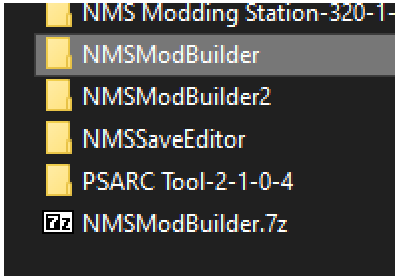
How to use NMSMB (for non-mod-authors)

This will help you step by step on how to install (and also merge) mods for NMSMB. If a section is marked as **(Advanced)** you're free to skip it.

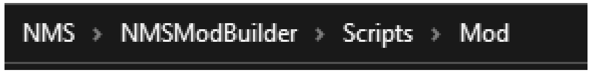
Step 1) Download the latest version from.

https://github.com/cmkushnir/NMSModBuilder/releases

Step 2) unzip the 7z file somewhere on your PC.



Step 3) Open the NMSModBuilder and navigate to the Mod folder

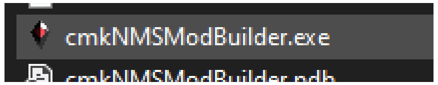


Step 3.1) In that folder you will find 2 folder cmk and new.

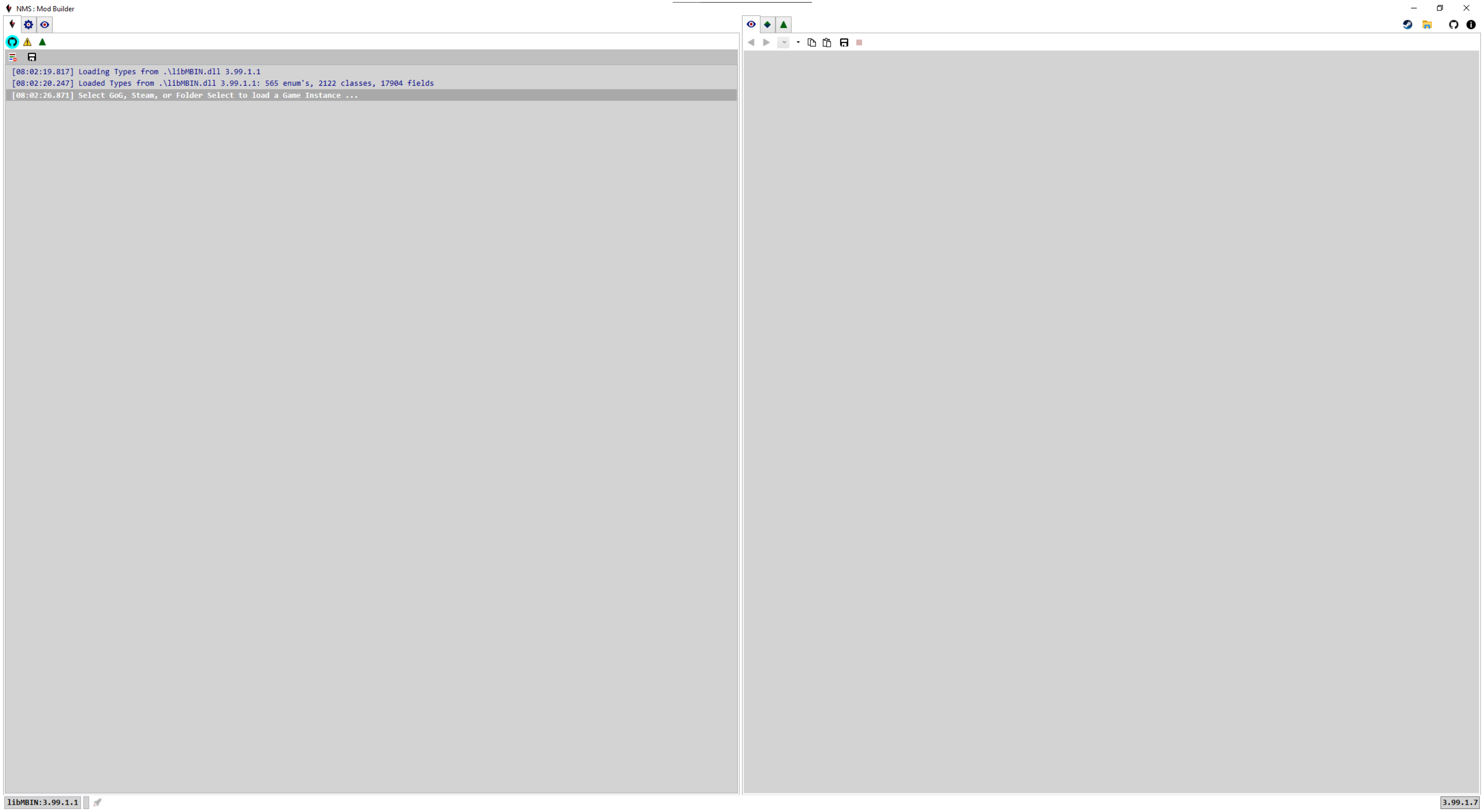
- Cmk folder contains script examples (ie. Mods)
- New: rename this folder to your liking fe. MyMods

Step 4) Drop the **mod.cs** files you got from NexusMods or from the cmk folder in this folder.

