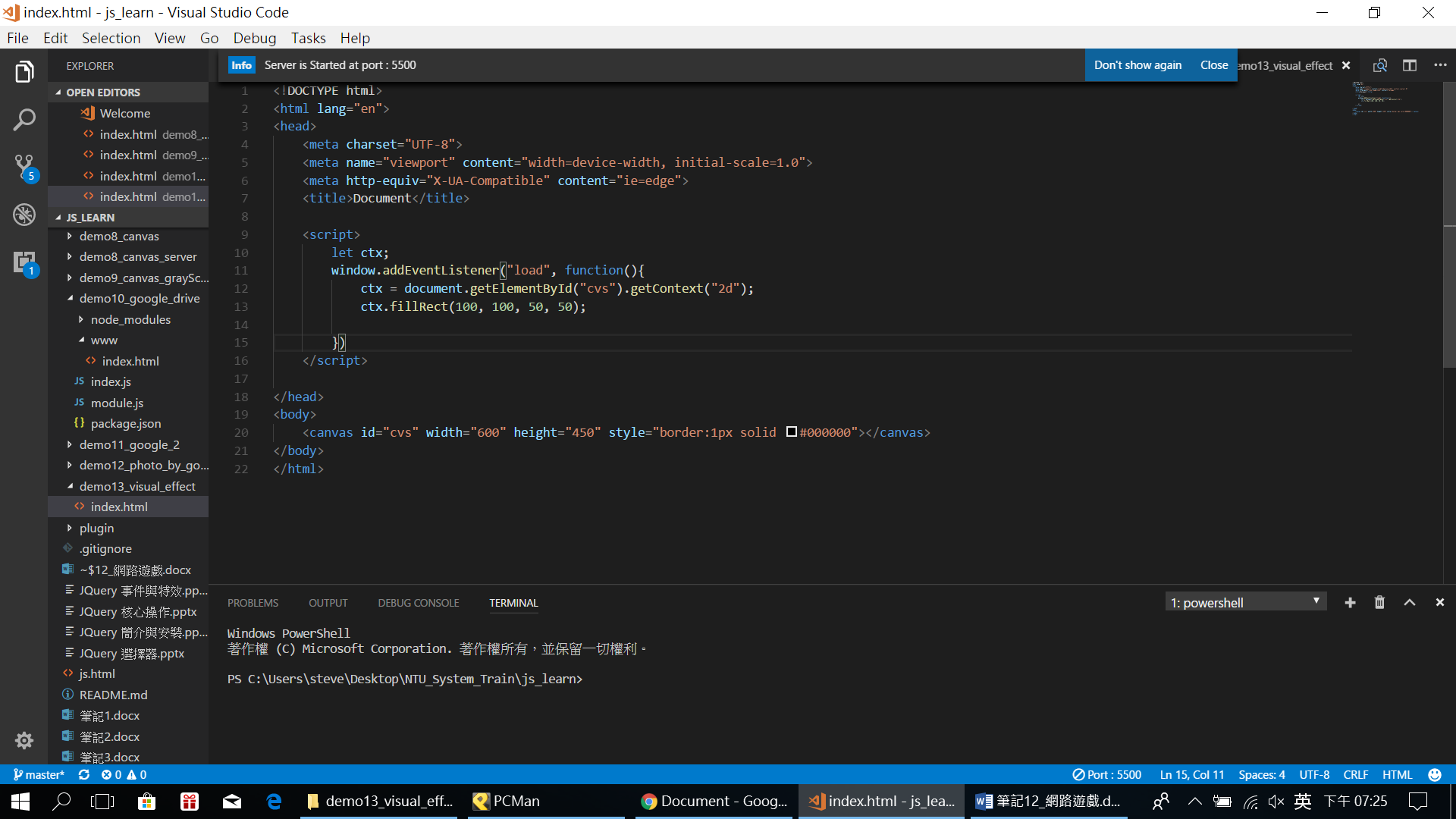
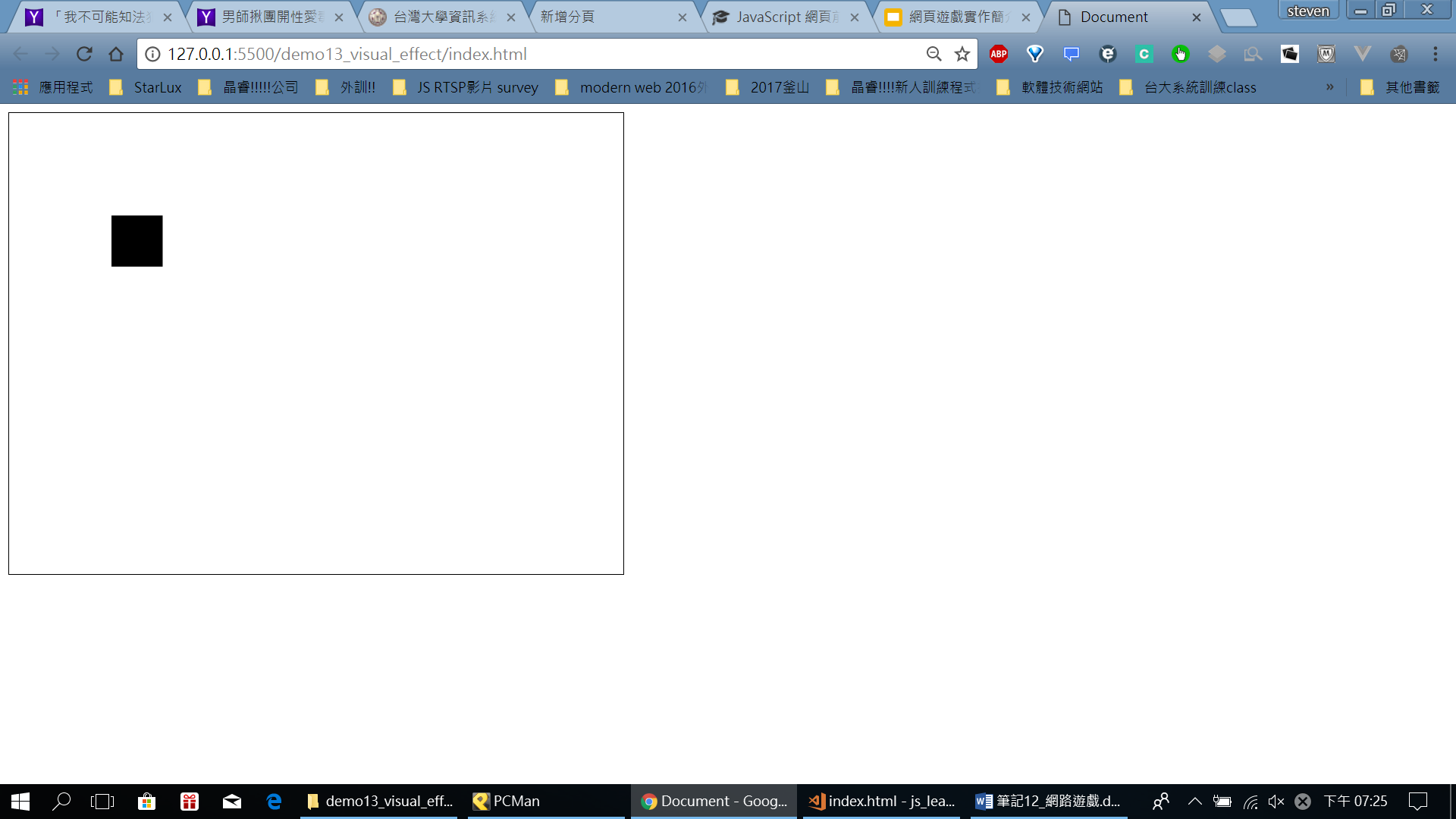


1. 先來示範用canvas

先:

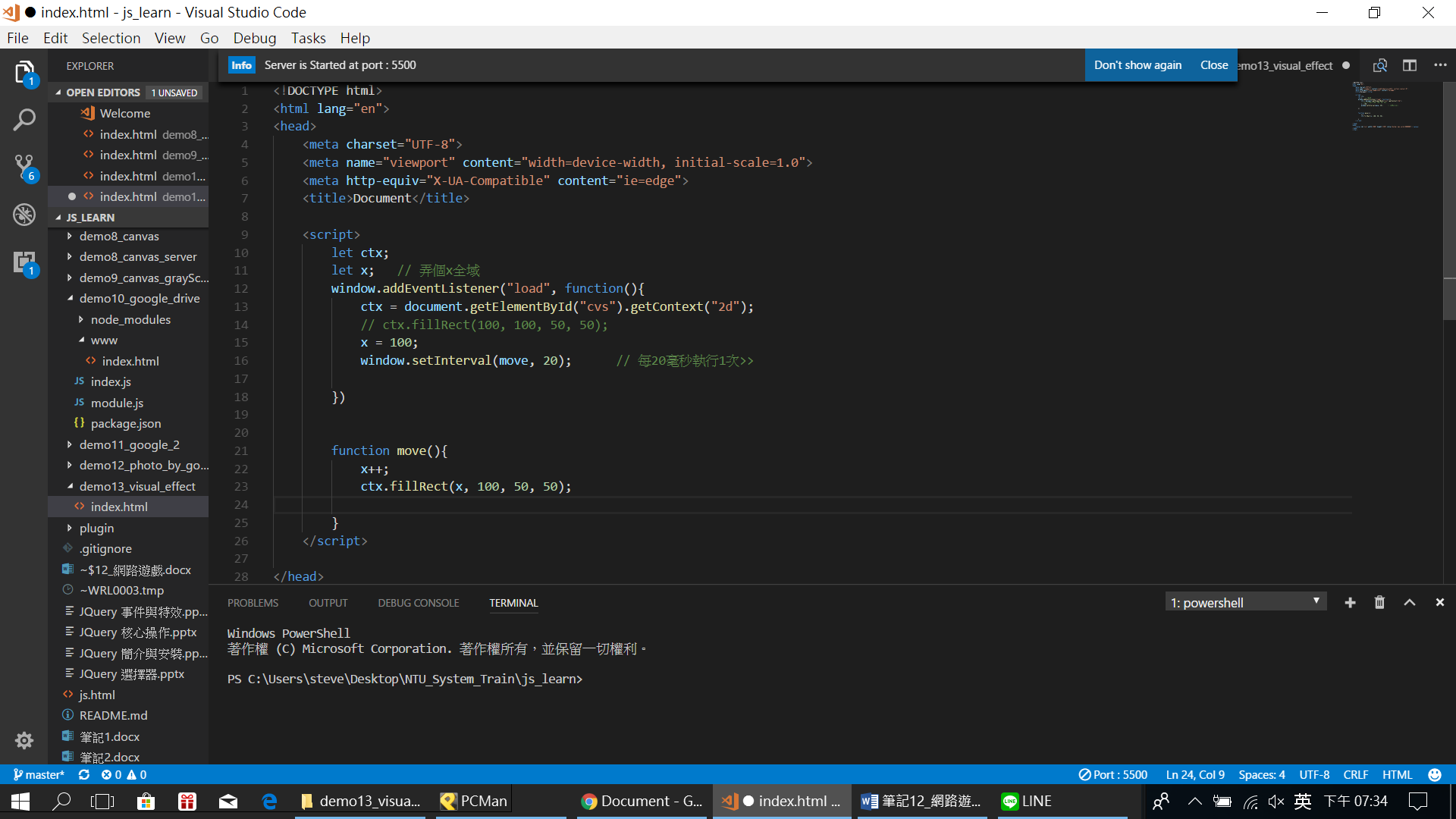




怎樣做事會讓使用者覺得畫面會動>> 不斷畫新的!

