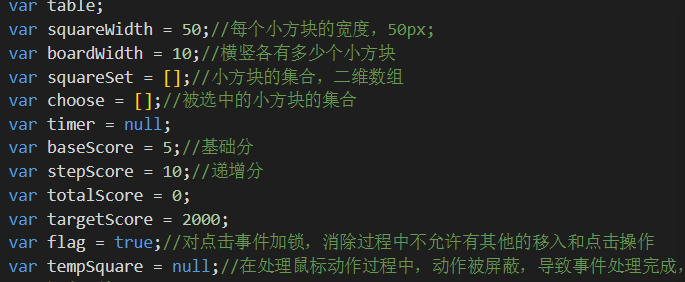
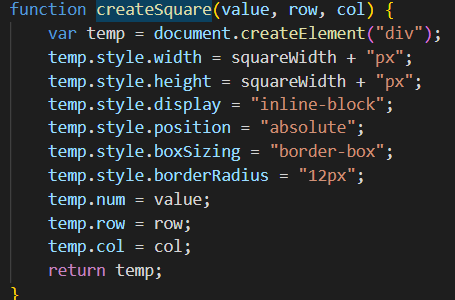
一．初始化变量



二.创建小方块函数 createSquare

1.创建小方块并设置小方块的属性 num表示当前是第几个 row是行 col是列 最后把设置好的小方块返回



三.更新小方块

通过双重for循环 给每一个小方块定位 并设置动画 和背景图片 （兼容判断 看小方块是否存在 如果不存在跳过当前循环进行下次循环）



四．当鼠标放到小方块上的时候调用

1.当flag为false的时候不允许对想方块有操作

2.调用 清除闪烁并恢复图片大小的函数 goback

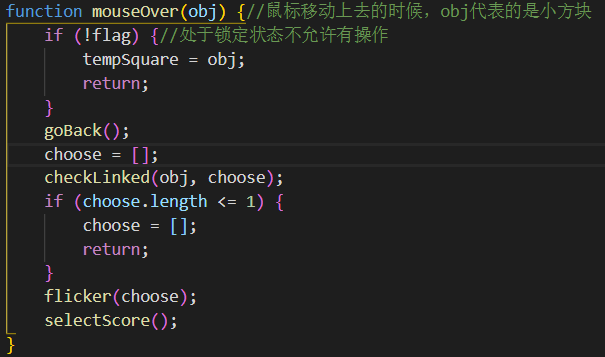
3.将存选中小方块的数组choose设为空

4.调用判断是否有相同颜色小方块链接的函数 checkLinked

5.判断choose的长度是不是小于1 如果是清空数组 并跳出

6.调用闪烁函数 flicker

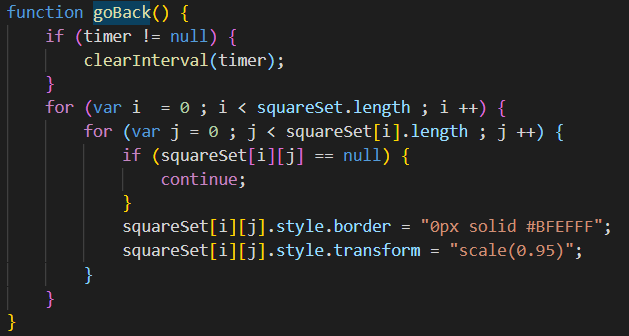
7.调用计算选中小方块总分的函数 selectScore



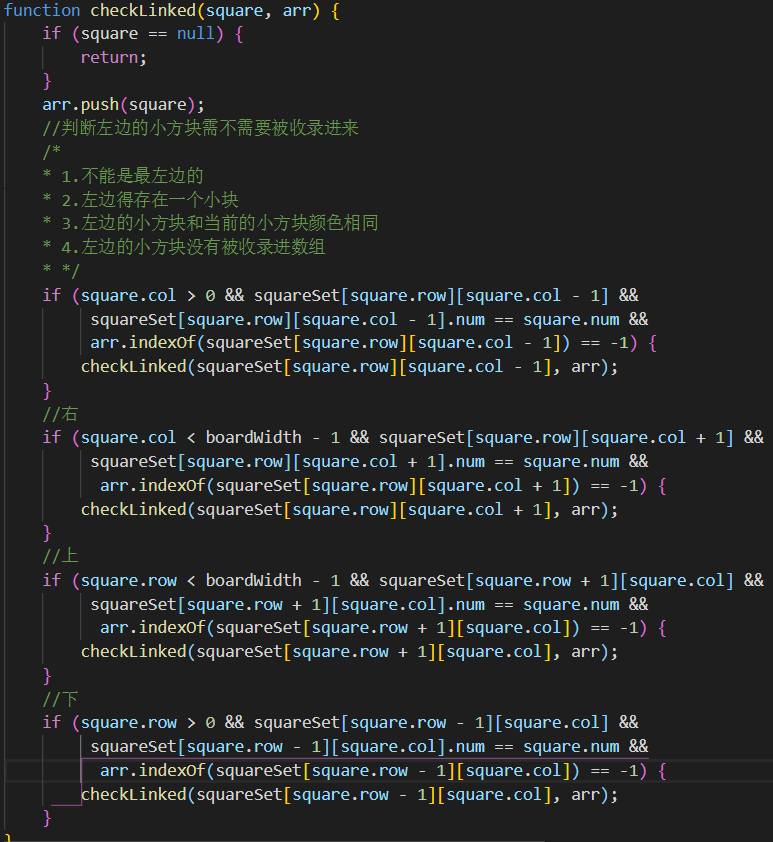
五.创建 清除闪烁的函数

1.判断timer是否有值 有值的话 就清除计时器

2.通过双重for循环 遍历所有小方块 当小方块为没有值的时候 跳过当前循环 否则清除小方块的边框 恢复大小

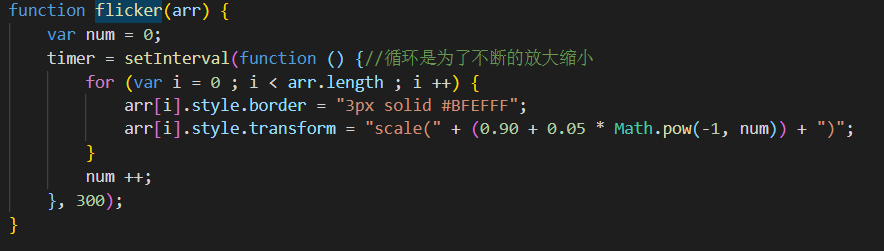


1. 判断小方格周围的小方格 有没有相同颜色的 函数checkLinked（递归的思想）
2. 判断右（上）（下）边小方块是否需要收录进来
3. 不能是最右（上）（下）边的
4. 右（上）（下）边要有一个存在的小方块
5. 右（上）边小方块的颜色要与当前小方块的颜色一致
6. 右（上）边小方块是否被收录进数组



1. 闪烁函数 flicker

创建一个数 num 通过循环 设置计时器在计时器中给每一个选中的方块设置3px的边框 让大小从0.85-0.95之间变动



八.计算选中小方块的总分的函数 selectScore