Step 5) Now you can go back to the main folder and execute. Make sure you installed .NET X x64 Desktop Runtime https://dotnet.microsoft.com/en-us/download/dotnet . Check on github which version you need.

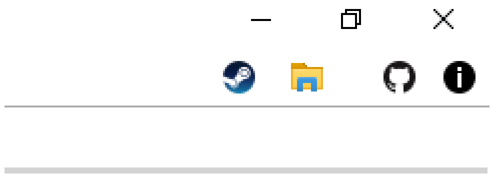


Step 6) If you installed correctly NMSMB will have started.

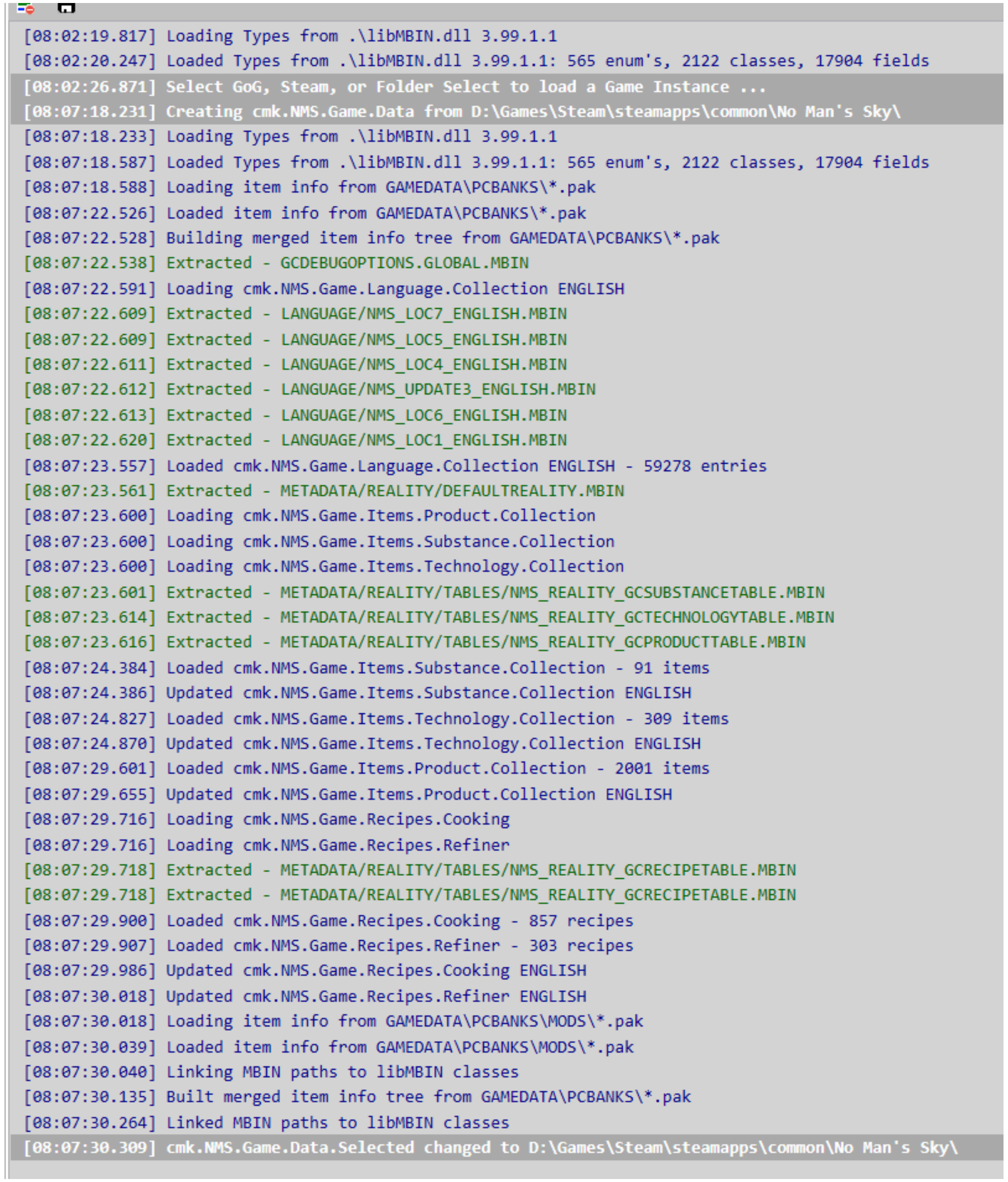


```
[08:02:19.817] Loading Types from .\libMBIN.dll 3.99.1.1
[08:02:20.247] Loaded Types from .\libMBIN.dll 3.99.1.1: 565 enum's, 2122 classes, 17904 fields
[08:02:26.871] Select GoG, Steam, or Folder Select to load a Game Instance ...
```

Step 7) To load your game data click on the corresponding GoG, Steam, or Game Pass icon. If none of those are present then select the folder manually by clicking the Select icon.

