

แผนการสอน/แผนการเรียนรู้ภาคทฤษฎี

	แผนการสอน/การเรียนรู้ภาคทฤษฎี	หน่วยที่ 9
	ชื่อวิชา การสร้างเว็บไซต์	สัปดาห์ที่ 14
	ชื่อหน่วย Cascading Style Sheets (CSS)	4 ชั่วโมง

ชื่อเรื่อง Cascading Style Sheets (CSS)

หัวข้อเรื่อง

ด้านความรู้

1. ความรู้เบื้องต้นเกี่ยวกับ CSS
2. การเขียน CSS แบบ Inline
3. การเขียน CSS แบบ Embedded
4. การเขียน CSS แบบ External
5. Selectors
 - 1) การระบุ Selector ด้วยชื่อแท็ก
 - 2) การระบุ Selector ด้วย id Attribute
 - 3) การระบุ Selector ด้วย class Attribute
 - 4) การระบุ Selector แบบอื่นๆ
6. รูปแบบการกำหนดคุณสมบัติใน CSS

ด้านทักษะ

1. การเขียน CSS แบบ Inline
2. การเขียน CSS แบบ Embedded
3. การเขียน CSS แบบ External

ด้านคุณธรรม จริยธรรม

1. ความรับผิดชอบ
2. ความสนใจใฝ่รู้

สาระสำคัญ

CSS ย่อมาจากคำว่า Cascading Style Sheets โดยทั่วไปจะเรียกว่า CSS หรือ Style Sheets เป็นวิธีการกำหนดการจัดรูปแบบการแสดงผลเอกสาร HTML ในรูปแบบหรือ Style ของเนื้อหาในเอกสาร เช่นสีของข้อความ สีพื้นหลัง ประเภทตัวอักษรและการจัดวางข้อความ ซึ่งการกำหนดรูปแบบหรือ Style นี้ใช้หลักการของการแยกเนื้อหาเอกสาร HTML ออกจากคำสั่งที่ใช้ในการจัดรูปแบบการแสดงผล ทำให้การกำหนดรูปแบบของการแสดงผลเอกสารไม่ขึ้นอยู่กับเนื้อหาของเอกสาร การนำสไตล์ชีทหรือ CSS มาใช้ในการจัดรูปแบบลักษณะต่างๆ ของข้อมูลบนเว็บเพจ ร่วมกับ HTML นั้น เราสามารถทำได้ 3 วิธีด้วยกัน ดังนี้

1. Inline Style เป็นการกำหนดค่าสำหรับ Tag เฉพาะ Tag โดยใช้คำสั่ง Style แทนการใช้ Tag Attribute

2. Embedded Style Sheet โดยการสร้าง Style ในส่วน Head แล้วกำหนดค่าต่างๆ หากส่วน Body มี Tag ที่กำหนดไว้ในส่วน Style ก็จะได้การกำหนดค่าตามนั้น ซึ่งเหมาะกับการใช้ในเว็บเพจเดียว

3. External Style Sheet เป็นการสร้าง Style Sheet เป็นไฟล์ไว้ และถูกเรียกใช้ในไฟล์ HTML โดยใช้คำสั่ง Link rel ในส่วน Head ซึ่งเหมาะกับการใช้กับหลายๆ เว็บเพจ

สมรรถนะอาชีพประจำหน่วย

1. แสดงความรู้เกี่ยวกับ Cascading Style Sheets (CSS)
2. สร้างและใช้งาน CSS

จุดประสงค์การสอน/การเรียนรู้

• จุดประสงค์ทั่วไป / บูรณาการเศรษฐกิจพอเพียง

1. เพื่อให้มีความรู้พื้นฐานเกี่ยวกับ Cascading Style Sheets (CSS) และมีทัศนคติที่ดี (ด้านความรู้)

2. เพื่อให้มีทักษะการสร้าง Cascading Style Sheets (CSS) พื้นฐาน (ด้านทักษะ)

3. เพื่อให้มีเจตคติที่ดีต่อการเตรียมความพร้อมด้าน วัสดุ อุปกรณ์ และการปฏิบัติงานอย่างถูกต้อง สำเร็จภายในเวลาที่กำหนด มีเหตุและผลตามหลักปรัชญาของเศรษฐกิจพอเพียง (ด้านคุณธรรม จริยธรรม)

• จุดประสงค์เชิงพฤติกรรม / บูรณาการเศรษฐกิจพอเพียง

1. บอกความหมายและรูปแบบคำสั่งของ CSS ได้ (ด้านความรู้)

2. บอกวิธีการเขียน CSS แบบ Inline ได้ (ด้านความรู้)

3. บอกวิธีการเขียน CSS แบบ Embedded ได้ (ด้านความรู้)

4. บอกวิธีการเขียน CSS แบบ External ได้ (ด้านความรู้)

5. บอกวิธีการระบุ Selector ด้วยชื่อแท็ก ได้ (ด้านความรู้)

6. บอกวิธีการระบุ Selector ด้วย id Attribute ได้ (ด้านความรู้)

7. บอกวิธีการระบุ Selector ด้วย class Attribute ได้ (ด้านความรู้)

8. บอกวิธีการระบุ Selector แบบผสมชื่อแท็กและ class ได้ (ด้านความรู้)

9. บอกวิธีการระบุ Selector แบบ Contextual ได้ (ด้านความรู้)

10. บอกวิธีการระบุ Selector แบบรวมกลุ่ม (Grouping) ได้ (ด้านความรู้)

11. การเตรียมความพร้อมด้านการเตรียมวัสดุ อุปกรณ์ ผู้เรียนจะต้องกระจายงานได้ทั่วถึง และตรงตามความสามารถของสมาชิกทุกคน มีการจัดเตรียมสถานที่ สื่อ วัสดุ อุปกรณ์ไว้อย่างพร้อมเพียง (ด้านคุณธรรม จริยธรรม/บูรณาการเศรษฐกิจพอเพียง)

12. ความมีเหตุมีผลในการปฏิบัติงาน ตามหลักปรัชญาของเศรษฐกิจพอเพียง ผู้เรียนจะต้องมีการใช้หลักการเรียนรู้และเวลาที่เหมาะสมกับการเรียนรู้ (ด้านคุณธรรม จริยธรรม/บูรณาการเศรษฐกิจพอเพียง)

เนื้อหาสาระการสอน/การเรียนรู้

• ด้านความรู้ (ทฤษฎี)

1. ความรู้เบื้องต้นเกี่ยวกับ CSS (จุดประสงค์เชิงพฤติกรรมข้อที่ 1)

CSS ย่อมาจาก Cascading Style Sheets เป็นภาษาที่มีรูปแบบการเขียนเฉพาะ ใช้สำหรับตกแต่งเอกสาร HTML/ XHTML ให้มีรูปแบบ สี ตัวอักษร เส้นขอบ พื้นหลัง และระยะห่างต่างๆ ตามต้องการ ด้วยการกำหนดคุณสมบัติให้กับ Element ต่างๆ ของ HTML ซึ่งการสร้างเว็บเพจส่วนใหญ่จะต้องมีการทำงานหลายอย่างที่ต้องทำซ้ำๆ ซึ่งจะทำให้เกิดความไม่สะดวกและทำให้เสียเวลา เช่น การปรับแต่งคุณสมบัติของข้อความ แบบตัวอักษร ขนาดตัวอักษร หรือสีของตัวอักษร ซึ่งจะต้องมีการปรับหรือกำหนดค่าหลายครั้ง ช่วยลดการใช้ภาษา HTML ในการตกแต่งเอกสารเว็บเพจ ทำให้คำสั่งภายในเอกสาร HTML เหลือเพียงส่วนเนื้อหา ทำให้เข้าใจง่ายขึ้น การแก้ไขเอกสารทำได้ง่ายและรวดเร็ว โหลดเว็บเพจได้เร็ว เวลาแก้ไขหรือปรับปรุงทำได้ง่าย ไม่ต้องตามแก้ที่ HTML tag ต่างๆ ทั้งเอกสาร สามารถควบคุมการแสดงผลให้เหมือนกันหรือใกล้เคียงกันในหลาย Web Browser ทำให้เป็นเว็บไซต์ที่มีมาตรฐาน ปัจจุบันการใช้ attribute ของ HTML ตกแต่งเอกสารเว็บเพจ นั้นล้าสมัยแล้ว W3C แนะนำให้เราใช้ CSS แทน ดังนั้นหากเราใช้ CSS กับเอกสาร HTML ของเราก็จะทำให้เข้ากับเว็บเบราว์เซอร์ในอนาคตได้ดี

