

MISCELLANEOUS OBJECT DEFINITIONS

interface HTMLCollection

```
readonly attribute unsigned long length;
Node item(unsigned long index);
Node namedItem(in DOMString name);
```

interface HTMLOptionsCollection

```
attribute unsigned long length;
raises(DOMException) on
```

setting

```
Node item(in unsigned long index);
Node namedItem(in DOMString name);
```

OBJECTS RELATED TO HTML DOCUMENTS

interface HTMLDocument : Document

```
attribute DOMString title;
readonly attribute DOMString referrer;
readonly attribute DOMString domain;
readonly attribute DOMString URL;
attribute HTMLElement body;
readonly attribute HTMLCollection images;
readonly attribute HTMLCollection applets;
readonly attribute HTMLCollection links;
readonly attribute HTMLCollection forms;
readonly attribute HTMLCollection anchors;
attribute DOMString cookie;
raises(DOMException) on
```

setting

```
void open();
void close();
void write(in DOMString text);
void writeln(in DOMString text);
NodeList getElementByName(in DOMString
elementName);
```

interface HTMLElement : Element

```
attribute DOMString id;
attribute DOMString title;
attribute DOMString lang;
attribute DOMString dir;
attribute DOMString className;
```

HTML ELEMENTS – DOCUMENT STRUCTURE

interface HTMLHtmlElement : HTMLElement

```
attribute DOMString version;
```

HTML ELEMENTS – DOCUMENT HEAD

interface HTMLHeadElement : HTMLElement

```
attribute DOMString profile;
```

interface HTMLLinkElement : HTMLElement

```
attribute boolean disabled;
attribute DOMString charset;
attribute DOMString href;
attribute DOMString hreflang;
attribute DOMString media;
attribute DOMString rel;
attribute DOMString rev;
attribute DOMString target;
attribute DOMString type;
```

interface HTMLTitleElement : HTMLElement

```
attribute DOMString text;
```

interface HTMLMetaElement : HTMLElement

```
attribute DOMString content;
attribute DOMString httpEquiv;
attribute DOMString name;
attribute DOMString scheme;
```

interface HTMLBaseElement : HTMLElement

```
attribute DOMString href;
attribute DOMString target;
```

interface HTMLStyleElement : HTMLElement

```
attribute boolean disabled;
attribute DOMString media;
attribute DOMString type;
```

HTML ELEMENTS – DOCUMENT BODY

interface HTMLBodyElement : HTMLElement

```
attribute DOMString aLink;
attribute DOMString background;
attribute DOMString bgColor;
attribute DOMString link;
attribute DOMString text;
attribute DOMString vLink;
```

interface HTMLDivElement : HTMLElement

```
attribute DOMString align;
```

interface HTMLScriptElement : HTMLElement

```
attribute DOMString text;
attribute DOMString htmlFor;
attribute DOMString event;
attribute DOMString charset;
attribute boolean defer;
attribute DOMString src;
attribute DOMString type;
```

HTML ELEMENTS – FORMS

interface HTMLInputElement : HTMLElement

```
readonly attribute HTMLFormElement form;
attribute DOMString prompt;
```

interface HTMLFormElement : HTMLElement

```
readonly attribute HTMLCollection elements;
readonly attribute long length;
attribute DOMString name;
attribute DOMString acceptCharset;
attribute DOMString action;
attribute DOMString enctype;
attribute DOMString method;
attribute DOMString target;
void submit();
void reset();
```

interface HTMLSelectElement : HTMLElement

```
readonly attribute DOMString type;
attribute long selectedIndex;
attribute DOMString value;
attribute unsigned long length;
raises(DOMException) on
```

setting

```
readonly attribute HTMLFormElement form;
readonly attribute HTMLOptionsCollection options;
attribute boolean disabled;
attribute boolean multiple;
attribute DOMString name;
attribute long size;
attribute long tabIndex;
void add(in HTMLElement element,
in HTMLElement before)
raises(DOMException);
void remove(in long index);
void blur();
void focus();
```

interface HTMLOptGroupElement : HTMLElement

```
attribute boolean disabled;
attribute DOMString label;
```