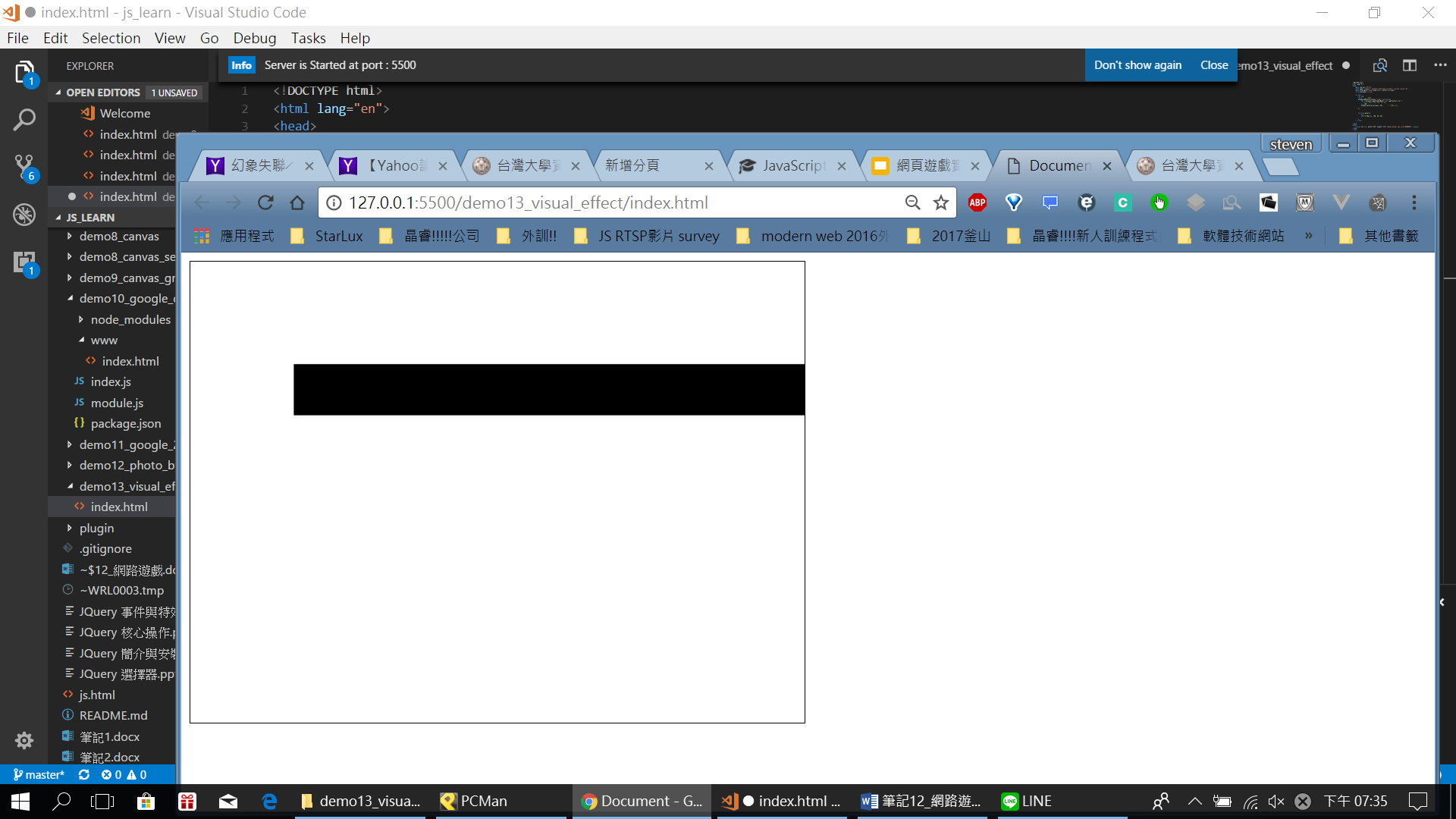
但時間的間距!

再改:



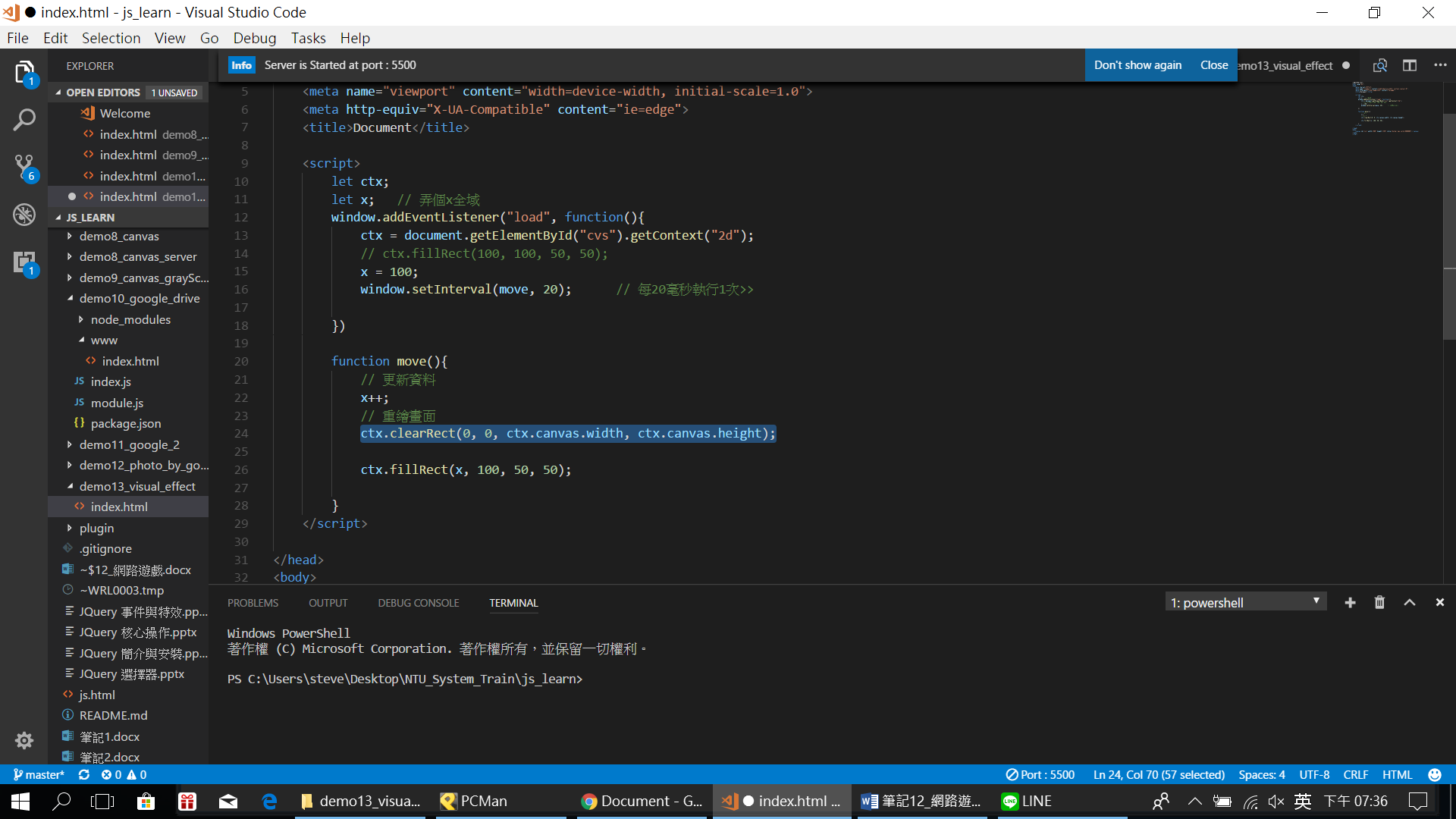
會發現不對啊!