创建一个分数的容器score 通过遍历计算总分 如果score小于等于0的时候跳出函数将得分写入html 清空动画 将几块几分的div的透明度设置为1 延迟一秒后 消失



1. 小方块移动函数 move

1.上下移动

通过遍历判断每一行是否有空的 如果没有pointer++ （pointer 每次等于j） 当有空的时候 pointer不会加一 但j会加1 导致j大于pointer 让下一层（pointer层）的小方块的位置等于上一层（j层）的小方块的位置 让上一层小方块的行等于pointer 之前上一层的位置设置为空 已达到小方块下落的目的

2.左右移动

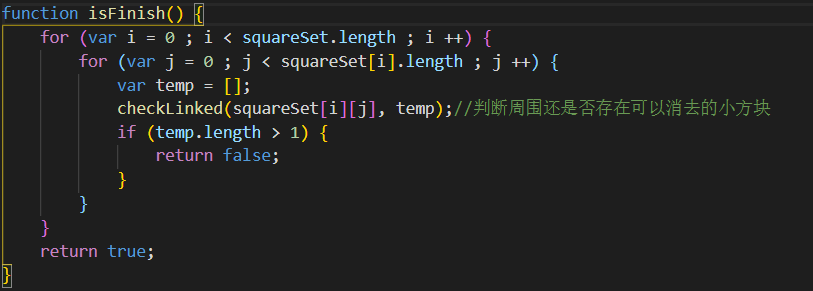
固定行为第0行 当第0行第i列为0的时候 通过遍历清空这一列 否则跳出这一循环 达到左右移动的目的

3.之后调用更新小方块的函数 refresh



十．判断是否结束的函数

通过双重for循环 遍历每一个小方块 创建一个数组temp 看每一个小方块周围是否还有相同颜色的小方块 如果有就存入数组temp 当temp大于1 的时候 返回false 否则返回true



十一.主函数

1.获取最外层的框

2.通过双for循环 创建二维数组

3.创建square变量来接收 创建的小方块

4.在小方块上 创建鼠标悬停事件

5.在小方块上 创建点击事件

6.点击后 判断flag是否为空 被选中小方块的数组是否等于0 如果是设置为锁定状态

7.点击后 flag设置为false tempSquare设置为空

8.创建总分变量 totalscore 每次计算消方块所获得的分数 之后加给总分 然后渲染到页面



1. 消灭星星 for循环遍历选中的小方块 通过立即执行函数的方法每次将选中小方块所对应的数组清空 并在dom上清除 每次延迟间隔0.1s
2. 移动 创建延迟计时器 调用move函数 （延迟时间choose数组长度\*100ms）然后再次创建延迟函数 （延迟时间choose数组长度\*100ms + 300ms）创建is 变量来接收isFinish函数返回的值 当is为true的时候 判断当前得分是否大于目标得分 大于为成功 否则为失败 当is为false的时候 清空choose数组 flag = true 调用mouseOver函数并向函数中传入变量 tempSquare

11.（在点击事件外）把创建的小方块存入数组 并渲染到dom里



最后调用 更新函数

