Hash Tables

/hmake

Added in 1.9.0

/hmake [-s] <name> <N>

Creates a new hash table with N slots.

See also /hsave, /hload, /hfree, /hadd, /hdel, $hget, $hfind.

Switches

-s - Displays a creation result.

Parameters

<name> - The hashtable name to create.
<N> - The initial number of hashtable items.

Example

```
; call the setup once
; /example
;
; //echo -a $abbr(lol)
;
alias example {
  ;Create the table.
  /hmake abbr 1000

  ;Populate the table.
  /hadd abbr lol laughing out load
  /hadd abbr omg oh my gosh
  /hadd abbr lmao laughing my a?? off
  /hadd abbr brb be right back
}
; get the abbreviation
alias abbr return $hget(abbr, $1)
```

/hfree

Added in 1.9.0

/hfree [-sw] <name>

Deletes a hash table.

See also /hsave, /hsave, /hmake, /hadd, /hdel, $hget, $hfind.

Switches

-s - Displays the result.
-w - Indicates name is a wildcard.

Parameters

<name> - The hashtable to delete.

Example

```
alias example {
  ;Create the hash table 'hash'.
  /hmake hash

  ;Print the hashtable size.
  echo -a $hget(hash) : $hget(hash).size

  ;Creates the hash table 'h' with 10 items.
  /hmake h 10

  ;Creates the hash table 'has' with 20 items.
  /hmake has 20

  ;Delete hash table 'hash'.
  /hfree -s hash

  ;Print the hashtable size.
  /echo -a $hget(h) : $hget(h).size - $hget(has) : $hget(has).size

  ;Deletes any hash tables matching h*
  /hfree -sw h*
}
```

/hadd

Added in 1.9.0

/hadd [-bcmsuNz] <name> <item> [text | &binvar]

Adds an item to an existing hash table.

See also /hsave, /hload, /hmake, /hfree, /hdel, $hget, $hfind.

Switches

-b - Indicates &binvar is set.
-c - Chop the &binvar to first null byte.
-m - Creates the table if it does not exist.
-s - Displays the result.
-uN - Unset the item after N seconds.
-z - Decreases the item once per second until it reaches zero and then unsets it.

Parameters

<name> - The hash table to add to.
<item> - The hash table item to add to.
[text | &binvar] - The text or &binvar value to add.

Example

```
; call the setup once
; /abbr_setup
;
; //echo -a $abbr(lol)
;
alias abbr_setup {
  ;Populate the table, create it if it does not exist.
  hadd -m abbr lol laughing out load
  hadd abbr omg oh my gosh
  hadd abbr lmao laughing my a?? off
  hadd abbr brb be right back
}
```

```
;Get the abbreviation.
alias abbr return $hget(abbr, $1)
```

/hdel

Added in 1.9.0

/hdel [-sw] <name> <item>

Deletes an item from a hash table.

See also /hsave, /hload, /hmake, /hfree, $hget, $hfind.

Switches

-s - Displays the result.
-w - Indicates item is a wildcard.

Parameters

<name> - The hashtable to delete from.
<item> - The hashtable item to delete.

Example

```
alias example {
  ; Create a hash table
  /hadd -m example academic a

   ; Add some items to the hash table..
  /hadd example academy a
  /hadd example accelerate a
  /hadd example accelerator a
  /hadd example accept a
  /hadd example access a
  /hadd example accident a
  /hadd example because b

  ; Print number of items.
  //echo -a $hget(example, 0).item

  ; Remove everything matching wildcard 'a*'.
  /hdel -w example a*

  ; Print number of items.
  //echo -a $hget(example, 0).item

  ; Free table.
  /hfree example
}
```

/hload

Added in 1.9.0

/hload [-bBins] <name> <filename> [section]

Loads a table from a file.

See also /hsave, /hmake, /hfree, /hadd, /hdel, $hget, $hfind.

Switches

-b - Loads binary files.]]$cr]] and]]$lf]] are preserved when saving as binary files. Ignored if -i switch used.

