;=================================================

; Script Name: GBot - Global Bot B

; Author: Cerveza

; Version: 1

; Shard OSI / FS: OSI

; Revision Date: 2007/08/08

; Purpose: Control second char using EUO interface

; This is SECONDARY player

; Special Thanks: TrailMyx and Endless Night

;=================================================

namespace Global NSG1

set %timer #sCnt

set %peacetime #sCnt

set %pt 8 ; <----- Time in seconds between peacemaking

;set %inst QRF\_NRF\_MQF\_PRF\_LQF\_OQF\_RGP\_KRF

set #lpc 1000

repeat

===== Cure Primary =====

;if C in !charstatus && !charGhost = NO

;gosub doit 15 24 !charID 40 !charstatus

===== Heal Primary =====

if !hits < 100 && !charGhost = NO

;gosub doit 15 28 !charID 40 !hits

gosub bandage !CharID

;===== Auto Peacemake =====

;if #sCnt > %peacetime && !peace = #TRUE

; gosub autopeace

===== Flamestrike Target =====

If !charGhost = NO && #Ltargetid <> !Enemy && #finddist < 2

{

gosub doit 15 24 !EnemyId 40

}

return

;===== Hide Bot =====

;if !hide = #TRUE

; gosub doit 13 21 NONE 10

;set !hide #FALSE

===== Follow =====

Finditem !charID G\_15

if #finddist > 1 && !follow = #TRUE

event pathfind #findx #findy #findz

===== Invis Bot if Targetted =====

;finditem #enemyid G

;if #findcnt > 0

; {

; event macro 15 43

; target 3s

; set #ltargetKind 1

; event macro 23 0

; wait 10

; set #enemyID n/a

; }

===== Anti Logout =====

;if #sCnt > %timer + 300

; {

; event macro 8 1

; set %timer #sCnt

; }

;

Until #Clilogged = 0 || #CharGhost = YES

halt

===== SUBS =====

sub autopeace

event macro 3 0 ; whisper next line

msg $ ;clears hotkey so add wont mess up

gosub TM\_SyncTextScan

event macro 13 9 ; peace

target

event macro 23 0

TM\_FindValidText The\_instrument\_becomes\_forever\_quiet NULL

if #RESULT

{

finditem %inst C\_ , %pack

if #findkind = -1

{

display ok out of instruments

return

}

set #ltargetid #findid

wait 5

event macro 22 0

wait 10

}

event macro 3 0 ; whisper next line

msg /add$

target

key esc

set %peacetime #sCnt + %pt

return

;-------------------------------------

; %1 = Event Macro Param 1

; %2 = Event Macro Param 2

; %3 = Target ID or NONE

; %4 = Spell Delay

sub doit

if %3 = NONE

{

event macro %1 %2

wait %4

return

}

finditem %3 G\_8

if #findkind = -1

return

set #ltargetID %3

set #ltargetKind 1

event macro %1 %2

target 5s

if C in !charstatus && !charGhost = NO

{

key esc

return

}

event macro 22 0

wait %4

return

;-------------------------------------

sub bandage

finditem ZLF CA\_ , #BACKPACKID

if #findstack < 15

{

event ExMsg !charID 2 38 Only #findstack Bandages left!

}

else

[

set #lobjectid #findid

event macro 17 0

finditem %3 G\_8

if #findkind = -1

return

set #ltargetID %1

set #ltargetKind 1

event macro 22 0

wait 5s

return

]

;=================================================================

; Script Name: TrailMyx's Journal Scanning subs (#jindex based)

; Author: TrailMyx

; Version: 1.3

; Shard OSI / FS: OSI / FS?

; Revision Date: 2007/07/09

;-------------------------------------------------------------------------------

; Brings !\_jindex up to the most recent #journal entry

sub TM\_SyncTextScan

namespace push

namespace local text

set !\_jindex #jindex + 1

namespace pop

return

;-------------------------------------------------------------------------------

; %1, %2, %3, etc strings to match

; returns #TRUE for match, #FALSE for no match

; Will not advance !\_jindex pointer to allow for scanning journal history for more than one search.