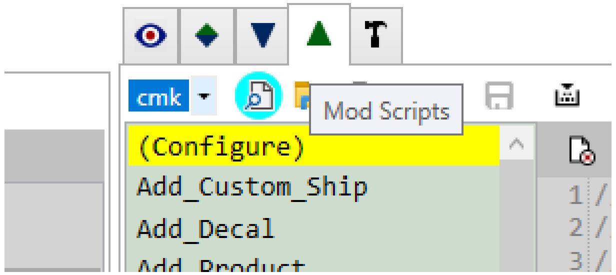


Step 7.1) When that is done NMSMB will start Loading the files.

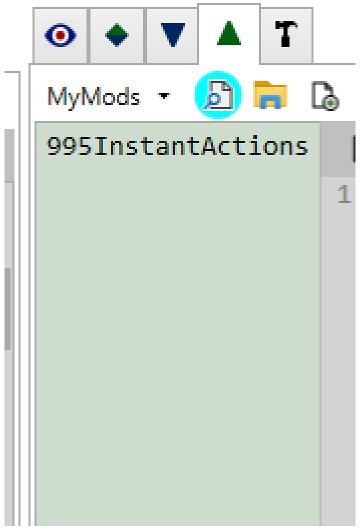


```
[08:02:19.817] Loading Types from .\libMBIN.dll 3.99.1.1
[08:02:20.247] Loaded Types from .\libMBIN.dll 3.99.1.1: 565 enum's, 2122 classes, 17904 fields
[08:02:26.871] Select GoG, Steam, or Folder Select to load a Game Instance ...
[08:07:18.231] Creating cmk.NMS.Game.Data from D:\Games\Steam\steamapps\common\No Man's Sky\
[08:07:18.233] Loading Types from .\libMBIN.dll 3.99.1.1
[08:07:18.587] Loaded Types from .\libMBIN.dll 3.99.1.1: 565 enum's, 2122 classes, 17904 fields
[08:07:18.588] Loading item info from GAMEDATA\PCBANKS\*.pak
[08:07:22.526] Loaded item info from GAMEDATA\PCBANKS\*.pak
[08:07:22.528] Building merged item info tree from GAMEDATA\PCBANKS\*.pak
[08:07:22.538] Extracted - GCDEBUGOPTIONS.GLOBAL.MBIN
[08:07:22.591] Loading cmk.NMS.Game.Language.Collection ENGLISH
[08:07:22.609] Extracted - LANGUAGE/NMS_LOC7_ENGLISH.MBIN
[08:07:22.609] Extracted - LANGUAGE/NMS_LOC5_ENGLISH.MBIN
[08:07:22.611] Extracted - LANGUAGE/NMS_LOC4_ENGLISH.MBIN
[08:07:22.612] Extracted - LANGUAGE/NMS_UPDATE3_ENGLISH.MBIN
[08:07:22.613] Extracted - LANGUAGE/NMS_LOC6_ENGLISH.MBIN
[08:07:22.620] Extracted - LANGUAGE/NMS_LOC1_ENGLISH.MBIN
[08:07:23.557] Loaded cmk.NMS.Game.Language.Collection ENGLISH - 59278 entries
[08:07:23.561] Extracted - METADATA/REALITY/DEFAULTREALITY.MBIN
[08:07:23.600] Loading cmk.NMS.Game.Items.Product.Collection
[08:07:23.600] Loading cmk.NMS.Game.Items.Substance.Collection
[08:07:23.600] Loading cmk.NMS.Game.Items.Technology.Collection
[08:07:23.601] Extracted - METADATA/REALITY/TABLES/NMS_REALITY_GCSUBSTANCETABLE.MBIN
[08:07:23.614] Extracted - METADATA/REALITY/TABLES/NMS_REALITY_GCTECHNOLOGYTABLE.MBIN
[08:07:23.616] Extracted - METADATA/REALITY/TABLES/NMS_REALITY_GCPRODUCTTABLE.MBIN
[08:07:24.384] Loaded cmk.NMS.Game.Items.Substance.Collection - 91 items
[08:07:24.386] Updated cmk.NMS.Game.Items.Substance.Collection ENGLISH
[08:07:24.827] Loaded cmk.NMS.Game.Items.Technology.Collection - 309 items
[08:07:24.870] Updated cmk.NMS.Game.Items.Technology.Collection ENGLISH
[08:07:29.601] Loaded cmk.NMS.Game.Items.Product.Collection - 2001 items
[08:07:29.655] Updated cmk.NMS.Game.Items.Product.Collection ENGLISH
[08:07:29.716] Loading cmk.NMS.Game.Recipes.Cooking
[08:07:29.716] Loading cmk.NMS.Game.Recipes.Refiner
[08:07:29.718] Extracted - METADATA/REALITY/TABLES/NMS_REALITY_GCRECIPETABLE.MBIN
[08:07:29.718] Extracted - METADATA/REALITY/TABLES/NMS_REALITY_GCRECIPETABLE.MBIN
[08:07:29.900] Loaded cmk.NMS.Game.Recipes.Cooking - 857 recipes
[08:07:29.907] Loaded cmk.NMS.Game.Recipes.Refiner - 303 recipes
[08:07:29.986] Updated cmk.NMS.Game.Recipes.Cooking ENGLISH
[08:07:30.018] Updated cmk.NMS.Game.Recipes.Refiner ENGLISH
[08:07:30.018] Loading item info from GAMEDATA\PCBANKS\MODS\*.pak
[08:07:30.039] Loaded item info from GAMEDATA\PCBANKS\MODS\*.pak
[08:07:30.040] Linking MBIN paths to libMBIN classes
[08:07:30.135] Built merged item info tree from GAMEDATA\PCBANKS\*.pak
[08:07:30.264] Linked MBIN paths to libMBIN classes
[08:07:30.309] cmk.NMS.Game.Data.Selected changed to D:\Games\Steam\steamapps\common\No Man's Sky\
```

