# class1绚丽的时钟效果

Canvas提供的函数

Canvas.width

Canvas.height

Canvas.getContext(‘2d’)

## canvas绘制基础

## 创建canvas

兼容性

<canvas id="canvas" style="border:1px solid #aaa;display:block;margin:50 auto;">

当前浏览器不支持Canvas

</canvas>

<script>

if(canvas.getContext('2d')){

var cantext = canvas.getContext('2d');

}else{

alert('当前浏览器不支持Canvas');

}

</script>

### 绘制直线、多边形

基于状态的绘图（先设定状态，后调用函数进行具体的绘制）

绘制直线

context.moveTo(100,100);

context.lineTo(700,700);

context.lineTo(100,700);

context.lineTo(100,100);

context.lineWidth = 5;

context.strokeStyle = "#005588";

context.stroke();

context.moveTo(100,100);

context.lineTo(700,700);

context.lineTo(100,700);

context.lineTo(100,100);

context.fillStyle = "rgb(2,100,300)";

context.fill();

context.lineWidth = 5;

context.strokeStyle = "red";

context.stroke();

基于状态的

context.moveTo(100,100);

context.lineTo(700,700);

context.lineTo(100,700);

context.lineTo(100,100);

context.lineWidth = 5;

context.strokeStyle = "red";

context.stroke();

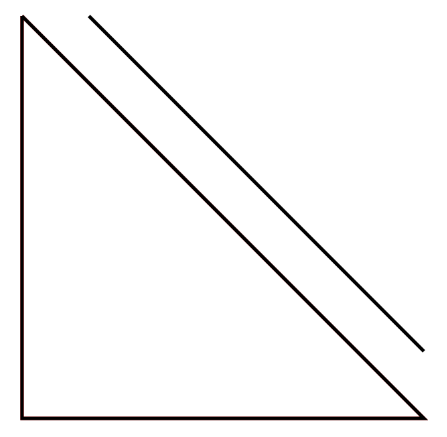
context.moveTo(200,100);

context.lineTo(700,600);

context.strokeStyle = "black";

context.stroke();

两段路径的颜色和宽度相同



多个路径分开处理

context.beginPath()

context.closePath

碰撞检测

实例：绘制七巧板

var tangram =[

{p:[{x:0,y:0},{x:800,y:0},{x:400,y:400}],color:"#caff67"},

{p:[{x:0,y:0},{x:400,y:400},{x:0,y:800}],color:"#67becf"},

{p:[{x:800,y:0},{x:800,y:400},{x:600,y:600},{x:600,y:200}],color:"#ef3d61"},

{p:[{x:600,y:200},{x:600,y:600},{x:400,y:400}],color:"#f9f51a"},

{p:[{x:400,y:400},{x:600,y:600},{x:400,y:800},{x:200,y:600}],color:"#a594c0"},

{p:[{x:200,y:600},{x:400,y:800},{x:0,y:800}],color:"#fa8ccc"},

{p:[{x:800,y:400},{x:800,y:800},{x:400,y:800}],color:"#f6ca29"}

]

window.onload = function(){

var canvas = document.getElementById('canvas');

canvas.width = 800;

canvas.height = 800;

var context = canvas.getContext('2d');

for(var i=0;i<tangram.length;i++){

draw(tangram[i],context);

}

function draw (piece ,cxt) {

cxt.beginPath();

cxt.moveTo(piece.p[0].x,piece.p[0].y);

for(var i =1;i<piece.p.length;i++)

cxt.lineTo(piece.p[i].x,piece.p[i].y);

cxt.closePath();

cxt.fillStyle = piece.color;

cxt.fill();

cxt.strokeStyle = "black";

cxt.lineWidth = 3;

cxt.stroke();

}

}

### 绘制弧线和圆

contex.arc(

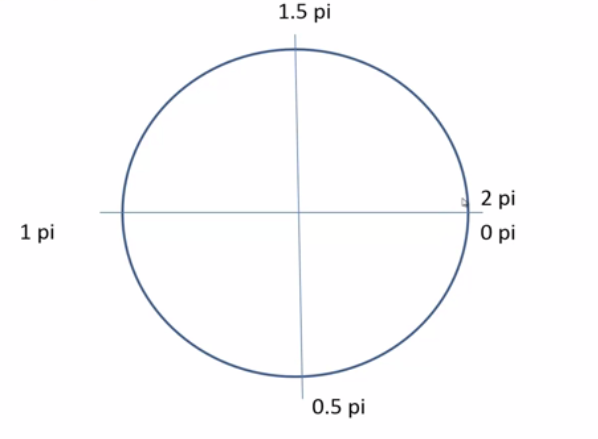
centerx,centery,radius,

startingAngle,endingAngle,

anticlochwise=false

)

无论是逆时针还是瞬时针，度数不变，如下：



var canvas = document.getElementById("canvas");

canvas.width=1024;

canvas.height=768;

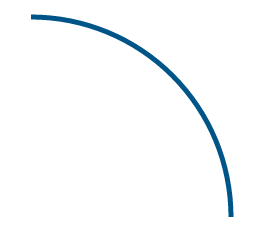
var context = canvas.getContext('2d');

context.lineWidth =5;

context.strokeStyle = "#005588";

