

HDeffo's EM sit Script **(Multi-sitter script)**

This script is provided free, and full perm, if you paid for it demand a refund, and notify HDeffo Resident in-world.

Instructions for use:

1. Drop script into your prim.
2. Set the parameters for who can, and cannot sit, with the white/black list
3. Set the number of seating positions, by providing offsets in the offsets list
4. Set the directions for the avatars to face, by setting one rotation per seat
5. Add the animations to the prim (12 max)
6. (Optional) customise messages

Detailed Instructions:

```
0 //if you payed more then L$0 for this YOU PAYED TOO MUCH!
1
2 //Now includes multi-avatars regardless of the number of prims(still follows max link limits sadly), multi-animations,
  animation offset, blacklist, whitelist,force or pick animations, and avatar height adjustment. IM the HDeffo resident for
  details on how to set the script and any suggestions to add to the script. Suggestions will be taken and added as quickly
  as possible and for free if they are possible.
3 list whitelist = [];
4 list blacklist = [];
5 list offsets = [<0.4,0.0,-0.25>];
6 list rotations = [ZERO_ROTATION];
7 list anim_offset = [<-0.25,0.0,0.0>];
8 string FullMessage = "Sorry the chair is full.";
9 string restricted_message = "Sorry you arnt aloud to sit here.";
10 integer FORCED = TRUE;
11
12 list keys;
13 list anims;
14 integer prims;
15 string request;
16 integer link;
17 vector size;
18 default
19 {
20     state_entry()
21     {
22         llListen(-1337,"",NULL_KEY,"");
23         llSitTarget(<0,0,0.01>,ZERO_ROTATION);
24         prims=llGetNumberOfPrims();
25         integer i;
26         integer t = llGetInventoryNumber(INVENTORY_ANIMATION);
```

Fig 1.

As with most scripts, you simply drop this into your prim with the edit window (don't use open, it will deactivate the script). Once its there, it's time to set up the various options.

The first thing you are likely to want to do is set up how many can sit. In the top portion of the script, below the orangey text (See Fig 1. above), you will see a list of options; whitelist, blacklist, offsets, rotations, anim_offset, FullMessage, restricted_message and FORCED.

Lets take you through what these do:

whitelist: This is where you put a list of UUIDs or 'Keys' for always-permitted avatars. They are contained in "" marks, and separated with commas. If blank, everyone is permitted

blacklist: Same as above, except that these UUIDs will always be denied. Again, contained in quotation marks, and separated by commas.

offsets: This is one of the tricky ones. Each sitting position you wish to have, needs an offset. The more you have in this list, the more seats you give your prim. These are stored as vectors, and we'll go into more depth in a moment on these.

rot: Another of the really complicated ones, for each vector you put in, above, you need a matching rotation in this list. These are stored as vectors too.

anim_offset: This is per animation, organised by inventory order. In other words, look at your prims inventory and jot down the order of the animations, only. number them with the top-most one being 0. This is how your list will be organised. Again, these are in vectors.

FullMessage: Message given when the chair is full, stored in quotation marks.

restricted_message: Message given out when an avatar is blacklisted. Also given out when a whitelist is present to any not on the whitelist

FORCED: This specifies whether animations can be changed or not. If set to TRUE, then the item has only one animation, ever. If set to FALSE, then you can change. These words need to be in caps.

Usage:

Lets start with something simple. Pop this script into your prim, along with any sit animation you wish. It's set to have a default sitting position for one person. Using any common freebie positioner tool (LearJeff's Easy Sit Target is a good one), you can play with the offset and rotation, til it works, then store these.

Lets do that now.

Put whatever positioner script(s) are needed into your prim, and do as it instructs til you look like you are sitting correctly. Once it gives you the numbers you need jot them down on paper or a text editor, and open up the EM sit script by HDeffo.

Remove the current offset vector, in offsets, thats the <0.4,0.0,-0.25> bit, and replace with yours. If your positioner gives a rotation, do the same for rotations, but on this occasion, remove the ZERO_VECTOR, for your own.



Technical Info Break:

Vectors are stored within <> brackets, using an X,Y,Z format. Vectors used for positioning, typically run from +/- 00000 to +/-256.0000, on both the X and Y axis, and up to 4096 for z. When using sitter code such as this, any offsets can be used, but typically will be less than 1.0m for any of the three.

In this script, these vectors are stored within lists, so a vector of <1.0,0.0,-0.25> will be prefixed with a square bracket [, which signifies the start of a list, and followed by a comma if you have additional vectors to add, or a closing square bracket if you do not].

Rotational vectors work differently, however, HDeffo has made things simpler for you, by using a conversion. Like the above example, this script uses three axis vectors, <X,Y,Z>. These are your basic 0-359 degrees numbers, like you'd find on a school compass/protractor. Note, I did 0-359. Thats right, when you reach 360, it becomes 0 again.

Back to the script. Once you've added your vectors and the matching rotations, for the first sitting position, you can use it. However, if you wish to add more seats, its back to the positioner again, this time, adding the vectors, instead of replacing.

Here's an example:

`list offsets = [<0.4,0.0,-0.25>];` **The line we want to change**

`<0.6,0.0,-0.24>` **The vector we wish to add.**

`list offsets = [<0.4,0.0,-0.25>,<0.6,0.0,-0.24>];` **New seat added.**

Simply repeat this, for the rotations list, one per seat. And keep going til all the seats you want, are added to both these lists.

Once all your seats are added, test it out, make sure all your animations are positioned right, and, if not, get your positioning tool out again with the offending animation, and pop in an offset.

Tricky bit.....

You may need to add an offset for an animation, but not for the others. You can't simply add it and assume that the animation you need altered will be adjusted, you will need to 'pad' out the anim_offset list. The simplest way is, if all the animations are fine, except for one. To put ZERO_VECTOR for each entry thats fine, until you get to the entry that needs the alteration, then pop it in, same method as you did for the seat offsets.

White/Black Lists

The script uses a language we call LSL, or sometimes called 'Mono'. There are some limits to it, and one of these is involving names. The black list and whitelist in this script requires a UUID, or 'Key' for an avatar in order to function, due to the limitations of LSL being unable to provide you one using just the name.

There are many ways to get this, ask the person in question if they know their key (it's sorta like a social security number after all), or various scripted HUDs. If you are in Phoenix or Firestorm Viewers, you can get this UUID from the profile box when you open it. (Not with web profiles!).

When you have a list of these keys, you add them to the appropriate list.

Keys must be stored within quote marks, and then these blocks separated from each other with a comma, see example below:

```
list whitelist = ["10a0a8c6-2f86-45e0-95f3-cd9ce39c68f5","10a0a8c6-2f86-45e0-95f3-cd9ce39c68f5"];
```

Try to keep the total entries, of both lists combined, as short as possible. Each entry gets checked and matched against for each person who sits down, the more you have, the slower that process gets.

