Webapp Development

NTU CSIE Mobile and HCI Research Lab

The New York Times
Deal for Deputy for Academics Clears Way for Schools Chief
Fruitful Talent Made Art World Multiply
Iraq's Troubles Drive Out Refugees Who Came Back
MacBook Pro

CSS

HTML

JS

<html></html>

HyperText Markup Language

```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type"
              content="text/html; charset=UTF-8">
        <title>A hello world page</title>
        <script src="js/jquery.js"
                type="text/javascript"></script>
    </head>
    <body>
        <p>Hello World!</p>
        <hr>
        <!-- A comment here -->
        <div id="visualBlock"></div>
    </body>
</html>
```

```html
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type"
              content="text/html; charset=UTF-8">
        <title>A hello world page</title>
        <script src="js/jquery.js"
                type="text/javascript"></script>
    </head>
    <body>
        <p>Hello World!</p>
        <hr>
        <!-- A comment here -->
        <div id="visualBlock"></div>
    </body>
</html>
```

```
<!DOCTYPE html>
<html>
    <he
         content="text/html; charset=UTF-8">
        <title>A hello world page</title>
        <script src="js/jquery.js"
                type="text/javascript"></script>
    </head>
    <body>
        <p>Hello World!</p>
        <hr>
        <!-- A comment here -->
        <div id="visualBlock"></div>
    </body>
</html>
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type"
              content="text/html; charset=UTF-8">
        <title>A hello world page</title>
        <script src="js/jquery.js"
                type="text/javascript"></script>
    </head>
    <body>
        <p>He
        <hr>
        <!-- A comment here -->
        <div id="visualBlock"></div>
    </body>
</html>
```

```
<meta charset="utf-8">
```

```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type"
              content="text/html; charset=UTF-8">
        <title>A hello world page</title>
        <script src="js/jquery.js"
                                cript"></script>
    </head>
    <body>
        <p>Hello World!</p>
        <hr>
        <!-- A comment here -->
        <div id="visualBlock"></div>
    </body>
</html>
```

```
<hr />
```

- `<br> <hr>`
  Change line

- `<ul></ul>`
  `<ol></ol>`
  `<li></li>`
  List items

- `<div></div>`
  Block items

- `<span></span>`
  Inline items

- `<a></a>`
  Hyperlink

- `<img>`
  Image

- `<table></table>`
  `<tr></tr> <td></td>`
  Table data

- `<embed></embed>`
  Embed plugin

- `<form></form>`
  `<input> ...`
  `<select></select>`
  `<textarea></textarea>`
  User input field

```
selector { property: value; }
```

Cascading Style Sheets

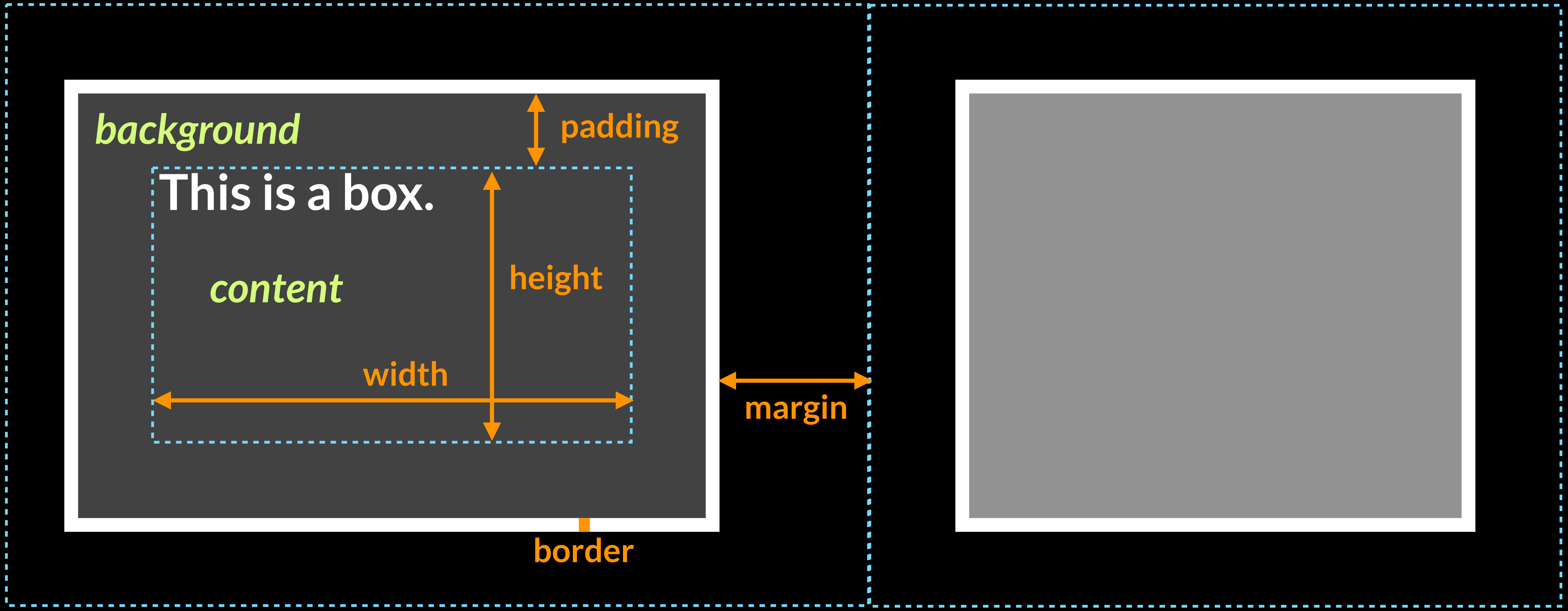
- **Inline style**
  inside the HTML document, style information on a single element, specified using the "style" attribute