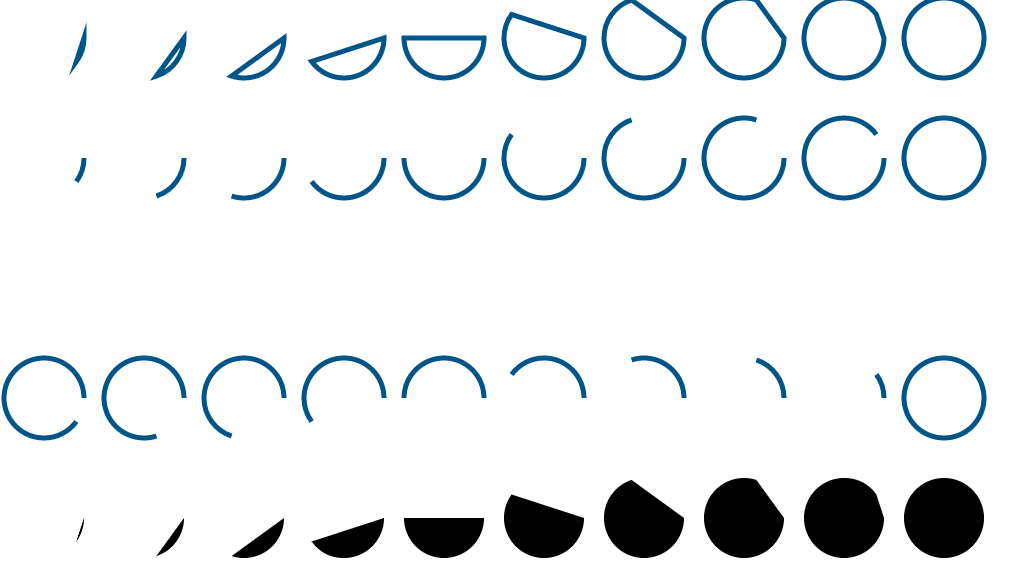
context.arc(300,300,200,0,1.5\*Math.PI,true);

context.stroke();



closePath:如果绘制的图形没有闭合，使用closePath会自动闭合所绘图形

beginPath和closePath不一定要重复出现



for(var i =0;i<10;i++){

context.beginPath();

context.arc(50+i\*100,60,40,0,2\*Math.PI\*(i+1)/10);

context.closePath();

context.stroke();

}

for(var i =0;i<10;i++){

context.beginPath();

context.arc(50+i\*100,180,40,0,2\*Math.PI\*(i+1)/10);

context.stroke();

}

for(var i =0;i<10;i++){

context.beginPath();

context.arc(50+i\*100,420,40,0,2\*Math.PI\*(i+1)/10,true);

context.stroke();

}

for(var i =0;i<10;i++){

context.beginPath();

context.arc(50+i\*100,540,40,0,2\*Math.PI\*(i+1)/10);

context.closePath();

context.fill();

}

如果没有closePath，就直接使用fill，调用fill的时候，canvas会把没有封闭的路径首尾相连

for(var i =0;i<10;i++){

context.beginPath();

context.arc(50+i\*100,730,40,0,2\*Math.PI\*(i+1)/10);

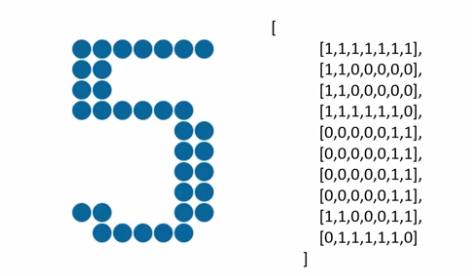
context.fill();

}

## 倒计时电子钟的实现

### 倒计时程序的基本架构

二维点阵模型



window.onload = function(){

var canvas = document.getElementById('canvas');

var context = canvas.getContext('2d');

canvas.width = WINDOW\_WINTH;

canvas.height = WINDOW\_HEIGHT;

render(context);

}

function render(cxt){

var hours =12;

var minutes =34;

var seconds =56;

renderDigit(0,0,parseInt(hours/10),cxt);//一定要传绘图上下文环境

}

function renderDigit(x,y,num,cxt){

cxt.fillStyle = "rgb(0,102,153)";

for(var i=0;i<digit[num].length;i++){

for(var j=0;j<digit[num][i].length;j++){

if(digit[num][i][j]==1){

cxt.beginPath();

cxt.arc();

cxt.closePath();

cxt.fill();