Step 8) When it is done loading navigate to the Mod scripts pane. (UP arrow)

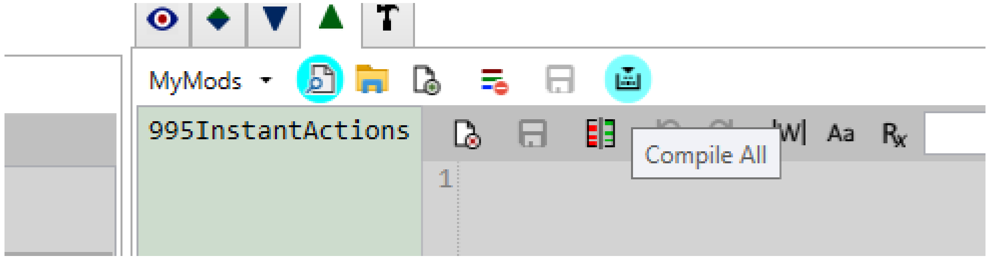


Step 8.1) Navigate to your folder. I used MyMods as an example.

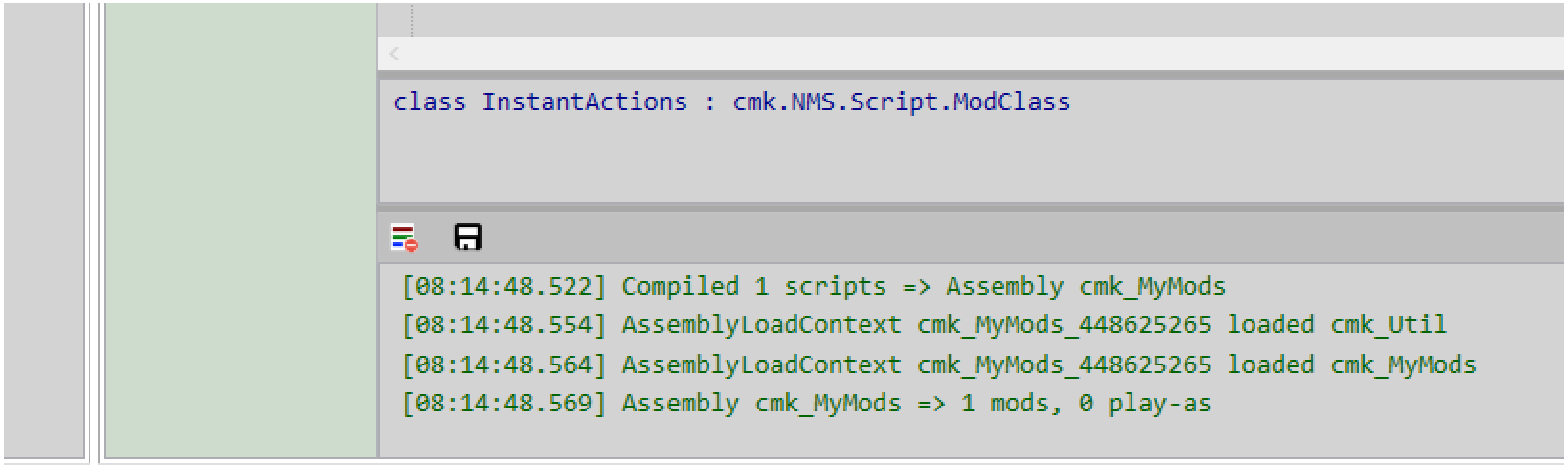


Here you will see all the scripts that you put in the MyMods folder

Step 9) Press the compile button.



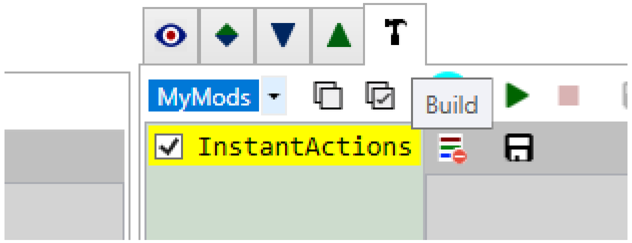
IF the script is fine you will get this.