interface HTMLOptionElement : HTMLElement

```
readonly attribute HTMLFormElement form;
attribute boolean defaultSelected;
readonly attribute DOMString text;
readonly attribute long index;
attribute boolean disabled;
attribute DOMString label;
```

```
attribute boolean selected;
attribute DOMString value;
```

interface HTMLInputElement : HTMLElement

```
attribute DOMString defaultValue;
attribute boolean defaultChecked;
readonly attribute HTMLFormElement form;
attribute DOMString accept;
attribute DOMString accessKey;
attribute DOMString align;
attribute DOMString alt;
attribute boolean checked;
attribute boolean disabled;
attribute long maxLength;
attribute DOMString name;
attribute boolean readOnly;
attribute unsigned long size;
attribute DOMString src;
attribute long tabIndex;
attribute DOMString type;
attribute DOMString useMap;
attribute DOMString value;
void blur();
void focus();
void select();
void click();
```

interface HTMLTextAreaElement : HTMLElement

```
attribute DOMString defaultValue;
readonly attribute HTMLFormElement form;
attribute DOMString accessKey;
attribute long cols;
attribute boolean disabled;
attribute DOMString name;
attribute boolean readOnly;
attribute long rows;
attribute long tabIndex;
readonly attribute DOMString type;
attribute DOMString value;
void blur();
void focus();
void select();
```

interface HTMLButtonElement : HTMLElement

```
readonly attribute HTMLFormElement form;
attribute DOMString accessKey;
attribute boolean disabled;
attribute DOMString name;
attribute long tabIndex;
attribute DOMString type;
attribute DOMString value;
```

interface HTMLLabelElement : HTMLElement

```
readonly attribute HTMLFormElement form;
attribute DOMString accessKey;
attribute DOMString htmlFor;
```

interface HTMLFieldSetElement : HTMLElement

```
readonly attribute HTMLFormElement form;
```

interface HTMLLegendElement : HTMLElement

```
readonly attribute HTMLFormElement form;
attribute DOMString accessKey;
attribute DOMString align;
```

HTML ELEMENTS – LISTS

interface HTMLULListElement : HTMLElement

```
attribute boolean compact;
attribute DOMString label;
```

interface HTMLLOListElement : HTMLElement

```
attribute boolean compact;
attribute long start;
attribute DOMString type;
```

interface HTMLDLListElement : HTMLElement

```
attribute boolean compact;
```

interface HTMLDirectoryElement : HTMLElement

```
attribute boolean compact;
```

interface HTMLMenuElement : HTMLElement

```
attribute boolean compact;
```

interface HTMLLIElement : HTMLElement

```
attribute DOMString type;
attribute long value;
```

HTML ELEMENTS – PARAGRAPHS

interface HTMLParagraphElement : HTMLElement

```
attribute DOMString align;
```

HTML ELEMENTS – HEADINGS

interface HTMLHeadingElement : HTMLElement

```
attribute DOMString align;
```

HTML ELEMENTS – PREFORMATTED TEXT

interface HTMLPreElement : HTMLElement

```
attribute long width;
```

HTML ELEMENTS – INLINE ELEMENTS

interface HTMLQuoteElement : HTMLElement

```
attribute DOMString cite;
```

interface HTMLBRElement : HTMLElement

```
attribute DOMString clear;
```

interface HTMLBaseFontElement : HTMLElement

```
attribute DOMString color;
attribute DOMString face;
attribute long size;
```

interface HTMLFontElement : HTMLElement

```
attribute DOMString color;
attribute DOMString face;
attribute DOMString size;
```

interface HTMLModElement : HTMLElement

```
attribute DOMString cite;
attribute DOMString dateTime;
```

HTML ELEMENTS – HORIZONTAL RULE

interface HTMLHRElement : HTMLElement

```
attribute DOMString align;
attribute boolean noShade;
attribute DOMString size;
attribute DOMString width;
```

HTML ELEMENTS – ANCHOR ELEMENT

interface HTMLAnchorElement : HTMLElement

```
attribute DOMString accessKey;
attribute DOMString charset;
attribute DOMString coords;
attribute DOMString href;
attribute DOMString hreflang;
attribute DOMString name;
attribute DOMString rel;
attribute DOMString rev;
attribute DOMString shape;
attribute long tabIndex;
attribute DOMString target;
attribute DOMString type;
void blur();
void focus();
```



