**สัปดาห์ที่ 5 โค้ด CSS Positioning**

โค้ด HTML และ CSS สำหรับอธิบายเรื่อง **CSS Position** ทั้ง 5 แบบ โดยใช้การแสดงข้อมูลรถยนต์เป็นตัวอย่าง และแยกเป็น 5 ไฟล์ ให้นักศึกษาสามารถคัดลอกโค้ดไปเขียนใน Visual Studio Code

**💡 สรุปการทำงานของ Position ทั้ง 5 แบบ**

|  |  |  |  |
| --- | --- | --- | --- |
| **Position** | **อ้างอิงจาก** | **Document Flow** | **พฤติกรรมเมื่อ Scroll** |
| **Static** | ตัวเอง (ปกติ) | **อยู่ใน** Flow | เลื่อนตามปกติ |
| **Relative** | ตำแหน่งเดิมของตัวเอง | **อยู่ใน** Flow | เลื่อนตามปกติ |
| **Absolute** | Parent ตัวแรกที่ไม่ใช่ static | **หลุดออก** จาก Flow | เลื่อนตามปกติ (แต่ตำแหน่งอ้างอิงกับ Parent) |
| **Fixed** | Viewport (หน้าจอ) | **หลุดออก** จาก Flow | **ไม่เลื่อน** (ติดกับหน้าจอ) |
| **Sticky** | Viewport/Parent | **อยู่ใน** Flow จนถึงจุดที่กำหนด | เป็น relative -> เป็น fixed -> เป็น relative อีกครั้ง (เมื่อพ้น Parent) |

**🚘 1. position-static.html และ static.css: Static (ตำแหน่งเริ่มต้น)**

**Static** คือการจัดตำแหน่งเริ่มต้นของทุกองค์ประกอบ โดยจะอยู่ใน **Document Flow** ตามลำดับการเขียนโค้ด การใช้คุณสมบัติ top, bottom, left, right จะ **ไม่มีผล** ใดๆ

**📄 position-static.html**

|  |
| --- |
| <!DOCTYPE html>  <html lang="th">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <title>CSS Position: Static</title>  <link rel="stylesheet" href="static.css">  </head>  <body>  <header>  <h1>ตำแหน่งรถยนต์: Static (ปกติ)</h1>  <p>องค์ประกอบจะจัดเรียงตามลำดับในโค้ด (Document Flow) โดยไม่สนใจค่า top/left/right/bottom</p>  </header>  <div class="car-container">  <div class="car-box car-box-static static-1">  <h3>รถคันที่ 1</h3>  <p><strong>ประเภท:</strong> Hatchback</p>  <p><strong>ตำแหน่ง:</strong> Static</p>  </div>    <div class="car-box car-box-static static-2">  <h3>รถคันที่ 2</h3>  <p><strong>ประเภท:</strong> Sedan</p>  <p><strong>ตำแหน่ง:</strong> Static</p>  </div>    <div class="car-box car-box-static static-3">  <h3>รถคันที่ 3</h3>  <p><strong>ประเภท:</strong> SUV</p>  <p><strong>ตำแหน่ง:</strong> Static</p>  </div>  </div>  </body>  </html> |

**🎨 static.css**

|  |
| --- |
| /\* การตั้งค่าพื้นฐาน \*/  body {  font-family: Arial, sans-serif;  margin: 20px;  background-color: #f4f4f9;  }  header {  text-align: center;  margin-bottom: 30px;  padding: 10px;  background-color: #e0e0e0;  }  .car-container {  border: 2px dashed #333;  padding: 15px;  background-color: #fff;  }  .car-box {  width: 200px;  height: 120px;  padding: 10px;  margin: 15px;  border: 1px solid #ccc;  background-color: #c9e4ff;  display: inline-block; /\* เพื่อให้เห็นการเรียงลำดับ \*/  }  /\* Static Positioning \*/  .car-box-static {  /\* นี่คือค่าเริ่มต้น ถ้าไม่กำหนด position องค์ประกอบทั้งหมดจะเป็น static \*/  position: static;  }  /\* ทดสอบ: กำหนดค่า top/left แต่จะไม่มีผลเพราะ position เป็น static \*/  .static-2 {  background-color: #a4c9ff;  top: 50px; /\* ไม่มีผล \*/  left: 100px; /\* ไม่มีผล \*/  border: 2px solid red;  } |

