



TRADERPLUS DOCUMENTATION

By TheDmitri



TABLE

I.	Installation of TraderPlus	4
II.	How does the Bank system work?.....	5
1.	How to add a bank trader.....	6
2.	TraderPlusBankingConfig.json	7
3.	Bank account storage location and variables	7
III.	How does the Receipt system work?.....	8
IV.	How does the CarLock system work?	9
V.	How to convert your TraderConfig.txt to TraderPlusPriceConfig	11
VI.	Files's explanations.....	13
4.	TraderPlusGeneralConfig.....	13
5.	TraderPlusPriceConfig	18
6.	TraderPlusIDsConfig	22
7.	TraderPlusVehiclesConfig.....	23
8.	TraderPlusSafeZoneConfig	25
9.	TraderPlusStock_ID_X	28
10.	TraderPlusLogs	29
VII.	How to make static object works with TraderPlus	30
VIII.	Explanations regarding the UI Interface.....	31
IX.	TraderPlus items and purpose.....	33
X.	Useful tools to have	33
XI	Conclusion	34

INTRODUCTION

That mod was created as a challenge to myself. A trader mod is a lot of work and a lot to think in order to fill every gap possible. It took me 6 weeks to complete it to release it.

The main idea is simple: **“You can only buy what has been sold previously to the trader”**

I wanted to make it more deep than the current Trader available, made by Dr.Jones. I made that trader to have dynamic price based on stock among other cool features. I hope you'll enjoy it as much as I enjoyed making it. More will come over time, so it will be even better.

I want to thank:

- Dr.Jones for being the first in the community releasing a trader. He's the one who shows us how a trading system is important for the community and he deserves our gratitude for that.
- Chubby for providing the currencies to the mod
- Morty for the nice feedback during testing
- All the testers that came testing the mod
- All the modding community that allowed me to learn and be able to make that

I. Installation of TraderPlus

This part is dedicated to the installation of TraderPlus.

Normally, you should already have subscribed to the mod in the Steam Workshop and opened the mod folder to get access to that documentation file. So what do we do next :

Step 1: We copy the mod folder @TraderPlus to your DayZ Server Root.

Step 2: We go inside @TraderPlus\key and we copy it to paste it to your keys folder in your DayZ Server Root.

Step 3: We go inside @TraderPlus\ServerProfile and we copy the TraderPlus folder to paste it in your profile folder located normally in your DayZ Server Root. (that's the folder where you can find RPT files, Logs files and Crashlog among other mod config files such as zomberry, vpp, expansion etc...)

Step 4: We have finished the basic step to make the mod work. The mod is ready to be launched for testing in Chernarus.

Step 5: Make sure that you have inside your TraderPlus folder the following folder:

Nom	Modifié le	Type	Tail
TraderPlusBankDatabase	28/04/2021 15:25	Dossier de fichiers	
TraderPlusConfig	28/04/2021 17:18	Dossier de fichiers	
TraderPlusDatabase	16/04/2021 17:48	Dossier de fichiers	
TraderPlusLogs	28/04/2021 17:18	Dossier de fichiers	

If you don't have some, you need to create them manually.

"All right Dmitri, but I don't run Chernarus, how can I make the mod work for my custom map ?"

So, dear user, if you're in that case, you'll have to edit my config file in order to set your traders' locations. I recommend to use any modding tools such as [DayZ Editor](#) or [Custom Offline mod](#) to create your traders' locations with objects.

I'll detail more about that in III.1.

II. How does the Bank system work?

Yes, indeed, there is a banking system in TraderPlus that allow players to store their money into a database. For now, the banking system works only with basics features such as:

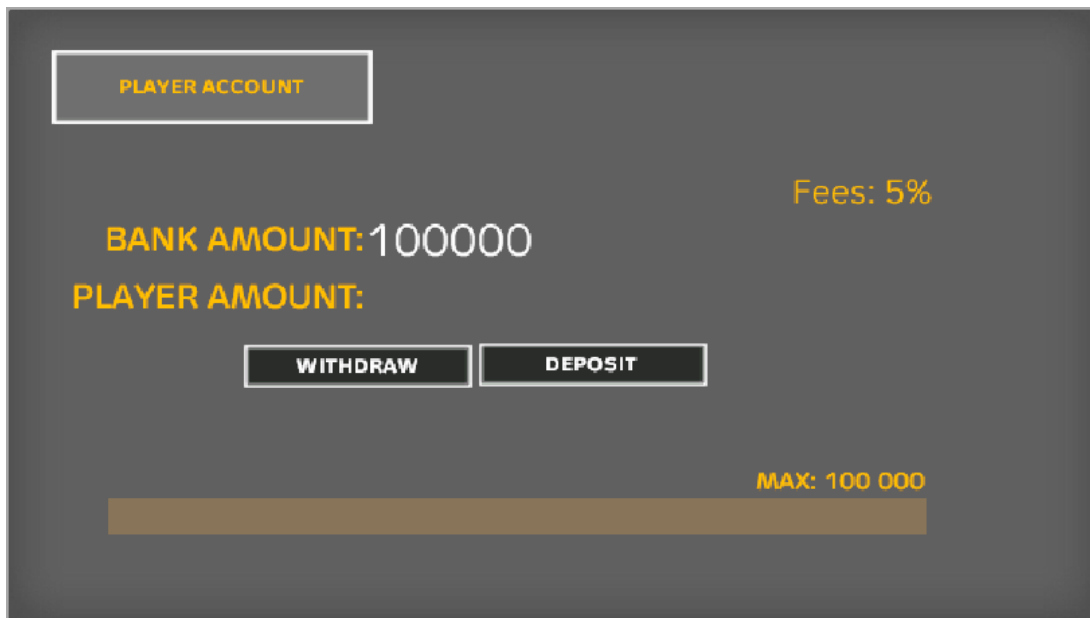
- Withdraw money from bank account
- Deposit money to bank account

Later, the following features will be included:

- Ability to create a clan account
(defined price to create one, permission settings for leader member to invite, kick, etc...)
- Ability to transfer money from one player's bank account to another

"Yes but Dmitri, there is already some bank mod in the workshop, what's the point of using this one ?"

Well, dear user, having a bank system inside the trader allows me to fully control it and makes it more adaptable for additional features if needed. Moreover, it reduces the amount of mod you need to load into your server and reduce the chance of incompatibility between mods.



This is a picture of the default interface, as you can see, you have:

BANK AMOUNT: Amount of money owned by the player in the bank

PLAYER AMOUNT: Amount of money owned by the player in his inventory. You can also edit it to enter the amount you want to deposit/withdraw.

