# Streamlabs Chatbot Documentation

Tuurau Dar	_
Тwitch Вот	
TWITCH STREAMER	
DISCORD BOT	
GAMEWISP	
STREAMLABS	
CLOUD	
OBS REMOTE	
IMPORTING DATA FROM ANOTHER BOT	17
FEATURES	19
Console	19
Dashboard	
COMMANDS	21
TIMERS	23
Quotes	24
Extra Quotes	25
Counter	26
GIVE AWAY	27
Sound Files	28
Currency	29
Betting	31
Poll	32
MINIGAMES - HEIST	33
MINIGAMES – DUEL	34
MINIGAMES – FREE FOR ALL	35
MINIGAMES – BOSS BATTLE	36
EVENTS	38
Songrequest	39
QUEUE	40
Notifications	41
Mod Tools	42
Discord	
Users	44
SETTINGS	45
GENERAL	45
Localization	
Macros	
HOTKEYS	
Style	
CHANGELOGS	
PERMISSION LEVELS	49
USAGE LEVELS	49
PARAMETERS	
Basic Parameters	50
Currency Parameters	54
TWITCH API PARAMETERS	
FILE READING PARAMETERS	
CUSTOM API READING PARAMETER	
Save File Parameters	
Miscellaneous Parameters	58

GENERATED TEXT FILES	58
STREAMLABS CHATBOT PYTHON SCRIPTING	61
Setup	61
FAO	62

# How to setup Streamlabs Chatbot?!

#### Twitch Bot

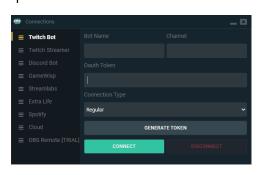
#### Step 1: Make sure you've made a Twitch.tv account for the bot

Go to Twitch.tv and create a new account for the bot to use.

#### Step 2: Go to Connections

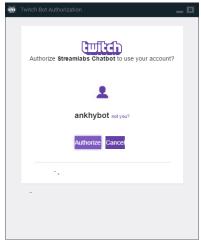
Simply navigate to the bottom left corner of the screen and click on open the Connections window and then click on "*Twitch Bot*".





#### Step 3: Generating a Token

Click on Generate Oauth-Token, this will open a the Authorization page on the bot.



#### Step 4: Click Authorize

Click "Authorize" and this will automatically fill in the token in to the token field.

#### Step 5: Picking your Connection Type

There are two options here "Regular or Secure". If you connect under Regular you will be connecting through Port 80. If you decide to connect under Secure you will be connecting through Port 443.

#### Step 6: Click Connect

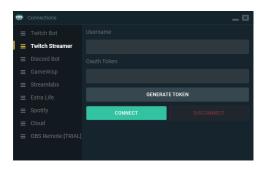
If you've done everything correctly your account will be connected to chat. If you get a pop up telling you the token does not belong to the Twitch name you typed into the username field then you were probably logged into the wrong account on Twitch.tv.

#### Twitch Streamer

#### Step 1: Go to Connections

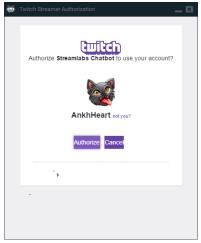
Simply navigate to the bottom left corner of the screen and click on open the Connections window and then click on "Twitch Streamer".





#### Step 2: Generating a Token

Click on Generate Oauth-Token, this will open a the Authorization page on the bot.



#### Step 3: Click Authorize

Click "Authorize" and this will automatically fill in the token in to the token field.

#### Step 4: Click Connect

If you've done everything correctly your account will be connected to chat. If you get a pop up telling you the token does not belong to the Twitch name you typed into the username field then you were probably logged into the wrong account on Twitch.tv.

#### Step 5: Access to Features

Follow, Host, Subscription (Chat) Notifications require your Streamlabs Account to be connected.

Cheer Notifications require you to have your Streamer Account connected.

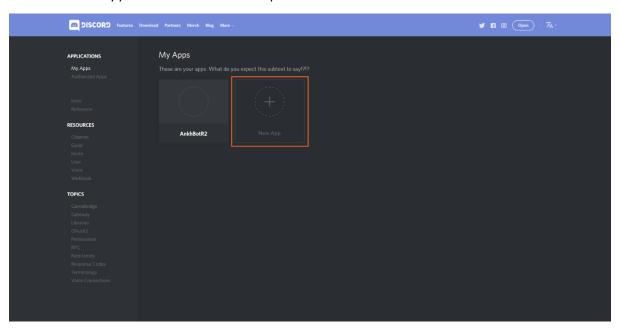
## **Discord Bot**

#### Step 1: Go to this website

Go to <a href="https://discordapp.com/developers/applications/me">https://discordapp.com/developers/applications/me</a>

#### Step 2: Click on New App

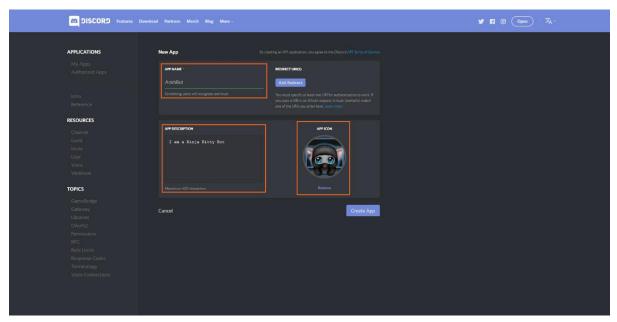
Click on "New App" to start the creation process.



#### Step 3: Create your Application

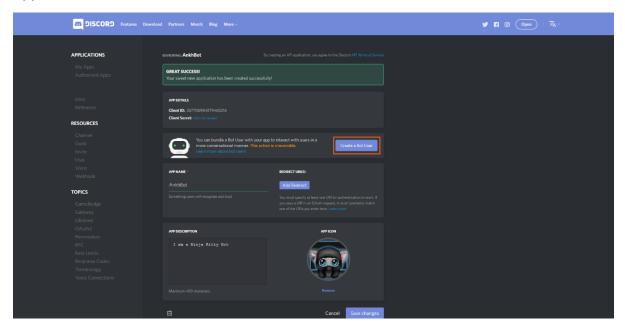
In order for the bot to connect to Discord you'll have to setup an application. This is what you will be doing on this page. Fill in the AppName this will be the name of your bot on Discord. I suggest using the same name on Twitch so it's easier for your viewers.