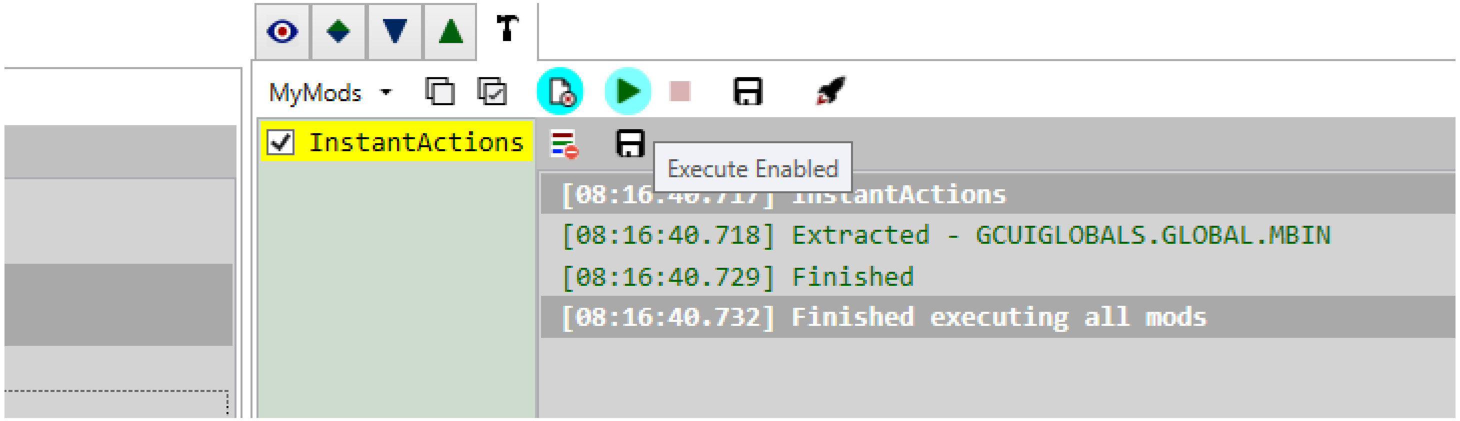
```
class InstantActions : cmk.NMS.Script.ModClass
```

```
[08:14:48.522] Compiled 1 scripts => Assembly cmk_MyMods
[08:14:48.554] AssemblyLoadContext cmk_MyMods_448625265 loaded cmk_Util
[08:14:48.564] AssemblyLoadContext cmk_MyMods_448625265 loaded cmk_MyMods
[08:14:48.569] Assembly cmk_MyMods => 1 mods, 0 play-as
```

Step 10) Navigate to the build pane (Hammer) and select your folder, in my case MyMods. Select the mods you want.

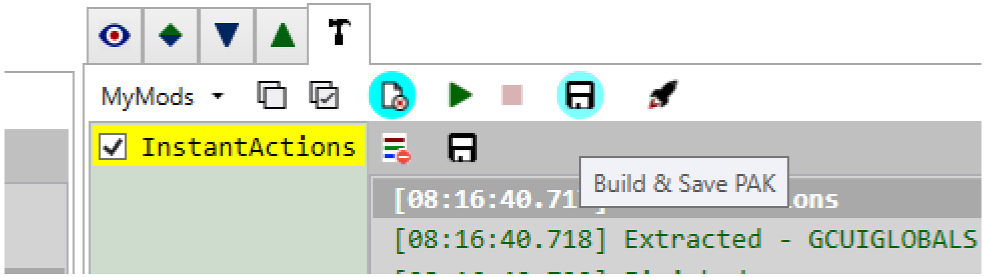


Step 11) Press execute enable (Right arrow) and wait until it finishes.



```
[08:16:40.717] InstantActions
[08:16:40.718] Extracted - GCUIGLOBALS.GLOBAL.MBIN
[08:16:40.729] Finished
[08:16:40.732] Finished executing all mods
```

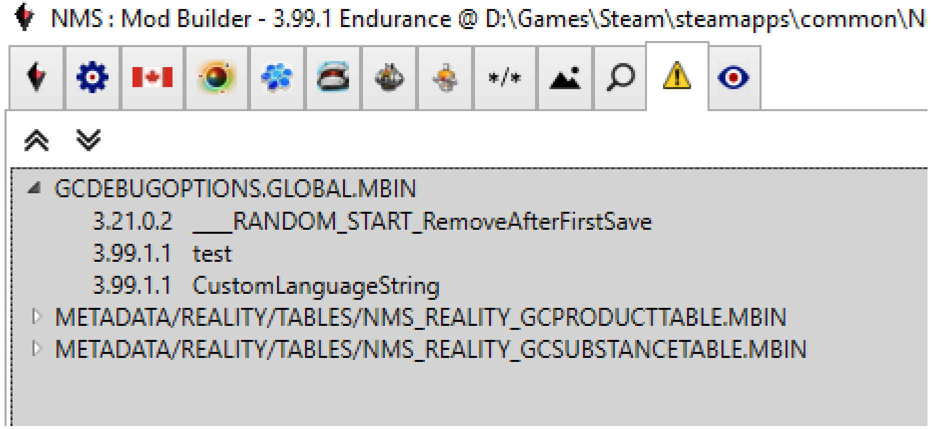
Step 12) Final Step press the Build and Save pak button (Floppy) and save it under whatever name you want. And you're ready to play now, click the space-ship icon to launch the game and minimize NMSMB.



```
[08:16:40.71    ons
[08:16:40.718] Extracted - GCUIGLOBALS
```

Merging and Mod conflict checking

- If you’re only using NMSMB mods your mod’s are merged automatically.
- NMSMB lets you quickly check for conflicts. If you’re using other .paks.
    - It watches the MODS folder and rechecks for any conflicts when any .pak files are added/changed.
    - If any conflicts are found it will display the confticts tab. (Danger triangle on left pane).
    - No conflicts tab is displayed if there are no conflicts found.
    - **(Advanced):** You can double-click the mod name to open a diff view of the game vs mod version of the item in conflict.