A PROGRESS BAR: It will be filled based on the max capacity of your bank account.

FEES: Value in %, that will be deduce from withdraw or deposit and lost. Call that fees from the bank.

Again, everything is configurable.

If you don't want the bank system, you can install in your server **@DisableBankTraderPlus**. if you want to change the fees, that's possible in the `TraderPlusBankingConfig.json` if you want to change the max money amount that can be stored in the bank, that's also in the config.

If you want to translate the UI, that's also possible.

1. How to add a bank trader

This part is gonna be easy if you already know how to add trader into the game.

If you don't know how to add a trader into the game, I invite you to check the VI.2 `GeneralConfig.json`. Then come back when you're familiar with that part.

All right, so, to add a trader bank into the game, you're gonna choose your object. It can be a

static object such as a ATM or a real NPC (SurvivorM_Peter). Whatever you want to put, you can. (Keep in mind that static object must inherit BuildingSuper to work, there is a tutorial in **VI.2 Traders**)

Then just do as you would do to add a Trader into the config.

There is only one change to make, that's the ID. **In order to make your trader a bank trader, you must set Id = -2.**

2. TraderPlusBankingConfig.json

So let's explain how this config file works:

"IsCreditCarNeededForTransaction": That's a Boolean variable that allows you to decide if the player need a credit card to access the bank menu. Set it to 1 to require the card for opening the menu, set it to 0 otherwise.

"TransactionFees": That's a float variable dealing with the fees taken by the bank each time the player is doing a transaction. By default, it's set at 0.05.

A friendly reminder, the value to set here must be set between 0.0 and 1.0 because it will be used as a %.

So 0.05 = 5%, 0.0 = 0% and 1.0 = 100%.

"DefaultMaxCurrency": That's an int variable dealing with the maximal amount a player can store inside his bank account. Keep in mind you can always change the max value for a specific player by going directly into their bank file.

By default, the amount is set to 1000000.

"TextUI": That's a string variable dealing with the translation of the UI, Just replace the sentences if you need to.

3. Bank account storage location and variables

As other bank system, you're able to get access to each player's bank account file. Thoses files are saved inside TraderPlus\TraderPlusBankDatabase\.

Nom	Modifié le	Type	Tail
TraderPlusBankDatabase	28/04/2021 15:25	Dossier de fichiers	
TraderPlusConfig	28/04/2021 17:18	Dossier de fichiers	
TraderPlusDatabase	16/04/2021 17:48	Dossier de fichiers	
TraderPlusLogs	28/04/2021 17:18	Dossier de fichiers	

Each file start by Account_ followed by the SteamID64 of the player.

Nom	Modifié le	Type
Account_76561198047475641.json	28/04/2021 17:22	Fichier JSON

So if you want to access someone account file, you'll need to know their SteamID64 or make a file search with their Steam Name.

Then inside, you'll find thoses informations:

```
{
  "Name": "TheDmitri",
  "MoneyAmount": 998615,
  "MaxAmount": 1000000
}
```

“**Name**”: that’s the player’s nickname on steam.

“**MoneyAmount**”: that’s the amount of money stored.

“**MaxAmount**”: that’s the maximal money that this player can store.

III. How does the Receipt system work?

So as you know, trader sell items but also vehicles. Dealing with trading vehicle can be resolve from many point of view such as having a parking spot in order to park the car to trade it. I decided to make something a bit different. In order to sell or buy car, you'll need a receipt that will act as a proof of ownership.

In order to make the receipt system work with your car, make sure your car is set in the VehicleConfig with its attachments. If you don't, the car will be popping on your head and you'll come to me saying my shit doesn't work 😊

To get a receipt from a car, you'll have to "pack it" with a reinforced wrench (classname: TraderPlusWrench).

So once packed, you'll be able to go to the car seller trader and sell it as a car.

There are multiple conditions regarding packing a car into a receipt:

=>There is a variable called IsReceiptTraderOnly in the GeneralConfig. If set to 1, you'll be able to pack and deploy your car only inside a safezone. If you set it to 0, that can be done from the entire map.

In addition to that:

-If you use the CarLock system, only the owner or the person who has the password can pack it. It will take between 15 and 30s to pack. (*Check III. for how to disable the carlock system*).

-if you don't use CarLock because you wanted to keep using your key system from MuchCarKey for example, then anyone can pack a car that is UNLOCK. (it means locked car can't be packed). It will take 2 min to pack.

Like mentioned above, you're also able to deploy it to get your car. There are script securities to make sure that we can't deploy a car on a player or on a roof.

You may think about the car's attachments, how does it works ?

Well, it's good and bad.

You have to make sure your car classname is well written in the VehicleConfig in order to be able to use the receipt system. Also, all attachments in that file will spawn everytime, no matter what. So it means that packing a car that miss attachments or fuel will give a full car once deployed again. That means players can actually try to glitch the system by sellings car parts. That's why I recommend not to allow players to sell car parts.

IV. How does the CarLock system work?

-You'll need to attach an item called **CarLock** in order to secure and own the car. Once a 8 digits (only numbers, not char) password set, you'll be able to lock, unlock your car forever. Password need to be set only once. Your friend can also enter the password once to lock unlock the car.

Password data are stored into a client database accessible in Documents\DayZ\CarLock\CLDatabase\CarLock_X.json.

you may find different json file. There is actually one for each car owner and all the password possible that worked once are inside. You're free to remove thoses files from time to time if it's too big. That will generate a new file but you'll loose all your passwords.

=>To get your own password file, you need to compare the right part of your SteamID64 to the id number of the file to find yours.

So regarding the CarLockSystem:

the CarLock can be detached with a plier if you're the owner or an admin. (to set an admin, check SafeZoneConfig explanations).

-A CarLockPick can be used to attempt to steal the car by removing the CarLock. According to LockPickChance (0.0 to 1.0 ~ 0% to 100%), you'll get the CarLock and a alarm will be displayed for 5 min that can be heard in a 500 meters radius.

-When buying a car at the trader, the car will be already locked and only a password will be required to finished the procedure.

Warning: That system needed variables on CarScript class so it will corrupted all your cars at first load. It may remove others key mod system so if you don't want to use the CarLock system, you'll be able to deactivate it by adding/repacking a small mod to your server list. It's actually a define value that disable all CarLock feature. Keep in mind that without the car lock system, car won't get locked after being deployed at the trading zone so carefull !! link of the mod:

<https://steamcommunity.com/sharedfiles/filedetails/?id=2458852374>

V. How to convert your TraderConfig.txt to TraderPlusPriceConfig

That part is dedicated to user who already use Dr.Jones trader mod and would like to convert their config to TraderPlus config.

