

Artículo

[Kurro Lopez](#) · Ago 31, 2021



Lectura de 2 min

[Open Exchange](#)

## FOREACH para ObjectScript

Como sabes, ObjectScript no tiene un comando o función de sistema FOREACH. Pero tiene un amplio margen para la creatividad.

Pero la tarea es recorrer una matriz global o local y hacer algo PARA CADA elemento.

Hay dos posibles soluciones:

- Crear una macro que genere las secuencias de código requeridas
- Crear un comando extendido para realizar la acción.

Ambas formas están presentes aquí.

La macro es una solución genérica y bastante flexible y fácil de adaptar si es necesario.

```
##;FOREACHMACRO ; macro definitions
##; %key = variable provide to loop trough array
##; %arr = the gobal or local array to be looped on
##; %the method or routine to be executed for each node.
##; $$$foreach = forward loop $$$foreachR = reverse loop
#define foreach(%key,%arr,%do) set %key="" for set %key=$o(%arr(%key)) q:%key="" do %do
#define foreachR(%key,%arr,%do) set %key="" for set %key=$o(%arr(%key),-1) q:%key="" do %do
```

Simplemente incluye la macro y aplícala.

E:jemplo

```
#include FOREACHMACRO
test $$$foreach(key,^rcc,show)
quit
show zwrite @$zr,! quit
```

La creación de una extensión de comando está disponible para todos los namespaces.

Debe incluirse en %ZLANGC00.mac por `#include ZZFOREACH` el código relacionado está aquí:

```
##; run $order() ascending or descending on global or local arrays
##; pass semicolon separated parameter string ("%array;%do;%fwd,;%key")
##; %array = global or local variable name
##; %do = routine or method to be executed for each run
##; %fwd = 1/-1 loop direction ascending / descending, default = 1
##; %key = first key if existing
ZZFOREACH(%par) public {
    set %par=$lfs(%par,";")
    new %array,%do,%fwd,%key,%val
```

```
set %array=$lg(%par,1),%do=$lg(%par,2),%fwd=$lg(%par,3),%key=$lg(%par,4)
if '%fwd set %fwd=1
if %key]" " set %key=$o(@%array@(%key), $s(%fwd<1:-1,1:1))
for set %key=$o(@%array@(%key),%fwd,%val) quit:%key="" do @%do
quit 1
}
```

Además de la macro, el comando permite opcionalmente ejecutar zzFOREACH desde un punto de inicio proporcionado hacia adelante o hacia atrás

Ejemplo:

```
DEMO>zzforeach "^rcc;show^dump(%array,%key,%val)"
```

```
^rcc(1) = 1
```

```
^rcc(2) = 2
```

```
^rcc(3) = 3
```

```
^rcc(4) = 4
```

```
^rcc(5) = 5
```

o del subíndice 3:

```
DEMO>zzforeach "^rcc;show^dump(%array,%key,%val);;3"
```

```
^rcc(3) = 3
```

```
^rcc(4) = 4
```

```
^rcc(5) = 5
```

O lo mismo inverso:

```
DEMO>zzforeach "^rcc;show^dump(%array,%key,%val);-1;3"
```

```
^rcc(3) = 3
```

```
^rcc(2) = 2
```

```
^rcc(1) = 1
```

Los archivos INC están disponibles en OEX con ZPM

[#Code Snippet](#) [#ObjectScript](#) [#Caché](#) [#Ensemble](#) [#InterSystems IRIS](#)  
[Compruebe la aplicación relacionada en InterSystems Open Exchange](#)

URL de fuente: <https://es.community.intersystems.com/post/foreach-para-objectscript>