**Type Selector**

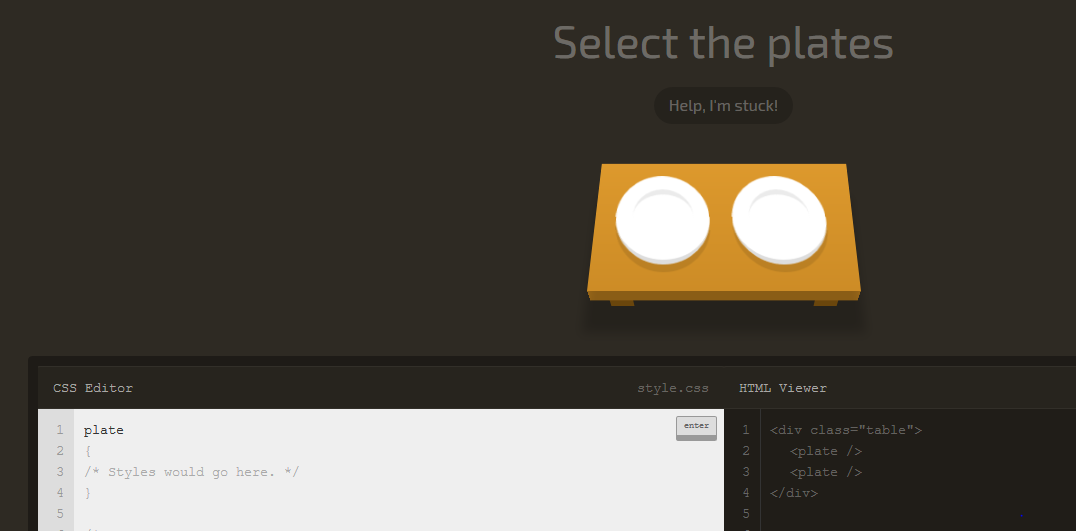
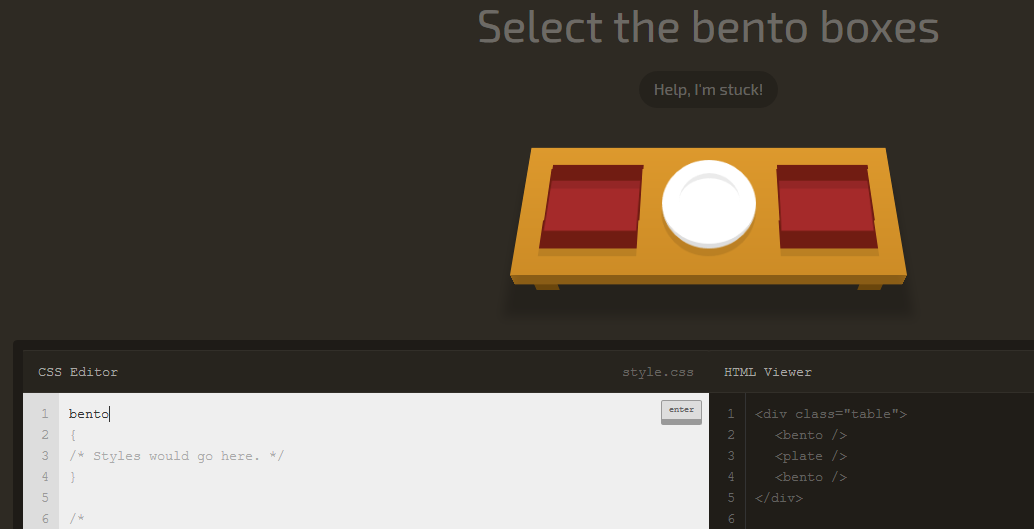
**Select elements by their type**

**A**

Selects all elements of type **A**. Type refers to the type of tag, so div, p and ul are all different element types.

**Examples**

**div** selects all div elements.

**p** selects all p elements.  
  


**ID Selector**

**Select elements with an ID**

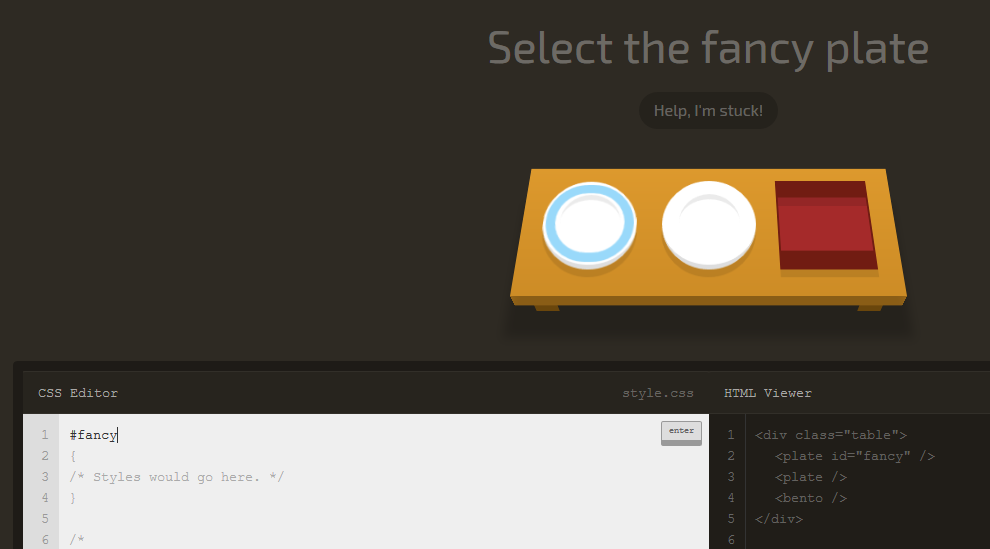
**#id**

Selects the element with a specific **id**. You can also combine the ID selector with the type selector.

**Examples**

**#cool** selects any element with **id="cool"**

**ul#long** selects ul id="long"



**Descendant Selector**

**Select an element inside another element**

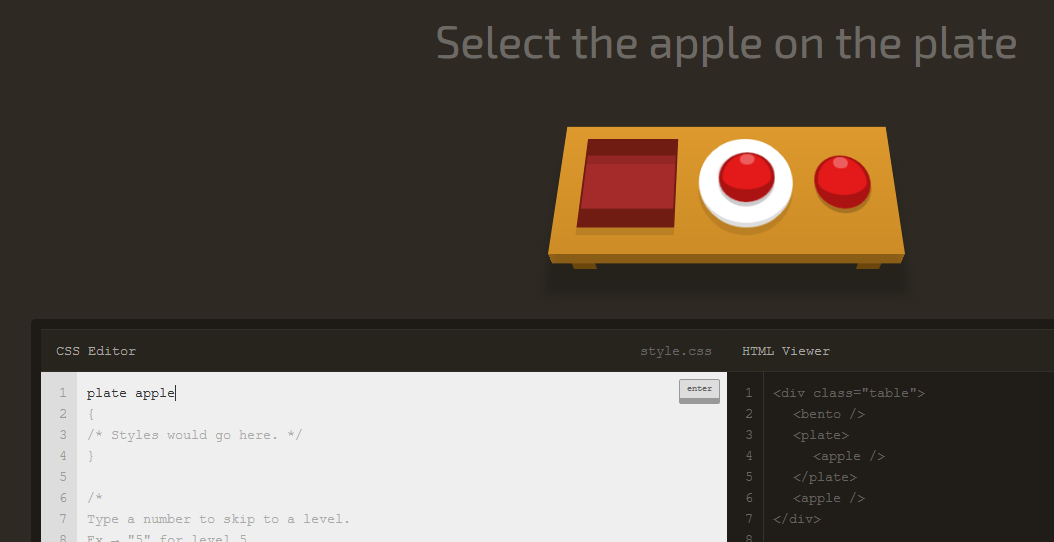
**A  B**

Selects all **B** inside of **A**. **B** is called a descendant because it is inside of another element.

**Examples**

**p  strong** selects all strong elements that are inside of any p

**#fancy  span** selects any span elements that are inside of the element with **id="fancy"**



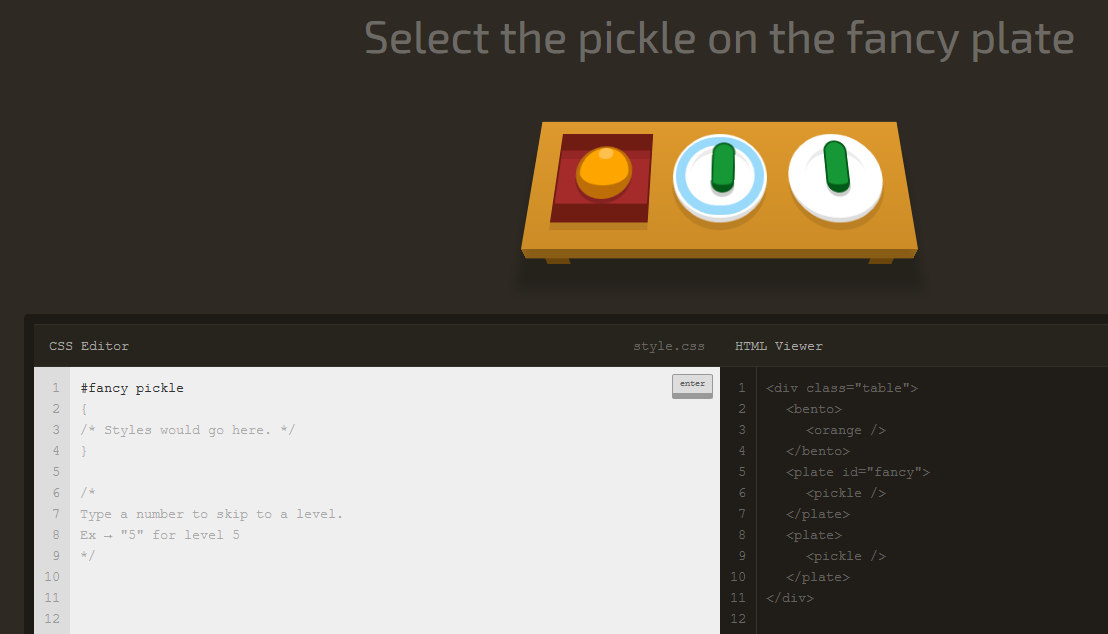
**Combine the Descendant & ID Selectors**

**#id  A**

You can combine any selector with the descendent selector.

**Examples**

**#cool span** selects all span elements that are inside of elements with **id="cool"**



**Class Selector**

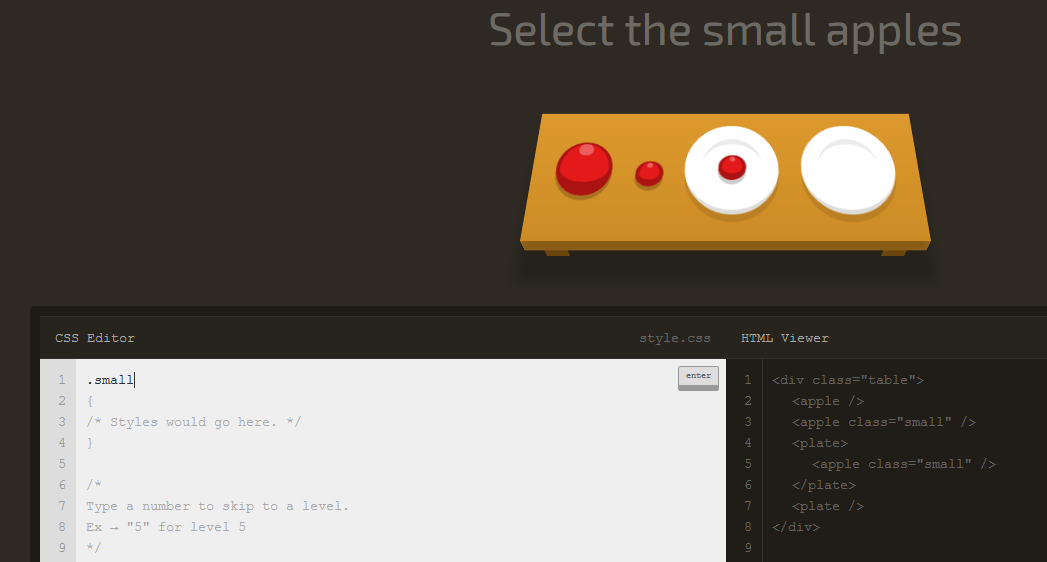
**Select elements by their class**

**.classname**

The class selector selects all elements with that class attribute. Elements can only have one ID, but many classes.

**Examples**

**.neato** selects all elements with **class="neato"**



**Combine the Class Selector**

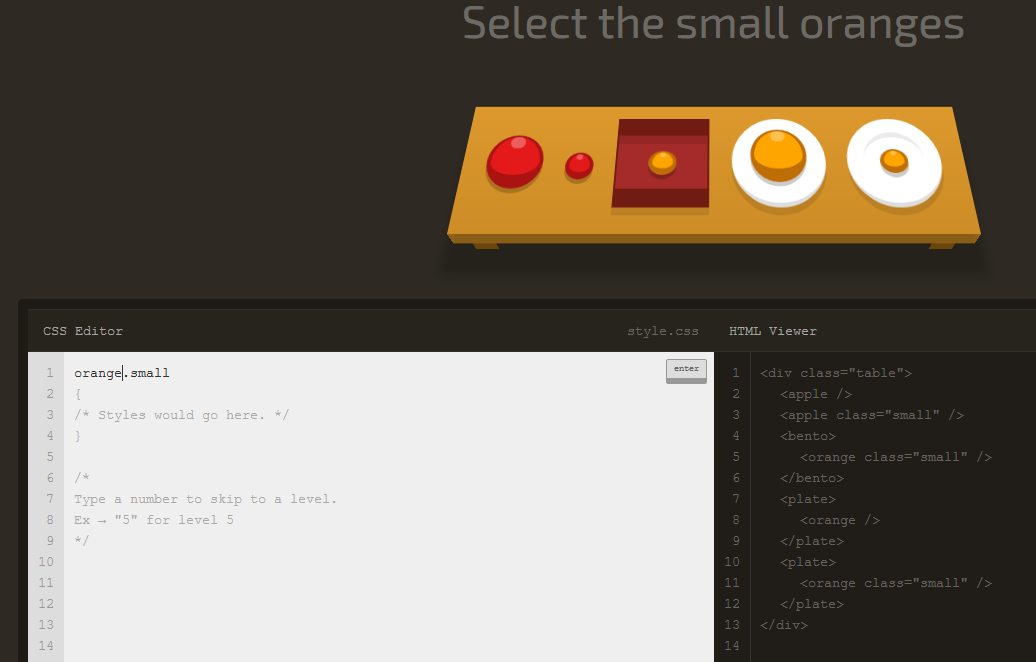
**A.className**

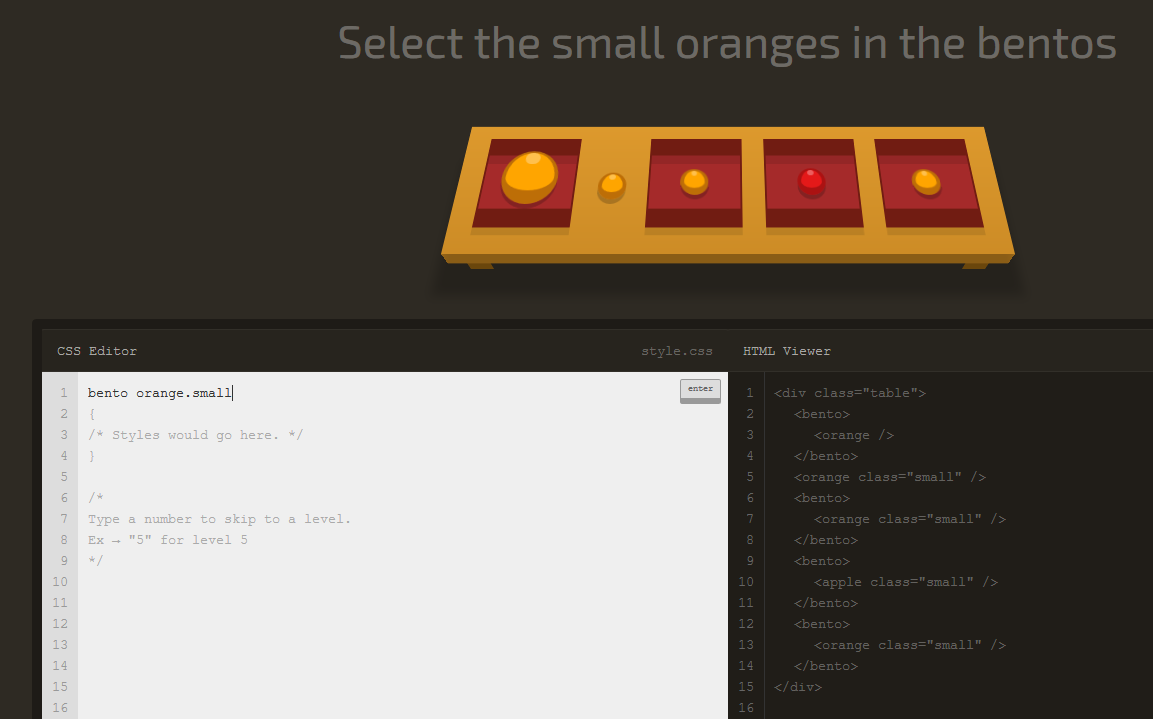
You can combine the class selector with other selectors, like the type selector.

**Examples**

**ul.important** selects all ul elements that have **class="important"**

**#big.wide** selects all elements with **id="big"** that also have **class="wide"**





**Comma Combinator**

**Combine, selectors, with... commas!**

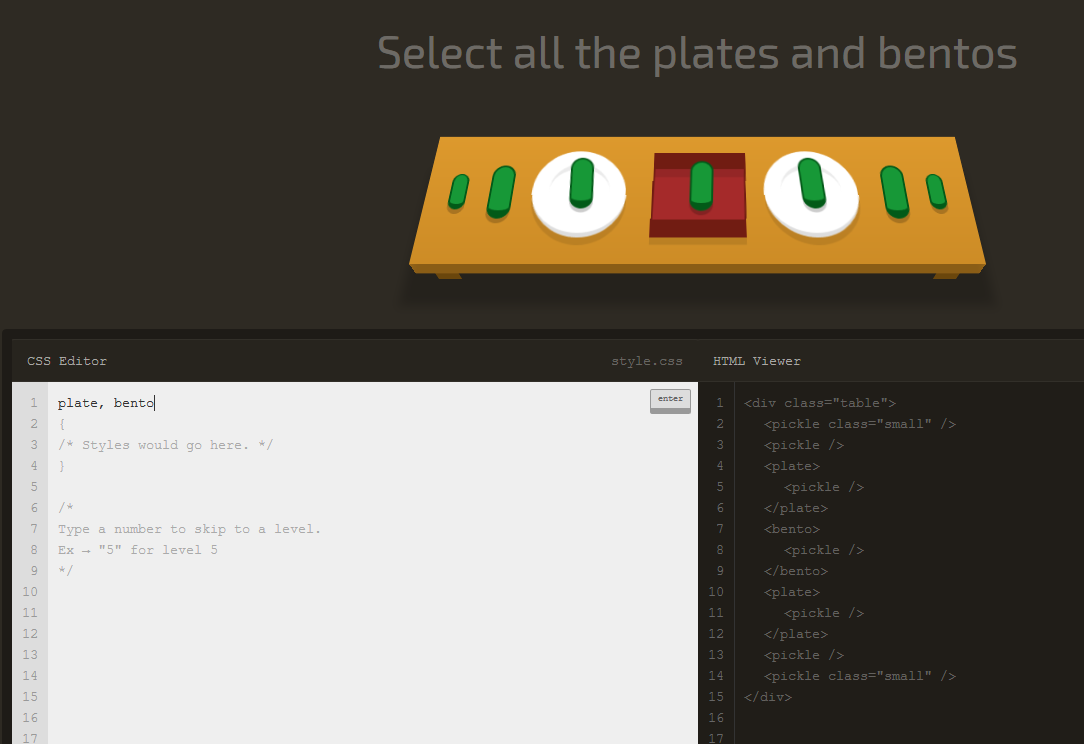
**A, B**

Thanks to Shatner technology, this selects all **A** and **B** elements. You can combine any selectors this way, and you can specify more than two.

**Examples**

**p, .fun** selects all p elements as well as all elements with **class="fun"**

**a, p, div** selects all a, p and div elements



**The Universal Selector**

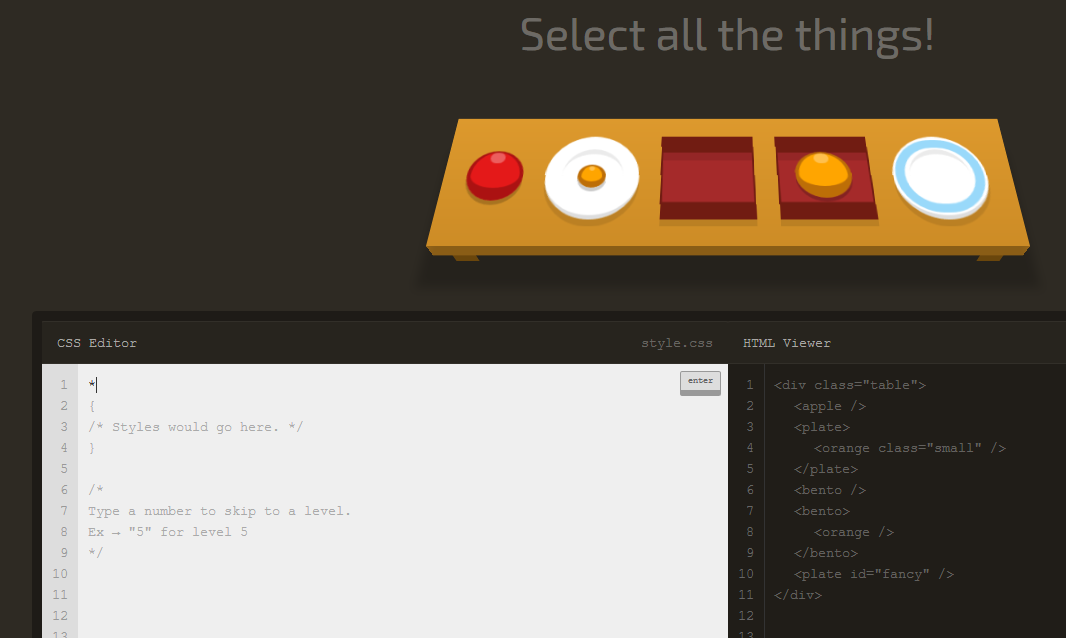
**You can select everything!**

**\***

You can select all elements with the universal selector!

**Examples**

**p \*** selects any element inside all p elements.



**Combine the Universal Selector**

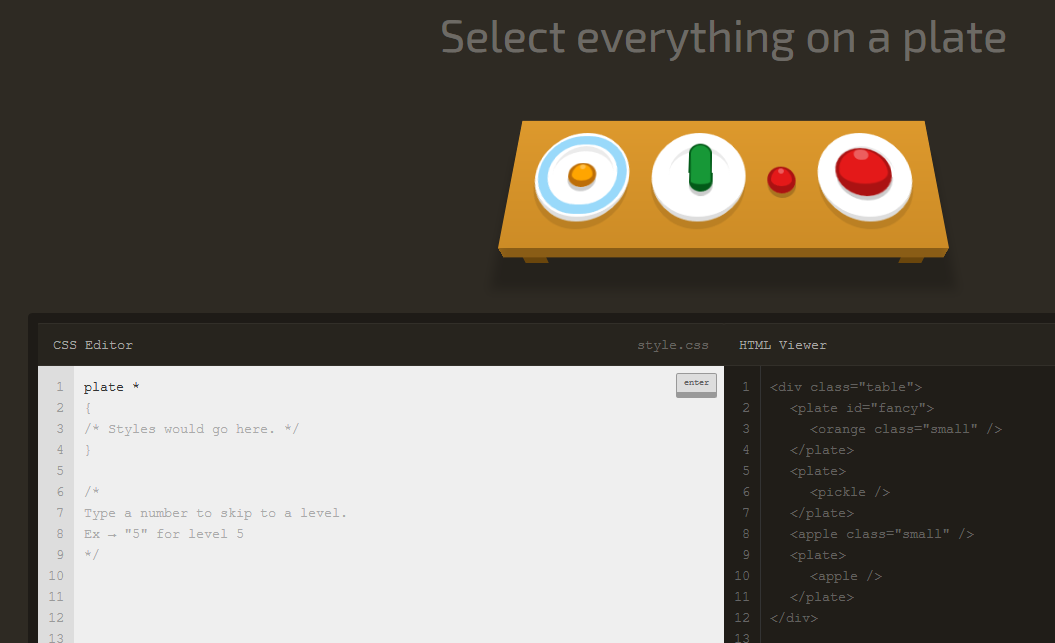
**A  \***

This selects all elements inside of **A**.

**Examples**

**p \*** selects every element inside all p elements.

**ul.fancy \*** selects every element inside all ul class="fancy" elements.



**Adjacent Sibling Selector**

**Select an element that directly follows another element**

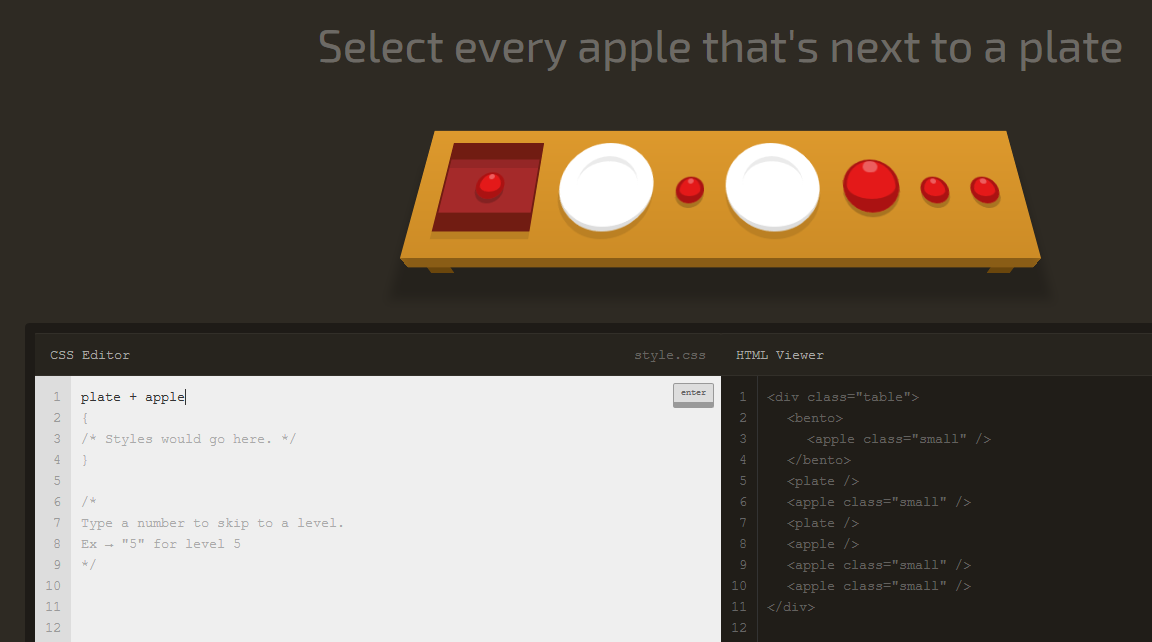
**A + B**

This selects all **B** elements that directly follow **A**. Elements that follow one another are called siblings. They're on the same level, or depth.   
  
In the HTML markup for this level, elements that have the same indentation are siblings.

**Examples**

**p + .intro** selects every element with **class="intro"** that directly follows a p

**div + a** selects every a element that directly follows a div



### General Sibling Selector

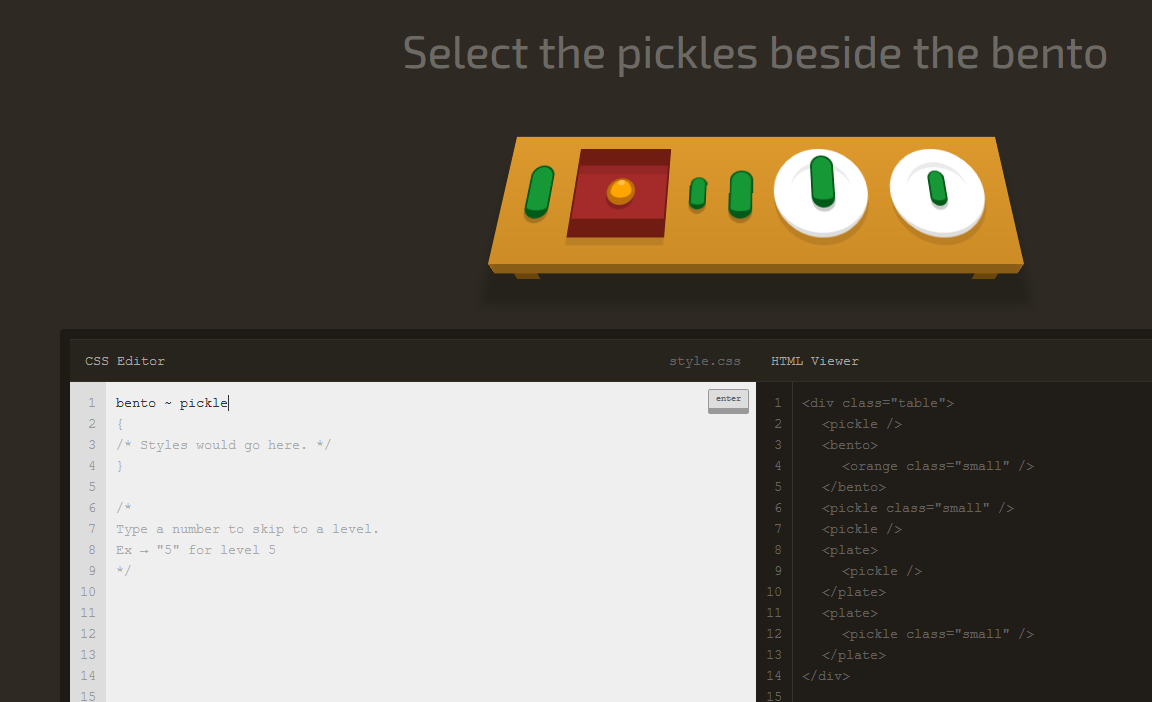
## Select elements that follows another element

### A ~ B

You can select all siblings of an element that follow it. This is like the Adjacent Selector (A + B) except it gets all of the following elements instead of one.

#### Examples

**A ~ B** selects all **B** that follow a **A**



### Child Selector

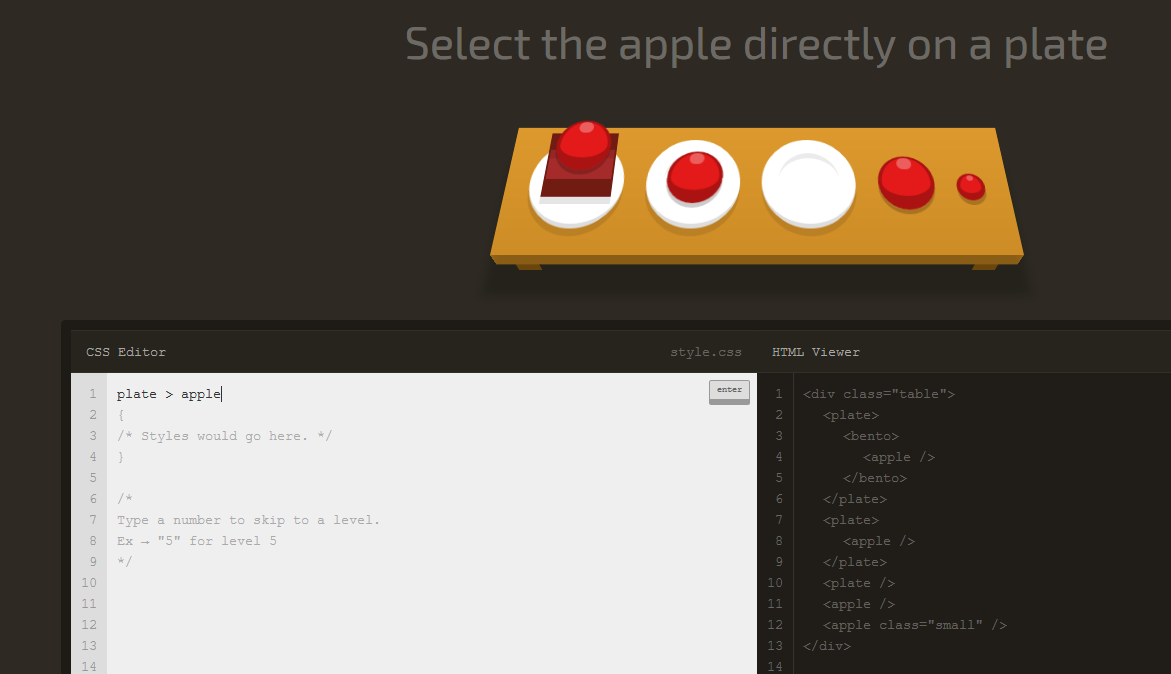
## Select direct children of an element

### A > B

You can select elements that are direct children of other elements. A child element is any element that is nested directly in another element.   
  
Elements that are nested deeper than that are called descendant elements.

#### Examples

**A > B** selects all **B** that are a direct children **A**



### First Child Pseudo-selector

## Select a first child element inside of another element

### :first-child

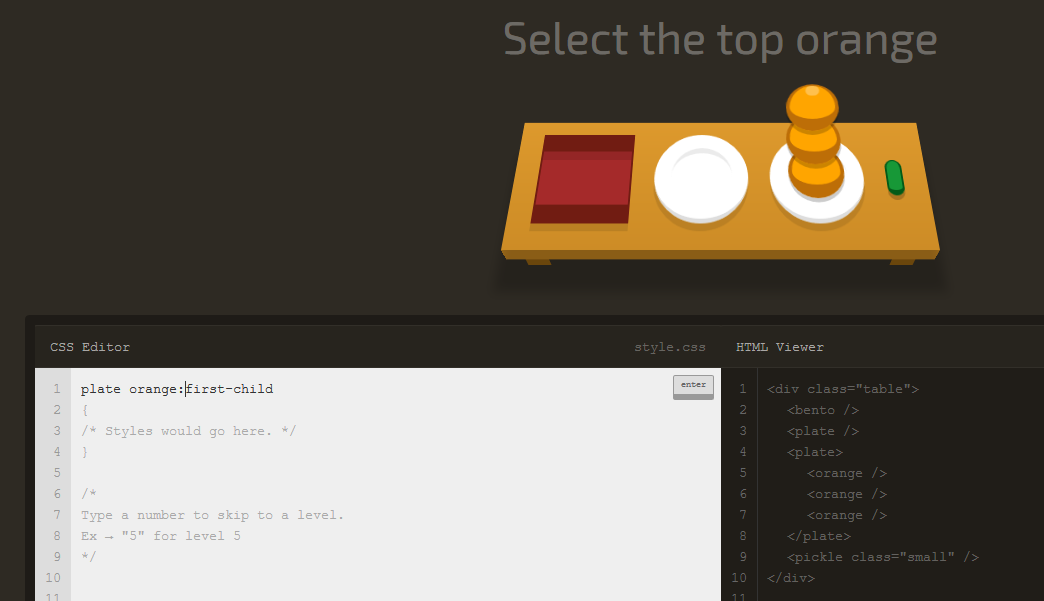
You can select the first child element. A child element is any element that is directly nested in another element. You can combine this pseudo-selector with other selectors.

#### Examples

**:first-child** selects all first child elements.

**p:first-child** selects all first child p elements.

**div p:first-child** selects all first child p elements that are in a div.



### Only Child Pseudo-selector

## Select an element that are the only element inside of another one.

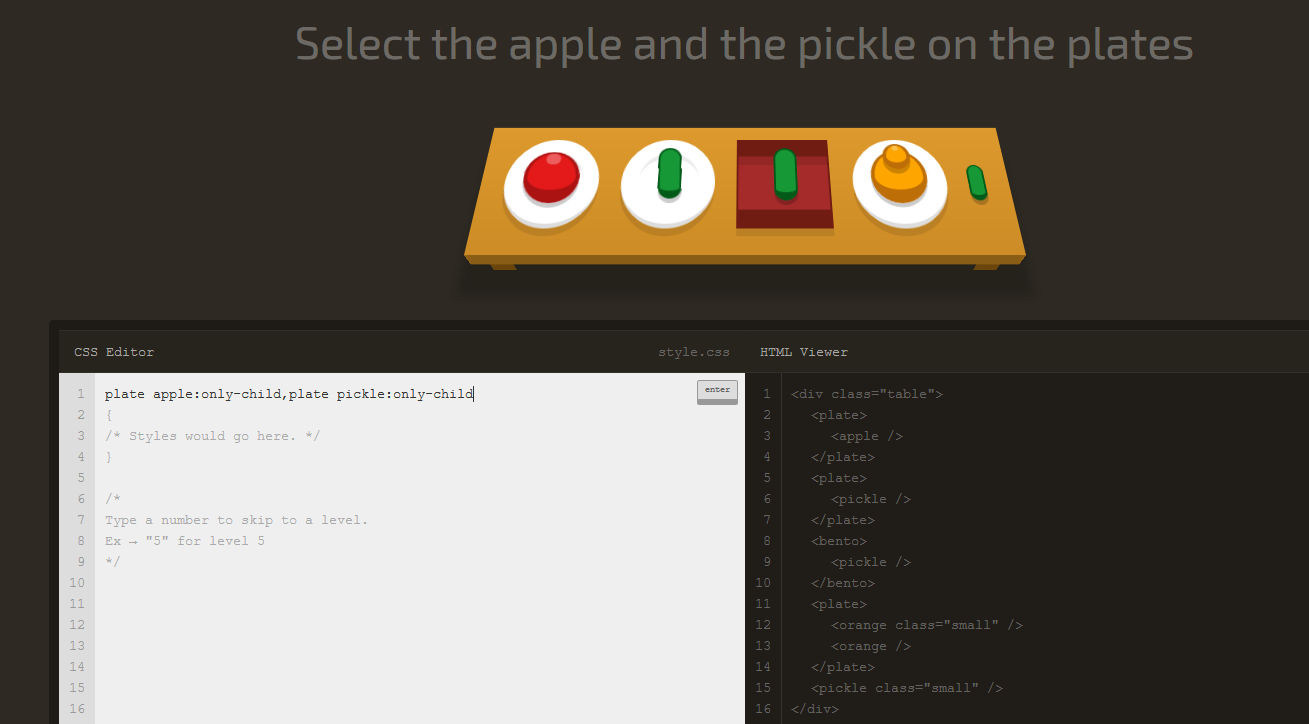
### :only-child

You can select any element that is the only element inside of another one.

#### Examples

**span:only-child** selects the span elements that are the only child of some other element.

**ul li:only-child** selects the only li element that are in a ul.



### Last Child Pseudo-selector

## Select the last element inside of another element

### :last-child

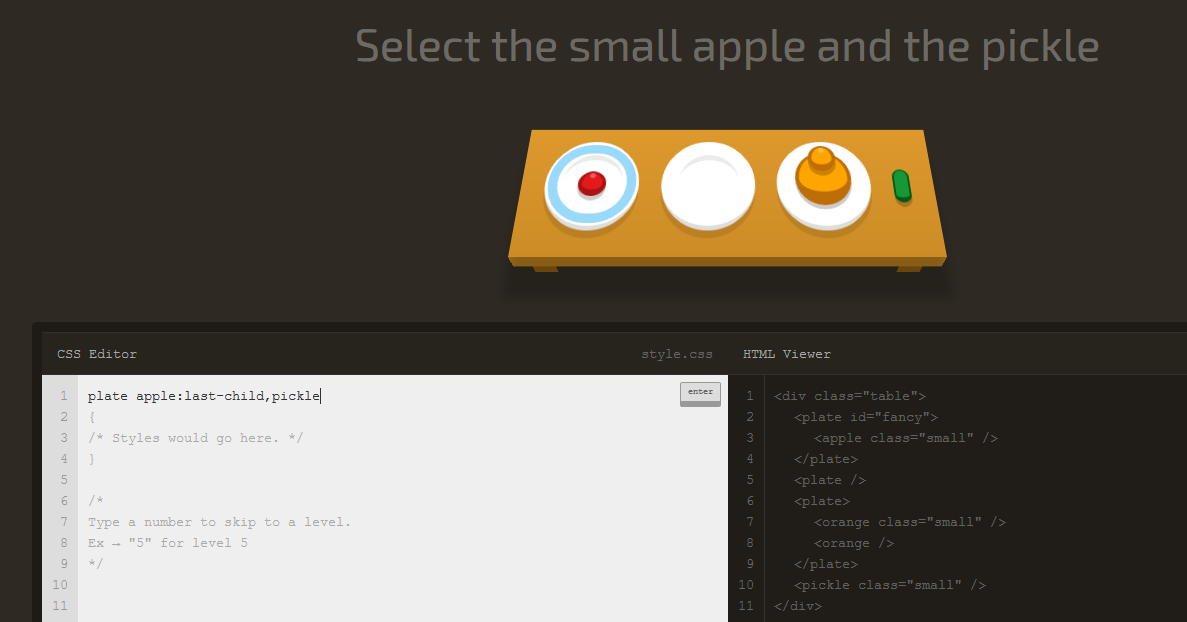
You can use this selector to select an element that is the last child element inside of another element.   
  
Pro Tip → In cases where there is only one element, that element counts as the first-child, only-child and last-child!

#### Examples

**:last-child** selects all last-child elements.

**span:last-child** selects all last-child span elements.

**ul li:last-child** selects the last li elements inside of any ul.

  
apple: last-child, pickle: last-child

### Nth Child Pseudo-selector

## Select an element by its order in another element

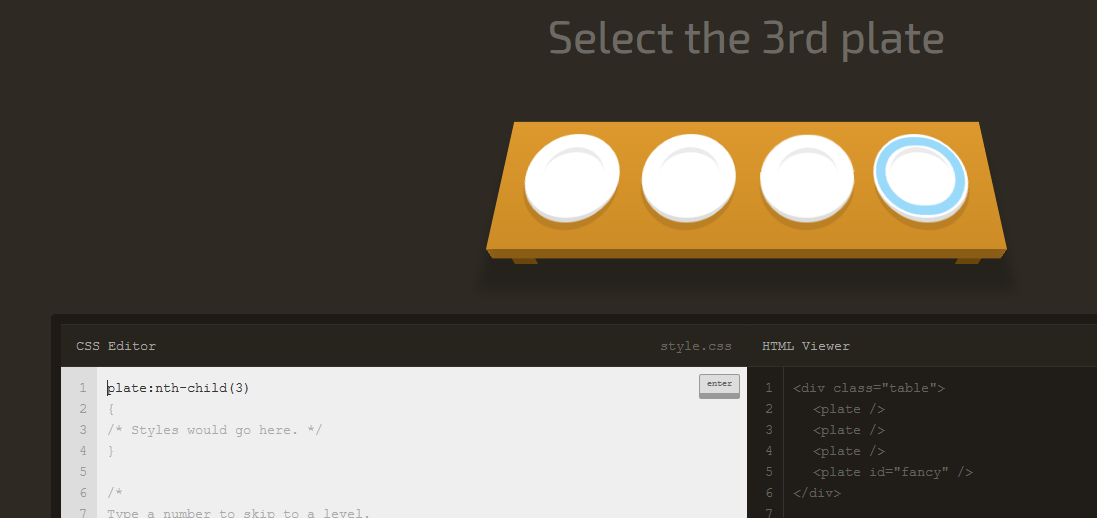
### :nth-child(A)

Selects the **nth** (Ex: 1st, 3rd, 12th etc.) child element in another element.

#### Examples

**:nth-child(8)** selects every element that is the 8th child of another element.

**div p:nth-child(2)** selects the second **p** in every **div**



### Nth Last Child Selector

## Select an element by its order in another element, counting from the back

### :nth-last-child(A)

Selects the children from the bottom of the parent. This is like nth-child, but counting from the back!

#### Examples

**:nth-last-child(2)** selects all second-to-last child elements.



### First of Type Selector

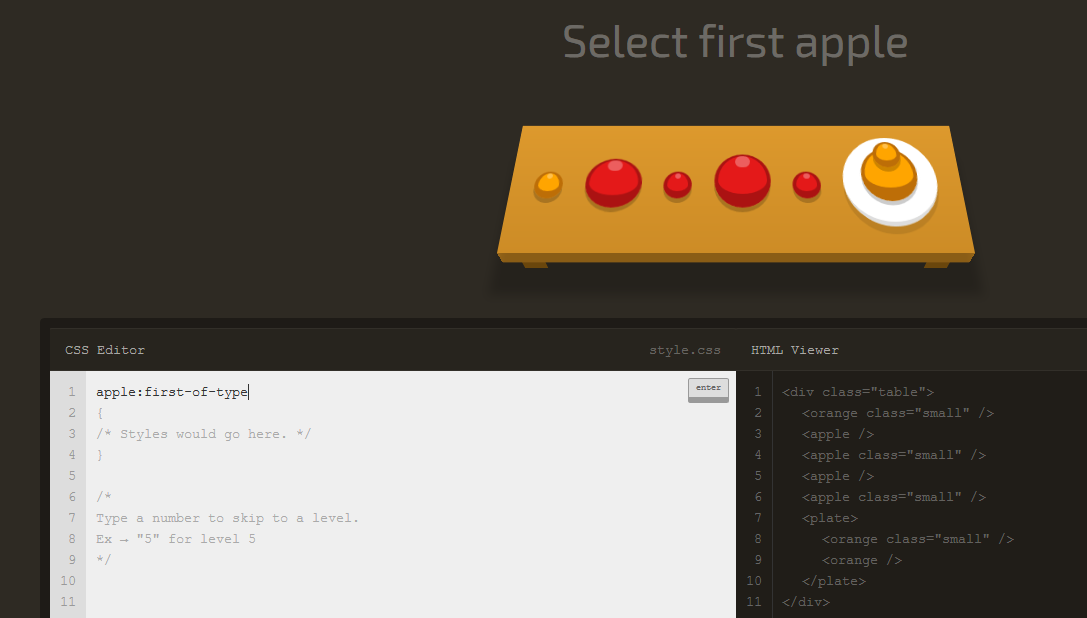
## Select the first element of a specific type

### :first-of-type

Selects the first element of that type within another element.

#### Examples

**span:first-of-type** selects the first span in any element.



### Nth of Type Selector

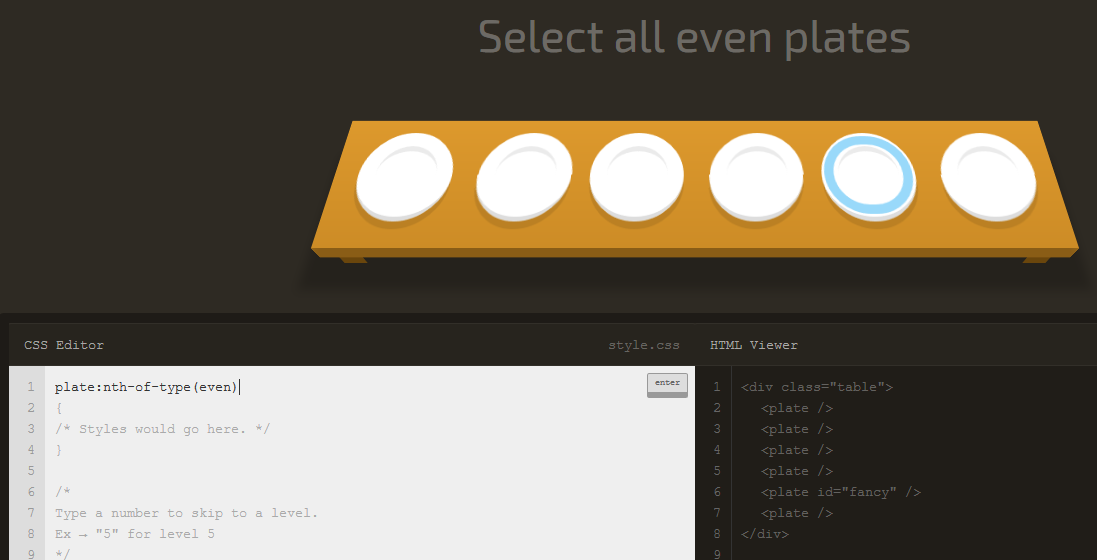
### :nth-of-type(A)

Selects a specific element based on its type and order in another element - or even or odd instances of that element.

#### Examples

**div:nth-of-type(2)** selects the second instance of a div.

**.example:nth-of-type(odd)** selects all odd instances of a the example class.



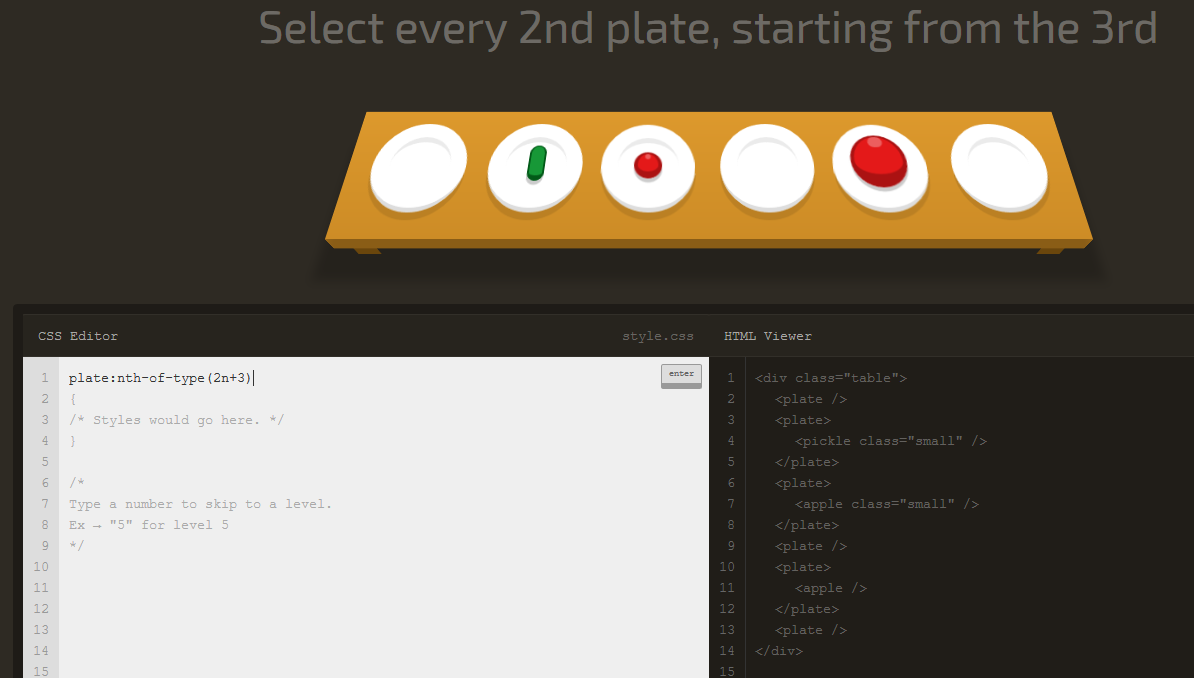
### Nth-of-type Selector with Formula

### :nth-of-type(An+B)

The nth-of-type formula selects every nth element, starting the count at a specific instance of that element.

#### Examples

**span:nth-of-type(6n+2)** selects every 6th instance of a span, starting from (and including) the second instance.



### Only of Type Selector

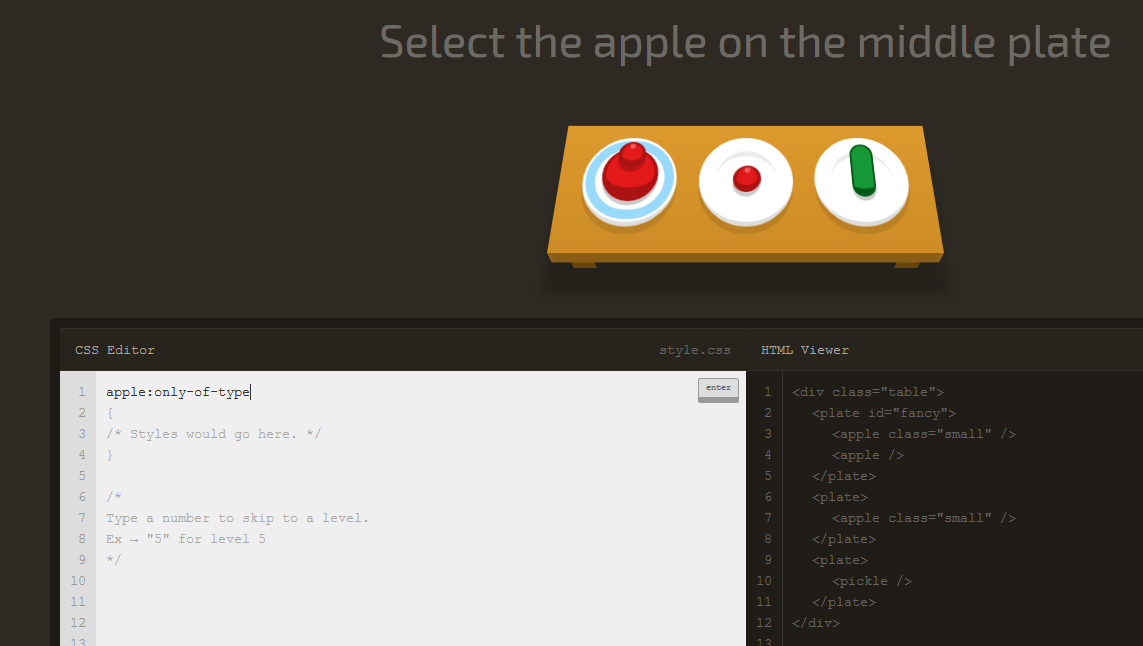
## Select elements that are the only ones of their type within of their parent element

### :only-of-type

Selects the only element of its type within another element.

#### Examples

**p span:only-of-type** selects a span within any p if it is the only span in there.

  
apple.small:only-of-type

### Last of Type Selector

## Select the last element of a specific type

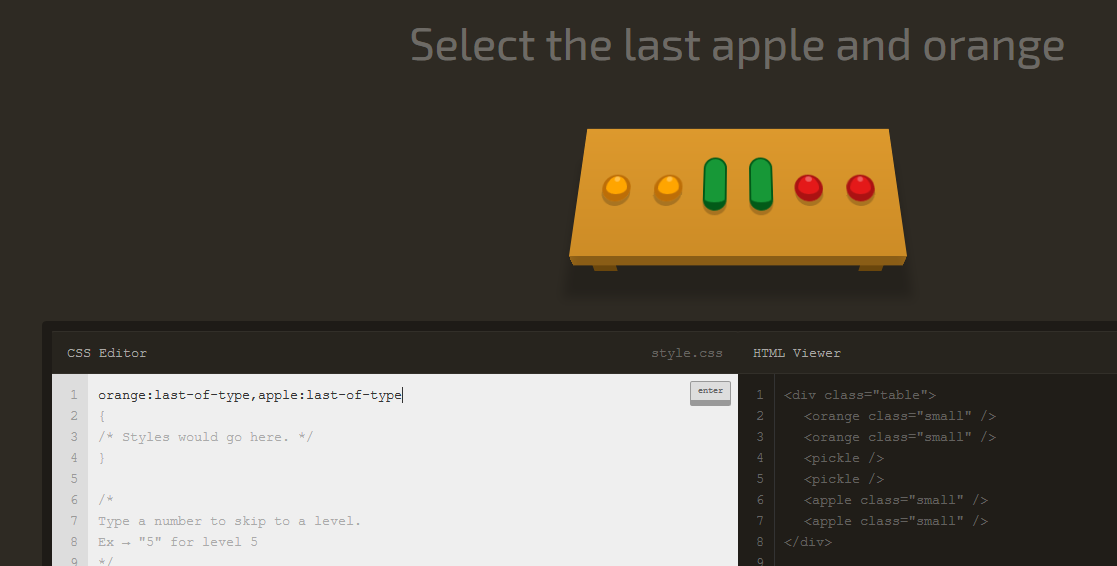
### :last-of-type

Selects each last element of that type within another element. Remember type refers the kind of tag, so p and span are different types.   
  
I wonder if this is how the last dinosaur was selected before it went extinct.

#### Examples

**div:last-of-type** selects the last div in every element.

**p span:last-of-type** selects the last span in every p.



### Empty Selector

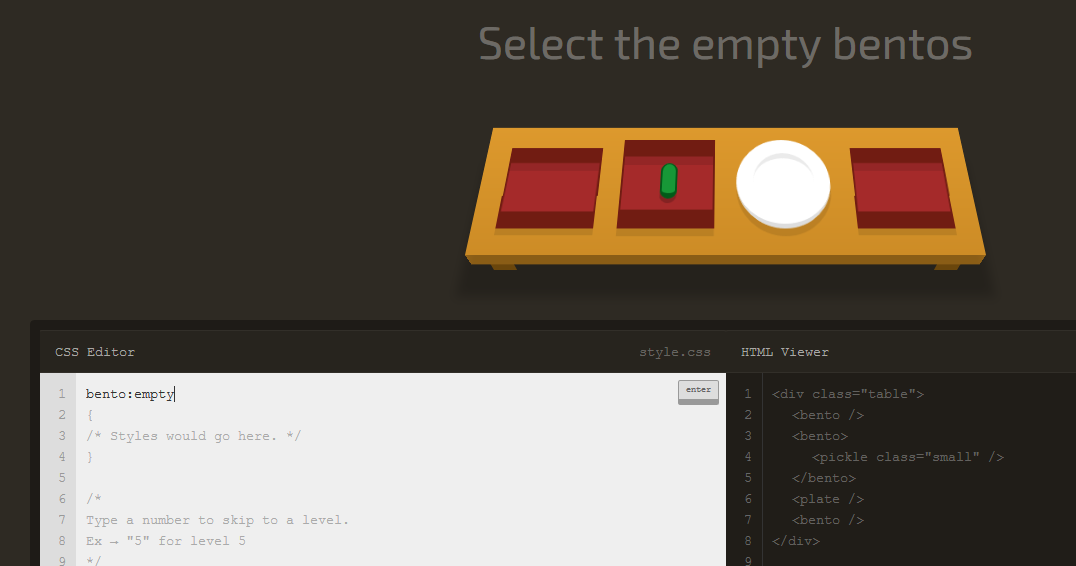
## Select elements that don't have children

### :empty

Selects elements that don't have any other elements inside of them.

#### Examples

**div:empty** selects all empty div elements.



### Negation Pseudo-class

## Select all elements that don't match the negation selector

### :not(X)

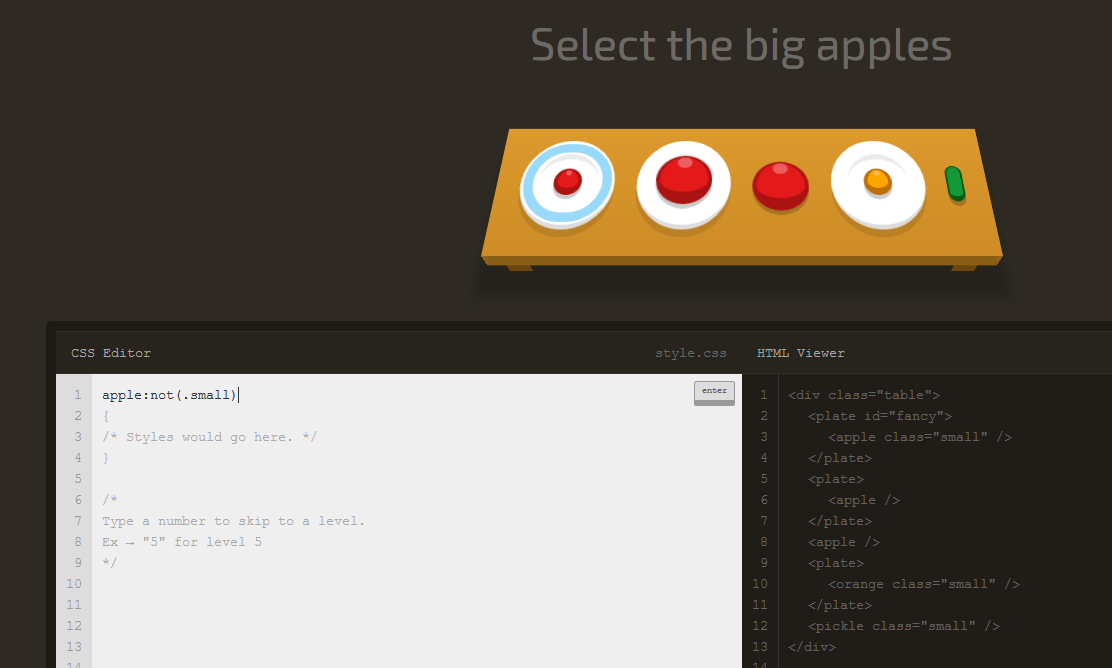
You can use this to select all elements that do not match selector **"X"**.

#### Examples

**:not(#fancy)** selects all elements that do not have **id="fancy"**.

**div:not(:first-child)** selects every div that is not a first child.

**:not(.big, .medium)** selects all elements that do not have **class="big"** or **class="medium"**.



### Attribute Selector

## Select all elements that have a specific attribute

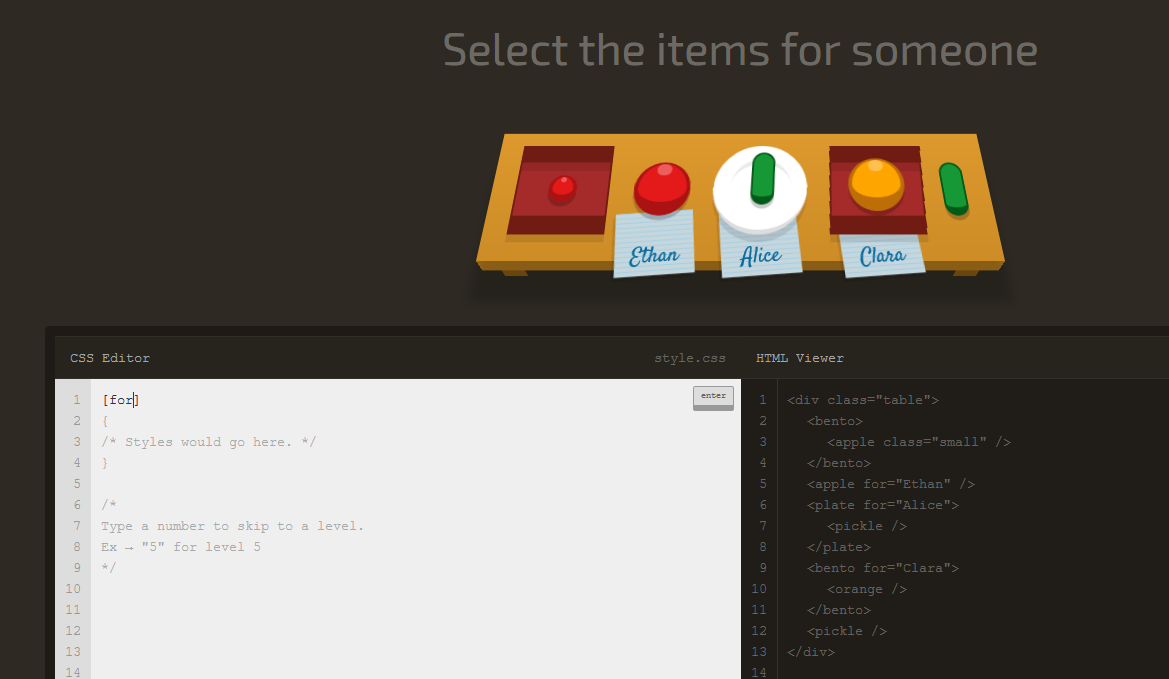
### [attribute]

Attributes appear inside the opening tag of an element, like this: span attribute="value". An attribute does not always have a value, it can be blank!

#### Examples

**a[href]** selects all a elements that have a **href="anything"** attribute.

**[type]** selects all elements that have a **type="anything"**. attribute



### Attribute Selector

## Select all elements that have a specific attribute

### A[attribute]

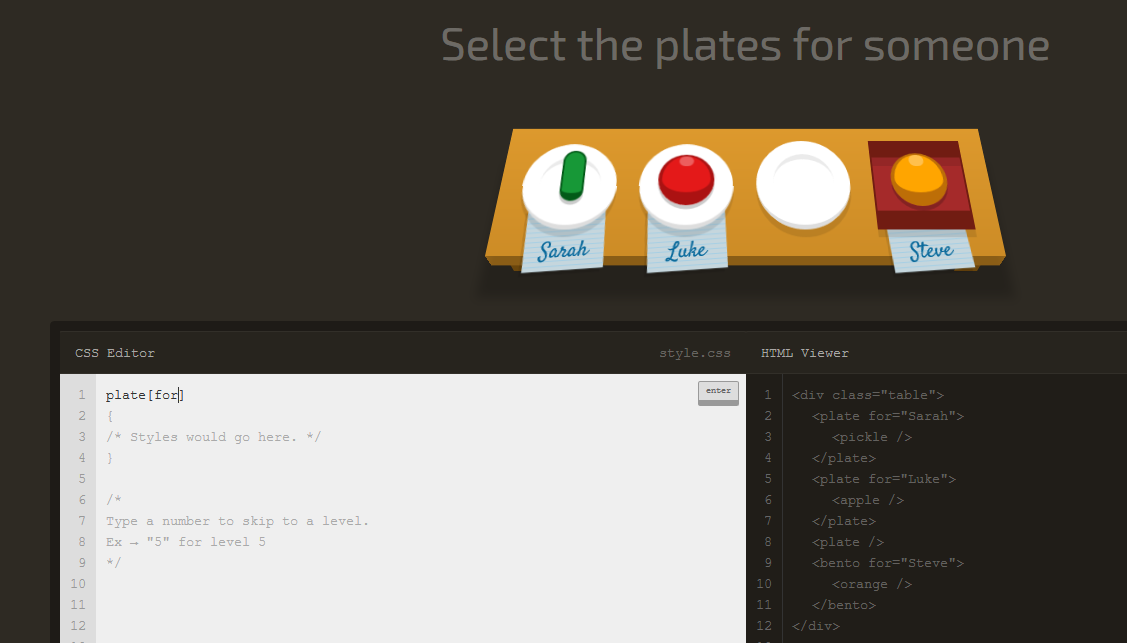
Combine the attribute selector with another selector (like the tag name selector) by adding it to the end.

#### Examples

**[value]** selects all elements that have a **value="anything"** attribute.

**a[href]** selects all a elements that have a **href="anything"** attribute.

**input[disabled]** selects all input elements with the **disabled** attribute



### Attribute Value Selector

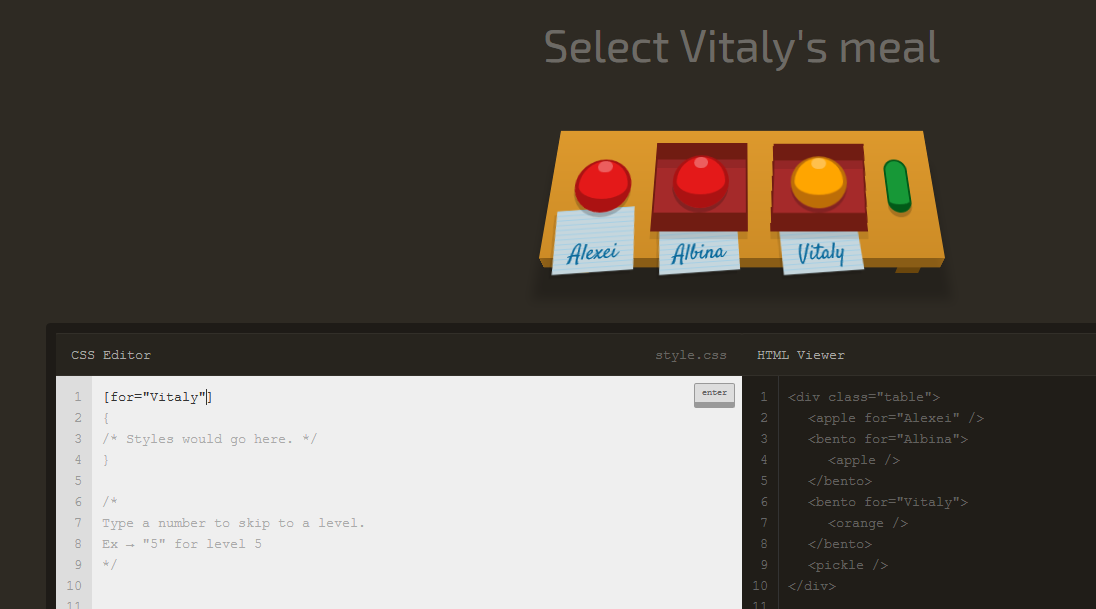
## Select all elements that have a specific attribute value

### [attribute="value"]

Attribute selectors are case sensitive, each character must match exactly.

#### Examples

**input[type="checkbox"]** selects all checkbox input elements.



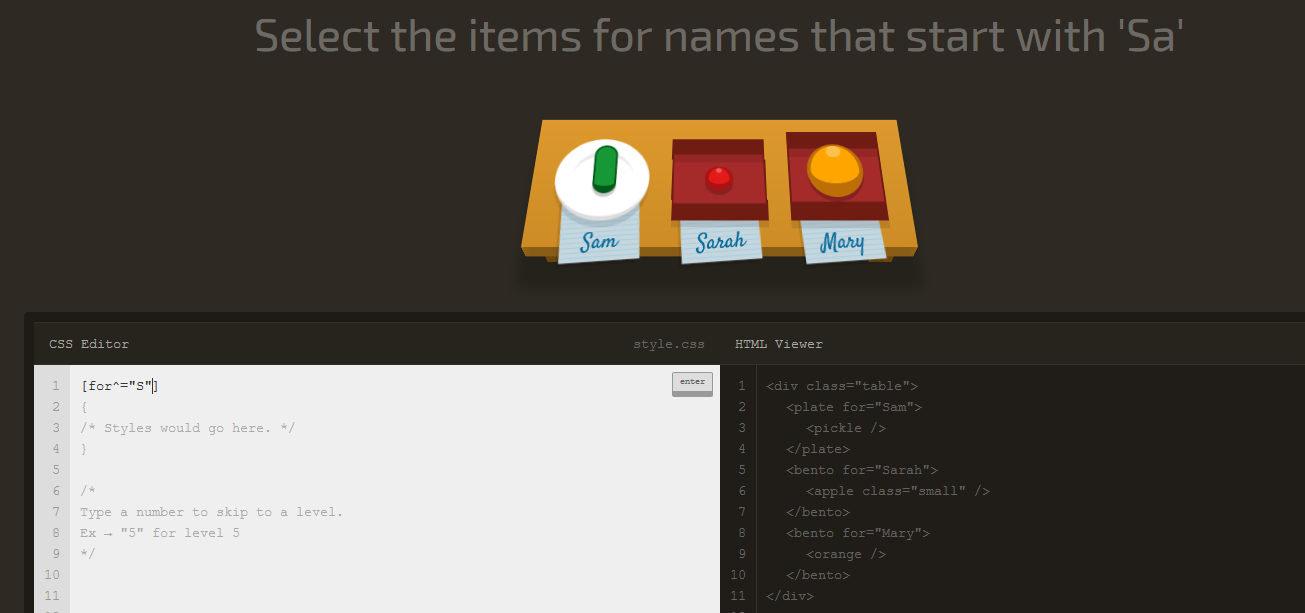
### Attribute Starts With Selector

## Select all elements with an attribute value that starts with specific characters

### [attribute^="value"]

#### Examples

**.toy[category^="Swim"]** selects elements with class **toy** and either **category="Swimwear** or **category="Swimming"**.



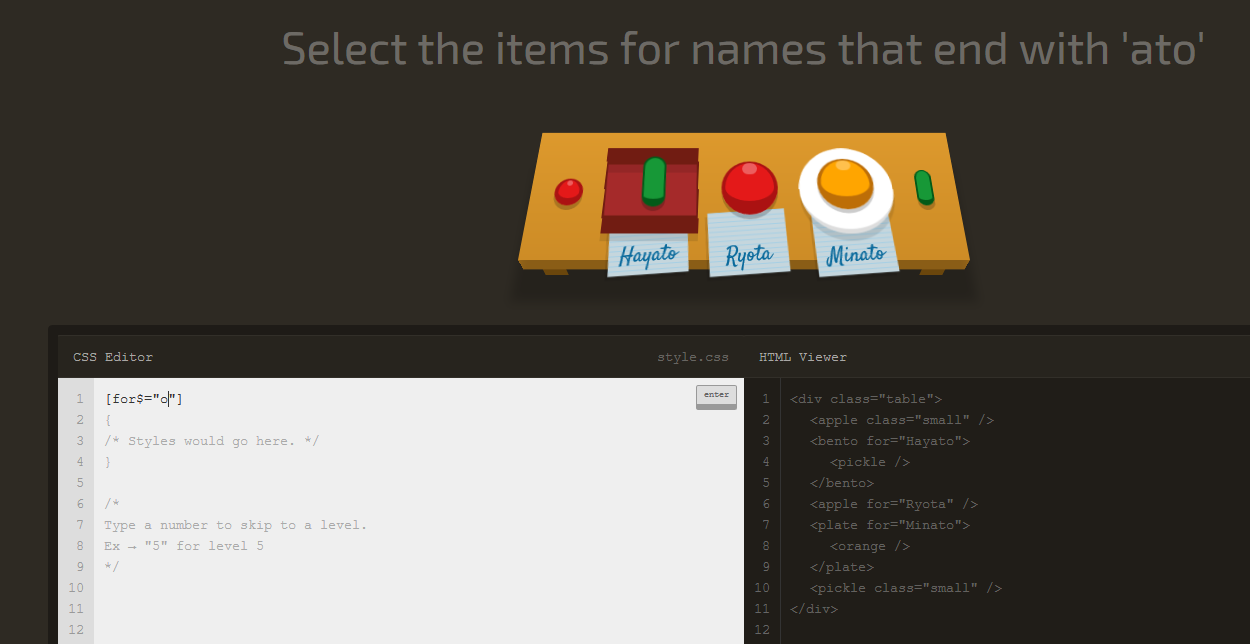
### Attribute Ends With Selector

## Select all elements with an attribute value that ends with specific characters

### [attribute$="value"]

#### Examples

**img[src$=".jpg"]** selects all images display a **.jpg** image.



### Attribute Wildcard Selector

## Select all elements with an attribute value that contains specific characters anywhere

### [attribute\*="value"]

A useful selector if you can identify a common pattern in things like **class**, **href** or **src** attributes.

#### Examples

**img[src\*="/thumbnails/"]** selects all image elements that show images from the "thumbnails" folder.

**[class\*="heading"]** selects all elements with "heading" in their class, like **class="main-heading"** and **class="sub-heading"**