- **Embedded style**
  blocks of CSS information inside the HTML itself

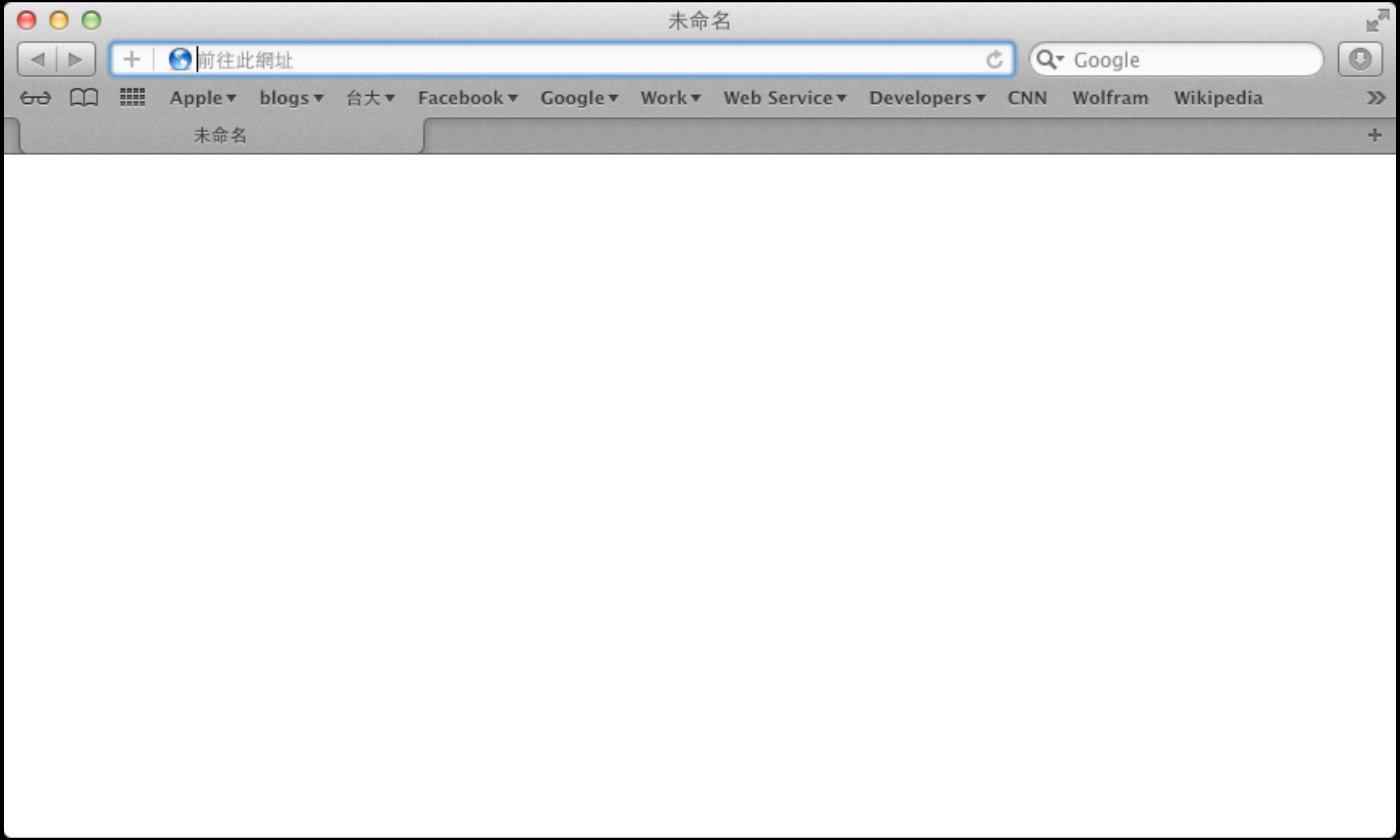
- **External style sheets**
  a separate CSS file referenced from the document

```css
h1 {
    color: #FFF;
    background: #333 !important;
}

#titleField {
    font-weight: bold;
}

.highlight {
    display: inline-block;
    background: #FF0;
}

#titleField, #contentField {
    font-size: 1.2em;
}
```

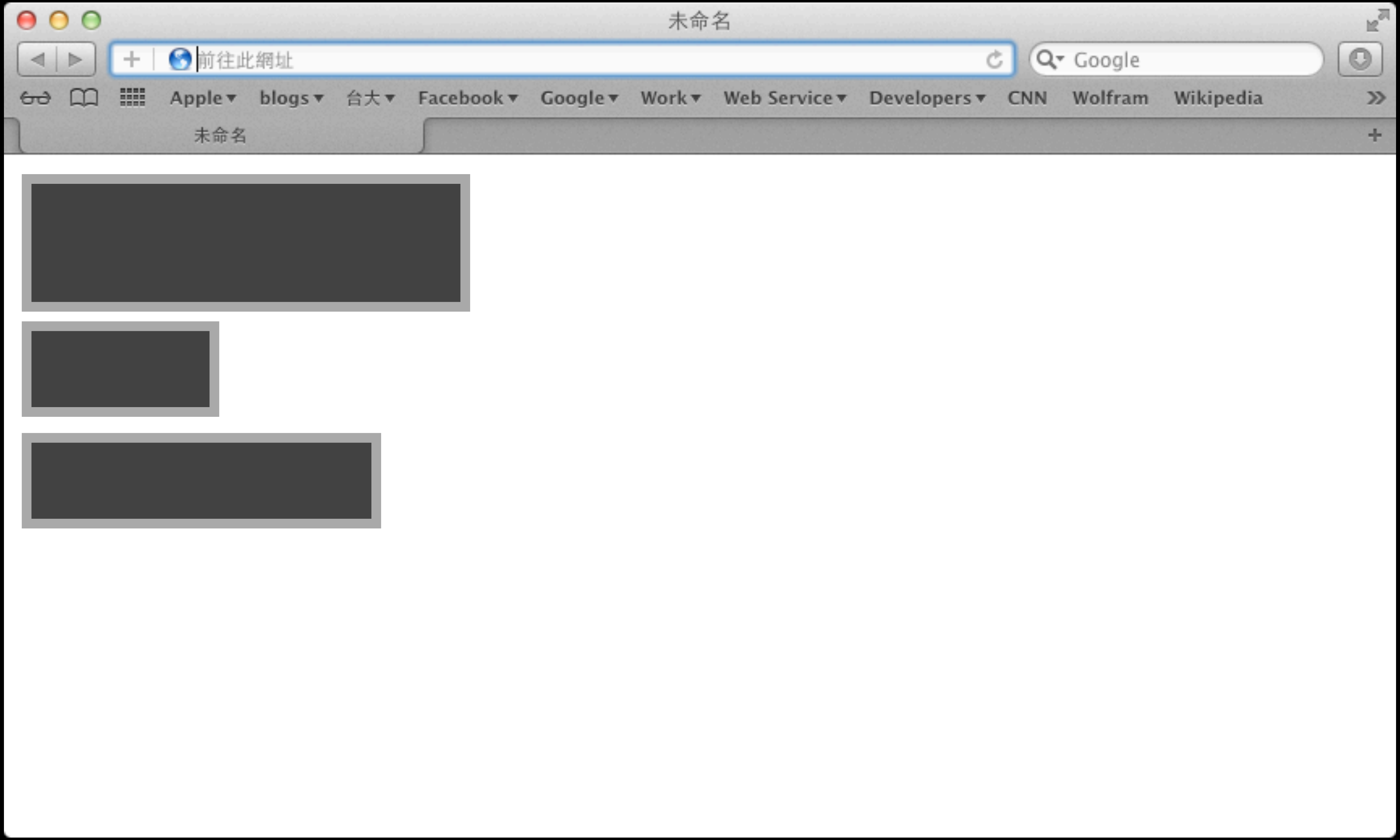
```html
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf=8">
    <title>Title</title>
    <style>
        body { color: #000; }
    </style>
    <link href="css/style.css"
          rel="stylesheet"
          type="text/css">
</head>
<body>
    <h1>Hello</h1>
    <div id="titleField">
        Title
        <span class="highlight">!!</span>
    </div>
    <div id="contentField">Content</div>
    <div style="font-size: 2em;">
        Big Size Text
    </div>
</body>
</html>
```