Cascading Style Sheets (CSS) คือภาษาที่ใช้กำหนดรูปแบบการแสดงผล HTML โดยสร้างขึ้นเพื่อใช้ร่วมกับ HTML นั่นคือ HTML ใช้สำหรับกำหนดโครงสร้างข้อมูล ส่วน CSS ใช้สำหรับกำหนดรูปแบบการแสดงผล ตัวอย่างเช่น

เอกสาร HTML

```
<!DOCTYPE html>
<html>
  <head>
    <title> การใช้งาน Cascading Style Sheets (CSS) </title>
    <meta charset= "UTF-8">
  </head>
  <body>
    <h1> การใช้งาน Cascading Style Sheets (CSS) </h1>
    <h3> 1. Inline Style </h3>
    <h3> 2. Embedded Style Sheet </h3>
    <h3> 3. External Style Sheet </h3>
  </body>
</html>
```

การกำหนดรูปแบบด้วย CSS

```
h3 {
  color: blue;
}
```

ในตัวอย่างข้างต้นเราใช้แท็ก <h3> เพื่อกำหนดหัวข้อเท่านั้น และให้ CSS กำหนดการแสดงผลสีน้ำเงิน จะเห็นได้ว่า CSS คือภาษาที่ใช้ในการกำหนดรูปแบบหรือสไตล์ (Style) สำหรับ HTML

การแยกการแสดงผลออกจากข้อมูลช่วยให้ง่ายต่อการเปลี่ยนแปลงรูปแบบในอนาคต เช่นถ้าต้องการเปลี่ยนสีให้กับหัวข้อในแท็ก <h3> ในทุกเว็บเพจ เราเพียงเปลี่ยนแปลงค่าที่ไฟล์ CSS เท่านั้น ทำให้ CSS เป็นที่นิยมอย่างมากในการจัดรูปแบบให้กับเว็บเพจในปัจจุบัน ซึ่งก่อนหน้านี้เราใช้ตารางในการจัดเลย์เอาต์และใช้พร็อพเพอร์ตี้ของ HTML ในการจัดรูปแบบ ซึ่งยุ่งยากต่อการเปลี่ยนรูปแบบของทุกเว็บเพจในเว็บไซต์

การนำสไตล์ชีทหรือ CSS มาใช้ในการจัดรูปแบบลักษณะต่างๆ ของข้อมูลบนเว็บเพจ ร่วมกับ HTML นั้น เราสามารถทำได้ วิธีด้วยกัน ดังนี้ 3

1. Inline Style เป็นการกำหนดค่าสำหรับ Tag เฉพาะ Tag โดยใช้คำสั่ง Style แทนการใช้ Tag Attribute
2. Embedded Style Sheet โดยการสร้าง Style ในส่วน Head แล้วกำหนดค่าต่างๆ หากส่วน Body มี Tag ที่กำหนดไว้ในส่วน Style ก็จะได้การกำหนดค่าตามนั้น ซึ่งเหมาะกับการใช้ในเว็บเพจเดียว
3. External Style Sheet เป็นการสร้าง Style Sheet เป็นไฟล์ไว้ และถูกเรียกใช้ในไฟล์ HTML โดยใช้คำสั่ง Link rel ในส่วน Head ซึ่งเหมาะกับการใช้กับหลายๆ เว็บเพจ

รูปแบบคำสั่ง CSS

คำสั่งของ CSS ประกอบด้วย selector, attribute และ value โดยมีโครงสร้างคำสั่งดังนี้

```
selector { attribute: value ;}
```

ตัวอย่าง

```
p {color: yellow;  
font-size: 16px;  
}
```

โดยที่

p	เป็นselector
color และ font-size	เป็น attribute
yellow และ 16px	เป็น value

2 การเขียน .CSS แบบ Inline (จุดประสงค์เชิงพฤติกรรมข้อที่ 2)

การเขียน CSS แบบ Inline เป็นการแทรกคำสั่ง CSS ลงในส่วนของแท็กนั้นๆ โดยทำหน้าที่เป็นเสมือนแอททริบิวต์ของแท็กนั้น มีรูปแบบดังนี้

```
<tag style="attribute: value; attribute: value; . . . ">
```

แอททริบิวต์ style ใช้ในการกำหนดค่าสไตล์ต่างๆ ซึ่งจะอยู่ในรูปของ “attribute: value” ซึ่งถ้ามีมากกว่าหนึ่งชุดจะคั่นด้วยเครื่องหมาย (;) Semicolon ตัวอย่างการใช้คำสั่ง

```
<!DOCTYPE html>
<html>
  <head>
    <title> การเขียน CSS แบบ Inline </title>
    <meta charset= "UTF-8">
  </head>
  <body>
    <h1 style= "color: blue">Cascading Style Sheets (CSS)</h1>
    <p style= "font-weight: bold; font-style: italic; font-size: 24px;">การกำหนดรูปแบบด้วย Inline Style</p>
  </body>
</html>
```

ในตัวอย่างข้างต้นเรากำหนดสไตล์ให้กับแท็ก <h1> โดยให้ข้อความเป็นสีฟ้าด้วยรูปแบบ color ส่วนการกำหนดรูปแบบให้กับแท็ก <p> เป็นการกำหนดสไตล์ที่มีหลายรูปแบบ

การเขียนแบบ Inline ไม่เป็นที่นิยมเนื่องจากต้องเขียนโค้ด CSS ลงไปในแท็ก HTML ดังนั้นจึงไม่สามารถนำสไตล์ไปประยุกต์ใช้กับแท็กอื่นๆ ได้ เช่นถ้าต้องระบุค่าสีให้กับทุกๆ แท็ก <h1> ในทุกๆ เว็บเพจ และมีความยุ่งยากในการเปลี่ยนสีภายหลัง

.3การเขียน CSS แบบ Embedded (จุดประสงค์เชิงพฤติกรรมข้อที่ 3)

การเขียน CSS แบบ Embedded เป็นการเขียน CSS ลงไปในไฟล์ HTML เช่นเดียวกับการเขียน CSS แบบ Inline ซึ่งจะเขียนไว้ในส่วนของ <head> โดยมีรูปแบบดังนี้

```
<style type= "text/css"> คำสั่ง CSS </style>
```

เช่น

```
<head>
  <title>.....</title>
  <style type= "text/css">
    selector {
      attribute: value;
      attribute: value;
      ...
    }
  </style>
</head>
```

การกำหนดรูปแบบต่างๆ จะถูกกำหนดไว้ภายในแท็ก <style> ซึ่งอยู่ภายในแท็ก <head> อีกที การกำหนดรูปแบบจะเริ่มต้นด้วยการระบุแท็กที่ต้องการ (selector) จากนั้นจึงเป็นรูปแบบหรือ attribute: value ต่างๆ ภายในเครื่องหมายปีกกา โดยรูปแบบแต่ละชุดจะถูกคั่นด้วยเครื่องหมาย (;) Semicolon