I'm happy to announce that most of the trader config can be converted:

What can be converted:

- TraderConfig.txt
- TraderVehiclesParts.txt

What can not be converted:

- TraderObjects.txt
- TraderVariables.txt

So basically the most important file that is TraderConfig.txt is converted so you won't have to do the price config again. And that, I bet, is making you happy because you're as lazy than I am 😊.

Let's stop the talking and get into the subject:

The conversion will generate 3 files located at DayZServer\profile\TraderPlus\TraderPlusConfig: TraderPlusPriceConfig.json, TraderPlusIDsConfig.json, TraderPlusVehiclesConfig.json.

The TraderPlusPriceConfig will take your TraderConfig.txt, add each classname, trade quantity, buyprice and sellprice to add it to TraderPlusPriceConfig to the dedicated category it was put in. In addition to that, new variables will be added for each products:

-a coefficient: between 0.0 and 1.0

-a maxstock: it will be the nominal value of that item indicated in your types.xml (for ammunition, it will be multiplied by the average number of ammo that spawn in map so it can be higher than current nominal value)

So from that line for example:

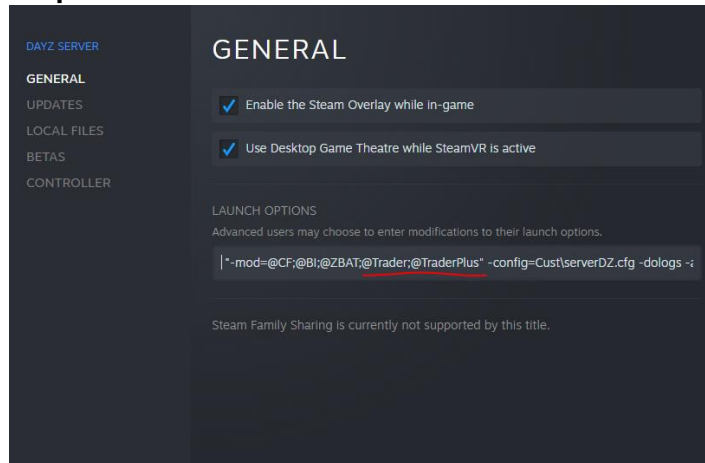
classname,	tradeqty,	buyprice,	sellprice
CombinationLock,	*	700,	300

it will be converted to something like this:

classname,	coefficient,	maxstock	tradeqty,	buyprice,	sellprice
CombinationLock,	0.97309	,60	,0	,700	,300

Now that you understand a bit more what's happening, let's see how we can convert Trader config to TraderPlus config.

Step 1: Make sure to load @Trader before @TraderPlus in your start load order.



Step 2: Go to the TraderPlusConfig folder and open TraderPlusGeneralConfig.json and set to 1 the first variable called : **"ConvertTraderConfigToTraderPlus": 1**

It should look like this:

```
{
  "ConvertTraderConfigToTraderPlus": 1,
  "TraderUIText": "It cannot be sold !,Trade Quantity is superior to item quantity !,Trade",
  "MaxItemsStockPerId": 500,
  "MaxPlayerPerID": 5,
  "IsReceiptTraderOnly": 0,
  "AcceptedStates": {
    "AcceptWorn": 1,
    "AcceptDamaged": 1,
    "AcceptBadlyDamaged": 1
  },
  "Currencies": [
    {
      "ClassName": "TraderPlus_Money_Dollar500,TraderPlus_Money_Euro500,TraderPlus_Money",
      "Value": 500
    },
    {
      "ClassName": "TraderPlus_Money_Dollar100,TraderPlus_Money_Euro100,TraderPlus_Money",
      "Value": 100
    },
    {
      "ClassName": "TraderPlus_Money_Dollar50,TraderPlus_Money_Euro50,TraderPlus_Money",
      "Value": 50
    }
  ]
}
```

Step 3: Start your server, go back to the TraderPlusConfig folder and wait to see TraderPlusPriceConfig.json, TraderPlusIDsConfig.json and TraderPlusVehiclesConfig.json been modified (check the modification date). It should be done 1 to 2 minutes after server started. Once it's done. Stop your server and set to 0 **"ConvertTraderConfigToTraderPlus": 0,**

Step 4: Biggest part is done, congrats ! Now I suggest that you take a look at those newly modified files and see if everything seems to be ok. Normally every category you set in TraderConfig can be found in the PriceConfig, every traders ids can be found in IDsConfig and every vehicles parts can be found in VehiclesConfig.

Step 5: Now only thing remaining it to define where your trader are located in addition to your objects if you have some. I'll explain that in III.1.

VI. Files's explanations

In this part, I'll detail how to set and the purpose of each variables for each config files. You're welcome to go directly to what you're looking for but I strongly advise you to fully read the document at least one time so you've got the big picture of how the mod works.

4. TraderPlusGeneralConfig

Let's begin with the general config file, you'll be able to set everything that is linked to the general's mod behavior such as TraderMoney, UI Translation, Accepted States, Traders Positions and Objects Positions.

So once you've opened the file, you'll see something that contains:

ConvertTraderConfigToTraderPlus	Boolean (true : 1 ; false : 0)
TraderUIText	String ("blablabla")
MaxItemsStockPerId	Integer
MaxPlayerPerID	Integer
IsReceiptTraderOnly	Boolean (true : 1 ; false : 0)
LockPickChance	float
AcceptedStates	<Boolean,boolean,boolean> class
Currencies	Array <String, integer> class
Traders	Array <Integer,string,string,string,vector,vector> class
TraderObjects	Array<string, vector, vector> class

So let's start the explanations of the purpose of each one:

"ConvertTraderConfigToTraderPlus": That Boolean var allow you to convert Dr.Jones TraderConfig.txt and TraderVehiclesParts.txt to TraderPlusPriceConfig.json, TraderPlusIDsConfig.json and TraderPlusVehiclesParts. You're invited to check out II. Regarding that part.

"TraderUIText": That string var allow you to change the language of the formulation in your own language. I could have done a csv file sure, but since it takes time to get a native speaker for making each translation, that way, you can already change it yourself.

"MaxItemsStockPerId": That integer allow you to define the product limit in stock for each trader id.

Example:

"MaxItemsStockPerId": 5,

Stock will be reach when the stock has for example:

Classname, stock, health

SKS	15	pristine
SKS	5	damaged
Mosin	2	pristine
AK74	8	worn
AK101	3	badly damaged

"MaxPlayerPerID": That integer allow you to define how many survivor can trade with one trader ID at a time. That way, we limit performances drop when trader is full of survivor doing trading.

