

## สัปดาห์ที่ 2 HTML5 Semantic Elements & Forms

### จุดประสงค์การเรียนรู้

หลังจากเรียนจบสัปดาห์นี้ นักศึกษาจะสามารถ

- อธิบายความหมายและประโยชน์ของ HTML5 Semantic Elements ได้
- เลือกใช้ Semantic Elements ให้เหมาะสมกับเนื้อหาแต่ละส่วนได้
- สร้าง HTML Forms พร้อม Input Types ต่างๆ ได้ถูกต้อง
- ใช้ Form Attributes เพื่อปรับปรุง User Experience ได้
- เข้าใจการทำงานของ Form Validation พื้นฐาน

### ส่วนที่ 1 HTML5 Semantic Elements

คำถาม ทำไมต้องใช้ Semantic Elements?

ลองนึกภาพว่านักศึกษาเข้าไปในห้องสมุดใหม่ที่ไม่เคยไปมาก่อน

- แบบที่ 1 หนังสือวางกระจัดกระจายไปทั่ว ไม่มีป้ายบอก ไม่มีการจัดหมวดหมู่
- แบบที่ 2 หนังสือจัดเรียงเป็นหมวดหมู่ มีป้ายชัดเจน แยกโซนนิยาย-วิชาการ-นิตยสาร

แบบไหนที่นักศึกษาจะหานั้งสือเลอจ่ายกว่ากัน?

#### 1.1 Semantic Elements คืออะไร?

Semantic แปลว่า "ความหมาย"

Semantic Elements = HTML Tags ที่ **บอกความหมาย** ของเนื้อหาข้างใน  
ไม่ใช่แค่กล่องใส่เนื้อหาเฉยๆ

#### ✗ ไม่ดี: ใช้ `<div>` ทั้งหมด

```
<div id="header">  
    <div id="nav">...</div>  
</div>  
<div id="content">  
    <div class="post">...</div>  
</div>  
<div id="footer">...</div>
```

##### ปัญหา:

- ไม่บอกความหมาย
- SEO ไม่ดี
- Accessibility ต่ำ
- อ่าน Code ยาก

#### ✓ ดี: ใช้ Semantic Elements

```
<header>  
    <nav>...</nav>  
</header>  
<main>  
    <article>...</article>  
</main>  
<footer>...</footer>
```

##### ข้อดี:

- บอกความหมายชัดเจน
- SEO ดีขึ้น
- Accessibility สูง
- อ่าน Code ง่าย

ตัวอย่างเปรียบเทียบ

การใช้ `<div>` ทั้งหมด (แบบที่ 1) ไม่ใช้ Semantic (ไม่บอกรความหมาย)

```
<!-- ✗ ไม่ใช้ Semantic (ไม่บอกรความหมาย) -->
<div id="header">
  <div id="nav">
    <a href="/">หน้าแรก</a>
  </div>
</div>
<div id="content">
  <div class="post">
    <h2>หัวข้อบทความ</h2>
    <p>เนื้อหา...</p>
  </div>
</div>
<div id="footer">
  <p>Copyright 2025</p>
</div>
```

การใช้ Semantic Elements (แบบที่ 2) ใช้ Semantic (บอกรความหมายชัดเจน)

```
<!-- ✓ ใช้ Semantic (บอกรความหมายชัดเจน) -->
<header>
  <nav>
    <a href="/">หน้าแรก</a>
  </nav>
</header>
<main>
  <article>
    <h2>หัวข้อบทความ</h2>
    <p>เนื้อหา...</p>
  </article>
</main>
<footer>
  <p>Copyright 2025</p>
</footer>
```

