**Variables**

**$mp** will return your current mana  
**$mppc** will return your mana percentage  
**$hp** will return your current health  
**$maxhp** will return your max health  
**$hppc** will return your health percentage  
**$cap** will return your capacity  
**$level** will return your level  
**$mlevel** will return your magic level  
**$posx** will return your current X position  
**$stamina** will return your current stamina time in minutes  
**$screenleft** it will return you the correct number to position a text in the left part of screen.  
**$screenright** it will return you the correct number to position a text in the right part of screen.  
**$screentop** it will return you the correct number to position a text in the top part of screen.  
**$screenbottom** it will return you the correct number to position a text in the bottom part of screen.  
**$name** will return your own name  
**$time** will return a time in seconds  
**$timems** will return a time in miliseconds  
**$deltatime** will return how long since you loaded the bot (in seconds)  
**$deltatimems** will return how long since you loaded the bot (in miliseconds)  
**$exptnl** will return how much experience left for your next level  
**$exph** will return how your average experience/hour  
**$expgained** will return how much experience you gained since you loaded the ElfBot  
**$timetnl** will return you how long would you take to level up with your current experience/hour  
**$exptolevel.xx** will return how much of experience left for XX level  
**$timetolevel.xx** will return how long would you take to level to XX level with your current experience/hour  
**$monstersaround.xx** will return you how many monsters are around XX SQMs around you  
**$playersaround.xx** will return you how many players are around XX SQMs around you  
**$sbtime** will return how much time of soft boots you still have (you must look the soft boots first)  
**$formattime.yy** will transform any number to a time number, the number must be in seconds, like 06:07:08 or 06h07m08s  
**$formatnum.yy** will transform any number to a number, like transforming 160293 to 160,293  
**$itemcount.ITEMID** will return the ammount of a visible item, you must change the ITEMID to a number, like 1234  
**$itemcount.’ITEMNAME’** will return the ammount of a item that you are using, you must change the ITEMNAME to a item name  
**$ringslot.property** will return the ID or COUNT of the item in the ring slot  
**$beltslot.property** will return the ID or COUNT of the item in the belt (ammunition) slot  
**$backslot.property** will return the ID or COUNT of the item in the backpack slot  
**$rhandslot.property** will return the ID or COUNT of the item in the right hand slot  
**$lhandslot.property** will return the ID or COUNT of the item in the left hand slot  
**$amuletslot.property** will return the ID or COUNT of the item in the amulet slot  
**$bootsslot.property** will return the ID or COUNT of the item in the boots slot  
**$legsslot.property** will return the ID or COUNT of the item in the legs slot  
**$chestslot.property** will return the ID or COUNT of the item in the chest (armor) slot

These are the properties the slot variables can have.

count -> it will return the amount of items that are in the desired slot.  
example: if [$beltslot.count < 20] refillammo, it will refill the ammunition if you have less than 20 items in the left hand id -> it will return the number ID of the item that are in the desired slot.

example: if [$lhandslot.id != 1234] equiplhand 1234, it will equip the left hand with a 1234 if there isn’t a 1234 equiped  
**$manashielded** will return if you are using mana shield