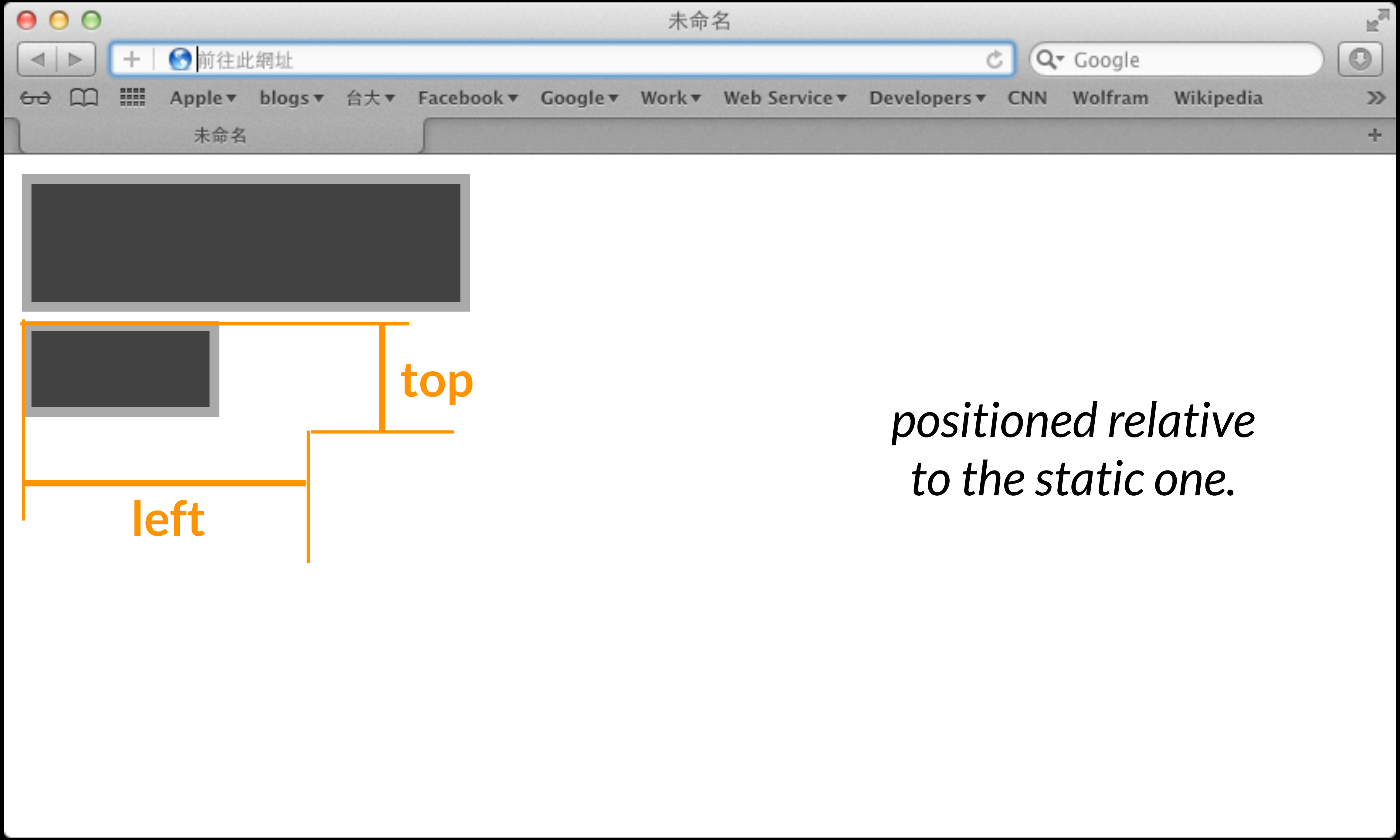
CSS box model



CSS positioning

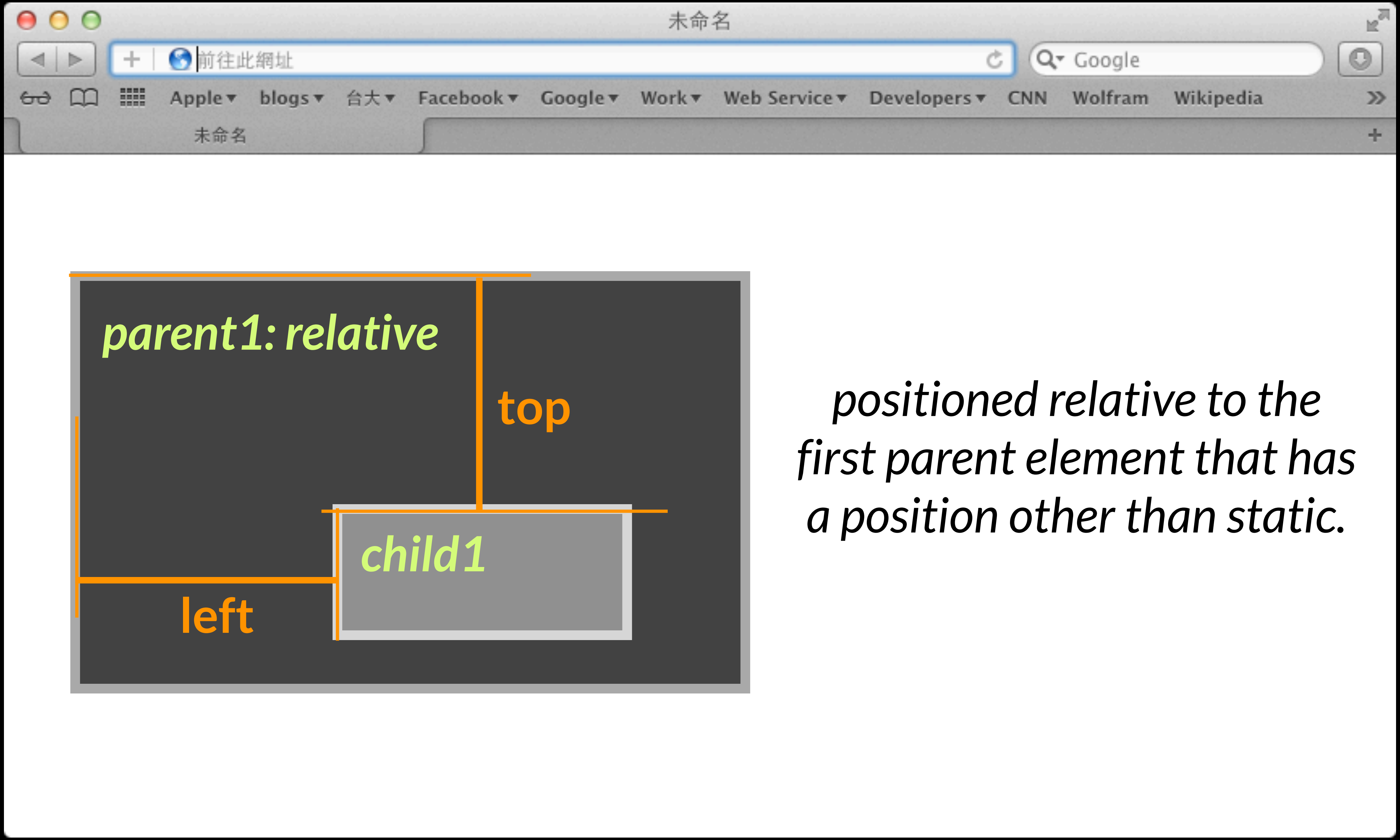


