**🔹 Tailwind CSS**

👉 **কি?**  
Tailwind CSS হলো একটি utility-first CSS framework যা দিয়ে খুব সহজে এবং দ্রুত Web Design/ কাস্টম ডিজাইন বা UI তৈরি করা যায়। যার মাধ্যমে এখানে প্রতিটি ছোট কাজের জন্য আলাদা আলাদা utility class আছে — যেমন margin, padding, color, font-size ইত্যাদি।

এতে আমরা আলাদা করে CSS না লিখে HTML ক্লাস দিয়ে স্টাইল করি। (যেমন bg-red-500, p-4, text-center)

উদাহরণ:

<button class="bg-blue-500 text-white px-4 py-2 rounded">Click Me</button>

👉 **তুমি কেন শিখবে?**

* অনেক বেশি কাস্টমাইজেশন।
* ডিজাইন নিয়ে বেশি কন্ট্রোল থাকে।
* বড় প্রজেক্টে অনেক জনপ্রিয় (Next.js, React, Vue ইত্যাদির সাথে ভালো চলে)।

🟨 নতুনদের জন্য একটু জটিল মনে হতে পারে।

নিচে **Tailwind CSS ইন্সটল করার সহজ গাইডলাইন** দিলাম — একদম নতুনদের জন্যও বোঝা যাবে।

### ১.১ Tailwind CSS ইনস্টলেশন প্রসেস

✅ ইনস্টল করার ২টা পপুলার উপায় আছে:

1. CDN (শুধু Practice বা ছোট প্রজেক্টের জন্য)
2. npm দিয়ে Proper Installation (রিয়েল প্রজেক্টে)

**☁ Option 1: CDN দিয়ে Tailwind ব্যবহার (সর্বোচ্চ সহজ)**

👉 এটা দ্রুত কিছু প্র্যাকটিস বা ছোট প্রজেক্টের জন্য।

**🧪 Step:**

HTML ফাইলে নিচের লিংকটা বসাও <head> ট্যাগে:

html

CopyEdit

<!DOCTYPE html>

<html>

<head>

<script src="https://cdn.tailwindcss.com"></script>

</head>

<body>

<h1 class="text-3xl font-bold text-blue-500">Hello, Tailwind!</h1>

</body>

</html>

✅ এখন তুমি যেকোনো Tailwind ক্লাস HTML-এ ব্যবহার করতে পারো।

**🛠️ Option 2: Proper Installation (npm + PostCSS + Tailwind CLI)**

👉 এটা পেশাদার ওয়েব ডেভেলপমেন্টের জন্য।

**🔧 প্রয়োজন:**

* Node.js & npm ইন্সটল করা থাকতে হবে

**🧪 Step by Step:**

**📁 1. প্রজেক্ট তৈরি করো**

bash

CopyEdit

mkdir my-project

cd my-project

npm init -y

**📦 2. Tailwind ইনস্টল করো**

bash

CopyEdit

npm install -D tailwindcss

npx tailwindcss init

➡️ এতে tailwind.config.js নামে একটি ফাইল তৈরি হবে।

**📄 3. নিজের CSS ফাইল তৈরি করো**

**src/input.css** নামে একটা ফাইল বানাও এবং লিখো:

css

CopyEdit

@tailwind base;

@tailwind components;

@tailwind utilities;

**🏗️ 4. Tailwind CSS Build করো**

bash

CopyEdit

npx tailwindcss -i ./src/input.css -o ./dist/output.css --watch

➡️ এটি input.css থেকে Tailwind কোড কম্পাইল করে output.css বানাবে।

**🌐 5. HTML ফাইলে CSS লিংক করো**

html

CopyEdit

<!DOCTYPE html>

<html>

<head>

<link href="./dist/output.css" rel="stylesheet">

</head>

<body>

<h1 class="text-3xl font-bold text-green-600">Hello, Tailwind!</h1>

</body>

</html>

**🤔 কোনটা ব্যবহার করবো?**

| **তুমি যদি...** | **তাহলে...** |
| --- | --- |
| শুধু শেখা বা টেস্ট করো | 👉 CDN দিয়েই শুরু করো |
| সিরিয়াস প্রজেক্ট করো | 👉 Proper (npm) ইন্সটলেশন করো |

### Tailwind CSS এর সুবিধা

* তৈরি করা খুব দ্রুত হয়
* HTML আর CSS আলাদা করে লিখতে হয় না
* সহজে কাস্টমাইজ করা যায়
* বড় প্রজেক্টেও maintain করা সহজ

**Tailwind CSS Utility Class’s**

নিচে Tailwind CSS এর ক্লাস এবং utilities গুলো সম্পর্কে সহজ ভাষায় ব্যাখ্যা দেওয়া হলো:

**🔹 ১. Spacing (Margin & Padding)**

**ব্যবহার:** element এর চারপাশে জায়গা নেওয়ার জন্য।

* m-4 → সব দিকে margin 1rem
* mt-2 → শুধু উপরে margin
* mx-3 → ডানে-বামে margin
* p-4 → সব দিকে padding
* py-2 → উপরে-নিচে padding
* px-, py-, pt-, mb- দিক নির্দেশ করার জন্য

📌 **m = margin, p = padding, t = top, b = bottom, x = left-right, y = top-bottom**

**Tailwind spacing scale (সংক্ষেপে):**

| **Class** | **Pixels** | **rem** |
| --- | --- | --- |
| mb-0 | 0px | 0rem |
| mb-1 | 4px | 0.25rem |
| mb-2 | 8px | 0.5rem |
| mb-3 | 12px | 0.75rem |
| mb-4 | 16px | 1rem |