}

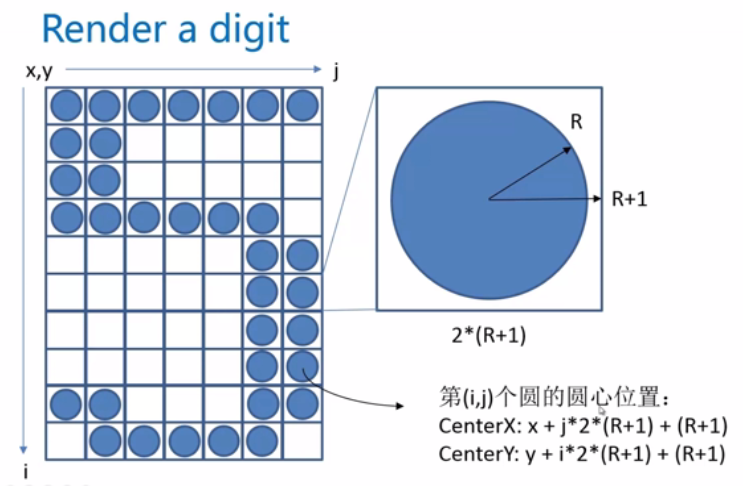
}

}

}

### 倒计时数字中的具体绘制

圆心位置的确定【格子系统】

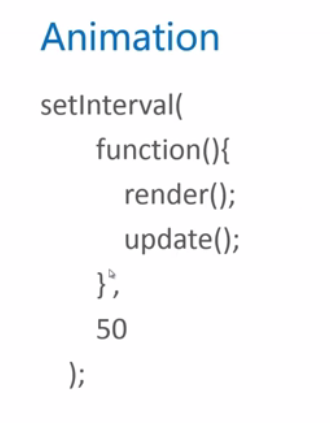


### 倒计时的时间计算

不用写死的时间

## 绚丽的动画效果

### 一个实现动画的基础模式



### 使用canvas做一个物理实验

<!DOCTYPE html>

<html>

<head>

<title></title>

</head>

<body>

<canvas id="canvas" style="border:1px solid #aaa;display:block;margin:50 auto;">

当前浏览器不支持Canvas

</canvas>

<script type="text/javascript">

var ball={x:512,y:100,r:20,g:2,vx:-4,vy:-10,color:"#005588"}

window.onload=function(){

var canvas = document.getElementById('canvas');

canvas.width = 1024;

canvas.height = 678;

var context = canvas.getContext('2d');

setInterval(function(){

render(context);

update();

},50);

}

function update(){

ball.x+=ball.vx;

ball.y+=ball.vy;

ball.vy+=ball.g;

}

function render(cxt){

cxt.clearRect(0,0,cxt.canvas.width,cxt.canvas.height);

cxt.fillStyle = ball.color;

cxt.beginPath();

cxt.arc(ball.x,ball.y,10,0,2\*Math.PI);

cxt.closePath();

cxt.fill();

}

</script>

小球掉出了屏幕

加入“碰撞检测”和”空气阻力“

<!DOCTYPE html>

<html>

<head>

<title></title>

</head>

<body>

<canvas id="canvas" style="border:1px solid #aaa;display:block;margin:50 auto;">

当前浏览器不支持Canvas

</canvas>

<script type="text/javascript">

var ball={x:512,y:100,r:20,g:2,vx:-4,vy:-10,color:"#005588"}

window.onload=function(){

var canvas = document.getElementById('canvas');

canvas.width = 1024;

canvas.height = 768;

var context = canvas.getContext('2d');

setInterval(function(){

render(context);

update();

},50);

}

function update(){

ball.x+=ball.vx;

ball.y+=ball.vy;

ball.vy+=ball.g;

if(ball.y>=768-ball.r){

ball.y = 768 -ball.r;

ball.vy=-ball.vy\*0.5;

}

}

function render(cxt){

cxt.clearRect(0,0,cxt.canvas.width,cxt.canvas.height);

cxt.fillStyle = ball.color;

cxt.beginPath();

cxt.arc(ball.x,ball.y,ball.r,0,2\*Math.PI);

cxt.closePath();

cxt.fill();

}

</script>

### 华丽的小球滚动效果

<!DOCTYPE html>

<html>

<head>

<title></title>

</head>

<body>

<canvas id="canvas" style="border:1px solid #aaa;display:block;margin:50 auto;">

当前浏览器不支持Canvas

</canvas>

<script type="text/javascript">

var WINDOW\_WIDTH =1024;

var WINDOW\_HEIGHT = 768;

var RADIUS =8;

var MARGIN\_TOP=60;

var MARGIN\_LEFT=30;

const endTime = new Date(2016,3,12,18,47,52);//设置截止日期2014-07

var curShowTimeSeconds = 0;

var balls = [];//生成小球

var colors = ["#33B5E5","#0099CC","#AA66CC","#9933CC","#99CC00","#669900","#FFBB33","#FF8800","#FF4444","#CC0000"];

digit =

[

[

[0,0,1,1,1,0,0],

[0,1,1,0,1,1,0],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[0,1,1,0,1,1,0],

[0,0,1,1,1,0,0]

],//0

[

[0,0,0,1,1,0,0],

[0,1,1,1,1,0,0],

[0,0,0,1,1,0,0],

[0,0,0,1,1,0,0],

[0,0,0,1,1,0,0],

[0,0,0,1,1,0,0],

[0,0,0,1,1,0,0],

[0,0,0,1,1,0,0],

[0,0,0,1,1,0,0],

[1,1,1,1,1,1,1]

],//1

[

[0,1,1,1,1,1,0],

[1,1,0,0,0,1,1],

[0,0,0,0,0,1,1],

[0,0,0,0,1,1,0],

[0,0,0,1,1,0,0],

[0,0,1,1,0,0,0],

[0,1,1,0,0,0,0],

[1,1,0,0,0,0,0],

[1,1,0,0,0,1,1],

[1,1,1,1,1,1,1]

],//2

[

[1,1,1,1,1,1,1],

[0,0,0,0,0,1,1],

[0,0,0,0,1,1,0],

[0,0,0,1,1,0,0],

[0,0,1,1,1,0,0],

[0,0,0,0,1,1,0],

[0,0,0,0,0,1,1],

[0,0,0,0,0,1,1],

[1,1,0,0,0,1,1],

[0,1,1,1,1,1,0]

