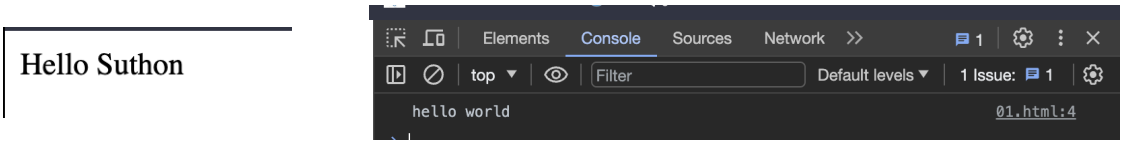
CHAVATIK THORARIT Group 9 6610110066

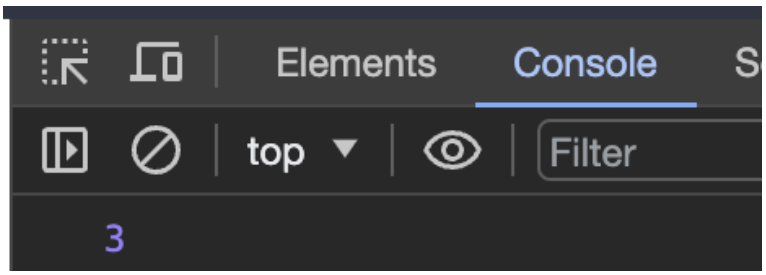
Html01



Run เราจะเห็นข้อความคำว่า
<body>Hello Suthon </body>
มีการexcucute แสดง<script>hello world ในส่วนของconsole

Html02

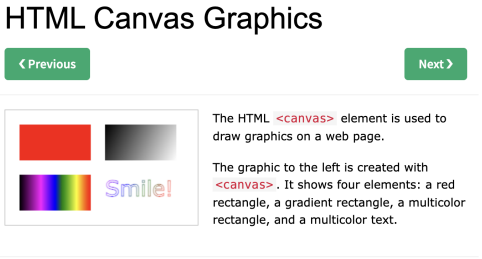
```
<html>
   <head>
       <script>
           let x = 3;
           console.log(x);
       </script>
   </head>
   <body>
       Hello Suthon
   </body>
</html>
```



Run เราก็จะเห็นข้อความheelo suthon เหมือนในข้อที่1 แต่มีการdefine ตัวแปร ให้ x =3 แล้วprint ค่าออกมา คือ 3 ตามที่เห็นในส่วนของ console

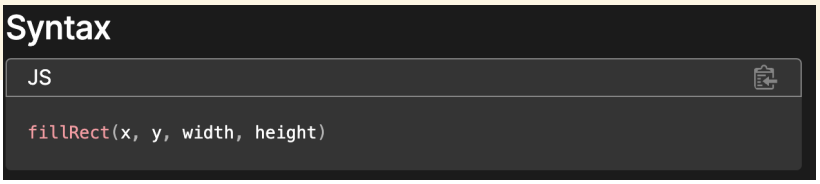
CHAVATIK THORARIT Group 9 6610110066

HTML03



view after run เห็น red square แถวๆมุมบนซ้ายของweb ในส่วนตรงดีได้มีการใช้canvaselement เพื่อกำหนด ขนาด ยาว กว้าง in this case 400*400

```
<html>
   <head>
   </head>
   <body>
       <canvas id="paper" width="400" height="400">  ->define element ID
       </canvas>
   </body>
   <script>
       let canvas = document.getElementById("paper")-->  docgetElementByid เรียกcanva
id ในที่นี้ คือ  paper canva = id'paper' something like that
       let ctx = canvas.getContext('2d')-->กำหนดรูปแบบเป็นแบบ2d
       ctx.fillStyle = 'red' ใส่สีแดง
       ctx.fillRect(10,10, 20, 20)       ->determine initial point เป็นจุด (10,10)
       โดยมี width = 20 height = 20
   </script>
</html>
```



CHAVATIK THORARIT Group 9 6610110066

HTML04

เริ่มมีอะไรน่าสนุกเกิดขึ้นมาละ จุดsquearеขยับไปทางขวาทีละนิดๆ แล้วที่สำคัญมันเดินทะลุออกไปจากจอได้ หายแว็บไปเลยหาไม่เจอ