ตัวอย่างการใช้คำสั่ง

```
<!DOCTYPE html>
<html>
  <head>
    <title> การเขียน CSS แบบ Embedded </title>
    <meta charset= "UTF-8">
    <style type= "text/css">
      p {color: blue;
        font-size: 16px;
        font-family: sans-serif;
      }
      span {
        color: #fd0000;
        font-style: italic;
        text-decoration: underline;
      }
    </style>
  </head>
  <body>
    <p>Cascading Style Sheets (CSS)</p>
    <p>หัวข้อที่ 2<span> การเขียน CSS แบบ Embedded </span> การเขียน
    CSS แบบ Embedded การกำหนดรูปแบบต่างๆ จะถูกกำหนดไว้ภายในแท็ก style ซึ่งอยู่ภายใน
    แท็ก head อีกที การกำหนดรูปแบบจะเริ่มต้นด้วยการระบุแท็กที่ต้องการ (Selector) จากนั้นจึงเป็น
    สไตล์ "attribute: value" ต่างๆ ภายในเครื่องหมายปีกกา แต่ละชุดจะถูกคั่นด้วยเครื่องหมาย
    Semicolon</p>
  </body>
</html>
```

จากตัวอย่างคำสั่ง CSS ทั้งหมดจะเขียนไว้ในกรอบ <style type= "text/css"> ... </style> ดังที่กล่าวไว้แล้วก่อนหน้านี้ สังเกตคำสั่งที่กำหนดให้แท็ก <p>...</p> ในเอกสารนี้ กำหนดเพียงครั้งเดียวแต่สามารถเปลี่ยนให้ข้อความที่อยู่ในแท็ก <p> ทั้งเอกสารได้ทั้งหมด

การเขียน CSS แบบ Embedded ไม่เป็นที่นิยมเพราะต้องเขียนรวมอยู่ในไฟล์ HTML เช่นเดียวกับการเขียน CSS แบบ Inline หากจะนำไปใช้กับไฟล์ HTML อื่นๆ ก็ต้องคัดลอกคำสั่งต่างๆ ที่ต้องการ ดังนั้นการเปลี่ยนแปลงแก้ไขจึงค่อนข้างยุ่งยาก เพราะต้องตามแก้ไขในทุกๆ ไฟล์

.4การเขียน CSS แบบ External (จุดประสงค์เชิงพฤติกรรมข้อที่ 4)

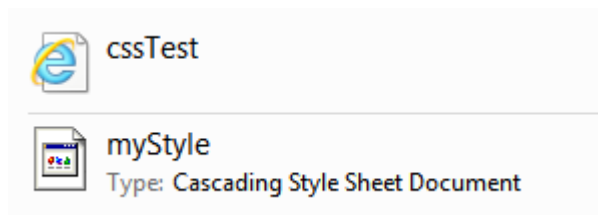
การเขียน CSS แบบ External เป็นการเขียน CSS ไว้ภายนอกเอกสาร HTML เป็นลักษณะ External Style Sheet ซึ่งสามารถทำได้ด้วยการสร้างไฟล์เอกสาร CSS ที่มีนามสกุล .css ที่ภายในเขียนคำสั่ง CSS ไว้ทั้งหมดไว้อีกหนึ่งไฟล์ เมื่อต้องการให้รูปแบบ CSS นี้มีผลกับไฟล์เอกสาร HTML ใดก็สามารถเชื่อมโยงมายังไฟล์นั้นๆ ได้ทันที ด้วยคำสั่งดังนี้

```
<link rel= "stylesheet" type= "text/css" href= "ตำแหน่งไฟล์.css">
```

เช่น

```
<head>
  <link rel= "stylesheet" type= "text/css" href= "myStyle.css">
</head>
```

รูปแบบนี้จะแยกคำสั่ง CSS ไปไว้อีกไฟล์หนึ่ง โดยมีนามสกุลเป็น “.css” ส่วนโครงสร้างการเขียนคำสั่งจะเหมือนกับแบบ Embedded ที่จะต้องเริ่มต้นด้วยการระบุแท็ก (Selector) แล้วตามด้วยรูปแบบ “attribute: value” ต่างๆ ภายในเครื่องหมายปีกกา และสังเกตว่าในไฟล์ CSS ไม่ต้องมีแท็ก <style> มีแต่คำสั่ง CSS เท่านั้น จากนั้นในไฟล์ HTML ให้ลิงค์มายังไฟล์ CSS ด้วยแท็ก <link> ตัวอย่างการสร้างไฟล์เอกสารไว้สองไฟล์ไว้ในโฟลเดอร์เดียวกัน ได้แก่ไฟล์ cssTest.html และไฟล์ myStyle.css ดังภาพที่ 9.3



ภาพที่ แสดงไฟล์เอกสาร 9.3HTML และไฟล์ CSS

ภายในไฟล์ cssTest.html เป็นเอกสาร HTML ที่ไม่มีคำสั่ง <style>...</style> แต่จะใช้แท็กคำสั่ง <link> แทน

cssTest.html

```
<!DOCTYPE html>
<html>
  <head>
    <title> การเขียน CSS แบบ Embedded </title>
    <meta charset= "UTF-8">
```

```
<link rel= "stylesheet" type= "text/css" href= "myStyle.css">
</head>
<body>
  <p>Cascading Style Sheets (CSS)</p>
  <p>หัวข้อที่ 3<span>การเขียน CSS แบบ External </span> มีรายละเอียดดังนี้ การเขียน CSS แบบ External เป็นการเขียน CSS ไว้ภายนอกเอกสาร HTML เป็นลักษณะ External Style Sheet ซึ่งวิธีการนี้ทำได้ด้วยการสร้างไฟล์เอกสาร CSS ที่มีนามสกุล .css ที่ภายในเขียนคำสั่ง CSS ไว้ทั้งหมดไว้อีกหนึ่งไฟล์ เมื่อต้องการให้รูปแบบ CSS นี้มีผลกับไฟล์เอกสาร HTML ใดก็สามารถเชื่อมโยงมายังไฟล์นั้นๆ ได้ทันที </p>
</body>
</html>
```

จากนั้นสร้างไฟล์จัดรูปแบบไว้ใน myStyle.css ในขั้นตอนนี้ถ้าทดสอบไฟล์ cssTest.html จะปรากฏผลลัพธ์ปกติโดยไม่มีการจัดรูปแบบใดๆ

myStyle.css

```
p { color: blue;
    font-size: 16px;
    font-family: sans-serif;
}
span {color: #fd0000;
      font-style: italic;
      text-decoration: underline;
}
```

เมื่อแสดงผลของไฟล์ cssTest.html จะปรากฏข้อความที่มีการจัดรูปแบบใหม่ตามคำสั่งที่เขียนไว้ในไฟล์ myStyle.css

ในตัวอย่างข้างต้นไฟล์ HTML และ CSS จะต้องอยู่ในไดเรกทอรีเดียวกัน ซึ่งไฟล์ CSS จะใช้ชื่อใดก็ได้ และถ้าไฟล์ HTML ใดต้องการใช้รูปแบบที่กำหนดในไฟล์ CSS ก็สามารถใช้แท็ก <link> เพื่อเชื่อมโยงเข้ามาได้ นอกจากนี้เรายังสามารถเชื่อมโยงไฟล์ CSS ได้มากกว่าหนึ่งไฟล์ ทำให้เราสามารถแยกไฟล์ CSS ตามจุดประสงค์การใช้งานได้ เช่นไฟล์ Menu.css สำหรับจัดการกับเมนู ไฟล์ Layout.css จัดการกับเลย์เอาต์ เป็นต้น เว็บไซต์ส่วนมากจะใช้รูปแบบนี้ในการสร้างธีม (Theme) กำหนดรูปแบบและประยุกต์ใช้กับทุกหน้าในเว็บไซต์