"IsReceiptTraderOnly": That Boolean allow you to define if you want the receipt system for the car to work in the entire map or only inside a trading area.

"LockPickChance": If you want to you the carlock system included in the mod, you'll have the possibility to define the chance value of successful lockpicking a car.

"AcceptedStates": {

"AcceptWorn": boolean var that allow you to decide if you want to accept item is such condition to be trade. 0: disable ; 1: enable

"AcceptDamaged": same than above

"AcceptBadlyDamaged": same than above

}

"Currencies": That array of <string, integer> allows you to decide what kind of currency you want to use for the mod. You can add a line if needed.

Example:

I want to add Lat_25 as a currency:

25 is between 20 and 50 so I'll add a line there as so:

```
{
    "ClassName":
    "TraderPlus_Money_Dollar50,TraderPlus_Money_Euro50,TraderPlus_Money_Rub
le50 ",
    "Value": 50
},
{
    "ClassName": "Lat_25",
    "Value": 25
},
{
    "ClassName": "TraderPlus_Money_Dollar20,TraderPlus_Money_Euro20",
```

```
    "Value": 20  
  },
```

Now that you understood how to add a currency value that wasn't already set, let's check how to add a currency that already is set.

I like having multiple currencies so I can either pay with Dollars or Euros or Rubbles. I want to add Lat_50 to the already existing currency.

I just have to add to the string Lat_50 as so:

```
"TraderPlus_Money_Dollar50,TraderPlus_Money_Euro50,TraderPlus_Money_Ruble  
50,Lat_50",
```

Keep in mind, that the first currency set in each string is the primary currency. It means that if you sell an item, the trader will give you that primary currency. In our case, that's dollars !

"Traders": This variable is the traders array. That's where you're gonna set up where and what spawn as trader in the map.

You can either set a Object npc (a sign for example) or a real player like SurvivorM_Peter.

So first of all, I need to define the ID: **"Id"**:

That id will represent what the npc is selling. Check out TraderPlusIDsConfig.json to get the id value.

"Hey Dmitri, can I have different trader with same Id ?":

Yes you can, it will mean they will all share the same stock.

How to know the id value:


```

{
  "IDs": [
    {
      "Categories": [
        "Vegetables",
        "Packaged Food",
        "Meat",
        "Animal Parts",
        "Drinks",
        "Medical Supplies",
        "Medications",
        "Money Exchange"
      ]
    },
    {
      "Categories": [
        "Tools",
        "Heavy Tools",
        "Electronics",
        "Fire & Lights",
        "Cooking & Hunting Supplies",
        "Hardware Supplies",
        "Seeds & Lime"
      ]
    },
    {
      "Categories": [
        "Melee",
        "Sidearms",
        "Rifles",
        "Shotguns",
        "Submachine Guns",
        "Assault Rifles",
        "Sniper Rifles",
        "Grenades"
      ]
    },
    {
      "Categories": [
        "Ammunition",
        "Magazines",
        "Handguards",
        "Buttstocks",
        "Optics",
        "Suppressors",
        "Compensators",
        "Bayonets"
      ]
    }
  ]
}

```

Handwritten annotations in red:

- Next to the first category list: $ID = 0$
- Next to the second category list: $ID = 1$
- Next to the third category list: $ID = 2$
- Next to the fourth category list: $etc...$ and $So on...$

Once you know what the trader should be selling, you need to define it's classname:

"Name": string var that correspond to the classname of the entity you want to use as trader, can be a static object or survivor.

(if you've got a special static object you want to use, make sure that static object extends BuildingSuper)

All right Dmitri, but how do I make my static object extends from BuildingSuper ? I'm a noob in scripting :/. Check **VII**.

"GivenName": string var that correspond to the name shown in the trader UI.

"Role": string var that correspond to the role shown in the trader UI

You can put Weapon Trader, Car Trader, what ever you want.

"Position": vector var that correspond to the position in the map of the trader.
(Can be obtained with any modding tool such as DayZ Editor or Community offline mod)

"Orientation": vector var that correspond to the orientation in the map of the trader.
(Can be obtained with any modding tool such as DayZ Editor or Community offline mod)

"Clothes": array <string> that allows you to add Attachments to the trader is it's a Survivor.

Example:

```
"Clothes": [  
  "NBCHood",  
  "NBCJacket",  
  "NBCPants",  
  "NBCBoots",  
  "M4A1"  
]
```

(For my sake and yours, please **never add a comma at the last element in the array**, in addition don't forget to use a json validator if something doesn't work)

"TraderObjects": array <string, vector, vector> var that allows you to add custom object to spawn with the trader.

ObjectName: string var of the classname of the object you want to spawn

"Position": vector var that correspond to the position in the map of the object.
(Can be obtained with any modding tool such as DayZ Editor or Community offline mod)

"Orientation": vector var that correspond to the orientation in the map of the object.

Now we're done explananing the GeneralConfig, let's move on the big one:
PriceConfig.

5. TraderPlusPriceConfig

That file is really important because he's the one that will allow you to edit your prices, your stock, how the dynamic price works, etc...

So let's start by opening the file and check how it looks. Normally you should see that products are located in different categories. Each category can contains as many products as you want.

Each product contains 6 elements that are :

-Classname -Coefficient -Max Stock -Trade quantity – BuyPrice – SellPrice

Keep in mind that you'll be able to do:

-Product that have dynamic price based on max stock value

-Product that have a static price and a max stock value

-Product that have a static price and unlimited stock (like Dr.Jones trader)

So let's start by trying to make a dynamic price based on max stock value

1.1. Make a product that have a dynamic price based on max stock value

Again, you've got two possibilities:

-You can let the mod calculate for you the maxstock value based on the nominal value in your types.xml in addition to the coefficient based on the lowest price desired when the stock is reached.

-You can set it manually by playing with the PriceCalculator.xls

If you're lazy, like me, this is how you're gonna make it auto calculate:

let's take an example: I want to add a Weapon form Morty's Weapon: the HK 416A7

Classname, coefficient, maxstock, trade quantity, buyprice, sellprice

HK 416A7, 0.4, -2, 1, 10000, 0.8

So let's try to understand what I set.

-I want the HK to have a **maxstock** based on the nominal value so I put **-2**. (don't try to understand that, just put -2)

-I want the lowest price for the HK to be 40% of the buyprice. (buyprice is always the highest price when the stock is low), so I put **0.4** (= 40%) in the **coefficient**.

