

Canvas คืออะไร



Canvas เป็น Feature หรือคุณสมบัติ
หนึ่งของ HTML5 สำหรับจัดการงานด้านก
ราฟิก เช่น การเขียนตัวอักษร , วาดเส้น ,
วาดกราฟ , แสดงผลรูปภาพ เป็นต้น

Canvas คืออะไร



เราสามารถเรียกใช้งานโดยใช้
<canvas> โดย canvas เป็นพื้นที่ว่าง
สำหรับวาดหรือเขียนกราฟิกอะไรลงไป
ก็ได้ และสามารถนำคำสั่ง JavaScript
มาควบคุมการทำงานของกราฟิกได้ด้วย

การใช้งาน Canvas

```
<canvas id="ไอดี"  
      width="ขนาดความกว้าง"  
      height="ขนาดความสูง">  
</canvas>
```

การใช้งาน Canvas

หากไม่กำหนด width และ height แล้ว canvas จะแสดงผลโดยมีความกว้าง 300px และความสูง 150px เป็นค่าเริ่มต้น

```
<canvas id="ไอดี"
```

```
width="ขนาดความกว้าง"
```

```
height="ขนาดความสูง">
```

```
</canvas>
```

Canvas Context



Context คือ Object ที่ใช้เรียกฟังก์ชันการทำงานต่างๆ เพื่อสั่งให้ Canvas แสดงผลลัพธ์ตามที่ต้องการ เช่น สีพื้นหลัง , วาดเส้น , ตัวอักษรหรือรูปภาพ เปรียบเสมือนผู้กันสำหรับวาดหรือระบายสีลงใน Canvas นั้นเอง


โครงสร้างคำสั่ง

```
const canvas = document.getElementById("canvas");  
const context = canvas.getContext("2d");
```

โครงสร้างคำสั่ง

```
const canvas = document.getElementById("canvas");
```

```
const context = canvas.getContext("2d");
```



เป็นการดึง Object context มาใช้งาน โดยพารามิเตอร์ 2d หมายถึง การเรียกใช้งาน Object context ให้จัดการกราฟิกในรูปแบบ 2 มิติเพื่อแสดงผลออกสู่ Canvas ซึ่ง Web Browser ทุกตัวจะรองรับการแสดงผลกราฟิก 2 มิตินี้

พิกัดของ Canvas

เมื่อประกาศแท็ก <canvas> โดยกำหนดความกว้างและความสูงด้วย width และ height เป็นหน่วย pixels แล้ว เราสามารถอ้างอิงพิกัดภายในพื้นที่ของ Canvas ได้ในรูปแบบคู่พิกัด (x , y)

- x คือ พิกัดในแนวนอนไปตามความกว้างของ Canvas
- y คือ พิกัดในแนวตั้งไปตามความสูงของ Canvas

การลากเส้นบน Canvas

Context ของ Canvas รองรับการทำงานในส่วนของการวาดเส้น เรียกว่า Path การสร้างและวาดเส้น Path

- `moveTo(x,y)` เป็นฟังก์ชันสำหรับกำหนดจุดเริ่มต้นที่จะลากเส้น
- `lineTo(x,y)` เป็นฟังก์ชันสำหรับสั่งให้ลากเส้นจากจุดเริ่มต้นไปยังจุดที่กำหนด โดยแต่ละเส้นจะมีขนาด 1 pixel และเปลี่ยนขนาดของเส้น Path ด้วย `lineWidth`

การลากเส้นบน Canvas

- **strokeStyle** เป็นการกำหนดสีของเส้นที่จะวาดออกสู่ Canvas
- **stroke()** เป็นฟังก์ชันเพื่อกำหนดให้วาดเส้นทุกเส้นของ Path ที่ได้ใช้คำสั่ง `moveTo()` และ `lineTo()` วาดไว้ ซึ่งจะทำให้เส้นทั้งหมดของ Path ที่วาดไว้แสดงบน Canvas ตามสีที่กำหนดไว้ใน `strokeStyle`