การเขียน CSS แบบ External เป็นรูปแบบที่นิยมที่สุดเพราะสามารถนำกลับมาใช้ใหม่ได้ สามารถใช้ได้กับหน้าเอกสาร HTML หลายๆ หน้าที่ต้องการจัดรูปแบบที่เหมือนกันได้ และหากต้องการเปลี่ยนแปลงรูปแบบใหม่ก็สามารถเปลี่ยนแปลงได้ที่ไฟล์ CSS ที่เดียว

.5Selectors

Selectors หมายถึงสิ่งที่ใช้ระบุแท็กคำสั่ง HTML ที่เราต้องการกำหนดรูปแบบ เช่นแท็ก <p>, <i>, <u> และอื่นๆ หรืออาจเป็น id หรือ class ที่อยู่ใน HTML ได้ โดยมีรูปแบบดังนี้

```
selector {  
    attribute: value;  
}
```

ตัวอย่างเช่น

```
p {  
    color: #fd0000;  
}
```

จากตัวอย่าง p หมายถึงส่วนของ Selector นั้นเอง การระบุ Selector มีความสำคัญในการนำรูปแบบที่กำหนดไว้ไปใช้งานที่ไม่เพียงระบุด้วยชื่อแท็กเท่านั้น แต่ยังสามารถระบุด้วยวิธีการอื่นได้ด้วย การใช้ Selector มีอยู่ 3 รูปแบบด้วยกัน คือ

- (1การระบุ Selector ด้วยชื่อแท็ก
- (2การระบุ Selector ด้วย id Attribute
- (3การระบุ Selector ด้วย class Attribute

ซึ่งแต่ละรูปแบบจะมีวิธีการใช้และความเหมาะสมที่แตกต่างกันไป ดังมีรายละเอียดจะกล่าวต่อไป

(1การระบุ Selector ด้วยชื่อแท็ก (จุดประสงค์เชิงพฤติกรรมข้อที่ 5)

เป็นการระบุ Selector ด้วยชื่อแท็กคำสั่งซึ่งจะส่งผลถึงทุกแท็กภายในเว็บเพจ รูปแบบนี้เหมาะสำหรับการใช้แท็กคำสั่งในรูปแบบเดียวกันทั้งเว็บเพจ ตัวอย่างเช่น ถ้าต้องการเปลี่ยนสีลิงค์ทั้งหมดให้เป็นสีเทาเข้ม มีรูปแบบการใช้งานดังนี้

```
tagName {  
    attribute: value;  
}
```

เช่น

```
a {  
    color: #504f50;  
}
```

จากผลของคำสั่ง จะทำให้ทุกส่วนในเว็บเพจที่ใช้แท็กคำสั่งกำหนดตำแหน่งการเชื่อมโยง <a> ในเว็บเพจนั้นจะเป็นสีเทาเข้มหมด

(2การระบุ Selector ด้วย id Attribute (จุดประสงค์เชิงพฤติกรรมข้อที่ 6)

การระบุด้วย id Attribute ใช้ในกรณีที่ต้องการระบุถึงแท็กเดียวภายในเว็บเพจเท่านั้น เนื่องด้วยเราสามารถตั้ง id ให้กับแท็ก HTML ใดๆ ก็ได้ และ id จะต้องไม่ซ้ำกันด้วย ดังนั้นถ้าต้องการระบุถึงแท็กใดเป็นการเฉพาะเจาะจง ให้ตั้ง id ให้กับแท็กนั้นๆ เสียก่อน ตัวอย่างเช่น

```
<p id= "topic"> Cascading Style Sheets (CSS)</p>
```

ถึงแม้ว่าจะมีแท็ก <p> มากกว่าหนึ่งแท็กในเว็บเพจเดียวกัน เราก็สามารถอ้างถึงแท็กดังกล่าวข้างต้นได้ด้วย id “topic” อย่างไรก็ตามในการระบุ Selector ด้วย id มีรูปแบบการเขียนที่แตกต่างจากการระบุด้วยชื่อแท็ก โดยจะต้องใช้เครื่องหมาย “#” นำหน้า Selector เสมอ ทั้งนี้เพื่อบอกให้เว็บเบราว์เซอร์รู้ว่ากำลังอ้างถึงด้วย id โดยมีรูปแบบคำสั่ง ดังนี้

```
#id {  
    attribute: value;  
}
```

ตัวอย่างการใช้งาน

```
<!DOCTYPE html>  
<html>  
  <head>  
    <title> การระบุ Selector ด้วย id Attribute </title>  
    <meta charset= "UTF-8">  
    <style type= "text/css">  
      #topic {  
        font-weight: bold;  
        text-align: center;  
      }  
    </style>  
  </head>  
  <body>  
    <p id= "topic"> Cascading Style Sheets (CSS)</p>  
    <p> Cascading Style Sheets (CSS)</p>  
  </body>  
</html>
```

จากตัวอย่าง Selector คือ #topic หมายถึงการกำหนดให้แท็กใดที่มีแอททริบิวต์ id= “topic” เท่านั้นที่มีผลกับคำสั่ง CSS นี้ ซึ่งจะเห็นได้ว่าแท็ก <p> อีกแท็กไม่มีผลต่อคำสั่ง CSS นี้เลย เนื่องจากไม่ได้ถูกกำหนดให้มี id= “topic” นั้นเอง

รูปแบบการระบุ Selector ด้วย id Attribute นี้ เหมาะกับการปรับแต่งส่วนใดส่วนหนึ่ง เฉพาะภายในเว็บเพจซึ่งจะไม่ซ้ำกันโดยเด็ดขาด เช่นการจัดองค์ประกอบเว็บเพจที่อาจแบ่งเว็บเพจ ออกเป็นส่วนต่างๆ เช่น Header, Content, Menu, Footer เป็นต้น ซึ่งในแต่ละส่วนจะมีรูปแบบ การแสดงผลที่ไม่เหมือนกัน ในกรณีนี้เราควรใช้การระบุ Selector ด้วย id Attribute และใช้แท็ก <div> แทนแท็ก <p>

(3การระบุ Selector ด้วย class Attribute (จุดประสงค์เชิงพฤติกรรมข้อที่ 7)

การระบุ Selector ด้วย class Attribute เป็นการระบุเพื่อเลือก Selector ให้มีการ เปลี่ยนแปลงรูปแบบให้กับแท็กคำสั่งได้มากกว่าหนึ่งแท็ก เช่นถ้าต้องการให้ลิงค์ภายในเว็บไซต์มีสีเทา แต่ลิงค์ไปยังเว็บไซต์อื่นเป็นสีฟ้า ซึ่งจะต้องกำหนดให้แท็ก <a> มีรูปแบบการแสดงผลที่แตกต่างกัน แม้จะเป็นแท็ก <a> เหมือนกันก็ตาม

การระบุ Selector ด้วย class Attribute จะต้องเริ่มต้นด้วยการกำหนด class Attribute ภายในแท็กนั้นๆ ก่อน เช่น

```
<a class= "dept" href= "/teacher">teacher</a>
```

```
<a class= "dept" href= "/activities">activities</a>
```

ภาษา HTML อนุญาตให้ใส่แอททริบิวต์ class ลงไปในแท็กใดๆ ได้ เพื่อใช้ในการจัดกลุ่มหรือ ประเภทเพื่อกำหนดรูปแบบ สามารถทำได้โดยการใช้เครื่องหมายจุด (.) แล้วตามด้วยชื่อ class โดยมีรูปแบบคำสั่ง ดังนี้

```
.class {  
    attribute: value;  
}
```