If you want the bot to have a nice little icon be sure to set it as well. The Description doesn't matter too much but you can just put anything you like in there.



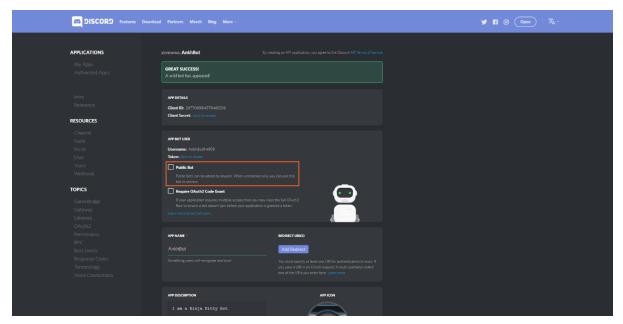
#### Step 4: Create a Bot User

After you have clicked "Create App" you will be redirected to the next page. Here you will have to Click on "Create a Bot User". This will create a Bot account for you using the AppName as the Bot's Username.



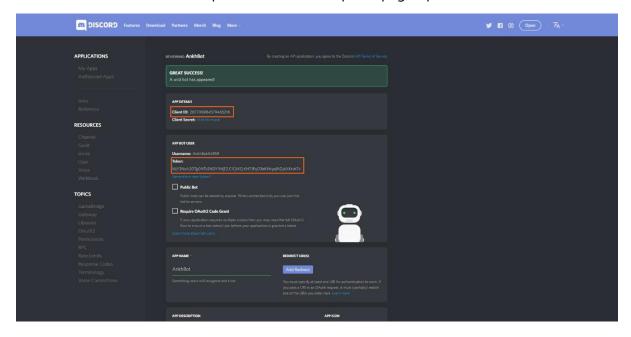
## **Step 5: Changing Settings**

After you've created the Bot User you will see a few more options have appeared above. Right under the App Bot User you will have to Uncheck Public bot since the bot will only work in the First channel it connects to. Streamlabs Chatbot does not support usage in multiple Discord channels.



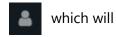
#### Step 5: Client ID & Token

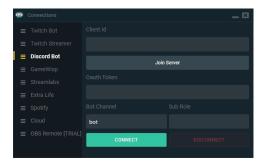
Click on "click to reveal" right next to the Token of the App Bot User. You will need the Client ID & Token in the next few steps so be sure to keep this page open.



#### Step 6: Go to Connections

Simply navigate to the bottom left corner of the screen and click on open the Connections window and then click on "*Discord Bot*".





#### Step 7: Enter Client ID & Token

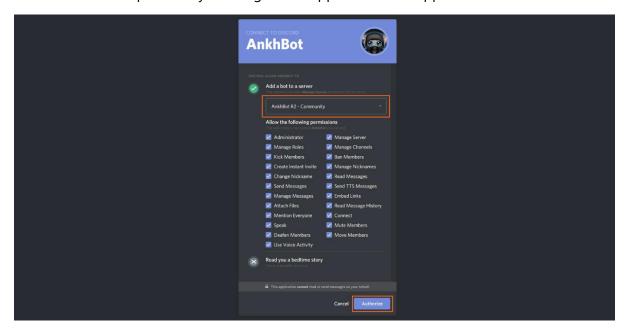
Enter your Client ID into the Client ID field, the Token into the Oauth-Token field. Next up enter the channel the bot will moderate in, by default this is "bot". This requires the text channel to actually exist on your discord server.

#### Step 8: Joining your Discord Server

You've done all the hard work now. The only thing that remains to be done is to have the bot Join your Discord Server. After filling out the fields as specified in the previous step click on "Join Server".

This will open a web page in which you will be able to pick which server the bot connects to. Pick the server and finally click on "Authorize".

On a side note the bot will only work in the first server it connects to. So be sure it's the right one otherwise you'll have to ask the server owner to kick the bot from their server or you have to restart the process by creating a new Application and App Bot User.



#### Step 9: Connecting the Bot to your Discord Server

Simply click on "Connect" and the bot should connect to Discord if you've done everything correctly.

#### Step 10: Linking Twitch & Discord Accounts to use Commands

The bot will not reply to users that are attempting to use commands if they've not linked their Twitch and Discord account. This is especially important seeing as the bot needs to know who the actual user of the command is on Twitch to Display their points, handle permissions, etc...

This process can be started by DMing the bot on Discord with !linkdiscord. The bot will tell you to message it something through Twitch Whispers. If that has been done correctly the accounts will be linked together and the user can use commands just as they would on twitch.

In case a user wants to unlink their accounts they can simply DM the bot !unlinkdiscord and it will be done.

## GameWisp

#### Step 1: Make sure you have a GameWisp account

If you're not partnered on Twitch but you want a way for your Viewers to Subscribe to you and support you then GameWisp is a good idea. Simply go to <a href="http://GameWisp.com">http://GameWisp.com</a> and create an account. If you already have one simply go to Step 2.

#### Step 2: Go to Connections

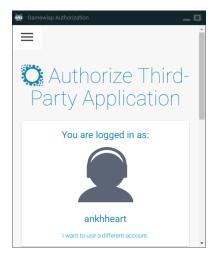
Simply navigate to the bottom left corner of the screen and click on open the Connections window and then click on "GameWisp".





#### Step 3: Generating a Token

Click on "Generate Token" this will open the Authorization page on the bot.



#### Step 4: Click Authorize

Click "Authorize" and this will automatically fill in the token in to the token field.

