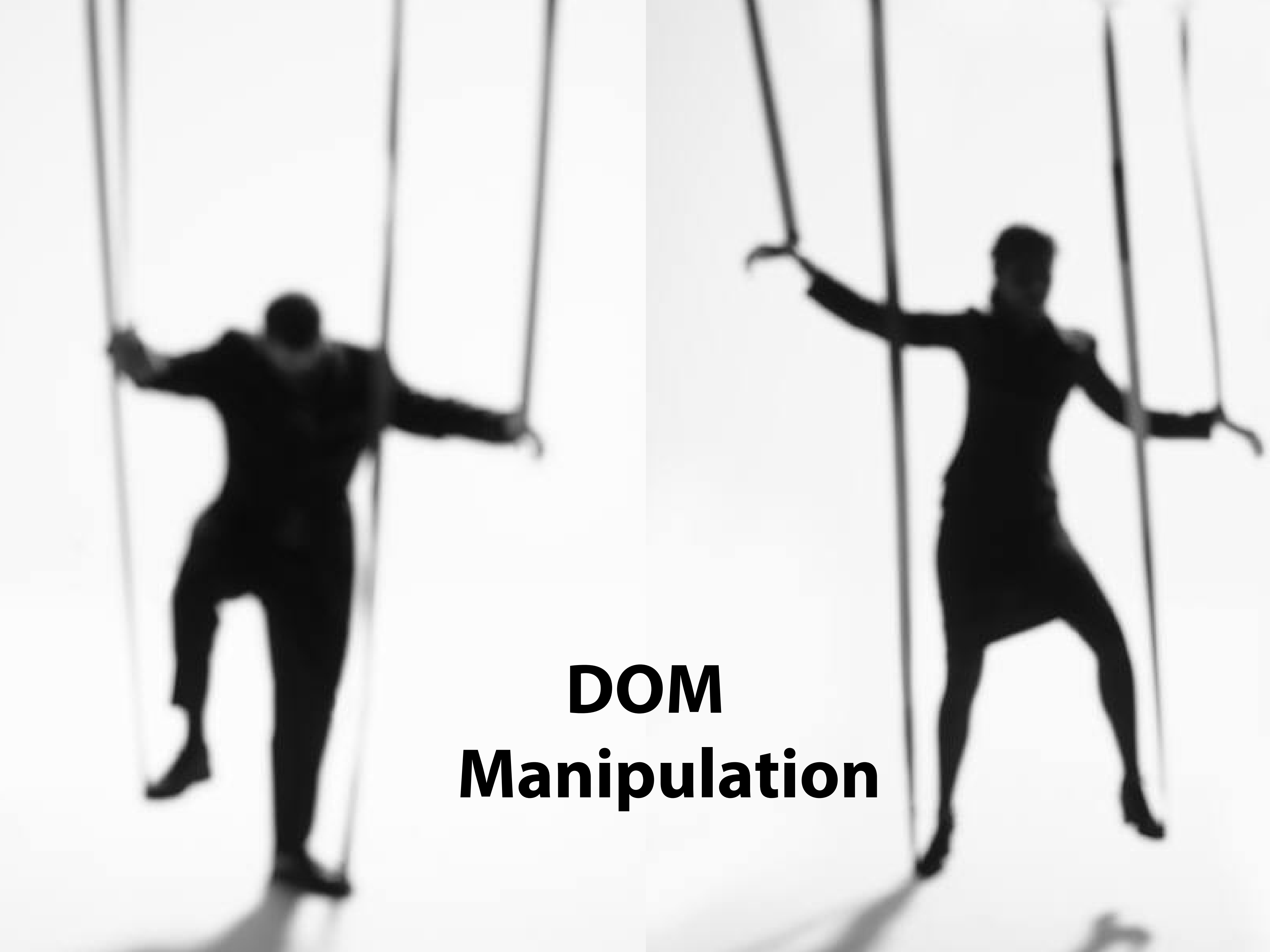
jQuery-free JavaScript

Native DOM Manipuation

Elijah Manor
http://elijahmanor.com
@elijahmanor

pluralsight
hardcore developer training



DOM Manipulation

Classes

jQuery

```
var widget = $("#widget");

widget.addClass("active");
widget.addClass("active dirty");
widget.removeClass("active");
widget.toggleClass("active");
widget.hasClass("active");
```