Example

**$drunk** will return if you are drunk

Example

**$hasted** will return if you are using haste  
example: if [$hasted == 0] say ‘Utani Hur’, it will cast ‘Utani Hur’ if you are not hasting  
**$paralyzed** will return if you are paralyzed  
example: if [$paralyzed == 1] say ‘Exura’, it will cast ‘Exura’ if you are paralyzed  
**$connected** will return if you are connected  
example: if [$connected == 0] reconnect, it will make you reconnect if you are not connected  
**$pkname** will return the name of player that’s attacking you  
example: ifplayerattacking attack $pkname, it will make attack back the player that’s attacking you  
**$fileisline.’filename’.'linetext’**, it will return 1 if a certain line in the ‘filename’ equals to ‘linetext’, 0 otherwise.  
example: if [$fileisline.'test.txt'.'utani gran hur'] say ‘Okay’, it will make you say ‘Okay’ if there is a line with the words ‘utani gran hur’ in the file ‘test.txt’.  
**$filerandomline.’filename’**, it will return a random line from the ‘filename’.  
example: if [$curmsg.content ? 'hello' && $curmsg.isdefault] say “$filerandomline.’himsgsresponder.txt’”, it will make you say a random line from the file ‘himsgsresponder.txt’ if someone says hello for you in the default channel.  
**$fileline.’filename.xyz’.n** will return the nth line of the file filename.xyz  
example: say ‘[$fileline."testbot.txt".30]‘, it will make you say the 30th line of the file testbot.txt  
**$token.’text’.n** will return the nth token (word/element) of a string. A token is either a word that ends with a space, or the inside of a singlequote/doublequote. If you omit to specify n, then the returned string will simply be the first token of ‘text’. If ‘text’ is ’12 “34 56″ 78′, n=1 will return: 12, n=2 will return: 34 56, and n=3 will return: 78  
example: if [$token.'$lastmsg.sender' == 'GM'] logout, it will make you logout if the first name of the last message you recieve is ‘GM’  
**$cutstr.’string’.startpos.lenght**, will let you cut a part of a sentence  
example: $cutstr.’123456′.2.3, will return 234, that is 3 characters starting at the 2nd character. You can also omit the lenght paramter, in which case the remainder of the string will be returned ($cutstr.’123456′.2 will return 23456)  
**$mshieldtime**, will return the time left in miliseconds to your magic shield goes out  
example: if [$mshieldtime  
**$hastetime**, will return the time left in miliseconds to your haste goes out  
example: if [$hastetime  
**$invistime**, will return the time left in miliseconds to your invisible goes out  
example: if [$invistime  
**$strengthtime**, will return the time left in miliseconds to your strenght goes out  
example: if [$strengthtime= 500] set $spell ‘Exura Vita’, it will make set the variable $spell to ‘Exura Vita’ if you recieved more than 500 of damage last second  
**$enemycount**, will return the amount of enemies on your screen  
example: if [$enemycount >= 1] xlog, it will make you force logout if an enemy appears on your screen  
**$friendcount**, will return the amount of friends on your screen  
example: if [$friendcount >= 1] say ‘Hi friend’, it will make you say ‘Hi Friend’ if a friend appears on your screen  
**$fishspots**, will return the amount of fishable spots on your screen  
example: if [$fishspots  
**$redbattlesign**, it will return 1 if you are red battle signed, and 0 if you are not  
example: ifnot $redbattlesign logout, it will make you logout if you aren't PvP PZ locked.  
**$inpz**, it will return 1 if you are inside a protection zone, and 0 if you are not  
example: if $inpz xlog, it will make you logout if you are inside a pz zone.  
**$rand.x.y** will return a random number from x to y, or just $rand.x for one from 0 to x  
example: auto 100 tradesay 'sell knight armor' | wait [($rand.200.350)\*1000], it will say in the Trade Channel: “sell knight armor”, and after a random time from 200 to 350 seconds it will say it again  
**$sstime** will return the time to serversave in seconds (only if a serversave message appears)  
example: auto 1000 setcaption ‘Time to Server Save: $sstime’, it will add to your caption the time left for the serversave.  
**$winitemcount.x** same as $itemcount.x, but returns only the amount of visible items of id x that are in windows (open containers/backpacks), excluding the ones equipped on you (excluding the ones in hand, belt, etc).  
example: if [$winitemcount.1234 > 20] equiprhand 1234 100, it will equip you with the item ’1234′ if there are more than 20 ’1234′ in your backpack  
**$fired** will return 1 if the preceding command has succesfully issued an attack or used a rune on someone.  
example: auto 100 sd target | if $fired wait 900, it will add a delay of 900 miliseconds only if the the sd target was succesfully issued. Note that the $fired DOES NOT tell you that a sd was actually thrown, but that a shoot sd command was sent because target was shootable.  
**$synctime** will return the time in milliseconds left for a sync shot if one is in progress.  
example: ifnot $synctime sd target, it will shoot a sudden death rune on your target only if a sync shot order isn’t in progress  
**$navion** will return 1 if you are connected to the navigation and 0 if not.  
example: auto 100 if $navion listas ‘Connected to the navigation’, if you are connected to the navigation it will show on the [hotkeys](http://www.elfbotdepo.com/category/scripts/hotkeys/) activated list the message ‘Connected to the navigation’.  
**$exectime** will return the time in miliseconds that the bot took to execute all the [hotkeys](http://www.elfbotdepo.com/category/scripts/hotkeys/) + some other important bot functionality. This will increase as you are running more [hotkeys](http://www.elfbotdepo.com/category/scripts/hotkeys/). This is updated 10 times per second, and shouldn’t become over 100ms.  
example: auto 100 setcaption ‘Exec Time: $exectime’, it will show on your [Tibia](http://www.tibia.com) window the time needed to execute the bot functions.  
**$topitem.posx.posy.posz** will return the ID of the top item on a tile

