New Class Creation

Creating a New Class:

(New Classes are created from old ones. After you create a "new" class you can alter it to work a different way by editing the new .bof for the new class. Depending on what it is it could be a small or big difference from the one you created it from.)

Step 1:

The first step is to copy the *kodbase* entry for the class you want to add a new type of and paste it to a text file in a folder that contains a copy of the default .bof file,.rsc file,default clients rsc0000.rsb file along with the text file that contains the kodbase info.

Use this cheese kodbase entry as an example of what a kodbase entry looks like.

```
C Cheese 14440 11649 Food
R   cheese_name_rsc 14441
R   cheese_name_plural_rsc 14442
R   cheese_icon_rsc 14443
R   cheese_desc_rsc 14444
V   viFilling 42
V   viNutrition 43
V   vrName 0
V   vrName_plural 40
V   vrIcon 1
V   vrDesc 2
V   viIndefinite 3
V   viBulk 5
V   viWeight 6
V   viValue_average 18
Y   viFilling 7
Y   viNutrition 8
Y   piNumber 6
```

Step 2:

The next step is to re-create the kodbase entry of the default class you want to create the new class from. I'd keep this in the same text file for Quick Reference.

This is an example of what your text file might look like with the new class you are created along with the default one. The ##### parts are numbers that need to be changed to new numbers that aren't already anywhere in the servers kodbase file. Search the kodbase for the new numbers you want to use first to make sure they aren't already being used and won't cause conflict errors. Easiest way is to look at the last number in the clients *rsc0000.rsb* file.

```
C NewFood ##### 11649 Food
R   newfood_name_rsc #####
R   newfood_name_plural_rsc #####
R   newfood_icon_rsc #####
R   newfood_desc_rsc #####
V   viFilling 42
V   viNutrition 43
V   vrName 0
V   vrName_plural 40
V   vrIcon 1
```

V vrDesc 2
V viIndefinite 3
V viBulk 5
V viWeight 6
V viValue_average 18
Y viFilling 7
Y viNutrition 8
Y piNumber 6

Default kodbase entry:

C Cheese 14440 11649 Food
R cheese_name_rsc 14441
R cheese_name_plural_rsc 14442
R cheese_icon_rsc 14443
R cheese_desc_rsc 14444
V viFilling 42
V viNutrition 43
V vrName 0
V vrName_plural 40
V vrIcon 1
V vrDesc 2
V viIndefinite 3
V viBulk 5
V viWeight 6
V viValue_average 18
Y viFilling 7
Y viNutrition 8
Y piNumber 6

Step 3:

The next step takes a bit longer to do. First you will need a Hex editor. You can find a hex editor like "Hex Workshop" at download.com and you can get the crack for it at bestserials.com. After you have those open up the default classes .bof file from the folder you made a copy of it too. Then use a Base Converter with the Decimal to Hex option on and put in the default resource number into the Decimal side and copy what comes out in the Hex side. Now use the Search option and search for the first hexed resource number. Now it should be highlighted. Now use the Base Converter again and use the same method to convert the new class your making's resource number to Hex and manually replace the new resource numbers Hex from the old one. Use this same method to replace all the resource numbers as needed.

Remember only the first few sets of Unique resource numbers in a class need to be changed to create any new class out of a default one. Unique meaning that only that class has them.

Step 4:

Now you will have to basically do the same thing you did in *Step 3* only apply the new resource numbers on the decoded RSC files. After you have changed default resource numbers and wrote in the new names,bgf graphic and description with the hex editor, you will have to open up a decoded version of the clients rsc0000.rsb file. Then you copy everything you can in the edited encoded RSC file and paste it and the very end of the decoded rsc0000.rsb file. Then recode both the new rsc file and the client rsb file.

Step 5:

Now add the new class's kodbase entry to the end of the servers kodbase.txt file. And copy the new RSC and BOF files for the new classes to the servers folders. The rsc file goes in the RSC folder the BOF goes in *memmap folder.*

Step 6:

Now start the server and check the "**Errors**" part. If you see any numbers that mention the new resource numbers that means you encoded a file with a wrong resource number and will have to go back and fix it.

Step 7:

Now put your item into the game *(a test server would be best)* and beta test it. At first it will just be a duplicate of the old class you created it from with a different name and graphic. You can edit the **INT**'s of the new class by further editing the classes .bof file using the same method you used for the resource numbers with the *Base Converter.*

Meridian files and what they do:

*.bof = encoded class file
*.roo = encoded map file
*.rsc = encoded text resource file
*.bgf = encoded graphic file
rsc0000.rsb = encoded main client text resource file
*.arq = compressed update zip type file

BOF Endbytes Reference:

0000 - $ (UNDEFINED)
0010 - INT
0020 - OBJECT
0030 - LIST
0040 - RESOURCE
0050 - TIMER
0060 - SESSION
0070 - ROOM_DATA
0080 - TEMP_STRING
0090 - STRING
00A0 - CLASS
00B0 - MESSAGE
00D0 - OVERRIDE
00F0 - INVALID

Or if that's all too much for you and you can't hex, then I recommend you download the editors instead:

M59 Admin Tools