-B - Uses a larger index than -b to allow longer binary data to be saved. This is not compatible with files created by the -b switch.
-i - Treats the file as an ini file. You can specify an optional section name after the filename.
-n - Load file as data only, with no items. When loading with -n each line of data is assigned an N item value, starting at N = 1.
-m - Creates the table if it does not exist. **(AdiIRC only)**
-s - Displays the result.

Parameters

<name> - The hashtable to load into.
<filename> - The filename of the saved hashtable to load.
[section] - Hashtable file section to load.

/hsave

Added in 1.9.0

/hsave [-abBinsu] <name> <filename> [section]

Saves a hash table to a file.

See also /hload, /hmake, /hfree, /hadd, /hdel, $hget, $hfind.

Switches

-s = Displays the result.
-b - Saves binary files.]]$cr]] and]]$lf]] are preserved when saving as binary files. Ignored if -i switch used.
-B - Uses a larger index than -b to allow longer binary data to be saved. This is not compatible with files created by the -b switch.
-i = Treats the file as an ini file. You can specify an optional section name after the filename.
-n = Load file as data only, with no items. When loading with -n each line of data is assigned an N item value, starting at N = 1.
-a = Appends to an existing file instead of overwriting it.
-u = Includes unset items.

Parameters

<name> - The hash table to save.
<filename> - The filename to save to.
[section] - Section of ini tile to save to. (used witth -i)

/hdec

Added in 1.9.0

/hdec [-bcmsuNzB] <name> <item> [num | &binvar]

Decreases the value of the item by 'num'.

Switches

-b - Indicates &binvar is set.
-c - Chop the &binvar to first null byte.
-m - Creates the table if it does not exist.
-s - Displays the result.
-uN - Unset the the item after N seconds.
-z - Decreases the item once per second until it reaches zero and then unsets it.
-B - Performs a $calcint calculation instead of regular $calc when arithmetic operators are used. **(AdiIRC only)**

Parameters

<name> - The hashtable where the item resides.
<item> - The hashtable item to decrease.
[num | &binvar] - The number or &binvar to decrease by.

Example

```
alias countdown {
  ;Adds the item down with a value of 10 to the table count.
  hadd -m count down 10

  echo -a $hget(count,down) $+ !
  :repeat

  ;Checks if the value of down returns true.
  if ($hget(count,down) > 1) {
    ;decreases down by 1
    hdec count down
    ;echos the current count
    echo -a $hget(count,down) $+ !
    ;repeats
    goto repeat
  }
  ;If the previous if statement returns false it carries on with this.
  else echo -a 0, Hurray!
  hfree -s count
}
```

/hinc

Added in 1.9.0

/hinc [-bcmsuNzB] <name> <item> [num | &binvar]

Increases the value of the item by 'num'.

Switches

-b - Indicates &binvar is set.
-c - Chop the &binvar to first null byte.
-m - Creates the table if it does not exist.
-s - Displays the result.
-uN - Unsets the item after N seconds.
-z - Decreases the item once per second until it reaches zero and then unsets it.
-B - Performs a $calcint calculation instead of regular $calc when arithmetic operators are used. **(AdiIRC only)**

Parameters

<name> - The hashtable where the <item> is stored.
<item> - The hashtable item to increase.
[num | &binvar] - The number or &binvar to increase by.

Example

```
alias example {
  ;Create a hashtable.
  /hmake -s example

  ;Add a few items.
  /hadd example item1 4
  /hadd example item2 7
  /hadd example item3 9

  ;Print the items.
  /echo -a item1 = $hget(example, item1)
  /echo -a item2 = $hget(example, item2)
  /echo -a item3 = $hget(example, item3)

  ;Increase the values.
  /hinc example item1 5
  /hinc example item2 12
  /hinc example item3 1
```

```
  /echo -e -

  ;Print the items.
  /echo -a item1 = $hget(example, item1)
  /echo -a item2 = $hget(example, item2)
  /echo -a item3 = $hget(example, item3)

  ;Cleanup.
  /hfree -s example
}
```

$hget

Added in 1.9.0

$hget(name|N)

Returns name of a hash table if it exists, or returns the name of the Nth hash table.

