

Theory: List style

🕒 11 minutes 0 / 4 problems solved

Skip this topic

Start practicing

63 users solved this topic. Latest completion was 1 day ago.

We all use some kind of lists in our daily lives, such as shopping lists, todo lists, etc. In Web development, we work with a lot of lists, and what’s better than to have a beautiful list on our website? In this topic we will learn how to style our lists and make them more interesting.

§1. Syntax

The `list-style` is a shorthand property that allows us to set all the three list-style properties at once.

- `list-style-type`: Sets the marker type. Default value is `disc`.
- `list-style-position`: Sets the position of the marker. Default value is `outside`.
- `list-style-image`: Sets an image as a marker. Default value is `none`.

We can use this property with the parent selector `ul` and `ol` or with the child selector `li`.

If we use it with the parent selector, the style will be applied to all of the children. If we use it with the child selector, the style will be applied only to that specific child.

When using `ol` the default value for `type` will be `decimal`.

```
1  /* type | position | image*/
2  ul {
3      list-style: type position image;
4  }
```

For example, the `CSS` code below:

```
1  ul {
2      list-style-type: circle;
3      list-style-position: inside;
4      list-style-image: none;
5  }
```

Would be the same as the following `CSS` code:

```
1  ul {
2      list-style: circle inside none;
3  }
```

§2. Values of type

Let’s start with the `type`. There are a lot of styles we can use, we’ll list the most common of them below. For a more complete list you can check the [MDN](#) website.

- `circle`
- `decimal`
- `georgian`
- `upper-roman`
- `lower-greek`

Current topic:

[List style](#) ...

Topic depends on:

- ×

Lists

Stage 3

...
- ×

Comments

Stage 2

...

Table of contents:

[↑ List style](#)

[§1. Syntax](#)

[§2. Values of type](#)

[§3. Values of position](#)

[§4. Values of image](#)

[§5. Conclusion](#)

[Feedback & Comments](#)

```
1 <p>Top 5 programming languages 2020</p>
2 <ul>
3   <li>JavaScript</li>
4   <li>Python</li>
5   <li>Java</li>
6   <li>Go</li>
7   <li>C++</li>
8 </ul>
```

If we use `georgian`, the list would look like this:

```
1 ul {
2   list-style-type: georgian;
3 }
```

Top 5 programming languages 2020

- ⴁ. JavaScript
- ⴂ. Python
- ⴃ. Java
- ⴄ. Go
- ⴅ. C++

And if we use `upper-roman`, the list would look like this:

```
1 ul {
2   list-style-type: upper-roman;
3 }
```

Top 5 programming languages 2020

- I. JavaScript
- II. Python
- III. Java
- IV. Go
- V. C++

§3. Values of position

With `position` we can set the place of the marker on the list. It accepts two values: `outside` and `inside`.

To get a better idea of how this works we will add a border to each `li` element.

```
1 ul {
2   border: 1px solid black;
3   list-style-position: outside;
4 }
```

Top 5 programming languages 2020

- JavaScript
- Python
- Java
- Go
- C++

Now let's see what our list will look like if we set the `position` to `inside`.

```
1 ul {
2   border: 1px solid black;
3   list-style-position: inside;
4 }
```

Top 5 programming languages 2020

- JavaScript
- Python
- Java
- Go
- C++

As you can see the marker is now inside the border along with the text.

§4. Values of image

We can also use an image as a marker.

When we use an image as a marker it will replace the `type` if there is any.

```
1 | ul {
2 |   list-style-
image: url('https://findicons.com/files/icons/573/must_have/16/check.png');
3 | }
```

Top 5 programming languages 2020

- ✔ JavaScript
- ✔ Python
- ✔ Java
- ✔ Go
- ✔ C++

Please note, that if we use very large images as markers, we may get an undesirable result.

Top 5 programming languages 2020

- ✔ JavaScript
- ✔ Python

Now we can combine all properties in one: as we are using `image`, we do not need to set the `type`.

```
1 | ul {
2 |   list-
style: inside url('https://findicons.com/files/icons/573/must_have/16/check.png');
3 | }
```

§5. Conclusion

With the `list-style` property we can make our lists look more beautiful and interesting. We can use different `types`, change their `position`, and we can also use an `image` as a marker. We use the `list-style` property to combine all the changes in one style. Remember that it is not possible to use type and image at the same time: an `image` will replace the `type`. Now let's practice with some tasks!

Report a typo

6 users liked this theory. 0 didn't like it. What about you?



Start practicing

[Comments \(1\)](#)

[Hints \(0\)](#)

[Useful links \(0\)](#)

[Show discussion](#)