example: auto 100 if [$topitem.32222.34444.7 == 3031] moveitemonground 32222 34444 7 32223 34445 7, if the top item of the tile 32222 34444 7 is a item of ID 3031, it will be moved from 32222 34444 7 to 32223 34445 7.  
**$istileitem.posx.posy.posz.itemid** it will return 1 there is a item of X [item ID](http://www.elfbotdepo.com/item-ids/)  
example: auto 100 if [$istileitem.32222.34444.7.3031] playsound ‘monster.wav’, if an item of ID 3031 is at the tile 32222 34444 7 it will play the sound ‘monster.wav’.  
**$lastdmg** it will return the last damage number you recieved  
example: auto 100 if [$lastdmg > 500] savescreen, it will save a screenshot if the last damage you recieved was higher than 500.  
**$lastdmgtype** it will return the type of the last damage you recieved  
example: auto 100 if [$lastdmgtype == 30 && $lastdmg > 50 && $poisoned] {say ‘Exana Pox’ | wait 900}, it will make you cast ‘Exana Pox’ if the last damage you recieved was of poison type, and you are poisoned and it was higher than 50.  
**$mcount.”**[**conditions**](http://www.elfbotdepo.com/programming/actions-conditions/)**“** it will return the amount of monster satisfying certain [conditions](http://www.elfbotdepo.com/programming/actions-conditions/).  
example: auto 100 if [$mcount."$m.name == 'Dragon Lord' && $m.isshootable" >= 2] equipring 3051, it will make you equip an energy ring if there are more than 2 dragon lords shootable on your screen.  
**$pcount.”**[**conditions**](http://www.elfbotdepo.com/programming/actions-conditions/)**“** it will return the amount of players satisfying certain [conditions](http://www.elfbotdepo.com/programming/actions-conditions/).  
example: auto 100 listas ‘There are [$pcount."$p.isenemy && $p.ismage && $p.level > 80" enemies mages 80+ on your screen', it will show the text 'There are X enemies mages 80+ on your screen'.  
**$screencount.'name'** it will return how many of creatures with name 'name' on the same level as you and on the visible/shootable portion of the screen.