QUICK REFERENCE

Document Object Model (DOM) Level 2 HTML

W3C Recommendation
09-January-2003

<http://www.w3.org/TR/DOM-Level-2-HTML/>

Table of Contents:

- MISCELLANEOUS OBJECT DEFINITIONS
- OBJECTS RELATED TO HTML DOCUMENTS
- HTML ELEMENTS – DOCUMENT STRUCTURE
- HTML ELEMENTS – DOCUMENT HEAD
- HTML ELEMENTS – DOCUMENT BODY
- HTML ELEMENTS – FORMS
- HTML ELEMENTS – LISTS
- HTML ELEMENTS – PARAGRAPHS
- HTML ELEMENTS – HEADINGS
- HTML ELEMENTS – PREFORMATTED TEXT
- HTML ELEMENTS – INLINE ELEMENTS
- HTML ELEMENTS – HORIZONTAL RULE
- HTML ELEMENTS – ANCHOR ELEMENT
- HTML ELEMENTS – IMAGES
- HTML ELEMENTS – OBJECT ELEMENT
- HTML ELEMENTS – JAVA APPLETS
- HTML ELEMENTS – CLIENT SIDE IMAGES
- HTML ELEMENTS – TABLE
- HTML ELEMENTS – FRAMES
- XHTML AND THE HTML DOM

HTML ELEMENTS – IMAGES

interface HTMLImageElement : HTMLElement

```

attribute DOMString    name;
attribute DOMString    align;
attribute DOMString    alt;
attribute DOMString    border;
attribute long         height;
attribute long         hspace;
attribute boolean      isMap;
attribute DOMString    longDesc;
attribute DOMString    src;
attribute DOMString    useMap;
attribute long         vspace;
attribute long         width;

```

HTML ELEMENTS – OBJECT ELEMENT

interface HTMLObjectElement : HTMLElement

```

readonly attribute HTMLFormElement form;
attribute DOMString    code;
attribute DOMString    align;
attribute DOMString    archive;
attribute DOMString    border;
attribute DOMString    codeBase;
attribute DOMString    codeType;
attribute DOMString    data;
attribute boolean      declare;
attribute DOMString    height;
attribute long         hspace;
attribute DOMString    name;
attribute DOMString    standby;
attribute long         tabIndex;
attribute DOMString    type;
attribute DOMString    useMap;
attribute long         vspace;
attribute DOMString    width;

readonly attribute Document
contentDocument;

```

interface HTMLParamElement : HTMLElement

```

attribute DOMString    name;
attribute DOMString    type;
attribute DOMString    value;
attribute DOMString    valueType;

```

HTML ELEMENTS – JAVA APPLETS

interface HTMLAppletElement : HTMLElement

```

attribute DOMString    align;
attribute DOMString    alt;
attribute DOMString    archive;
attribute DOMString    code;
attribute DOMString    codeBase;
attribute DOMString    height;
attribute long         hspace;
attribute DOMString    name;
attribute DOMString    object;
attribute long         vspace;
attribute DOMString    width;

```

HTML ELEMENTS – CLIENT SIDE IMAGES

interface HTMLMapElement : HTMLElement

```

readonly attribute HTMLCollection areas;
attribute DOMString    name;

```

interface HTMLAreaElement : HTMLElement

```

attribute DOMString    accessKey;
attribute DOMString    alt;
attribute DOMString    coords;
attribute DOMString    href;
attribute boolean      noHref;
attribute DOMString    shape;
attribute long         tabIndex;
attribute DOMString    target;

```

HTML ELEMENTS – TABLE

interface HTMLTableElement : HTMLElement

```

attribute HTMLTableCaptionElement
caption;

setting raises(DOMException) on

attribute HTMLTableSectionElement tHead;
setting raises(DOMException) on

attribute HTMLTableSectionElement tFoot;
setting raises(DOMException) on

readonly attribute HTMLCollection rows;
readonly attribute HTMLCollection tBodies;
attribute DOMString    align;
attribute DOMString    bgColor;
attribute DOMString    border;
attribute DOMString    cellPadding;
attribute DOMString    cellSpacing;
attribute DOMString    frame;
attribute DOMString    rules;
attribute DOMString    summary;
attribute DOMString    width;

HTMLElement
void createTHead();
void deleteTHead();
HTMLElement
void createTFoot();
void deleteTFoot();
HTMLElement
void createCaption();
void deleteCaption();
HTMLElement
void insertRow(in long index)
raises(DOMException);

void deleteRow(in long index)
raises(DOMException);