จากตัวอย่างลิงค์ก่อนหน้านี เราสามารถระบุ Selector ได้ดังนี้

```
.dept {  
    color: #504f50;  
}
```

จากตัวอย่างข้างต้นเราสามารถกำหนดรูปแบบให้กับแท็ก <a> ที่มี class เป็น “dept” เท่านั้น ส่วนแท็ก <a> อื่นๆ ยังคงมีรูปแบบเป็นค่าตั้งต้นเช่นเดิม

(4การระบุ Selector แบบอื่นๆ

นอกจากการระบุ Selector ใน 3 รูปแบบหลักที่ได้กล่าวไปแล้วนั้น ยังมีการระบุ Selector ในรูปแบบอื่นๆ ซึ่งเป็นในลักษณะของลูกเล่นพิเศษเพื่อเพิ่มความยืดหยุ่นในการเขียนคำสั่ง CSS ดังนี้ การระบุ-Selector แบบผสมชื่อแท็กและ class (จุดประสงค์เชิงพฤติกรรมข้อที่ 8)

ตัวอย่างเช่น

ไฟล์เอกสาร HTML

```
<!DOCTYPE html>
```

```

<html>
  <head>
    <title> การระบุ Selector แบบ Contextual </title>
    <meta charset= "UTF-8">
    <link rel= "stylesheet" type= "text/css" href= "myStyle.css">
  </head>
  <body>
    <h1 class= "itdept">Information Technology</h1>
    <p class= "itdept"> Surattthani Technical College</p>
  </body>
</html>

```

ไฟล์ CSS (myStyle.css)

```

.itdept {
  color: #fd0000;
}
p.itdept {
  font-weight: bold;
  font-style: italic;
}

```

จากตัวอย่างข้างต้นนี้แท็กใดก็ตามที่มีแอททริบิวต์ class เป็น itdept จะมีตัวอักษรเป็นสีแดง จะมีเฉพาะแท็ก <p> ที่มีแอททริบิวต์ class เป็น itdept ด้วยเท่านั้น ที่จะมีตัวอักษรเป็นตัวหนาและตัวเอียง สังเกตว่าชื่อแท็ก เครื่องหมายจุดและชื่อ class ต้องเขียนติดกัน

การระบุ-Selector แบบ Contextual (จุดประสงค์เชิงพฤติกรรมข้อที่ 9) เป็นการระบุถึงแท็กที่อยู่ภายในแท็กเฉพาะอีกที ตัวอย่างเช่น

ไฟล์เอกสาร HTML

```

<!DOCTYPE html>
<html>
  <head>
    <title> การระบุ Selector แบบ Contextual </title>
    <meta charset= "UTF-8">
    <link rel= "stylesheet" type= "text/css" href= "myStyle.css">
  </head>
  <body>
    <div id= "footer">

```

```
<em>แผนกเทคโนโลยีสารสนเทศ</em> วิทยาลัยเทคนิคสุราษฎร์ธานี สำนักงาน
คณะกรรมการการอาชีวศึกษา

</div>
</body>
</html>
```

ไฟล์ CSS (myStyle.css)

```
#footer em {
    color: #fd0000;
}
```

จากตัวอย่าง เป็นการระบุถึงแท็ก ที่อยู่ภายในแท็ก <div> ให้มีผลตามรูปแบบที่กำหนดไว้ในไฟล์ CSS ซึ่งเป็นการระบุแท็กที่อยู่ใน Context ของแท็กอื่นอีกที

การระบุ-Selector แบบรวมกลุ่ม (Grouping) (จุดประสงค์เชิงพฤติกรรมข้อที่ 10) ซึ่งสามารถกำหนด Selector ได้มากกว่าหนึ่งในคราวเดียวกัน ตัวอย่างเช่น

```
h1, h2, h3 {
    color: #fd0000;
}
h3 {
    font-style: italic;
}
```

จากตัวอย่างแท็ก <h1>, <h2>, และ <h3> เป็นแท็กที่กำหนดให้มีรูปแบบเดียวกัน ส่วนแท็ก <h3> เท่านั้นที่กำหนดให้เป็นตัวเอียงด้วย

6. รูปแบบการกำหนดคุณสมบัติใน CSS

การนำ CSS ไปใช้งานจะต้องทราบถึงการกำหนดคุณสมบัติในรูปแบบต่างๆ เพื่อการใช้งานเบื้องต้น โดยมีรูปแบบการกำหนดคุณสมบัติใน CSS ดังนี้

-Fonts

คำสั่งที่เกี่ยวข้องกับข้อความ ได้แก่ ขนาดข้อความ(font-size) ฟอนต์ของข้อความ(font-family) ความหนาของข้อความ(font-weight) และสีของข้อความ(color) เป็นต้น ซึ่งมีคุณสมบัติและค่าที่กำหนดได้ดังนี้

font-family	serif, sans-serif, monospace, etc
font-size	16pt, larger, 120%, 1.5em
font-style	normal, italic, oblique
font-variant	normal, small-caps
font-weight	normal, bold, bolder, lighter, 100, 200, ..., 800

font 16pt sans-serif

-Colors and Background

คำสั่งที่เกี่ยวข้องกับสีและพื้นหลัง ที่สามารถกำหนดพื้นหลังเป็นสี (background-color) และพื้นหลังเป็นรูปภาพ (background-image) เช่น background-color: yellow; หรือ background-image: url('img/bg.jpg'); ซึ่งมีคุณสมบัติและค่าที่กำหนดได้ดังนี้

color	green, blue, #fd0000, rgb(255,165,99)
background-color	green, #fd0000, transparent
background-image	url(image.jpg), none
background-position	top, center, bottom, left, right, 80% 40%
background-repeat	repeat, repeat-x, repeat-y, no-repeat
background-attachment	scroll, fixed

-Text

คำสั่งที่เกี่ยวข้องกับตัวอักษรหรือข้อความ ซึ่งมีคุณสมบัติและค่าที่กำหนดได้ดังนี้

text-decoration	underline, overline, line-through, blink
text-transform	capitalize, uppercase, lowercase, none
text-align	left, right, center, justify
vertical-align	top, middle, bottom, etc

-Border

คำสั่งที่เกี่ยวข้องกับเส้นขอบและระยะห่าง เพื่อความสะดวกในการจัดการเส้นขอบ เพื่อให้ง่ายต่อการออกแบบและพัฒนาเว็บไซต์ ซึ่งมีคุณสมบัติและค่าที่กำหนดได้ดังนี้

border-style	none, dotted, dashed, solid, double, etc
border-color	blue, green, #fd0000

-Margins and Paddings

คำสั่งที่เกี่ยวข้องกับการจัดระยะห่างจากขอบ ซ้าย ขวา บน ล่าง โดย padding คือระยะห่างภายใน และ margin คือระยะห่างภายนอก ซึ่งมีคุณสมบัติและค่าที่กำหนดได้ดังนี้

margin-top	2cm, 30%, 60px, 0.5in, 5em
margin-right	2cm, 30%, 60px, 0.5in, 5em
margin-bottom	2cm, 30%, 60px, 0.5in, 5em
margin-left	2cm, 30%, 60px, 0.5in, 5em
margin	15px 0 10px 0 (top, right bottom, left)
padding-top	2cm, 30%, 60px, 0.5in, 5em
padding-right	2cm, 30%, 60px, 0.5in, 5em
padding-bottom	2cm, 30%, 60px, 0.5in, 5em
padding-left	2cm, 30%, 60px, 0.5in, 5em
padding	15px 0 15px 0 (top, right bottom, left)