Example

**$lastdmgername** it will the name of the creature that has last attacked you. This is the name taken from the message “You lose xxx hitpoints due to an attack by mmmm”  
example: auto 100 if [$screencount."Stalker" == 0 && $lastdmgtime < 300 && $lastdmgername == "Stalker"] {useoncreature 3191 self | wait 300}, it will shoot a great fireball rune on yourself if you was damaged by a stalker in the last 300 miliseconds and there are no stalkers visible on your screen.  
**$lastdmgtime** it will return the time since you were last attacked by a creature in miliseconds.  
example: auto 100 if [$screencount."Warlock" == 0 && $lastdmgtime < 300 && $lastdmgername == "Warlock"] {useoncreature 3175 self | wait 300}, it will shoot a stoneshower rune on yourself if you was damaged by a warlock in the last 300 miliseconds and there are no warlocks visible on your screen. $ctrl it will return if you’re holding the CTRL Key example: if [$ctrl] {say ‘Exori Vis’ | wait 300} | else {say ‘Exevo Vis Hur’ | wait 300}, it will cast energy strike if you’re holding the CTRL Key and will cast energy wave if you’re not holding the CTRL Key. $shift it will return if you’re holding the SHIFT Key example: if [$shift] {makerune ‘Adori Frigo’ [$maxmp-20] | wait 300} | else {makerune ‘Adura Vita’ [$maxmp-20] | wait 300}, it will make an icicle rune if you’re holding the SHIFT Key and will make an ultimate healing rune if you’re not holding the SHIFT Key. $alt it will return if you’re holding the ALT Key example: if [$alt] crosshair 3003 | else crosshair 5710, it will make a rope crosshair appear if you’re holding the ALT Key and will make a light shovel crosshair appear if you’re not holding the ALT Key. $key.id it will return if you’re holding some key. More information about the keys on the end of this article example: if [$key.68] countitems 3031 | else statusmessage “You must hold the ‘D’ Key.”, it will count the gold coins if you’re holding the key of ID 68 (D Key) and will send the statusmessage “You must hold the ‘D’ Key” if you’re not holding the key of ID 68 (D Key). $skill.type it will return your current skill. example: auto 100 if [$skill.'sword' >= 90] {playsound ‘advancedskill.wav’ | flash}, it will play the sound ‘advancedskill.wav’ and flash if your sword fighting is higher than 90.  
**$skillpc.type** it will return your current skill percent.  
example: auto 100 if [$skillpc.'mlevel' == 1] {playsound ‘advancedskill.wav’ | flash}, it will play the sound ‘advancedskill.wav’ and flash if your magic level is about to advance.  
**$skilltime.type** it will return the estimated time left in seconds to advance a skill.  
example: auto 100 if [$skilltime.'fishing' will check for your fist fighting  
**club** -> will check for your club fighting  
**sword** -> will check for your sword fighting  
**axe** -> will check for your axe fighting  
**distance** -> will check for your distance fighting  
**shielding** -> will check for your shielding  
**fishing** -> will check for your fishing  
**mlevel** -> will check for your magic level  
Those variables below will return if someone is in a leader or an enemy or a subenemy or a friend or a subfriend  
**$isleader.'name'** it will check if the player 'name' is on the leaders list on aimbot  
**$isfriend.'name'** it will check if the player 'name is on the friends list on lists  
**$issubfriend.'name'**  it will check if the player 'name is on the subfriends list on lists  
**$isenemy.'name'**  it will check if the player 'name is on the enemies list on lists  
**$issubenemy.'name'** it will check if the player 'name is on the subenemies list on lists  
example: auto 100 foreach 'newmessages' $i if [$isfriend.'$i.sender' == 0 && $i.ispvt] {playsound ‘privatemessage.wav’ | flash | break}, it will play the sound ‘privatemessage.wav’ if a player who is not on your friends list sends you a private message.  
You can use your personal variable, using foreach command (allcreatures, allplayers, allmonsters, floorcreatures, floorplayers, floormonsters, screencreatures, screenplayers, screenmonsters, shootablecreatures, shootableplayers, shootablemonsters). Or the internal variables listed below.  
**$self.property**it will check something about you  
**$target.property** it will check something about your target  
**$followed.property** it will check something of your current followed player  
**$attacked.property** it will check something of the creature that has currently red box around it  
**$attacker.property** it will check something of the last creature that has attacked you  
**$pk.property** it will check something of the last player that attacked you that the bot has chosen  
**$lastdmger.property** it will check something of the last creature that made a damage on you  
**$pattacker.property** it will check something of the last player that has attacked you (same as $pk.property)  
**$mttacker.property** it will check something of the last monster that has has attacked you  
**$enemy.property** it will check something of the best enemy that the bot has chosen  
**$friend.property** it will check something of the best friend that the bot has chosen  
**$subenemy.property** it will check something of the best sub-enemy that the bot has chosen  
**$subfriend.property** it will check something of the best sub-friend that the bot has chosen  
**$anyenemy.property** it will check something of the best enemy that the bot has chosen, if none than the best sub-enemy  
**$anyfriend.property** it will check something of the best enemy that the bot has chosen, if none than the best sub-friend  
**$coretarget.property** it will check something of the best valid target that the core aimbot subsystem of the bot has chosen  
**$triggertarget.property** it will check something of the best valid target that the trigger aimbot subsystem has chosen (only if shot commands have been issued and a target is valid)  
**$autoaimtarget.property** it will check something of the best target that both aimbot subsystems combined have chosen. This is simply the triggertarget if valid, otherwise it is the coretarget.

