# เปรียบเทียบ Utility Classes ของ TailwindCSS กับ CSS

นาย เพชรโกเมนทร์ เมฆดำ 66160028

## 1. ส่วน Header

| **Utility Class** | **CSS ปกติ** | **คำอธิบาย** |
| --- | --- | --- |
| bg-white | background-color: #ffffff; | กำหนดพื้นหลังสีขาว |
| shadow-lg | box-shadow: 0 10px 15px -3px rgba(0, 0, 0, 0.1), 0 4px 6px -2px rgba(0, 0, 0, 0.05); | กำหนดเอฟเฟกต์เงาสำหรับองค์ประกอบ |
| fixed | position: fixed; | การกำหนด **ตำแหน่งแบบคงที่** |
| w-full | width: 100%; | กำหนดความกว้างขององค์ประกอบ |
| top-0 | top: 0; | กำหนดตำแหน่งขององค์ประกอบให้ติดอยู่ที่ด้านบนสุด |

## 2. ส่วน Navigation

| **Utility Class** | **CSS ปกติ** | **คำอธิบาย** |
| --- | --- | --- |
| container | width: 100%;  max-width: [breakpoint];  margin-left: auto;  margin-right: auto; | กำหนดความกว้างสูงสุดและจัดกึ่งกลาง |
| mx-auto | margin-left: auto; margin-right: auto; | กำหนดระยะขอบในแนวนอน |
| px-4 | padding-left: 1rem;  padding-right: 1rem; | กำหนด **padding ในแนวนอน** (ทั้ง padding-left และ padding-right) ให้มีค่า **1rem** |
| py-4 | padding-top: 1rem;  padding-bottom: 1rem; | กำหนด **padding ในแนวตั้ง** (ทั้ง padding-top และ padding-bottom) ให้มีค่า **1rem** |
| flex | display: flex; | กำหนด **การแสดงผล (display)** ขององค์ประกอบให้เป็น **Flexbox** |
| justify-between | justify-content: space-between; | กำหนดการจัดวางองค์ประกอบลูก (child elements) ในคอนเทนเนอร์ที่ใช้ **Flexbox** ให้กระจายตัวเต็มพื้นที่ |
| items-center | align-items: center; | จัดตำแหน่งองค์ประกอบลูกใน **แนวแกนรอง (cross axis)** ใน Flexbox หรือ Grid ให้มีตำแหน่งกึ่งกลาง |
| space-x-8 | > \* + \* {  margin-left: 2rem;  } | เพิ่ม **ระยะห่างในแนวนอน** (horizontal spacing) ระหว่างองค์ประกอบลูก (child elements) ภายในคอนเทนเนอร์ที่ใช้ **Flexbox** โดยจะเพิ่มช่องว่างระหว่างแต่ละองค์ประกอบลูกเท่ากัน |

## 3. ส่วน Typography

| **Utility Class** | **CSS ปกติ** | **คำอธิบาย** |
| --- | --- | --- |
| text-2xl | font-size: 1.5rem; | กำหนดขนาดตัวอักษรให้ใหญ่ขึ้นเป็นขนาด **2xl** ซึ่งมีขนาด **1.5rem** หรือ **24px** |
| text-xl | font-size: 1.25rem; | กำหนดขนาดตัวอักษรให้เป็น **xl** ซึ่งมีขนาด **1.25rem** หรือ **20px** |
| font-bold | font-weight: 700; | กำหนดให้ข้อความมีความหนา (bold) โดยการตั้งค่า **font-weight** เป็น **700** ซึ่งทำให้ตัวอักษรหนาขึ้นจากปกติ |
| text-gray-600 | color: #4b5563; | กำหนดสีของข้อความให้เป็น **สีเทา** ที่มีความเข้มระดับ 600 |
| text-blue-600 | color: #2563eb; | กำหนดสีของข้อความให้เป็น **สีน้ำเงิน** ที่มีความเข้มระดับ 600 |

## 4. ส่วน Hero Section

| **Utility Class** | **CSS ปกติ** | **คำอธิบาย** |
| --- | --- | --- |
| bg-gradient-to-r | background-image: linear-gradient(to right, color1, color2); | กำหนดพื้นหลังเป็น **gradient** (ไล่สี) จากทิศทางซ้ายไปขวา |
| from-blue-500 | background-image: linear-gradient(to right, #3b82f6, ...); | กำหนด **สีเริ่มต้น (starting color)** สำหรับ **gradient** ในพื้นหลังหรือองค์ประกอบที่ใช้ไล่สี (gradient) โดยจะกำหนดสีที่เริ่มต้นจาก **สีน้ำเงินที่ระดับ 500** |
| to-blue-700 | background-image: linear-gradient(to right, #3b82f6, #1d4ed8); | กำหนดสีสุดท้ายของ **gradient** เป็น **สีน้ำเงิน** ที่มีความเข้มระดับ 700 |
| text-white | color: white; | กำหนดสีของข้อความให้เป็น **สีขาว** (white). |
| py-20 | padding-top: 5rem;  padding-bottom: 5rem; | กำหนด **padding** (ระยะห่างภายใน) ด้านบนและด้านล่างขององค์ประกอบ โดยจะมีค่า **20 หมายถึงการใช้ค่า 5rem (80px) ตามค่าเริ่มต้นใน Tailwind CSS** |