],//3

[

[0,0,0,0,1,1,0],

[0,0,0,1,1,1,0],

[0,0,1,1,1,1,0],

[0,1,1,0,1,1,0],

[1,1,0,0,1,1,0],

[1,1,1,1,1,1,1],

[0,0,0,0,1,1,0],

[0,0,0,0,1,1,0],

[0,0,0,0,1,1,0],

[0,0,0,1,1,1,1]

],//4

[

[1,1,1,1,1,1,1],

[1,1,0,0,0,0,0],

[1,1,0,0,0,0,0],

[1,1,1,1,1,1,0],

[0,0,0,0,0,1,1],

[0,0,0,0,0,1,1],

[0,0,0,0,0,1,1],

[0,0,0,0,0,1,1],

[1,1,0,0,0,1,1],

[0,1,1,1,1,1,0]

],//5

[

[0,0,0,0,1,1,0],

[0,0,1,1,0,0,0],

[0,1,1,0,0,0,0],

[1,1,0,0,0,0,0],

[1,1,0,1,1,1,0],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[0,1,1,1,1,1,0]

],//6

[

[1,1,1,1,1,1,1],

[1,1,0,0,0,1,1],

[0,0,0,0,1,1,0],

[0,0,0,0,1,1,0],

[0,0,0,1,1,0,0],

[0,0,0,1,1,0,0],

[0,0,1,1,0,0,0],

[0,0,1,1,0,0,0],

[0,0,1,1,0,0,0],

[0,0,1,1,0,0,0]

],//7

[

[0,1,1,1,1,1,0],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[0,1,1,1,1,1,0],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[0,1,1,1,1,1,0]

],//8

[

[0,1,1,1,1,1,0],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[0,1,1,1,0,1,1],

[0,0,0,0,0,1,1],

[0,0,0,0,0,1,1],

[0,0,0,0,1,1,0],

[0,0,0,1,1,0,0],

[0,1,1,0,0,0,0]

],//9

[

[0,0,0,0],

[0,0,0,0],

[0,1,1,0],

[0,1,1,0],

[0,0,0,0],

[0,0,0,0],

[0,1,1,0],

[0,1,1,0],

[0,0,0,0],

[0,0,0,0]

]//:

];

window.onload = function(){

var canvas = document.getElementById('canvas');

var context = canvas.getContext('2d');

canvas.width = WINDOW\_WIDTH;

canvas.height = WINDOW\_HEIGHT;

curShowTimeSeconds = getCurrentShowTimeSeconds();

setInterval(function(){

render(context);

update();//对当前数据进行调整

},50)

}

function getCurrentShowTimeSeconds(){

var curTime = new Date();

var ret =endTime.getTime() - curTime.getTime();

ret = Math.round(ret/1000);

return ret>0?ret:0;

}

function update(){

var nextShowTimeSeconds = getCurrentShowTimeSeconds();

var nextHours = parseInt(nextShowTimeSeconds/3600);

var nextMinutes =parseInt((nextShowTimeSeconds-nextHours\*3600)/60);

var nextSeconds =parseInt(nextShowTimeSeconds%60);

var curHours =parseInt(curShowTimeSeconds/3600);

var curMinutes =parseInt((curShowTimeSeconds-curHours\*3600)/60);

var curSeconds =parseInt(curShowTimeSeconds%60);

if(nextSeconds!=curSeconds){

if(parseInt(curHours/10)!=parseInt(nextHours/10)){

addBalls(MARGIN\_LEFT+0,MARGIN\_TOP,parseInt(curHours/10));

}

if(parseInt(curHours%10)!=parseInt(nextHours%10)){

addBalls(MARGIN\_LEFT+15\*(RADIUS+1),MARGIN\_TOP,parseInt(curHours%10));

}

if(parseInt(curMinutes/10)!=parseInt(nextMinutes/10)){

addBalls(MARGIN\_LEFT+39\*(RADIUS+1),MARGIN\_TOP,parseInt(curMinutes/10));

}

if(parseInt(curMinutes%10)!=parseInt(nextMinutes%10)){

addBalls(MARGIN\_LEFT+54\*(RADIUS+1),MARGIN\_TOP,parseInt(curMinutes%10));

}

if(parseInt(curSeconds/10)!=parseInt(nextSeconds/10)){

addBalls(MARGIN\_LEFT+78\*(RADIUS+1),MARGIN\_TOP,parseInt(curSeconds/10));

}

if(parseInt(curSeconds%10)!=parseInt(nextSeconds%10)){

addBalls(MARGIN\_LEFT+93\*(RADIUS+1),MARGIN\_TOP,parseInt(curSeconds%10));

}

curShowTimeSeconds = nextShowTimeSeconds;

}

updateBalls();

}

function updateBalls () {

for(var i=0;i<balls.length;i++){

balls[i].x+=balls[i].vx;

balls[i].y+=balls[i].vy;

balls[i].vy+=balls[i].g;

if(balls[i].y>=WINDOW\_HEIGHT-RADIUS){

balls[i].y=WINDOW\_HEIGHT-RADIUS;

balls[i].vy = -balls[i].vy\*0.75;