-List

คำสั่งที่เกี่ยวกับการกำหนดลำดับรายการ ด้วยการกำหนดรูปแบบสัญลักษณ์ที่นำมาใช้เป็นลำดับรายการ ซึ่งมีคุณสมบัติและค่าที่กำหนดได้ดังนี้

list-style-type none, circle, square, decimal, upper-roman, roman, etc
list-style-image url(image.jpg), none

-Others

คำสั่งที่เกี่ยวกับการกำหนดการแสดงผลอื่นๆ ซึ่งมีคุณสมบัติและค่าที่กำหนดได้ดังนี้

display none, inline, block, list-item, etc
visibility visible, hidden, collapse
width 250px, 20%, auto, etc

• ด้านทักษะ (ปฏิบัติ) (จุดประสงค์เชิงพฤติกรรมข้อที่ 1-12)

- 1.แบบฝึกหัดทำยบพที่ 9
- 2.กิจกรรมการเรียนรู้

• ด้านคุณธรรม/จริยธรรม/จรรยาบรรณ/บูรณาการเศรษฐกิจพอเพียง

- 1.การเตรียมความพร้อมด้าน วัสดุ อุปกรณ์นักศึกษาจะต้องทำความสะอาดห้องเรียน จัดเตรียมอุปกรณ์ในการเรียนรู้ ให้มีความพร้อมเพียงและเหมาะสมกับเวลาที่ใช้ในการเรียน
- 2.ความมีเหตุมีผลในการปฏิบัติงาน ตามหลักปรัชญาของเศรษฐกิจพอเพียงนักศึกษาจะต้องมีการใช้เทคนิคการจดบันทึกงาน การสืบค้นข้อมูล ก่อนการเรียนรู้และหลังเรียนรู้เพื่อให้การเรียนรู้เหมาะสมกับเวลา คุ่มค่าและประหยัด

กิจกรรมการเรียนการสอนหรือการเรียนรู้	
ขั้นตอนการสอนหรือกิจกรรมของครู	ขั้นตอนการเรียนรู้หรือกิจกรรมของนักเรียน
<p>1. ชี้นำเข้าสู่บทเรียน (20 นาที)</p> <ol style="list-style-type: none">1. ผู้สอนจัดเตรียมเอกสาร2. ผู้สอนแจ้งจุดประสงค์การเรียนของหน่วยเรียนที่ 9 และขอให้ผู้เรียนร่วมกันทำกิจกรรมการเรียนการสอน3. ผู้สอนให้ผู้เรียนแสดงความรู้ โดยตั้งคำถามว่า Cascading Style Sheets (CSS) มีหลักการอย่างไรพร้อมให้เหตุผลประกอบ	<p>1. ชี้นำเข้าสู่บทเรียน (20 นาที)</p> <ol style="list-style-type: none">1. ผู้เรียนเตรียมอุปกรณ์และ ฟังครูผู้สอนแนะนำรายวิชา วิธีการให้คะแนนและวิธีการเรียนเรื่อง Cascading Style Sheets (CSS)2. ผู้เรียนทำความเข้าใจเกี่ยวกับจุดประสงค์การเรียนของหน่วยเรียนที่ 9 และการให้ความร่วมมือในการทำกิจกรรม3. ผู้เรียนแสดงความรู้ว่า Cascading Style Sheets (CSS) มีหลักการอย่างไร พร้อมให้เหตุผลประกอบ
<p>2. ชี้นำให้ความรู้ (100 นาที)</p> <ol style="list-style-type: none">1. ผู้สอนทดสอบความรู้ความเข้าใจเกี่ยวกับ	<p>2. ชี้นำให้ความรู้ (100 นาที)</p> <ol style="list-style-type: none">1. ผู้เรียนทำแบบทดสอบก่อนเรียนเพื่อแสดง

<p>Cascading Style Sheets (CSS) โดยให้ผู้เรียนทำแบบทดสอบก่อนเรียน</p> <ol style="list-style-type: none"> 2. ผู้สอนอธิบายเนื้อหาเกี่ยวกับบทเรียนวิชาการสร้างเว็บไซต์ หน่วยที่ 9 เรื่อง Cascading Style Sheets (CSS) และให้ผู้เรียนศึกษาเอกสารประกอบการสอน หน่วยที่ 9 3. ผู้สอนและผู้เรียนร่วมกันอธิบายพร้อมยกตัวอย่างถึง Cascading Style Sheets (CSS) ตามที่ได้ศึกษาจากบทเรียนคอมพิวเตอร์ช่วยสอน <p>3. ชั้นประยุกต์ใช้ (90 นาที)</p> <ol style="list-style-type: none"> 1. ผู้สอนให้ผู้เรียนทำแบบฝึกหัดท้ายบทที่ 9 2. ให้ผู้เรียนเขียนเว็บเพจด้วยภาษา HTML ตามตัวอย่างที่กำหนด แล้วแสดงผลจากคำสั่ง <p>4. ชั้นสรุปและประเมินผล (30 นาที)</p> <ol style="list-style-type: none"> 1. ให้ผู้เรียนทำแบบทดสอบหลังเรียน 2. ผู้สอนและผู้เรียนร่วมกันสรุปเนื้อหาที่ได้เรียนให้มีความเข้าใจในทิศทางเดียวกัน 3. ผู้สอนให้ผู้เรียนศึกษาเพิ่มเติมนอกห้องเรียนด้วยบทเรียนคอมพิวเตอร์ช่วยสอนที่จัดทำขึ้น <p>(บรรจุจุดประสงค์เชิงพฤติกรรมข้อที่ 1-12) (รวม 240 นาที หรือ 4 ชั่วโมงเรียน)</p>	<p>ความรู้และความเข้าใจก่อนการเรียน เรื่อง Cascading Style Sheets (CSS) ผู้เรียนศึกษาบทเรียนวิชาการสร้างเว็บไซต์ หน่วยที่ 9 เรื่อง Cascading Style Sheets (CSS) และศึกษาเอกสารประกอบการสอน หน่วยที่ 9</p> <ol style="list-style-type: none"> 2. ผู้เรียนอธิบายและยกตัวอย่างถึง Cascading Style Sheets (CSS) ได้ศึกษาจากบทเรียนคอมพิวเตอร์ช่วยสอน <p>3. ชั้นประยุกต์ใช้ (90 นาที)</p> <ol style="list-style-type: none"> 1. ผู้เรียนทำแบบฝึกหัดท้ายบทที่ 9 2. ผู้เรียนเขียนเว็บเพจด้วยภาษา HTML ตามตัวอย่างที่กำหนด แล้วแสดงผลจากคำสั่ง <p>4. ชั้นสรุปและประเมินผล (30 นาที)</p> <ol style="list-style-type: none"> 1. ผู้เรียนทำแบบทดสอบหลังเรียน 2. ผู้สอนและผู้เรียนร่วมกันสรุปเนื้อหาที่ได้เรียนเพื่อให้ความเข้าใจในทิศทางเดียวกัน 3. ผู้เรียนศึกษาเพิ่มเติมนอกห้องเรียน ด้วยบทเรียนคอมพิวเตอร์ช่วยสอนที่จัดทำขึ้น <p>(บรรจุจุดประสงค์เชิงพฤติกรรมข้อที่ 1-12)</p>
--	---

งานที่มอบหมายหรือกิจกรรมการวัดผลและประเมินผล

ก่อนเรียน

1. จัดเตรียมเอกสาร สื่อการเรียนการสอนหน่วยที่ 9
2. ทำความเข้าใจเกี่ยวกับจุดประสงค์การเรียนรู้ของหน่วยที่ 9 และให้ความร่วมมือในการทำกิจกรรมในหน่วยที่ 9
3. ตอบคำถามก่อนเรียน โดยการถาม – ตอบ หน่วยที่ 9