มีการต่อยอด มาจากข้อตะกี้ คือมี การสร้างfunction draw

```
<html>
  <head>
  </head>
  <body>
      <canvas id="paper" width="400" height="400">
      </canvas>
  </body>
  <script>
      let canvas = document.getElementById("paper")
       let x = 0;      มีการกำหนด x = 0 เป็นอ่าแนวราบ =0
      let ctx = canvas.getContext('2d')
         function draw(){
              ctx.clearRect(0, 0, canvas.width, canvas.height)
                 clearค่า ลบค่าก่อนออกประมาณนั้น
              ctx.fillStyle = 'red'
              ctx.fillRect(x, 10, 5, 5) = ว่าจะเริ่มที่     x = 0 y =10 ขนาด5 5
              x = x + 2;
         }
         setInterval(draw, 500) execute draw ที่ ตำแหน่งx += 2ทุกๆ 500 milisec
   </script>
</html>
```

Syntax

JS

```
setInterval(code)
setInterval(code, delay)
```

CHAVATIK THORARIT Group 9 6610110066

HTML05 สิ่งที่เห็นชัดเจนที่สุดหลังrun คือ โคตรsmooth ดูด

ไม่ทะลุจอละ มันเด้งไปเด้งกลับ if statement ทำให้มันเด้งไปแด้งมา

```
<html>
    <head>
    </head>
    <body>
        <canvas id="paper" width="400" height="400">
        </canvas>
    </body>
    <script>
        let canvas = document.getElementById("paper")
        let x = 0;
        let direction = 1;
        let ctx = canvas.getContext('2d')
        let lastFrameStamp = new Date().getTime()   เก็บtime stamp → ???
        function draw(){
            ctx.clearRect(0, 0, canvas.width, canvas.height)
            ctx.fillStyle = 'red'
            ctx.fillRect(x, 10, 5, 5)
            if(x >= 400)     ส่วนที่ทำให้เด้งจัดเด้งกลับคือส่วนนี้เลย
                direction = -1
            else if(x <= 0)
                direction = 1
            let now = new Date().getTime()
            let diff = now - lastFrameStamp →NOw you a little bit knew
            let s = (50 * diff)/1000 //s = vt →calculate ระยะที่เคลื่อนที่
            x = x + (direction * s)
            window.requestAnimationFrame(draw)--> Update next fraame
            lastFrameStamp = now
        }
        window.requestAnimationFrame(draw)
    </script>
</html>
```

CHAVATIK THORARIT Group 9 6610110066

HTML06

```
<html>
   <head>
   </head>
   <body>
       <canvas id="paper" width="400" height="400">
       </canvas>
   </body>
   <script>
       class Sprite { →  มีการสร้าง Class ละ
           constructor(name){
               this.name = name
           }
           greet(){
               console.log('My name is', this.name)
           }
       }
       let s1 = new Sprite("A1") ก็คือ ถ้าใช้greetfunction จะprint 'My name is A 1'
       let s2 = new Sprite("A2")------------- My name is A 2
       s2.greet()
       s1.greet()

   </script>
</html>
```

HTML07

ก็คือ 06 version advanced ขึ้นอีก step เริ่มเอ๋อละ
run แล้วมี2 player 1 red and green วิ่งขนานกันไป มีความรู้สึกเหมือนว่ามันวิ่งไม่เชิงขนานขนาดนั้น มีความเหลื่อมสลับไปสลับมา แล้วเด้งกลับได้ด้วยนะ

มีดูdynamic มากกว่า html06

ลื่นกว่า html06

เอาClassมาวาดภาพ moveandpaint

```
<html>
   <head>
   </head>
   <body>
       <canvas id="paper" width="400" height="400">
       </canvas>
   </body>
   <script>
```

```
        let canvas = document.getElementById("paper")
        let ctx = canvas.getContext('2d')

        class Sprite {
            constructor(color, y){
                this.color = color
                this.y = y
                this.x = 0
                this.direction = -1
            }
            moveAndPaint(diff){
                ctx.fillStyle = this.color
                ctx.fillRect(this.x, this.y, 5, 5)
                if(this.x >= 400)
                    this.direction = -1
                else if(this.x <= 0)
                    this.direction = 1
                let s = (50 * diff)/1000 //s = vt
                this.x = this.x + (this.direction * s)
            }
        }

        let s1 = new Sprite('red', 10) คาดเป็นส่วนที่ทำให้มันดูเหลื่อมๆกัน
        let s2 = new Sprite('green', 30)

        let lastFrameStamp = new Date().getTime()
        function draw(){
            ctx.clearRect(0, 0, canvas.width, canvas.height)
            let now = new Date().getTime()
            let diff = now - lastFrameStamp →cal distance send to Class
            s1.moveAndPaint(diff)
            s2.moveAndPaint(diff)
            window.requestAnimationFrame(draw)
            lastFrameStamp = now
        }
        window.requestAnimationFrame(draw)

    </script>
</html>
```