-I set to 1 the **trade quantity** because it's a gun and that's actually the max quantity I can set. (You can't put a tradeqty bigger than the max qty of the item or superior to the max count for ammunition => 20 for 7.62x39mm)

You can also set tradeqty to **-1** if you want it to be max qty of the item or to a coefficient such as **0.75**; that mean you can trade if the product has at least 75% of quantity.

Let's talk a bit about the different scenarios we can meet while working on a priceconfig based on the products for the Tradeqty.

There is products without quantity such as **weapons**:
Tradeqty **must** be equal to **0** or **1**.

There is product with quantity such as **ammunitions**:
Tradeqty **can** be equal to **1** (minimal value) to **maxqty (20 for 7.62x39mm)**

There is product without quantity but that contains ammunicions such as **mag**
Tradeqty **must** be equal to **0**.

There is product with quantity such as **consumable** like apple, meats etc...
Tradeqty **can** be equal to **1** or **maxqty** or a **coefficient**.
coefficient will represent a % of maxqty. (useful when you want to sell a % of wolfmeat).

There is product with quantity that contains water such as cooking pot or canteen.
Tradeqty **can** be equal to **0** (we don't care of the qty inside) to **maxqty (item filled completely with water)** **Keep In mind that value between 0 and maxqty will sell portion of the item until quantity reach 0.**

-I set the **buyprice** to **10000** as the highest price. To disable the ability to buy, you can set it to **-1**.Keep in mind that a buyprice below 8 can't be dynamic

-I set the **sellprice** to **0.8**. That mean it will be equal to **80%** of the buyprice. You can also put a number such as **8000** or **7500** or whatever. The two are possible. To disable the ability to sell, you can set it to **-1**.Keep in mind that a sellprice below 8 can't be dynamic in case buyprice is disable.

Once done, I can either config a new product or make the auto calculation happen.

To activate the autocalculation, here is what you have to do:

Step 1: There is a variable called "**EnableAutoCalculation**": in the price config. You need to set it to 1.

Step 2: Then start your server and wait a minute. Once the price config has been modified you can stop your server.

Step 3: Disable the "**EnableAutoCalculation**" with **0**.

Step 4: Check out the line you added and normally you should see that it has changed 😊

Now, if you want to set your product with a custom stock and a custom coefficient then you can open and run the PriceCalculator.xls located in the mod folder in the Folder called Tool.

Once open, you'll be able to set it up to your product values and put your own max stock and your own coefficient (between 0.0 and 0.99999999), I suggest to start with 0.993 and increase with a step of 0.001 to get interesting result.

Once done, you'll have three more variables called

stock = n	DynamicBuyPrice	DynamicSellPrice
20	1	1

You can define for a potential stock of n and get the buyprice and sellprice when the stock is equal to n.

Once your satisfied, put the chosen value into the priceconfig, start your server and enjoy !

1.2. Make a product that have a static price with a max stock value

So let's be back to our previous example with the the HK 416A7

Classname, coefficient, maxstock, trade quantity, buyprice, sellprice

HK 416A7, 1, 10, 1, 10000, 0.8

As you can see, the coefficient is set to 1, that way the price will always be buyprice (10000) and sellprice ($0.8 \times 10000 = 8000$).

You can define the stock based on your nominal value, you can either do a. and then put coefficient equal to 1 or go directly to your types and look for nominal value.

As before, tradeqty equal to 1 because that's the max we can get.
keep in mind that thoses are the possibilities for the trade qty variables:

There is products without quantity such as **weapons**:

Tradeqty **must** be equal to **0** or **1**.

There is product with quantity such as **ammunitions**:

Tradeqty **can** be equal to **1** (minimal value) to **maxqty (20 for 7.62x39mm)**

There is product without quantity but that contains ammunicions such as **mag**

Tradeqty **must** be equal to **0**.

There is product with quantity such as **consumable** like apple, meats etc...

Tradeqty **can** be equal to **1** or **maxqty** or a **coefficient**.

coefficient will represent a % of maxqty. (useful when you want to sell a % of wolfmeat).

There is product with quantity that contains water such as cooking pot or canteen.

Tradeqty **can** be equal to **0** (we don't care of the qty inside) to **maxqty (item filled completely with water)** **Keep In mind that value between 0 and maxqty will sell portion of the item until quantity reach 0.**

1. n value that we need to trade (can't be higher than max item qty)
2. -1 will make it max quantity
3. x coefficient (float) if we want a % of max qty

-I set the **buyprice** to **10000** as the highest price. To disable the ability to buy, you can set it to **-1**. Keep in mind that a buyprice below 8 can't be dynamic

-I set the **sellprice** to **0.8**. That mean it will be equal to **80%** of the buyprice. You can also put

a number such as 8000 or 7500 or whatever. The two are possible. To disable the ability to sell, you can set it to -1. Keep in mind that a sellprice below 8 can't be dynamic in case buyprice is disable.

1.3. Make a product that have a static price with unlimited stock

So let's be back to our previous example with the the HK 416A7

Classname, coefficient, maxstock, trade quantity, buyprice, sellprice

HK 416A7, 1, -1, 1, 10000, 0.8

If you looked at the two previous a. and b. you'll see that maxstock is set to -1. So you can conclude than -1 will make the stock unlimited.

As before, tradeqty equal to 1 because that's the max we can get.
keep in mind that thoses are the possibilities for the trade qty variables:

Let's talk a bit about the different scenarios we can meet while working on a priceconfig based on the products for the Tradeqty.

There is products without quantity such as **weapons**:
Tradeqty **must** be equal to 0 or 1.

There is product with quantity such as **ammunitions**:
Tradeqty **can** be equal to 1 (minimal value) to **maxqty (20 for 7.62x39mm)**

There is product without quantity but that contains ammunitions such as **mag**
Tradeqty **must** be equal to 0.

There is product with quantity such as **consumable** like apple, meats etc...
Tradeqty **can** be equal to 1 or **maxqty** or a **coefficient**.
coefficient will represent a % of maxqty. (useful when you want to sell a % of wolfmeat).

There is product with quantity that contains water such as cooking pot or canteen.
Tradeqty **can** be equal to 0 (we don't care of the qty inside) to **maxqty (item filled completely with water)**. **Keep In mind that value between 0 and maxqty will sell portion of the item until quantity reach 0.**

1. n value that we need to trade (can't be higher than max item qty)
2. -1 will make it max quantity
3. x coefficient (float) if we want a % of max qty

-I set the **buyprice** to 10000 as the highest price. To disable the ability to buy, you can set it to -1. Keep in mind that a **buyprice below 8 can't be dynamic**

-I set the **sellprice** to 0.8. That mean it will be equal to 80% of the buyprice. You can also put a number such as 8000 or 7500 or whatever. The two are possible. To disable the ability to sell, you can set it to -1. Keep in mind that a **sellprice below 8 can't be dynamic** in case buyprice is disable.