static, default



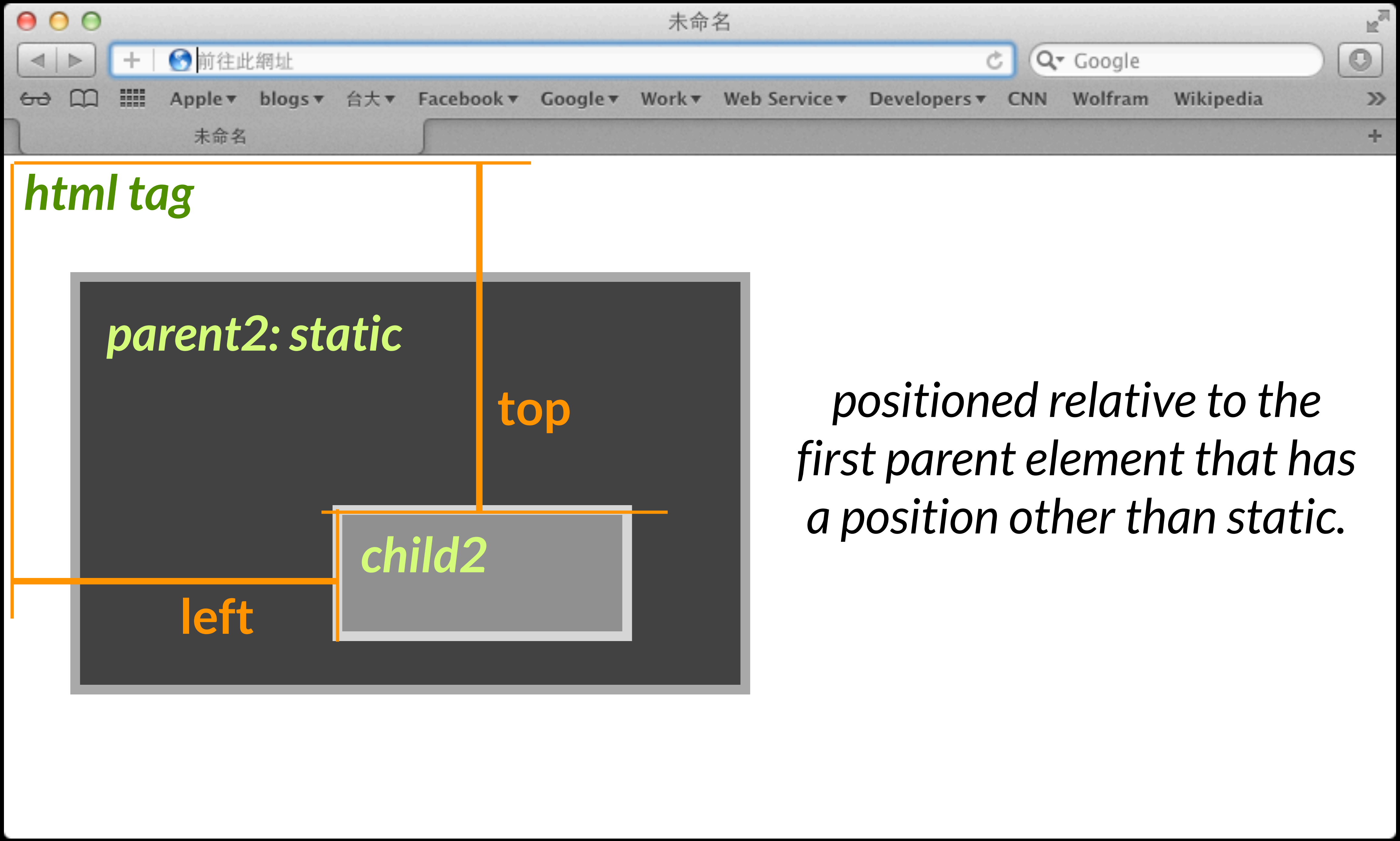
positioned relative to the static one.