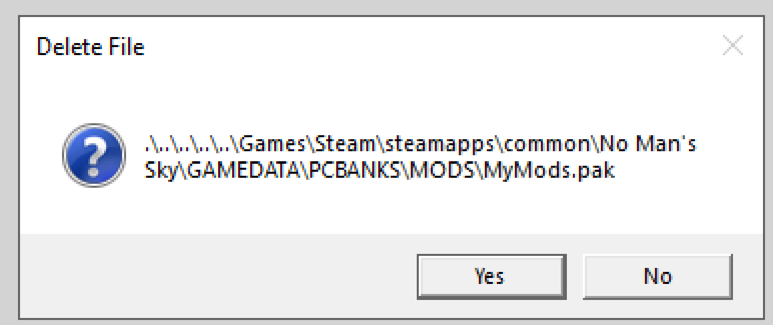
NMS : Mod Builder - 3.99.1 Endurance @ D:\Games\Steam\steamapps\common\N

```
GCDEBUGOPTIONS.GLOBAL.MBIN
    3.21.0.2   ___RANDOM_START_RemoveAfterFirstSave
    3.99.1.1   test
    3.99.1.1   CustomLanguageString
METADATA/REALITY/TABLES/NMS_REALITY_GCPRODUCTTABLE.MBIN
METADATA/REALITY/TABLES/NMS_REALITY_GCSUBSTANCETABLE.MBIN
```

Here you can see I have a conflict of GCDEBUGOPTIONS

- You will only see this IF you have conflicts.
- If this is not empty
    - Try to find or ask on NMS modding discord if a script exists.
    - If there is no such script. See next step.

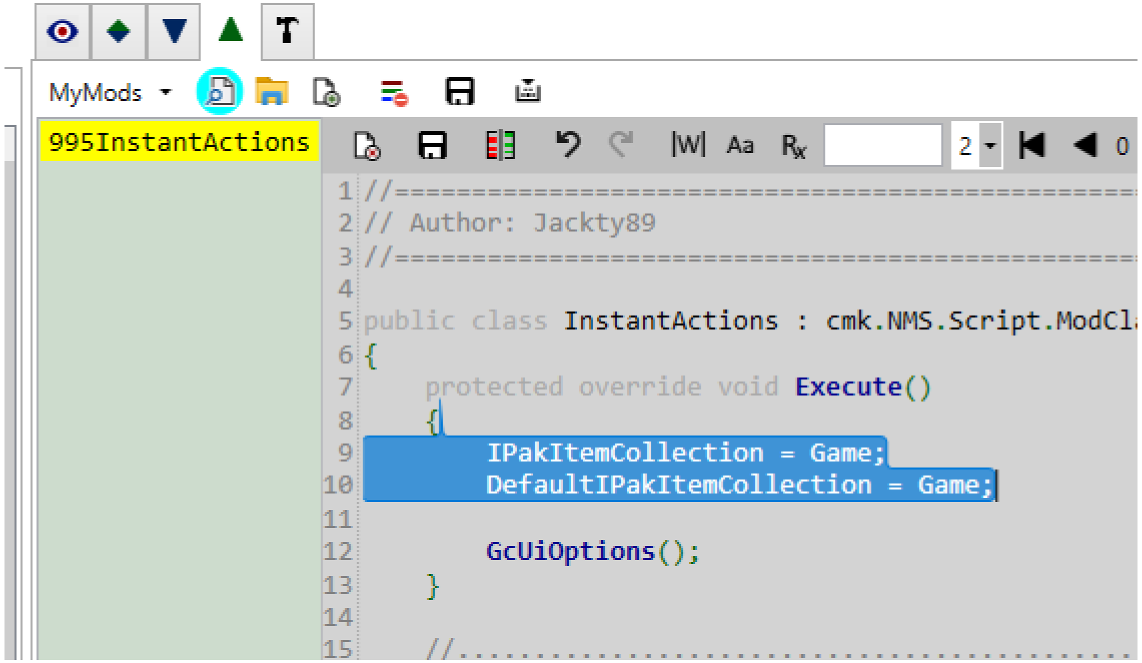
- **How to merge with an already existing paks/mods.**
    - Delete the pak you just created.
        - If your mod has not been renamed you'll get prompted this when building your mod again.



Delete File

.\..\..\..\..\Games\Steam\steamapps\common\No Man's Sky\GAMEDATA\PCBANKS\MODS\MyMods.pak

        - Otherwise manual delete is recommended.
    - Go the script that gets loaded first (you can see this when you build your mods). As for this example we only have 1 script.



```
//==============================================
// Author: Jackty89
//==============================================

public class InstantActions : cmk.NMS.Script.ModCl
{
    protected override void Execute()
    {
        IPakItemCollection = Game;
        DefaultIPakItemCollection = Game;

        GcUiOptions();
    }

    //.........................................
```