**কবে ব্যবহার করব?**

যখন আপনি চান কোনো এলিমেন্টের **নিচে একটু ফাঁকা** বা *margin* থাকুক, তখন mb-1,

mb-2 ইত্যাদি ব্যবহার করবেন।

**🔹 ২. Text Styling**

* text-xl → বড় টেক্সট
* text-sm → ছোট টেক্সট
* font-bold → মোটা হরফ
* text-center → টেক্সট সেন্টারে
* text-gray-500 → ধুসর রঙের টেক্সট

**🔹 ৩. Background & Colors**

প্রি-ডিফাইন্ড রঙের ক্লাস ব্যবহার করি যেমন:

* bg-blue-500 → নীল ব্যাকগ্রাউন্ড
* bg-green-100 → হালকা সবুজ ব্যাকগ্রাউন্ড
* text-white → সাদা টেক্সট
* hover:bg-red-600 → মাউস নিলেই লাল হয়ে যাবে

উদাহরণ:

<p class="text-blue-600">This is blue text</p>

**🔹 ৪. Flexbox & Layout**

* flex → ফ্লেক্সবক্স শুরু
* flex-row → পাশাপাশি বসবে
* flex-col → উপর-নিচে বসবে
* items-center → ভরদুপুরে বসানো
* justify-between → দুই পাশে জায়গা দিয়ে বসানো

**🔹 ৫. Width & Height**

* w-full → ১০০% প্রস্থ
* w-1/2 → ৫০% প্রস্থ
* h-10 → উচ্চতা ২.৫ রেম
* max-w-md → মাঝারি সর্বোচ্চ প্রস্থ

**🔹 ৬. Borders & Radius**

* border → বর্ডার দিবে
* border-gray-300 → ধুসর বর্ডার
* rounded → কোণ গোল করবে
* rounded-lg → বড় গোল কোণ

**🔹 ৭. Shadow & Effects**

* shadow → হালকা ছায়া
* shadow-lg → বড় ছায়া
* hover:shadow-xl → হোভার করলে বড় ছায়া

**🔹 ৮. Responsive Breakpoints**

* sm: → ছোট স্ক্রিন (≥ 640px)
* md: → মাঝারি স্ক্রিন (≥ 768px)
* lg: → বড় স্ক্রিন (≥ 1024px)

উদাহরণ:

html

CopyEdit

<p class="text-sm md:text-lg lg:text-xl">Responsive Text</p>

**🔹 ৯. Hover, Focus এবং অন্যান্য States**

Tailwind এ state ব্যবহারের জন্য prefix ব্যবহার করা হয়:

* hover:
* focus:
* active:
* disabled:

উদাহরণ:

<button class="bg-blue-500 hover:bg-blue-700 focus:outline-none">Hover Me</button>

### ১০. পজিশনিং

relative, absolute, top-\*, left-\* ক্লাস ব্যবহার করি।

উদাহরণ:

<div class="relative">

<div class="absolute top-0 left-0">Overlay</div>

</div>

### ১২. Overflow এবং Visibility

overflow-\*, invisible, visible, hidden ক্লাস ব্যবহার করি।

<div class="overflow-scroll h-32">Scrollable content</div>

### ১৩. Typography

লেখার সাইজ, ওজন, স্পেসিং নিয়ন্ত্রণ করতে পারি।

<p class="text-lg font-bold leading-relaxed tracking-wide">Text</p>

### ১৪. Text Alignment, Decoration, Transformation

text-center, underline, uppercase ক্লাস ব্যবহার করি।

<p class="text-center underline uppercase">Stylish Text</p>

### ১৫. Background Properties

bg-\*, bg-cover, bg-center, bg-gradient-to-r ক্লাস ব্যবহার করি।

<div class="bg-gradient-to-r from-blue-400 to-green-400">Gradient</div>

### ১৬. Borders এবং Outlines

border, border-2, border-red-500, outline-none ক্লাস ব্যবহার করি।

<input class="border border-gray-300 outline-none" />

### ১৭. Shadow এবং Opacity

shadow, shadow-md, opacity-50 ক্লাস ব্যবহার করি।

<div class="shadow-lg opacity-75">Box</div>

### ১৮. Aspect Ratio, Columns, Layout Enhancements

aspect-square, columns-2, gap-x-4 ক্লাস ব্যবহার করি।

<div class="aspect-square bg-red-300"></div>

### ১৯. তালিকা এবং টেবিল স্টাইলিং

list-disc, list-inside, table-auto, border ক্লাস ব্যবহার করি।

<ul class="list-disc list-inside">

<li>Item</li>

</ul>

### ২০. ফর্ম কাস্টমাইজ

Tailwind form plugin বা নিজের ক্লাস ব্যবহার করা যায়।

<input class="border px-3 py-2 rounded w-full" />

### ২১. Cursor, Pointer Events, User Select

cursor-pointer, pointer-events-none, select-none ক্লাস ব্যবহার করি।

<div class="cursor-pointer select-none">Clickable</div>

### ২২. Transitions

transition, duration-\*, ease-\* ক্লাস ব্যবহার করি।

<button class="transition duration-300 ease-in-out hover:bg-red-500">Hover</button>

### ২৩. Animations

animate-bounce, animate-spin (plugin দরকার হতে পারে)

<div class="animate-bounce">Bouncing</div>

### ২৪. Transformations

transform, scale-\*, rotate-\*, translate-x-\* ক্লাস ব্যবহার করি।

<div class="transform scale-105 rotate-6">Transformed</div>

#### 🌀 Skew (বাঁকা করা)

skew-x-\* এবং skew-y-\* দিয়ে এলিমেন্টকে x বা y অক্ষ বরাবর বাঁকানো যায়।

<div class="transform skew-x-12 bg-pink-300 p-4">

Skewed Box

</div>

নোট: অবশ্যই transform ক্লাসটি থাকতে হবে যেন skew কাজ করে।

### ২৫. Scroll Behaviour

overflow-auto, scroll-smooth ক্লাস ব্যবহার করি।

<div class="h-32 overflow-auto scroll-smooth">Content</div>

### ২৬. Resizing

resize, resize-none, resize-x, resize-y ক্লাস ব্যবহার করি।

<textarea class="resize-y w-full h-32"></textarea>

**✅ সংক্ষেপে:**

| **Utility** | **কাজ** |
| --- | --- |
| m-4 | margin |
| p-2 | padding |
| text-lg | বড় লেখা |
| bg-red-500 | লাল ব্যাকগ্রাউন্ড |
| flex | flex layout |
| rounded | গোল কোণ |
| shadow-md | ছায়া |