relative



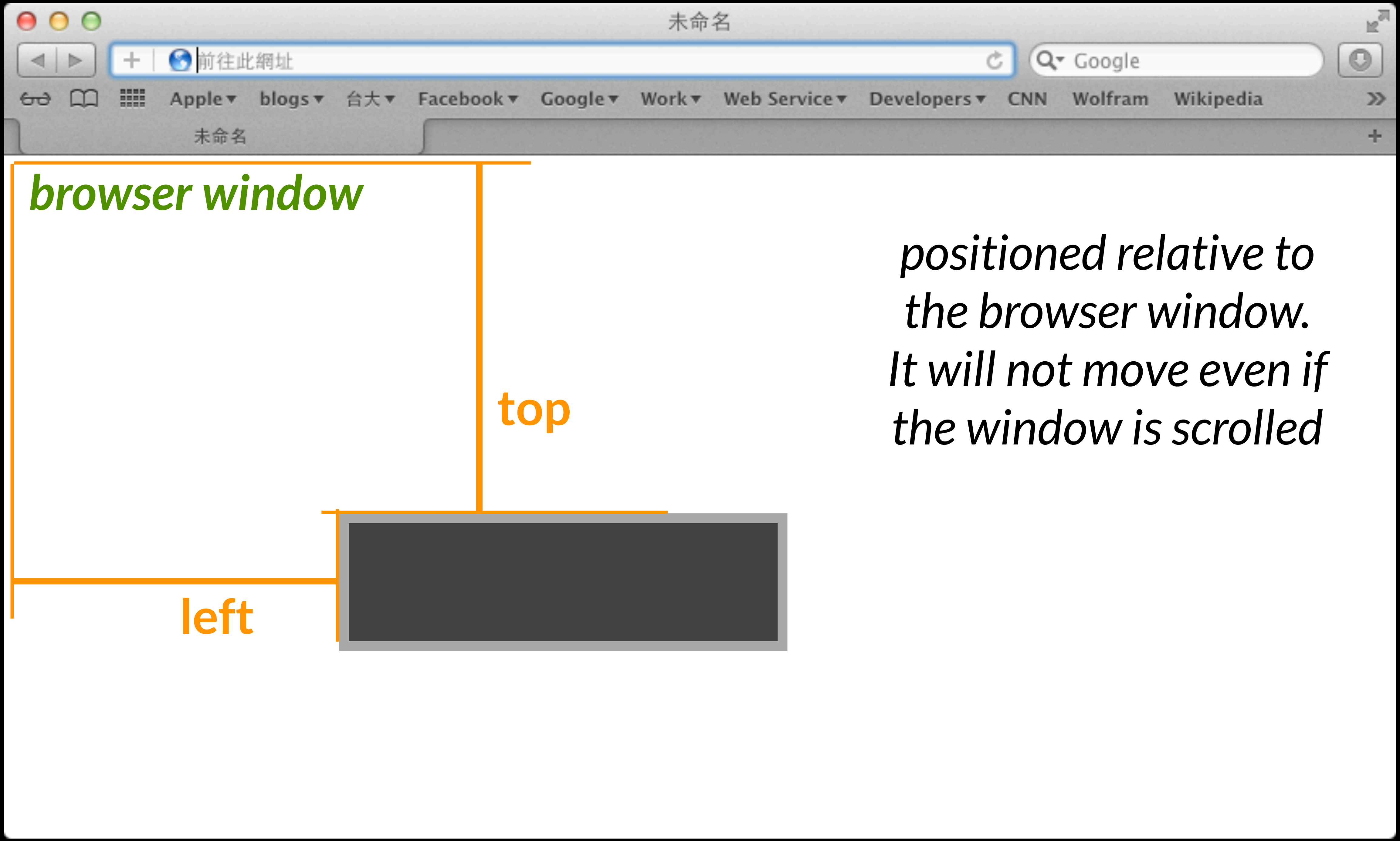
positioned relative to the first parent element that has a position other than static.

absolute



positioned relative to the first parent element that has a position other than static.

absolute



positioned relative to the browser window. It will not move even if the window is scrolled

fixed

```
#FF00CC, #FFF,
rgba(255,255,255,1),
hsla(360,100%,100%,1)
```

CSS Color

`em, px, %`

CSS Units

CSS Values

```
padding: 5px;

padding-top: 5px;
padding-right: 5px;
padding-bottom: 5px;
padding-left: 5px;
```

CSS Values

```
padding: 5px 10px;

padding-top: 5px;
padding-right: 10px;
padding-bottom: 5px;
padding-left: 10px;
```

