

แน่นอน! มาดูขอบเขตการทำงานของโค้ด HTML/CSS/JS ที่คุณส่งมาแบบละเอียดกันครับ

---

## 1 โครงสร้างหลัก (HTML)

```
<div class="wrapper">
  <div class="rainbow-bg"></div>

  <div class="container">
    <div class="title">ROCK AUDIO WAVE</div>
    <canvas id="waveCanvas" width="420" height="140"></canvas>
  </div>
</div>
```

- **.wrapper**

- เป็นตัวห่อหลักของเนื้อหา ทำหน้าที่จัดตำแหน่งและเป็น reference สำหรับ background และ container

- **.rainbow-bg**

- เป็น background แบบ gradient สายรุ้ง มี blur effect ทำให้เหมือนแสงเรืองรอง
- animation rainbowMove ทำให้ gradient เลื่อนไปทางขวาเรื่อย ๆ

- **.container**

- เป็นกล่องหลักสีดำกรอบหัวข้อและ canvas wave
- มี border และ radius ให้ดูเหมือนกล่องกระจกบาง ๆ

- **.title**

- แสดงข้อความ "ROCK AUDIO WAVE"
- ใช้ animation colorChange ให้สีตัวอักษรเปลี่ยนตามสายรุ้ง

- **canvas#waveCanvas**

- เป็นพื้นที่วาดคลื่นเสียงแบบ real-time (waveform)
  - ขนาด 420x140 px
- 

## 2 CSS (สไตล์)

### 2.1 การตั้งค่าพื้นฐาน

```
* { margin:0; padding:0; box-sizing:border-box; }
body { ... }
```

- รีเซ็ต margin/padding

- กำหนดให้ body เป็น black background, จัด center ทั้งแนวนอนและแนวตั้ง
- overflow: hidden ป้องกัน scrollbar

## 2.2 Background สายรุ้ง

```
.rainbow-bg {
  background: linear-gradient(...);
  background-size: 400% 400%;
  animation: rainbowMove 6s linear infinite;
  filter: blur(16px);
}
```

- ทำให้ gradient เลื่อนไปเรื่อย ๆ
- blur 16px ทำให้ดู glow

## 2.3 Container & Title

```
.container { ... }
.container::before { ... }
.title { ... }
@keyframes colorChange { ... }
```

- .container::before เป็น border โปร่งแสง
- .title เปลี่ยนสีเป็นสายรุ้งทุก 1 วินาที

## 2.4 Canvas

```
canvas { background: transparent; }
```

- ทำให้ canvas มีพื้นหลังใสเพื่อให้เห็น rainbow glow ด้านหลัง

## 3 JavaScript (วาดคลื่นเสียง)

```
const canvas = document.getElementById("waveCanvas");
const ctx = canvas.getContext("2d");
let time = 0;
```

```
function drawWave() {
  ctx.clearRect(0, 0, canvas.width, canvas.height);

  const gradient = ctx.createLinearGradient(0, 0, canvas.width, 0);
  gradient.addColorStop(0, "red");
  gradient.addColorStop(0.25, "yellow");
  gradient.addColorStop(0.5, "lime");
  gradient.addColorStop(0.75, "cyan");
  gradient.addColorStop(1, "magenta");

  ctx.strokeStyle = gradient;
  ctx.lineWidth = 3;
  ctx.shadowBlur = 12;
  ctx.shadowColor = "rgba(255, 255, 255, 0.6)";
```

```

    ctx.beginPath();
    for (let x = 0; x < canvas.width; x++) {
      const rockNoise = Math.random() * 12;
      const y = canvas.height/2 + Math.sin(x*0.05 + time)*40 + Math.sin(x*0.15
+ time*1.5)*20 + rockNoise;
      if (x === 0) ctx.moveTo(x, y);
      else ctx.lineTo(x, y);
    }
    ctx.stroke();

    time += 0.12;
    requestAnimationFrame(drawWave);
  }
}

drawWave();

```

### รายละเอียดการทำงาน

1. `clearRect` → ล้าง canvas ก่อนวาดใหม่ทุก frame
2. Gradient stroke → คลื่นมีสีสายรุ้ง
3. `shadowBlur` + `shadowColor` → ทำให้คลื่นมี glow
4. for loop x → คำนวณค่า y ของคลื่นที่แต่ละพิกเซล
  - มี 3 ส่วนหลัก:
    - `Math.sin(x*0.05 + time) * 40` → sine wave หลัก
    - `Math.sin(x*0.15 + time*1.5) * 20` → sine wave เสริม
    - `Math.random() * 12` → noise ทำให้คลื่นดู “rocky”
5. `requestAnimationFrame(drawWave)` → ทำให้คลื่นเคลื่อนไหวต่อเนื่อง

## 4 ขอบเขตการทำงาน (Scope)

ส่วน	ขอบเขตการทำงาน
<code>.rainbow-bg</code>	แสดง gradient สายรุ้งและเคลื่อนไหวเป็น glow background
<code>.container</code>	เป็นกล่องหลัก ครอบ title + canvas มี border โปร่งใส
<code>.title</code>	แสดงชื่อ “ROCK AUDIO WAVE” พร้อมเปลี่ยนสีตามสายรุ้ง
<code>canvas#waveCanvas</code>	วาดคลื่นเสียง (waveform) แบบ animated ด้วย gradient และ noise
JavaScript	คำนวณค่า y ของคลื่น, ทำ animation, ใช้ sine + noise + gradient + shadow

💡 สรุป:

- ระบบนี้เป็น visualizer แบบสายรุ้ง

- ไม่ใช่ audio real-time จริง (ไม่ได้ดึงเสียงจากไฟล์หรือไมโครโฟน)
- แสดง คลื่นเสมือนแบบสุ่ม ให้ดูเหมือนคลื่นเสียง rock ที่มี dynamism

---

ถ้าคุณอยาก ผมสามารถ วาดเพิ่มให้คลื่นตอบสนองเสียงจริง ๆ จากไมโครโฟน หรือไฟล์เพลงได้ด้วย JavaScript Web Audio API ซึ่งจะทำให้คลื่นไหวตามเสียงจริงแบบ “rock audio wave” จริง ๆ ได้ครับ

คุณอยากให้ผมทำตัวอย่างแบบนั้นไหม?