Classes

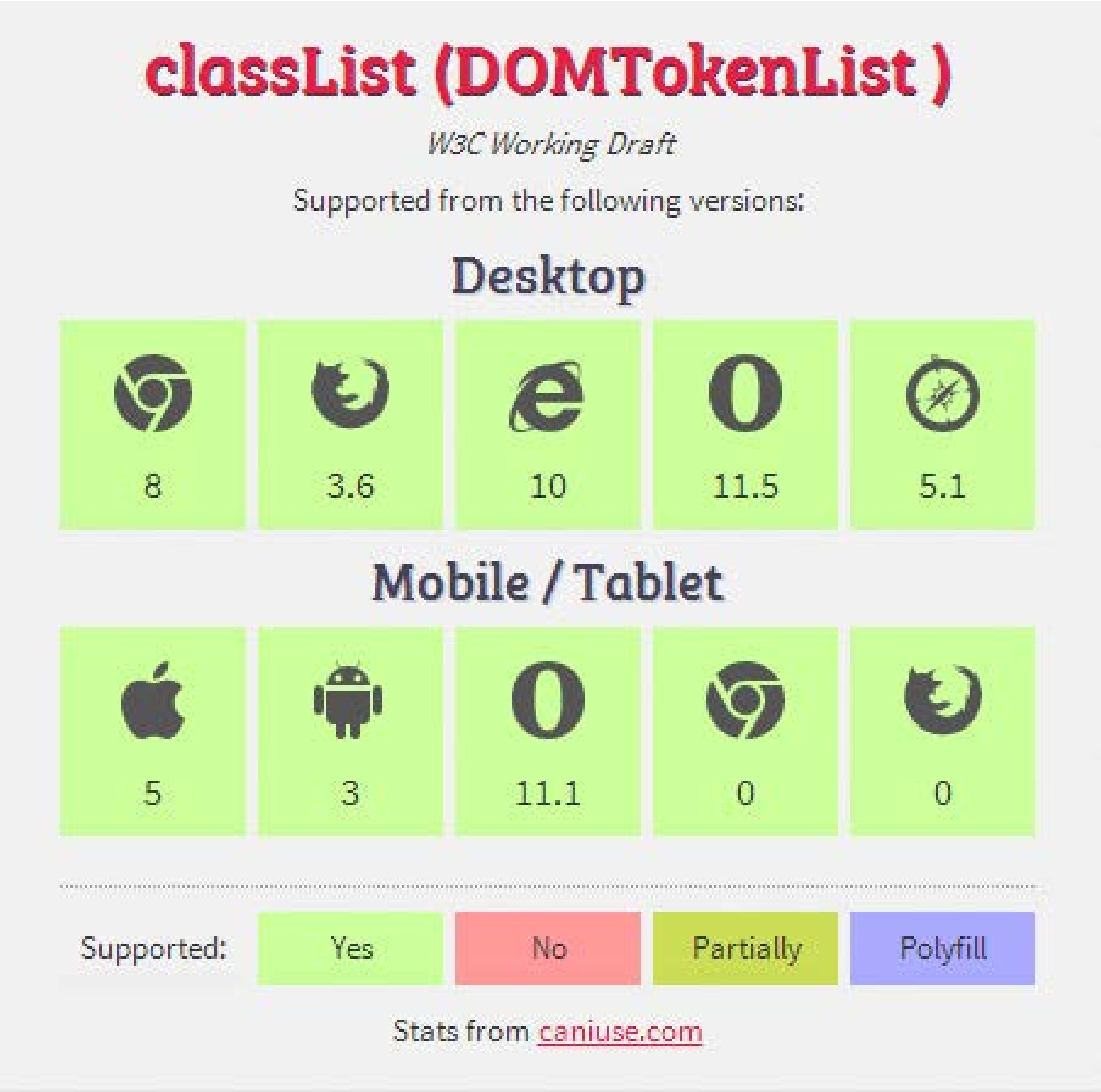
Native: Opt

```
document.                                           active";
document.
```

Native: Opt

```
// https:
var widge

widget.cl
widget.cl
widget.cl
widget.cl
widget.cl
```



classList (DOMTokenList)

W3C Working Draft

Supported from the following versions:

Desktop

8	3.6	10	11.5	5.1

Mobile / Tablet

5	3	11.1	0	0

Supported:	Yes	No	Partially	Polyfill

Stats from caniuse.com

HTML & Text

jQuery

```
var widget = $("#widget");
widget.html("<strong>hello</strong>");       // Setter
widget.html;                                 // Getter
widget.text("goodbye");                      // Setter
widget.text;                                 // Getter
```

Native

```
var widget = document.getElementById("widget");
widget.innerHTML = "<strong>hello</strong>"; // Setter
widget.innerHTML;                            // Getter
widget.innerText = "goodbye";                // Setter
widget.innerText;                            // Getter
```

Append & Prepend

jQuery

```
$("#jquery").prepend("<div>Hello jQuery</div>");
$("#jquery").append("<div>Goodbye jQuery</div>");
```

