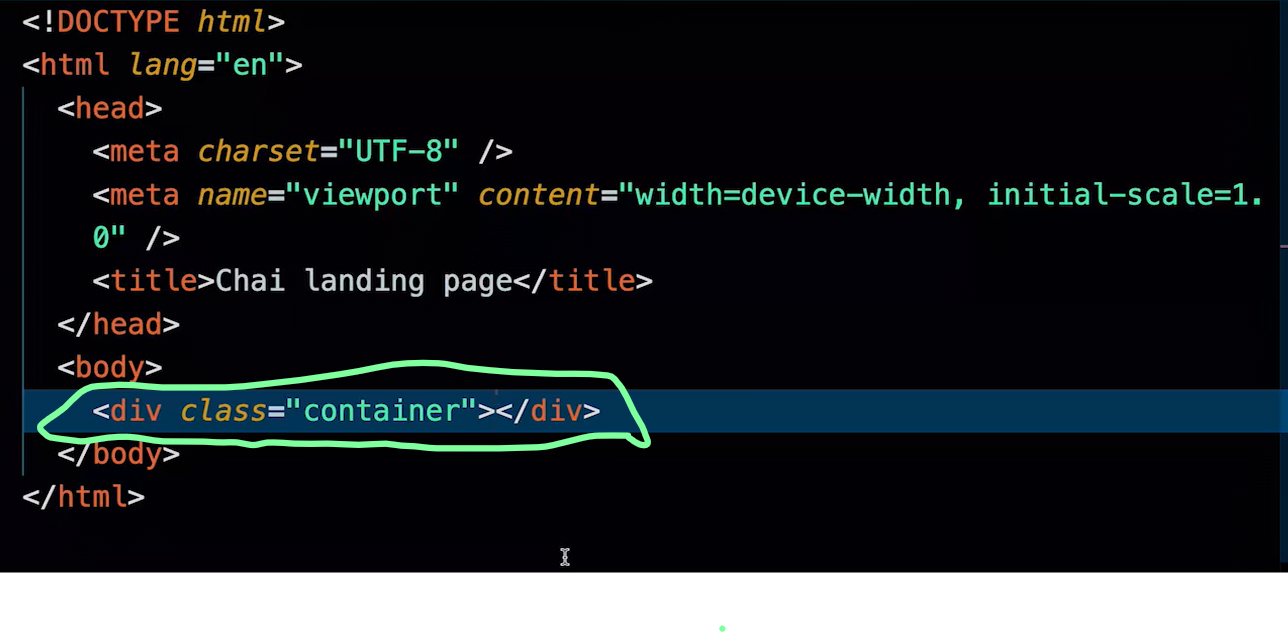
**24. Build a custom coming soon Target Project**

* I don’t want to teach in a traditional, boring way.
* My goal is to dive straight into real-world development, so you see how things actually work.
* With that in mind, I want to build something practical with you.
* Not a full website, but a real, working web page.
* The faster you put up the HTML content, the easier it is for you to design.



* Div.container + tab => <div class=”container”></div>. It is common practice to wrap everything around container.
* What is <div class=”container”></div>?