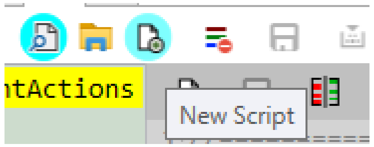
- Add these 2 lines in the Execute() section
    - IPakItemCollection = Game;
    - DefaultIPakItemCollection = Game;
- Recompile and build the mod
- Save the mod under a name **so it get loaded LAST.** Fe.: _MyMods
    - Adding a '_' to the front of your mod name is recommended. The more '_' you add the later it get loaded.
        - *So MyMods will load before _MyMods. _MyMods this will load before __MyMods. So__MyMods get loaded last.*
            - *So your load-order will be*
                - *Mod A*
                - *Mod B*
                - *_MyMods*
- NMSMB will still complain about conflict but if you followed the steps above it ***should*** be fine
- ***BE AWARE: If you add/change/delete any mods AFTER you did this patching it is recommended to redo the process.***

Profiles (advanced)

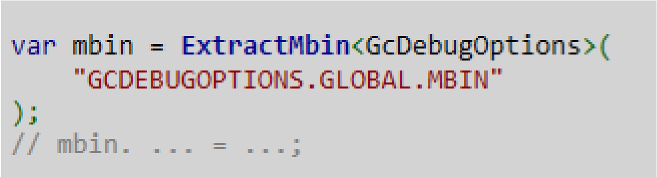
This is the final step on truly having custom mod list and for most users Profiles will not be needed. So you can skip this if your load order is static.

You can skip the (advanced) parts, if you don’t understand and code at all or what you see/read is incomprehensible.

- Click the NewScript button and name the file (configure)

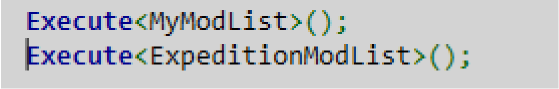


- Delete the contents of Execute()



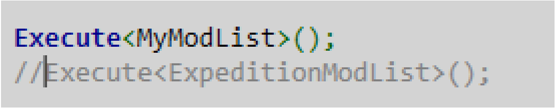
```
var mbin = ExtractMbin<GcDebugOptions>(
    "GCDEBUGOPTIONS.GLOBAL.MBIN"
);
// mbin. ... = ...;
```

- In this example I’ll teach you how set up 2 simple profiles
    - One for general use with all you mods enabled.
    - One for expeditions with only a select few mods enabled.
    - For each new profile we add Execute<Profilename>();



```
Execute<MyModList>();
Execute<ExpeditionModList>();
```

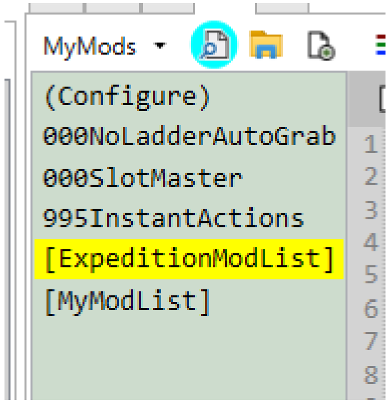
- Above we would load both Profiles at the same which is not good so add a // in front of the profile you’re **not going to use**.
    - Lets say you want to load MyModList in that case we add a // in front of your other profiles.



```
Execute<MyModList>();
//Execute<ExpeditionModList>();
```

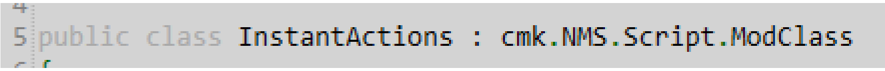
- Now we’re going to add 2 more scripts
    - One called [MyModList]
    - One called [ExpeditionModList]
    - This is the same for any profile: [Profilename]

- Clear contents of Execute() for your profiles.
- In Execute() we are now going to add **Script**<ModName>().IsExecutable = (True/False); for every mod you have.



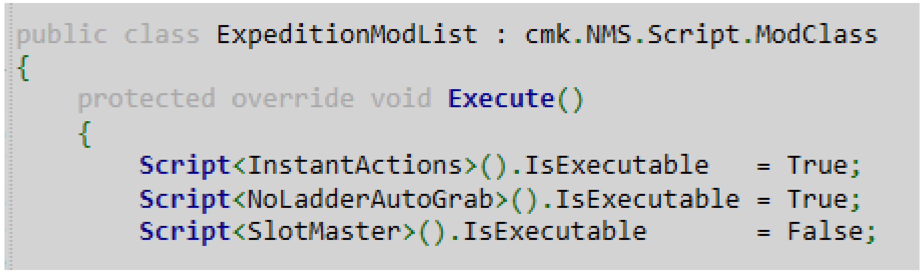
```
(Configure)
000NoLadderAutoGrab
000SlotMaster
995InstantActions
[ExpeditionModList]
[MyModList]
```

- In current setup I only have 3 scripts
  - Warning <ModName> is not always script name => so in this case <995InstantActions> is incorrect.
  - Make sure to see that the mod name is the right one
    - By clicking on the script and use the name that is next public class