## ⌚ HTML5 Semantic Elements ที่ควรรู้จัก

Element	ความหมาย	ใช้เมื่อไหร่
<code>&lt;header&gt;</code>	ส่วนหัว	หัวเว็บไซต์, หัวบทความ, หัวส่วน
<code>&lt;nav&gt;</code>	เมนูนำทาง	ลิงก์สำหรับนำทางหลักๆ
<code>&lt;main&gt;</code>	เนื้อหาหลัก	เนื้อหาสำคัญของหน้า (ใช้ได้ 1 ครั้ง/หน้า)
<code>&lt;article&gt;</code>	เนื้อหาที่สมบูรณ์ในตัวเอง	บทความ, โพสต์บล็อก, ข่าว
<code>&lt;section&gt;</code>	กลุ่มนื้อหาที่เกี่ยวข้องกัน	แบ่งส่วนเนื้อหา มักมี heading
<code>&lt;aside&gt;</code>	เนื้อหาเสริม	Sidebar, โฆษณา, ลิงก์เกี่ยวข้อง
<code>&lt;footer&gt;</code>	ส่วนท้าย	ท้ายเว็บ, ท้ายบทความ
<code>&lt;figure&gt;</code>	รูปภาพพร้อมคำอธิบาย	รูป + caption
<code>&lt;figcaption&gt;</code>	คำอธิบายรูป	ใช้ใน <code>&lt;figure&gt;</code>

## โครงสร้างเว็บเพจแบบ Semantic



```
<!DOCTYPE html>
<html lang="th">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>บล็อกของฉัน</title>
</head>
<body>


<header>
<h1>บล็อกเทคโนโลยี</h1>
<p>แหล่งรวมความรู้ด้าน Web Development</p>


<nav>
<ul>
<li><a href="/">หน้าแรก</a></li>
<li><a href="/about">เกี่ยวกับเรา</a></li>
<li><a href="/contact">ติดต่อ</a></li>
```

```

</ul>
</nav>
</header>

<!-- เนื้อหาหลักของหน้า -->
<main>

    <!-- บทความที่ 1 -->
    <article>
        <header>
            <h2>รู้จักกับ HTML5 Semantic Elements</h2>
            <p>เขียนโดย <strong>นิกร</strong> | วันที่ 15 มกราคม 2025</p>
        </header>

        <section>
            <h3>HTML5 คืออะไร?</h3>
            <p>HTML5 เป็นเวอร์ชันล่าสุดของ HTML ที่เพิ่ม Semantic Elements เข้ามา...</p>
        </section>

        <section>
            <h3>ทำไนต้องใช้ Semantic?</h3>
            <p>การใช้ Semantic Elements ช่วยให้เว็บไซต์มีโครงสร้างที่ชัดเจน...</p>
        </section>

        <figure>
            
            <figcaption>รูปที่ 1 โครงสร้าง Semantic HTML</figcaption>
        </figure>
    </article>

    <!-- Sidebar เนื้อหาเสริม -->
    <aside>
        <h3>บทความแนะนำ</h3>
        <ul>
            <li><a href="#">CSS Flexbox เมื่องต้น</a></li>
            <li><a href="#">JavaScript สำหรับมือใหม่</a></li>
        </ul>
    </aside>

</main>

<!-- ส่วนท้ายของเว็บ -->
<footer>
    <p>&copy; 2025 บล็อกเทคโนโลยี. All rights reserved.</p>
    <nav>
        <a href="/privacy">นโยบายความเป็นส่วนตัว</a> |
        <a href="/terms">เงื่อนไขการใช้งาน</a>
    </nav>
</footer>

</body>
</html>

```

## เคล็ดลับการเลือกใช้ Semantic Elements

### 1. Article vs Section - ต่างกันอย่างไร?

คำถามง่ายๆ "ถ้าเอาส่วนนี้ออกไปแสดงที่อื่น มันยังเข้าใจได้ไหม?"

- ✓ เข้าใจได้ → ใช้ <article> (โพสต์บล็อก, ข่าว, รีวิวสินค้า)
- ✗ เข้าใจไม่ได้ ต้องอาศัยบริบทรอบข้าง → ใช้ <section> (แบ่งส่วนในบทความ)

```

<!-- ✓ Article เนื้อหาสมบูรณ์ในตัวเอง -->
<article>
  <h2>รีวิวมือถือ iPhone 15</h2>
  <p>เมื่อวานได้ลองใช้ iPhone 15...</p>
  <!-- สามารถนำไปแสดงในหน้าอื่นได้ -->
</article>

<!-- ✓ Section แบ่งส่วนในบทความ -->
<article>
  <h2>คุณมีการทำขั้นตอน</h2>

  <section>
    <h3>ส่วนผสม</h3>
    <ul><li>แป้ง 2 ถ้วย</li></ul>
  </section>

  <section>
    <h3>วิธีทำ</h3>
    <ol><li>ผสานแป้งกับน้ำ</li></ol>
  </section>
  <!-- ถ้าเรา section ตรงนี้นำไปแสดงคนเดียว คนอ่านจะไม่เข้าใจว่ากำลังพูดถึงอะไร -->
</article>