## 5. ส่วน Features

| **Utility Class** | **CSS ปกติ** | **คำอธิบาย** |
| --- | --- | --- |
| grid | display: grid; | กำหนดให้คอนเทนเนอร์หรือองค์ประกอบเป็น **grid container** โดยการจัดเรียงองค์ประกอบลูกในรูปแบบของตาราง (grid layout) ซึ่งสามารถจัดตำแหน่งในแถวและคอลัมน์ได้ |
| grid-cols-1 | grid-template-columns: repeat(1, 1fr); | กำหนดจำนวนคอลัมน์ใน **grid layout** ให้เป็น **1 คอลัมน์** ซึ่งจะทำให้องค์ประกอบทั้งหมดในคอนเทนเนอร์อยู่ในคอลัมน์เดียว |
| md:grid-cols-3 | @media (min-width: 768px) {  grid-template-columns: repeat(3, 1fr);  } | กำหนดจำนวนคอลัมน์ใน **grid layout** ให้เป็น **3 คอลัมน์** เมื่อขนาดหน้าจอมีขนาด **medium** (ตั้งแต่ **768px** ขึ้นไป) และใหญ่กว่า ซึ่งจะทำให้คอนเทนเนอร์จัดวางองค์ประกอบเป็น 3 คอลัมน์เมื่อขนาดหน้าจอถึงขนาดที่กำหนด |
| gap-8 | gap: 2rem; | กำหนดระยะห่างระหว่างองค์ประกอบภายใน **grid layout** หรือ **flexbox layout** โดยมีค่าเป็น **2rem** (32px) ตามค่าเริ่มต้นของ Tailwind CSS |
| rounded-lg | border-radius: 0.5rem; | กำหนด **ขอบมน (border-radius)** ให้กับองค์ประกอบ โดยมีค่า **8px** (หรือ **0.5rem**) ซึ่งทำให้ขอบขององค์ประกอบโค้งมนขึ้นจากขอบที่คม |

## 6. ส่วน Responsive Design

| **Utility Class** | **CSS ปกติ** | **คำอธิบาย** |
| --- | --- | --- |
| md:flex | @media (min-width: 768px) {  display: flex;  } | กำหนดว่าให้คอนเทนเนอร์หรือองค์ประกอบเป็น **flex container** เมื่อขนาดหน้าจอมีขนาด **medium** (ตั้งแต่ **768px** ขึ้นไป) และใหญ่กว่า ซึ่งจะทำให้จัดเรียงลูกภายในคอนเทนเนอร์ในรูปแบบของ **flex layout** |
| hidden | display: none; | เพื่อซ่อนองค์ประกอบ โดยการตั้งค่า **display: none;** ให้กับองค์ประกอบนั้น ซึ่งทำให้มันไม่แสดงผลในหน้าเว็บเลย |
| md:block | @media (min-width: 768px) {  display: block;  } | กำหนดให้คอนเทนเนอร์หรือองค์ประกอบแสดงในรูปแบบ **block** เมื่อขนาดหน้าจอมีขนาด **medium** (ตั้งแต่ **768px** ขึ้นไป) และใหญ่กว่า ซึ่งจะทำให้องค์ประกอบแสดงผลแบบเต็มความกว้างและขึ้นบรรทัดใหม่ |

## 7. ส่วน Interactive Elements

| **Utility Class** | **CSS ปกติ** | **คำอธิบาย** |
| --- | --- | --- |
| hover:bg-blue-50 | .element:hover {  background-color: #eff6ff;  } | ใช้เพื่อเปลี่ยน **background color** ขององค์ประกอบเป็น **สีฟ้าอ่อน (blue-50)** เมื่อมีการนำเมาส์ไปวาง (hover) บนองค์ประกอบนั้น |
| hover:text-blue-600 | .element:hover {  color: #2563eb;  } | ใช้เพื่อเปลี่ยน **สีของข้อความ** (text color) เป็น **ฟ้าเข้ม (blue-600)** เมื่อมีการนำเมาส์ไปวาง (hover) บนองค์ประกอบนั้น |
| transition-colors | transition-property: color, background-color, border-color; | ช้เพื่อเพิ่ม **การเปลี่ยนแปลงสี** (color transition) ให้กับองค์ประกอบเมื่อมีการโต้ตอบ เช่นการ hover, focus, หรือการเปลี่ยนแปลงในสถานะต่าง ๆ |