Or you can select whoever you want setting someone trought the variables below

**$creature.name** it will check for a creature on your screen that has the name ‘name’, and then you can hold the informations in another variable, for example: auto 100 set $j $creature.’Bubble’  
**$mostexposed.’targetgroup’** it will check for every enemy on your screen, and then choose the one that more friends of you can shoot it, the target group can be: sorcerer/druid/knight/paladin/mage/nonmage. And then you hold the informations in another variable, for example: auto 100 set $i $mostexposed.’mage’  
**$mostshot.’targetgroup’** it will check for the creature that has recieved most sudden death rune shots in the last two seconds, the target group can be: enemy/subenemy/friend/subfriend, if you want to specify a vocation also, you can append a space and one of the following: sorcerer/druid/knight/paladin/mage/nonmage. And then you hold the informations in another variable, for example: auto 100 set $i $mostshot.’enemy mage’

These creature variables have some properties

**name** -> will return the name of someone, for example: $target.name will return your target’s name  
**id** -> will return the ID of someone, for example: $enemy.id will return your best enemy’s ID  
**posx** -> will return the posx of someone, for example: $friend.posx will return your best friend’s posx  
**posy** -> will return the posy of someone, for example: $friend.posy will return your best friend’s posy  
**posz** -> will return the posz of someone, for example: $friend.posz will return your best friend’s posz  
**dir** -> will return the direction that someone is facing (check the end of this thread for more information), for example: $self.dir will return your own direction number  
**isonnavi** -> will return 1 if someone is connected to the navigation, and 0 if not, for example: $friend.isonnavi will return 1 if the best friend chosen by the bot is connected to the navigation server, and 0 if not  
**outfit** -> will return the outfit number of someone, for example: $subenemy.outfit will return your best sub-enemy’s outfit number  
**hppc** -> will return the percentage of HP of someone, for example: $target.hppc will return your target’s HP percent  
**speed** -> will return the speed of someone, for example: $anyenemy.speed will return your best enemy (or best sub-enemy)’s speed  
**skull** -> will return the skull type of someone (check the end of this thread for more information), for example: $self.skull will return your own skull type  
**party** -> will return the party icon of someone is in a party (check the end of this thread for more information), and 0 if not on party, for example: $self.party will return 4 if you are in the leader of a party  
**warbanner** -> will return the war banner of someone (check the end of this thread for more information), and 0 if not on a guild war, for example: $target.warbanner will return 2 if your target is a guild enemy  
**isenemy** -> will return 1 if someone is an enemy, and 0 if not, for example: $target.isenemy will return 1 if your current target is an enemy, and 0 if not  
**isfriend** -> will return 1 if someone is a friend, and 0 if not, for example: $target.isfriend will return 1 if your current target is a friend, and 0 if not  
**issubenemy** -> will return 1 if someone is a sub-enemy, and 0 if not, for example: $target.issubenemy will return 1 if your current target is a sub-enemy, and 0 if not  
**issubfriend** -> will return 1 if someone is a sub-friend, and 0 if not, for example: $target.issubfriend will return 1 if your current target is a sub-friend, and 0 if not  
**isleader** -> will return 1 if someone is a combo leader listed in Aimbot dialog, and 0 if not, for example: $target.isleader will return 1 if your current target is a leader, and 0 if not  
**distx** -> will return how many sqms to the west or east someone is from you, $target.distx will return how many sqms to the west or east your target is from you  
**disty** -> will return how many sqms to the north or south someone is from you, $target.disty will return how many sqms to the north or south your target is from you  
**distance** -> will return the highest number of .distx and .disty, $target.distance will return how many sqms a target is from you  
**ismonster** -> will return 1 if someone is a monster or a npc, 0 if not, $followed.ismonster will return 1 if your current followed is a monster, 0 if not  
**isnpc** -> will return 1 if someone is a npc, 0 if not, $followed.isnpc will return 1 if your current followed is a npc, 0 if not  
**isplayer** -> will return 1 if someone is a player, 0 if not, $followed.isplayer will return 1 if your current followed is a player, 0 if not  
**isonscreen** -> will return 1 if someone is on the same floor as you and on the visible portion of the screen, 0 if not, $followed.ismonster will return 1 if your current followed is on your screen, 0 if not  
**isshootable** -> will return 1 if nothing obstructs a projectile trajectory between you and him, 0 if not, $followed.ismonster will return 1 if your current followed is shootable, 0 if not  
**isparalyzed** -> will return 1 if someone is paralyzed, 0 if not, $target.isparalyzed will return 1 if your target is paralyzed, 0 if not  
**color1** -> will return the head color’s number of someone, $self.color1 will return yours head color’s number, 0 if not  
**color2** -> will return the primary color’s number of someone, $self.color2 will return yours primary color’s number, 0 if not  
**color3** -> will return the secondary color’s number of someone, $self.color3 will return yours secondary color’s number, 0 if not  
**color4** -> will return the detail color’s number of someone, $self.color4 will return yours detail color’s number, 0 if not  
**canshoot.’name’** -> will return 1 if someone can shoot the creature ‘name’, $friend.canshoot.’Bubble’ will return 1 if your best friend can shoot ‘Bubble’, 0 if not  
**exposedcount** -> will return how many friends/yourself can shoot someone, $coretarget.exposedcount will return how many friends + you can shoot the coretarget  
**shotcount** -> will return how many sudden death runes someone has recieved on the last 2 seconds, $self.shotcount will return how many sudden death runes you have recieved on the last two seconds