}

}

}

function addBalls(x,y,num){

for(var i=0;i<digit[num].length;i++){

for(var j=0;j<digit[num][i].length;j++){

if(digit[num][i][j]==1){

var aBall={

x:x+j\*2\*(RADIUS+1)+(RADIUS+1),

y:y+i\*2\*(RADIUS+1)+(RADIUS+1),

g:1.5+Math.random(),

vx:Math.pow(-1,Math.ceil(Math.random()\*1000))\*4,

vy:-5,

colors:colors[Math.floor(Math.random()\*colors.length)]

}

balls.push(aBall);

}

}

}

}

function render(cxt){

cxt.clearRect(0,0,WINDOW\_WIDTH,WINDOW\_HEIGHT);//清空屏幕

var hours =parseInt(curShowTimeSeconds/3600);

var minutes =parseInt((curShowTimeSeconds-hours\*3600)/60);

var seconds =parseInt(curShowTimeSeconds%60);

renderDigit(MARGIN\_LEFT,MARGIN\_TOP,parseInt(hours/10),cxt);//一定要传绘图上下文环境

renderDigit(MARGIN\_LEFT+15\*(RADIUS+1),MARGIN\_TOP,parseInt(hours%10),cxt);

renderDigit(MARGIN\_LEFT+30\*(RADIUS+1),MARGIN\_TOP,10,cxt);

renderDigit(MARGIN\_LEFT+39\*(RADIUS+1),MARGIN\_TOP,parseInt(minutes/10),cxt);

renderDigit(MARGIN\_LEFT+54\*(RADIUS+1),MARGIN\_TOP,parseInt(minutes%10),cxt);

renderDigit(MARGIN\_LEFT+69\*(RADIUS+1),MARGIN\_TOP,10,cxt);

renderDigit(MARGIN\_LEFT+78\*(RADIUS+1),MARGIN\_TOP,parseInt(seconds/10),cxt);

renderDigit(MARGIN\_LEFT+93\*(RADIUS+1),MARGIN\_TOP,parseInt(seconds%10),cxt);

for(var i=0;i<balls.length;i++){

cxt.fillStyle=balls[i].colors;

cxt.beginPath();

cxt.arc(balls[i].x,balls[i].y,RADIUS,0,2\*Math.PI,true);

cxt.closePath();

cxt.fill();

}

}

function renderDigit(x,y,num,cxt){

cxt.fillStyle = "rgb(0,102,153)";

for(var i=0;i<digit[num].length;i++){

for(var j=0;j<digit[num][i].length;j++){

if(digit[num][i][j]==1){

cxt.beginPath();

cxt.arc(x+j\*2\*(RADIUS+1)+(RADIUS+1),y+i\*2\*(RADIUS+1)+(RADIUS+1),RADIUS,0,2\*Math.PI);

cxt.closePath();

cxt.fill();

}

}

}

}

</script>

</body>

</html>

## 优化、扩展

### 性能优化

存在问题：

balls数组越来越长;

解决方法：

让滚出屏幕的小球从数组删除

### 屏幕自适应

记得把body和canvas的height设置为100%，使其撑开。

<!DOCTYPE html>

<html>

<head>

<title></title>

</head>

<body style="height:100%">

<canvas id="canvas" style="height:100%">

当前浏览器不支持Canvas

</canvas>

<script type="text/javascript">

var WINDOW\_WIDTH=1024;

var WINDOW\_HEIGHT=768;

var RADIUS =8;

var MARGIN\_TOP=60;

var MARGIN\_LEFT=30;

const endTime = new Date(2016,3,12,18,47,52);//设置截止日期2014-07

var curShowTimeSeconds = 0;

var balls = [];//生成小球

var colors = ["#33B5E5","#0099CC","#AA66CC","#9933CC","#99CC00","#669900","#FFBB33","#FF8800","#FF4444","#CC0000"];

digit =

[

[

[0,0,1,1,1,0,0],

[0,1,1,0,1,1,0],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[0,1,1,0,1,1,0],

[0,0,1,1,1,0,0]

],//0

[

[0,0,0,1,1,0,0],

[0,1,1,1,1,0,0],

[0,0,0,1,1,0,0],

[0,0,0,1,1,0,0],

[0,0,0,1,1,0,0],

[0,0,0,1,1,0,0],

[0,0,0,1,1,0,0],

[0,0,0,1,1,0,0],

[0,0,0,1,1,0,0],

[1,1,1,1,1,1,1]

],//1

[

[0,1,1,1,1,1,0],

[1,1,0,0,0,1,1],

[0,0,0,0,0,1,1],

[0,0,0,0,1,1,0],

[0,0,0,1,1,0,0],

[0,0,1,1,0,0,0],

[0,1,1,0,0,0,0],

[1,1,0,0,0,0,0],

[1,1,0,0,0,1,1],

[1,1,1,1,1,1,1]

],//2

[

[1,1,1,1,1,1,1],

[0,0,0,0,0,1,1],

[0,0,0,0,1,1,0],

[0,0,0,1,1,0,0],

[0,0,1,1,1,0,0],

[0,0,0,0,1,1,0],

[0,0,0,0,0,1,1],

[0,0,0,0,0,1,1],

[1,1,0,0,0,1,1],

[0,1,1,1,1,1,0]

],//3

[

[0,0,0,0,1,1,0],

[0,0,0,1,1,1,0],

[0,0,1,1,1,1,0],

[0,1,1,0,1,1,0],

[1,1,0,0,1,1,0],

[1,1,1,1,1,1,1],

[0,0,0,0,1,1,0],

[0,0,0,0,1,1,0],

[0,0,0,0,1,1,0],

[0,0,0,1,1,1,1]

