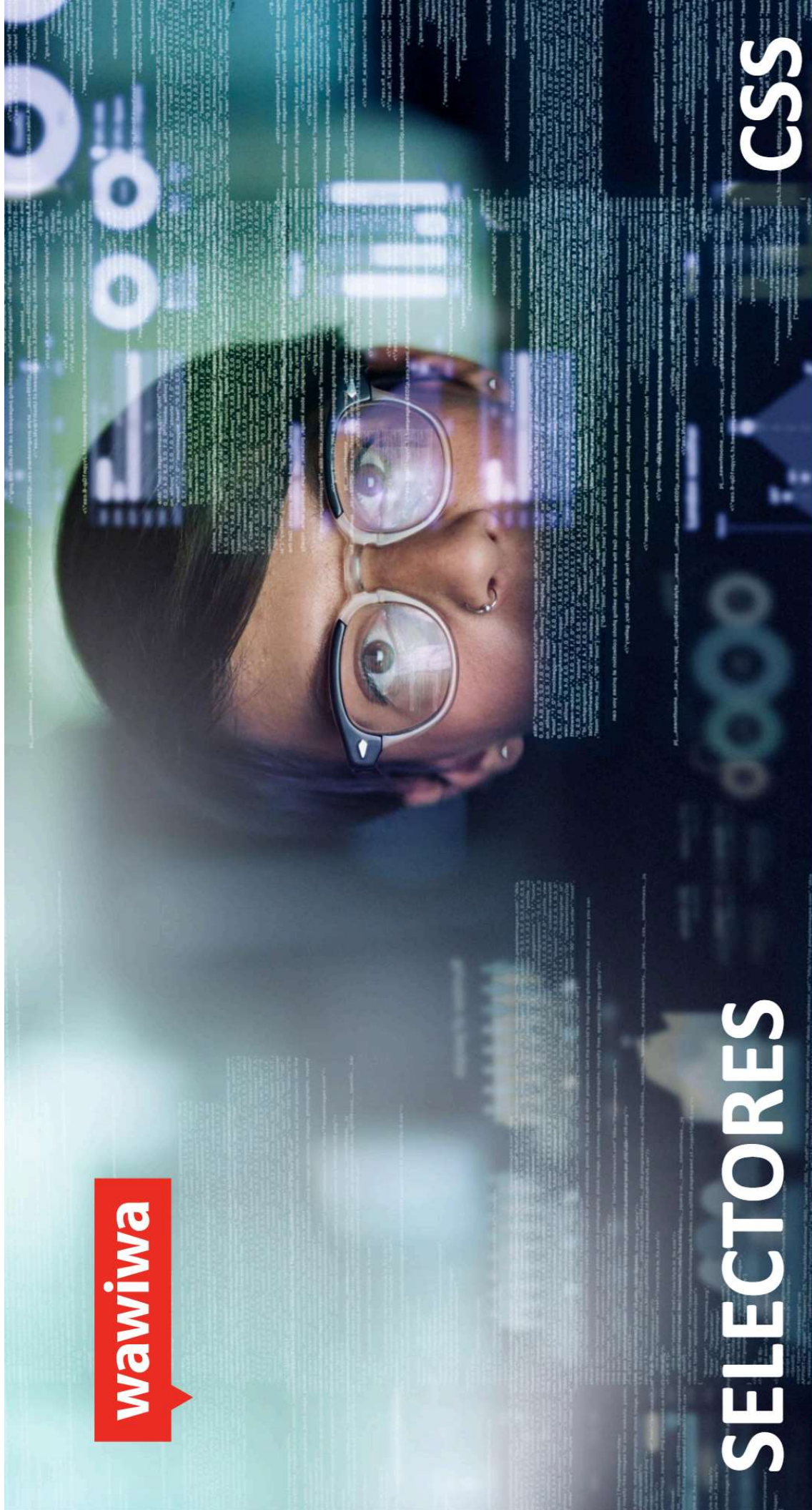


wawiwa

# SELECTORES

2022 © Wawiwa Tech | Confidential

# CSS



# Selectors (Selectores)

¿Cómo podemos manipular los elementos HTML en CSS? Usamos selectores

Los selectores simplemente se refieren a los elementos a los cuales se les aplican las reglas CSS, son lo que está siendo “selecto” para cada regla.

Las siguientes subsecciones no cubren cada selector que existe, pero son, por mucho, los más comunes y los que deberías acostumbrarte a usar primero.

<https://www.theodinproject.com/lessons/foundations-css-foundations#selectors>

# Selector Universal

El selector universal selecciona los elementos de cualquier tipo, por ende el nombre “universal” y la sintaxis para él es un simple asterisco. En el ejemplo de abajo, todos los elementos deberían tener el color: morado aplicado.

```
1 | * {  
2 |   color: purple;  
3 | }
```

# Selector Type (tipo)

Un selector type (o selector de elemento) selecciona todos los elementos de un mismo tipo y su sintaxis es solamente el nombre del elemento:

```
1 <!-- index.html -->
2
3 <div>Hello, World!</div>
4 <div>Hello again!</div>
5 <p>Hi...</p>
6 <div>Okay, bye.</div>
```

```
1 /* styles.css */
2
3 div {
4   color: white;
5 }
```

Aquí, los tres elementos `<div>` serían seleccionados, mientras que el elemento `<p>` no.



# Selector Class (class)

Los selectores Class van a seleccionar a todos los elementos que tengan esa clase, que es solamente un atributo que se asigna en el elemento HTML. Así es como puedes agregar una clase a un tag HTML y se selecciona con CSS:

```
1 <!-- index.html -->
2
3 <div class="alert-text">
4   Please agree to our terms of service.
5 </div>
```

```
1 /* styles.css */
2
3 .alert-text {
4   color: red;
5 }
```

# Selector ID

Los selectores ID son parecidas a los selectores de clase. Señalan un elemento con el ID asignado, que es otro atributo que se puede poner en el elemento HTML:

```
1 | <!-- index.html -->
2 |
3 | <div id="title">My Awesome 90's Page</div>
```

```
1 | /* styles.css */
2 |
3 | #title {
4 |     background-color: red;
5 | }
```

# Grouping Selectores (de grupo)

¿Qué pasa si tenemos dos grupos de elementos que comparten algunas de las declaraciones de estilo?

Nuestros selectores `.read` y `.unread` comparten las declaraciones `color: white;` y `background-color: black;`, pero también tienen varias

declaraciones únicas cada una. Para evitar la repetición, podemos agrupar esos dos selectores como una lista separada por comas.

Los dos ejemplos (agrupando y sin agrupar) van a tener el mismo resultado, pero el segundo ejemplo reduce la repetición en las declaraciones y hace más fácil editar el `color` o el `background-color` de ambas clases a la vez.

```
1 .read {  
2   color: white;  
3   background-color: black;  
4   /* several unique declarations */  
5 }  
6  
7 .unread {  
8   color: white;  
9   background-color: black;  
10  /* several unique declarations */  
11 }
```

```
1 .read,  
2 .unread {  
3   color: white;  
4   background-color: black;  
5 }  
6  
7 .read {  
8   /* several unique declarations */  
9 }  
10  
11 .unread {  
12   /* several unique declarations */  
13 }
```

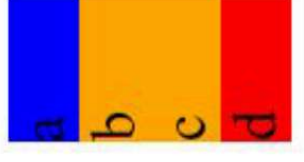
# Descendant Combinator (Combinador de descendencia)

Los combinadores nos permiten combinar varios selectores en diferente manera que agrupándolos o encadenándolos, ya que muestran una relación entre ellos.

Un combinador de descendencia solo hará que los elementos que coincidan con el último selector sean seleccionados si también tienen un ancestro (padre, abuelo, etc) que coincida con el selector anterior. Entonces algo como `.ancestor .child` va a seleccionar el elemento de la clase `child` solamente si tiene como ancestro la clase `ancestor`. Otra manera de entenderlo es: `child` será seleccionado únicamente si está anidado dentro de `ancestor`, no importa que tan profundo.

Por ejemplo:

```
<div class="alert warning">  
  This is a warning  
</div>  
  
.ancestor .contents { background-color: orange; }  
  
.ancestor { background-color: blue; }  
  
.contents { background-color: red; }
```





# Orden de las reglas

La regla que sea definida al final, es la ganadora

Para un elemento que tenga las dos clases alert y warning, la “cascada” recorre todos los factores, incluyendo herencia (aquí no hay ninguna) y especificidad (ninguna regla de estas es más específica que otra).

Ya que la regla .warning es la última, ningún otro factor puede determinar cuál regla aplicar, es la única que se aplica al elemento.

```
<div class="alert warning">  
  This is a warning  
</div>
```

```
.alert {  
  color: red;  
}
```

This is a warning

```
.warning {  
  color: orange;  
}
```



**wawiwa**

¿Preguntas?