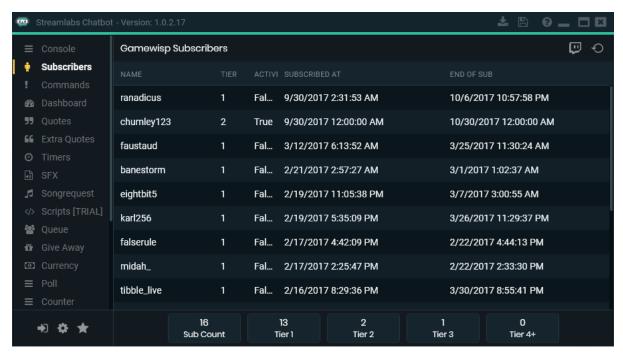
#### Step 5: Click Connect

In case you're also partnered on Twitch and want GameWisp subscribers to be able to join Twitch Subscriber giveaways and make use of Twitch Subscriber commands check the checkbox.

Finally click "Connect" and if everything went well then your GameWisp will be connected.

#### Step 6: Refreshing the Database

Next go to the Subscribers tab on the Main UI and click the Refresh button to grab all your current GameWisp Subscribers.

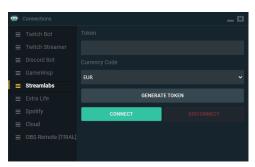


#### Streamlabs

#### Step 1: Go to Connections

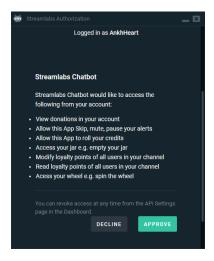
Simply navigate to the bottom left corner of the screen and click on open the Connections window and then click on "Streamlabs".





#### Step 2: Generating a Token

Click on "Generate Token" this will open the Authorization page in on the bot.



#### Step 3: Click Authorize

Click "Approve" and this will automatically fill in the token in to the token field.

#### Step 4: Click Connect

Finally click "Connect" and if everything went well then your Streamlabs will be connected. Now you can set how much someone gains for every \$/€/... someone donates under the currency system.

If you wish for the bot to post an in chat notification then go to Notifications and enable the Streamlabs Donate Notification.

#### Step 5: Enable Chatbot pages on your Streamlabs Tipping Page

Go to Settings -> General -> Scroll to Show Chatbot Tab on Donations Page and enable it. Afterwards pick which tabs you want to show. Next go to Connections -> Streamlabs and click Sync to sync your Commands & Quotes. The Playlist, Songlist and Queue auto sync every 2.5 minutes so no need to manually sync those.

#### Cloud

#### Step 1: Disclaimer

In order to use this you will either have to have Dropbox, Google Drive or another similar Cloud service's client installed on your system.

These services usually come with a dedicated server on your Computer from where data will automatically be synced to the cloud.

You'll have to set these up on your own seeing as there are more than enough tutorials on youtube.

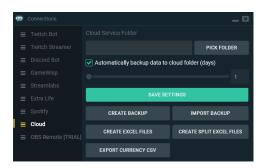
Dropbox: <a href="https://www.dropbox.com/install">https://www.dropbox.com/install</a>

Google Drive: <a href="https://www.google.com/drive/download/">https://www.google.com/drive/download/</a>

#### Step 2: Go to Connections

Simply navigate to the bottom left corner of the screen and click on open the Connections window and then click on "Cloud".





#### Step 3: Picking your Cloud Folder Path

Click on "Pick Folder" and Navigate to Cloud Service of Choice's Local folder and click "Save". This is where the bot will be able to output Automated Backups and Excel files which you can share with the stream.

In case you want the bot to create automated backups check the box and set the interval of the backups. Do mind though that the bot does not delete older backups so this is your responsibility. Once in a while be sure to delete some of the older ones so your Cloud data doesn't get capped out.

Click "Save Settings" to finish the process.

#### Step 4: Sharing a Link to Excel Files

Click "Create Excel Files" this will generate 3 excel files based on your data. The Data.xlxs will contain your Commands, Timers, SFX, Events, Points, Ranks, ... As for the Songlist.xlsx this will contain your Songlist. The Queue.xlsx will contain your Queue.

Now that those files exist navigate to your Cloud folder, right click on the file for which you want a link.

In case you're using Dropbox click on "Copy Dropbox Link" this will have a link copied to your clipboard.

In case you're using Google Drive click on "Google Drive" -> "Share" -> "Get Shareable Link" and copy the link.

Now you can either a short link using <a href="http://tinyurl.com">http://tinyurl.com</a> or use the full link in your commands.

The Data files only get created whenever you click "Create Read Only Excel Files".

#### **OBS** Remote

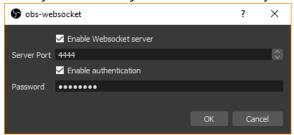
#### Step 1: Open your Browser

Next up make sure you have the OBS Remote Plugin installed if you do not then go here: <a href="https://obsproject.com/forum/resources/obs-websocket-remote-control-of-obs-studio-made-easy.466/">https://obsproject.com/forum/resources/obs-websocket-remote-control-of-obs-studio-made-easy.466/</a> and install the plugin.

#### Step 2: Open OBS

Go to Tools -> Websocket Server Settings inside of OBS and it will pop up a small window that allows you to configure the OBS Remote Plugin. Set which port you wish to use there and Enable Authentication. After doing so plug in a password that you wish to use so only authorized clients may connect to your OBS.

Once you are done just click OK to save your settings.



#### Step 3: Go to Connections

Simply navigate to the bottom left corner of the screen and click on open the Connections window and then click on "OBS Remote".





#### Step 4: Fill in the fields

If Streamlabs Chatbot is running on the same PC as your OBS then simply leave the IP to 127.0.0.1 if you are running a two PC setup and OBS is on another system the fill in the IP of your second PC which is running OBS.

If you changed the port in the OBS Websocket Server Settings then change it here as well. If you are using Authentication then fill in your password.

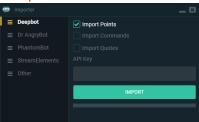
#### Step 5: Click Connect

Finally click connect and the bot will connect to your OBS allowing you to create commands and scripts which hide/show specific sources, Unmute your mic when you're being a dummy, Stop your stream when you pass out directly from chat so people can't watch you snooze away,...

