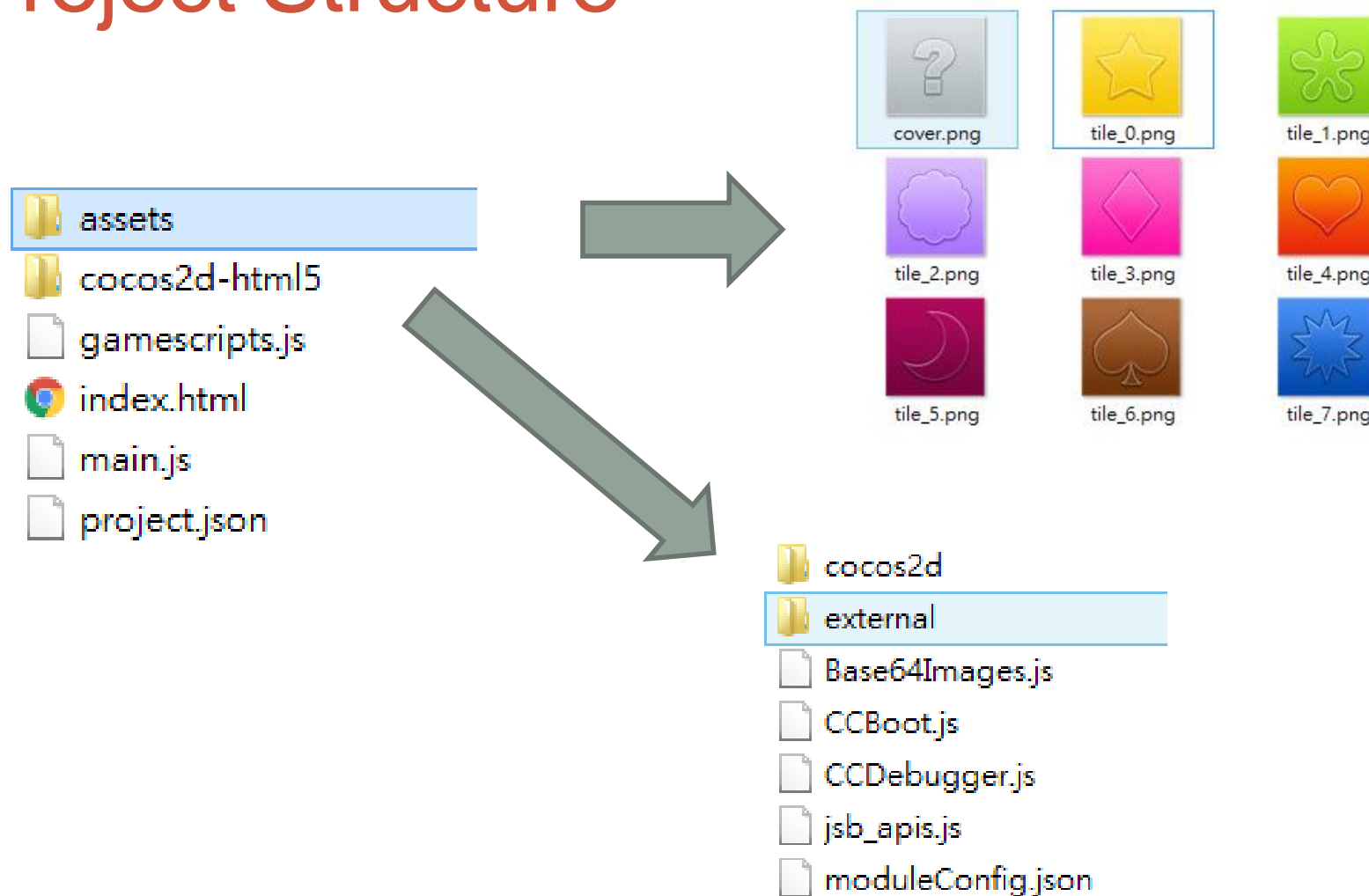
Products Learn ▾ SDKBOX ^{new} ▾ Games Forum Download Blog [SIGN IN](#)

WHAT IS COCOS2D-X?

Cocos2d-x is a suite of open-source, cross-platform, game-development tools used by thousands of developers all over the world.

[DOWNLOAD V3.9](#) [LEARN MORE](#)

Project Structure



Project Structure

File Name	Usage
index.html	Start page of the game, used to hold the canvas and load the Cocos2d libraries
main.js	Act as the delegate of the game, starting the game and call the director to load the first scene
project.json	A configuration file holding various game related information
gamescripts.js	Main Program, defining every resources used and the program codes.
assets	Contains the images, sounds and other resources
cocos2d-html5	The Cocos2d library

Sample Memory Game



Select 2 tiles and
they will be
disappeared when
matches

File Content – index.html

```
<head>
  <script src="cocos2d-html5/CCBoot.js"></script>
  <script src="main.js"></script>
</head>
<body>
  <canvas id="c1"></canvas>

</body>
```

Load the library

Load the delegate
main.js

Setup the canvas

File Content – project.json

```
{  
  "id": "c1",  
  "engineDir": "cocos2d-html5/",  
  "jsList": [  
    "gamescripts.js"  
  ]  
}
```

- 1) Locate the Canvas
- 2) Locate the library folder
- 3) Load the gamescripts.js in advance

File Content – main.js

```
cc.game.onStart = function() {  
    cc.view.setDesignResolutionSize(320,480,cc.ResolutionPolicy.SHOW_ALL);  
    cc.LoaderScene.preload(gameResources,function() {  
        cc.director.runScene(new myscene());  
    },this);  
};  
cc.game.run();
```

Set the canvas size

Let the director to load the first scene when the resource is loaded successfully

Run the game and call the onStart function

Cocos2d will utilize the director to load the scene. It will start the game.