],//4

[

[1,1,1,1,1,1,1],

[1,1,0,0,0,0,0],

[1,1,0,0,0,0,0],

[1,1,1,1,1,1,0],

[0,0,0,0,0,1,1],

[0,0,0,0,0,1,1],

[0,0,0,0,0,1,1],

[0,0,0,0,0,1,1],

[1,1,0,0,0,1,1],

[0,1,1,1,1,1,0]

],//5

[

[0,0,0,0,1,1,0],

[0,0,1,1,0,0,0],

[0,1,1,0,0,0,0],

[1,1,0,0,0,0,0],

[1,1,0,1,1,1,0],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[0,1,1,1,1,1,0]

],//6

[

[1,1,1,1,1,1,1],

[1,1,0,0,0,1,1],

[0,0,0,0,1,1,0],

[0,0,0,0,1,1,0],

[0,0,0,1,1,0,0],

[0,0,0,1,1,0,0],

[0,0,1,1,0,0,0],

[0,0,1,1,0,0,0],

[0,0,1,1,0,0,0],

[0,0,1,1,0,0,0]

],//7

[

[0,1,1,1,1,1,0],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[0,1,1,1,1,1,0],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[0,1,1,1,1,1,0]

],//8

[

[0,1,1,1,1,1,0],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[1,1,0,0,0,1,1],

[0,1,1,1,0,1,1],

[0,0,0,0,0,1,1],

[0,0,0,0,0,1,1],

[0,0,0,0,1,1,0],

[0,0,0,1,1,0,0],

[0,1,1,0,0,0,0]

],//9

[

[0,0,0,0],

[0,0,0,0],

[0,1,1,0],

[0,1,1,0],

[0,0,0,0],

[0,0,0,0],

[0,1,1,0],

[0,1,1,0],

[0,0,0,0],

[0,0,0,0]

]//:

];

window.onload = function(){

WINDOW\_WIDTH =document.documentElement.clientWidth||document.body.clientWidth;

WINDOW\_HEIGHT = document.documentElement.clientHeight||document.body.clientHeight;

MARGIN\_LEFT=Math.round(WINDOW\_WIDTH/10);

RADIUS =Math.round(WINDOW\_WIDTH\*4/5/108)-1;

MARGIN\_TOP=Math.round(WINDOW\_HEIGHT/5);

canvas = document.getElementById('canvas');

context = canvas.getContext('2d');

canvas.width = WINDOW\_WIDTH;

canvas.height = WINDOW\_HEIGHT;

curShowTimeSeconds = getCurrentShowTimeSeconds();

setInterval(function(){

render(context);

update();//对当前数据进行调整

},50)

}

function getCurrentShowTimeSeconds(){

var curTime = new Date();

var ret =endTime.getTime() - curTime.getTime();

ret = Math.round(ret/1000);

return ret>0?ret:0;

}

function update(){

var nextShowTimeSeconds = getCurrentShowTimeSeconds();

var nextHours = parseInt(nextShowTimeSeconds/3600);

var nextMinutes =parseInt((nextShowTimeSeconds-nextHours\*3600)/60);

var nextSeconds =parseInt(nextShowTimeSeconds%60);

var curHours =parseInt(curShowTimeSeconds/3600);

var curMinutes =parseInt((curShowTimeSeconds-curHours\*3600)/60);

var curSeconds =parseInt(curShowTimeSeconds%60);

if(nextSeconds!=curSeconds){

if(parseInt(curHours/10)!=parseInt(nextHours/10)){

addBalls(MARGIN\_LEFT+0,MARGIN\_TOP,parseInt(curHours/10));

}

if(parseInt(curHours%10)!=parseInt(nextHours%10)){

addBalls(MARGIN\_LEFT+15\*(RADIUS+1),MARGIN\_TOP,parseInt(curHours%10));

}

if(parseInt(curMinutes/10)!=parseInt(nextMinutes/10)){

addBalls(MARGIN\_LEFT+39\*(RADIUS+1),MARGIN\_TOP,parseInt(curMinutes/10));

}

if(parseInt(curMinutes%10)!=parseInt(nextMinutes%10)){

addBalls(MARGIN\_LEFT+54\*(RADIUS+1),MARGIN\_TOP,parseInt(curMinutes%10));

}

if(parseInt(curSeconds/10)!=parseInt(nextSeconds/10)){

addBalls(MARGIN\_LEFT+78\*(RADIUS+1),MARGIN\_TOP,parseInt(curSeconds/10));

}

if(parseInt(curSeconds%10)!=parseInt(nextSeconds%10)){

addBalls(MARGIN\_LEFT+93\*(RADIUS+1),MARGIN\_TOP,parseInt(curSeconds%10));

}

curShowTimeSeconds = nextShowTimeSeconds;

}

updateBalls();

}

function updateBalls () {

for(var i=0;i<balls.length;i++){

balls[i].x+=balls[i].vx;

balls[i].y+=balls[i].vy;

balls[i].vy+=balls[i].g;

if(balls[i].y>=WINDOW\_HEIGHT-RADIUS){

balls[i].y=WINDOW\_HEIGHT-RADIUS;

balls[i].vy = -balls[i].vy\*0.75;