Tailwind CSS শেখা অনেকটা LEGO দিয়ে ডিজাইন বানানোর মতো – ছোট ছোট অংশ (utility) দিয়ে পুরো ডিজাইন তৈরি করা যায়।

**🔍 Axis (অক্ষ) কী?**

**Axis** শব্দটি সাধারণত গণিত, জ্যামিতি, গ্রাফিক্স ডিজাইন, ও ওয়েব ডেভেলপমেন্ট (বিশেষ করে CSS বা Tailwind CSS এ) ব্যবহার হয়। সহজ ভাষায়:

👉 **Axis মানে হলো: কোনো একটি জায়গার রেফারেন্স ধরে কোনো কিছুর অবস্থান বা দিক বোঝানোর জন্য কল্পিত রেখা।**

**🎯 দুইটি প্রধান ধরনের Axis আছে:**

**১️⃣ X-Axis (এক্স অক্ষ)**

* এটি **আড়াআড়ি (horizontal)** লাইন
* **বাম → ডান** দিকে চলে
* উদাহরণ: একজন মানুষ সোজা সামনে তাকিয়ে থাকলে তার হাত ছড়ালে দুই দিকে যতদূর যায় – সেটি x-axis

**২️⃣ Y-Axis (ওয়াই অক্ষ)**

* এটি **উল্লম্ব (vertical)** লাইন
* **উপরে ↕ নিচে** দিকে চলে
* উদাহরণ: মাথা থেকে পায়ের দিকে যা যায় – সেটি y-axis

**🧭 Tailwind CSS বা CSS-এ Axis ব্যবহার**

CSS/Tailwind-এ spacing ও layout এর ক্ষেত্রে **axis** অনুযায়ী class গুলো ব্যবহার হয়।

| **ক্লাস** | **অর্থ** |
| --- | --- |
| mx-4 | x-axis (left-right) এ margin |
| px-2 | x-axis (left-right) এ padding |
| my-3 | y-axis (top-bottom) এ margin |
| py-1 | y-axis (top-bottom) এ padding |

📌 **x → left & right**, **y → top & bottom**

**📊 Graph বা Chart এ Axis**

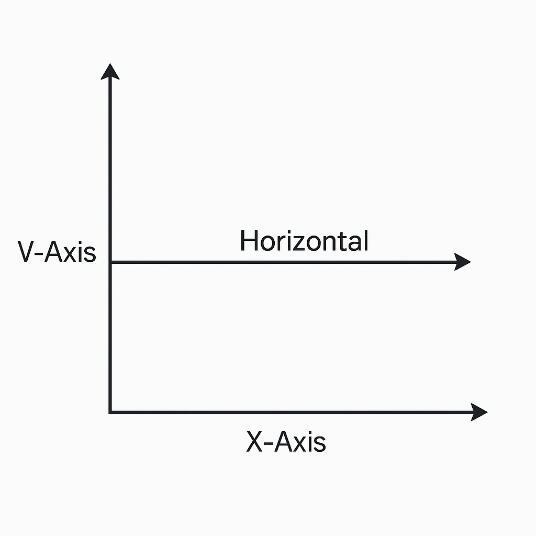
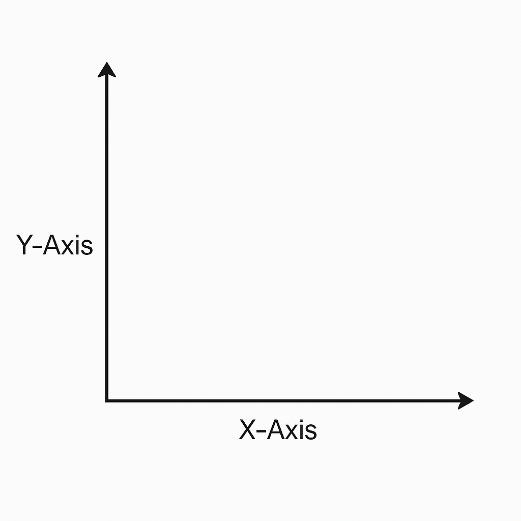
* **X-axis** → নিচে থাকে, সময় বা ক্যাটাগরি বোঝাতে ব্যবহার হয়
* **Y-axis** → পাশ দিয়ে থাকে, মান বা সংখ্যা বোঝাতে ব্যবহার হয়

🧮 উদাহরণ:  
একটা বার চার্টে,

* X-axis → বছর: 2021, 2022, 2023
* Y-axis → বিক্রির পরিমাণ: 100, 200, 300

**✅ সহজভাবে মনে রাখার টিপস:**

* **X = Across (আড়াআড়ি)**
* **Y = Yoyo (উপর ↕ নিচে)**



**ROUNDED Utility of Tailwind CSS**

Tailwind CSS-এ rounded class গুলো ব্যবহার হয় HTML element-এর কোণ (corner) গোল বা বৃত্তাকার করার জন্য। নিচে এগুলোকে **ব্রেকডাউন সহ** ব্যাখ্যা করছি:

**🌀 🔹 Base: rounded**

* rounded  
  👉 সব কোণ (top-left, top-right, bottom-left, bottom-right) একটু গোল করে দেয়।  
  **সমান** ভাবে সব পাশে গোল হয়।

