Rocket Uniface Library 10.4

Notices

Copyright

© 1996-2025 Rocket Software, Inc. or its affiliates. All Rights Reserved.

Trademarks

Rocket is a registered trademark of Rocket Software, Inc. For a list of Rocket registered trademarks go to: www.rocketsoftware.com/about/legal. All other products or services mentioned in this document may be covered by the trademarks, service marks, or product names of their respective owners.

Examples

This information might contain examples of data and reports. The examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

License agreement

This software and the associated documentation are proprietary and confidential to Rocket Software, Inc. or its affiliates, are furnished under license, and may be used and copied only in accordance with the terms of such license.

Note: This product may contain encryption technology. Many countries prohibit or restrict the use, import, or export of encryption technologies, and current use, import, and export regulations should be followed when exporting this product.

Corporate information

Rocket Software, Inc. develops enterprise infrastructure products in four key areas: storage, networks, and compliance; database servers and tools; business information and analytics; and application development, integration, and modernization.

Website: www.rocketsoftware.com

Rocket Global Headquarters

77 4th Avenue, Suite 100

Waltham, MA 02451-1468

USA

To contact Rocket Software by telephone for any reason, including obtaining pre-sales information and technical support, use one of the following telephone numbers.

Country and Toll-free telephone number

- United States: 1-855-577-4323
- Australia: 1-800-823-405
- Belgium: 0800-266-65
- Canada: 1-855-577-4323
- China: 400-120-9242
- France: 08-05-08-05-62
- Germany: 0800-180-0882
- Italy: 800-878-295
- Japan: 0800-170-5464
- Netherlands: 0-800-022-2961
- New Zealand: 0800-003210
- South Africa: 0-800-980-818
- United Kingdom: 0800-520-0439

Contacting Technical Support

The Rocket Community is the primary method of obtaining support. If you have current support and maintenance agreements with Rocket Software, you can access the Rocket Community and report a problem, download an update, or read answers to FAQs. To log in to the Rocket Community or to request a Rocket Community account, go to www.rocketsoftware.com/support. In addition to using the Rocket Community to obtain support, you can use one of the telephone numbers that are listed above or send an email to support@rocketsoftware.com.

Table of contents

for...endfor

Define a counter-based processing loop.

```
for Counter = StartValue
 to
 EndValue
  {
 step
 StepValue}
  Your ProcScript
endfor
```

Parameters

Table: Parameters

Parameter	Data Type	Description
Counter	Number	Current value of the counter
StartValue	Number	Initial value of the counter
EndValue	Number	End value of the counter
StepValue	Number	Step size by which the counter is incremented (or decremented); if not specified, the default value is 1.

Return Values

None

Use

Allowed in all component types.

Description

The `for` statement defines a counter-based processing loop, which initializes *Counter* with the *StartValue* and repeatedly executes the code in the block until one of the following conditions is met:

- *StepValue* >= 0 and *Counter* > *EndValue*
- *StepValue* < 0 and *Counter* < *EndValue*
- A `break` statement is encountered

After the loop completes (after the `endfor` statement), the *Counter* holds the value it had when the loop ended.

The *Counter*, *EndValue* and *StepValue* can be altered by the code in the loop, which enables you to conditionally make changes in the loop as it executes.

> ⚠️ **Note:** If the *EndValue* is not specified, the loop runs forever, so it must be broken by a `break`.

In the following example, the counter (`vCounter`) is decreased by 2 each time the `for` loop is executed. A `break` statement stops the loop after it has executed 6 times.

```
variables
 numeric vCounter
 numeric vLoops
endvariables

vLoops = 0
for vCounter = 100 to 0 step -2
 vLoops += 1
 putmess "Counter: %%vCounter, Loop count: %%vLoops "
 if (vLoops >= 6)
 putmess "Loop processing stopped"
 break
 endif
endfor
```

The resulting output looks like this:

```
Counter: 100, Loop count: 1
Counter: 98, Loop count: 2
Counter: 96, Loop count: 3
Counter: 94, Loop count: 4
Counter: 92, Loop count: 5
Counter: 90, Loop count: 6
Loop processing stopped
```

Related concepts

[forentity...endfor](#)

[forlist...endfor](#)

[forlist/id...endfor](#)