Native

```
var pDiv = document.createElement("div");
pDiv.innerText = "Hello Vanilla";
document.getElementById("vanilla").insertBefore(pDiv);

var aDiv = document.createElement("div");
aDiv.innerText = "Goodbye Vanilla";
document.getElementById("vanilla").appendChild(aDiv);
```

Remove

jQuery

```
var widget = $("#widget");

widget.empty();
widget.html("");
widget.remove();
```

Native

```
var widget = document.getElementById("widget");
widget.innerHTML = "";
while (widget.lastChild) {
  widget.removeChild(widget.lastChild);
}
widget.parentNode.removeChild(widget);
```

CSS

jQuery: Option 1

```
var widget = $("#widget");
widget.css("color", "red");
widget.css("background-color", "blue");
widget.css("font-size", "20px");
```

jQuery: Option 2

```
var widget = $("#widget");
widget.css({
    color: "red",
    "background-color": "blue",
    fontSize: "20px",
});
```

CSS

Native: Option 1

```
var widget = document.getElementById("widget");
widget.style.color = "red";
widget.style.backgroundColor = "blue";
widget.style.fontSize = "20px";
```

Native: Option 2

```
var widget = document.getElementById("widget");

widget.style["color"] = "red";
widget.style["backgroundColor"] = "blue";
widget.style["fontSize"] = "20px";
```

CSS

Native: Option 3

```
var widget = document.getElementById("widget");

function updateStyle(element, styles) {
  for (var style in styles) {
    if (styles.hasOwnProperty(style)) {
      element.style[style] = styles[style];
    }
  }
}

updateStyle(widget, {
    color: "red",
    backgroundColor: "blue"
});
```

Attributes & Properties

jQuery

```
var search = $("#search"),
  toggle = $("#toggle"),
  anchor = $("#about");

search.attr("placeholder", "Search here...");
search.attr("placeholder");

toggle.prop("checked", true);
toggle.prop("checked");

anchor.attr("href"); //./pages/about.html
anchor.prop("href"); //http://domain.com/pages/about.html
```

Attributes & Properties

Native

```
var search = document.getElementById("search"),
  toggle = document.getElementById("toggle"),
  anchor = document.getElementById("about");

search.setAttribute("placeholder", "Search here...");
search.getAttribute("placeholder");

toggle.checked = true;
toggle.checked;

anchor.getAttribute("href"); //./pages/about.html
anchor.href; //http://domain.com/pages/about.html
```

Value

jQuery

```
var search = $("#search");

search.val("Doctor Who");    // Getter
search.val();                // Setter
```

Native

```
var search = document.getElementById("search");

search.value = "Doctor Who"; // Getter
search.value;                // Setter
```