File Content – gamescripts.js

```
var gameResources = [  
  "assets/cat.png",  
  "assets/cover.png",  
  "assets/tile_0.png",  
  "assets/tile_1.png",  
  "assets/tile_2.png",  
  "assets/tile_3.png",  
  "assets/tile_4.png",  
  "assets/tile_5.png",  
  "assets/tile_6.png",  
  "assets/tile_7.png"  
];
```

The image file names
are stored in the
array

The value assigned for
each tile

```
var gameArray = [  
  [0, 0, 1, 1],  
  [2, 2, 3, 3],  
  [4, 4, 5, 5],  
  [6, 6, 7, 7]  
];
```

```
var pickedTiles = [];
```

The array which contains
the picked tiles

File Content – gamescripts.js

```
var mytile = cc.Sprite.extend({  
  
    ctor: function() {  
  
        this._super();  
        this.initWithFile("assets/cover.png");  
  
        cc.eventManager.addListener(listener.clone(), this);  
    }  
  
});
```

Define a new object based on the Sprite class and then assign the cover image and an event listener to it. 16 mytiles will be created later and 16 event listeners will be assigned.

File Content – gamescripts.js

```
var background;  
  
var mylayer = cc.Layer.extend({  
  
    init:function() {  
        this._super();  
        background = cc.LayerColor.create(cc.color(0,0,256,128));  
  
        this.addChild(background);  
  
        for(var i=0; i<4; i++) {  
            for(var j=0; j<4; j++) {  
  
                tile = new mytile();  
                tile.value = gameArray[i][j];  
                tile.setPosition(i*80+40,j*80+150);  
                background.addChild(tile,0);  
            }  
        }  
    }  
});
```

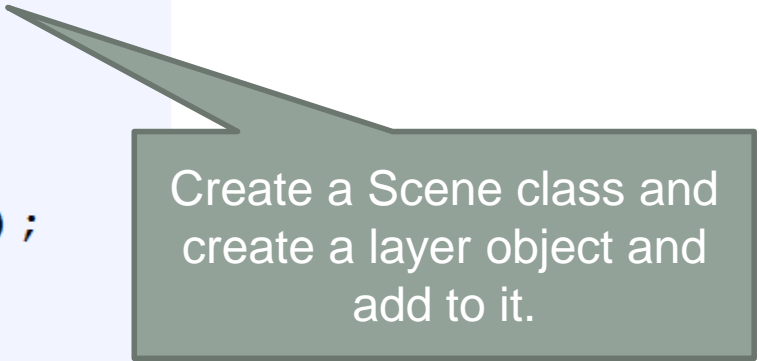
Create a new Layer class

Background color Layer

Create 16 tiles and then put onto the background layer

File Content – gamescripts.js

```
var myscene = cc.Scene.extend({  
    onEnter: function() {  
        this._super();  
        var layer1 = new mylayer();  
        layer1.init();  
        this.addChild(layer1);  
    }  
});
```



Create a Scene class and create a layer object and add to it.

File Content – gamescripts.js

```
var listener = cc.EventListener.create({  
    event:cc.EventListener.TOUCH_ONE_BY_ONE,  
    swallowTouches:true,  
    onTouchBegan: function(touch,event) {  
        if(pickedTiles.length < 2) {  
            var target = event.getCurrentTarget();  
            var location = target.convertToNodeSpace(touch.getLocation());  
            var targetSize = target.getContentSize();  
            var targetRect = cc.rect(0,0,targetSize.width,targetSize.height);
```

Create an event listener monitoring the touch event

Convert the touch location to the local coordinates relative to the target and check collision

File Content – gamescripts.js

```
if(cc.rectContainsPoint(targetRect, location)) {  
  
    if(pickedTiles.indexOf(target) == -1) {  
  
        console.log("I picked a tile");  
        target.initWithFile("assets/tile_" + target.value + ".png");  
  
        pickedTiles.push(target);  
        if(pickedTiles.length == 2) {  
  
            if(pickedTiles[0].value == pickedTiles[1].value) {  
  
                setTimeout(function() {  
  
                    background.removeChild(pickedTiles[0]);  
                    background.removeChild(pickedTiles[1]);  
                    pickedTiles = [];  
  
                }, 500);  
  
            }  
  
        }  
  
    }  
  
}
```

If hit inside the tile,
change its image

Compare the picked
tiles values

Remove from the
layer if matched
and empty the
array. Delay the
operation by
500ms

File Content – gamescripts.js

```
    } else {  
  
        setTimeout(function() {  
  
            pickedTiles[0].initWithFile("assets/cover.png");  
            pickedTiles[1].initWithFile("assets/cover.png");  
            pickedTiles = [];  
  
        }, 1000);  
    }  
}  
}  
}  
});
```

Change back to the cover image if not matched, let the player to memorize by 1000 ms