**📐 2. position-relative.html และ relative.css: Relative (เลื่อนจากตำแหน่งเดิม)**

องค์ประกอบจะยังคงอยู่ใน **Document Flow** แต่สามารถเลื่อนออกจาก **ตำแหน่งเดิม** ด้วยคุณสมบัติ top, bottom, left, right ได้ **ตำแหน่งเดิมของมันจะยังคงถูกสงวนไว้** ทำให้องค์ประกอบอื่นไม่เลื่อนมาแทนที่

**📄 position-relative.html**

|  |
| --- |
| <!DOCTYPE html>  <html lang="th">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <title>CSS Position: Relative</title>  <link rel="stylesheet" href="relative.css">  </head>  <body>  <header>  <h1>ตำแหน่งรถยนต์: Relative (เลื่อนจากเดิม)</h1>  <p>รถคันที่ 2 ถูกเลื่อนจากตำแหน่งเดิม <strong>30px</strong> ลงล่าง และ <strong>50px</strong> ขวา</p>  </header>  <div class="car-container">  <div class="car-box relative-1">  <h3>รถคันที่ 1</h3>  <p><strong>ประเภท:</strong> Truck</p>  <p><strong>ตำแหน่ง:</strong> Static</p>  </div>    <div class="car-box relative-2">  <h3>รถคันที่ 2</h3>  <p><strong>ประเภท:</strong> Convertible</p>  <p><strong>ตำแหน่ง:</strong> Relative (เลื่อน)</p>  </div>    <div class="car-box relative-3">  <h3>รถคันที่ 3</h3>  <p><strong>ประเภท:</strong> Minivan</p>  <p><strong>ตำแหน่ง:</strong> Static</p>  </div>  </div>  <div class="slider-example">  <h2>ทดสอบการปรับตำแหน่ง Relative</h2>  <p>เลื่อนจากซ้าย: <input type="range" min="-100" max="100" value="50" oninput="this.nextElementSibling.value=this.value; document.getElementById('relative-car').style.left = this.value + 'px'">  <output>50</output>px</p>    <div class="car-slider-base">  <div class="car-box car-slider" id="relative-car">  <h4>รถทดลองเลื่อน</h4>  <p>เลื่อนด้วย Relative</p>  </div>  </div>  </div>  <script>  // (ส่วนนี้ไม่มี JS สำหรับ Interactive แต่เพิ่มเพื่อให้ range input แสดงผลค่าที่เลือก)  </script>  </body>  </html> |

**🎨 relative.css**

|  |
| --- |
| /\* การตั้งค่าพื้นฐาน \*/  body {  font-family: Arial, sans-serif;  margin: 20px;  background-color: #f4f4f9;  }  header {  text-align: center;  margin-bottom: 30px;  padding: 10px;  background-color: #e0e0e0;  }  .car-container {  border: 2px dashed #333;  padding: 15px;  background-color: #fff;  height: 200px; /\* กำหนดความสูงเพื่อให้เห็นการเลื่อน \*/  }  .car-box {  width: 200px;  height: 120px;  padding: 10px;  margin: 15px;  border: 1px solid #ccc;  background-color: #c9e4ff;  display: inline-block;  }  /\* Relative Positioning \*/  .relative-1, .relative-3 {  position: static;  background-color: #c9e4ff;  }  /\* กล่องที่มีการเลื่อนตำแหน่ง \*/  .relative-2 {  position: relative; /\* กำหนดให้เป็น Relative \*/  top: 30px; /\* เลื่อนลงมา 30px จากตำแหน่งเดิม \*/  left: 50px; /\* เลื่อนไปทางขวา 50px จากตำแหน่งเดิม \*/  background-color: #ffc9a4;  border: 2px solid orange;  z-index: 10; /\* ให้รถคันนี้อยู่ด้านบน \*/  }  /\* Slider Example \*/  .slider-example {  margin-top: 50px;  padding: 20px;  border: 1px solid #0056b3;  background-color: #e9f7ff;  }  .car-slider-base {  width: 500px;  height: 150px;  border: 1px dashed #0056b3;  margin-top: 15px;  }  .car-slider {  position: relative; /\* ต้องเป็น Relative ถึงจะเลื่อนได้ \*/  background-color: #007bff;  color: white;  width: 150px;  height: 100px;  text-align: center;  line-height: 1.2;  padding: 10px;  margin: 0;  left: 50px; /\* ค่าเริ่มต้น \*/  } |

**🎯 3. position-absolute.html และ absolute.css: Absolute (วางตำแหน่งตาม Parent)**

องค์ประกอบจะ **หลุดออกจาก Document Flow** และตำแหน่งของมันจะอ้างอิงจาก **Parent องค์ประกอบแรกที่มี position เป็นอย่างอื่นที่ไม่ใช่ static** (เช่น relative, absolute, fixed, หรือ sticky) ถ้าหา Parent ที่มี position ไม่เจอ มันจะอ้างอิงกับ <html> หรือ <body> แทน

**📄 position-absolute.html**

|  |
| --- |
| <!DOCTYPE html>  <html lang="th">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <title>CSS Position: Absolute</title>  <link rel="stylesheet" href="absolute.css">  </head>  <body>  <header>  <h1>ตำแหน่งรถยนต์: Absolute (อ้างอิง Parent)</h1>  <p>รถคันที่ 2 (Absolute) จะอ้างอิงตำแหน่งกับ <strong>Container (Relative)</strong></p>  </header>  <div class="car-container-relative">  <h2>ศูนย์กลางการจัดส่ง (Parent: Relative)</h2>    <div class="car-box absolute-1">  <h3>รถคันที่ 1</h3>  <p><strong>ประเภท:</strong> Sedan</p>  <p><strong>ตำแหน่ง:</strong> Static</p>  </div>    <div class="car-box absolute-2">  <h3>รถคันที่ 2</h3>  <p><strong>ประเภท:</strong> \*\*Prototype\*\*</p>  <p><strong>ตำแหน่ง:</strong> Absolute (Top: 0, Right: 0)</p>  </div>    <div class="car-box absolute-3">  <h3>รถคันที่ 3</h3>  <p><strong>ประเภท:</strong> Wagon</p>  <p><strong>ตำแหน่ง:</strong> Static</p>  </div>  </div>  <div class="car-container-static">  <h2>ลานจอดรถ (Parent: Static)</h2>  <div class="car-box absolute-4">  <h4>รถคันที่ 4</h4>  <p><strong>ตำแหน่ง:</strong> Absolute (อ้างอิง Body)</p>  </div>  <p>กล่องนี้จะอ้างอิงตำแหน่งกับ body แทน เพราะ parent (ลานจอดรถ) เป็น static</p>  </div>  </body>  </html> |

**🎨 absolute.css**

|  |
| --- |
| /\* การตั้งค่าพื้นฐาน \*/  body {  font-family: Arial, sans-serif;  margin: 20px;  background-color: #f4f4f9;  height: 1000px; /\* เพิ่มความสูงเพื่อให้เห็นความแตกต่าง \*/  }  header {  text-align: center;  margin-bottom: 30px;  padding: 10px;  background-color: #e0e0e0;  }  .car-box {  width: 200px;  height: 120px;  padding: 10px;  margin: 15px;  border: 1px solid #ccc;  background-color: #c9e4ff;  display: inline-block;  }  /\* Absolute Positioning \*/  /\* Parent Container: กำหนด position: relative เพื่อเป็นจุดอ้างอิง \*/  .car-container-relative {  position: relative; /\* \*\*Key:\*\* องค์ประกอบ Absolute จะอ้างอิงกับกล่องนี้ \*/  border: 2px dashed #0056b3;  padding: 20px;  background-color: #fff;  margin-bottom: 50px;  height: 300px;  }  /\* กล่องที่มีการจัดตำแหน่ง Absolute \*/  .absolute-2 {  position: absolute; /\* หลุดจาก Document Flow และอ้างอิง Parent \*/  top: 0;  right: 0;  width: 150px;  height: 80px;  margin: 0; /\* ลบ margin เพื่อให้ตำแหน่งชัดเจน \*/  background-color: #ff6347; /\* สีส้มอมแดง \*/  color: white;  border: 2px solid darkred;  display: block; /\* Absolute element เป็น block โดยปริยาย \*/  }  /\* ตัวอย่าง Parent Static: จะอ้างอิงกับ Body แทน \*/  .car-container-static {  border: 2px dashed #333;  padding: 20px;  background-color: #f0f0f0;  /\* position: static; (ค่าเริ่มต้น) \*/  height: 200px;  }  .absolute-4 {  position: absolute;  top: 500px; /\* จะเริ่มนับจากขอบบนของ Body (หรือองค์ประกอบแรกที่ไม่ใช่ static) \*/  left: 250px;  background-color: #a4a4ff;  border: 2px solid purple;  } |

**🔒 4. position-fixed.html และ fixed.css: Fixed (ติดกับหน้าจอ)**

องค์ประกอบจะ **หลุดออกจาก Document Flow** และตำแหน่งของมันจะอ้างอิงกับ **Viewport (หน้าจอของผู้ใช้งาน)** เสมอ เมื่อผู้ใช้เลื่อนหน้าจอ (Scroll) องค์ประกอบนี้จะ **ยังคงอยู่ที่ตำแหน่งเดิม** บนหน้าจอ

**📄 position-fixed.html**

|  |
| --- |
| <!DOCTYPE html>  <html lang="th">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <title>CSS Position: Fixed</title>  <link rel="stylesheet" href="fixed.css">  </head>  <body>  <header>  <h1>ตำแหน่งรถยนต์: Fixed (ติดหน้าจอ)</h1>  <p>กรุณาลองเลื่อนหน้าจอลงและสังเกตกล่องสีส้มที่มุมล่างขวา</p>  </header>  <div class="content-area">  <p>ข้อมูลรถยนต์มากมาย...</p>  <div class="car-info">  <h3>รถยนต์โมเดล X ปี 2025</h3>  <p><strong>รายละเอียด:</strong> ดีไซน์ใหม่ล่าสุด เน้นความประหยับพลังงาน และเทคโนโลยีความปลอดภัยขั้นสูง</p>  <p><strong>ราคาเริ่มต้น:</strong> 1,500,000 บาท</p>  </div>  <div class="car-info">  <h3>รถยนต์โมเดล Y ปี 2024</h3>  <p><strong>รายละเอียด:</strong> รถยนต์ครอบครัวขนาดใหญ่ เน้นความสะดวกสบายภายในห้องโดยสาร และพื้นที่เก็บของ</p>  <p><strong>ราคาเริ่มต้น:</strong> 1,200,000 บาท</p>  </div>  <p>...เนื้อหาเยอะแยะมากมายเพื่อให้เกิดการเลื่อนหน้าจอ (Scroll)...</p>  <p>...เลื่อนลงมาอีกเพื่อดูการทำงานของ Fixed Position...</p>  <p>...Fixed element ใช้สำหรับเมนูนำทาง หรือ ปุ่มแจ้งเตือนด่วน...</p>  <p>...เมื่อ scroll องค์ประกอบ fixed จะไม่เลื่อนตาม...</p>  <p>...สิ้นสุดเนื้อหา...</p>  </div>  <div class="car-fixed-box">  <h4>ติดต่อเราด่วน!</h4>  <p>รถคันพิเศษพร้อมจัดส่ง!</p>  </div>  </body>  </html> |

**🎨 fixed.css**

|  |
| --- |
| /\* การตั้งค่าพื้นฐาน \*/  body {  font-family: Arial, sans-serif;  margin: 0;  background-color: #f4f4f9;  height: 1500px; /\* \*\*Key:\*\* กำหนดความสูงมากๆ เพื่อให้เกิด Scroll \*/  }  header {  text-align: center;  margin-bottom: 30px;  padding: 20px;  background-color: #e0e0e0;  }  .content-area {  width: 80%;  margin: 0 auto;  padding: 20px;  }  .car-info {  border: 1px solid #ccc;  padding: 15px;  margin-bottom: 20px;  background-color: #fff;  }  /\* Fixed Positioning \*/  .car-fixed-box {  position: fixed; /\* \*\*Key:\*\* ติดอยู่กับ Viewport (หน้าจอ) \*/  bottom: 20px; /\* ระยะห่างจากขอบล่างของหน้าจอ \*/  right: 20px; /\* ระยะห่างจากขอบขวาของหน้าจอ \*/  width: 150px;  padding: 15px;  background-color: #ff8c00; /\* สีส้ม \*/  color: white;  border: 3px solid #cc7000;  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);  text-align: center;  z-index: 1000; /\* ให้กล่องนี้อยู่ด้านบนสุด \*/  } |

**📌 5. position-sticky.html และ sticky.css: Sticky (ติดเมื่อ Scroll)**

องค์ประกอบจะถูกจัดตำแหน่งแบบ **Relative** จนกว่าจะถึงจุดที่กำหนด (เช่น top: 0) จากนั้นจะเปลี่ยนพฤติกรรมเป็นแบบ **Fixed** เพื่อยึดติดกับ Viewport จนกว่า Parent องค์ประกอบจะสิ้นสุดลง

