

# XPath、XQuery 以及 XSLT 函数函数参考手册

下面的参考手册定义了XPath 2.0 , XQuery 1.0和XSLT 2.0中的函数。

## 函数参考手册

- [存取](#)
- [AnyURI](#)
- [节点](#)
- [错误和跟踪](#)
- [逻辑](#)
- [序列](#)
- [数值](#)
- [持续时间/日期/时间](#)
- [Context](#)
- [字符串](#)
- [QName](#)

- 💡 函数命名空间的默认前缀为 fn:
- 💡 函数命名空间的 URI为 : http://www.w3.org/2005/xpath-functions

### 存取函数

| 名称  | 说明                          |
|---|-----------------------------|
| fn:node-name(node)  | 返回参数节点的节点名称。                |
| fn:nilled(node)   | 返回是否拒绝参数节点的布尔值。             |
| fn:data(item.item,...)                                    | 接受项目序列，并返回原子值序列。            |
| <div>● fn:base-uri()</div> <div>● fn:base-uri(node)</div> | 返回当前节点或指定节点的 base-uri 属性的值。 |
| fn:document-uri(node)                                     | 返回指定节点的 document-uri 属性的值。  |

### 错误和跟踪函数

| 名称  | 说明  |
|---|---|
| <div>● fn:error()</div> <div>● fn:error(error)</div> <div>● fn:error(error,description)</div> <div>● fn:error(error,description,error-object)</div> | <div>例子 : error(fn:QName('http://example.com/test', 'err:toohigh'), 'Error: Price is too high')</div> <div>结果 : 向外部处理环境返回 http://example.com/test#toohigh 以及字符串 "Error: Price is too high".</div> |
| fn:trace(value,label)   | 用于对查询进行 debug。  |

## 有关数值的函数

| 名称                      | 说明   |
|-------------------------|--|
| fn:number(arg)          | 返回参数的数值。参数可以是布尔值、字符串或节点集。<br>例子：number('100')<br>结果：100  |
| fn:abs(num)             | 返回参数的绝对值。<br>例子：abs(3.14)<br>结果：3.14<br>例子：abs(-3.14)<br>结果：3.14   |
| fn:ceiling(num)         | 返回大于 num 参数的最小整数。<br>例子：ceiling(3.14)<br>结果：4  |
| fn:floor(num)           | 返回不大于 num 参数的最大整数。<br>例子：floor(3.14)<br>结果：3   |
| fn:round(num)           | 把 num 参数舍入为最接近的整数。<br>例子：round(3.14)<br>结果：3   |
| fn:round-half-to-even() | 例子：round-half-to-even(0.5)<br>结果：0<br>例子：round-half-to-even(1.5)<br>结果：2<br>例子：round-half-to-even(2.5)<br>结果：2 |

## 有关字符串的函数

| 名称                                   | 说明   |
|--------------------------------------|--|
| fn:string(arg)                       | 返回参数的字符串值。参数可以是数字、逻辑值或节点集。<br>例子：string(314)<br>结果："314"                   |
| fn:codepoints-to-string(int,int,...) | 根据代码点序列返回字符串。<br>例子：codepoints-to-string(84, 104, 233, 114, 232, 115, 101) |