ขณะเรียน

1. ปฏิบัติตามใบความรู้ที่ 9 เรื่อง Cascading Style Sheets (CSS)
2. ร่วมกันสรุป “Cascading Style Sheets (CSS)”
3. ร่วมกันทำกิจกรรมการเรียนรู้

หลังเรียน

1. ทำแบบฝึกหัดท้ายบทที่ 9
2. ทำแบบประเมินการเรียนรู้

คำถาม

1. จงบอกความหมายและรูปแบบคำสั่งของ CSS มาพอเข้าใจ
2. จงบอกวิธีการเขียน CSS แบบ Inline มาพอเข้าใจ
3. จงบอกวิธีการเขียน CSS แบบ Embedded มาพอเข้าใจ
4. จงบอกวิธีการเขียน CSS แบบ External มาพอเข้าใจ
5. จงบอกวิธีการระบุ Selector ด้วยชื่อแท็ก มาพอเข้าใจ
6. จงบอกวิธีการระบุ Selector ด้วย id Attribute มาพอเข้าใจ
7. จงบอกวิธีการระบุ Selector ด้วย class Attribute มาพอเข้าใจ
8. จงบอกวิธีการระบุ Selector แบบผสมชื่อแท็กและ class มาพอเข้าใจ
9. จงบอกวิธีการระบุ Selector แบบ Contextual มาพอเข้าใจ
10. จงบอกวิธีการระบุ Selector แบบรวมกลุ่ม (Grouping) มาพอเข้าใจ

ผลงาน/ชิ้นงาน/ความสำเร็จของผู้เรียน
กิจกรรม ให้ผู้เรียนทำแบบฝึกหัด

สมรรถนะที่พึงประสงค์

ผู้เรียนสร้างความเข้าใจเกี่ยวกับ Cascading Style Sheets (CSS)

1. วิเคราะห์และตีความหมาย
2. ตั้งคำถาม
3. อภิปรายแสดงความคิดเห็นระดมสมอง
4. การประยุกต์ความรู้สู่งานอาชีพ

สมรรถนะการสร้างค่านิยม

การปลูกฝังให้มีความรับผิดชอบ มีความคิดริเริ่มสร้างสรรค์ และมีความซื่อสัตย์ในอาชีพของตนเอง

สมรรถนะการปฏิบัติงานอาชีพ

1. นำความรู้เกี่ยวกับแนะนำความรู้เบื้องต้นเกี่ยวกับ Cascading Style Sheets (CSS) ไปประยุกต์ใช้ในการดำเนินชีวิตประจำวัน

สมรรถนะการขยายผล

ความสอดคล้อง

แนะนำความรู้เบื้องต้นเกี่ยวกับ Cascading Style Sheets (CSS) ทำให้ผู้เรียนมีความรู้เพิ่มขึ้น เมื่อผู้เรียนได้เรียนรู้ทำให้มีความรู้ในเรื่อง Cascading Style Sheets (CSS) มากขึ้น

สื่อการเรียนการสอน/การเรียนรู้

สื่อสิ่งพิมพ์

1. เอกสารประกอบการสอนวิชาการสร้างเว็บไซต์ (ใช้ประกอบการเรียนการสอนจุดประสงค์เชิงพฤติกรรมข้อที่ 1-12)
2. ใบความรู้และใบงาน

สื่อโสตทัศน (ถ้ามี)

1. บทเรียนคอมพิวเตอร์ช่วยสอน เรื่อง Cascading Style Sheets (CSS)
2. สื่อของจริง

แหล่งการเรียนรู้

ในสถานศึกษา

1. ห้องสมุดวิทยาลัย
2. ห้องปฏิบัติการคอมพิวเตอร์ ศึกษาหาข้อมูลทางอินเทอร์เน็ต

นอกสถานศึกษา

ผู้ประกอบการ สถานประกอบการ ในท้องถิ่น

การบูรณาการ/ความสัมพันธ์กับวิชาอื่น

1. บูรณาการกับวิชาชีวิตและวัฒนธรรมไทย ด้านการพูด การอ่าน การเขียนและการฝึกปฏิบัติตนทางสังคมด้านการเตรียมความพร้อม ความรับผิดชอบ และความสนใจใฝ่รู้
2. บูรณาการกับวิชาภาษาอังกฤษ การใช้คำศัพท์ ความหมาย การสืบค้นข้อมูล
3. บูรณาการกับวิชากีฬาเพื่อพัฒนาสุขภาพและบุคลิกภาพ ด้านบุคลิกภาพในการนำเสนอหน้าชั้นเรียน

การประเมินผลการเรียนรู้

• หลักการประเมินผลการเรียนรู้

ก่อนเรียน

1. ตรวจสอบทดสอบก่อนเรียน

ขณะเรียน

1. สังเกตการทำงานกลุ่ม

หลังเรียน

1. ตรวจสอบฝึกหัดท้ายบท
2. ตรวจสอบทดสอบหลังเรียน
3. สรุปผลการรายงานหน้าชั้นเรียน

ผลงาน/ชิ้นงาน/ผลสำเร็จของผู้เรียน

ตรวจผลงาน กิจกรรม

- กิจกรรม แบบฝึกหัด

รายละเอียดการประเมินผลการเรียนรู้

- จุดประสงค์เชิงพฤติกรรม ข้อที่ 1 บอกความหมายและรูปแบบคำสั่งของ CSS ได้
 - 1.วิธีการประเมิน : ทดสอบ
 - 2.เครื่องมือ : แบบทดสอบ
 - 3.เกณฑ์การให้คะแนน : บอกความหมายและรูปแบบคำสั่งของ CSS ได้ จะได้ 1

คะแนน

- จุดประสงค์เชิงพฤติกรรม ข้อที่ 2 บอกวิธีการเขียน CSS แบบ Inline ได้
 - 1.วิธีการประเมิน : ทดสอบ
 - 2.เครื่องมือ : แบบทดสอบ
 - 3.เกณฑ์การให้คะแนน : บอกวิธีการเขียน CSS แบบ Inline ได้ จะได้ 1 คะแนน

- จุดประสงค์เชิงพฤติกรรม ข้อที่ 3 บอกวิธีการเขียน CSS แบบ Embedded ได้
 - 1.วิธีการประเมิน : ทดสอบ
 - 2.เครื่องมือ: แบบทดสอบ
 - 3.เกณฑ์การให้คะแนน : บอกวิธีการเขียน CSS แบบ Embedded ได้ จะได้ 1

คะแนน

- จุดประสงค์เชิงพฤติกรรม ข้อที่ 4 บอกวิธีการเขียน CSS แบบ External ได้
 - 1.วิธีการประเมิน : ทดสอบ
 - 2.เครื่องมือ: แบบทดสอบ
 - 3.เกณฑ์การให้คะแนน : บอกวิธีการเขียน CSS แบบ External ได้ จะได้ 1 คะแนน

- จุดประสงค์เชิงพฤติกรรม ข้อที่ 5 บอกวิธีการระบุ Selector ด้วยชื่อแท็ก ได้
 - 1.วิธีการประเมิน: ทดสอบ
 - 2.เครื่องมือ: แบบทดสอบ
 - 3.เกณฑ์การให้คะแนน : บอกวิธีการระบุ Selector ด้วยชื่อแท็ก ได้ จะได้ 1

คะแนน

- จุดประสงค์เชิงพฤติกรรม ข้อที่ 6 บอกวิธีการระบุ Selector ด้วย id Attribute ได้
 - 1.วิธีการประเมิน : ทดสอบ
 - 2.เครื่องมือ : แบบทดสอบ
 - 3.เกณฑ์การให้คะแนน : บอกวิธีการระบุ Selector ด้วย id Attribute ได้ จะได้ 1

