

Microsoft AJAX Library: Boolean Type Extensions

Boolean.parse (value) S

Converts a string representation of a logical value to its `Boolean` object equivalent. The `value` argument must be a string representation of a `Boolean` value containing either "true" or "false" (case insensitive). The string can contain white space.

```
var b = Boolean.parse("true");
```

Microsoft AJAX Library: Date Type Extensions

Date.format (format)

Formats a date using the invariant (culture-independent) culture.

Remarks

The invariant culture is culture-insensitive. It is associated with the English language but not with any country or region. If a security decision depends on a string comparison or a case-change operation, use the `Date.format` method. This makes sure that the behavior will be consistent regardless of the culture settings of the system.

The invariant culture must be used only by processes that require culture-independent results, such as system services. Otherwise, the method can produce results that might be linguistically incorrect or culturally inappropriate.

```
var d = new Date();  
Sys.Debug.trace (d.format("dddd, dd MMMM yyyy HH:mm:ss"));
```

Date.localeFormat (format)

Formats a date using the current culture.

Remarks

The `format` parameter determines how the date will be presented. The `localeFormat` method provides the date based on a specific culture value (locale). The culture value is also used to display web information for specific language and country combinations.

This function uses the `Sys.CultureInfo.CurrentCulture` property to determine the culture value.

```
var d = new Date();  
Sys.Debug.trace (d.localeFormat("dddd, dd MMMM yyyy HH:mm:ss"));
```

Date.parseLocale (value, formats) S

Creates a date from a locale-specific string. `value` is a locale-specific string that can be parsed as a date, and `formats` (optional) is an array of custom formats. Returns an object of type `Date`.

Remarks

This function uses the `Sys.CultureInfo.CurrentCulture` property to determine the culture value.

```
Sys.Debug.trace (Date.parseLocale ("4/10/2001", "yyyy-MM-dd",  
"MM/dd/yyyy"));  
// Date.parseLocale will skip the first format here as invalid and use  
// the second one. If it does not find an appropriate format,  
// the function throws an exception.
```

Date.parseInvariant (value, formats) S

Creates a date from a string. `value` is a locale-specific string that can be parsed as a date, and `formats` (optional) is an array of custom formats. Returns an object of type `Date`.

```
Sys.Debug.trace (Date.parseInvariant ("4/10/2001", "yyyy-MM-dd",  
"MM/dd/yyyy"));  
// Date.parseInvariant will skip the first format here as invalid and use  
// the second one. If it does not find an appropriate format,  
// the function throws an exception.
```

Supported formats

Below are examples of supported formats to use with `Date.format` and `Date.localeFormat` (only invariant culture shown):

| Format | Formatted date |
|-----------------|---|
| d | Short date pattern (e.g.: 02/17/2007) |
| D | Long date pattern (e.g: Saturday, 17 February 2007) |
| t | Short time pattern (e.g.: 22:10) |
| T | Long time pattern (e.g.: 22:10:30) |
| F | Full date pattern (e.g.: Saturday, 17 February 2007 22:10:30) |
| m (or M) | Month and day pattern (e.g.: February 17) |
| s | Sortable date and time pattern (e.g.: 2007-02-17T22:10:30) |
| y (or Y) | Year and month pattern (e.g.: 2007 February) |