|  |   |
|--|---|
|  | 结果：'Thérèse'  |
| fn:string-to-codepoints(string)  | <p>根据字符串返回代码点序列。</p> <p>例子：string-to-codepoints("Thérèse")</p> <p>结果：84, 104, 233, 114, 232, 115, 101</p>   |
| fn:codepoint-equal(comp1,comp2)  | <p>根据 Unicode 代码点对照，如果 comp1 的值等于 comp2 的值，则返回 true。(http://www.w3.org/2005/02/xpath-functions/collation/codepoint)，否则返回 false。</p>   |
| <ul style="list-style-type: none"> <li>fn:compare(comp1,comp2)</li> <li>fn:compare(comp1,comp2,collation)</li> </ul> | <p>如果 comp1 小于 comp2，则返回 -1。如果 comp1 等于 comp2，则返回 0。如果 comp1 大于 comp2，则返回 1。（根据所用的对照规则）。</p> <p>例子：compare('ghi', 'ghi')</p> <p>结果：0</p>  |
| fn:concat(string,string,...)   | <p>返回字符串的拼接。</p> <p>例子：concat('XPath ','is ','FUN!')</p> <p>结果：'XPath is FUN!'</p>  |
| fn:string-join((string,string,...),sep)  | <p>使用 sep 参数作为分隔符，来返回 string 参数拼接后的字符串。</p> <p>例子：string-join(('We', 'are', 'having', 'fun!'), ' ')</p> <p>结果：' We are having fun! '</p> <p>例子：string-join(('We', 'are', 'having', 'fun!'))</p> <p>结果：'Wearehavingfun!'</p> <p>例子：string-join((), 'sep')</p> <p>结果：''</p> |
| <ul style="list-style-type: none"> <li>fn:substring(string,start,len)</li> <li>fn:substring(string,start)</li> </ul> | <p>返回从 start 位置开始的指定长度的子字符串。第一个字符的下标是 1。如果省略 len 参数，则返回从位置 start 到字符串末尾的子字符串。</p> <p>例子：substring('Beatles',1,4)</p> <p>结果：'Beat'</p> <p>例子：substring('Beatles',2)</p> <p>结果：'eatles'</p>   |
| <ul style="list-style-type: none"> <li>fn:string-length(string)</li> <li>fn:string-length()</li> </ul>               | <p>返回指定字符串的长度。如果没有 string 参数，则返回当前节点的字符串值的长度。</p> <p>例子：string-length('Beatles')</p> <p>结果：7</p>  |
| <ul style="list-style-type: none"> <li>fn:normalize-space(string)</li> </ul>   | <p>删除指定字符串的开头和结尾的空白，并把内部的所有空白序列替换为一个，然后返回结果。如果没有 string 参数，则处理当前节点。</p>   |

|                                       |   |
|---------------------------------------|---|
| ● fn:normalize-space()                | 例子：normalize-space(' The XML ')<br>结果：'The XML'   |
| fn:normalize-unicode()                | 执行 Unicode 规格化。   |
| fn:upper-case(string)                 | 把 string 参数转换为大写。<br>例子：upper-case('The XML')<br>结果：'THE XML'   |
| fn:lower-case(string)                 | 把 string 参数转换为小写。<br>例子：lower-case('The XML')<br>结果：'the xml'   |
| fn:translate(string1,string2,string3) | 把 string1 中的 string2 替换为 string3。<br>例子：translate('12:30','30','45')<br>结果：'12:45'<br>例子：translate('12:30','03','54')<br>结果：'12:45'<br>例子：translate('12:30','0123','abcd')<br>结果：'bc:da'  |
| fn:escape-uri(stringURI,esc-res)      | 例子：escape-uri("http://example.com/test#car", true())<br>结果："http%3A%2F%2Fexample.com%2Ftest#car"<br>例子：escape-uri("http://example.com/test#car", false())<br>结果："http://example.com/test#car"<br>例子：escape-uri ("http://example.com/~bébé", false())<br>结果："http://example.com/~b%C3%A9b%C3%A9" |
| fn:contains(string1,string2)          | 如果 string1 包含 string2，则返回 true，否则返回 false。<br>例子：contains('XML','XM')<br>结果：true  |
| fn:starts-with(string1,string2)       | 如果 string1 以 string2 开始，则返回 true，否则返回 false。<br>例子：starts-with('XML','X')<br>结果：true  |
| fn:ends-with(string1,string2)         | 如果 string1 以 string2 结尾，则返回 true，否则返回 false。<br>例子：ends-with('XML','X')<br>结果：false   |
| fn:substring-before(string1,string2)  | 返回 string2 在 string1 中出现之前的子字符串。  |

|                                     |  |
|-------------------------------------|--|
|                                     | XPath、XQuery 以及 XSLT 函数   菜鸟教程<br>例子：substring-before('12/10','/')<br>结果：'12'  |
| fn:substring-after(string1,string2) | 返回 string2 在 string1 中出现之后的子字符串。<br>例子：substring-after('12/10','/')<br>结果：'10'   |
| fn:matches(string,pattern)          | 如果 string 参数匹配指定的模式，则返回 true，否则返回 false。<br>例子：matches("Merano", "ran")<br>结果：true   |
| fn:replace(string,pattern,replace)  | 把指定的模式替换为 replace 参数，并返回结果。<br>例子：replace("Bella Italia", "l", "**")<br>结果：'Be**a Ita*ia'<br>例子：replace("Bella Italia", "l", "")<br>结果：'Bea Itaia' |
| fn:tokenize(string,pattern)         | 例子：tokenize("XPath is fun", "\s+")<br>结果：("XPath", "is", "fun")  |