If you want that item to automatically be added to your stock, You'll have to enable AutoCalculation and follow step like 1.2.

That should be all for that file, hope it was understandable, let's keep up and check how the TradeIDsConfig works.

6. TraderPlusIDsConfig

All right, now that we have seen general config and price config, let's take a look at the TraderPlusIDsConfig.

That file shouldn't be too hard to understand. It's basically where you're gonna define for each trader ID, what they are selling by adding the categories to each id.

As shown earlier, first one in the array is ID 1, then ID 2, then ID 3 etc... and so on...

normally just by seeing the file you should get it but in case you're tired because it's late when you're reading this, let's do an example:

I want to add a category called Morty's Weapons to my weapon trader. So I check where all the guns are. It seems by default it's trader id 3.

So I'll add it this way:

```
{
  "Categories": [
    "Melee",
    "Sidearms",
    "Rifles",
    "Shotguns",
    "Submachine Guns",
    "Assault Rifles",
    "Sniper Rifles",
    "Grenades",
    "Morty's Weapons"
  ]
}
```

(For my sake and yours, please **never add a comma at the last element in the array**, in addition don't forget to use a json validator if something doesn't work)

All right, that's all for that file, let's keep up and see how the TraderPlusVehiclesConfig works.

7. TraderPlusVehiclesConfig

So after we opened the file, we should see array of vehicles along side their attachments. Keep in mind, that all attachment filled in that file will be spawn when you deploy a car with a receipt. As I'm not checking and saving parts when packing, players could actually try to glitch it buy selling parts and packing then deploying to get then again. What I advise is to disable the sell of all car parts. That way they won't glitch it.

So let's see how we can add a vehicle in that file.

```
{
  "VehicleName": "CivilianSedan",
  "Height": 0,
  "VehicleParts": [
    "SparkPlug",
    "CarBattery",
    "CarRadiator",
    "HeadlightH7",
    "HeadlightH7",
    "CivSedanDoors_Driver",
    "CivSedanDoors_CoDriver",
    "CivSedanDoors_BackLeft",
    "CivSedanDoors_BackRight",
    "CivSedanHood",
    "CivSedanTrunk",
    "CivSedanWheel",
    "CivSedanWheel",
    "CivSedanWheel",
    "CivSedanWheel"
  ]
},
{
  "VehicleName": "Dmns_MRAP",
  "Height": 2,
  "VehicleParts": [
    "SparkPlug",
    "CarBattery",
    "CarRadiator",
    "HeadlightH7",
    "HeadlightH7",
    "MRAPDriverDoor",
    "MRAPCoDriverDoor",
    "MRAPCargo1Door",
    "MRAPCargo2Door",
    "MRAP_wheel",
    "MRAP_wheel",
    "MRAP_wheel",
    "MRAP_wheel"
  ]
}
```


<pre>] }</pre>

Not really difficult, copy one and paste it at the end of the current array and change VehicleName to the classname of the vehicle and put the attachment needed for that car.

About the “**Height**”:

0 means height is default set by the hologram during the deployment.

]0;x] means heights is increase by the defined value to fix the deploy above ground.

Normally, cars should be deployed correctly above ground. For vanilla cars and most of them it’s working well.

However, since deploying a car like a pokemon seems a bit weird, some modders didn’t think of that and something is missing in the vehicle geometry making the hologram of the car appearing in the ground.

So only way I found to make them deploy above ground is to increase the height. So if you see that your modded car is being deployed in the ground, increase the height until it’s not.

Keep in mind that the actual Hologram will still appear in the ground, only the final result is fixed that way !

Last thing, if you don’t put all the cars you’re selling and buying in this files, that will make a full car appear instead of a receipt and fucked up the trader so only sell a car that is in that file !!!

All right, let’s take a look on how to set up a safezone for your traders.

8. TraderPlusSafeZoneConfig

First of all, I want to talk a bit about the safezone and its features:

That mod include safezone that disable most of the annoying action that can happen in the trader such as:

- Injection on target
- Desinfect on target
- Force consume on target
- Pack tent
- GiveBlood to target
- Collect blood from target
- Deploy Object
- Restrain yourself or target
- Force feed someone
- Lock doors
- Unpin grenades

In addition, zombies will be cleaned when they see a player.

You're also able to set item that will be cleaned over time and set admins that can bypass disable action while in safezone !

Finally, as server owner, you've got the possibility to place barrel hole or fireplace in the safezone. (don't place them in general config, place them by hand or with an admin tool directly in the safezone). Thoses objects can be ignite on even without wood. That way, it won't make the fire burn for nothing !

That file should be easy to get, You want to add a safezone here is how:

With default file (before adding a new safezone):

```
{
  "MsgEnterZone": "You Have Entered The Safe Zone!",
  "MsgExitZone": "You Have Left The Safe Zone!",
  "SafeAreaLocation": [
    {
      "SafeZoneStatut": "Trader",
      "X": 11595.0,
      "Y": 14688.0,
      "Radius": 200
    }
  ],
  "CleanUpTimer": 1200,
  "ObjectsToDelete": [
    "Paper"
  ],
  "SZSteamUIDs": [
    "76561198047475641"
  ]
}
```

After adding a new safezone:

```
{
  "MsgEnterZone": "You Have Entered The Safe Zone!",
  "MsgExitZone": "You Have Left The Safe Zone!",
  "SafeAreaLocation": [
    {
      "SafeZoneStatut": "Trader",
      "X": 11595.0,
      "Y": 14688.0,
      "Radius": 200
    },
    {
      "SafeZoneStatut": "Trader 2",
      "X": 1155.0,
      "Y": 1468.0,
      "Radius": 100
    }
  ],
  "CleanUpTimer": 1200,
  "ObjectsToDelete": [
    "Paper"
  ],
  "SZSteamUIDs": [
    "76561198047475641"
  ]
}
```

So let's take a look at the variables:

"MsgEnterZone": string var that correspond to the notification message when a player get in the safezone

"MsgExitZone": string var that correspond to the notification message when a player get out of the safezone

"CleanUpTimer": time in second where items selected gets deleted inside the safezone.

