What you will need!

- **An HTML file**
  - Well, what's a hairdresser without a customer right? We're going to need something to "style" with CSS. It doesn't have to be a very big HTML file. A simple html file with a few headers and paragraphs will do.
- **A browser**
  - You're going to need a browser. Popular browsers include google chrome, Microsoft edge, safari, and firefox. We're going to use the browser to open the HTML document that we will style using CSS. Browsers are going to interpret the CSS and HTML documents we will be creating.
- **Text editor**
  - There are numerous text editors out there. The most common being "notepad" in windows. Other fancy text editors such as "atom" and "sublime" could also be used but for this course I wouldn't recommend you using those text editors because it is simply overkill. The goal of this course is to teach you the rudiments of CSS and not advanced styling techniques. It does not matter which text editor you will use, the output will still be the same. You might be asking, "if all the outputs will still be the same no matter which text editor I use, then why even bother use other text editors?". Well it's because fancy text editors make it easier for people to code. Text editors such as "atom" have features similar to autocomplete and I don't recommend you to get used to autocomplete because we have to get the basic embedded into your mind first. Links to popular HTML and CSS text editors will be displayed down below.

Believe it or not, these three things are all you're going to need for this course. I will be seeing you on the next chapter.

- **What is CSS?**
  - CSS stands for Cascading Style Sheet. It is most commonly used alongside HTML. In this course, we're going to use CSS to give life to the HTML document we have created in the previous HTML course.
  - It is important to know that there are different ways to do the same thing in CSS. This pathway will not cover all the possible ways to get a certain task done. Instead, it will cover the basics of CSS.

Lesson 1: Methods for adding CSS:

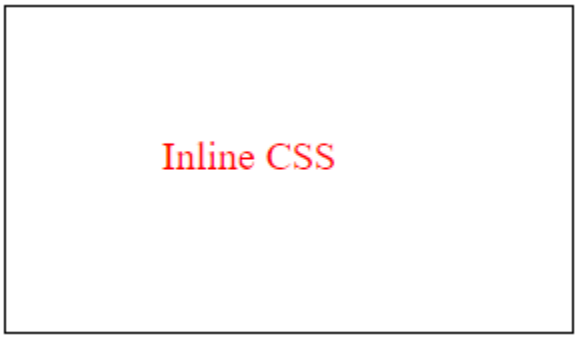
1. **Inline CSS** – Inline CSS is just styling your HTML inside the actual element tag you want to style.

CODE:

```
<head>
</head>
<body>
  <p style="color:red"> Inline CSS </p>
</body>

```

OUTPUT:

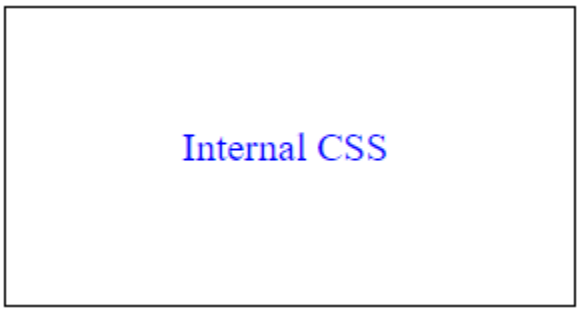


2. **Internal CSS** – Internal CSS is styling your HTML in the same document.

CODE:

```
<head>
  <style>
    p {color:blue;}
  </style>
</head>
<body>
  <p> Internal CSS </p>
</body>
```

OUTPUT:



3. **External CSS** – External CSS is creating a separate document for styling your CSS document. In this course, we will use this method as it is the best of all the three existing methods.

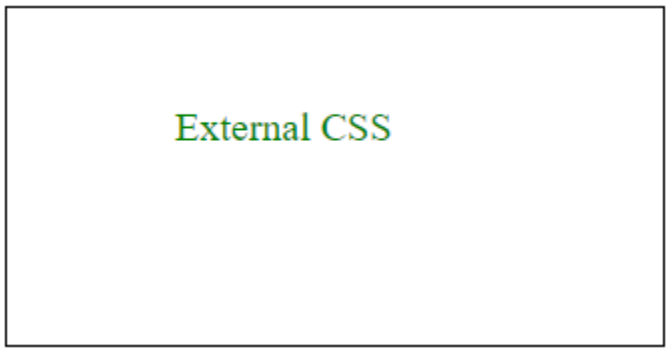
Inside the HTML document:

```
<head>
  <link rel = "stylesheet" href = "stylesheet.css">
</head>
<body>
  <p> External CSS </p>
</body>
```

Inside the External CSS document:

```
p{
  color: green;
}
```

OUTPUT:



Lesson 2: UNITS in CSS

Absolute lengths - are fixed no matter how big your screen/display is. For example, I created a box that is 10px by 10px (2mm x 2mm). If I were to display that box in a giant arena tv, the size would still be 10 pixels by 10 pixels because px is an absolute unit of measurement. See picture below for the list of absolute units of measurement in CSS

Unit	Description
cm	centimeters
mm	millimeters
in	inches (1in = 96px = 2.54cm)
px *	pixels (1px = 1/96th of 1in)
pt	points (1pt = 1/72 of 1in)
pc	picas (1pc = 12 pt)

Relative lengths - Relative length units specify a length relative to another length property. Relative length units scale better between different rendering medium. See picture below for full details of all relative length units in CSS.

Unit	Description
em	Relative to the font-size of the element (2em means 2 times the size of the current font)
ex	Relative to the x-height of the current font (rarely used)
ch	Relative to the width of the "0" (zero)
rem	Relative to font-size of the root element
vw	Relative to 1% of the width of the viewport*
vh	Relative to 1% of the height of the viewport*
vmin	Relative to 1% of viewport's* smaller dimension
vmax	Relative to 1% of viewport's* larger dimension
%	Relative to the parent element