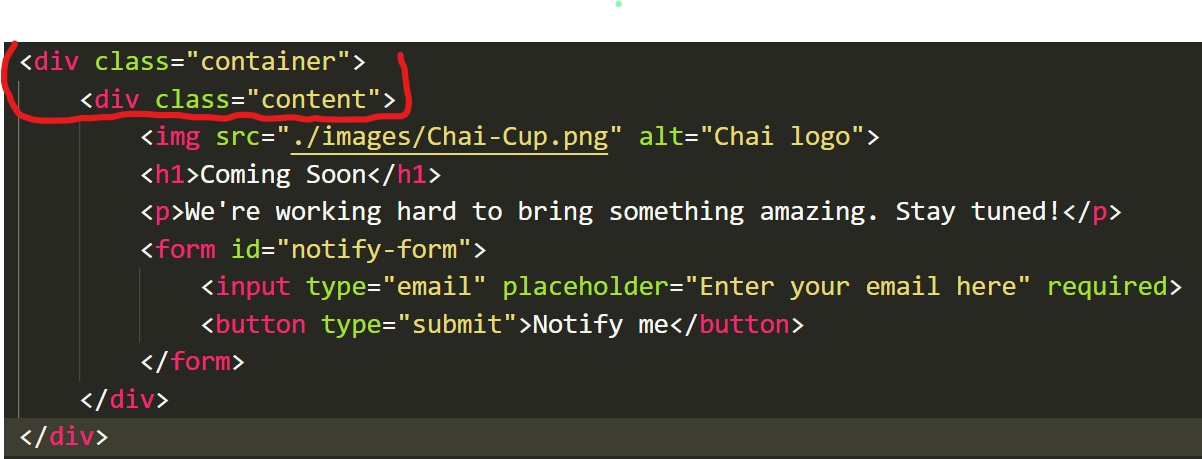
**Questions Powered by Copilot**👾👾👾

**📦 What is <div class="container">?**

* It’s a **div element** with a class called "container", often used in CSS frameworks like **Bootstrap**.
* Think of it as a **content wrapper**—a box that holds and organizes everything inside your webpage.
* The .container class typically does three important things:
  + 🧲 **Centers** your content horizontally on the page
  + 📏 **Limits the width** so it doesn’t stretch edge-to-edge on large screens
  + 🧱 **Adds padding** on the sides to keep content from touching the edges

**🧰 Why wrap everything in a container?**

* Imagine building a house—you wouldn’t just place furniture on the ground; you’d put it inside rooms. Similarly, the .container is like the **room** that holds all your webpage elements.
* It provides a **structured layout**, making your design:
  + ✅ **Neat** – everything is aligned and spaced properly
  + 📱 **Responsive** – it adapts to different screen sizes
  + 🎯 **Consistent** – spacing and alignment stay uniform across pages



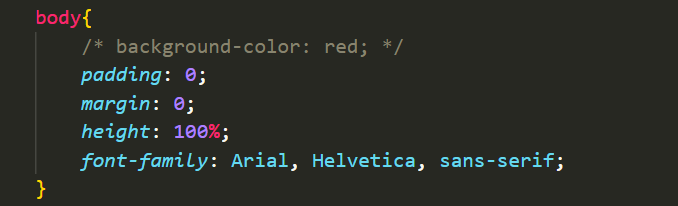
* <div class="container"> - we've created a block here.
* <div class="content"> - Created another sub-block for content.

We can create sub-blocks for various needs like for images, content etc.

* <input type="email" placeholder="Enter your email here" required>

1. Anything written in placeholder property will be showing-up in a ghost way in the text field. Ghost way means it is visible there, but as soon as you write something, it stops showing up.
2. "required" property added in as an input element, will not allow users to submit empty text fields while submitting the form.

**Style.css**



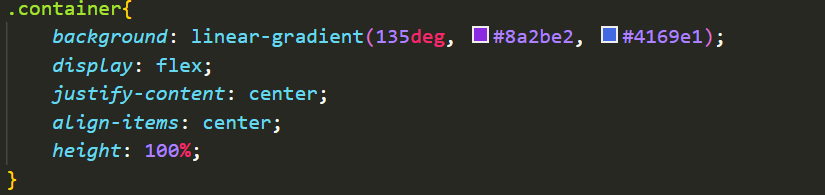
* At first we select body element and remove all the padding & margin, set the height to 100% and font-family is set to “Arial, Helvetica, sans-serif”.

Properties to keep in mind while styling body:

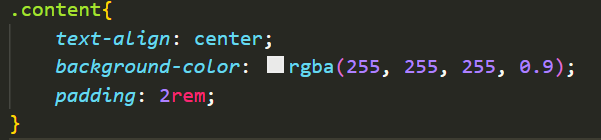
1. Padding.
2. Margin.
3. Font-family.
4. Height.
5. Background color.
6. Color.