CSS Values

```css
padding: 5px 6px 7px 8px;

padding-top: 5px;
padding-right: 6px;
padding-bottom: 7px;
padding-left: 8px;
```

CSS Values

```
border: 1px solid #CCC;

border-width: 1px;
border-style: solid;
border-color: #CCC;
```

CSS selectors

```html
<!DOCTYPE html>
<html>
    <head>...</head>
    <body>
        <p>Hello World!</p>
        <hr>
        <!-- A comment here -->
        <div id="visualBlock"></div>
        <div class="endBlock"></div>
    </body>
</html>
```

- `html`
  Element
- `#titleField`
  ID
- `.highlight`
  Class
- `[type=”text”]`
  Attribute
- `:hover`
  Pseudo-Class
- `:first-child`
  Pseudo-Element
- ` , >`
  Combinator
- `div.highlight:hover`
  Mixable

HTML Sample

```html
<div id="a" class="b">
    <div class="b">
        <div></div>
        <input name="key">
    </div>
</div>
```

css: div

```
<div id="a" class="b">
    <div class="b">
        <div></div>
        <input name="key">
    </div>
</div>
```

css: div#a

```
<div id="a" class="b">
    <div class="b">
        <div></div>
        <input name="key">
    </div>
</div>
```

css: div.b

```
<div id="a" class="b">
    <div class="b">
        <div></div>
        <input name="key">
    </div>
</div>
```

css: div#a.b

```
<div id="a" class="b">
    <div class="b">
        <div></div>
        <input name="key">
    </div>
</div>
```

css: div#a > div

```
<div id="a" class="b">
    <div class="b">
        <div></div>
        <input name="key">
    </div>
    <div></div>
</div>
```

css: div#a > div:first-child

```
<div id="a" class="b">
    <div class="b">
        <div></div>
        <input name="key">
    </div>
    <div></div>
</div>
```

css: div#a div

```
<div id="a" class="b">
    <div class="b">
        <div></div>
        <input name="key">
    </div>
    <div></div>
</div>
```

css: input[name="key"]

```
<div id="a" class="b">
    <div class="b">
        <div></div>
        <input name="key">
        <input name="pwd">
    </div>
</div>
```

css: #a > input[name="key"]

```
<div id="a" class="b">
    <div class="b">
        <div></div>
        <input name="key">
        <input name="pwd">
    </div>
</div>
```

css: #a input[name="key"]

```
<div id="a" class="b">
    <div class="b">
        <div></div>
        <input name="key">
        <input name="pwd">
    </div>
</div>
```

```
document.getElementById("title")[0].innerHTML
```

Javascript

JavaScript is a prototype-based, object-oriented scripting language that is dynamic, weakly typed.

It has first-class functions and also supports higher-order functions.

- **Navigator**
  Top-level object in javascript
- **Window**
  object corresponding to the browser window
- **Document**
  the container for all objects within the HTML tags of an HTML document
- **Location**
  a property of the window object
- **History**
  a property of the window object

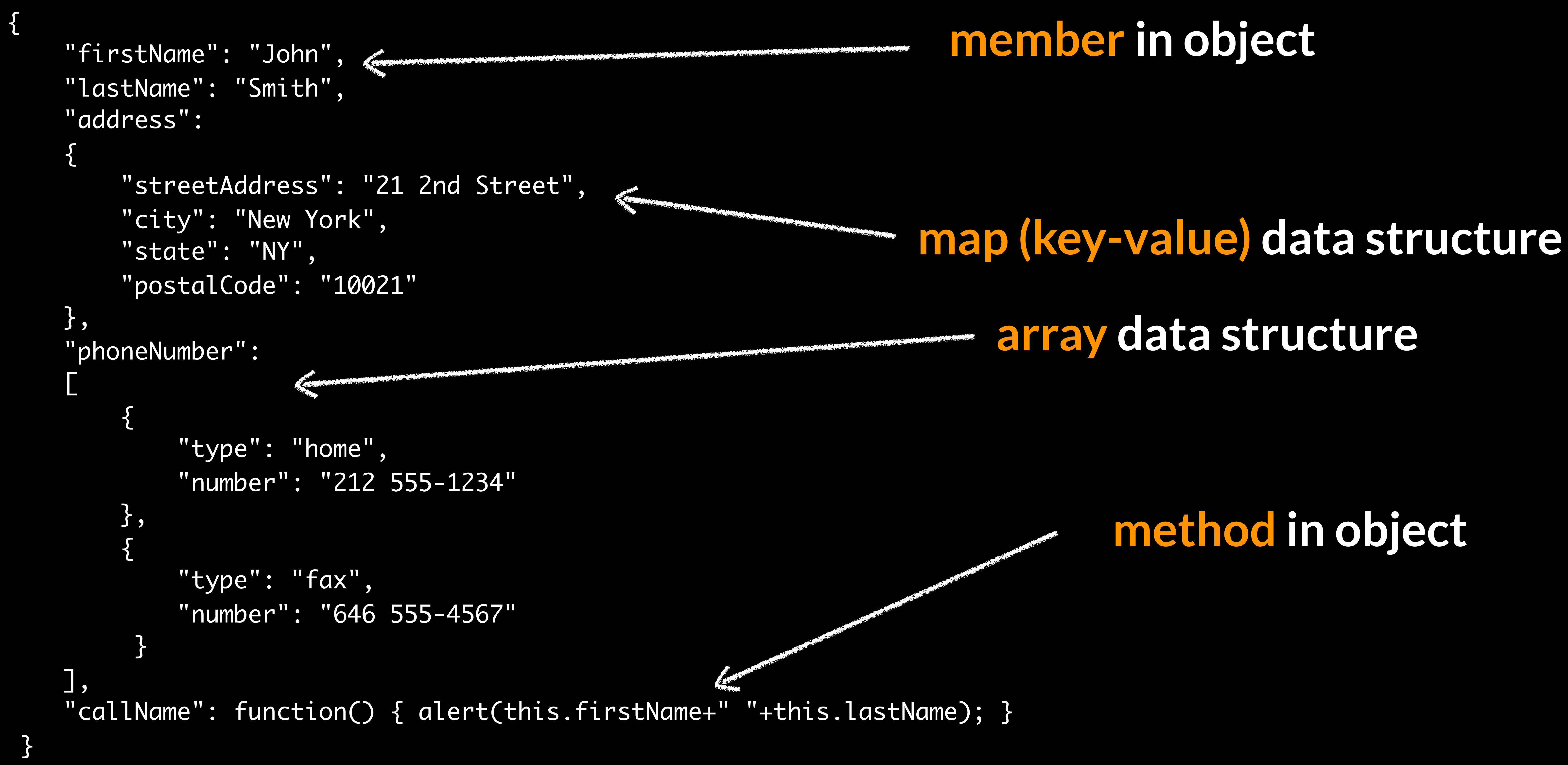
```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf=8">
    <title>Title</title>
    <script src="js/index.js" type="text/javascript"></script>
    <script type="text/javascript">
        alert("XD");
    </script>
</head>
<body>
</body>
</html>
```