See also /hsave, /hload, /hmake, /hfree, /hdel, $hfind.

Parameters

name|N - Name of the hashtable or If N = 0, number of hashtables, otherwise the Nth hashtable name.

Properties

.size - Returns the size of the hashtable as specified in /hmake.

Example

```
; Create a hashtable.
/hmake Example

; Print number of hashtables.
//echo -ag $hget(0)

; Print name and size of the first hashtable.
//echo -ag $hget(1) - $hget(1).size
```

$hget(name|N, item|N, [&binvar])

Returns the data associated with an item in the specified hash table.

Parameters

name|N - Name of the hashtable or the Nth hashtable name.
item|N - Name of the item or if N = 0 number of items, otherwise the Nth item.
[&binvar] - Assigns the contents of the item to a &binvar. (optional)

Properties

.data - Returns the item value.
.item - Returns the item name.
.unset - Returns number of seconds before the item is automatically removed (if set).
.hash - TODO
.state - TODO

Example

```
; Create a hashtable
/hmake Example
```

```
; Add a item to the hashtable
/hadd Example ExampleItem ExampleValue

; Print number of items in the hastable.
//echo -ag $hget(Example, 0)

; Print information about the first item in the hashtable.
//echo -ag Name is $hget(Example, 1).item Value is $hget(Example, 1).data Unset at $hget(Example,
1).unset
```

$hfind

Added in 1.9.0

$hfind(name|N,text,[N],[M],[@window|command])

Searches table for the Nth item name which matches text and returns the item name.

See also /hsave, /hload, /hmake, /hfree, /hdel, $hget.

See also $hmatch, $hregex.

Parameters

name|N - Name of the hash table or the Nth hash table.
text - Text to search for.
[N] - If N = 0, number of matches, otherwise the Nth match. (optional)
[M] - (optional)

```
 n - Normal text comparison. (default if M is not specified)
 w - Text is wildcard text.
 W - Hash table item/data is wildcard text.
 r - Text is a regular expression.
 R - Hash table item/data is a regular expression.
 N - Use case-sensitive match. *(AdiIRC only)*
 H - /halt won't halt the running script if used inside the [command]. it will only halt the $hfin
d's search. *(AdiIRC only)*
```

[@window|command] - if @window is defined, fills the side-listbox with the results, otherwise the command is run on each result.

Properties

.data - Search for a matching data value instead of item name.

Example

```
; Searches hash table 'Example' for all items using and runs the command 'echo -ag $1-' on each re
sult, $1- holds the item name or data.
//echo -ag result: $hfind(Example, *, 0, w, echo -ag $1-)

; Searches the hash table 'Example' and prints the number of matches.
//echo -ag result: $hfind(Example, *, 0, w)

; Searches the hash table 'Example' and prints the first matched item.
//echo -ag result: $hfind(Example, *, 1, w)

; Searches the hash table 'Example' and prints the first matched data.
//echo -ag result: $hfind(Example, *, 1, w).data
```

$hmatch

Added in 2.0

$hmatch(name|N,text,[N])

Searches table for the Nth item name which matches text using a wildcard pattern and returns the item name.

See also /hsave, /hload, /hmake, /hfree, /hdel, $hget.

See also $hfind, $hregex.

Parameters

name|N - Name of the hash table or the Nth hash table.
text - Text to search for.
[N] - If N = 0, number of matches, otherwise the Nth match. (optional)

Properties

.data - Search for a matching data value instead of item name.

Example

```
; Searches the hash table 'Example' and prints the number of matches.
//echo -ag result: $hmatch(Example, *, 0)

; Searches the hash table 'Example' and prints the first matched item.
//echo -ag result: $hmatch(Example, *, 1)

; Searches the hash table 'Example' and prints the first matched data.
//echo -ag result: $hmatch(Example, *, 1).data
```