A screen shot of a computer

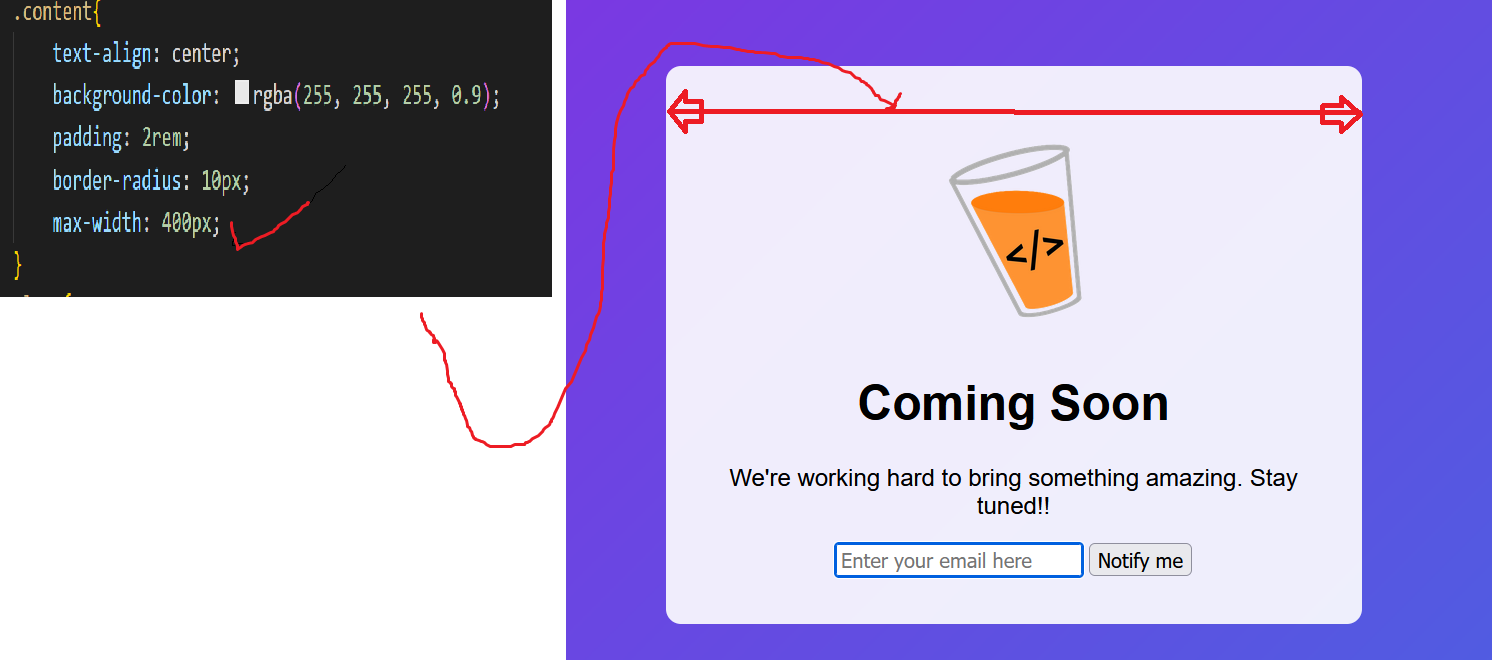
AI-generated content may be incorrect.



* We select the container, hard-code linear-gradient(135deg, #8a2be2, #4169e1);, set display prop. to flex, then justify-content to center and align-items to center respectively, also height to 100%.



* We select the content class, as content class is a div element that stores text elements, we set the text-align property to center, background-color to rgba(255, 255, 255, 0.9), 2rem padding is set.
* rem means with respect to m, that is, 1 rem of length is equal to the width of letter “M” in that particular font.



**Prompt**:

What does max-width:400; in the above context shown in the above image, .content class is a selector for the white box and the content present in it.

**Response from Copilot**:  
Great question, PC! In the CSS rule:

.content {

max-width: 400px;

}

💡 **What it means:**  
The max-width: 400px; property ensures that the .content box — which includes the white background, text, image, and input field — will never grow wider than **400 pixels**, no matter how big the screen or browser window gets.

