Selectors in CSS 3

CSS (Cascading Style Sheets) selectors are used to select the HTML elements you want to style. CSS3, the latest version of CSS, introduces a variety of new selectors that make it easier to select and style elements with greater precision. Here's a detailed look at the different types of selectors available in CSS3:

Basic Selectors

Universal Selector (*)

Selects all elements on the page.

```
* {
  margin: 0;
  padding: 0;
}
```

Type Selector

Selects all elements of a given type.

```
p {
  color: blue;
}
```

Class Selector (.)

Selects all elements with a given class.

```
.example {
  font-size: 20px;
}
```

ID Selector (#)

Selects a single element with a given ID.

```
#unique-element {
  background-color: yellow;
}
```

Attribute Selector

Selects elements based on the presence or value of their attributes.

```
[title] {
  border-bottom: 1px dotted;
```

```
}
input[type="text"] {
  width: 200px;
}
```

Combinator Selectors

Descendant Selector (space)

Selects all elements that are descendants of a specified element.

```
div p {
  color: red;
}
```

Child Selector (>)

Selects all elements that are direct children of a specified element.

```
ul > li {
  list-style-type: none;
}
```

Adjacent Sibling Selector (+)

Selects an element that is immediately preceded by a specified element.

```
h1 + p {
  margin-top: 0;
}
```

General Sibling Selector (~)

Selects all elements that are siblings of a specified element.

```
h1 ~ p {
  color: green;
}
```

Pseudo-Class Selectors

Pseudo-classes are used to define a special state of an element.

Structural Pseudo-Classes

:first-child, :last-child, :nth-child(n), :nth-last-child(n), etc.

```
p:first-child {
  font-weight: bold;
}
```

```
p:nth-child(2) {
  color: blue;
}
```

UI Element States

:hover, :focus, :active, etc.

```
a:hover {
  color: red;
}
input:focus {
  border-color: blue;
}
```

Other Pseudo-Classes

:not(selector): Selects elements that do not match a given selector.

```
p:not(.intro) {
  color: gray;
}
```

Pseudo-Element Selectors

Pseudo-elements are used to style specified parts of an element.

::before and ::after

Insert content before or after the content of an element.

```
p::before {
  content: "Note: ";
  font-weight: bold;
}
```

::first-line and ::first-letter

Style the first line or first letter of an element.

```
p::first-line {
  font-variant: small-caps;
}
p::first-letter {
  font-size: 200%;
  color: red;
}
```

Attribute Selectors

CSS3 expanded attribute selectors to include:

[attribute]: Elements with the specified attribute.
[attribute=value]: Elements with the attribute equal to the specified value.
[attribute^=value]: Elements with the attribute value starting with the specified value.
[attribute$=value]: Elements with the attribute value ending with the specified value.
[attribute*=value]: Elements with the attribute value containing the specified value.
[attribute~=value]: Elements with the attribute value containing a specified word.
[attribute|=value]: Elements with the attribute value either equal to the specified value or starting with the specified value followed by a hyphen.

Grouping Selectors

Grouping

Multiple selectors can be grouped to apply the same styles.

```
h1, h2, h3 {
  font-family: Arial, sans-serif;
}
```

Specificity and Inheritance

Specificity determines which styles are applied when multiple rules match the same element. It is calculated based on the types of selectors used:
1. Inline styles have the highest specificity.
2. ID selectors are next.
3. Class, attribute, and pseudo-class selectors follow.
4. Type selectors and pseudo-elements have the lowest specificity.

In conclusion, CSS3 selectors offer powerful and flexible ways to target elements in a document for styling, enabling more precise and maintainable CSS code. By understanding and utilizing these selectors effectively, developers can create more sophisticated and responsive web designs.