CHAVATIK THORARIT Group 9 6610110066

HTML08

red greenวิ่งเหมือนรถสวนlaneกัน โดยเขียวจะช้ากว่าแดงพอสมควร เด้งกลับได้เหมือนเดิม

น่าจะเป็นพื้นฐานของการทำanimationที่ดีเลยทีเดียวสำหรับ08ณตอนนี้ เช่น เราควรกำหนดattribute ยังไง ตัวแปร การคำนวณ ขนาด ตำแหน่ง เป็นต้นประมาณนี้

```
<html>
   <head>
   </head>
   <body>
       <canvas id="paper" width="400" height="400">
       </canvas>
   </body>
   <script>
       let canvas = document.getElementById("paper")
       let ctx = canvas.getContext('2d')

       class Sprite {
           constructor(color, x, y, direction, speed){ มีการadd speedเข้ามาแบ้ว
               this.color = color
               this.y = y
               this.x = x
               this.direction = direction
               this.speed = speed
           }
           moveAndPaint(diff){
               ctx.fillStyle = this.color
               ctx.fillRect(this.x, this.y, 5, 5)
               if(this.x >= 400)
                   this.direction = -1
               else if(this.x <= 0)
                   this.direction = 1
               let s = (this.speed * diff)/1000 //s = vt
               this.x = this.x + (this.direction * s)
           }
       }

       let s1 = new Sprite('red', 0, 10, 1, 50) -> color x y direction and speed
       let s2 = new Sprite('green', 40, 30, -1, 30)-->color x y direction and speed

       let lastFrameStamp = new Date().getTime()
       function draw(){
           ctx.clearRect(0, 0, canvas.width, canvas.height)
           let now = new Date().getTime()
```

```
            let diff = now - lastFrameStamp
            s1.moveAndPaint(diff)
            s2.moveAndPaint(diff)
            window.requestAnimationFrame(draw)
            lastFrameStamp = now
        }
        window.requestAnimationFrame(draw)

    </script>
</html>
```

HTML 09

ฟรุ้งฟริ้งมาก ดึงดูดสายตา ดูเพลินๆ เด้งกลับได้เหมือนเดิม ตอนเด้งเหมือนมันจะflipตัวก่อนเด้งกลับด้วย

เพิ่มsprite จากเดิม re and green
NOW red green blue Color code(#ee7722) & Color code(#00ffff)

ตำแหน่ง ทิศทาง คามเร็ว ค่อนค้างหลากหลาย

```
let sprites = [
            new Sprite('red', 0, 10, 1, 50),
            new Sprite('green', 40, 30, -1, 30),
            new Sprite('blue', 200, 50, 1, 70),
            new Sprite('#ee7722', 80, 70, -1, 100),
            new Sprite('#00ffff', 300, 90, -1, 150),
```

HTML 10 โคตรดี

ตอนนี้มันเด้งไปมาเ**แบบอิสระได้แล้ว** เหมือนเกม ปิงปอง

ทิศทางไม่เป็นpattern มีการใช้มุมองศา ตำแหน่งในการมาคำนวณ การเด้งของลูกบอลจุด

การกำหนดขอบเขตของcanva mathmin max กันไม่ให้ บอลจุดไปเลยทะลุ มั้งนะ

```
class Sprite {
        constructor(color, x, y, heading, speed) {
            this.color = color
            this.y = y
            this.x = x
            this.heading = heading
            this.speed = speed
        }
        moveAndPaint(diff) {
```

```
            if (this.x >= canvas.width) {
                if (this.heading > 0)
                    this.heading += 90
                else
                    this.heading -= 90
            } else if (this.x <= 0) {
                if (this.heading > 0)
                    this.heading -= 90
                else
                    this.heading += 90
            }
            if (this.y >= canvas.height || this.y <= 0) {
                this.heading *= -1
            }

            let s = (this.speed * diff) / 1000 //s = vt
            let degree = Math.PI * this.heading / 180;
            this.x = this.x + Math.cos(degree) * s
            this.y = this.y + Math.sin(degree) * s

            this.x = Math.min(this.x, canvas.width)
            this.x = Math.max(this.x, 0)
            this.y = Math.min(this.y, canvas.height)
            this.y = Math.max(this.y, 0)

            ctx.fillStyle = this.color
            ctx.fillRect(Math.round(this.x), Math.round(this.y), 5, 5)
        }
    }
```