}

}

var count = 0;

for(var i=0;i<balls.length;i++)

if(balls[i].x+RADIUS>0&&balls[i].x-RADIUS<WINDOW\_WIDTH)

balls[count++] = balls[i];

while(balls.length>count){

balls.pop();

}

}

function addBalls(x,y,num){

for(var i=0;i<digit[num].length;i++){

for(var j=0;j<digit[num][i].length;j++){

if(digit[num][i][j]==1){

var aBall={

x:x+j\*2\*(RADIUS+1)+(RADIUS+1),

y:y+i\*2\*(RADIUS+1)+(RADIUS+1),

g:1.5+Math.random(),

vx:Math.pow(-1,Math.ceil(Math.random()\*1000))\*4,

vy:-5,

colors:colors[Math.floor(Math.random()\*colors.length)]

}

balls.push(aBall);

}

}

}

}

function render(cxt){

cxt.clearRect(0,0,WINDOW\_WIDTH,WINDOW\_HEIGHT);//清空屏幕

var hours =parseInt(curShowTimeSeconds/3600);

var minutes =parseInt((curShowTimeSeconds-hours\*3600)/60);

var seconds =parseInt(curShowTimeSeconds%60);

renderDigit(MARGIN\_LEFT,MARGIN\_TOP,parseInt(hours/10),cxt);//一定要传绘图上下文环境

renderDigit(MARGIN\_LEFT+15\*(RADIUS+1),MARGIN\_TOP,parseInt(hours%10),cxt);

renderDigit(MARGIN\_LEFT+30\*(RADIUS+1),MARGIN\_TOP,10,cxt);

renderDigit(MARGIN\_LEFT+39\*(RADIUS+1),MARGIN\_TOP,parseInt(minutes/10),cxt);

renderDigit(MARGIN\_LEFT+54\*(RADIUS+1),MARGIN\_TOP,parseInt(minutes%10),cxt);

renderDigit(MARGIN\_LEFT+69\*(RADIUS+1),MARGIN\_TOP,10,cxt);

renderDigit(MARGIN\_LEFT+78\*(RADIUS+1),MARGIN\_TOP,parseInt(seconds/10),cxt);

renderDigit(MARGIN\_LEFT+93\*(RADIUS+1),MARGIN\_TOP,parseInt(seconds%10),cxt);

for(var i=0;i<balls.length;i++){

cxt.fillStyle=balls[i].colors;

cxt.beginPath();

cxt.arc(balls[i].x,balls[i].y,RADIUS,0,2\*Math.PI,true);

cxt.closePath();

cxt.fill();

}

}

function renderDigit(x,y,num,cxt){

cxt.fillStyle = "rgb(0,102,153)";

for(var i=0;i<digit[num].length;i++){

for(var j=0;j<digit[num][i].length;j++){

if(digit[num][i][j]==1){

cxt.beginPath();

cxt.arc(x+j\*2\*(RADIUS+1)+(RADIUS+1),y+i\*2\*(RADIUS+1)+(RADIUS+1),RADIUS,0,2\*Math.PI);

cxt.closePath();

cxt.fill();

}

}

}

}

</script>

</body>

</html>

### 改进成时间管理小工具

getTime()，获取距离1970-01-01 零点的毫秒数

var endTime = new Date();

endTime.setTime(endTime.getTime()+3600\*1000);//当前时间向后退1个小时

### 改进为绚丽的时钟

function getCurrentShowTimeSeconds(){

var curTime = new Date();

var ret = curTime.getHours()\*3600+curTime.getMinutes()\*60+curTime.getSeconds();

return ret;

//var ret =endTime.getTime() - curTime.getTime();

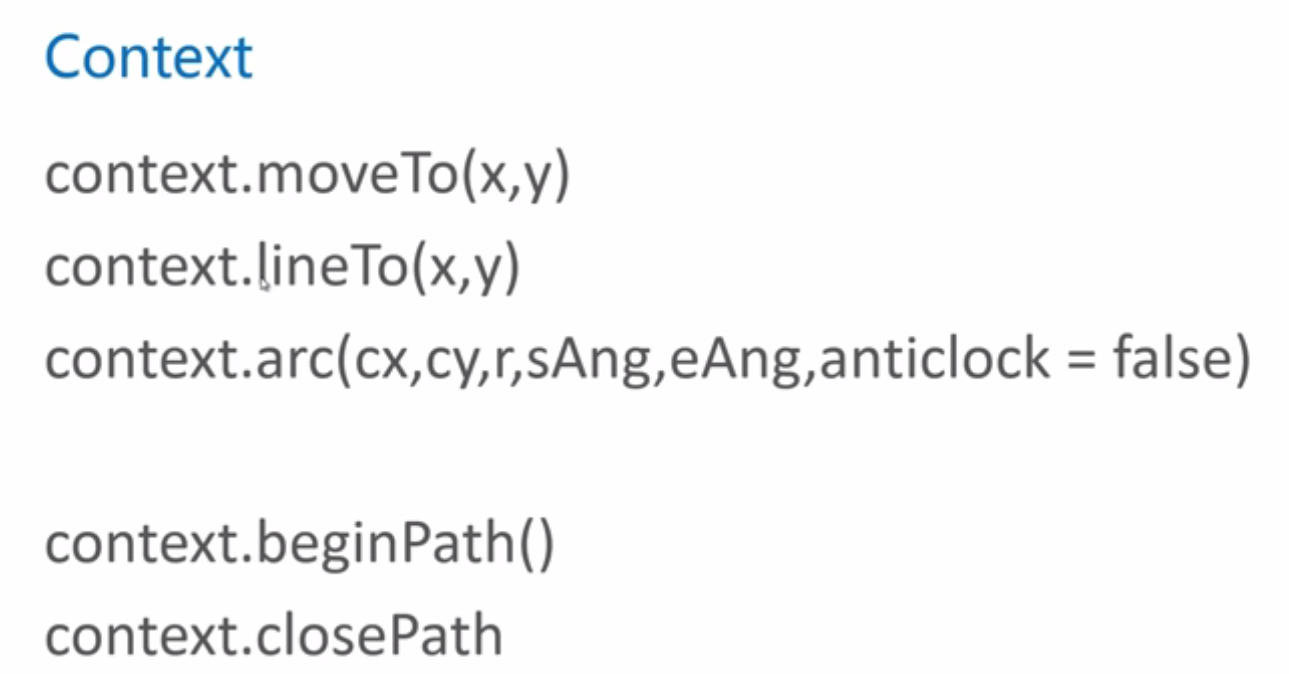
//ret = Math.round(ret/1000);