```

interface HTMLTableCaptionElement : HTMLElement

```

attribute DOMString    align;

```

interface HTMLTableColElement : HTMLElement

```

attribute DOMString    align;
attribute DOMString    ch;
attribute DOMString    chOff;
attribute long         span;
attribute DOMString    vAlign;
attribute DOMString    width;

```

interface HTMLTableSectionElement : HTMLElement

```

attribute DOMString    align;
attribute DOMString    ch;
attribute DOMString    chOff;
attribute DOMString    vAlign;

readonly attribute HTMLCollection rows;
HTMLElement
void insertRow(in long index)
raises(DOMException);

void deleteRow(in long index)
raises(DOMException);

```

interface HTMLTableRowElement : HTMLElement

```

readonly attribute long    rowIndex;
readonly attribute long    sectionRowIndex;

readonly attribute HTMLCollection cells;
attribute DOMString    align;
attribute DOMString    bgColor;
attribute DOMString    ch;
attribute DOMString    chOff;
attribute DOMString    vAlign;

HTMLElement
void insertCell(in long index)
raises(DOMException);

void deleteCell(in long index)
raises(DOMException);

```

interface HTMLTableCellElement : HTMLElement

```

readonly attribute long    cellIndex;
attribute DOMString    abbr;
attribute DOMString    align;
attribute DOMString    axis;

```

```

attribute DOMString    bgColor;
attribute DOMString    ch;
attribute DOMString    chOff;
attribute long         colSpan;
attribute DOMString    headers;
attribute DOMString    height;
attribute boolean      noWrap;
attribute long         rowSpan;
attribute DOMString    scope;
attribute DOMString    vAlign;
attribute DOMString    width;

```

HTML ELEMENTS – FRAMES

interface HTMLFrameSetElement : HTMLElement

```

attribute DOMString    cols;
attribute DOMString    rows;

```

interface HTMLFrameElement : HTMLElement

```

attribute DOMString    frameBorder;
attribute DOMString    longDesc;
attribute DOMString    marginHeight;
attribute DOMString    marginWidth;
attribute DOMString    name;
attribute boolean      noResize;
attribute DOMString    scrolling;
attribute DOMString    src;

readonly attribute Document
contentDocument;

```

interface HTMLIFrameElement : HTMLElement

```

attribute DOMString    align;
attribute DOMString    frameBorder;
attribute DOMString    height;
attribute DOMString    longDesc;
attribute DOMString    marginHeight;
attribute DOMString    marginWidth;
attribute DOMString    name;
attribute DOMString    scrolling;
attribute DOMString    src;
attribute DOMString    width;

readonly attribute Document
contentDocument;

```

XHTML AND THE HTML DOM

The DOM HTML Level 1 API were originally intended to be used only for HTML 4.01 documents. The APIs were defined well before XHTML 1.0 became a specification, or before it was worked on by the HTML Working Group.

From the DOM point of view, The biggest difference between HTML 4.01 (and earlier) and XHTML 1.0 is that XHTML is case sensitive, whereas HTML 4.01 is case insensitive. The HTML case insensitivity is also reflected in the DOM HTML API. For instance, element and attribute names are exposed as all uppercase (for consistency) when used on an HTML document, regardless of the character case used in the markup. Since XHTML is based on XML, in XHTML everything is case sensitive, and element and attribute names must be lowercase in the markup.

Developers need to take two things into account when writing code that works on both HTML and XHTML documents. When comparing element or attribute names to strings, the string compare needs to be case insensitive, or the element or attribute name needs to be converted into lowercase before comparing against a lowercase string. Second, when calling methods that are case insensitive when used on a HTML document (such as `getElementByTagName()` and `documentItem()`), the string that is passed in should be lowercase.

Note: The interfaces provided in this document are only for [HTML 4.01] and [XHTML 1.0] documents and are not guaranteed to work with any future version of XHTML.

(<http://www.w3.org/TR/DOM-Level-2-HTML>, §1.3)