**🌀 🔹 Size অনুযায়ী Rounded**

| **ক্লাস** | **কোণের পরিমাণ** | **ব্যাখ্যা** |
| --- | --- | --- |
| rounded-none | 0px | একেবারে গোল না |
| rounded-sm | 2px | হালকা গোল |
| rounded | 4px | ডিফল্ট গোল |
| rounded-md | 6px | একটু বেশি |
| rounded-lg | 8px | বড় গোল |
| rounded-xl | 12px | আরও বড় |
| rounded-2xl | 16px | অনেক বেশি |
| rounded-3xl | 24px | খুব বেশি |
| rounded-full | 9999px | একদম বৃত্তাকার (বৃত্ত বানাতে) |

**🌀 🔹 নির্দিষ্ট দিক অনুযায়ী Rounded**

Tailwind-এ তুমি চাইলে শুধু কোন এক বা দুই পাশে গোল করতে পারো।

**🔸 দিকভিত্তিক কোড:**

| **ক্লাস** | **কোণ** | **অর্থ** |
| --- | --- | --- |
| rounded-t | top-left + top-right | উপরের দুই কোণ গোল |
| rounded-b | bottom-left + bottom-right | নিচের দুই কোণ |
| rounded-l | top-left + bottom-left | বাম পাশে |
| rounded-r | top-right + bottom-right | ডান পাশে |

**🔸 নির্দিষ্ট কোণ (Corner) ক্লাস:**

| **ক্লাস** | **কোণ** | **ব্যাখ্যা** |
| --- | --- | --- |
| rounded-tl | top-left | শুধু উপরের বাঁ দিক |
| rounded-tr | top-right | শুধু উপরের ডান দিক |
| rounded-bl | bottom-left | নিচের বাঁ দিক |
| rounded-br | bottom-right | নিচের ডান দিক |

✳️ এই ক্লাসগুলোর সাথেও তুমি সাইজ দিতে পারো, যেমনঃ

* rounded-tl-lg
* rounded-br-sm
* rounded-r-xl

**✅ উদাহরণ (HTML সহ)**

html

CopyEdit

<div class="w-32 h-32 bg-blue-500 rounded-lg"></div>

<!-- সব কোণ মাঝারি গোল -->

<div class="w-32 h-32 bg-red-400 rounded-t-full"></div>

<!-- শুধু উপরের দুই কোণ সম্পূর্ণ গোল -->

<div class="w-32 h-32 bg-green-400 rounded-bl-3xl"></div>

<!-- নিচের বাঁ কোণ বড় গোল -->

**🧠 মনে রাখার টিপস:**

* rounded = সব কোণ
* t = top, b = bottom, l = left, r = right
* full = একদম বৃত্ত (যদি সমান width & height হয়)

**📱🖥️ Responsive Breakpoints কি?**

**Breakpoints** হলো কিছু নির্দিষ্ট স্ক্রিনের প্রস্থ (width), যার ভিত্তিতে Tailwind CSS আলাদা ক্লাস কাজ করায়।  
এগুলো mobile → tablet → desktop → large screen অনুযায়ী সাজানো।

**🔓 Tailwind এর Default Breakpoints তালিকা**

| **Breakpoint** | **ক্লাস Prefix** | **স্ক্রিন সাইজ (min-width)** | **ডিভাইস টাইপ উদাহরণ** |
| --- | --- | --- | --- |
| sm | sm: | **640px** | মোবাইল (landscape) |
| md | md: | **768px** | ট্যাবলেট |
| lg | lg: | **1024px** | ছোট ডেস্কটপ |
| xl | xl: | **1280px** | বড় ডেস্কটপ |
| 2xl | 2xl: | **1536px** | অতিরিক্ত বড় স্ক্রিন |

**🧠 কিভাবে কাজ করে?**

Tailwind-এর responsive ক্লাসগুলো এভাবে কাজ করে:

👉 **md:text-lg**  
মানে হলো — স্ক্রিন যদি **768px বা তার বেশি হয়**, তখন টেক্সট হবে **large (text-lg)**।

**📦 উদাহরণ:**

html

CopyEdit

<p class="text-sm sm:text-base md:text-lg lg:text-xl xl:text-2xl">

Responsive Text!

</p>

এই টেক্সটটি:

* 📱 মোবাইল এ থাকবে text-sm
* 🧾 640px+ হলে text-base
* 💻 768px+ হলে text-lg
* 🖥️ 1024px+ হলে text-xl
* 🧲 1280px+ হলে text-2xl

**💡 বাস্তব জীবনের উদাহরণ:**

html

CopyEdit

<div class="bg-red-300 md:bg-blue-300 lg:bg-green-300 p-4">

Responsive Background

</div>

* ছোট স্ক্রিনে ব্যাকগ্রাউন্ড হবে লাল
* মাঝারি স্ক্রিনে নীল
* বড় স্ক্রিনে সবুজ

**✅ সংক্ষেপে মনে রাখার টিপস:**

| **Prefix** | **মানে কী?** | **শুরু হয় কত px থেকে?** |
| --- | --- | --- |
| sm: | small | ≥ 640px |
| md: | medium | ≥ 768px |
| lg: | large | ≥ 1024px |
| xl: | extra large | ≥ 1280px |
| 2xl: | double XL | ≥ 1536px |

**Tailwind CSS-এ লিখে দেওয়া width ক্লাস (w-[value], w-full, w-64 ইত্যাদি)** সাধারণত responsive হয় **না**, যদি না তুমি **breakpoint prefixes** (sm:, md:, lg:, xl:) ব্যবহার করো।