會一直變大:

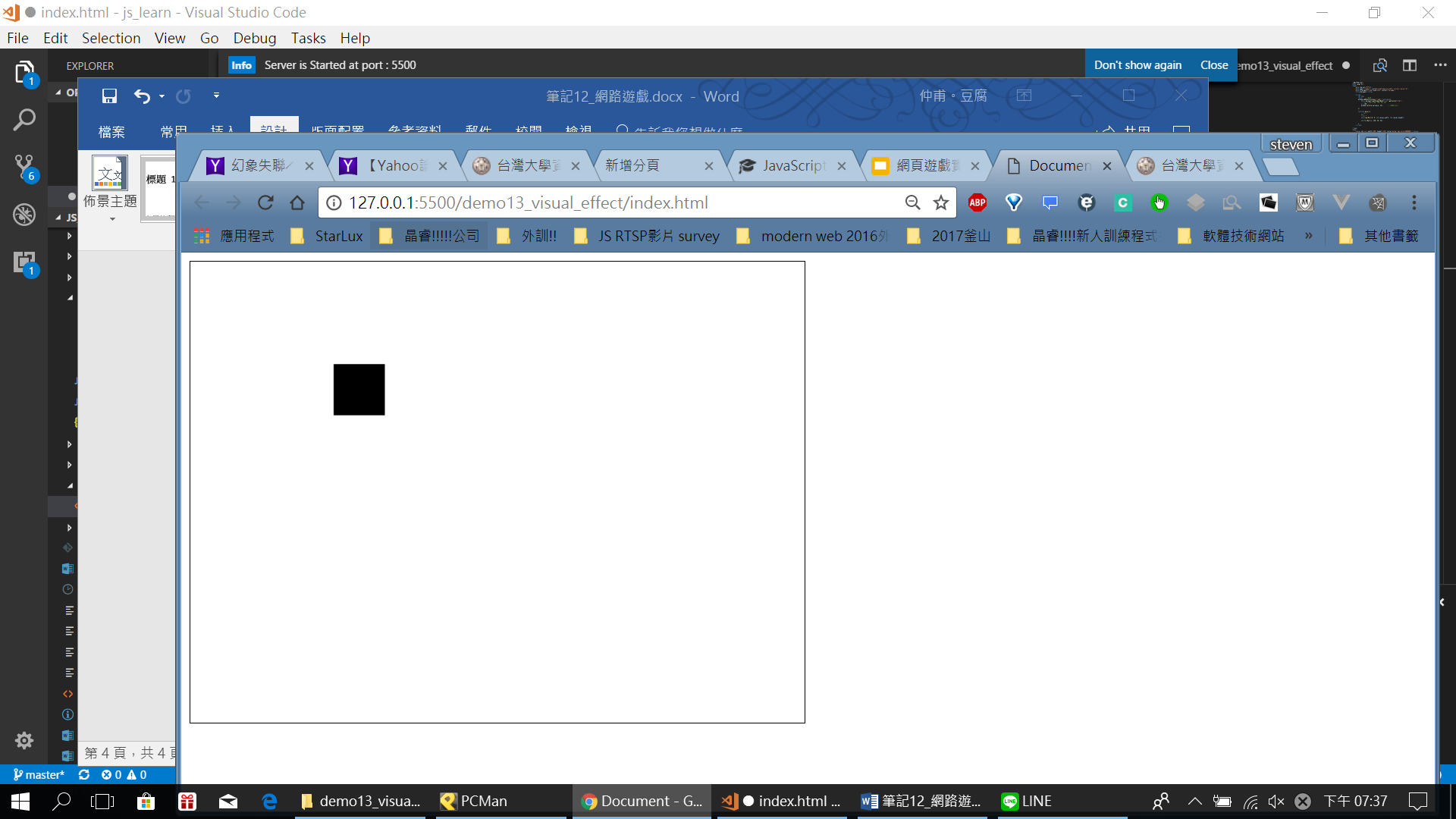


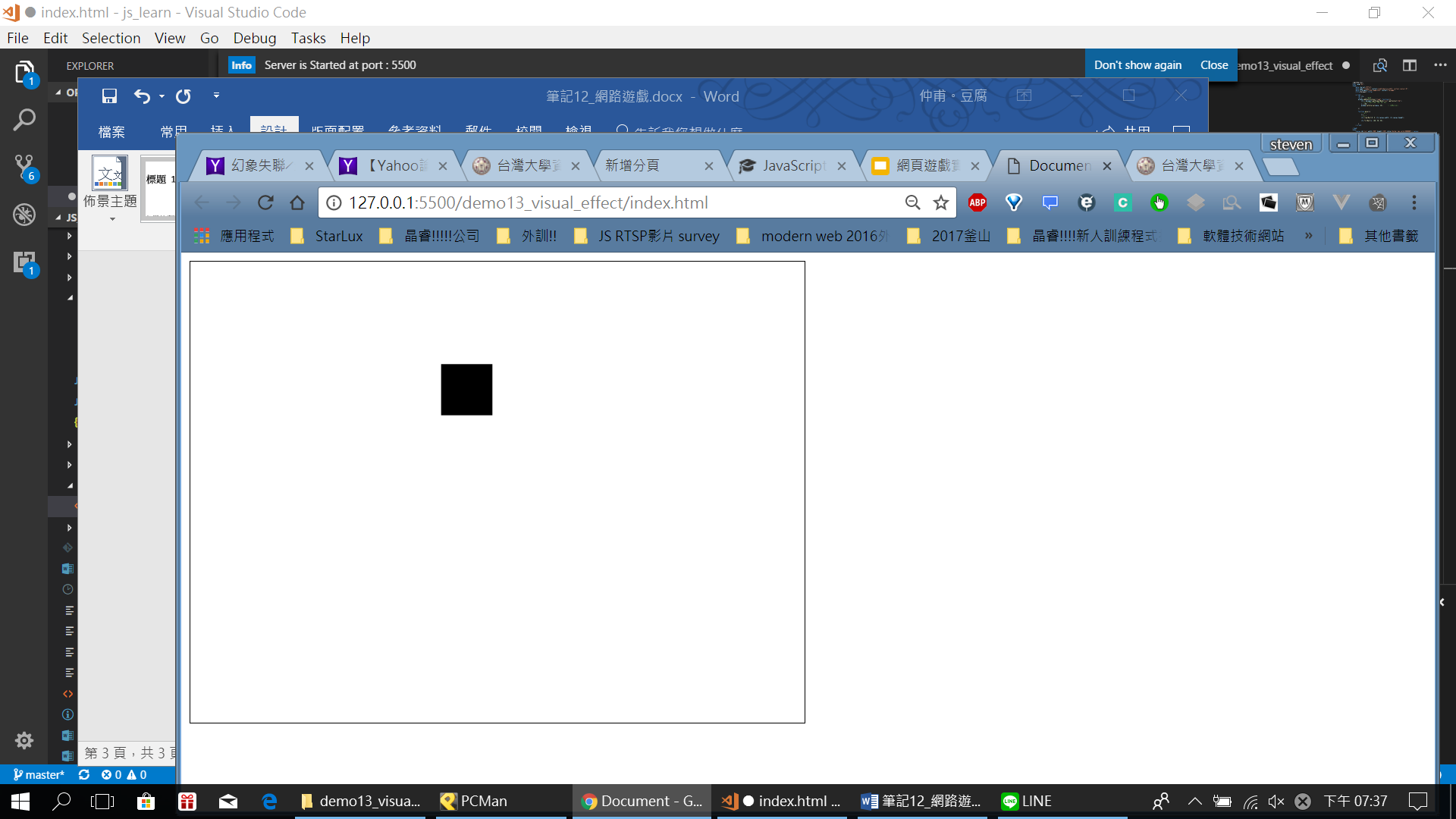
因為canvas是一直覆蓋!!!