Also, there will be some exclusive variables if the bot managed to get a look at the player, it will have the ‘haslookinfo’ property set. You should verify that it’s set before using the below fields:

**haslookinfo** -> will return 1 if player has the below properties available: if [$target.haslookinfo && $target.hp < 400] sd $target.id (since $target.hp is only available if $target.haslookinfo is set. Otherwise, $target.hp would return 0 and the alone condition $target.hp < 400 would be true!) guild -> will return someone’s guild, for example: $target.guild will return your target’s guild  
**level** -> will return someone’s level, for example: $enemy.level will return the best enemy’s level  
**voc** -> will return someone’s vocation (short), for example: $pattacker.voc will return the current player attacking you’s vocation (like NV, D, S, P, K, ED, MS, RP, EK)  
**vocation** -> will return someone’s vocation (long), for example: $attacked.vocation will return the attacked player’s vocation (like No Vocation, Druid, Sorcerer, Paladin, Knight, Elder Druid, Master Sorcerer, Royal Paladin, Elite Knight)  
**isknight** -> will return 1 if someone is a knight, and 0 if not, for example: $target.isknight will return 1 if your current target is a knight, and 0 if not  
**ispaladin** -> will return 1 if someone is a paladin, and 0 if not, for example: $target.ispaladin will return 1 if your current target is a paladin, and 0 if not  
**issorcerer** -> will return 1 if someone is a sorcerer, and 0 if not, for example: $target.issorcerer will return 1 if your current target is a sorcerer, and 0 if not  
**isdruid** -> will return 1 if someone is a druid, and 0 if not, for example: $target.isdruid will return 1 if your current target is a druid, and 0 if not  
**ismage** -> will return 1 if someone is a druid or a sorcerer, and 0 if not, for example: $followed.ismage will return 1 if the current followed player is a mage, 0 if not  
**maxhp** -> will return someone’s max hp, for example: $friend.maxhp will return the best friend’s max hp  
**hp** -> will return someone’s current hp, for example: $friend.hp will return the best friend’s current hp  
**maxmp** -> will return someone’s max mp, for example: $subenemy.maxmp will return the best sub-enemy’s maxmp  
**mp** -> will return someone’s max hp, for example: $subenemy.mp will return the best sub-enemy’s current mp  
**attackedme** -> will return 1 if someone attacked you, and 0 if not, for example: $friend.attackedme will return 1 if the best friend attacked you, and 0 if not  
**$curmsg.property** it will check something about each message you are recieving.  
**$lastmsg.property** it ill check something about the last message you recieved.  
**$lastnavmsg.property** it will check something about the last navigation message you recieved.

