|  |  |
| --- | --- |
| **爱心鱼设计文档** | |
| 将网页框架搭建好 | tinyHeart.html  由两个canvas进行图像的绘制  Canvas.width = 800  Canvas.height = 600 |
| Main.js  作为js的入口 | Can1,can2 是两个canvas的对象引用  canWidth canHeight 是canvas的宽和高  ctx1 ,ctx2 是两个canvas的画布  lastTime deltaTime 是帧的时间  bgPic 背景图片  Game(){  作为起始函数；  }  Init(){  初始化函数；完成canvas的获取，定位，属性的获取  }  Gameloop(){  完成背景的渲染  } |
| Background.js | drawBackground(){  完成背景图片的填充  } |
| CanvasAPI熟悉 | 1.beginPath() : 使用lineWidth strokeStyle moveTo()(定位起始地点) lineTo()(定位终止地点)  (起始一定要在终止前定位)(多次调用lineTo则会生成多条线)  2.closePath() ：会生成一条从当前位置到起始位置的线  3.strokeStyle() : color gradient用于填充绘图的渐变对象 pattern用于创建pattern笔触的pattern对象  ctx.font="30px Verdana";  // Create gradient  Var gradient=ctx.createLinearGradient(0,0,c.width,0);  gradient.addColorStop("0","black");  gradient.addColorStop("0.1","blue");  gradient.addColorStop("0.3","red");  // Fill with gradient  ctx.strokeStyle=gradient;  ctx.strokeText("TEXT",10,50);  4.stroke（）：stroke() 方法会实际地绘制出通过 moveTo() 和 lineTo() 方法定义的路径。默认颜色是黑色。  **提示：**请使用 [strokeStyle](http://www.w3school.com.cn/tags/canvas_strokestyle.asp" \o "HTML5 canvas strokeStyle 属性) 属性来绘制另一种颜色/渐变。  5.lineCap（）：lineCap 属性设置或返回线条末端线帽的样式。  **注释：**"round" 和 "square" 会使线条略微变长。   |  |  | | --- | --- | | butt | 默认。向线条的每个末端添加平直的边缘。 | | round | 向线条的每个末端添加圆形线帽。 | | square | 向线条的每个末端添加正方形线帽。 | |
| Ane.js  绘制海葵 | 使用上述api进行海葵绘制，运用Math.random()函数生成随机的高度 |
| 海葵产生果实  Fruit.js | 设计了30个果实池，用以实现屏幕上有大概15个果实在漂浮  计算变换间隔等数据时：使用了deltaTime(两帧之间的时间)，这样能够保证图像显示比较平滑，自然一点；  drawImage()函数 |
| 大鱼mom.js | 1.大鱼的图像是由3张图片合成的，所以，需要重新定位原点，调用translate()函数进行对画布的原点重新定位，然后再相对于新的原点进行画图，可以调出相互影响变化的效果。  2.使用Math.atan2（x,y）+Math.PI 函数获得极坐标值的偏移量角度值，然后使用rotate（angle）函数对画布进行旋转，形成掉头等动作 |
| 大鱼吃果实的碰撞检测collision.js | 通过计算两个中心点的距离小于一个阈值，然后将果实的alive值置为false，实现消失。 |
| 小鱼baby.js | 大致的内容如大鱼 |
| 绘制小鱼尾巴游动，以及眼睛眨动，身体变色  （大鱼变化相似） | 使用babyTimer进行计时，使用babyCount进行计数，获得相应的图片 |
| gameOver | gameOver为false时，才能吃，才能鼠标控制，才能碰撞  控制透明度rgba = red green blue alpha  shadowBlur shadowColor 阴影 |
| 大鱼吃果实和与小鱼碰撞的特效 | 有一个圈子的pool，然后当碰撞将其激活后，就可以将其画出来，需要一个检测里面那些激活的圈画出来，然后需要半径逐渐增大，颜色逐渐减弱，反比关系。  绘图API  Context.arc(x,y,r,sAngle,eAngle,counterclockwise) ：起始x,y值圆心，r半径,sangle起始角，eangle结束角，counterclockwise为false时顺时针绘图，为true时逆时针绘图 |
| 海葵摆动的动画 | 二次贝塞尔曲线 ：开始点，结束点，控制点  moveTo（x,y）开始点  quadraticCurve(x1,y1,x2,y2) ：x1,y1控制点 x2,y2结束点  使用正弦函数，使海葵的头部来回摆动 |
| 水藻长在海葵上面 |  |
| 绘制浮动的尘埃dust.js |  |
| 总结 | 1. 小鱼动画，大鱼动画   采用轮播序列帧动画，尾巴，身体，眼睛   1. 大鱼喂小鱼，大鱼吃果实   采用碰撞检测   1. 游戏分值计算   绘制文字的api：shadowBlur shadowColor fontSize   1. 特效   物体池（pool）  检测是否有闲着的物体  半径逐渐增大，颜色逐渐减弱，反比关系  绘图API   1. 海葵摆动动画   绘制二次贝塞尔曲线  正弦函数 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |