

CSS

CSS FUNDAMENTALS

SELECTORS

Attribute



IN A ROCKET

Learn front-end development at rocket speed.

**ATTRIBUTE
PRESENCE
AND VALUE
SELECTORS**

**SUBSTRING
MATCHING
ATTRIBUTE
SELECTORS**

**ATTRIBUTE
PRESENCE
AND VALUE
SELECTORS**

**SUBSTRING
MATCHING
ATTRIBUTE
SELECTORS**

Attribute presence and value selectors

[attribute]

[attribute=value]

[attribute~=value]

[attribute|=value]

ATTRIBUTE SELECTOR

Selects **all elements with a specific attribute**.

Syntax `[attribute] {style properties}`

`[target] {color: green}`

With this code all elements with the `target` attribute are shown in green.

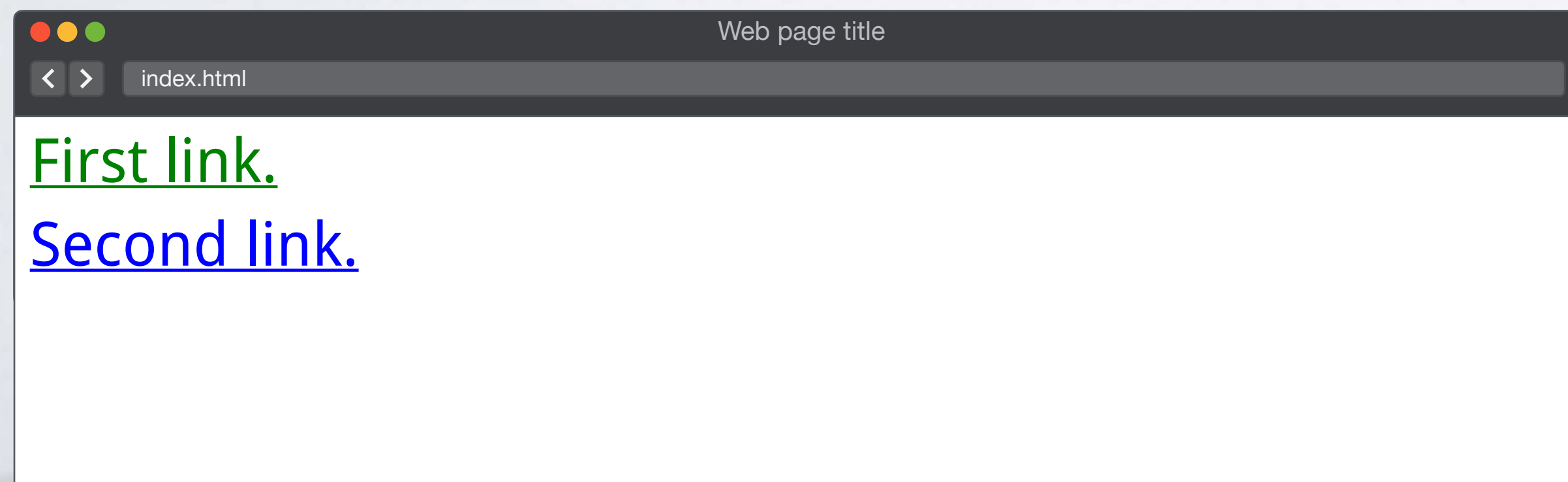
HTML

```
<body>
<a href="#" target="_blank">First link.</a>
<a href="#">Second link.</a>
</body>
```

CSS

```
[target] { color: green; }
```

Browser



Attribute presence and value selectors

[attribute]

[attribute=value]

[attribute~=value]

[attribute|=value]

ATTRIBUTE SELECTOR

Selects all elements with a **specific attribute and value**.

Syntax `[attribute=value] {style properties}`

`[target=_blank] {color: green}`

With this code all elements with the attribute `target` and the value `_blank` are shown in green.

HTML

```
<body>  
<a href="#" target="_blank">First link.</a><br>  
<a href="#" target="_top">Second link.</a><br>  
<a href="http://wikipedia.org">Wikipedia.</a><br>  
  
</body>
```

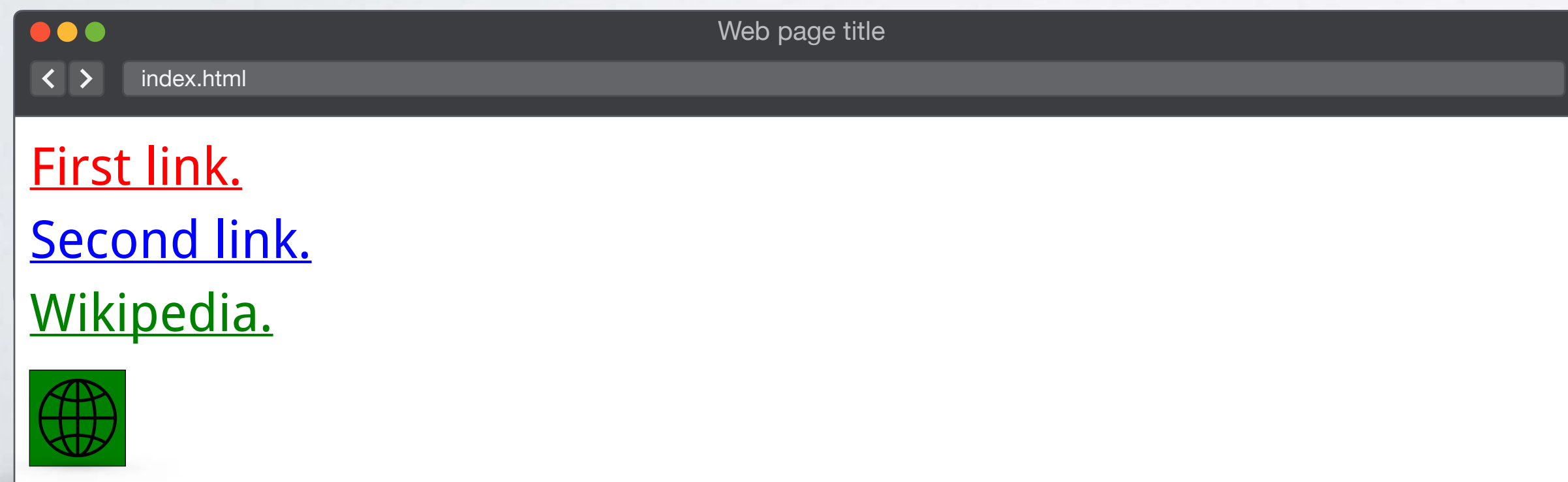
CSS

```
[target=_blank] { color: red; }
```

```
[href=http://wikipedia.org] { color: green; }
```

```
[src=world.png] { background: green; }
```

Browser



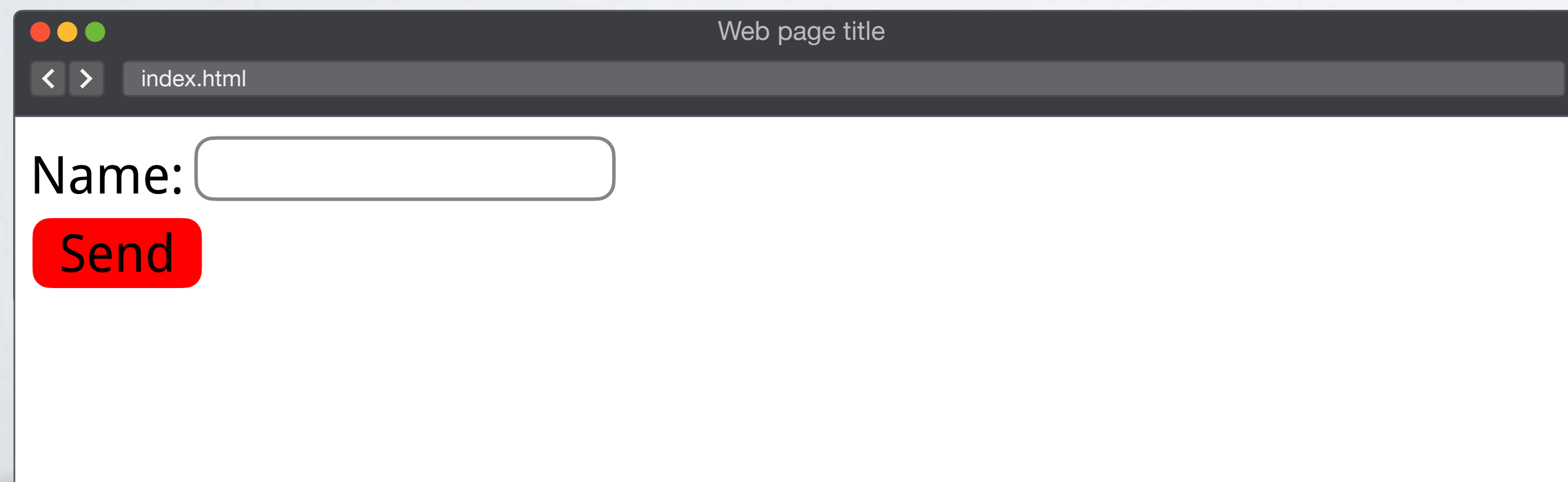
HTML

```
<body>
<form>
Name: <input type="text">
<input type="submit" value="Send">
</form>
</body>
```

CSS

```
[type=submit] { background: red; }
```

Browser



Attribute presence and value selectors

[attribute]

[attribute=value]

[attribute~=value]

[attribute|=value]

ATTRIBUTE SELECTOR

Selects all elements with a **specific attribute**, but only if the value is one of a space-separated list of words.

Syntax `[attribute~=value] {style properties}`

`[data-item~=deal] {color: green}`

With this code all elements with the attribute target and the value deal are shown in green.

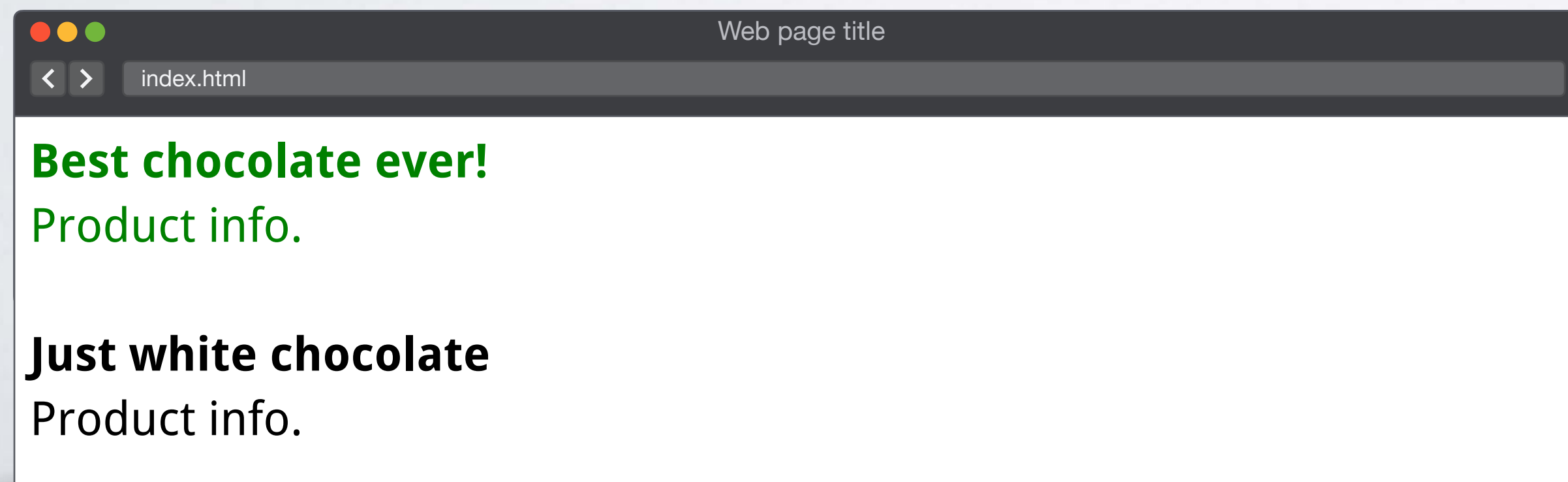
HTML

```
<body>
<article data-item="food deal choco">
<h2>Best chocolate ever!</h2>
<p>Product info.</p>
</article>
<article data-item="food choco">
<h2>Just white chocolate</h2>
<p>Product info.</p>
</article>
</body>
```

CSS

```
[data-item~="deal"] { color: green; }
```

Browser



Attribute presence and value selectors

[attribute]

[attribute=value]

[attribute~=value]

[attribute|=value]

ATTRIBUTE SELECTOR

Selects all elements with a **specific attribute** if their value is **exactly** a particular text or begins with it.

Syntax `[attribute=value] {style properties}`

`[hreflang=en] {color: green}`

With this code all elements with the hreflang attribute and its value beginning with en are shown in green.

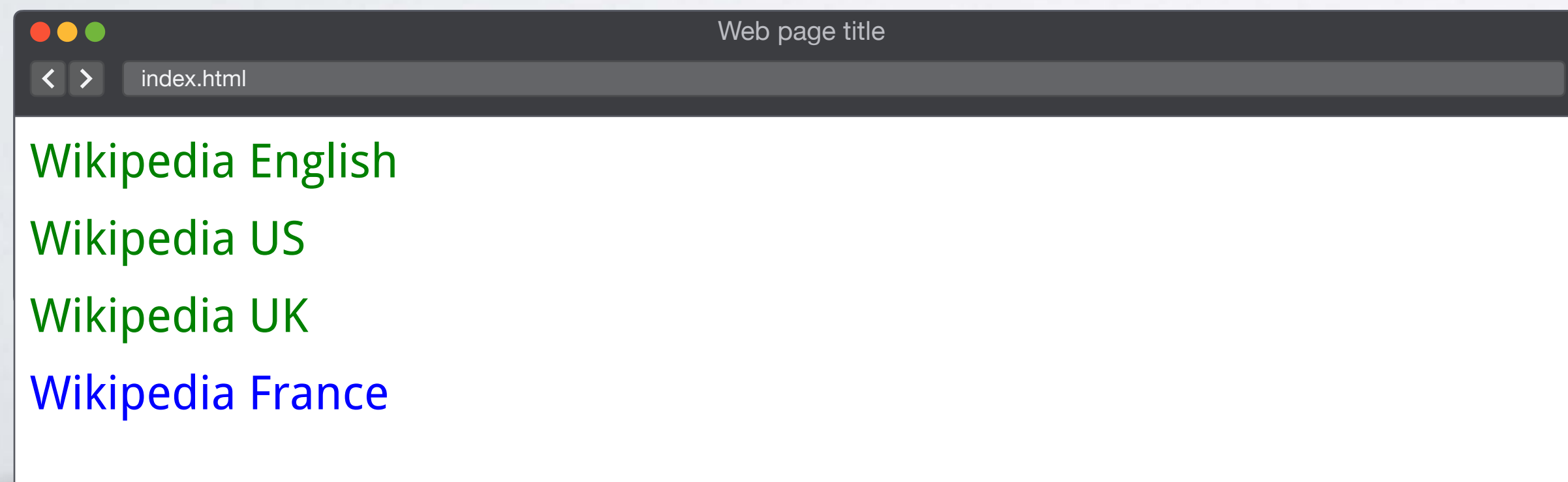
HTML

```
<body>
<a href="#" hreflang="en">Wikipedia English</a><br>
<a href="#" hreflang="en-us">Wikipedia US</a><br>
<a href="#" hreflang="en-gb">Wikipedia UK</a><br>
<a href="#" hreflang="fr">Wikipedia France</a>
</body>
```

CSS

```
[hreflang|=en] { color: green; }
```

Browser



**ATTRIBUTE
PRESENCE
AND VALUE
SELECTORS**

**SUBSTRING
MATCHING
ATTRIBUTE
SELECTORS**

Substring matching attribute selectors

[attribute[^]=value]

[attribute^{*}=value]

[attribute^{\$}=value]

value



Begins with



Contains



Ends with

Substring matching attribute selectors

[attribute^=value]

[attribute=value]*

[attribute\$=value]

ATTRIBUTE SELECTOR

Selects all elements with a **specific attribute** if their value begins with a particular text.

Syntax `[attribute^=value] {style properties}`

`[data-item^=food] {color: green}`

With this code all elements with the data-item attribute and its value beginning with food are shown in green.

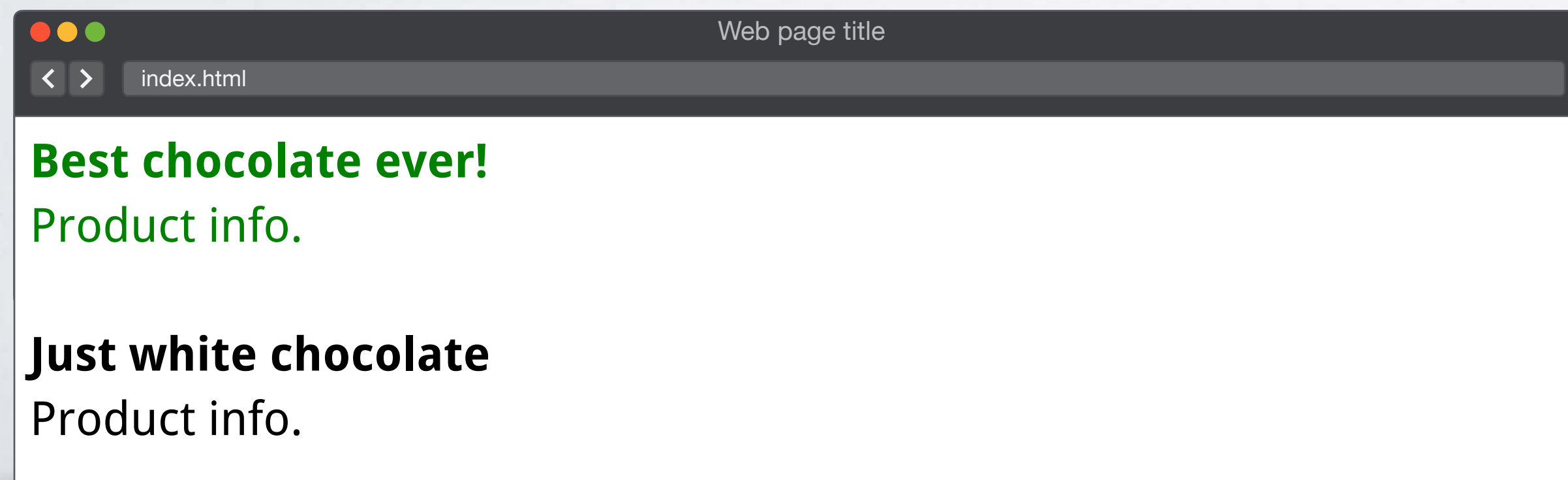
HTML

```
<body>
<article data-item="food choco black">
<h2>Best chocolate ever!</h2>
<p>Product info.</p>
</article>
<article data-item="choco food white">
<h2>Just white chocolate</h2>
<p>Product info.</p>
</article>
</body>
```

CSS

```
[data-item^=food] { color: green; }
```

Browser



Substring matching attribute selectors

[attribute^=value]

[attribute*=value]

[attribute\$=value]

ATTRIBUTE SELECTOR

Selects all elements with a **specific attribute** if their value contains a particular text.

Syntax `[attribute*=value] {style properties}`

`[data-item*=food] {color: green}`

With this code all elements with the data-item attribute and its value containing food are shown in green.

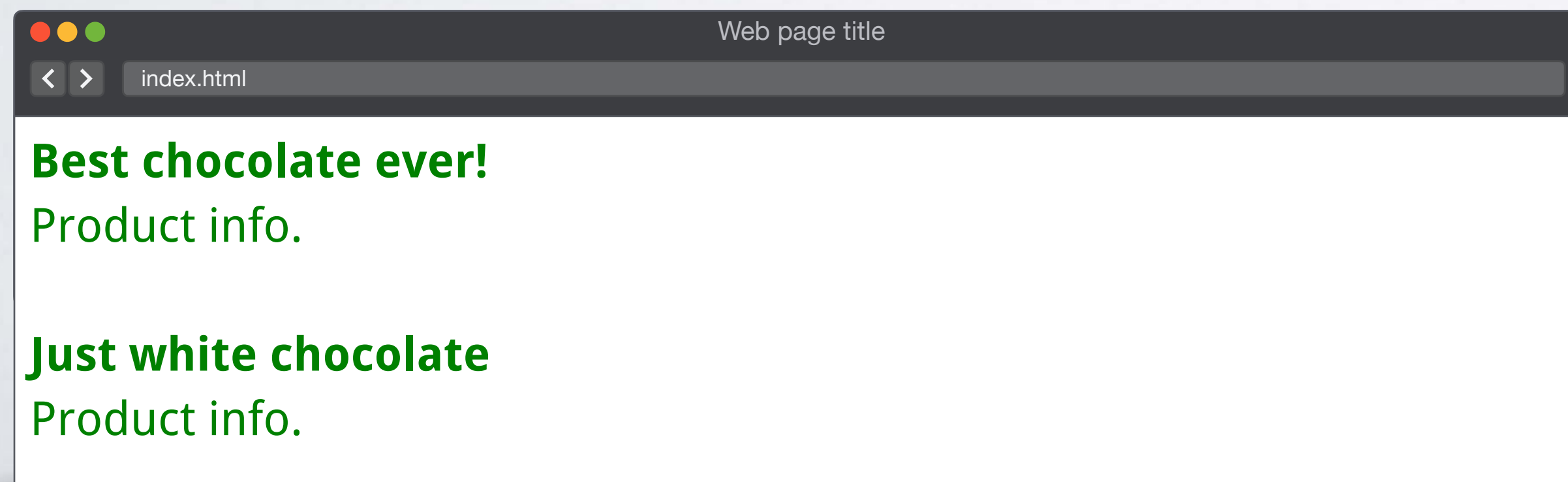
HTML

```
<body>
<article data-item="choco food black">
<h2>Best chocolate ever!</h2>
<p>Product info.</p>
</article>
<article data-item="choco white food">
<h2>Just white chocolate</h2>
<p>Product info.</p>
</article>
</body>
```

CSS

```
[data-item*=food] { color: green; }
```

Browser



Substring matching attribute selectors

[attribute[^]=value]

[attribute^{*}=value]

[attribute^{\$}=value]

ATTRIBUTE SELECTOR

Selects all elements with a **specific attribute** if their value ends with a particular text.

Syntax `[attribute$=value] {style properties}`

`[data-item$=food] {color: green}`

With this code all elements with the data-item attribute and its value ending with food are shown in green.

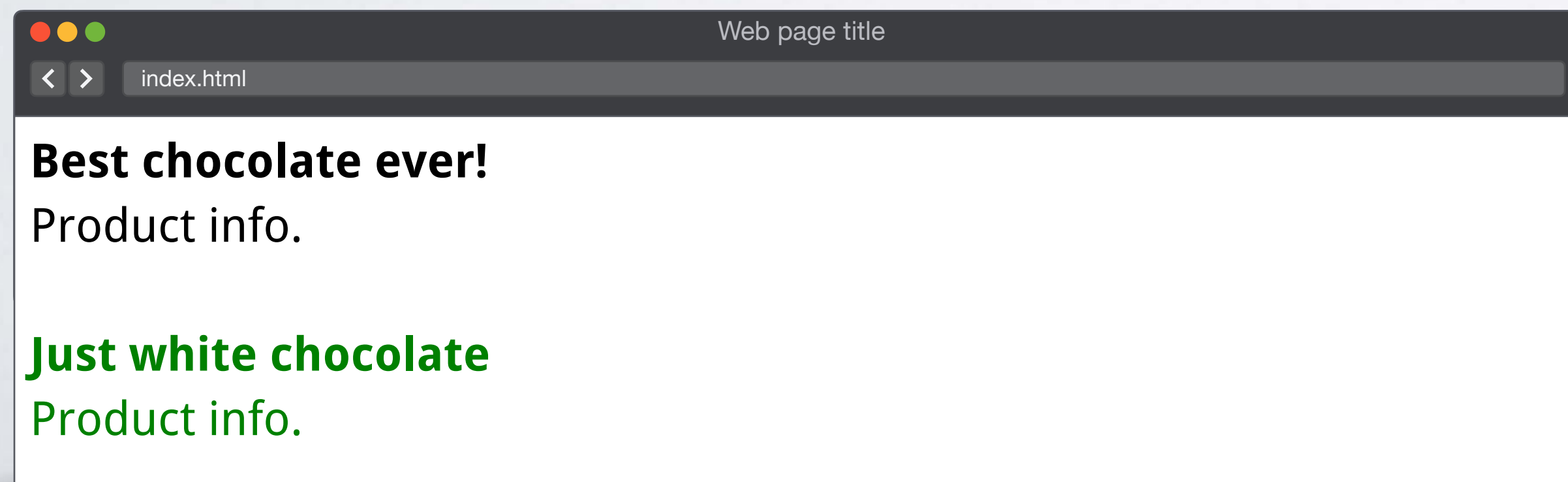
HTML

```
<body>
<article data-item="choco food black">
<h2>Best chocolate ever!</h2>
<p>Product info.</p>
</article>
<article data-item="choco white food">
<h2>Just white chocolate</h2>
<p>Product info.</p>
</article>
</body>
```

CSS

```
[data-item$=food] { color: green; }
```

Browser



**ATTRIBUTE
PRESENCE
AND VALUE
SELECTORS**



**SUBSTRING
MATCHING
ATTRIBUTE
SELECTORS**



W3C Recommendation

TABLE OF CONTENTS

1.

Introduction

1.1.

Dependencies

1.2.

Terminology

1.3.

Changes from CSS2

2.

Selectors

3.

Case sensitivity

4.

Selector syntax

5.

Groups of selectors

6.

Simple selectors

6.1.

Type selector

6.1.1.

Type selectors and namespaces

6.2.

Universal selector

6.2.1.

Universal selector and namespaces

6.3.

Attribute selectors

6.3.1.

Attribute presence and value selectors

6.3.2.

Substring matching attribute selectors

6.3.3.

Attribute selectors and namespaces

6.3.4.

Default attribute values in DTDs

6.4.

Class selectors

6.5.

ID selectors

6.6.

Pseudo-classes

6.6.1.

Dynamic pseudo-classes

6.6.1.1.

The link pseudo-classes: :link and :visited

6.6.1.2.

The user action pseudo-classes :hover, :active, and :focus

6.6.2.

The target pseudo-class :target

6.6.3.

The language pseudo-class :lang

Selectors Level 3

W3C Recommendation 06 November 2018



This version:

<https://www.w3.org/TR/2018/REC-selectors-3-20181106/>

Latest version:

<https://www.w3.org/TR/selectors-3/>

Previous version:

<https://www.w3.org/TR/2018/PR-selectors-3-20180911/>

Latest version of Selectors:

<https://www.w3.org/TR/selectors/>

Editor's Draft

<https://drafts.csswg.org/selectors-3/>

Feedback:

File an [issue on GitHub](#)

Editors:

[Tantek Çelik](#) (Invited Expert)

[Elika J. Etemad](#) (Invited Expert)

Daniel Glazman (Disruptive Innovations SARL)

[Ian Hickson](#) (Google)

Peter Linss (former editor, [Netscape/AOL](#))

John Williams (former editor, [Quark, Inc.](#))

Please check the [errata](#) for any errors or issues reported since publication.

Copyright © 2018 [W3C](#)® ([MIT](#), [ERCIM](#), [Keio](#), [Beihang](#)). [W3C liability](#), [trademark](#) and [document use](#) rules apply.

Abstract

Selectors are patterns that match against elements in a tree, and as such form one of several technologies that

SOURCE: [Selectors Level 3 by W3C](#).

inrocket.com

Learn front-end development at rocket speed



We respect your time

Just direct, clear
videos and slides.



Step-by-step guides

Actionable steps for creating
real-world solutions.



Ready-to-use code

Code you can just copy &
paste into your projects.



No more subscriptions

Subscription-free
premium courses.

TRADEMARK DISCLOSURE

This course is an independent initiative and is not affiliated, endorsed, sponsored, or officially connected with any of the trademarks mentioned in these educational materials. Trademarks, service marks, and logos used in this course belong to their respective owners.

The use of these marks in our course is solely for educational purposes and descriptive identification. It does not imply any association with the trademark holder or endorsement by them.

All information provided is intended as educational material and should not be taken as legal advice. If you have questions regarding the use of copyrighted material or trademarks, please consult with an intellectual property lawyer.



This work is licensed under
CC Attribution-NonCommercial-NoDerivatives 4.0 International.