การลากเส้นบน Canvas

- `context.beginPath()` เป็นคำสั่งสำหรับขึ้น Path ใหม่เพื่อแยกเส้นจาก Path เดิม จากนั้นจึงเรียกใช้ฟังก์ชัน `moveTo()` และ `lineTo()` แล้วเรียกใช้ฟังก์ชัน `stroke()` เพื่อวาดเส้นของ Path ใหม่ที่สร้างขึ้นตามต้องการ โดยไม่มีผลต่อ Path เดิม
- `context.closePath()` เป็นคำสั่งสำหรับจบการทำงานของ Path

การระบายสีภายใน Path

- **fillStyle** ใช้สำหรับกำหนดสีและรูปแบบของแรเงาของสีที่จะระบายลงไปยังพื้นที่ที่กำหนด โดยจะต้องกำหนดคุณสมบัตินี้ก่อนเรียกใช้งานฟังก์ชัน **fill()**
- **fill()** เป็นฟังก์ชันสำหรับระบายสีที่ถูกกำหนดด้วยคุณสมบัติ **fillStyle** ลงไปยังพื้นที่ภายใน Path

การวาดสี่เหลี่ยม

- `fillRect(x,y,ความกว้าง,ความสูง)` เป็นฟังก์ชันสำหรับสร้างพื้นที่สี่เหลี่ยมลงบน Canvas ที่พิกัด (x,y) ตามความกว้างและความสูงที่กำหนด พร้อมทั้งระบายสีในพื้นที่สี่เหลี่ยมนั้น

การวาดสี่เหลี่ยม

- `strokeRect(x,y,ความกว้าง,ความสูง)` เป็นฟังก์ชันสำหรับสร้างพื้นที่สี่เหลี่ยมลงบน Canvas ที่พิกัด (x,y) ตามความกว้างและความสูงที่กำหนด แต่จะไม่มีการระบายสีในพื้นที่สี่เหลี่ยมนั้น

การแสดงข้อความบน Canvas

- `fillText(ข้อความ , x , y)` เป็นฟังก์ชันสำหรับแสดงข้อความบน Canvas ที่ตำแหน่ง `x , y` สามารถระบุสีข้อความโดยใช้งานร่วมกับ `fillStyle`



การแสดงความความบน Canvas

- `strokeText(ข้อความ , x , y)` เป็นฟังก์ชันที่มีการทำงานคล้ายกับ `fillText` แต่ `strokeText` จะไม่เติมสีในพื้นที่ของข้อความการกำหนดสีต้องกำหนดผ่าน `strokeStyle` แทน `fillStyle`

การแสดงความความบน Canvas

- **font** เป็นการกำหนดรูปแบบฟอนต์สำหรับแสดงความความ โดยมีโครงสร้างคำสั่ง ดังนี้

font=[normal | italic][normal | bold][ขนาด][ชื่อฟอนต์]

การแสดงรูปภาพบน Canvas

- `drawImage(รูปที่จะแสดงผล , x , y)` ฟังก์ชันสำหรับแสดงผลรูปภาพตามขนาดจริงของภาพนั้นที่ตำแหน่ง x, y



การแสดงรูปภาพบน Canvas

- `drawImage`(รูปที่จะแสดงผล , x , y , ความกว้าง , ความสูง)
ฟังก์ชันสำหรับแสดงผลรูปภาพตามขนาดจริงของภาพนั้นที่
ตำแหน่ง x , y โดยสามารถปรับขนาดของภาพได้ตามขนาด
ความกว้างและความสูงที่กำหนด

requestAnimationFrame

การเล่น Animation บน HTML5 Canvas ในภาษา JavaScript จะมีการเรียกอัปเดตการเล่น Animation ผ่าน Method ที่ชื่อว่า **requestAnimationFrame** ซึ่งเป็น Method สำหรับจัดการ **FPS (Frames Per Second)** บน Web Browser ส่งผลให้ภาพ Animation นั้นมีความต่อเนื่องลื่นไหลมากยิ่งขึ้น



โครงสร้างคำสั่ง

`requestAnimationFrame(callback)`

ช่องทางการสนับสนุน



ช่องยูทูป : <https://www.youtube.com/c/KongRuksiamTutorial>



คอร์สเรียน : <https://www.udemy.com/user/kong-ruksiam/>



ซื้อของผ่าน Shopee : <https://shope.ee/3plB9kVnPd>



แฟนเพจ : <https://www.facebook.com/KongRuksiamTutorial/>