**🔹 সোজা কথায়:**

| **তুমি কী লিখলে** | **Responsive কি?** | **ব্যাখ্যা** |
| --- | --- | --- |
| w-full | ✅ হ্যাঁ | প্যারেন্ট container এর উপর নির্ভর করে |
| w-[300px] | ❌ না | সব স্ক্রিনে একই সাইজ থাকবে |
| sm:w-full md:w-[400px] | ✅ হ্যাঁ | স্ক্রিন সাইজ ভেদে ভিন্ন width |
| lg:w-1/2 | ✅ হ্যাঁ | বড় স্ক্রিনে ৫০% width নিবে |

✅ উদাহরণ – একটি responsive ইনপুট ফিল্ড:

html

CopyEdit

<input

type="text"

placeholder="Your name"

class="border p-2 rounded-md w-full sm:w-[300px] md:w-[400px] lg:w-[500px]"

/>

⬆️ এতে ছোট স্ক্রিনে w-full, তারপর ধীরে ধীরে বড় স্ক্রিনে input ফিল্ড বড় হবে।

**✅ Button ও Text ক্ষেত্রেও একই নিয়ম:**

html

CopyEdit

<button class="bg-blue-600 text-white px-4 py-2 rounded w-full sm:w-auto">

Submit

</button>

* w-full: ছোট স্ক্রিনে পুরোটা নেবে
* sm:w-auto: বড় স্ক্রিনে content অনুযায়ী width

## 🔚 সংক্ষেপে:

Tailwind-এ যেকোনো width বা অন্য ক্লাসকে responsive করতে হলে **sm: md: lg: ইত্যাদি prefix** ব্যবহার করো।  
w-[300px] নিজে নিজে responsive হবে না।

**📚 Tailwind CSS max-w-\* ক্লাসগুলোর সম্পূর্ণ তালিকা:**

| **ক্লাস** | **প্রস্থ (px)** | **rem** | **ব্যাখ্যা** |
| --- | --- | --- | --- |
| max-w-0 | 0px | 0rem | একদম কিছুই না |
| max-w-none | none | — | কোনো সীমা নেই (unlimited width) |
| max-w-xs | **320px** | 20rem | ছোট এলিমেন্টের জন্য |
| max-w-sm | **384px** | 24rem | মিডিয়াম ছোট |
| max-w-md | **448px** | 28rem | মাঝারি |
| max-w-lg | **512px** | 32rem | একটু বড় |
| max-w-xl | **576px** | 36rem | আরো বড় |
| max-w-2xl | **672px** | 42rem | বড় টেক্সট ব্লক, পোস্ট |
| max-w-3xl | **768px** | 48rem | বড় পোস্ট বা কনটেইনার |
| max-w-4xl | **896px** | 56rem | প্রায় ফুল কনটেইনার |
| max-w-5xl | **1024px** | 64rem | ওয়াইড কনটেইনার |
| max-w-6xl | **1152px** | 72rem | ওয়াইডার |
| max-w-7xl | **1280px** | 80rem | এক্সট্রা ওয়াইড লেআউট |
| max-w-full | 100% | — | প্যারেন্টের পুরো প্রস্থ |
| max-w-min | content size | — | যতটুকু দরকার, তার চেয়ে কম না |
| max-w-max | content size | — | যতটুকু দরকার, তার চেয়ে বেশি না |
| max-w-fit | fit-content | — | CSS fit-content |
| max-w-screen-sm | 640px | — | ছোট স্ক্রিনের পুরো প্রস্থ |
| max-w-screen-md | 768px | — | মিডিয়াম স্ক্রিন |
| max-w-screen-lg | 1024px | — | লার্জ স্ক্রিন |
| max-w-screen-xl | 1280px | — | এক্স-লার্জ স্ক্রিন |
| max-w-screen-2xl | 1536px | — | এক্সট্রা এক্সট্রা লার্জ |

**🧠 কবে কোনটা ব্যবহার করব?**

| **প্রয়োজনে** | **ক্লাস** |
| --- | --- |
| ছোট কার্ড, মোবাইল এলিমেন্ট | max-w-xs, max-w-sm |
| ফর্ম, আর্টিকেল | max-w-md, max-w-lg |
| ব্লগ বা কন্টেন্ট কনটেইনার | max-w-xl - max-w-2xl |
| হোমপেজ কন্টেইনার | max-w-4xl - max-w-6xl |
| ফুল স্ক্রীন ডিভাইস ভিত্তিক | max-w-screen-\* |

**🎯 উদাহরণ:**

html

CopyEdit

<div class="max-w-md mx-auto bg-white p-4 shadow">

<p>This box is max 448px wide.</p>

</div>

**✨ টিপস:**

* max-w মানে সর্বোচ্চ প্রস্থ, তার বেশি হবে না।
* এটি width: 100% এর সাথে ব্যবহার করলে, স্ক্রীনের চেয়ে বড় হতে দেবে না।
* container ক্লাসটিও max-w-screen-\* গুলোর responsive version ব্যবহার করে।

**🔹 Flex vs Grid → মূল পার্থক্য:**

| **বিষয়** | **Flex** | **Grid** |
| --- | --- | --- |
| লে-আউট কেমন হয়? | **এক লাইনে বা এক দিকে** | **দুই দিকেই** (row + column) |
| নিয়ন্ত্রণ | "একটা দিক" (row **অথবা** column) | "দুইটা দিকেই" |
| কবে ব্যবহার করবো? | আইটেমগুলো এক লাইনে রাখতে চাইলে | পুরো টেমপ্লেট বা কমপ্লেক্স লে-আউট চাইলে |

**🧱 ১. Flexbox (এক লাইনে সাজানো)**

**📌 মনে রাখো:**

Flexbox = "এক সারি বা এক কলামে" আইটেম গুলো সাজাও

**উদাহরণ:**

html

CopyEdit

<div class="flex">

<div class="bg-red-400 p-4">Box 1</div>

<div class="bg-blue-400 p-4">Box 2</div>

<div class="bg-green-400 p-4">Box 3</div>

</div>

🔍 এটা "এক লাইন"-এ সব বক্স পাশাপাশি বসবে।

**✅ গুরুত্বপূর্ণ ক্লাস:**

| **ক্লাস** | **কাজ** |
| --- | --- |
| flex | ফ্লেক্স শুরু করে |
| flex-row | বাম → ডানে (default) |
| flex-col | উপর → নিচে |
| justify-center | হরাইজন্টালি মাঝখানে |
| items-center | ভার্টিকালি মাঝখানে |