## 针对 anyURI 的函数

| 名称                            | 说明 |
|-------------------------------|----|
| fn:resolve-uri(relative,base) |    |

## 关于布尔值的函数

| 名称              | 说明  |
|-----------------|---|
| fn:boolean(arg) | 返回数字、字符串或节点集的布尔值。   |
| fn:not(arg)     | 首先通过 boolean() 函数把参数还原为一个布尔值。如果该布尔值为 false，则返回 true，否则返回 false。<br>例子：not(true())<br>结果：false |
| fn:true()       | 返回布尔值 true。<br>例子：true()<br>结果：true   |
| fn:false()      | 返回布尔值 false。<br>例子：false()<br>结果：false  |

有关持续时间、日期和时间的函数

日期、时间、持续时间的组件提取函数

| 名称                                    | 说明  |
|---------------------------------------|---|
| fn:dateTime(date,time)                | 把参数转换为日期和时间。  |
| fn:years-from-duration(datetimedur)   | 返回参数值的年份部分的整数，以标准词汇表示法来表示。  |
| fn:months-from-duration(datetimedur)  | 返回参数值的月份部分的整数，以标准词汇表示法来表示。  |
| fn:days-from-duration(datetimedur)    | 返回参数值的天部分的整数，以标准词汇表示法来表示。   |
| fn:hours-from-duration(datetimedur)   | 返回参数值的小时部分的整数，以标准词汇表示法来表示。  |
| fn:minutes-from-duration(datetimedur) | 返回参数值的分钟部分的整数，以标准词汇表示法来表示。  |
| fn:seconds-from-duration(datetimedur) | 返回参数值的分钟部分的十进制数，以标准词汇表示法来表示。  |
| fn:year-from-dateTime(datetime)       | 返回参数本地值的年部分的整数。<br>例子：year-from-dateTime(xs:dateTime("2005-01-10T12:30-04:10"))<br>结果：2005      |
| fn:month-from-dateTime(datetime)      | 返回参数本地值的月部分的整数。<br>例子：month-from-dateTime(xs:dateTime("2005-01-10T12:30-04:10"))<br>结果：01       |
| fn:day-from-dateTime(datetime)        | 返回参数本地值的天部分的整数。<br>例子：day-from-dateTime(xs:dateTime("2005-01-10T12:30-04:10"))<br>结果：10         |
| fn:hours-from-dateTime(datetime)      | 返回参数本地值的小时部分的整数。<br>例子：hours-from-dateTime(xs:dateTime("2005-01-10T12:30-04:10"))<br>结果：12      |
| fn:minutes-from-dateTime(datetime)    | 返回参数本地值的分钟部分的整数。<br>例子：minutes-from-dateTime(xs:dateTime("2005-01-10T12:30-04:10"))<br>结果：30    |
| fn:seconds-from-dateTime(datetime)    | 返回参数本地值的秒部分的十进制数。<br>例子：seconds-from-dateTime(xs:dateTime("2005-01-10T12:30:00-04:10"))<br>结果：0 |
|                                       |   |

|   |  |
|---|--|
| fn:timezone-from-dateTime(datetime)               | 返回参数的时区部分，如果存在。  |
| fn:year-from-date(date)                           | 返回参数本地值中表示年的整数。<br>例子：year-from-date(xs:date("2005-04-23"))<br>结果：2005   |
| fn:month-from-date(date)                          | 返回参数本地值中表示月的整数。<br>例子：month-from-date(xs:date("2005-04-23"))<br>结果：4     |
| fn:day-from-date(date)                            | 返回参数本地值中表示天的整数。<br>例子：day-from-date(xs:date("2005-04-23"))<br>结果：23      |
| fn:timezone-from-date(date)                       | 返回参数的时区部分，如果存在。  |
| fn:hours-from-time(time)                          | 返回参数本地值中表示小时部分的整数。<br>例子：hours-from-time(xs:time("10:22:00"))<br>结果：10   |
| fn:minutes-from-time(time)                        | 返回参数本地值中表示分钟部分的整数。<br>例子：minutes-from-time(xs:time("10:22:00"))<br>结果：22 |
| fn:seconds-from-time(time)                        | 返回参数本地值中表示秒部分的整数。<br>例子：seconds-from-time(xs:time("10:22:00"))<br>结果：0   |
| fn:timezone-from-time(time)                       | 返回参数的时区部分，如果存在。  |
| fn:adjust-dateTime-to-timezone(datetime,timezone) | 如果 timezone 参数为空，则返回没有时区的 dateTime。否则返回带有时区的 dateTime。                   |
| fn:adjust-date-to-timezone(date,timezone)         | 如果 timezone 参数为空，则返回没有时区的 date。否则返回带有时区的 date。                           |
| fn:adjust-time-to-timezone(time,timezone)         | 如果 timezone 参数为空，则返回没有时区的 time。否则返回带有时区的 time。                           |