"ObjectsToDelete": array<string> that allow you to add items that will be cleaned over time in the safezone

"SZSteamUIDs": array<string> that allow you to add steamID64 of admins that aren't concern about all the blocking action in the safezone such as unpinning a grenade, placing object, etc...

```
"SZSteamUIDs": [  
    "76561198047475641",  
    "76561198047475857"  
]
```

We're getting to the end, nice. I'm getting tired of writing here. All right, let's move on to a little explanation on the Stock files.

9. TraderPlusStock_ID_X

If you're using that mod, it's probably for the stock feature, so it means you'll have the possibility to see what's in each trader stock id. In your DayZServer\profile\TraderPlus\TraderPlusDatabase, you should see a stock file for each id you set in TraderPlusIDsConfig.json. By default, they will be empty because my way of thinking is that it will be filled by player.

In addition to that, you may already see a couple of line in the stock regarding product with unlimited stock because when you enable the autocalculation in TraderPlusPriceConfig, it automatically add them in the stock. If they aren't added, that's not bad. The player will be able to see that he can sell that product and then after the first sell it will be added to the stock.

If you want to add a stock for a couple of items here is how:

```
{
  "TraderPlusItems": [
    "WolfSteakMeat 47 0",
    "Apple 40 0",
    "Plum 11 0",
    "WolfSteakMeat 45 3"
  ]
}
```

Keep in mind that the last value is the health of the product:

0: pristine

1: worn

2: damaged

3: badly damaged

Also, the stock value, the middle one doesn't always mean it's the quantity of the product. For example, here, the wolfsteakmeat is set at tradeqty = 0.75 in my price config, so each stock unit will give you or request that you have at least 75% of the quantity of the product.

Also, if you're messing by hand any of the stock file, keep in mind that the format is really important. You need to have EXACTLY one space between each variable, and you must make sure your json is valid. Again, use a json validator like <https://jsonlint.com/?code=> before coming to me saying you do not have anything in stock anymore.

All right, that should be all for that part. Finally let's talk briefly about the logs.

10. TraderPlusLogs

In order to get informations about how the mod is doing, you'll have some logs to help you see if config is loading correctly and also a trace of each players transaction.

Thoses logs can be found in *TraderPlus\TraderPlusLogs*

Keep in mind that you may find answer when something is wrong in there so don't forget to check that out !

For example, for no *fucking* reason, your trader doesn't work anymore. So of course, you think it's because dmitri's shit is buggy so you come to me for help saying my shit is broken. (I'm not saying it's not possible, I'm saying to make sure it's not you first 😊)

So before coming to the discord to me because it doesn't work, open your logs and check the couple of start lines:

```
[Creation Time: 2021_4_28-18-41-46
18-41-46 | [INFO] SafeZoneCore - Started !
18-41-46 | [INFO] TraderPlusCore - Started !
18-41-46 | [INFO] DEBUG MODE ENABLED
18-41-46 | [INFO] TraderPlus file exist ! loading...
18-41-46 | [INFO] TraderPlusJsonLoader::LoadFromFile Loaded file: $profile:TraderPlus\TraderPlusConfig\TraderPlusGeneralConfig.json
18-41-46 | [INFO] TraderPlus file exist ! loading...
18-41-46 | [INFO] TraderPlusJsonLoader::LoadFromFile Loaded file: $profile:TraderPlus\TraderPlusConfig\TraderPlusVehiclesConfig.json
18-41-46 | [INFO] TraderPlus file exist ! loading...
18-41-46 | [ERROR] TraderPlusJsonLoader::LoadFromFile ERROR Parsing %1JSON ERROR:
Missing a comma or ']' after an array element.
18-41-46 | [INFO] CheckForSpacesToRemove - Started
18-41-46 | [INFO] CheckForSpacesToRemove - Ended
18-41-46 | [INFO] TraderPlus file exist ! loading...
18-41-46 | [INFO] TraderPlusJsonLoader::LoadFromFile Loaded file: $profile:TraderPlus\TraderPlusConfig\TraderPlusIDsConfig.json ( Fi
18-41-46 | [INFO] TraderPlusServer - Started
18-41-46 | [INFO] TraderPlusBankingCore - Started !
18-41-46 | [INFO] DEBUG MODE ENABLED
18-41-46 | [INFO] BankingSettings file exist ! loading...
18-41-46 | [INFO] TraderPlusJsonLoader::LoadFromFile Loaded file: $profile:TraderPlus\TraderPlusConfig\TraderPlusBankingConfig.json
18-41-46 | [INFO] TraderPlusBankingServer - Started
18-41-46 | [INFO] CarLockCore - Started !
18-41-46 | [INFO] CarLockServer - Started
```

"All right Dmitri, but I'm no scripter, I don't know this stuff."

*Well, dear user, I want you to look for the keyword **[ERROR]**. As you can see in the screenshot, there is config file that is showing an error. You need to find that file. You can deduce it's the PriceConfig because it's the only one not showing in the logs or verify each traderplus json with <https://jsonlint.com/?code=>*

It will allow you to find the line where the mistake is in order to fix it 😊

VII. How to make static object works with TraderPlus

During your configuration attempt, you may have tried to had static object has trader in the GeneralConfig. And maybe, after starting your server and joining the game, you did not found thoses static objects. It probably means they need a quick fix to make them compatible with the mod.

In order to make thoses static objects compatible you need to make sure they **extends BuildingBase**

For example, I want to make ZT Vending Machine works with the mod:

I need create a file (StaticTrader.c) to add into my server mod in 4_World:

And add in there:

```
class ZT_Vending_Crafting_rusty extends BuildingBase{};
class ZT_Vending_Hunting_rusty extends BuildingBase{};
so on...
```

Here are the instruction:

First of all, I'm assuming you know how to pack and publish a mod, if you don't know how to do that, find someone who does so you can do last step.