**🧩 ২. Grid (দুই দিকেই সাজানো)**

**📌 মনে রাখো:**

Grid = "সারি + কলাম" দুইটাই — টেবিলের মতো layout

**উদাহরণ:**

html

CopyEdit

<div class="grid grid-cols-3 gap-4">

<div class="bg-red-400 p-4">1</div>

<div class="bg-blue-400 p-4">2</div>

<div class="bg-green-400 p-4">3</div>

<div class="bg-yellow-400 p-4">4</div>

<div class="bg-pink-400 p-4">5</div>

</div>

🔍 এখানে ৩টা কলাম হবে, নিচে গিয়ে নতুন রো শুরু হবে।

**✅ গুরুত্বপূর্ণ ক্লাস:**

| **ক্লাস** | **কাজ** |
| --- | --- |
| grid | গ্রিড শুরু |
| grid-cols-3 | ৩টা কলাম |
| grid-rows-2 | ২টা সারি |
| gap-4 | সারি ও কলামের মাঝে ফাঁকা |
| col-span-2 | ২ কলাম জুড়ে বসবে |

**🧠 মনে রাখার টিপস:**

**🎯 Flex:**

* তুমি জাস্ট এক লাইন/দিক-এ জিনিস সাজাতে চাইলে
* যেমনঃ বাটনগুলো পাশাপাশি রাখতে

👉 **"এক লাইনে বসাও" → Flex**

**🎯 Grid:**

* তুমি যদি layout বানাও (যেমনঃ কার্ড ৩টা করে, সারি করে বসাতে চাও)

👉 **"পুরো টেবিল/লে-আউট ভাব" → Grid**

**✅ চোখে দেখা Example Code: Flex vs Grid**

**👉 Flex:**

html

CopyEdit

<div class="flex justify-between">

<button>Home</button>

<button>About</button>

<button>Contact</button>

</div>

**👉 Grid:**

html

CopyEdit

<div class="grid grid-cols-2 gap-4">

<div class="bg-red-300">Box 1</div>

<div class="bg-blue-300">Box 2</div>

<div class="bg-green-300">Box 3</div>

<div class="bg-yellow-300">Box 4</div>

</div>

**🔚 শেষ কথা:**

| **তুমি যদি চাও...** | **ব্যবহার করো** |
| --- | --- |
| সব আইটেম এক লাইনে বসুক | flex |
| আইটেমগুলো সারি + কলামে বসুক | grid |

Tailwind CSS-এর **প্রয়োজনীয় সব ইমেজ রিলেটেড ক্লাসগুলো** সহজভাবে ব্যাখ্যা করছি — যেন তুমি সমস্যা ছাড়াই যেকোনো ইমেজ সুন্দরভাবে বসাতে পারো, crop করতে পারো, এবং responsive বানাতে পারো।

**🖼️ ১. Image Sizing (Width & Height)**

| **ক্লাস** | **কাজ** |
| --- | --- |
| w-full | প্যারেন্ট ডিভের সম্পূর্ণ প্রস্থ নেবে |
| w-[300px] | নির্দিষ্ট px দিয়ে প্রস্থ নির্ধারণ |
| h-[200px] | নির্দিষ্ট উচ্চতা |
| max-w-[500px] | সর্বোচ্চ প্রস্থ নির্ধারণ |
| object-contain | ইমেজ স্কেল হবে পুরো জায়গায় ঢুকতে, **crop হবে না** |
| object-cover | ইমেজ পুরো জায়গা ভরবে এবং **crop হবে**, সুন্দরভাবে ফিট করতে |
| object-fill | ইমেজ stretch হবে, **distort হতে পারে** |

🔸 সবচেয়ে বেশি ব্যবহার হয়: object-cover, w-full, h-full, w-[value], h-[value]

**Breakdown:**

| **ক্লাস** | **কাজ** |
| --- | --- |
| bg-[url('...')] | background image সেট করে |
| bg-cover | image পুরো div-কে কভার করে |
| bg-contain | image container-এর মধ্যে adjust করে |
| bg-center | image center-এ অবস্থান করে |
| bg-no-repeat | ইমেজ রিপিট না করে |
| h-64 / w-full | height এবং width নির্ধারণ করে |

**Background Image ব্যবহার করলে নিচের ইউটিলিটি গুলো ব্যবহার করতে হবে**

| **ক্লাস** | **কাজ** |
| --- | --- |
| w-full | container এর পুরো width নেয় |
| flex flex-col items-center justify-center | text গুলোকে vertically & horizontally center করে |
| text-white text-center | টেক্সট সাদা এবং সেন্টার করে |
| px-6 | padding দেয় যাতে text গুলো ইমেজে ডানে-বামে না লাগে |
| mb-2 | heading ও paragraph এর মাঝে গ্যাপ |

**🧪 Example:**

html

CopyEdit

<div class="bg-[url('https://example.com/bg.jpg')] bg-cover bg-center h-[300px] w-full rounded-lg">

<h1 class="text-white text-3xl p-10">Welcome</h1>

</div>

**🔐 দয়া করে খেয়াল রাখো:**

* লোকাল ইমেজ ব্যবহার করলে সঠিক path দাও: bg-[url('/images/your-image.png')]
* Tailwind CSS এর config ফাইল (tailwind.config.js) এ content paths ঠিকঠাক থাকতে হবে যাতে Tailwind এই ক্লাস detect করতে পারে।

**✅ Tips for Text on Background:**

* যদি ইমেজ dark হয় → text-white
* যদি ইমেজ light হয় → text-black বা bg-black bg-opacity-50 overlay ব্যবহার করতে পারো।

**⚠️ Image Path সমস্যা হলে:**

* Ensure image path is correct: /daily-todo/250719\_practice/product image.png
* যদি image লোড না হয়, space এর জায়গায় %20 encode করে দেখতে পারো বা নাম পরিবর্তন করে product-image.png রাখো।

**✂️ ২. Image Cropping & Fitting (Object Position & Cover)**

| **ক্লাস** | **কাজ** |
| --- | --- |
| object-cover | image জায়গা ভরবে এবং crop হবে |
| object-contain | image সম্পূর্ণ দেখাবে, crop হবে না |
| object-center | crop করার সময় image এর মাঝখান থেকে crop হবে |
| object-top, object-bottom, object-left, object-right | crop করার জায়গা কন্ট্রোল করবে |

🔸 উদাহরণ:

html

CopyEdit

<img src="..." class="w-[300px] h-[200px] object-cover object-top rounded-xl" />

**📐 ৩. Border & Shape**

| **ক্লাস** | **কাজ** |
| --- | --- |
| rounded | হালকা বর্ডার রেডিয়াস |
| rounded-lg, rounded-xl, rounded-full | ভিন্ন ভিন্ন পরিমাণে গোলাকৃতি |
| border, border-2, border-gray-200 | বর্ডার যুক্ত করবে ও রঙ দিবে |

**🔄 ৪. Responsive Utility**

| **ক্লাস** | **কাজ** |
| --- | --- |
| sm:w-[100px], md:h-[200px] | আলাদা স্ক্রিন সাইজ অনুযায়ী width/height |
| max-w-full, h-auto | responsive height/proportion রাখতে |
| aspect-[16/9] | নির্দিষ্ট aspect ratio ধরে রাখতে |

🔸 Responsive বানাতে এটা করো:

html

CopyEdit

<img src="..." class="w-full h-auto object-cover rounded-lg" />

**🧭 ৫. Alignment & Display**

| **ক্লাস** | **কাজ** |
| --- | --- |
| mx-auto | horizontally center করবে |
| block | display block বানাবে (inline না) |
| flex justify-center items-center | flex layout-এ image center-এ রাখতে সাহায্য করে |

**🌈 ৬. Shadow & Effects**

| **ক্লাস** | **কাজ** |
| --- | --- |
| shadow, shadow-lg, shadow-xl | ইমেজে ছায়া দিবে |
| hover:scale-105, transition | hover effect ও smooth animation |

**✅ চূড়ান্ত উদাহরণ (Perfect image card):**

html

CopyEdit

<div class="w-[300px] h-[250px] rounded-xl overflow-hidden shadow-lg">

<img src="/path/image.jpg"

alt="Sample Image"

class="w-full h-full object-cover object-center transition hover:scale-105 duration-300" />

</div>

* w-full h-full: container ফিল করে
* object-cover: জায়গায় ফিট করে crop করে
* object-center: মাঝ থেকে crop
* hover:scale-105: সুন্দর zoom-in effect

**🔚 উপসংহার**

ইমেজ নিয়ে কাজ করলে নিচের ৫টি ক্লাস সবচেয়ে বেশি কাজে লাগে:

1. object-cover – crop সহ ফিট করতে
2. w-full / w-[value] – প্রস্থ কন্ট্রোল করতে
3. h-[value] – উচ্চতা ঠিক রাখতে
4. rounded-lg – বর্ডার রেডিয়াস দিতে
5. mx-auto, block – center করতে

**Tailwind Table Design Releted Clases:**

Tailwind CSS-এ **টেবিল (Table)** বানানোর সময় সবচেয়ে বেশি ব্যবহৃত ক্লাসগুলো **সহজ ভাষায় ব্যাখ্যা সহ**

**🔹 1. table, table-auto, table-fixed**

| **ক্লাস** | **কাজ** |
| --- | --- |
| table | HTML টেবিলের মতো ডিজাইন তৈরি করে |
| table-auto | কলামের প্রস্থ content অনুযায়ী হবে *(default)* |
| table-fixed | প্রতিটি কলামের প্রস্থ সমান করে (fixed layout) (column width নিজে define করতে হয়) |

✅ উদাহরণ:

html

CopyEdit

<table class="table table-auto w-full">

**🔹 2. w-\* এবং h-\* (Width & Height)**

| **ক্লাস** | **কাজ** |
| --- | --- |
| w-full | টেবিল container-এর পুরো width নিবে |
| w-[600px] | নির্দিষ্ট width দিবে |
| h-[300px] | নির্দিষ্ট height দিবে |

**🔹 3. border, border-collapse, border-separate**

| **ক্লাস** | **কাজ** |
| --- | --- |
| border | টেবিলের cell ও বাইরের বর্ডার তৈরি করে |
| border-collapse | cell গুলোর বর্ডার একত্র করে (টাইট দেখায়) |
| border-separate | cell গুলোর মাঝে ফাঁকা থাকে |

✅ উদাহরণ:

html

CopyEdit

<table class="table w-full border border-collapse">

**🔹 4. Padding & Text Alignment**

| **ক্লাস** | **কাজ** |
| --- | --- |
| px-4 / py-2 | horizontal / vertical padding |
| text-left / text-center | টেক্সটের অবস্থান /টেক্সট অ্যালাইনমেন্ট |
| text-sm / text-lg | font size |

**🔹 5. Row Color (Zebra Effect)**

| **ক্লাস** | **কাজ** |
| --- | --- |
| odd:bg-gray-100 | বিজোড় row গুলোতে ব্যাকগ্রাউন্ড |
| even:bg-white | জোড় row গুলোতে ব্যাকগ্রাউন্ড |