```

## 2. ควรใช้ `<div>` เมื่อไหร่?

- ✓ ใช้ `<div>` เมื่อ
  - ต้องการจัดกลุ่มเพื่อ styling (CSS) เท่านั้น
  - ไม่มี semantic element ที่เหมาะสม

```

<!-- ✓ ใช้ div สำหรับ styling -->
<article>
  <div class="card-container"> <!-- ใช้จัดกลุ่มเพื่อใส่ CSS -->
    <h2>หัวข้อ</h2>
    <p>เนื้อหา</p>
  </div>
</article>

```

## ❖ ประโยชน์ของ Semantic Elements

### 1. SEO ดีขึ้น (Search Engine Optimization)

- Google เข้าใจโครงสร้างเว็บไซต์ได้ดีขึ้น
- จัดอันดับการค้นหาได้แม่นยำกว่า

### 2. Accessibility สูงขึ้น (การเข้าถึงได้)

- Screen Reader (โปรแกรมอ่านหน้าจอสำหรับคนตาบอด) สามารถนำทางได้ง่าย
- ผู้ใช้สามารถข้ามไปยังส่วนที่ต้องการได้โดย

### 3. Code อ่านง่าย บำรุงรักษาง่าย

- นักพัฒนาคนอื่นเข้าใจ Code ได้เร็วขึ้น
- แก้ไข Debug ง่ายกว่า

### 4. Maintainability

- โครงสร้างชัดเจน ทำงานเป็นทีมได้ง่าย

## ส่วนที่ 2 HTML Forms

### 2.1 Forms ใช้ทำอะไร?

Forms คือ ช่องทางให้ผู้ใช้ส่งข้อมูลเข้ามา เช่น

- กรอกแบบฟอร์มลงทะเบียน
- เข้าสู่ระบบ (Login)
- ค้นหาข้อมูล
- ส่งข้อความติดต่อ
- สั่งซื้อสินค้า

### 2.2 โครงสร้างพื้นฐานของ Form

```
<form action="/submit" method="POST">  
  <!-- ใส่ input fields ตรงนี้ -->  
  <button type="submit">ส่งข้อมูล</button>  
</form>
```

Attributes สำคัญของ `<form>`

- `action` = URL ที่จะส่งข้อมูลไป
- `method` = วิธีส่งข้อมูล (GET หรือ POST)

### 2.3 Input Types ที่ควรรู้จัก

HTML5 มี Input Types มากมาย แต่ละแบบมีจุดประสงค์ต่างกัน

#### 1. Text Inputs (ข้อความ)

```
<!-- Input ธรรมดา -->  
<label for="username">ชื่อผู้ใช้</label>  
<input type="text" id="username" name="username">  
  
<!-- Email (มี validation อัตโนมัติ) -->  
<label for="email">อีเมล</label>  
<input type="email" id="email" name="email">  
  
<!-- Password (ช่องตัวอักษร) -->  
<label for="password">รหัสผ่าน</label>  
<input type="password" id="password" name="password">  
  
<!-- ข้อความหลายบรรทัด -->  
<label for="message">ข้อความ</label>  
<textarea id="message" name="message" rows="5"></textarea>
```

#### 2. Number & Date Inputs

```

<!-- ตัวเลข -->
<label for="age">อายุ</label>
<input type="number" id="age" name="age" min="1" max="120">

<!-- วันที่ -->
<label for="birthday">วันเกิด</label>
<input type="date" id="birthday" name="birthday">

<!-- เวลา -->
<label for="meeting">เวลาอัดหน้าย</label>
<input type="time" id="meeting" name="meeting">

```

### 3. Selection Inputs (เลือก)

```

<!-- Radio Buttons (เลือกได้ 1 อาย่าง) -->
<p>เพศ</p>
<input type="radio" id="male" name="gender" value="male">
<label for="male">ชาย</label>

<input type="radio" id="female" name="gender" value="female">
<label for="female">หญิง</label>

<!-- Checkboxes (เลือกได้หลายอย่าง) -->
<p>ความสนใจ</p>
<input type="checkbox" id="coding" name="interests" value="coding">
<label for="coding">เขียนโค้ด</label>

<input type="checkbox" id="design" name="interests" value="design">
<label for="design">ออกแบบ</label>

<!-- Dropdown (Select) -->
<label for="country">ประเทศ</label>
<select id="country" name="country">
  <option value="">-- เลือกประเทศ --</option>
  <option value="th">ไทย</option>
  <option value="us">สหรัฐอเมริกา</option>
  <option value="jp">ญี่ปุ่น</option>
</select>