Nitro, V8, SpiderMonkey, Chakra

JavaScript Engine

JSON

JavaScript Object Notation



```
{
    "firstName": "John",
    "lastName": "Smith",
    "address":
    {
        "streetAddress": "21 2nd Street",
        "city": "New York",
        "state": "NY",
        "postalCode": "10021"
    },
    "phoneNumber":
    [
        {
            "type": "home",
            "number": "212 555-1234"
        },
        {
            "type": "fax",
            "number": "646 555-4567"
         }
    ],
    "callName": function() { alert(this.firstName+" "+this.lastName); }
 }
```

member in object

map (key-value) data structure

array data structure

method in object

HTML

HTML5

- New Semantic tags
- Clear tags
- Canvas tag
- Timed media playback
- Offline storage database
- Document editing
- Drag-and-drop
- Cross-document messaging
- Browser history management
- MIME type and protocol handler registration
- Microdata
- GeoLocation
- Web SQL Database
- Indexed DB API
- SVG, MathML, WebGL

HTML5 SEMANTICS

OFFLINE & STORAGE

DEVICE ACCESS

CONNECTIVITY

MULTIMEDIA

3D, GRAPHICS, EFFECTS

PERFORMANCE & INTEGRATION

CSS3 STYLING

HTML5 Semantic tags

- `<section></section>`
- `<nav></nav>`
- `<article></article>`
- `<aside></aside>`

- `<hgroup></hgroup>`
  Head groups
- `<header></header>`
- `<footer></footer>`

Geolocation API

EMBED vs. AUDIO/VIDEO

Video Formats

- **OGG Theora**
  Free, open standard container format maintained
  Safari and IE are not support.
- **H.264**
  Part of MPEG4. Youtube use H.264 in HTML5 mode.
  Supported by Safari and Chrome only.
- **Vp8/WebM**
  Sponsored by Google.
  Currently supported by Chrome and Opera. Firefox will support in the feature.

Audio Formats

- **OGG Vorbis**
  Firefox and Opera only.
- **WAV PCM**
  Supported by Safari and Chrome only.
- **MP3**
  Supported by Safari and Chrome only.
- **AAC**
  Supported by Safari and Chrome only.

Cookies vs. Local Storage

Local Storage

- More disk space available than cookie
- Access more easily
- Less bandwidth
- **Poor Security**
- **Local/Session Storage**

Local Database vs. Cloud Database

WebSQL Database vs. Indexed DB API

http://caniuse.com/

check browser compatibility

Let’s see some **example** about **HTML5**, **CSS3**, and **Javascript**.

jQuery

write less, do more.

jQuery is a new kind of Javascript Library.

Browser Compatibility

```
$("html").click();
```

```
function sampleA() { alert("3"); }

$("body").click(sampleA);
```

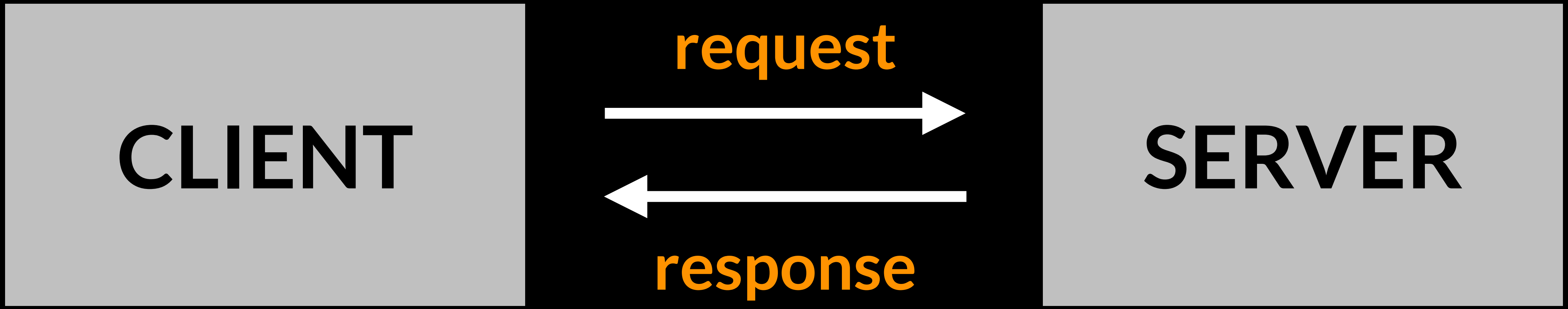
```
$("div.button").click(function () {
   alert("3");
});
```

```
$("input").focus(function () {
   $(this).addClass("focus");
});
$("input").blur(function () {
   $(this).removeClass("focus");
});
```

```
$("input").focus(function () {
   $(this).addClass("focus");
}).blur(function () {
   $(this).removeClass("focus");
});
```

http://docs.jquery.com/

Let’s check some **example** about **jQuery.**



CLIENT

request

response

SERVER

```
GET /index.html HTTP/1.1
Host: www.example.com
```

- **HEAD**

  Retrieve meta-information written in response headers only. No content.