**📄 position-sticky.html**

HTML

|  |
| --- |
| <!DOCTYPE html>  <html lang="th">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <title>CSS Position: Sticky</title>  <link rel="stylesheet" href="sticky.css">  </head>  <body>  <header>  <h1>ตำแหน่งรถยนต์: Sticky (ติดเมื่อ Scroll)</h1>  </header>  <p>เลื่อนหน้าจอลงเพื่อดูการทำงานของหัวข้อ "ตารางรุ่นรถยนต์" ที่จะติดอยู่ด้านบน</p>  <div class="scroll-test-area">  <h2>พื้นที่ทดสอบ Scroll</h2>  <div class="content-block">  <h3>ข้อมูลแนะนำรถยนต์ทั่วไป</h3>  <p>เลื่อนลงมาเรื่อยๆ จนกว่าจะถึงส่วนหัวข้อตาราง</p>  <p class="filler">...</p>  <p class="filler">...</p>  <p class="filler">...</p>  </div>    <div class="table-container">    <h3 class="sticky-header">ตารางรุ่นรถยนต์ (Position: Sticky)</h3>    <table class="car-table">  <thead>  <tr>  <th>รุ่น</th>  <th>สี</th>  <th>ราคา (บาท)</th>  </tr>  </thead>  <tbody>  <tr>  <td>Sedan S</td>  <td>ขาว</td>  <td>850,000</td>  </tr>  <tr>  <td>SUV C</td>  <td>ดำ</td>  <td>1,250,000</td>  </tr>  <tr>  <td>Hatchback R</td>  <td>แดง</td>  <td>790,000</td>  </tr>  <tr><td>Pickup T</td><td>น้ำเงิน</td><td>980,000</td></tr>  <tr><td>Van V</td><td>เทา</td><td>1,550,000</td></tr>  <tr><td>Sport Z</td><td>เหลือง</td><td>2,500,000</td></tr>  <tr><td>Electric E</td><td>เขียว</td><td>1,800,000</td></tr>  <tr><td>Sedan S (Premium)</td><td>ขาว</td><td>1,000,000</td></tr>  <tr><td>SUV C (Pro)</td><td>ดำ</td><td>1,400,000</td></tr>  <tr><td>Hatchback R (Sport)</td><td>แดง</td><td>850,000</td></tr>  </tbody>  </table>  </div>    <div class="content-block">  <h3>ส่วนสรุปท้ายตาราง</h3>  <p>เมื่อเลื่อนหน้าจอเลยส่วนตารางนี้ไป หัวข้อ Sticky ก็จะเลื่อนหายไปด้วย เพราะมันถูกจำกัดอยู่ภายใน Parent (table-container)</p>  <p class="filler">...</p>  <p class="filler">...</p>  <p class="filler">...</p>  </div>  </div>  </body>  </html> |

**🎨 sticky.css**

CSS

|  |
| --- |
| /\* การตั้งค่าพื้นฐาน \*/  body {  font-family: Arial, sans-serif;  margin: 0;  background-color: #f4f4f9;  }  header {  text-align: center;  margin-bottom: 30px;  padding: 20px;  background-color: #e0e0e0;  }  .scroll-test-area {  width: 80%;  margin: 0 auto;  padding: 20px;  background-color: #fff;  border: 2px solid #555;  }  .content-block {  padding: 10px 0;  }  .filler {  height: 100px; /\* เพิ่มช่องว่าง \*/  background-color: #f8f8f8;  line-height: 100px;  text-align: center;  margin: 10px 0;  border-bottom: 1px dashed #ccc;  }  /\* Sticky Positioning \*/  .table-container {  background-color: #e6f7ff;  padding: 10px;  border: 1px solid #00aaff;  /\* กล่องนี้เป็น Parent ของ Sticky Element \*/  }  .sticky-header {  position: sticky; /\* \*\*Key:\*\* กำหนดให้เป็น Sticky \*/  top: 0; /\* \*\*Key:\*\* จะติดอยู่เมื่อขอบบนของกล่องมาชนกับขอบบนของ Viewport \*/  background-color: #007bff;  color: white;  padding: 10px;  margin: 0 -10px; /\* ชดเชย padding ของ table-container \*/  text-align: center;  z-index: 10;  border-bottom: 3px solid #0056b3;  }  .car-table {  width: 100%;  border-collapse: collapse;  margin-top: 10px;  }  .car-table th, .car-table td {  border: 1px solid #ddd;  padding: 8px;  text-align: left;  }  .car-table th {  background-color: #f2f2f2;  } |