You can also use foreach ‘newmessages’ $varname action.

These variables have some properties, I will use a message as example 16:16 Lucas Terra [2]: hey, u there?, that was sent in Default.  
**sender** -> will return the name of the message sender, in the example: $curmsg.sender will return ‘Lucas Terra’  
**level** -> will return the level of message sender, in the example: $curmsg.level will return ’2′  
**content** -> will return what was wrote in the message, in the example: $curmsg.content will return ‘hey, u there?’  
**formatted** -> will return the message formatted (the original message without time), in the example: $curmsg.formatted will return ‘Lucas Terra [2]: hey, u there?’  
**channel** -> will return in what channel the message was sent, in the example: $curmsg.channel will return ‘Default’  
**color** -> will return the message’s color, in the example: $curmsg.color will return 16838639  
**isdefault** -> will return 1 if the message was sent in default, and 0 if not, in the example: $curmsg.isdefault will return ’1′  
**isyell** -> will return 1 if the message was a yell, and 0 if not, in the example: $curmsg.isyell will return ’0′  
**iswhisper** -> will return 1 if the message was a whisper, and 0 if not, in the example: $curmsg.iswhisper will return ’0′  
**isprivate** -> will return 1 if the message was a private, and 0 if not, in the example: $curmsg.isprivate will return ’0′  
**isbroadcast** -> will return 1 if the message was a broadcast (message in green that appears in the center of screen), and 0 if not, in the example: $curmsg.isbroadcast will return ’0′  
**isredtext** -> will return 1 if the message was a red text (like a server save message), and 0 if not, in the example: $curmsg.isredtext will return ’0′  
**isorangetext** -> will return 1 if the message was a orange text (like a tutor message in help), and 0 if not, in the example: $curmsg.isredtext will return ’0′  
**isnpc** -> will return 1 if the message was a sent in NPC channel, and 0 if not, in the example: $curmsg.isnpc will return ’0′  
**ischannel** -> will return 1 if the message was sent in help or trade or game-chat or private-channels, and 0 if not, in the example: $curmsg.ischannel will return ’0′  
**isbotlook** -> will return 1 if the message was a bot look (when the bot looks at someone it isn’t displayed), and 0 if not, in the example: $curmsg.isbotlook will return ’0′  
**isguild** -> will return 1 if the message was sent in guild-chat, and 0 if not, in the example: $curmsg.isguild will return ’0′  
**isaction** -> will return 1 if the message was an action (like the ‘Ahhh…’ of mana potions), and 0 if not, in the example: $curmsg.isaction will return ’0′  
**isstatus** -> will return 1 if the message was a status message (like the ‘Sorry, not possible’), and 0 if not, in the example: $curmsg.isstatus will return ’0′  
**isinfo** -> will return 1 if the message was a green text (like a look at players), and 0 if not, in the example: $curmsg.isinfo will return ’0′  
**isgame** -> will return 1 if the message was sent in game-chat, and 0 if not, in the example: $curmsg.isgame will return ’0′  
**isrl** -> will return 1 if the message was sent in RL-chat, and 0 if not, in the example: $curmsg.isrl will return ’0′  
**istrade** -> will return 1 if the message was sent in trade channel, and 0 if not, in the example: $curmsg.istrade will return ’0′  
**ishelp** -> will return 1 if the message was sent in help channel, and 0 if not, in the example: $curmsg.ishelp will return ’0′  
**isnavi** -> will return 1 if the message was sent in the navigation server, and 0 if not, in the example: $curmsg.isnavi will return ’0′  
To understand better the variables, test each one by yourself changing some values and so.

