**\* { box-sizing: border-box }**

**🔹 What it means**

* \* = the **universal selector** (matches every element on the page).
* box-sizing tells the browser how to calculate an element’s **width** and **height**.

There are two main modes:

1. **content-box** (default in browsers):
   * width = only the content area.
   * Padding and border get **added on top**.
   * Example:
   * div {
   * width: 200px;
   * padding: 20px;
   * border: 5px solid;
   * }

👉 Total width = 200 + 20 + 20 + 5 + 5 = **250px**

1. **border-box**:
   * width = content + padding + border all together.
   * The content area shrinks to make room.
   * Example (same as above):  
     👉 Total width stays **200px**.

**🎯 Why it’s used**

* Makes layouts easier to manage.
* Avoids unexpected growth when adding padding/borders.
* That’s why developers reset everything to border-box.

**2️⃣ html, body { height: 100% }**

**🔹 What it means**

* Applies to <html> and <body> elements.
* Sets their height to **100% of the viewport** (the visible browser window).

By default:

* <html> and <body> don’t stretch the full height of the screen — they only expand as much as the content needs.
* This rule forces them to fill the screen vertically.

**🎯 Why it’s used**

* Helps when making **full-height layouts**, like:
  + Backgrounds that should cover the whole screen.
  + Flexbox or grid layouts that need to align to the viewport.
  + Sticky footers at the bottom of the page.

**📌 Put Together**

\* { box-sizing: border-box; }

html, body { height: 100%; }

👉 Means:

1. **Every element** sizes itself more predictably (border-box).
2. **The page wrapper (html and body)** always fills the screen height.

⚡ That’s why you’ll see these two lines in almost every modern CSS starter template.