## Importing Data from another Bot

Simply click on the ? in the top Right -> Open External Bot Importer

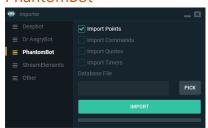
#### DeepBot



In order to import Data from Deepbot you need to have Deepbot Premium. If you have DeepBot Premium follow these simple steps in order:

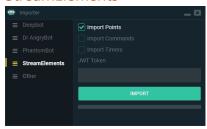
- 1. Open DeepBot
- 2. Navigate to your Master Settings
- 3. Look for your API Secret
- 4. If the secret is empty simply refresh it
- 5. Copy your API Secret
- 6. Click Save in the Master Settings
- 7. Then open Streamlabs Chatbot
- 8. Plug your API Secret into the API Key Field
- 9. Select what type of data you wish to Import
- 10. Click Import to start Importing data

#### **PhantomBot**



In order to Import Data from Phantombot you need to navigate to your phantombot.db file. Once you have done that select which data you want to import and afterwards start the import by clicking on Import Data.

#### **StreamElements**



In order to Import Data from StreamElements you have to retrieve your JWT token from your StreamElements account page. After you have done this place it in to the JWT Token field, select which data you want to import and start the process by clicking Import Data.

#### Other



In case you are importing data from a service we are not supporting make use of the Other tab. Here you will be able to import .txt files for both Quotes & Points. Once you have select the files that you want to import start the process by clicking Import Data

#### Points.txt Structure

ankhheart 8500 momo 3450 castorr91 500

#### Quotes.txt Structure

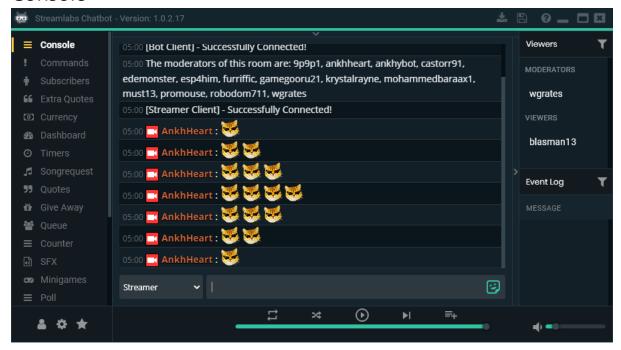
"I am a cat!" – AnkhHeart

"I am a cow!" – Momo

"This is me saying something stupid/funny!" – AnkhHeart

## **Features**

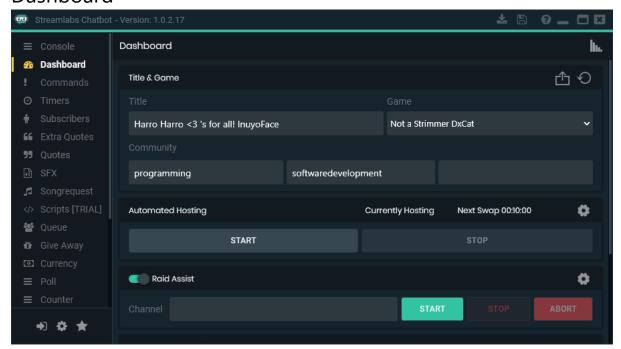
#### Console



On the console you will see all the incoming chat messages and the viewer list. In case you dislike seeing who's watching you can simply click the small button left of the viewer list to dock it to the side.

Aside from this at the top of the console you have access to Macro buttons which you can bind commands to. Further in the document this will be explained in more detail.

#### Dashboard



In order to access the full capability of the Dashboard you need to have your own Twitch account connected under Connections -> Twitch Streamer. This should be done if you followed the entire setup guide in the beginning.

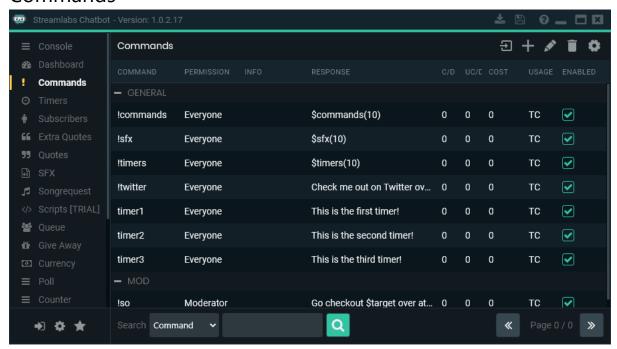
This is where you will be able to change your Title & Game, run commercials if you're partnered with Twitch, have the bot automatically host streams of your choosing and where the bot will track news Followers, Subscribers, Raiders, Hosts and GameWisp Subscribers.

We also have Raid Assist which is a system which allows you to reward viewers for joining you on a raid.

The Session Event View which is located at the bottom of the Dashboard will have to be manually cleared before each stream by right clicking and Clearing the data. Otherwise new Hosts, Raids, Subs/Resubs won't be logged if they're done by the same person.

!Status (messag	e)	[EDITOR]
Example	!Status [24h] Charity Stream!	
Response	{user} -> Title has been updated: [24h] Charity Stream!	
!Game (message		[EDITOR]
Example	!Game The Last of Us	
Response	{user}> Game has been updated: The Last of Us	
!StartHost		[EDITOR]
Example	!StartHost	
Response	{user}> Started Automated Hosting!	
!StopHost		[EDITOR]
Example	!StopHost	
Response	{user}> Stopped Automated Hosting!	

#### Commands



This is where you would start off if you want to create Commands. There are \$parameters that you can use in the commands to achieve various result. More information on these parameters can be found on page XYZ.

\$Parameters & Permission levels can be found further in to the documentation.

There is also support for Command Grouping any group starting with [GAME] will only work when you're actually playing the game which is defined behind the tag ex: [GAME] Pokemon Go

!Command Add	(command) (permlvl) (response)	[EDITOR]
Example	!Command Add !Cookie +r All your cookies belong to me!	
Response	{user}> Successfully added !Cookie. Permission: Regular - Mess your cookies belong to me!	age: All
!Command Edit	(command) (permlvl) and/or (response)	[EDITOR]
Example	!Command Edit !Cookie +a /me ate \$count cookies!	
Response	{user}> Successfully edited !Cookie. Permission: Everyone. Mess ate \$count cookies!	sage: /me
!Command Rem	ove (command)	[EDITOR]
Example	!Command Remove !Cookie	
Response	{user}> Successfully removed !Cookie.	
!Command Coun	nt (command) (num)	[EDITOR]
Example	!Command Count !cookie 10	
Response	{user}> Successfully set the count for !cookie to 10.	
!Command Usag	e (command) (usage ex: SC,SW,SB,DC,DW,DB,CB,WB,A)	[EDITOR]
Example	!Command Usage !cookie SC	
Response	{user}> Successfully set the usage of \$command to \$value.	

#### !Enable (command) (true/false)

[EDITOR]

Example	!Enable !cookie true	
Response	{user}> Succesfully enabled !Cookie	
!Command Cool	ldown (command) (minutes)	[EDITOR]
Example	!Command Cooldown !cookie 2	
Response	{user}> Successfully set the cooldown of !cookie to 2.	
!Command User	Cooldown (command) (minutes)	[EDITOR]
Example	!Command Cooldown !cookie 5	
Response	{user}> Successfully set the user cooldown of !cookie to 5.	

#### **Sharing Commands**

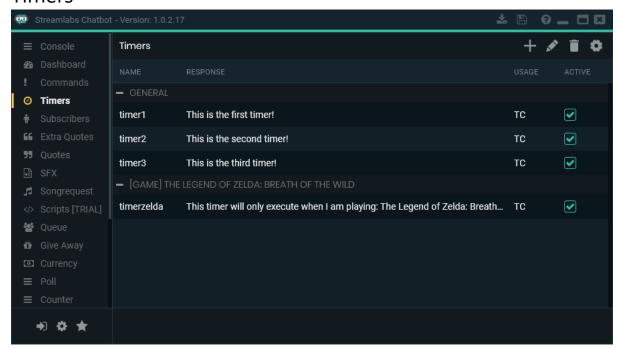
If you wish to share commands with your fellow streamer you can export them as .abcom (Streamlabs Chatbot Command) or .abcomg (Streamlabs Chatbot Command Group) by right clicking on a command. You have two options Export Command to export the single command or Export Group to Export all commands in that specific Group.

#### **Importing Commands**

Importing a script is simple. Simply click the Import Button in Avigate to the Zip File and Open it. Afterwards the bot will import the script for you and reload your scripts so it's ready to go.

#### **Timers**

23



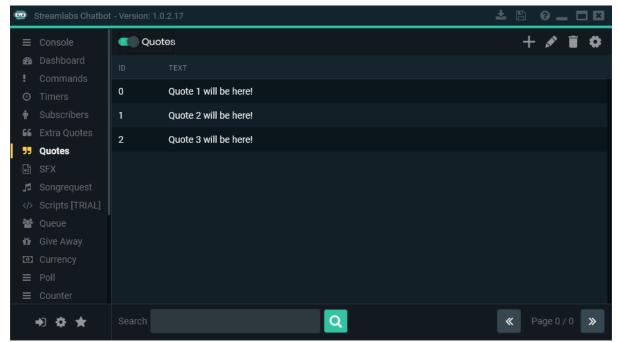
This is where you will create your own Timers. These are messages that the bot will automatically post into chat after an interval of X minutes. The interval is completely based on the Setting at the top.

All the timers will follow this same interval so this means the bot will post the first timer after the interval passes. Then it will start timing again, once the interval passes again it will post the second timer and so on eventually going through all of them and then starting back at the top.

There is also support for Timer Grouping any group starting with [GAME] will only trigger when you're actually playing the game which is defined behind the tag ex: [GAME] Pokemon Go

!Timer Add (nan	ne) (response)	[EDITOR]
Example	!Timer Add !Meow /me meows at \$randusername	
Response	{user}> Successfully added !meow. Permission: Everyone - Mess /me meows at \$ randusername	age:
!Timer Edit (nan	ne) (response)	[EDITOR]
Example	!Timer Edit !Meow /me growls at \$randusername	
Response	{user}> Successfully edited !Meow. Message: /me growls at	
	\$ randusername !timer remove (name) / Doesn't remove comman	d [Ed
!Timer Remove	(name)	[EDITOR]
Example	!Timer Remove !Meow	
Response	{user}> Successfully removed !Meow.	
!Activate (name	) (true/false)	[EDITOR]
Example	!Activate !Meow false	
Response	{user}> Succesfully de-activated !Cookie	

## Quotes

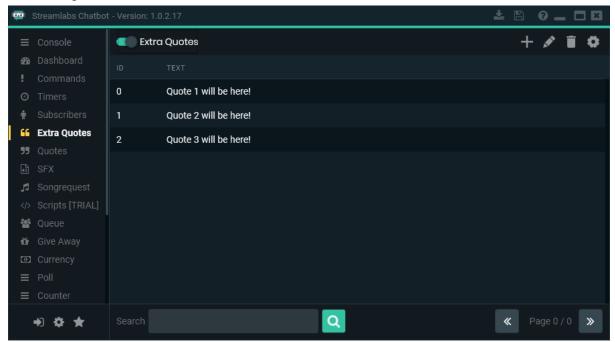


This is where things you've said on stream can be stored. You can change the permission on who can request a random quote and who can add them for you through chat.

You can also set the Cooldown and the Date Format. Every quote that gets added will automatically contain the Game & Date when the quote was created. So whenever someone calls upon the random quote they'll see when it happened and what you were playing at the time.

!Quote Add (tex	(t) [ADD PERMISSION]	
Example	!Quote Add "I am a cat!" - AnkhHeart	
Response	{user}> Succesfully added Quote #0: "I am a cat!" – AnkhHeart [Thief] [01/01/2015]	
!Quote Edit (id)	(text) [EDITOR]	
Example	!Quote Edit 0 "I am not a cat!" – AnkhHeart [Thief] [02/01/2015]	
Response	{user}> Successfully edited Quote #0: "I am not a cat!" – AnkhHeart	
	[Thief] [02/01/2015]	
!Quote Remove	(id) [EDITOR]	
Example	!Quote Remove 0	
Response	{user}> Successfully deleted Quote #0	
!Quote	[VIEW PERMISSION]	
Example	!Quote	
Response	Quote #2: "Duct tape solves all problems!" - AnkhHeart	
!Quote (id)	[VIEW PERMISSION]	
Example	!Quote 0	
Response	Quote #0: "I am not a cat!" - AnkhHeart	

## Extra Quotes

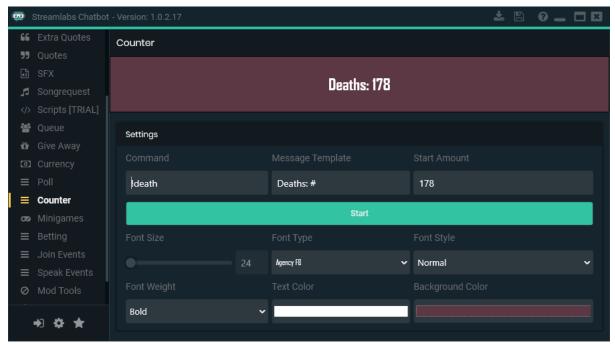


Using the Extra Quotes you can create your own version of the Quote System to store things that aren't specifically quotes. You can change the command, decide whether you want the Game & Date to show or not, change the Permissions and Response.

The underlying chat commands function the same way except if you do change the command you will also have to adjust the commands. By default this is !Gif if you change it to !Pun then you will have to use the commands starting with !Pun instead of !Gif.

!Gif Add (text)	[ADD PERMISSION]
Example	!Gif Add http://tinyurl.com/randomGif.gif
Response	{user}> Succesfully added Gif #0: http://tinyurl.com/randomGif.gif
!Gif Edit (id) (te	ext) [EDITOR]
Example	!Gif Edit 0 http://tinyurl.com/randomGif2.gif
Response	{user}> Successfully edited Gif #0: http://tinyurl.com/randomGif2.gif
!Gif Remove (id	[EDITOR]
Example	!Gif Remove 0
Response	{user}> Successfully deleted Gif #0
!Gif	[VIEW PERMISSION]
Example	!Gif
Response	Gif #2: http://randomURL.com/randomGif15.gif
!Gif (id)	[VIEW PERMISSION]
Example	!Gif 0
Response	Gif #0: http://randomURL.com/randomGif2.gif

#### Counter



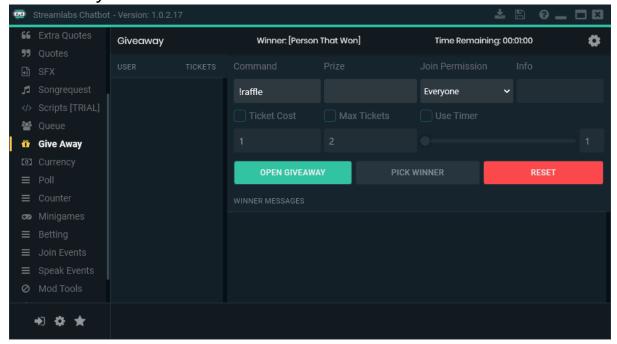
You can use the Counter to create a Death Counter, Hug Counter, Cookie Counter, etc.. It's used to count anything. You can change the settings to your liking just be sure to keep a # in the Msg Template since this will be replaced by the number.

In case you want to use the Counter but do not want to Capture the Display Area you can make use of a Death.txt file that is Located in the Bot's Install Directory -> Services -> Twitch -> Files Folder.

This file will be generated when you've added your first death. If you want to manually create this file then simply type !death 0 in chat. This will create the file with 0 Deaths inside. Do mind though if you changed the Command to something else you will have to use that instead.

!Death +		[MOD]
Example	!Death +	
Response	[Increased] Deaths: 124	
!Death -		[MOD]
Example	!Death -	
Response	[Decreased] Deaths: 123	
!Death (num)		[MOD]
Example	!Death 10	
Response	[Set] Deaths: 10	
!Death		[EVERYONE]
Example	!Death	
Response	Deaths: 10	

## Give Away

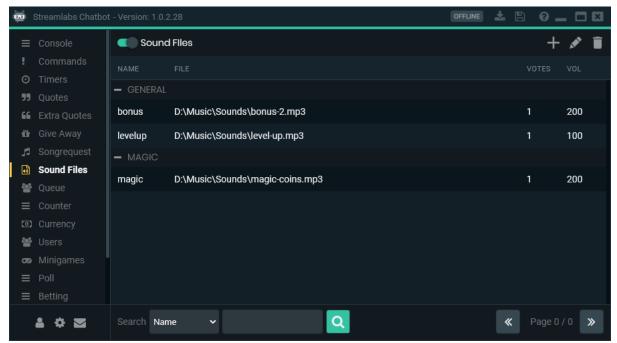


This is where you will be able to start Give Aways. You can either have people join the Give Away for free or have them pay a fee to enter or have them pay per ticket using in Channel Currency.

On the left side you will find all the people that are entered in the Give Away and how many tickets they possess. At the bottom of the window you will see all the messages posted by the Winner when one has been picked. That way you'll know if the user is active in case chat is moving really quickly.

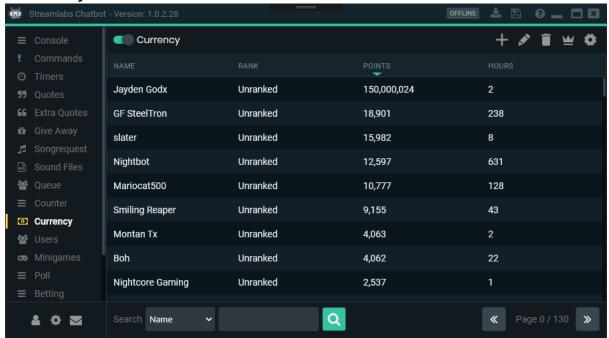
!Giveaway Start	Command Prize MaxEntriesPerUser EntryCost Permission	[EDITOR]
Example	!Giveaway Start !raffle cookies 1 25 regular	
Description	This starts a giveaway through chat with your own settings	
!GiveAway Start	: Command Prize Permission	[EDITOR]
Example	!Giveaway Start !raffle cookies everyone	
Description	This starts a very simple give away without tickets and entry costs	
!GiveAway Close	9	[EDITOR]
Example	!Giveaway Close	
Description	Prevents anyone from entering past this point	
!GiveAway Winr	ner	[EDITOR]
Example	!Giveaway Winner	
Description	Randomly picks the winner for the Give Away	

## Sound Files



The Sound Files tab allows you to add sounds to the bot which you can attach to notifications and commands. From within this tab you are able to control the Volume and Votes. The votes option only applies to commands as it determines how many times a command has to be used before the sound goes off.

## Currency



If the currency System is enabled everyone in your chat will start earning points based on your settings. These can be spent using the various other Systems in the bot such as Give Aways, SFX, Bet/Vote and enter Minigames.

The bot also supports Streamlabs currency. For this you need to connect Streamlabs and enable this functionality in your currency settings inside of the bot.

You can create up to four Ranking Trees: One for Viewers, Subscribers, Mods and GameWisp Subscribers. Ranks are only assigned whenever the bot pays out points or when you use !points add +viewers 1 for example.

There is also room for customizing your own Payout amounts and intervals. This way you have full control over how many points people can accumulate in your stream.

If the Offline Payout amount is set to 0 the bot will not pay out any points with the stream is offline. Also replace !points with your own custom currency command.

!Points Add (na	me) (amount)	[EDITOR]
Example	!Points Add AnkhHeart 10000	
Response	{user}> Successfully given AnkhHeart 10000 Points	
!Points Remove	(name) (amount)	[EDITOR]
Example	!Points Remove AnkhHeart 1234	
Response	{user}> Successfully removed 1234 Points from AnkhHeart	
!Points Add +Vi	iewers / +active (amount)	[EDITOR]
Example	!Points Add +viewers 100	
Response	{user}> Done giving 100 Points to everyone in chat	
!Points Remove	+Viewers / +active (amount)	[EDITOR]
Example	!Points Remove +viewers 100	
Response	{user}> Done removing 50 Points from everyone in chat	

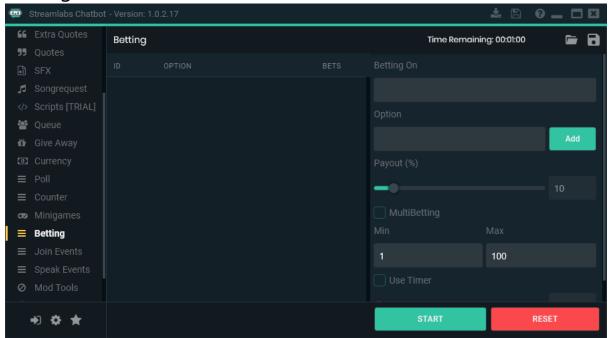
!Points [EVERYONE]

Example	!Points
Response	AnkhHeart [Ninja Kitty] - Hours: 13 - Points: 1337

!Transfer	[EDITOR]
-----------	----------

Example	!Transfer AnkhHeart MohammedBaraax1
Response	{user}> Successfully transferred currency from AnkhHeart to MohammedBaraax1

## **Betting**

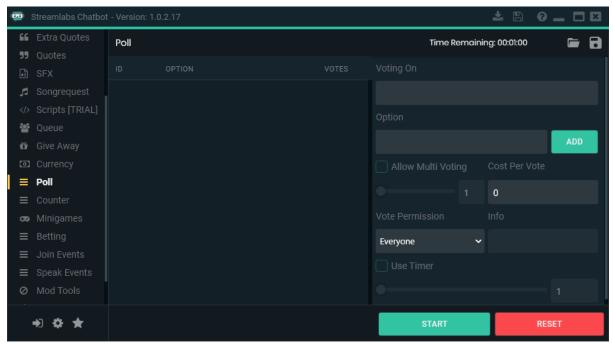


Using the Betting System you can open up the ability for Viewers to bet on the outcome of situations. These options can be saved into a present and loaded later in case you are playing the same game again.

If you wish to pick a winning option simply right click on the option and Pick it as the Winner. In case there are multiple correct Options this can be done for each of them.

!Bet (id) (amou	!Bet (id) (amount) [EVERYON	
Example	!Bet 0 1000	
Response	[None to prevent chat spam from the bot]	
!Betting Start B	ettingOn PayoutPercent Min Max MultiBetting Options [EDI	ΓOR]
Example	!Betting Start Will Ankh Survive? 35 1 100 true Yes No Maybe	
Description	This starts a custom betting session with custom settings	
!Betting Start B	ettingOn Options [EDI	ΓOR]
Example	!Betting Start Will Ankh Survive? Yes No Maybe	
Description	This starts a custom betting session that will use the settings that have	
	been set in the UI	
!Betting Stop		TOR]
Example	!Betting Stop	
Description	Prevents anyone from betting once used	
<b>!Betting Abort</b>	[EDIT	ΓOR]
Example	!Betting Abort	
Description	Cancels betting entirely and refunds anyone that has bet	
!Betting Winner (id) [EDITO		ΓOR]
Example	!Betting Winner 0	
Description	Picks the winning option and pay out points to everyone that bet on it	

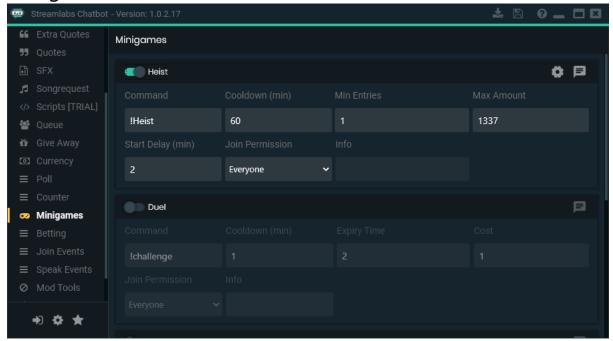
## Poll



The Poll System allows you to start a poll in your channel and have your viewers vote. In case you want people to spend points for each vote they cast then you can enable this by checking Allow Multi Voting and increase the limit.