要改為清空!!>

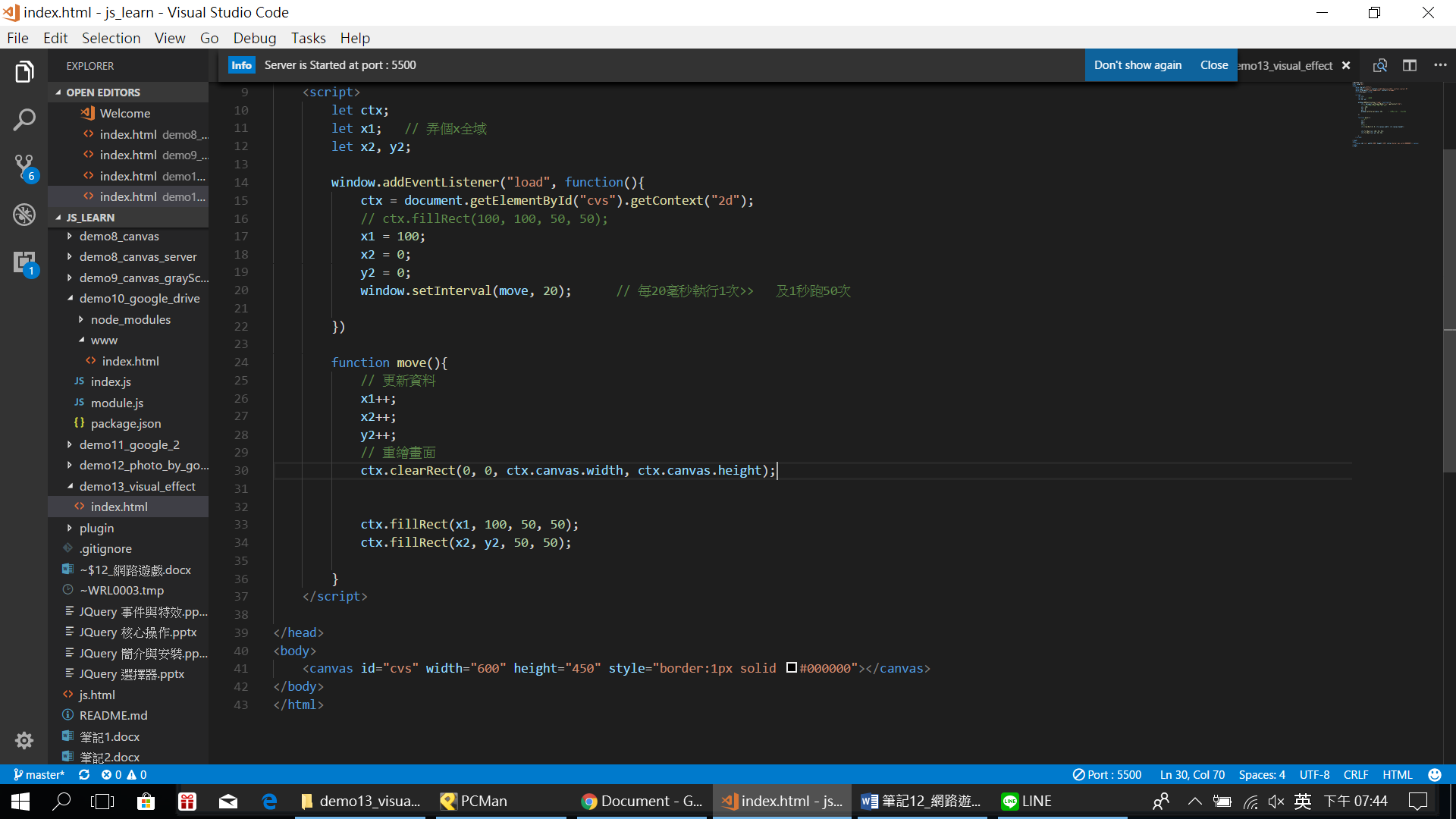


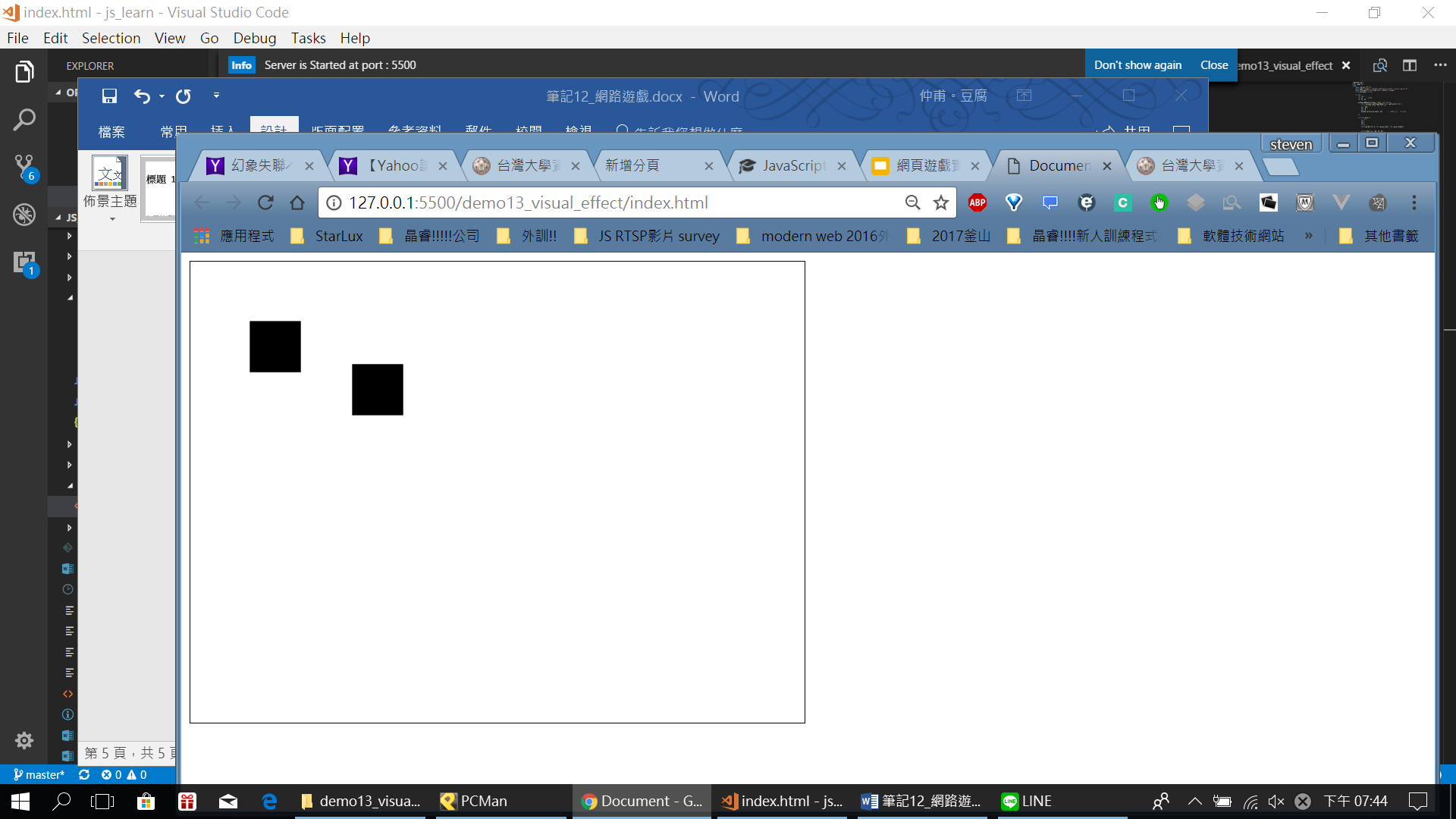
就正常動惹!!!>



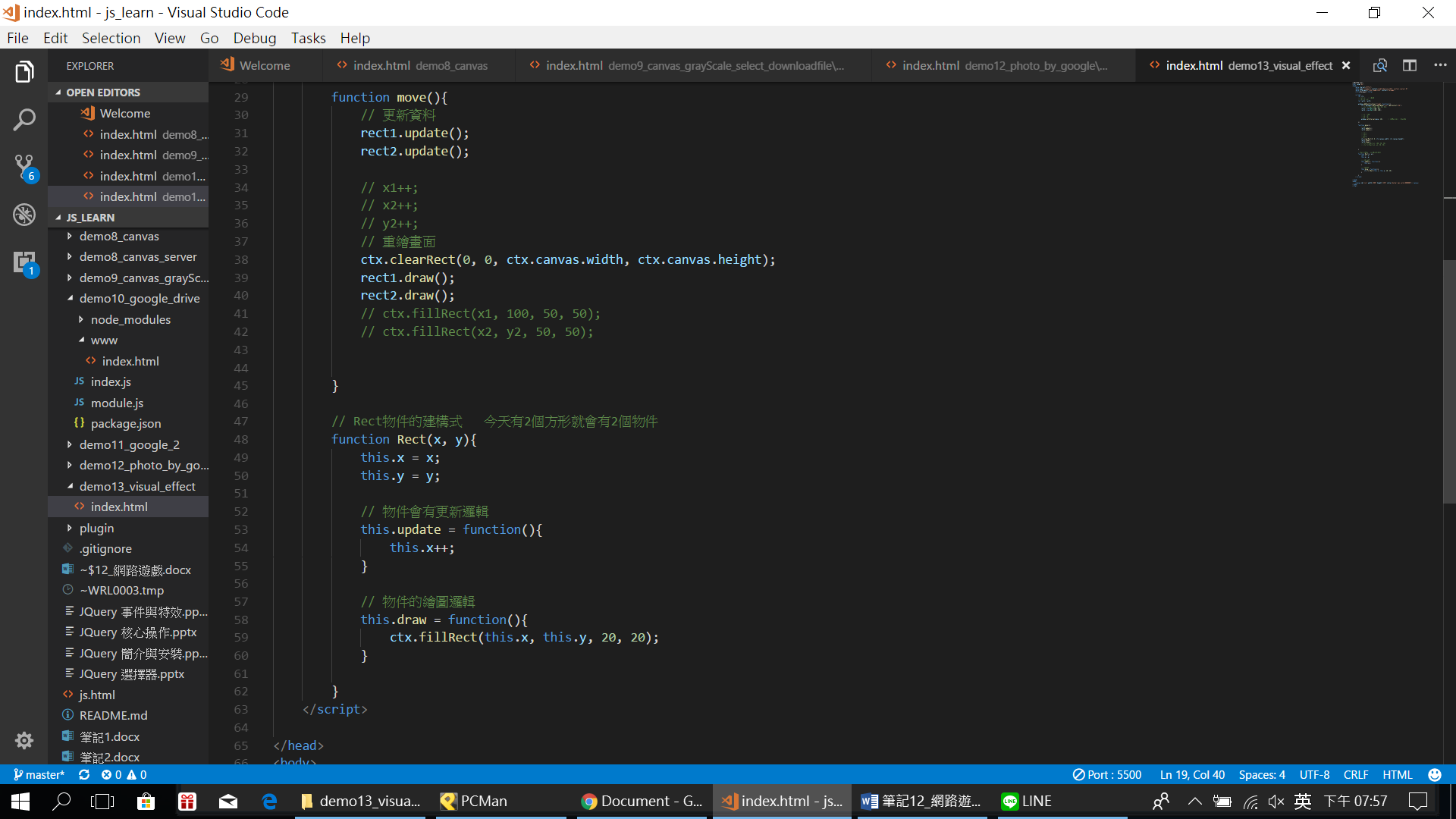
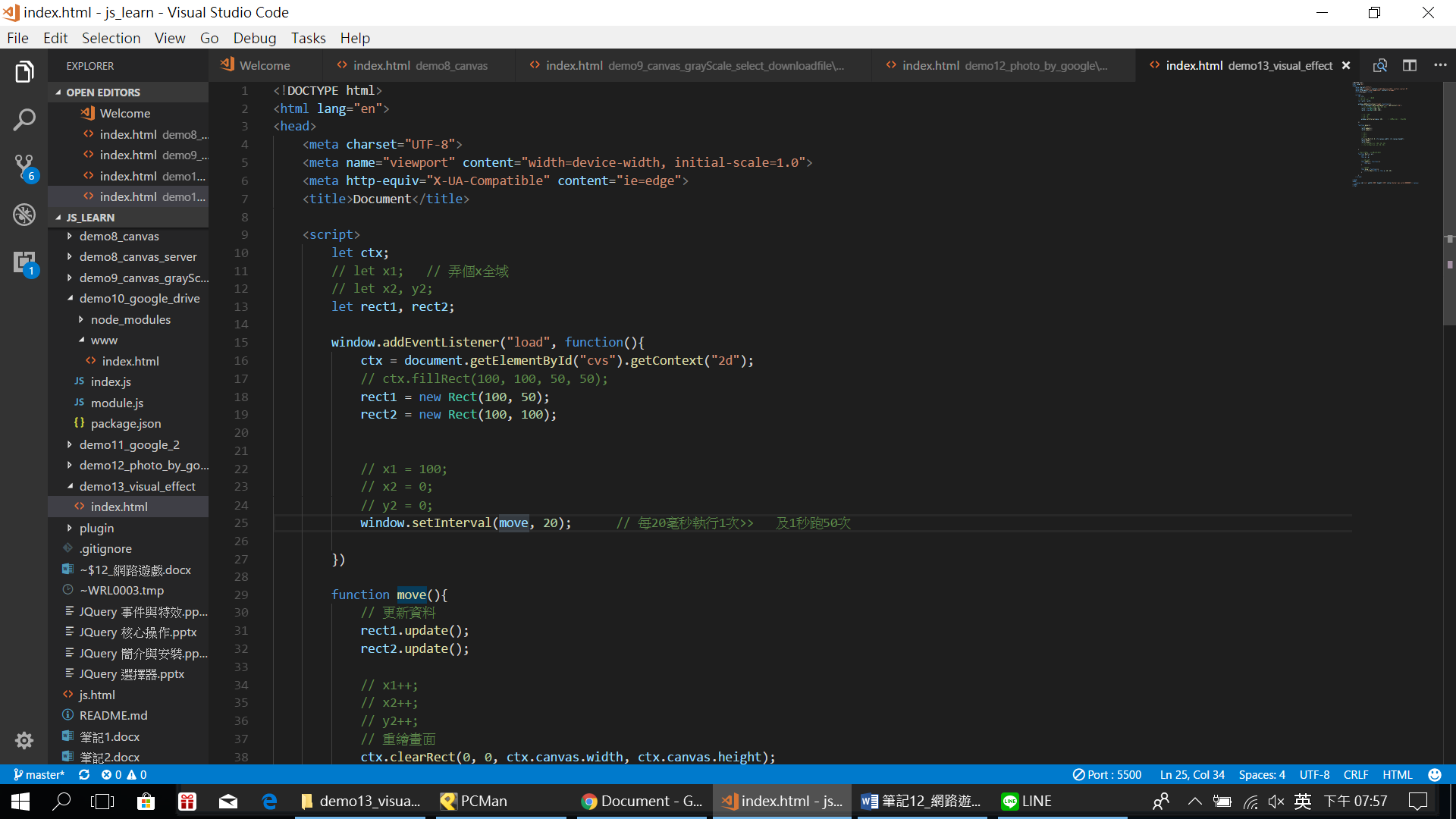