```

### 4. Other Useful Inputs

```

<!-- ไฟล์ -->
<label for="avatar">อัปโหลดรูปโปรไฟล์</label>
<input type="file" id="avatar" name="avatar" accept="image/*">

<!-- สี -->
<label for="color">เลือกสี</label>
<input type="color" id="color" name="color" value="#ff0000">

<!-- URL -->
<label for="website">เว็บไซต์</label>
<input type="url" id="website" name="website">

<!-- โทรศัพท์ -->
<label for="phone">เบอร์โทร</label>
<input type="tel" id="phone" name="phone">

```

## 2.4 Form Attributes ที่ช่วยปรับปรุง UX

```
<form>
  <!-- Required (บังคับกรอก) -->
  <input type="text" name="fullname" required>

  <!-- Placeholder (ข้อความตัวอย่าง) -->
  <input type="email" name="email" placeholder="example@email.com">

  <!-- Pattern (กำหนดรูปแบบด้วย Regex) -->
  <input type="text" name="phone" pattern="[0-9]{10}"
    placeholder="0812345678">

  <!-- Min & Max Length -->
  <input type="password" name="password"
    minlength="8" maxlength="20">

  <!-- Disabled (ปิดการใช้งาน) -->
  <input type="text" name="country" value="Thailand" disabled>

  <!-- Readonly (อ่านได้อย่างเดียว) -->
  <input type="text" name="id" value="12345" readonly>

  <!-- Autofocus (ไฟกีสวัตโนมัติ) -->
  <input type="text" name="search" autofocus>

  <button type="submit">ส่ง</button>
</form>
```

## 2.5 ตัวอย่าง Form สมบูรณ์ แบบฟอร์มติดต่อ

```
<!DOCTYPE html>
<html lang="th">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>แบบฟอร์มติดต่อ</title>
  </head>
  <body>

    <header>
      <h1>ติดต่อเรา</h1>
    </header>

    <main>
      <section>
        <h2>ส่งข้อความถึงเรา</h2>

        <form action="/contact" method="POST">

          <!-- ชื่อ-นามสกุล -->
          <div>
            <label for="fullname">ชื่อ-นามสกุล *</label>
            <input type="text" id="fullname" name="fullname"
              required placeholder="ใส่ชื่อเดิม">
          </div>

          <!-- อีเมล -->
          <div>
            <label for="email">อีเมล *</label>
            <input type="email" id="email" name="email"
              required placeholder="example@email.com">
          </div>
        </form>
      </section>
    </main>
  </body>
</html>
```

```
</div>

<!-- ເບອຣໂທ -->
<div>
  <label for="phone">ເບອຣໂທຮັສພ໌</label>
  <input type="tel" id="phone" name="phone"
    pattern="[0-9]{10}" placeholder="0812345678">
  <small>ກຽກເລີກຕົວເລີກ 10 ພັດກ</small>
</div>

<!-- ທີ່ວ່າງ -->
<div>
  <label for="subject">ຫ້າຂໍອ *</label>
  <select id="subject" name="subject" required>
    <option value="">-- ເລືອກຫ້າຂໍອ --</option>
    <option value="general">ສອບຄາມທົ່ວໄປ</option>
    <option value="support">ຂອຄວາມຂ່າຍເຫຼືອ</option>
    <option value="feedback">ແສດງຄວາມຄິດເຫັນ</option>
  </select>
</div>

<!-- ຂໍອຄວາມ -->
<div>
  <label for="message">ຂໍອຄວາມ *</label>
  <textarea id="message" name="message" rows="5"
    required minlength="10"
    placeholder="ເຂົ້າໃຈຂໍອຄວາມຂອງນັກສຶກສາທີ່ນີ້... "></textarea>
</div>

<!-- ປິບຍອນເຈື້ອນໄຂ -->
<div>
  <input type="checkbox" id="agree" name="agree" required>
  <label for="agree">
    ຈັນຍອນຮັບ <a href="/terms">ເຈື້ອນໄຂການໃໝ່ງານ</a> *
  </label>
</div>

<!-- ຖຸ່ມສົ່ງ -->
<button type="submit">ສົ່ງຂໍອຄວາມ</button>
<button type="reset">ລັງຂໍອມຸລ</button>

</form>
</section>

<aside>
  <h3>ຂອງທາງດິດຕໍ່ອ່ນໆ</h3>
  <address>
    <p> Email contact@example.com</p>
    <p> Tel 02-123-4567</p>
    <p> ທີ່ອຸ້ນ 123 ຄະນະສູນວິທ ກຽມທະເພາ</p>
  </address>
</aside>
</main>

<footer>
  <p>&copy; 2025 My Website</p>
</footer>

</body>
</html>
```