; Also searches for : , #SPC in journal entry to be sure someone isn't spamming the text

sub TM\_FindValidTextNoAdvance

namespace push

namespace local text

set !temp\_lpc #LPC

set #LPC 1000

set !num\_args %0

set !first\_arg 1

if !\_jindex = N/A

set !\_jindex #jindex

set !index !first\_arg

repeat

set !temp\_jindex !\_jindex

set !text % . !index

while !temp\_jindex <= #jindex

{

scanjournal !temp\_jindex

str pos #JOURNAL #CHARNAME 1

set !namepos #STRRES

str pos #JOURNAL :\_ 1

set !smcpos #STRRES

str pos #JOURNAL !text 1

set !textpos #STRRES

if !textpos < !smcpos && !smcpos <> 0

set !pass #TRUE

else

set !pass #FALSE

if ( !text in #journal && ( :\_ notin #JOURNAL || !namepos = 1 || !pass ) )

{

set !temp\_jindex !temp\_jindex + 1

set #LPC !temp\_lpc

namespace pop

return #TRUE

}

set !temp\_jindex !temp\_jindex + 1

}

set !index !index + 1

until ( ( !index - !first\_arg ) >= ( !num\_args - 1 ) )

set #LPC !temp\_lpc

namespace pop

return #FALSE

;-------------------------------------------------------------------------------

; %1, %2, %3, etc strings to match

; returns #TRUE for match, #FALSE for no match

; Also searches for : , #SPC in journal entry to be sure someone isn't spamming the text

sub TM\_FindValidText

namespace push

namespace local text

set !temp\_lpc #LPC

set #LPC 1000

set !num\_args %0

set !first\_arg 2

if !\_jindex = N/A

set !\_jindex #jindex

set !index !first\_arg

repeat

set !text % . !index

while !\_jindex <= #jindex

{

scanjournal !\_jindex

if !text in #journal

{

if :\_ , !text notin #journal

{

set !\_jindex !\_jindex + 1

set #LPC !temp\_lpc

namespace pop

return #TRUE

}

}

set !\_jindex !\_jindex + 1

}

set !index !index + 1

until ( ( !index - !first\_arg ) >= ( !num\_args - 1 ) )

set #LPC !temp\_lpc

namespace pop

return #FALSE

;-------------------------------------------------------------------------------

; %1, %2, %3, etc strings to match

; returns #TRUE for match, #FALSE for no match

; Will not advance %\_jindex pointer to allow for scanning journal history for more than one search.

sub TM\_FindTextNoAdvance

namespace push

namespace local text

set !temp\_lpc #LPC

set #LPC 1000

set !num\_args %0

set !first\_arg 2

if !\_jindex = N/A

set !\_jindex #jindex

set !index !first\_arg

repeat

set !temp\_jindex !\_jindex

set !text % . !index

while !temp\_jindex <= #jindex

{

scanjournal !temp\_jindex

if !text in #journal

{

set !temp\_jindex !temp\_jindex + 1

set #LPC !temp\_lpc

namespace pop

return #TRUE

}

set !temp\_jindex !temp\_jindex + 1

}

set !index !index + 1

until ( ( !index - !first\_arg ) >= ( !num\_args - 1 ) )

set #LPC !temp\_lpc

namespace pop

return #FALSE

;-------------------------------------------------------------------------------

; %1, %2, %3, etc strings to match

; returns #TRUE for match, #FALSE for no match

; Advances !\_jindex so that you won't scan for past journal entries.

sub TM\_FindText

namespace push

namespace local text

set !temp\_lpc #LPC

set #LPC 1000

set !num\_args %0

set !first\_arg 2

if !\_jindex = N/A

set !\_jindex #jindex

set !index !first\_arg

repeat

set !text % . !index

while !\_jindex <= #jindex

{

scanjournal !\_jindex

if !text in #journal

{

set !\_jindex !\_jindex + 1

set #LPC !temp\_lpc

namespace pop

return #TRUE

}

set !\_jindex !\_jindex + 1

}

set !index !index + 1

until ( ( !index - !first\_arg ) >= ( !num\_args - 1 ) )

set #LPC !temp\_lpc

namespace pop

return #FALSE