# 拖拽+重力+碰撞

* **一.拖拽**
* 拖拽要实现的效果：1.当鼠标在div上点下去后，移动鼠标，div会跟着鼠标移动；2.当鼠标快速移动时，即使移出div，div也相应移动。
* 分析需求：1.把.onmousemove放在.onmousedown里面，这样只有点击了，再移动才有效果；2.希望“即使是移出div，div也能移动”，说明.onmousemove事件不是绑定到div的，而是绑定到document上的。
* **oDiv.onmousedown** = function(ev){
* //存储距离
* var oEvent = ev || window.event;
* var disX = oEvent.clientX – oDiv.offsetLeft;
* var disY = oEvent.clientY – oDiv.offsetTop;
* **document.onmousmove** = function(ev){
* //根据距离，计算div最新的距离
* var oEvent = ev || window.event;
* var scrollLeft = document.body.scrollLeft || document.documentElement.scrollLeft;
* var scrollTop = document.body.scrollTop || document.documentElement.scrollTop;
* var l = oEvent.clientX + scrollLeft - disX;
* var t = oEvent.clientY + scrollTop – dixY;
* //做一个判断，看是否会被拖出可视区
* if(l < 0){l = 0;}
* else if(l>document.documentElement.clientWidth – oDiv.offsetWidth){} l = document.documentElement.clientWidth - oDiv.offsetWidth;
* if(t < 0){t=0;}
* else if(t >document.documentElement.clientHeight-oDiv.offsetHeight){t = document.documentElement.clientHeight-oDiv.offsetHeight;}
* oDiv.style.left = l +'px';
* oDiv.style.top = t +'px';
* }//document.onmousemove结束
* **document.onmouseup** = function(ev){
* document.onmousemove = null;
* document.onmouseup = null;
* }
* }
* **二.碰撞+重力**
* 判断是否超出可视区的时候，要超过下面时，纵向速度反向且减小\*-0.8，横向速度\*0.8，要超过上面时，纵向速度只需反向\*-1，横向速度\*0.8，要超过右面时，横向速度反向且减小\*-0.8，要超过左面时，横向速度\*-0.8；当速度很小的时候（绝对值小于1），则将速度设为0；
* 碰撞运动的终止条件：当横向和纵向速度等于0时，并且碰到了底端时，关闭定时器，必须有后面这个碰到底端，否则可能在空中最高的时候速度为0就停下来
* **三.将拖拽和（碰撞+重力）结合起来：**
* 将（碰撞+重力）封装成一个函数，然后在鼠标点下去，又再抬起来，即document.onmouseup里面，调用这个函数，在这个函数外面，即oDiv.onmousedown里面，清除计时器，就不会按着鼠标时，方块还往下掉。还有通过**iSpeedX(iSpeedY)**,这个（碰撞+重力）会用到，速度是通过距离算的，所以在拖拽中的document.onmousemove里设为iSpeedX = l - lastX; //当前一瞬间的速度：后一个点的横坐标-前一个点的横坐标。lastX = l;// 更新：当前的在下一次中即为前一个