**Applet Tag**

Required elements are in **bold**. Optional elements are in regular typeface. Elements your specify are in *italics*.

**<APPLET**

CODEBASE = *codebaseURL*

ARCHIVE = *archiveList*

**CODE = *appletFile*** ...or...

**OBJECT = *serializedApplet***

ALT = *alternateText*

NAME = *appletInstanceName*

**WIDTH = *pixels*** **HEIGHT =**

***pixels***

ALIGN = *alignment*

VSPACE = *pixels* HSPACE = *pixels*

**>**

<PARAM NAME = *appletAttribute1* VALUE = *value />*

*<PARAM NAME = appletAttribute2 VALUE = value />*

*. . .*

*alternateHTML*

***</APPLET>***

*CODE, CODEBASE, and so on are attributes of the applet tag; they give the browser information about the applet. The only mandatory attributes are CODE, WIDTH, and HEIGHT. Each attribute is described below.*

*CODEBASE = codebaseURL*

*This OPTIONAL attribute specifies the base URL of the applet--the directory that contains the applet's code. If this attribute is not specified, then the document's URL is used.*

*ARCHIVE = archiveList*

*This OPTIONAL attribute describes one or more archives containing classes and other resources that will be "preloaded". The classes are loaded using an instance of an AppletClassLoader with the given CODEBASE. The archives in archiveList are separated by ",". NB: in Java Development Kit (JDKTM) 1.1, multiple APPLET tags with the same CODEBASE share the same instance of a ClassLoader. This is used by some client code to implement inter-applet communication. Future JDK releases \*may\* provide other mechanisms for inter-applet communication.*

***CODE = appletFile***

*This REQUIRED attribute gives the name of the file that contains the applet's compiled Applet subclass. This file is relative to the base URL of the applet. It cannot be absolute. One of CODE or OBJECT must be present.*

***OBJECT = serializedApplet***

*This attribute gives the name of the file that contains a serialized representation of an Applet. The Applet will be deserialized. The init() method will \*not\* be invoked; but its start() method will. Attributes valid when the original object was serialized are \*not\* restored. Any attributes passed to this APPLET instance will be available to the Applet; we advocate very strong restraint in using this feature. An applet should be stopped before it is serialized. One of CODE or OBJECT must be present.*

*ALT = alternateText*

*This OPTIONAL attribute specifies any text that should be displayed if the browser understands the APPLET tag but can't run Java technology-based applets.*

*NAME = appletInstanceName*

*This OPTIONAL attribute specifies a name for the applet instance, which makes it possible for applets on the same page to find (and communicate with) each other.*

***WIDTH = pixels HEIGHT = pixels***

*These REQUIRED attributes give the initial width and height (in pixels) of the applet display area, not counting any windows or dialogs that the applet brings up.*

*ALIGN = alignment*

*This OPTIONAL attribute specifies the alignment of the applet. The possible values of this attribute are the same as those for the IMG tag: left, right, top, texttop, middle, absmiddle, baseline, bottom, absbottom.*

*VSPACE = pixels HSPACE = pixels*

*These OPTIONAL attributes specify the number of pixels above and below the applet (VSPACE) and on each side of the applet (HSPACE). They're treated the same way as the IMG tag's VSPACE and HSPACE attributes.*

*<PARAM NAME = appletAttribute1 VALUE = value />   
<PARAM NAME = appletAttribute2 VALUE = value /> . . .*

*This tag is the only way to specify an applet-specific attribute. Applets access their attributes with the getParameter() method.*