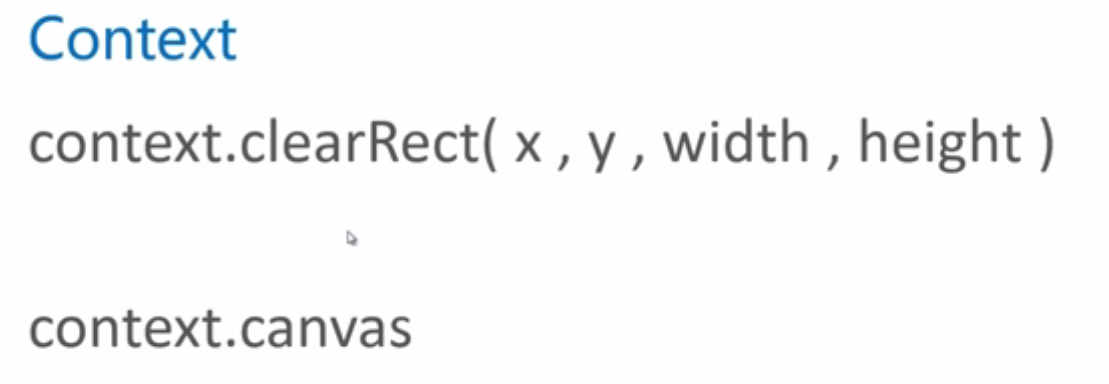
//return ret>0?ret:0;

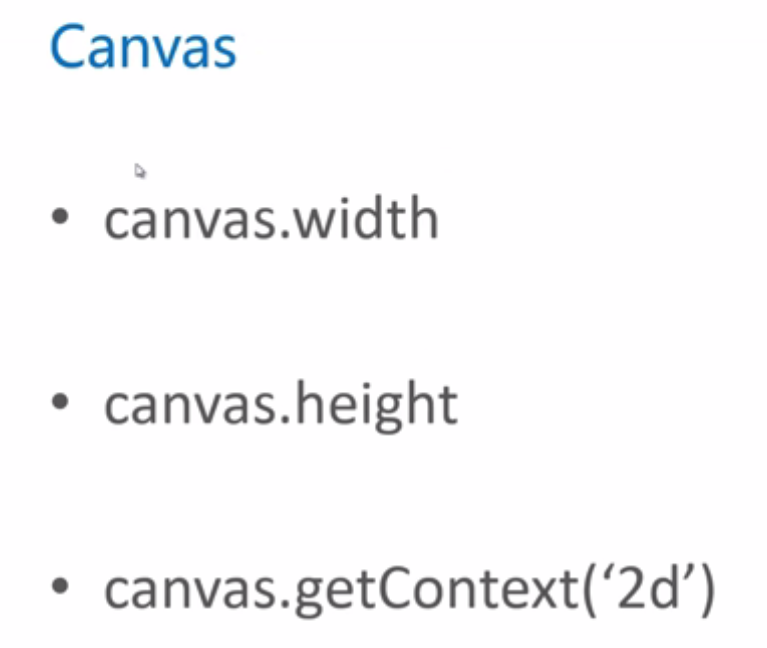
}

# 总结

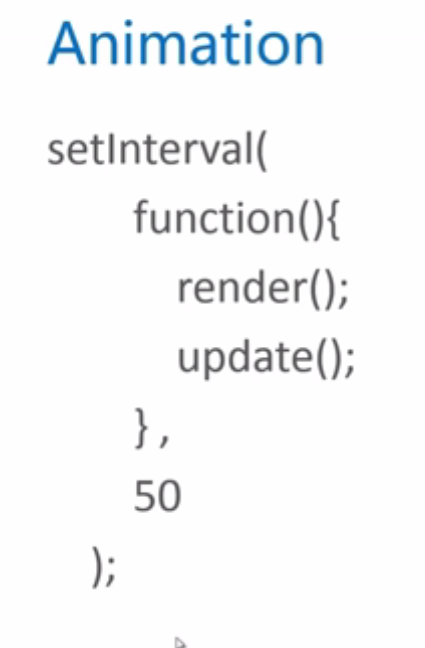








canvas绘制动画 一次渲染+一次底层数据更新



# class2综合介绍