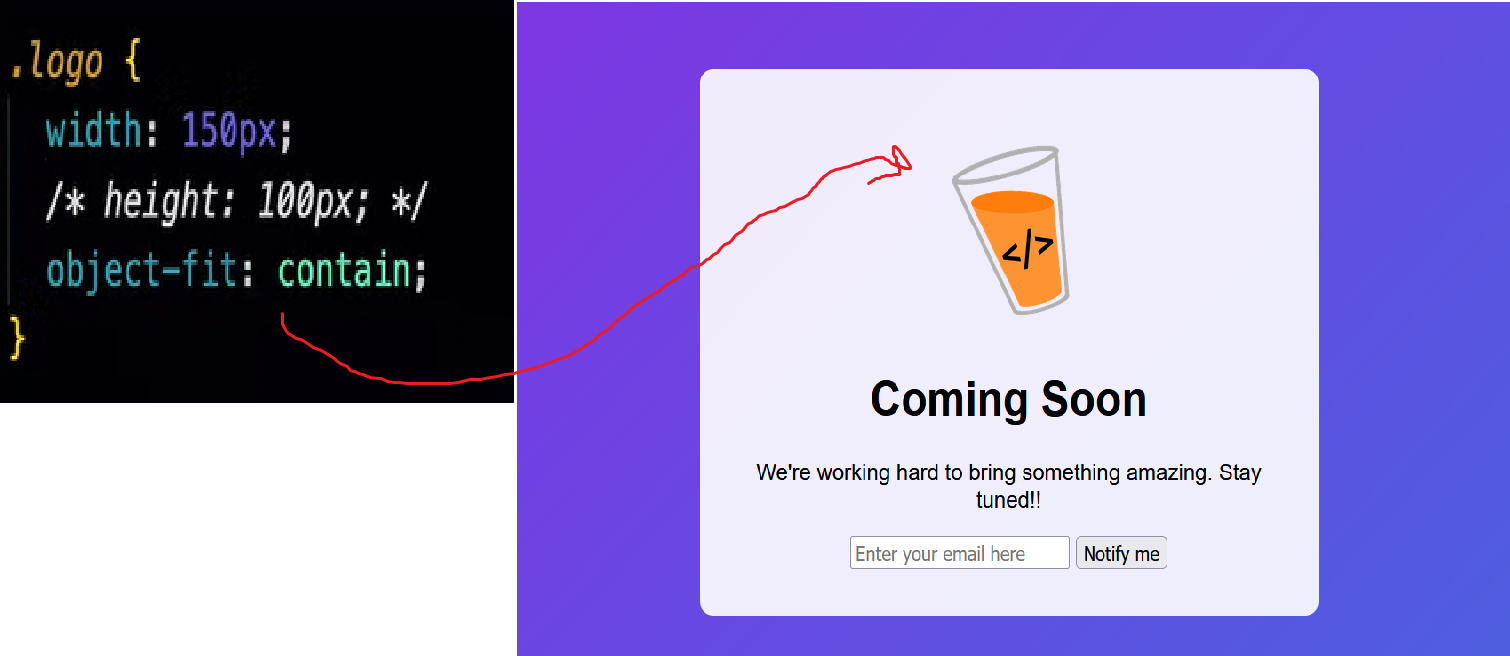
📱 **Responsive behavior:**  
If the screen is **smaller than 400px**, the .content box will **shrink to fit**. But if it’s larger, it stops expanding once it hits 400px.

🔎 **Why it's useful in your layout:**  
It maintains a clean, readable design. On big screens, it prevents the content from stretching too wide, while still looking good on mobile.

Let me know if you’d like to see how that affects alignment or how to make it responsive with @media queries!

A black background with white text

AI-generated content may be incorrect.



**Prompt**:

In the above image, what object-fit:contain; mean, connect with the context. .logo class is a selector for the logo image shown in the above image.

**Response from Copilot**:

Absolutely, PC! Let’s break it down in context:

.logo {

width: 150px;

object-fit: contain;

}

🧩 **What object-fit: contain; means:**  
It tells the browser:

“Fit the entire logo image inside the box without cutting any part off—and keep the proportions intact.”

🔍 **Applied to your context:**

* The .logo class is controlling how the logo image (that glass-with-orange-liquid and code symbol) behaves inside its box.
* The box has a fixed width of 150px, and since height is commented out, the height will auto-adjust based on the image’s aspect ratio.
* object-fit: contain; ensures the image scales down (or up) to fit entirely **within** that box.
* **No cropping happens**, and it won’t stretch or distort the image. You get the full visual, scaled neatly inside the container.