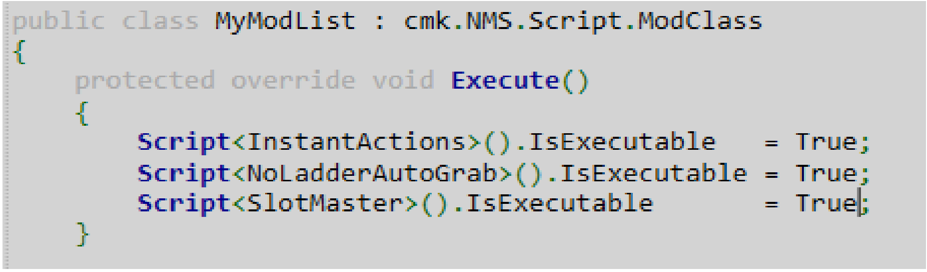


```
public class InstantActions : cmk.NMS.Script.ModClass
```

  - So <InstantActions> is the correct choice.
- So now you should have something like this



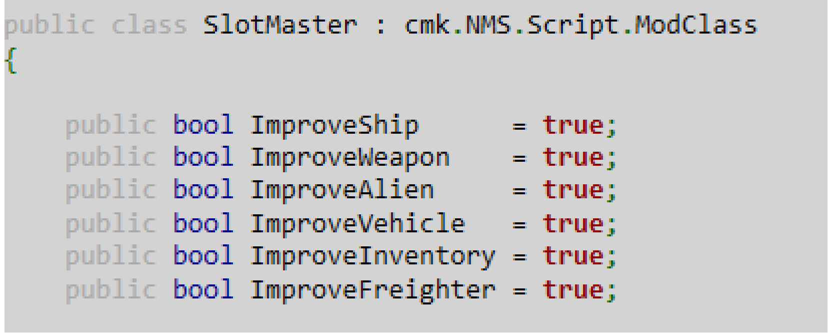
```
public class ExpeditionModList : cmk.NMS.Script.ModClass
{
    protected override void Execute()
    {
        Script<InstantActions>().IsExecutable    = True;
        Script<NoLadderAutoGrab>().IsExecutable  = True;
        Script<SlotMaster>().IsExecutable        = False;
```



```
public class MyModList : cmk.NMS.Script.ModClass
{
    protected override void Execute()
    {
        Script<InstantActions>().IsExecutable    = True;
        Script<NoLadderAutoGrab>().IsExecutable  = True;
        Script<SlotMaster>().IsExecutable        = True;
    }
```

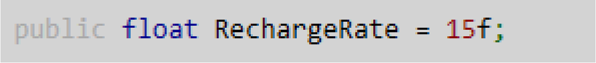
- No depending on the profile you selected only those mods will be loaded.
  - So no more manual (un)selecting mods,

- **(Advanced)** Some mods have public fields available that the user can edit to configure on how the mod runs. SlotMaster has this, so let’s take a look.
    - Inside SlotMaster at the top of the file under public Class: SlotMaster we see these fields.



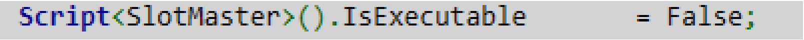
```
public class SlotMaster : cmk.NMS.Script.ModClass
{

    public bool ImproveShip      = true;
    public bool ImproveWeapon    = true;
    public bool ImproveAlien     = true;
    public bool ImproveVehicle   = true;
    public bool ImproveInventory = true;
    public bool ImproveFreighter = true;
```

    - These are field we can access on our profiles to fine-tune a mod.
    - Not all scripts will be bool (true/false). Other types might be public values fe.



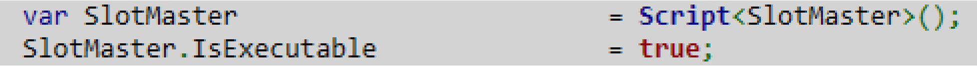
```
public float RechargeRate = 15f;
```

- This brings a small change inside our profile>
    - We have to edit:



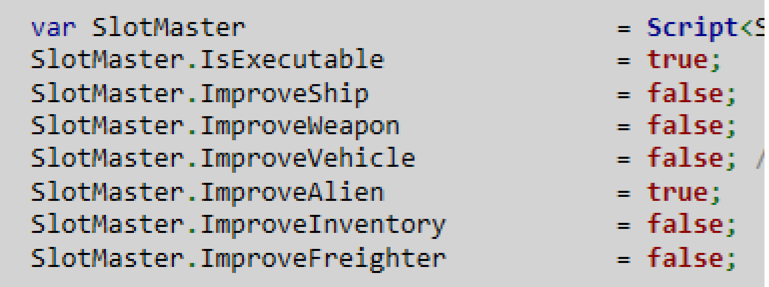
```
Script<SlotMaster>().IsExecutable       = False;
```

    - To this



```
var SlotMaster                                = Script<SlotMaster>();
SlotMaster.IsExecutable                       = true;
```

    -
        - The var SlotMaster can be a different name but for simplicity we’re using the same name as the Mod
- With this var in place we can now access and fine-tune those public fields to our liking.



```
var SlotMaster                                = Script<S
SlotMaster.IsExecutable                       = true;
SlotMaster.ImproveShip                        = false;
SlotMaster.ImproveWeapon                      = false;
SlotMaster.ImproveVehicle                     = false; /
SlotMaster.ImproveAlien                       = true;
SlotMaster.ImproveInventory                   = false;
SlotMaster.ImproveFreighter                   = false;
```

- This will override the values that are inside mod-script. Without you having to edit the mods itself.

For more questions on NMSMB visit the NMS-Modding discord.

NMS-Modding discord link: https://discord.gg/22ZAU9H