## 2.6 Form Validation พื้นฐาน

HTML5 มีการตรวจสอบข้อมูลใน input (Validation) ในตัวแล้ว! ไม่ต้องเขียน JavaScript ตรวจสอบ

ตัวอย่าง Validation ที่ Browser ทำให้

```
<!-- 1. ตรวจสอบว่ากรอกครบใหม -->
<input type="text" name="name" required>

<!-- 2. ตรวจสอบรูปแบบอีเมล -->
<input type="email" name="email" required>
<!-- ถ้าไม่ใส @ จะขึ้น error อัตโนมัติ -->

<!-- 3. ตรวจสอบตัวเลข -->
<input type="number" name="age" min="18" max="100">
<!-- ไม่สามารถใส่อายุต่ำกว่า 18 ได -->

<!-- 4. ตรวจสอบความยาว -->
<input type="password" name="password" minlength="8">
<!-- ต้องมีอย่างน้อย 8 ตัวอักษร -->

<!-- 5. ตรวจสอบรูปแบบด้วย Pattern -->
<input type="text" name="zipcode" pattern="[0-9]{5}">
<!-- ต้องเป็นตัวเลข 5 หลักเท่านั้น -->
↳ เคล็ดลับ ใช้ attribute title เพื่อแสดงค่าแนะนำ
html
<input type="text" name="phone"
       pattern="[0-9]{10}"
       title="กรุณากรอกเบอร์โทรศัพท์ 10 หลัก">
```

## แบบฝึกหัดในชั้นเรียน

### แบบฝึกหัดที่ 1 สร้างหน้า Blog Post

โจทย์ สร้างหน้าเว็บบทความบล็อก โดยใช้ Semantic Elements ให้เหมาะสม

#### ข้อกำหนด

- ✓ มี header ของเว็บไซต์ (ชื่อบล็อก + navigation)
- ✓ มี main content (บทความ) ที่แบ่งเป็น sections
- ✓ มี sidebar (aside) แสดงบทความแนะนำ
- ✓ มี footer พร้อม copyright
- ✓ บทความต้องมี header, sections, figure (รูปภาพ), footer

#### คำถามช่วยคิด

- ส่วนไหนควรใช้ <article>?
- ส่วนไหนควรใช้ <section>?
- Navigation bar ควรอยู่ใน tag ไหน?

### แบบฝึกหัดที่ 2 สร้าง Contact Form

โจทย์ สร้างแบบฟอร์มลงทะเบียนเข้าร่วมอีเวนต์

#### ข้อกำหนด

- ✓ ชื่อ-นามสกุล (บังคับกรอก)
- ✓ อีเมล (บังคับกรอก + validate format)
- ✓ เบอร์โทร (validate เป็นตัวเลข 10 หลัก)
- ✓ เพศ (radio buttons)
- ✓ ช่วงอายุ (dropdown <18, 18-25, 26-35, >35)
- ✓ ความสนใจ (checkboxes อย่างน้อย 3 ตัวเลือก)
- ✓ ยินยอมเงื่อนไข (checkbox บังคับติ๊ก)
- ✓ ปุ่มส่งและปุ่มล้างข้อมูล

#### คำถามช่วยคิด

- Input type ไหนเหมาะสมกับแต่ละพิลาร์?
- Attribute ไหนที่จะทำให้ UX ดีขึ้น?
- จะ validate รูปแบบเบอร์โทรศัพท์ยังไง?