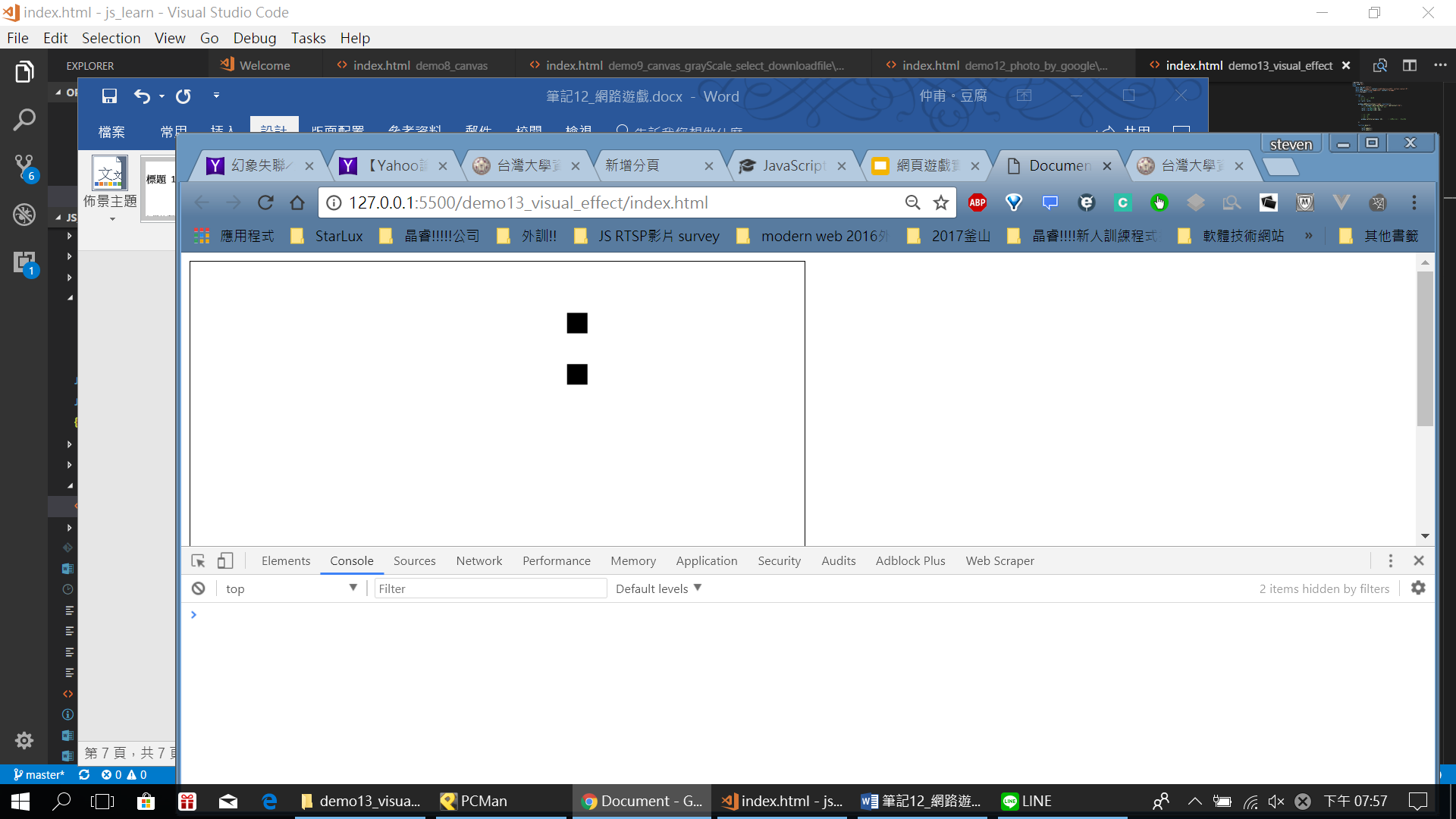
1. 畫2個

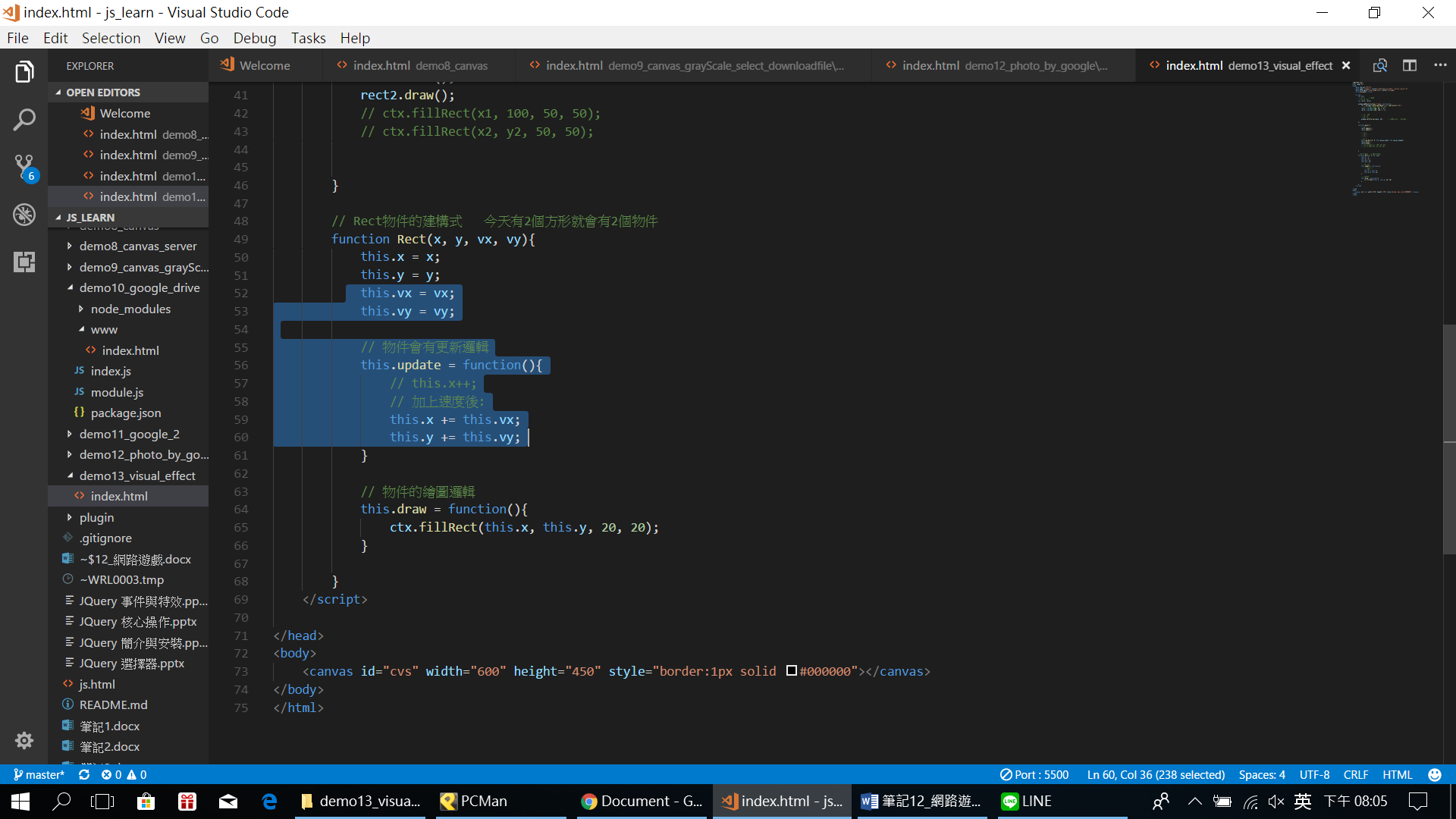


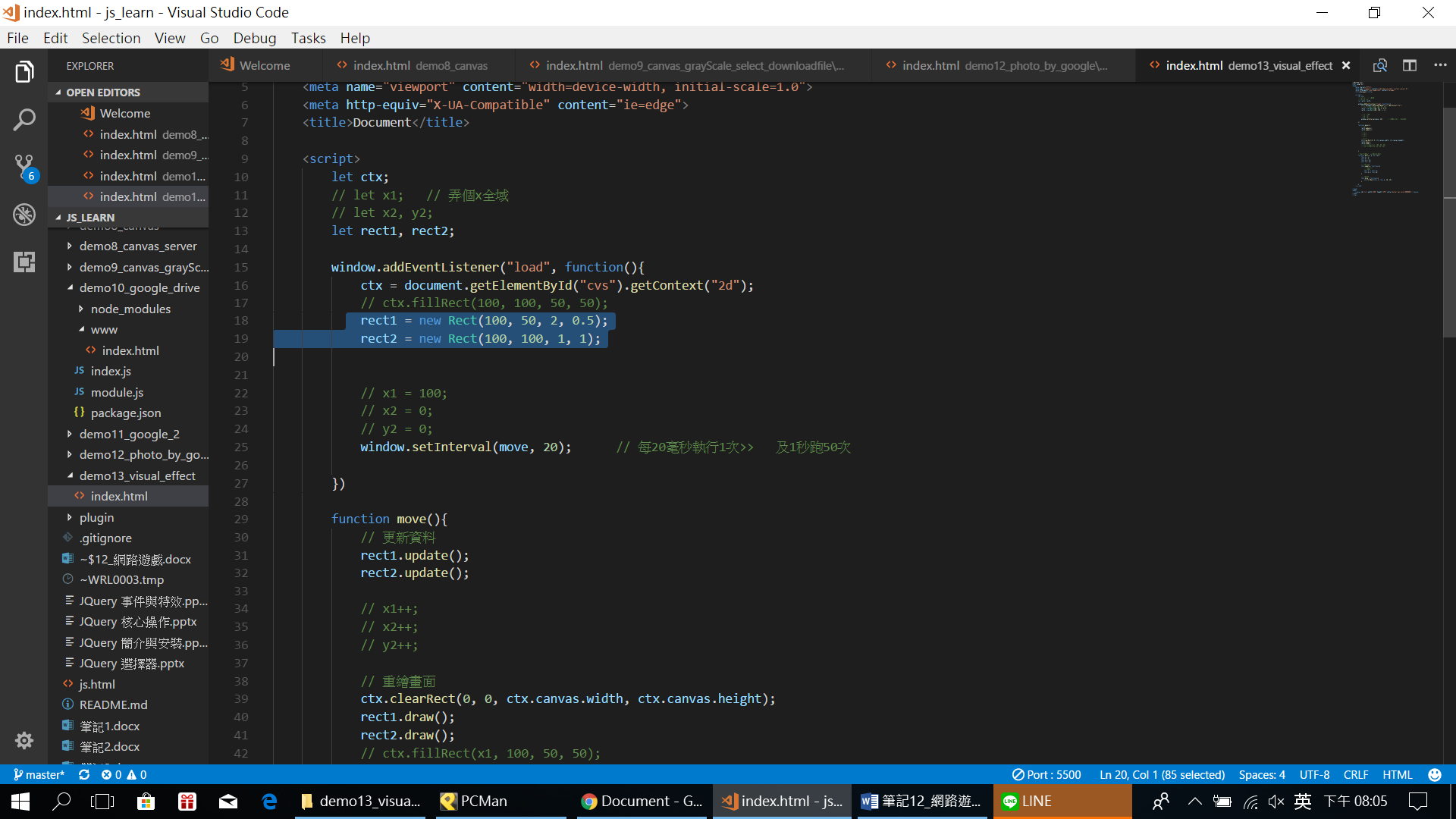


1. 改為物件導向:





1. 來設定不同速度吧! 



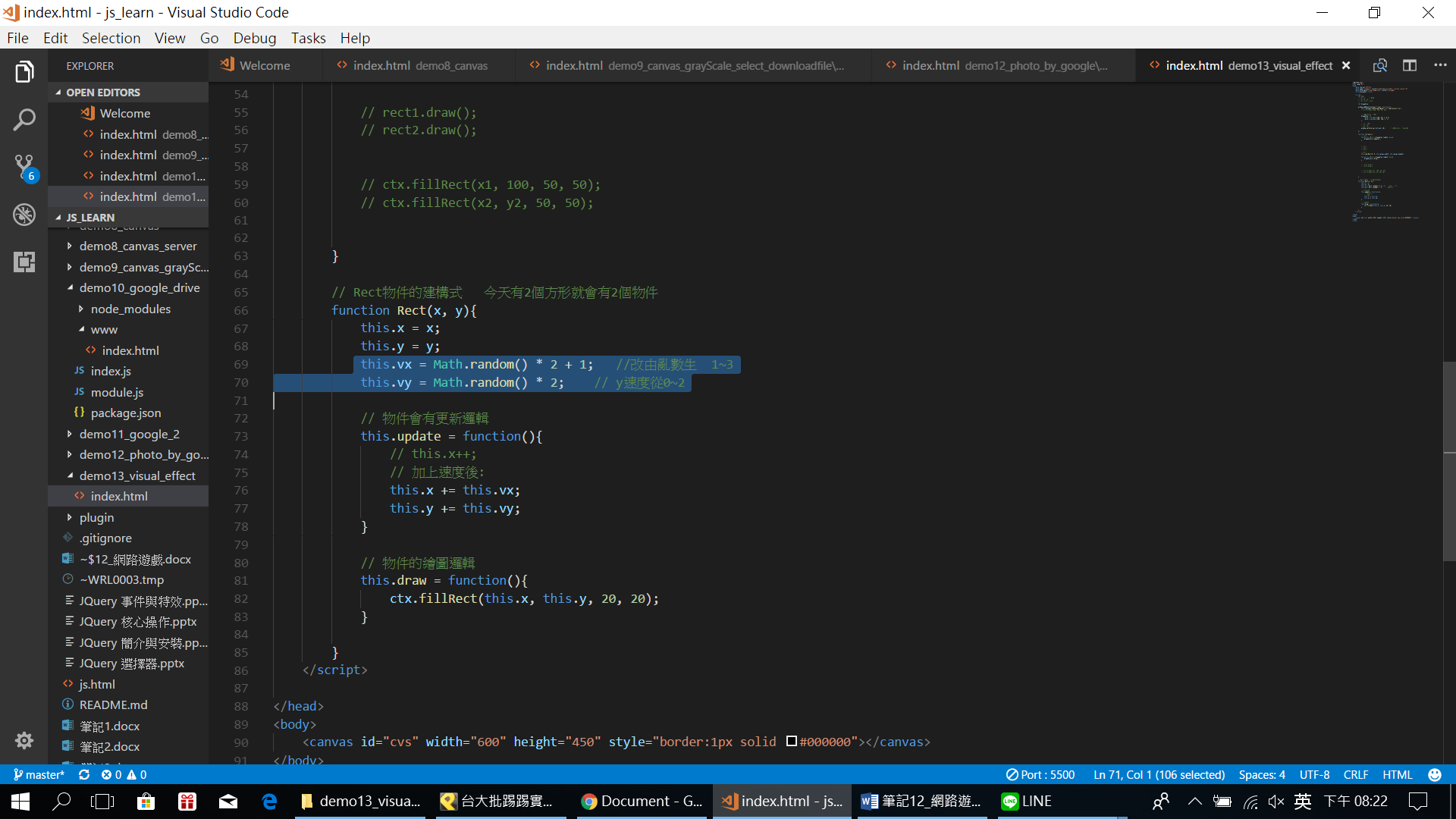
1. 再改成只痊癒一個全部可畫的
   * 再用一個陣列管理所有可畫到畫面的東西
2. <!DOCTYPE html>
3. <html lang="en">
4. <head>
5. <meta charset="UTF-8">
6. <meta name="viewport" content="width=device-width, initial-scale=1.0">
7. <meta http-equiv="X-UA-Compatible" content="ie=edge">
8. <title>Document</title>
10. <script>
11. let ctx;
12. // let x1; // 弄個x全域
13. // let x2, y2;
14. // let rect1, rect2;
15. //
16. let drawables;
17. window.addEventListener("load", function(){
18. ctx = document.getElementById("cvs").getContext("2d");
19. // ctx.fillRect(100, 100, 50, 50);

22. // 用一個陣列管理所有可畫到畫面的東西
23. drawables = [
24. rect1 = new Rect(100, 50, 2, 0.5),
25. rect2 = new Rect(100, 100, 1, 1)
26. ]
28. // x1 = 100;
29. // x2 = 0;
30. // y2 = 0;
31. window.setInterval(move, 20); // 每20毫秒執行1次>> 及1秒跑50次
32. })