## 与 QNames 相关的函数

| 名称         | 说明 |
|------------|----|
| fn:QName() |    |

|                               |  |
|-------------------------------|--|
| fn:local-name-from-QName()    |  |
| fn:namespace-uri-from-QName() |  |
| fn:namespace-uri-for-prefix() |  |
| fn:in-scope-prefixes()        |  |
| fn:resolve-QName()            |  |

关于节点的函数

| 名称   | 说明  |
|--|---|
| <ul style="list-style-type: none"><li>fn:name()</li><li>fn:name(nodeset)</li></ul>                   | 返回当前节点的名称或指定节点集中的第一个节点。   |
| <ul style="list-style-type: none"><li>fn:local-name()</li><li>fn:local-name(nodeset)</li></ul>       | 返回当前节点的名称或指定节点集中的第一个节点 - 不带有命名空间前缀。   |
| <ul style="list-style-type: none"><li>fn:namespace-uri()</li><li>fn:namespace-uri(nodeset)</li></ul> | 返回当前节点或指定节点集中第一个节点的命名空间 URI。  |
| fn:lang(lang)  | 如果当前节点的语言匹配指定的语言，则返回 true。<br>例子：Lang("en") is true for <p xml:lang="en">...</p><br>例子：Lang("de") is false for <p xml:lang="en">...</p> |
| <ul style="list-style-type: none"><li>fn:root()</li><li>fn:root(node)</li></ul>                      | 返回当前节点或指定的节点所属的节点树的根节点。通常是文档节点。   |

有关序列的函数

一般性的函数

| 名称                                      | 说明  |
|---|---|
| fn:index-of((item,item,...),searchitem) | 返回在项目序列中等于 searchitem 参数的位置。<br>例子：index-of ((15, 40, 25, 40, 10), 40)<br>结果：(2, 4)<br>例子：index-of (("a", "dog", "and", "a", "duck"), "a")<br>Result (1, 4) |



|   |   |
|---|---|
|   | <p>例子：index-of ((15, 40, 25, 40, 10), 18)</p> <p>结果：()</p>  |
| fn:remove((item,item,...),position)           | <p>返回由 item 参数构造的新序列 - 同时删除 position 参数指定的项目。</p> <p>例子：remove(("ab", "cd", "ef"), 0)</p> <p>结果：("ab", "cd", "ef")</p> <p>例子：remove(("ab", "cd", "ef"), 1)</p> <p>结果：("cd", "ef")</p> <p>例子：remove(("ab", "cd", "ef"), 4)</p> <p>结果：("ab", "cd", "ef")</p>  |
| fn:empty(item,item,...)                       | <p>如果参数值是空序列，则返回 true，否则返回 false。</p> <p>例子：empty(remove(("ab", "cd"), 1))</p> <p>结果：false</p>  |
| fn:exists(item,item,...)                      | <p>如果参数值不是空序列，则返回 true，否则返回 false。</p> <p>例子：exists(remove(("ab"), 1))</p> <p>结果：false</p>  |
| fn:distinct-values((item,item,...),collation) | <p>返回唯一不同的值。</p> <p>例子：distinct-values((1, 2, 3, 1, 2))</p> <p>结果：(1, 2, 3)</p>   |
| fn:insert-before((item,item,...),pos,inserts) | <p>返回由 item 参数构造的新序列 - 同时在 pos 参数指定位置插入 inserts 参数的值。</p> <p>例子：insert-before(("ab", "cd"), 0, "gh")</p> <p>结果：("gh", "ab", "cd")</p> <p>例子：insert-before(("ab", "cd"), 1, "gh")</p> <p>结果：("gh", "ab", "cd")</p> <p>例子：insert-before(("ab", "cd"), 2, "gh")</p> <p>结果：("ab", "gh", "cd")</p> <p>例子：insert-before(("ab", "cd"), 5, "gh")</p> <p>结果：("ab", "cd", "gh")</p> |
| fn:reverse((item,item,...))                   | <p>返回指定的项目的颠倒顺序。</p> <p>例子：reverse(("ab", "cd", "ef"))</p> <p>结果：("ef", "cd", "ab")</p> <p>例子：reverse(("ab"))</p> <p>结果：("ab")</p>  |