**🔹 6. Scroll & Responsive**

| **ক্লাস** | **কাজ** |
| --- | --- |
| overflow-x-auto | ছোট স্ক্রিনে টেবিল স্ক্রল করা যাবে (responsive scroll (mobile/tablet view এ overflow হলে) |
| min-w-[600px] | মিনিমাম টেবিল সাইজ সেট করা |
| min-w-\* /max-w-\* | মিনিমাম/ম্যাক্সিমাম width কন্ট্রোল |

**🔚 সংক্ষেপে:**

* ✅ w-[value], h-[value] দিয়ে সাইজ নির্ধারণ করো
* ✅ table-auto or table-fixed দিয়ে layout কন্ট্রোল করো
* ✅ overflow-x-auto দিয়ে responsive scroll দাও

✅ উদাহরণ:

html

CopyEdit

<div class="overflow-x-auto">

<table class="table w-full min-w-[600px]">

**🧪 Full Example (Responsive Table):**

html

CopyEdit

<div class="overflow-x-auto">

<table class="table-auto w-full border border-gray-300 text-left text-sm">

<thead class="bg-gray-100">

<tr>

<th class="px-4 py-2 border">Name</th>

<th class="px-4 py-2 border">Email</th>

<th class="px-4 py-2 border">Role</th>

</tr>

</thead>

<tbody>

<tr class="odd:bg-white even:bg-gray-50">

<td class="px-4 py-2 border">John Doe</td>

<td class="px-4 py-2 border">john@example.com</td>

<td class="px-4 py-2 border">Admin</td>

</tr>

<tr class="odd:bg-white even:bg-gray-50">

<td class="px-4 py-2 border">Jane Smith</td>

<td class="px-4 py-2 border">jane@example.com</td>

<td class="px-4 py-2 border">Editor</td>

</tr>

</tbody>

</table>

</div>

**🧠 সংক্ষেপে মনে রাখার জন্য:**

* ✅ table + table-auto বা table-fixed
* ✅ w-full / h-\* / overflow-x-auto = responsive layout
* ✅ border + border-collapse = টেবিল বাউন্ডারি
* ✅ text-left / px-4 py-2 = readable layout

**🔹 gap – (row এবং column দুইটার মাঝেই গ্যাপ)**

**Tailwind CSS**-এ gap-related সব utility class-গুলো নিচে ভাগ করে সহজভাবে ব্যাখ্যা করছি — যাতে তুমি সহজে বুঝতে পারো কোনটা কখন ব্যবহার করতে হয়।

**🔹 1. Utilities**

| **ক্লাস** | **কাজ** |
| --- | --- |
| gap-0 | কোন গ্যাপ থাকবে না |
| gap-1 | 0.25rem (4px) |
| gap-2 | 0.5rem (8px) |
| gap-3 | 0.75rem (12px) |
| gap-4 | 1rem (16px) |
| gap-5 | 1.25rem (20px) |
| gap-6 | 1.5rem (24px) |
| gap-8 | 2rem (32px) |
| gap-10 | 2.5rem (40px) |
| gap-12 | 3rem (48px) |
| gap-[30px] | কাস্টম গ্যাপ (30px ইত্যাদি) |

✅ gap-\* মূলত **flex/grid** layout-এর row এবং column এর মাঝখানে gap দেয়।

**🔹 2. gap-x-\* – Horizontal (বামে-ডানে) গ্যাপ**

| **ক্লাস** | **কাজ** |
| --- | --- |
| gap-x-2 | বাম-ডান গ্যাপ 8px |
| gap-x-4 | বাম-ডান গ্যাপ 16px |
| gap-x-8 | বাম-ডান গ্যাপ 32px |
| gap-x-[50px] | কাস্টম horizontal গ্যাপ (50px) |

👉 যখন তুমি flex-row বা grid-cols-\* ব্যবহার করো, তখন horizontal গ্যাপ এর জন্য ব্যবহার করো।

**🔹 3. gap-y-\* – Vertical (উপর-নিচে) গ্যাপ**

| **ক্লাস** | **কাজ** |
| --- | --- |
| gap-y-2 | ওপর-নিচ গ্যাপ 8px |
| gap-y-6 | ওপর-নিচ গ্যাপ 24px |
| gap-y-10 | ওপর-নিচ গ্যাপ 40px |
| gap-y-[20px] | কাস্টম vertical গ্যাপ (20px) |

👉 যখন তুমি flex-col বা grid-rows-\* ব্যবহার করো, তখন vertical গ্যাপ এর জন্য ব্যবহার করো।

**🧪 উদাহরণ ১: Flex Row + Horizontal Gap**

html

CopyEdit

<div class="flex gap-x-6">

<div class="bg-blue-200 p-4">Box 1</div>

<div class="bg-green-200 p-4">Box 2</div>

</div>

**🧪 উদাহরণ ২: Flex Column + Vertical Gap**

html

CopyEdit

<div class="flex flex-col gap-y-4">

<div class="bg-red-200 p-4">Top Item</div>

<div class="bg-yellow-200 p-4">Bottom Item</div>

</div>

**🧠 Bonus: Responsive Gap**

html

CopyEdit

<div class="flex gap-2 sm:gap-4 md:gap-6 lg:gap-10">

<!-- Content Here -->

</div>

* ছোট স্ক্রিনে gap-2, মাঝারি স্ক্রিনে gap-4, আরও বড় স্ক্রিনে gap-6...

**✅ সংক্ষেপে মনে রাখার ট্রিক:**

| **চাইলে** | **ব্যবহার করো** |
| --- | --- |
| সব দিকেই gap | gap-\* |
| শুধু বাম-ডানে (Horizontal) | gap-x-\* |
| শুধু উপর-নিচে (Vertical) | gap-y-\* |
| স্ক্রিন অনুযায়ী gap | sm:gap-\*, md:gap-\* |