คะแนน

- จุดประสงค์เชิงพฤติกรรม ข้อที่ 7 บอกวิธีการระบุ Selector ด้วย class Attribute ได้
 - 1.วิธีการประเมิน : ทดสอบ
 - 2.เครื่องมือ: แบบทดสอบ
 - 3.เกณฑ์การให้คะแนน : บอกวิธีการระบุ Selector ด้วย class Attribute ได้ จะได้

1 คะแนน

- จุดประสงค์เชิงพฤติกรรม ข้อที่ 8 บอกวิธีการระบุ Selector แบบผสมชื่อแท็กและ class

ได้

- 1.วิธีการประเมิน : ทดสอบ
- 2.เครื่องมือ : แบบทดสอบ

3.เกณฑ์การให้คะแนน : บอกวิธีการระบุ Selector แบบผสมชื่อแท็กและ class ได้
จะได้ 1 คะแนน

•จุดประสงค์เชิงพฤติกรรม ข้อที่ 9 บอกวิธีการระบุ Selector แบบ Contextual ได้

1.วิธีการประเมิน : ทดสอบ

2.เครื่องมือ: แบบทดสอบ

3.เกณฑ์การให้คะแนน : บอกวิธีการระบุ Selector แบบ Contextual ได้ จะได้ 1

คะแนน

•จุดประสงค์เชิงพฤติกรรม ข้อที่ 10 บอกวิธีการระบุ Selector แบบรวมกลุ่ม (Grouping)
ได้

1.วิธีการประเมิน : ทดสอบ

2.เครื่องมือ: แบบทดสอบ

3.เกณฑ์การให้คะแนน : บอกวิธีการระบุ Selector แบบรวมกลุ่ม (Grouping) ได้

จะได้ 1 คะแนน

แบบฝึกหัดหน่วยที่ 9

เรื่อง Cascading Style Sheets (CSS)

จงตอบคำถามต่อไปนี้ให้มีความสมบูรณ์

1. จงบอกความหมายและรูปแบบคำสั่งของ CSS มาพอเข้าใจ

.....
.....
.....

2. จงบอกวิธีการเขียน CSS แบบ Inline มาพอเข้าใจ

.....
.....
.....

3. จงบอกวิธีการเขียน CSS แบบ Embedded มาพอเข้าใจ

.....
.....
.....

4. จงบอกวิธีการเขียน CSS แบบ External มาพอเข้าใจ

.....
.....
.....

5. จงบอกวิธีการระบุ Selector ด้วยชื่อแท็ก มาพอเข้าใจ

.....
.....
.....

6. จงบอกวิธีการระบุ Selector ด้วย id Attribute มาพอเข้าใจ

.....
.....
.....

7. จงบอกวิธีการระบุ Selector ด้วย class Attribute มาพอเข้าใจ

.....
.....
.....

8. จงบอกวิธีการระบุ Selector แบบผสมชื่อแท็กและ class มาพอเข้าใจ

.....
.....
.....

9. จงบอกวิธีการระบุ Selector แบบ Contextual มาพอเข้าใจ

.....
.....
.....

10. จงบอกวิธีการระบุ Selector แบบรวมกลุ่ม (Grouping) มาพอเข้าใจ

.....
.....
.....

แบบประเมินผลการนำเสนอผลงาน

ชื่อกลุ่ม.....ชั้น.....ห้อง.....

รายชื่อสมาชิก

- 1.....เลขที่.....
 2.....เลขที่.....
 3.....เลขที่.....
 4.....เลขที่.....

ที่	รายการประเมิน	คะแนน			ข้อคิดเห็น
		3	2	1	
1	ความรู้เกี่ยวกับเนื้อหา ความถูกต้อง ปฏิภาณในการตอบ และการแก้ไข ปัญหาเฉพาะหน้า				
2	รูปแบบการนำเสนอ				
3	การมีส่วนร่วมของสมาชิกในกลุ่ม				
4	บุคลิกลักษณะ กิริยา ท่าทางในการพูด น้ำเสียง ซึ่งทำให้ผู้ฟังมีความ สนใจ				
	รวม				

ผู้ประเมิน.....

เกณฑ์การให้คะแนน

1. เนื้อหาสาระครอบคลุมชัดเจนถูกต้อง
 - 3 คะแนน = มีสาระสำคัญครบถ้วนถูกต้อง ตรงตามจุดประสงค์
 - 2 คะแนน = สาระสำคัญไม่ครบถ้วน แต่ตรงตามจุดประสงค์
 - 1 คะแนน = สาระสำคัญไม่ถูกต้อง ไม่ตรงตามจุดประสงค์
2. รูปแบบการนำเสนอ
 - 3 คะแนน = มีรูปแบบการนำเสนอที่เหมาะสม มีการใช้เทคนิคที่แปลกใหม่ ใช้สื่อและเทคโนโลยี ประกอบการ นำเสนอที่น่าสนใจน่าที่สุดในท้องถิ่นมาประยุกต์ใช้อย่างคุ้มค่าและประหยัด
 - 2 คะแนน = มีเทคนิคการนำเสนอที่แปลกใหม่ ใช้สื่อและเทคโนโลยีประกอบการนำเสนอที่น่าสนใจ ใจแต่ ขาดการประยุกต์ใช้ วัสดุในท้องถิ่น
 - 1 คะแนน = เทคนิคการนำเสนอไม่เหมาะสม และไม่น่าสนใจ
3. การมีส่วนร่วมของสมาชิกในกลุ่ม
 - 3 คะแนน = สมาชิกทุกคนมีบทบาทและมีส่วนร่วมกิจกรรมกลุ่ม
 - 2 คะแนน = สมาชิกส่วนใหญ่มีบทบาทและมีส่วนร่วมกิจกรรมกลุ่ม
 - 1 คะแนน = สมาชิกส่วนน้อยมีบทบาทและมีส่วนร่วมกิจกรรมกลุ่ม
4. ความสนใจของผู้ฟัง
 - 3 คะแนน = ผู้ฟังมากกว่าร้อยละ 90 สนใจ และให้ความร่วมมือ
 - 2 คะแนน = ผู้ฟังร้อยละ 70-90 สนใจ และให้ความร่วมมือ
 - 1 คะแนน = ผู้ฟังน้อยกว่าร้อยละ 70 สนใจ และให้ความร่วมมือ

บันทึกหลังการสอน
หน่วยที่ 9 Cascading Style Sheets (CSS)

ผลการใช้แผนการเรียนรู้

1. เนื้อหาสอดคล้องกับจุดประสงค์เชิงพฤติกรรม
2. สามารถนำไปใช้ปฏิบัติการสอนได้ครบตามกระบวนการเรียนการสอน
3. เวลาที่กำหนดไว้ในแผนการสอนมีความเหมาะสม

ผลการเรียนของนักเรียน

1. นักศึกษาส่วนใหญ่มีความสนใจใฝ่รู้ เข้าใจในบทเรียน อภิปรายตอบคำถามในกลุ่ม และร่วมกันปฏิบัติใบงานที่ได้รับมอบหมาย
2. นักศึกษากระตือรือร้นและรับผิดชอบในการทำงานกลุ่มเพื่อให้งานสำเร็จทันเวลาที่กำหนด
3. นักศึกษาเลือกสามารถนำความรู้ไปใช้ได้จริงในชีวิตประจำวัน

ผลการสอนของครู

1. สอนเนื้อหาได้ครบตามหลักสูตร
2. แผนการสอนและวิธีการสอนครอบคลุมเนื้อหาการสอนทำให้ผู้สอนสอนได้อย่างมั่นใจ
3. สอนได้ทันตามเวลาที่กำหนด

ลงชื่อ.....
(.....)
รองผู้อำนวยการฝ่ายวิชาการ

ลงชื่อ.....
(.....)
ครูผู้สอน