33. function move(){
34. // 更新資料
35. for(let i = 0; i < drawables.length; i++){
36. drawables[i].update();
37. }
39. // x1++;
40. // x2++;
41. // y2++;
43. // 重繪畫面
44. ctx.clearRect(0, 0, ctx.canvas.width, ctx.canvas.height);
46. for(let i = 0; i < drawables.length; i++){
47. drawables[i].draw();
48. }
49. // rect1.draw();
50. // rect2.draw();

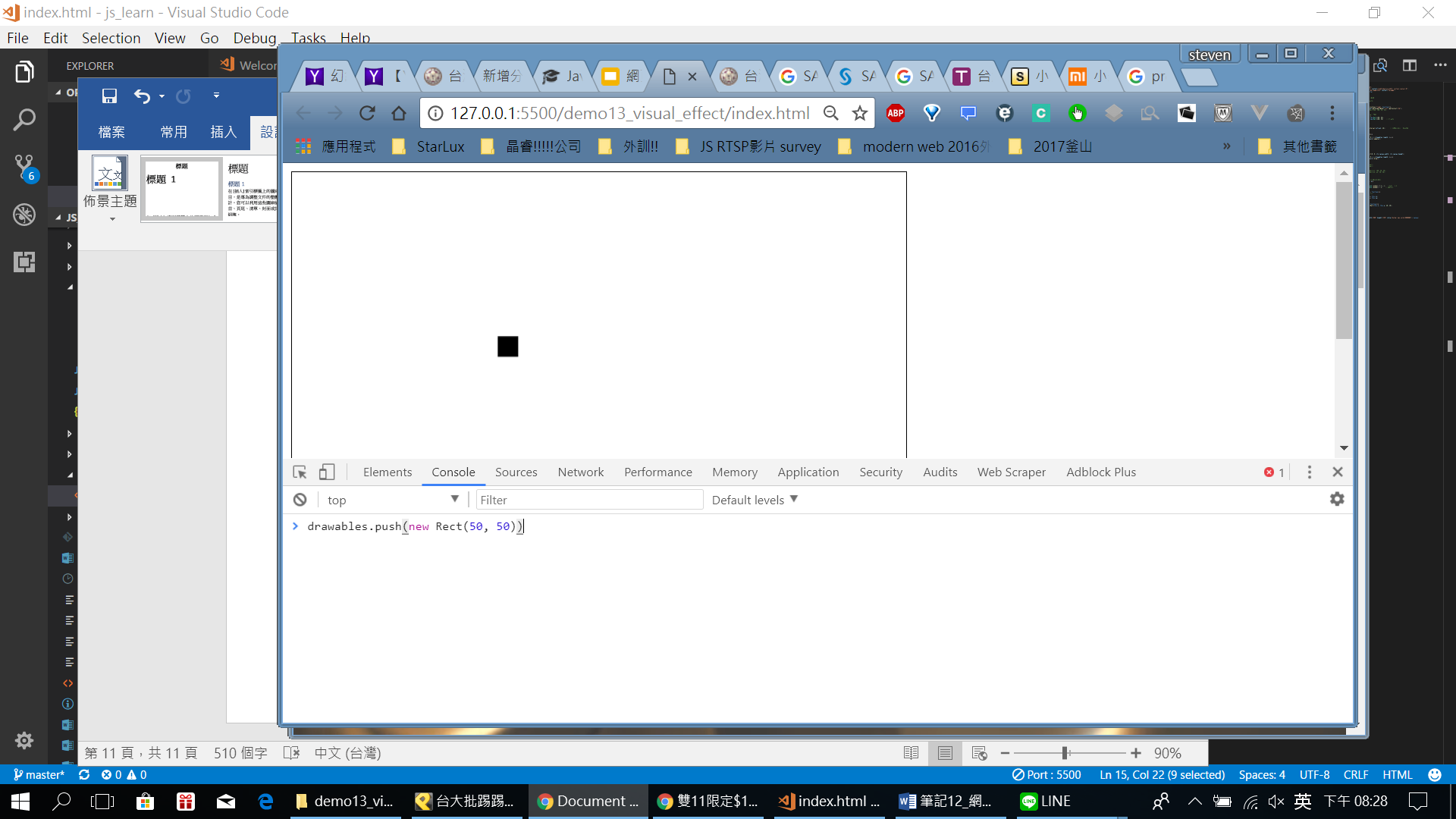
53. // ctx.fillRect(x1, 100, 50, 50);
54. // ctx.fillRect(x2, y2, 50, 50);
56. }
57. // Rect物件的建構式 今天有2個方形就會有2個物件
58. function Rect(x, y, vx, vy){
59. this.x = x;
60. this.y = y;
61. this.vx = vx;
62. this.vy = vy;
63. // 物件會有更新邏輯
64. this.update = function(){
65. // this.x++;
66. // 加上速度後:
67. this.x += this.vx;
68. this.y += this.vy;
69. }
70. // 物件的繪圖邏輯
71. this.draw = function(){
72. ctx.fillRect(this.x, this.y, 20, 20);
73. }
74. }
75. </script>
76. </head>
77. <body>
78. <canvas id="cvs" width="600" height="450" style="border:1px solid #000000"></canvas>
79. </body>
80. </html>