Step 1: Download that mod template: [download link](#)

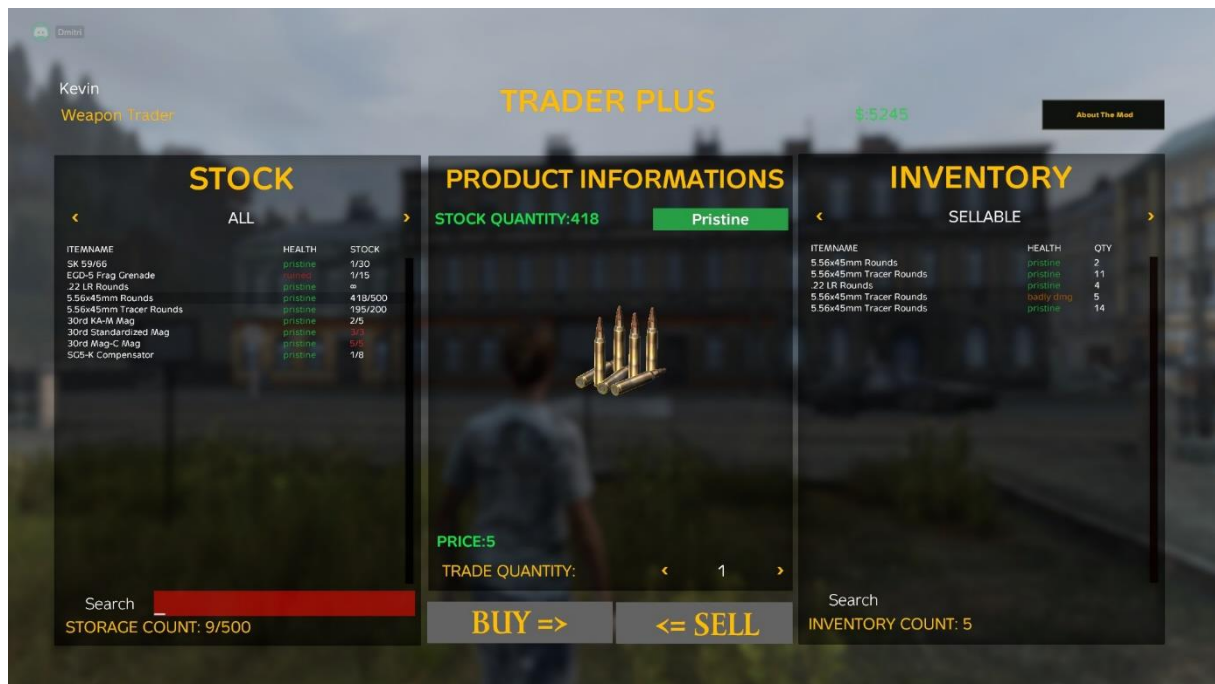
Step 2: Unzip it, go inside scripts\4_World\entities` and open the .c file called *Static Object*.

Step 3: Follow the commented instruction, again, Replace CLASSNAME with your classname of your static object you want to make compatible.

Step 4: Pack and publish the mod into your modpack or standalone, I don't really care at that point what you do. Just make sure to load that mod before TraderPlus so it will make your static object compatible with the mod.

Now you can't say that you don't know. 😊

VIII. Explanations regarding the UI Interface



The interface will give you three frames: - STOCK – PRODUCT – INVENTORY

LEFT FRAME:

You'll be able to see what the trader has in stock in the left frame. Items that a **red STOCK** in the list means the **stock limit is reached**.

Also, you'll be able to see multiple times the same item but in different health condition. (if you want to limit the trade for only pristine or worn check GeneralConfig). It also means each item with a different state can be filled with the defined stock filled in the priceconfig. So your stock can be multiply by 4 if you allow pristine, worn, damaged and badly damaged. Keep in mind that the price is reduce by 25% for each condition.

an item that is worn will be buy/sell = 75% of the pristine one.

an item that is damaged will be buy/sell = 50% of the pristine one.

an item that is badly damaged will be buy/sell = 25% of the pristine one.

At the bottom of the left frame, you can see the STORAGE COUNT: that's the number of line that you can see in stock. (max value is equal to MaxStockPerID in General Config)

MIDDLE FRAME:

You can see the product selected from either the STOCK LIST or the INVENTORY LIST. You'll see quantity of the item or stock quantity on the top, the health condition, the price based on the trade quantity value.

You've got the two buttons BUY and SELL, I don't think that needs more explanations.

RIGHT FRAME:

You're able to see by default all the items that you can sell to that trader. Keep in mind that to avoid some loss, there is a few features:

- Weapons or Tools in the back (shoulder) can't be seen in the list, so you don't accidentally sell your primary gun. Use your hands to sell a gun or put them in your backpack.
- Weapons attachments are given back to the player if he sell a gun that has attachments.
- Clothes, Bag that have items inside, can't be sold. That way you don't accidentally sell your bag or your jacket will all your money.
- Clothes weared by the player can't be seen. It needs to be in hand or in inventory to be sold.
- A third category called HANDS ATTACHMENTS allows you to see your weapon's attachment or the cargo inventory of your item in hand.

NEW FEATURES:

- You're able to quick buy/sell by double clicking in the list
- You can now sell multiples items at a time
- Bullets inside your mag are automatically removed when selling them => thanks to MaD for that

IX. TraderPlus items and purpose

If you take a look at the economy file that comes with the mod, you'll find some items that aren't present in the default dayz game.

here is the list of those items including those who are needed to accomplish some action:

-TraderPlus_Money_XXXXXX : That's all the money that can be used to trade with the mod. It includes coins, bills and others.

-CarLock: All is said but in case of, it's used to lock a car. Take it in your hand, then attach it and it will require to set a password from you.

-CarLockpick: All said too. It allows to attempt to detach the carlock on a car in order to steal it. Keep in mind that an alarm will be played in a 500m radius for 5 min after the breaking attempt.

-TraderPlusWrench: That's a reinforced wrench needed to pack a car. In order to pack a car, you must be the owner or an admin. Details are given in this doc somewhere.

-Plier: already in game, but in case you forgot, it's needed to detach a carlock from a car. Can only be done if you're the owner or an admin.

-TraderPlusCreditCard: That's the credit card needed (or not) to access the bank menu.

-Receipt: That's the proof of ownership that you get after packing or buying a car. It can be deployed inside a safezone or everywhere, depending on the settings.

X. Useful tools to have

To make sure your json syntax is correct: <https://jsonlint.com/?code=>

To add object or trader in the map:

-DayZ Editor: <https://steamcommunity.com/sharedfiles/filedetails/?id=2250764298>

(how to use it: <https://www.youtube.com/watch?v=GJRfJPqANdw>)

-Custom offline mod: <https://github.com/Arkensor/DayZCommunityOfflineMode>

(how to use it: <https://www.youtube.com/watch?v=27KElpExhKM>)

To set your TraderConfig or TraderPlusPriceConfig with an interface:

<https://trader-editor.iwapp.fr/> <https://traderplus-editor.iwapp.fr/>

XI. Conclusion

This is the end of the traderplus documentations, I hope you found an answer to your questions. It took me a full day to make that file so hope it was worth it. Enjoy the mod and have a good time in DayZ.

Paypal: <https://www.paypal.me/thedmitri>

Patreon: <https://www.patreon.com/thedmitri>