

# Antes de Empezar

Vamos a aprender la estructura de la aplicación y qué son HTML y CSS.

[html 1](#)

# Empezamos con HTML

Vamos a hacer un código muy simple con HTML!

## El Primer Código con HTML

1. Abrimos la aplicación de NotePad

*Asumimos que se usan Windows OS*

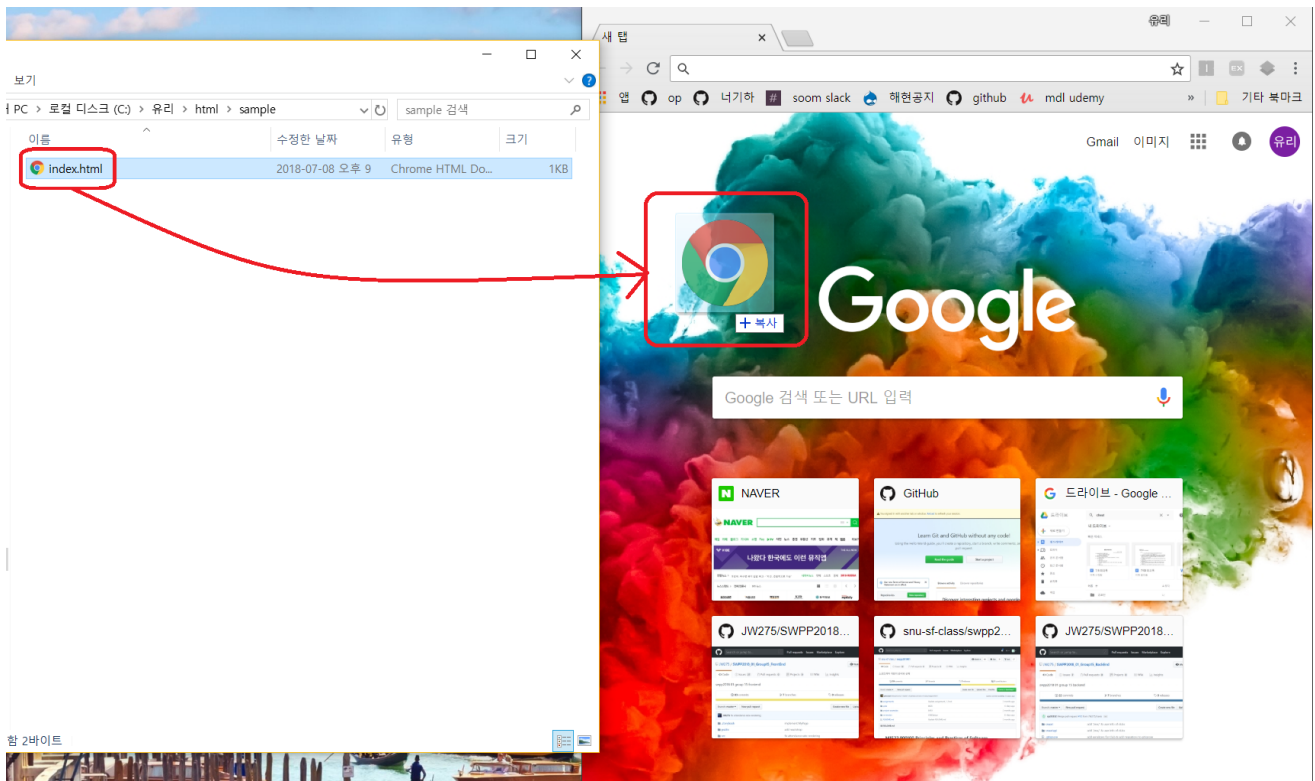
2. Escribimos el siguiente código

```
<h1>This is my first page.</h1>
```

3. Guardamos el archivo como `index.html`

*El formato debe ser 'All files'*

4. Buscamos el archivo desde Explorador de Archivos y arrastramos el archivo al tab nuevo en Chrome



5. Ya hicimos nuestro primer código con HTML!

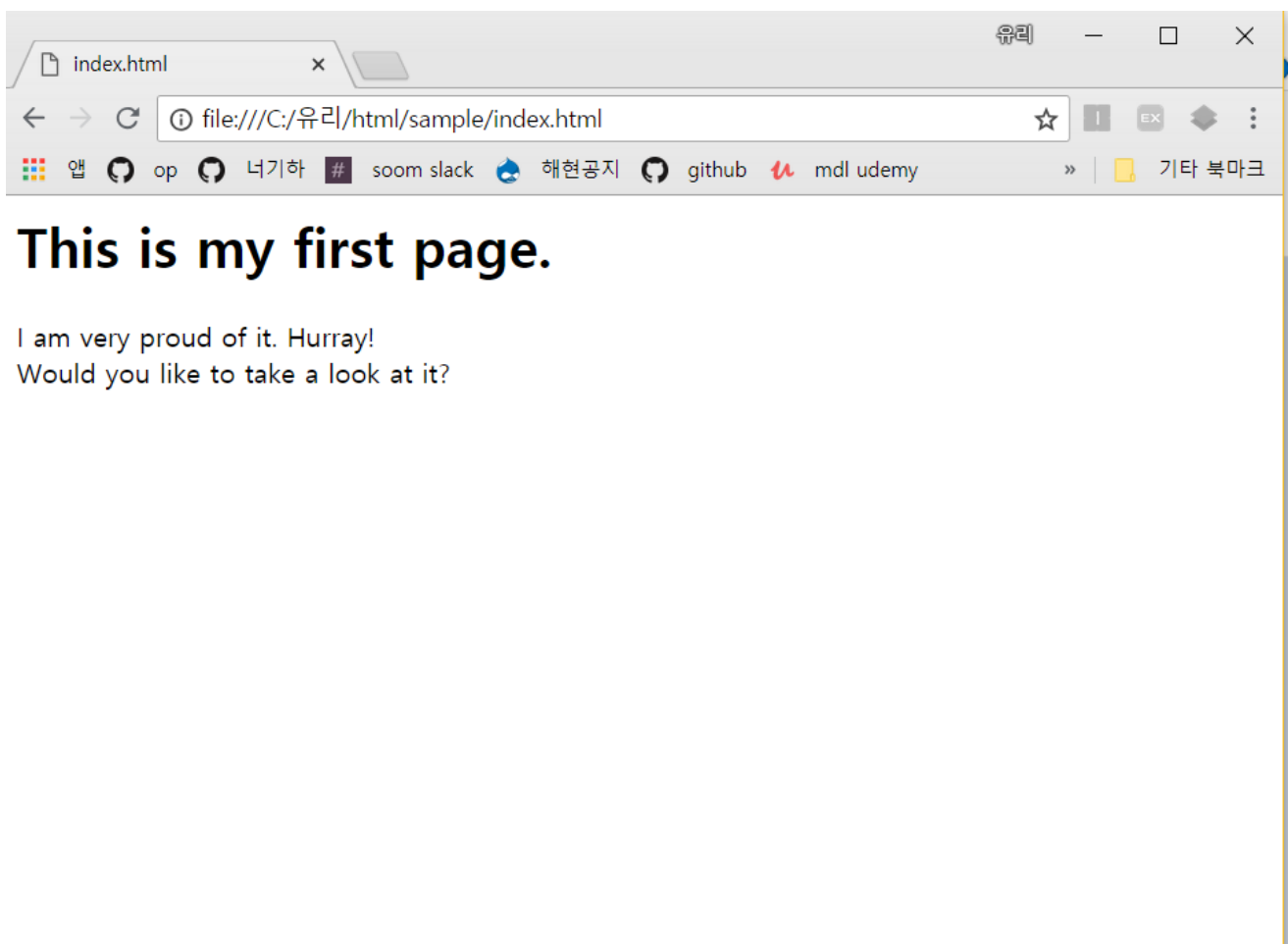
El `<h1>` es heading 1 tag, que se usa para definir el título es muy importante

# Vamos a añadir más códigos!

Añadimos más códigos a nuestro archivo de `index.html` como abajo.

```
<!-- start of page -->
<h1>This is my first page.</h1>
<!-- explanation about my feelings -->
<p>I am very proud of it.
  Hurray! <br />
  Would you like to take a look at it?</p>
```

Después, lo arrastramos de nuevo al navegador para ver cómo ha cambiado la página. La página aparece como abajo.



Qué cambios podemos notar? Vamos a revisar cada uno.

- **No podemos ver los contenidos entre `<!--` and `-->` en el navegador!**

Es porque `<!--` and `-->` se usa para indicar el comienzo y el final de un comentario.

Un **comment** es una pieza del código que ignora el navegador. Se usa para aclarar los códigos. Usar comments es una buena práctica para mejorar la legibilidad de código sobre todo cuando los códigos sean más complicados.

- **El contenido mostrado por `<p>` tag es más pequeño**

El `<p>` tag se usa para indicar los párrafos.

- **Pulsar Enter en el código no significa una nueva línea en el navegador!**  
HTML ignora la nueva línea hecha por pulsar <Enter> key.  
(Tampoco se refleja dos o más espacios entre letras)
- El `<br>` tag cambia la línea. (br=break the line)
- El `<br>` tag no tiene end tag!  
Los elementos con solo un start tag sin contenido o sin un end tag se llaman **empty elements**.  
Empty elements puede ser ambos opening tag y closing tag.( `<br />` es exactamente igual con `<br>` )

## Nested Structure de HTML

---

### Lista de HTML

Cambiamos el archivo de `index.html` con el siguiente código.

```
<h1>Things to do</h1>
<ol>
  <li>Listen carefully to today's class</li>
  <li>Have a splendid dinner</li>
  <li>Go to the grocery store to buy the following items</li>
</ol>
```

Vamos a revisar cómo aparece en la página en el navegador.

- El `<ol>` tag se usa para lista ordenada
- El `<li>` tag se usa para representar cada artículo en la lista

Si queremos añadir otra lista abajo del tercer `<li>` elemento? Como ponemos `<li>` tags entre la comienzo y el fin de `<ol>` tags, podemos añadir elementos en otros elementos! Eso se llama **nesting**.

Cambiamos parte del archivo `index.html` como siguiente.

```
...
  <li>Go to the grocery store to buy the following items
    <ul>
      <li>Eggs</li>
      <li>Onions</li>
      <li>Cereal</li>
    </ul>
  </li>
...
```

Revisamos cómo se ve en el navegador.

Como podemos suponer, el `<ul>` tag se usa para la lista desordenada.

En HTML, es general cambiar líneas y dejar 2 espacios cuando se usa los elementos nesting, pero no es obligatorio. (Aun así, les recomiendo seguir esta costumbre para la legibilidad de código.)

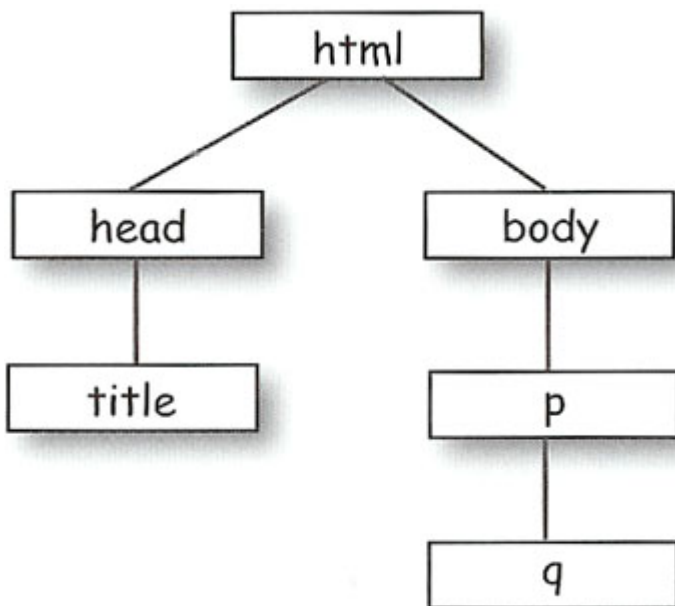
[Learn more about HTML lists](#)

## Entendimiento de Nested Relations

Podemos mejorar nuestro entendimiento de nested relations usando **trees** para visualizarlas. Aquí hay un ejemplo de un código simple de HTML.

```
<html>
  <head>
    <title>About Nested Relations</title>
  </head>
  <body>
    <p>
      To quote Yuri, <q>It is very important to understand nested relations in HTML.</q>
    </p>
  </body>
</html>
```

La estructura del código puede convertirse en el diagrama de árboles(tree diagram).



En el diagrama, el tag `<html>` es el root tag.

El padre de `<head>` es `<html>`, mientras `<head>` tiene un hijo, `<title>`.

## Estructura de Documento HTML

La siguiente es la estructura básica de documento HTML.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Yuri's Page!</title>
  </head>
  <body>
    <h1>This is my first page.</h1>
    <p>I am very proud of it.</p>
  </body>
</html>
```

- `<!DOCTYPE html>` permite al navegador para que sepa que estamos usando el código moderno de HTML.  
Debe estar en el inicio del código para que el navegador lea el código con HTML5.
- El código entero está dentro del elemento de `<html>` . El navegador ya sabe que el archivo es de HTML.
- El elemento de `<head>` tiene información sobre la página web.
  - El `<meta>` tag con el atributo `charset` definido como `"utf-8"` informa al navegador que queremos acceso a utf-8 character sets en la página.

\_ `<meta>` es también empty element, como `<br>!`\_

(Vamos a aprender qué son atributos pronto. No te preocupes si no los sabes ahora.)

- El `<title>` tag contiene el título de la página, que se ve en la parte arriba en el tab del navegador.
- El contenido está dentro del elemento de `<body>` .  
**Solo** el contenido dentro del elemento de `<body>` está mostrado en el navegador.