6. 從方形生小方形

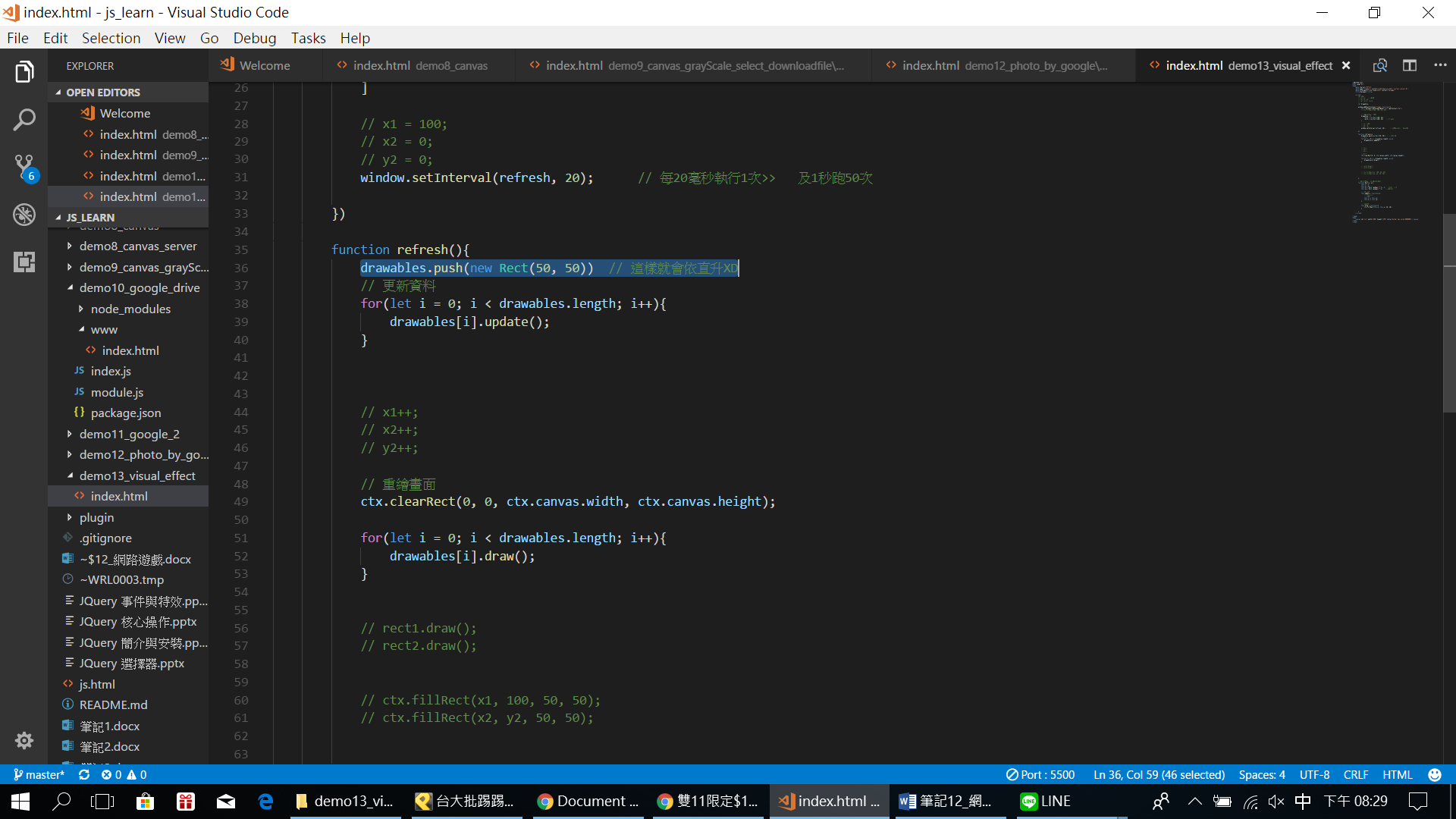
先拿掉速度參數，改由電腦產生:

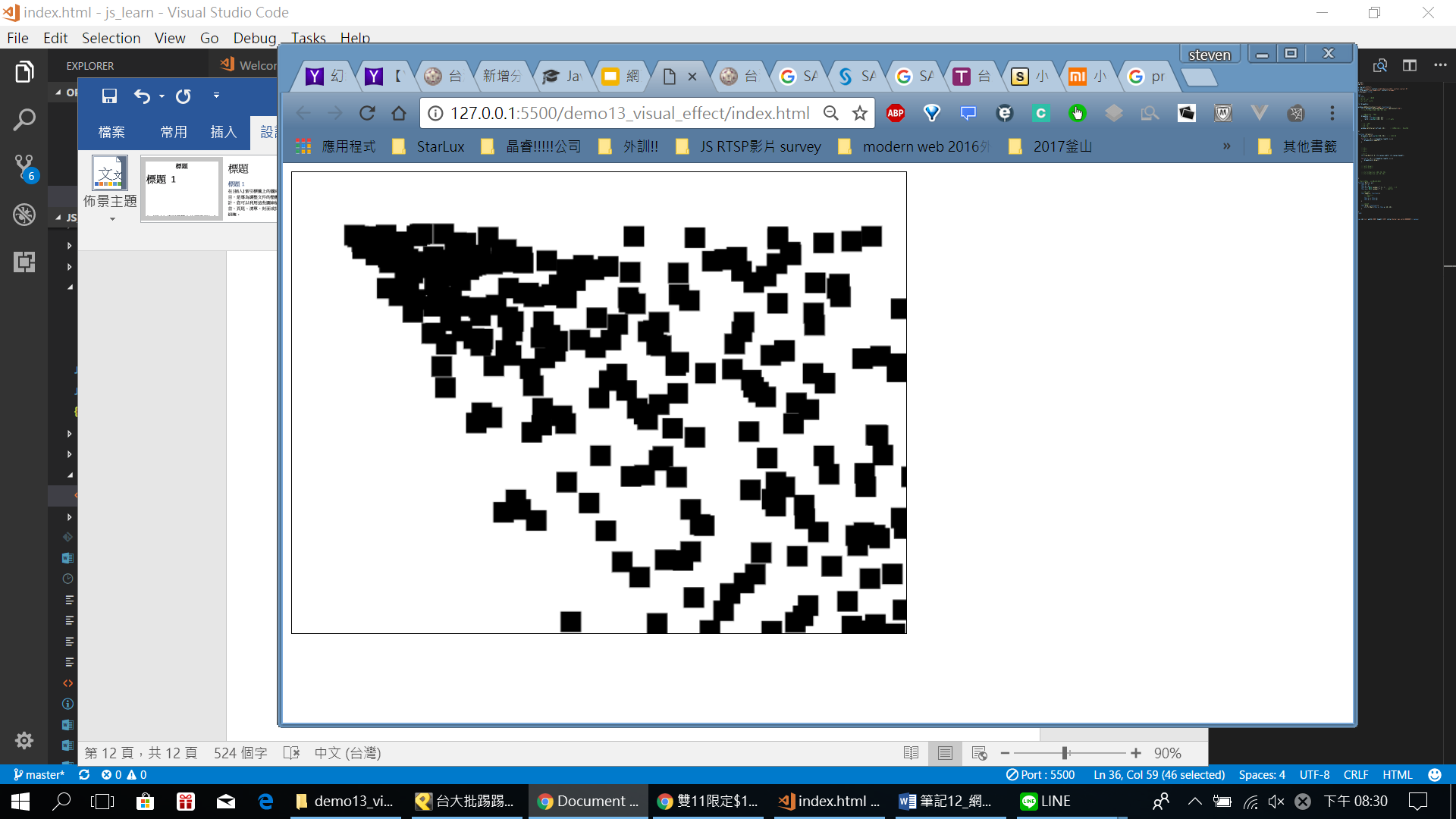


此時可以看到:

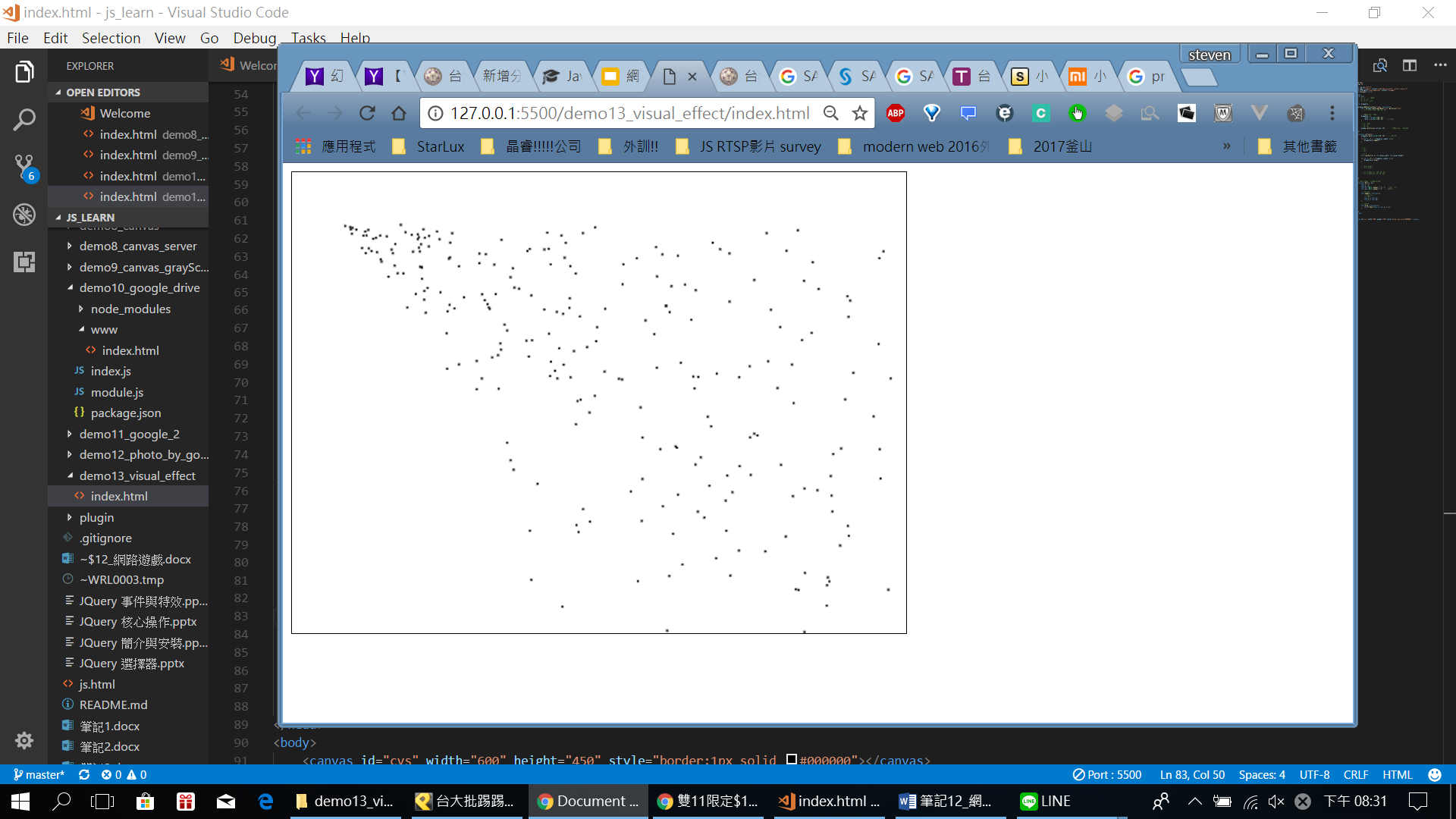


此時在refresh函式:



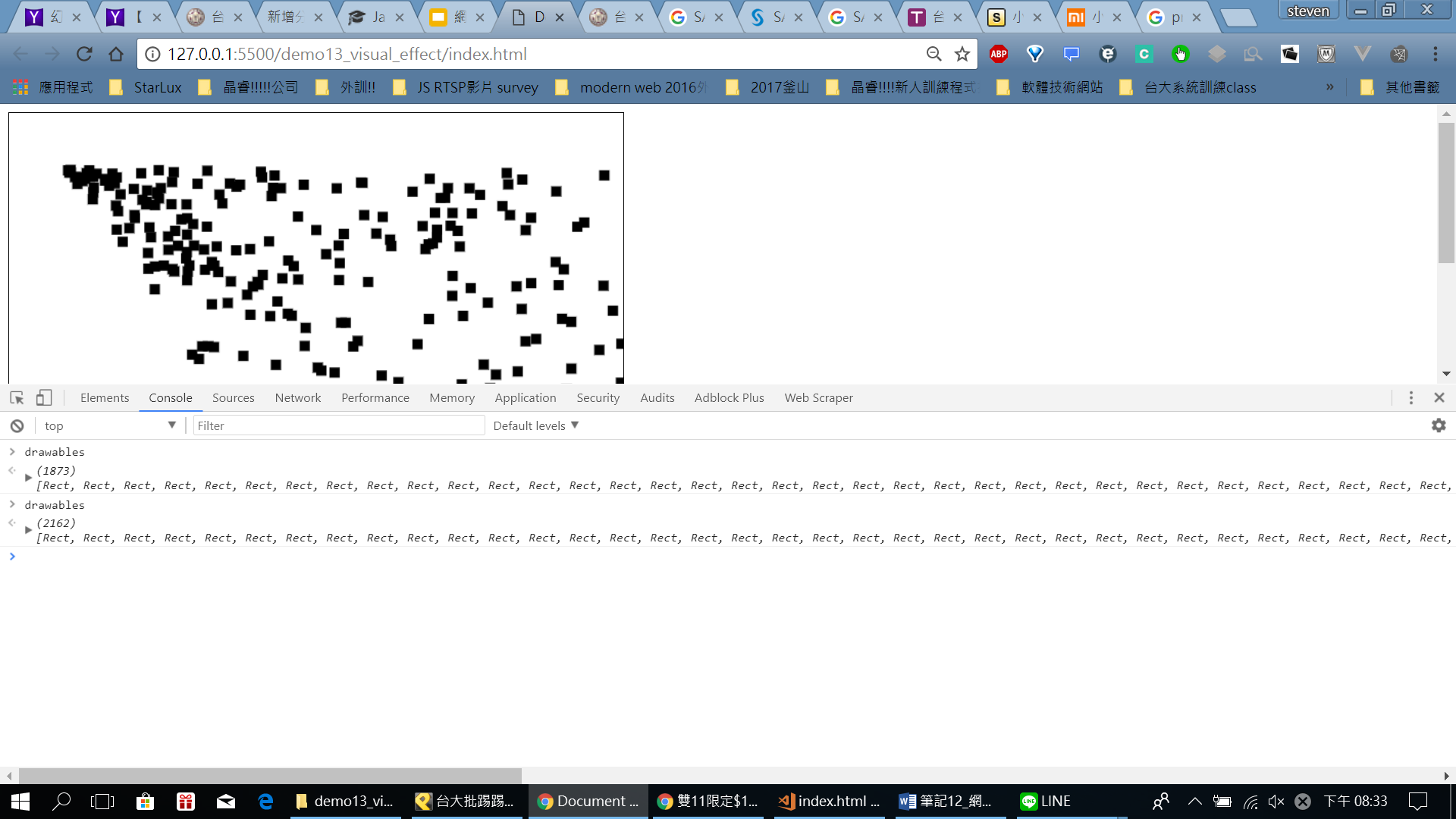


可以把塗改小:

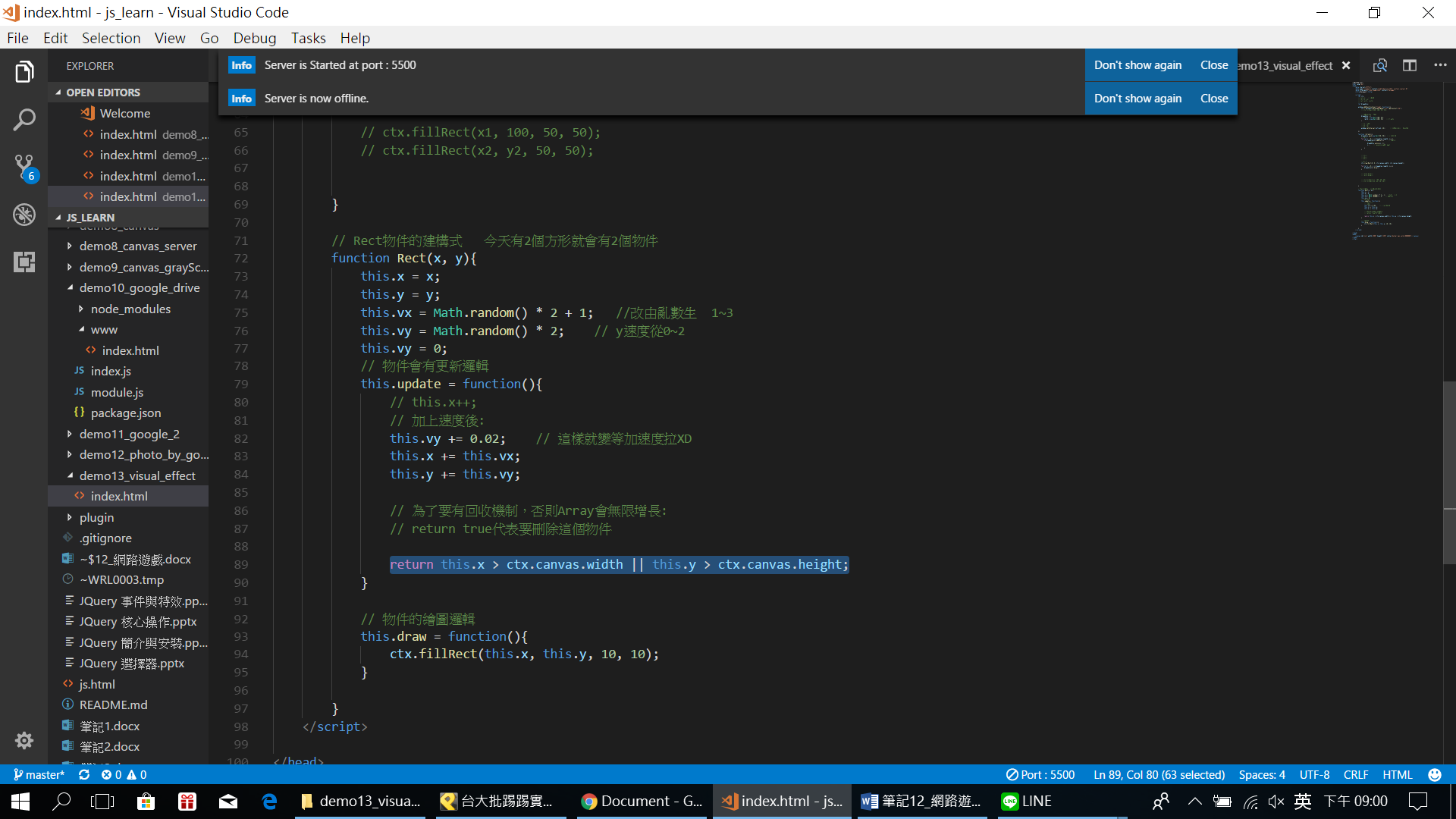


但是>>>>>>>>

陣列目前會無限成長啊 (1秒就多了50個欸!!!) 遲早遊戲會爆炸!!!



所以要有回收的機制





變成等加速度:

