

# URI Library

The URI Library provides handlers that help you retrieve information from your URI strings. If you use URI routing, you can also retrieve information about the re-routed segments.

**Note:** This library is initialized automatically by the system so there is no need to do it manually.

## rigFetchSegment(n)

Permits you to retrieve a specific segment. Where **n** is the segment number you wish to retrieve. Segments are numbered from left to right. For example, if your full URL is this:

```
http://example.com/index.lc/news/local/metro/crime_is_up
```

The segment numbers would be this:

- news
- local
- metro
- crime\_is\_up

By default the function returns FALSE (boolean) if the segment does not exist. There is an optional second parameter that permits you to set your own default value if the segment is missing. For example, this would tell the function to return the number zero in the event of failure:

```
put rigFetchSegment(3, 0) into tProductID
```

It helps avoid having to write code like this:

```
if rigFetchSegment(3) is FALSE then
    put 0 into tProductID
else
    put rigFetchSegment(3) into tProductID
end if
```

## rigFetchRsegment(n)

This function is identical to the previous one, except that it lets you retrieve a specific segment from your re-routed URI in the event you are using revlgniter's [URI Routing](#) feature.

## rigSlashSegment(n)

This function is almost identical to **rigFetchSegment()**, except it adds a trailing and/or leading slash based on the second parameter. If the parameter is not used, a trailing slash added. Examples:

```
get rigSlashSegment(3)
get rigSlashSegment(3, "leading")
get rigSlashSegment(3, "both")
```

Returns:

- segment/
- /segment
- /segment/

## rigSlashRsegment(n)

This function is identical to the previous one, except that it lets you add slashes to a specific segment from your re-routed URI in the event you are using revlgniter's [URI Routing](#) feature.

## rigUriToAssoc(n)

This function lets you turn URI segments into an associative array of key/value pairs. Consider this URI:

```
index.lc/user/search/name/joe/location/UK/gender/male
```

Using this function you can turn the URI into an associative array with this prototype:

```
tArray["name"] -- "joe"
tArray["location"] -- "UK"
tArray["gender"] -- "male"
```

The first parameter of the function lets you set an offset. By default it is set to **3** since your URI will normally contain a controller/handler in the first and second segments. Example:

```
put rigUriToAssoc(3) into tArray

put tArray["name"] into gData["name"]
```

The second parameter lets you set default key names, so that the array returned by the function will always contain expected indexes, even if missing from the URI. Example:

```
put "name,gender,location,type,sort" into tDefault
split tDefault using comma

put rigUriToAssoc(3, tDefault) into tArray
```

If the URI does not contain a value in your default array, an array index will be set to that name, with a value of FALSE.

Lastly, if a corresponding value is not found for a given key (if there is an odd number of URI segments) the value will be set to FALSE (boolean).

## rigRuriToAssoc(n)

This function is identical to the previous one, except that it creates an associative array using the re-routed URI in the event you are using revlgniter's [URI Routing](#) feature.

## rigAssocToURI()

Takes an associative array as input and generates an URI string from it. The array keys will be included in the string. Example:

```
put "product=shoes,size=large,color=red" into tArray
split tArray using comma and "="

put rigAssocToURI(tArray) into tStr

-- Produces:  product/shoes/size/large/color/red
```

## rigUriString()

Returns a string with the complete URI. For example, if this is your full URL:

```
http://example.com/index.lc/news/local/345
```

The function would return this:

```
/news/local/345
```

## rigRuriString()

This function is identical to the previous one, except that it returns the re-routed URI in the event you are using revlgniter's [URI Routing](#) feature.

## rigTotalSegments()

Returns the total number of segments.

## rigTotalRsegments()

This function is identical to the previous one, except that it returns the total number of segments in your re-routed URI in the event you are using revlgniter's [URI Routing](#) feature.

## rigSegmentArray()

Returns an array containing the URI segments. For example:

```
put rigSegmentArray() into tSegs

repeat for each key tKey in tSegs
    put tSegs[tKey] into tSegment
    put tSegment & "<br />" after tSegments
end repeat
```

## rigRSegmentArray()

This function is identical to the previous one, except that it returns the array of segments in your re-routed URI in the event you are using revlgniter's [URI Routing](#) feature.