|   |   |
|---|---|
| fn:subsequence((item,item,...),start,len) | 返回 start 参数指定的位置返回项目序列，序列的长度由 len 参数指定。<br>第一个项目的位置是 1。<br>例子：subsequence((\$item1, \$item2, \$item3,...), 3)<br>结果：(\$item3, ...)<br>例子：subsequence((\$item1, \$item2, \$item3, ...), 2, 2)<br>结果：(\$item2, \$item3) |
| fn:unordered((item,item,...))             | 依据实现决定的顺序来返回项目。   |

## 测试序列容量的函数

| 名称                            | 说明                          |
|-------------------------------|-----------------------------|
| fn:zero-or-one(item,item,...) | 如果参数包含零个或一个项目，则返回参数，否则生成错误。 |
| fn:one-or-more(item,item,...) | 如果参数包含一个或多个项目，则返回参数，否则生成错误。 |
| fn:exactly-one(item,item,...) | 如果参数包含一个项目，则返回参数，否则生成错误。    |

## Equals, Union, Intersection and Except

| 名称                                     | 说明  |
|--|---|
| fn:deep-equal(param1,param2,collation) | 如果 param1 和 param2 与彼此相等（deep-equal），则返回 true，否则返回 false。 |

## 合计函数

| 名称                        | 说明  |
|---------------------------|---|
| fn:count((item,item,...)) | 返回节点的数量。  |
| fn:avg((arg,arg,...))     | 返回参数值的平均数。 例子：avg((1,2,3)) 结果：2                             |
| fn:max((arg,arg,...))     | 返回大于其它参数的参数。 例子：max((1,2,3)) 结果：3 例子：max(('a', 'k')) 结果：'k' |
| fn:min((arg,arg,...))     | 返回小于其它参数的参数。 例子：min((1,2,3)) 结果：1 例子：min(('a', 'k')) 结果：'a' |
| fn:sum(arg,arg,...)       | 返回指定节点集中每个节点的数值的总和。   |

## 生成序列的函数

| 名称                              | 说明   |
|---------------------------------|--|
| fn:id((string,string,...),node) | Returns a sequence of element nodes that have an ID value equal to the value of one or more of the values specified in the string argument |

|   |  |
|---|--|
| fn:idref((string,string,...),node)  | Returns a sequence of element or attribute nodes that have an IDREF value equal to the value of one or more of the values specified in the string argument |
| fn:doc(URI)   |  |
| fn:doc-available(URI)   | 如果 doc() 函数返回文档节点，则返回 true，否则返回 false。   |
| <ul style="list-style-type: none"><li>fn:collection()</li><li>fn:collection(string)</li></ul> |  |

上下文函数

| 名称                     | 说明  |
|------------------------|---|
| fn:position()          | 返回当前正在被处理的节点的 index 位置。<br>例子：//book[position()<=3]<br>结果：选择前三个 book 元素 |
| fn:last()              | 返回在被处理的节点列表中的项目数目。<br>例子：//book[last()]<br>结果：选择最后一个 book 元素            |
| fn:current-dateTime()  | 返回当前的 dateTime（带有时区）。   |
| fn:current-date()      | 返回当前的日期（带有时区）。  |
| fn:current-time()      | 返回当前的时间（带有时区）。  |
| fn:implicit-timezone() | 返回隐式时区的值。   |
| fn:default-collation() | 返回默认对照的值。   |
| fn:static-base-uri()   | 返回 base-uri 的值。   |

← XPath 总结

点我分享笔记