jQu

va

bo

bo

bo

Na

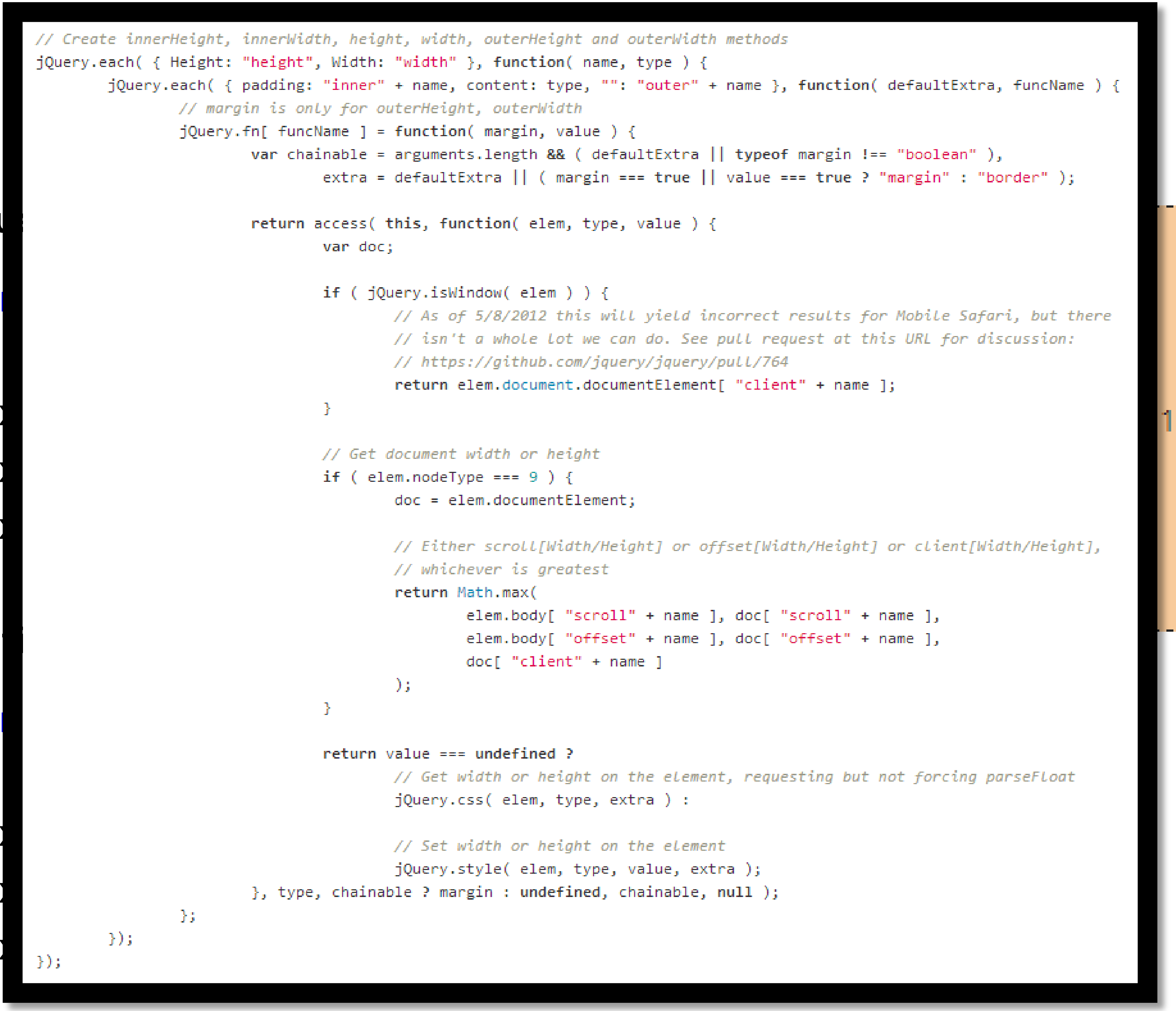
va

bo

bo

bo

10



```
// Create innerHeight, innerWidth, height, width, outerHeight and outerWidth methods
jQuery.each( { Height: "height", Width: "width" }, function( name, type ) {
    jQuery.each( { padding: "inner" + name, content: type, "": "outer" + name }, function( defaultExtra, funcName ) {
        // margin is only for outerHeight, outerWidth
        jQuery.fn[ funcName ] = function( margin, value ) {
            var chainable = arguments.length && ( defaultExtra || typeof margin !== "boolean" ),
                extra = defaultExtra || ( margin === true || value === true ? "margin" : "border" );

            return access( this, function( elem, type, value ) {
                var doc;

                if ( jQuery.isWindow( elem ) ) {
                    // As of 5/8/2012 this will yield incorrect results for Mobile Safari, but there
                    // isn't a whole lot we can do. See pull request at this URL for discussion:
                    // https://github.com/jquery/jquery/pull/764
                    return elem.document.documentElement[ "client" + name ];
                }

                // Get document width or height
                if ( elem.nodeType === 9 ) {
                    doc = elem.documentElement;

                    // Either scroll[Width/Height] or offset[Width/Height] or client[Width/Height],
                    // whichever is greatest
                    return Math.max(
                        elem.body[ "scroll" + name ], doc[ "scroll" + name ],
                        elem.body[ "offset" + name ], doc[ "offset" + name ],
                        doc[ "client" + name ]
                    );
                }

                return value === undefined ?
                    // Get width or height on the element, requesting but not forcing parseFloat
                    jQuery.css( elem, type, extra ) :

                    // Set width or height on the element
                    jQuery.style( elem, type, value, extra );
            }, type, chainable ? margin : undefined, chainable, null );
        };
    });
});
```

DEMO

- Classes
- Prepend/Append
- Remove
- CSS
- HTML/Text
- Attributes/Properties
- Value
- Height/Width

Summary

```
var $w = $("#widget");          var w = document.getElementById("widget");
$w.addClass("active");          w.classList.add("active");

$w.html("Hello World");         w.innerHTML = "Hello World";

$w.append("<div>Bye</div>");    var div = document.createElement("div");
                                div.innerText = "Bye"; w.appendChild(div);

$w.find("div").remove();        w.removeChild(div);

$w.css("color", "blue");        w.style.color = "blue";

$w.prop("title", "Howdy");      w.title = "Howdy";

var input = $("#search");       var input = document.getElementById("search");
var search = input.val();       var search = input.value;

input.val($w.outerWidth());     input.value = widget.offsetWidth;
```