- **GET**

  Requests a representation of the specified resource. Default method of HTTP Request.

- **POST**

  Submits data to be processed to the identified resource.

- **PUT**

  Uploads a representation of the specified resource.

- **DELETE**

  Deletes the specified resource

```
HTTP/1.1 200 OK
Date: Mon, 23 May 2005 22:38:34 GMT
Server: Apache/1.3.3.7 (Unix) (Red-Hat/Linux)
Last-Modified: Wed, 08 Jan 2003 23:11:55 GMT
Etag: "3f80f-1b6-3e1cb03b"
Accept-Ranges: bytes
Content-Length: 438
Connection: close
Content-Type: text/html; charset=UTF-8
```

- **1xx Informational**
  Request received, continuing process.

- **2xx Success**
  This class of status codes indicates the action requested by the client was received, understood, accepted and processed successfully.

- **3xx Redirection**
  The client must take additional action to complete the request.

- **4xx Client Error**
  Client seems to have erred. These are typically the most common error codes encountered while online.

- **5xx Server Error**
  The server failed to fulfill an apparently valid request.

RESTful Web Services

http://api.food.com/cities/

http://api.food.com/cities/tw/taipei/

http://api.food.com/cities/tw/taipei/?type=json

Work with GET, POST, PUT, and DELETE

http://api.food.com/cities/tw/taipei/?action=PUT

Alternative type of RESTful API

AJAX

Asynchronous Javascript And XML

```
$.ajax({
    // Connection Setting
    async: true,
    cache: false,
    global: true,
    // Target and Data transfer
    url: 'http://api.food.com/cities/tw/taipei/',
    type: 'GET',
    data: {
      a: 'b'
    },
    // Return type
    dataType: 'html',
    // Event
    beforeSend: null,
    error: function (xhr) { alert("AJAX failed!"); },
    success: function (response) { $("div#ajaxResult").html(response); },
    complete: null,
});
```

Client Side

Server Side

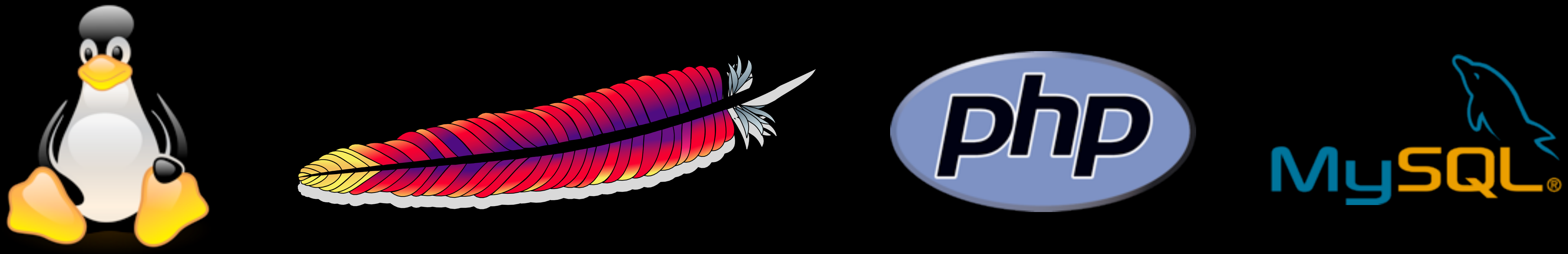
RAILS

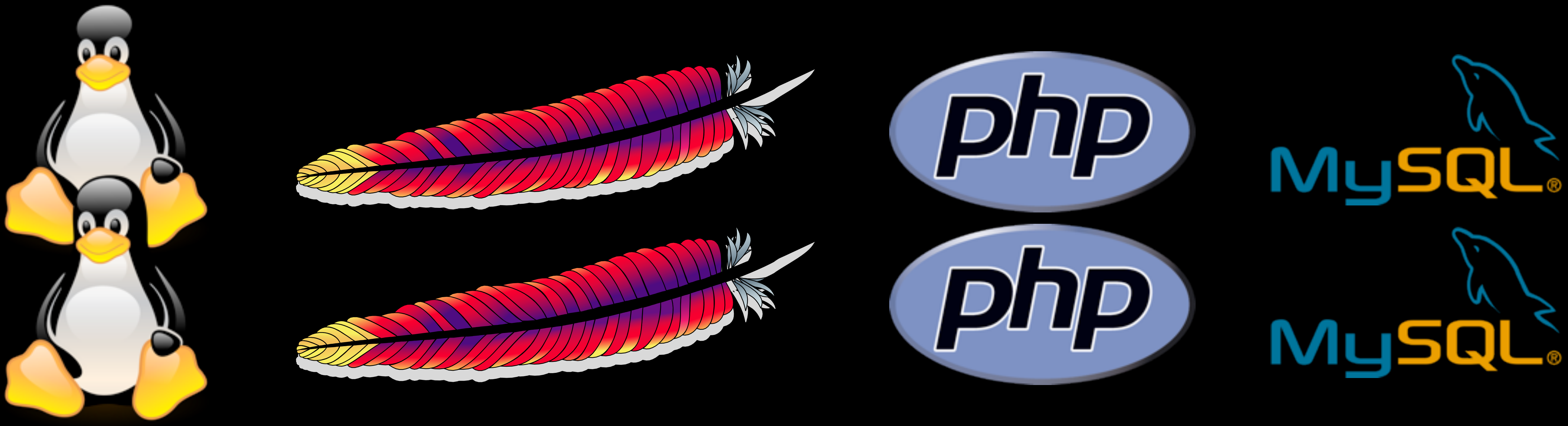
django

Server Side

- **Client Side**
  User experience, User Interface, Preference, Artwork, User Input, Short-term storage, and Animation.

- **Server Side**
  Record account information, User auth/audit, Complex computation, Database, and Long-term storage.


php
MySQL®


php
php
MySQL®
MySQL®

amazon
web services™