Signals and Operators

+ -> addition

example: if [($itemcount.'mana potions'+$itemcount.'strong mana potions') < 50] gotolabel buymfs, it will go to the label ‘buymfs’ if the ammount of mana potions + strong mana potions is below 50 – -> subtraction

example: if [($maxhp-$hp) >= 300] say ‘Exura Vita’, it will cast ‘Exura Vita’ if your maxhp-hp is higher than 300

/ -> division

example: if [$poisondmg/5 > 5] say ‘Exana Pox’, it will cast ‘Exana Pox’ if the poison damage/5 is higher than 5

\* -> multiplication

example: if [$mp\*5 < $maxmp] mana self, it will use a mana potion on yourself if your mana is below 1/5 of your max mana % -> modulus, remainder of a division

example: if [$timems%4000 < 2000] say ‘Exori Flam’, it will cast ‘exori flam’ if the modulus of timems/4000 is below 2000 > -> higher

example: if [$count > 5] gmana self, it will use a great mana potion if the count of last counted item is higher than 5

< -> lower

example: if [$count < 5] gotolabel buymfs, it will go to the label ‘buymfs’ if the count of last counted item is below 5 >= -> higher or equal

example: if [$maxhp-$hp >= 700] mana self, it will use a mana potion on yourself if your maxhp-hp is higher than 700

lower or equal

example: if [$hp and

example: if [$hp or

example: if [$hp equal

example: if [$connected == 0] reconnect, it will reconnect if you are disconnected

!= -> different, not equal

example: if [$manashielded != 1] say ‘Utamo Vita’, it will use cast ‘Utamo Vita’ if you are not using mana shield

? -> stringpos, returns the position of string2 inside of string1, or 0 if string2 is not present there: [string1 ? string2]

example: if [$curmsg.content ? 'soft'] playsound tradealarm.wav, it will play the sound tradealarm.wav, if you recieve a message contaning the word soft, like ‘sell soft boots’, ‘buy soft boots and boots of haste, (because it will return a non-zero position and the if will be evaluated as true)’

Equalities

If an equality is equal to 0, means it’s not true, if it’s equal to 1, means it’s true

example: if [$connected == 0] reconnect, it will make you reconnect if you are not connected

Keyboard Keys’re IDs:

Check out this website:

http://www.tronan.com/macromachine/scripthelp/VK.htm

Choose the key you want to use, for example: ‘Y key’. Go to the website, do CTRL+F and type ‘Y key’, it will find the decimal value ’89′, then just use it on your hotkey. For example: if [$key.89] say ‘Yeah!’, it will say ‘Yeah!’ if you’re holding the ‘Y key’.

This is useful for making [hotkeys](http://www.elfbotdepo.com/category/scripts/hotkeys/) like: CTRL+SHIFT+END do Exori San, CTRL+END do Exori Con, END do SD Target. Make a hotkey on the END Key:

if [$target.isshootable] {if [$ctrl && $shift] {if [$mp >= 20 && $attacked.distance = 25 && $attacked.distance 0

Yellow Skull -> 1

Green Skull -> 2

White Skull -> 3

Red Skull -> 4

Black Skull -> 5

Party Types:

No Party -> 0

Inviting (Leader Yellow) -> 1

Inviting (Member Blue) -> 2

on Party (Member Blue) -> 3

on Party (Leader Yellow) -> 4

Exp Sharing, working (Member Blue) -> 5

Exp Sharing, working (Leader Yellow) -> 6

Exp Sharing, on standby (Member Blue) -> 7

Exp Sharing, on standby (Leader Yellow) -> 8

Exp Sharing, not working (Member Blue) -> 9

Exp Sharing, not working (Leader Yellow) -> 10

War Banner Types:

No War -> 0

Green Banner (Friend) -> 1

Red Banner (Enemy) -> 2

Blue Banner (In a War that you’re not fighting) -> 3

Direction Types:

North -> 0

East -> 1

South -> 2

West -> 3