!Vote (id)	Vote (id) [VOTE PERMISS	
Example	!Vote 1	
Response	[None to prevent chat spam from the bot]	
!Poll Start VotingOn Cost MaxVotes MultiVoting Options		
Example	!poll start What Game should I play Next? 10 1 false Witcher III Pokemon	
Description	This starts a custom poll with your own settings (overwrites UI settings)	
!Poll Start VotingOn Options [EDITOR		
Example	!poll start What Game should I play Next? Witcher III Pokemon	
Description	This starts a custom poll that will use the settings that have been set in the	
	UI	
!Poll Stop [EDITOR		
Example	!Poll Stop	
Description	Ends the poll and posts the result in chat	

## Minigames - Heist



The Group Minigame allows you to create your own Minigame. You can start the customization by determining the Command that will be used, what the cooldown is, how many users have to enter before it starts, the Max amount someone can invest and who can Join.

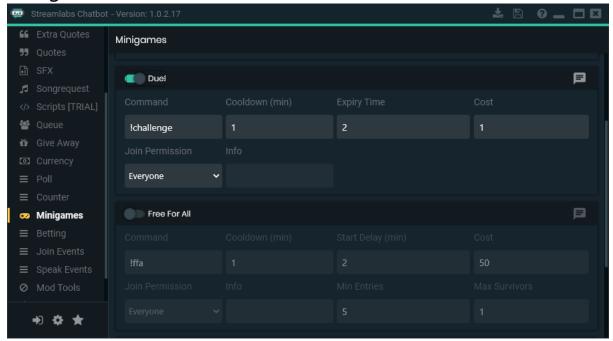
Aside from all those options you can set the Probability for each usergroup. This determines how much chance people within that usergroup have to survive. The Payout can also be set that way you can choose how much someone gets ontop of the amount they invested in the minigame.

Finally you can fully customize all the messages that the bot will be posting in chat depending on the situation and how well/bad things are going for the ones that have joined. So if you wanted you could turn it into something completely different and not use the default Heist preset.

!Heist (amount) [JOIN PERMISSION]

Example	!Heist 123
Response(1x)	{user} is trying to get a team together in order to hit the nearest bank
	Everyone can Join!- In order to join type !Heist (amount).

## Minigames - Duel



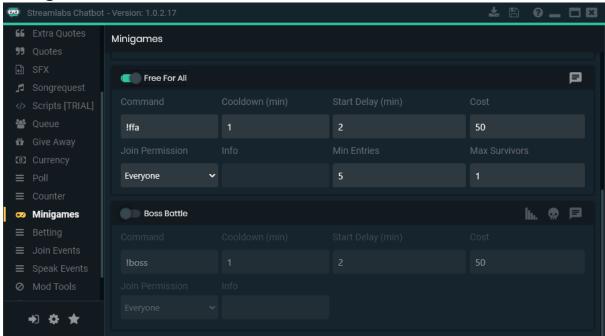
The Duel minigame allows viewers to challenge each other to a battle. The bot will process a secretive battle in the background, the winner will receive twice the cost. The loser will get nothing.

Aside from this both the challenger and challenged will go on cooldown once their fight concludes and can no longer challenge or be challenged till their cooldown expires.

!challenge (name) [JOIN PERMISSION]

· · · · · · · · · · · · · · · · · · ·	
Example	!challenge ankhheart
Response(1x)	{user} has challenged {target} to a fight! Type !challenge {user} to accept the challenge!

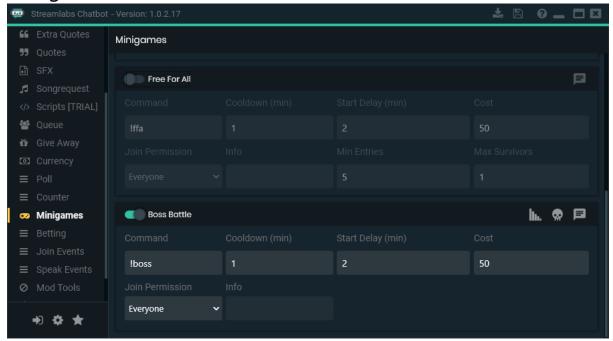
## Minigames – Free for All



In the Free for All minigame multiple viewers can face off against one another. You can determine how many people end up surviving. The more people join the larger the prize pool becomes and the winner walks away with the pot. In more than one person can survive then it gets split amongst the survivors.

!ffa	[JOIN PERMISSION]
Example	!ffa
Response(1x)	The arena is now open! Type !ffa to join!

## Minigames – Boss Battle



This allows you to create custom bosses for your viewers to fight based on how many people join. The difficulty / loot is completely up to you do mind that balancing it fairly is also your responsibility.

#### The Basics:

Balancing of the minigame is completely up to you so let's go down some of the basic concepts so you know how it functions in the background. That way you can determine what values would be best.

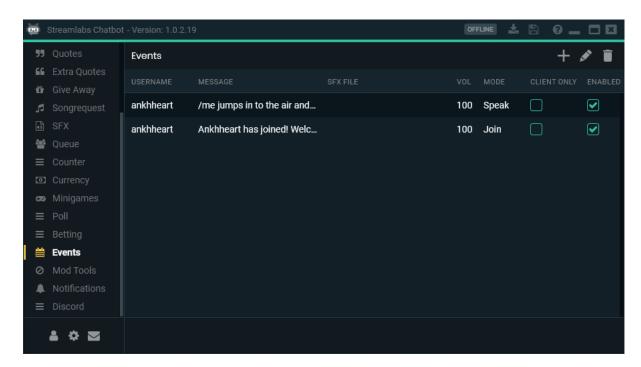
- 1) Players sign up for the battle and get a Stat sheet assigned based on their permission
- 2) Boss gets picked based on the group size (Between Min Max Entries)
- 3) The fight starts against the boss
- 4) Damage Calculation: (User Attack Target Defense) ex: 10 att 5 def = 5 dmg that the target will receive
- 5) Attack order: The boss has to be attacked 3 times before it counters the last attacker. So let's say we have a group with Ankh, Momo and Gooru and Ankh attacks first and then Momo and then Gooru. After Gooru finishes his attack he would get countered by the boss and be the only person to receive damage. Now prior to every attack phase taking place the order of people attacking will be shuffled so it's not always the same person getting countered.

- 6) Make sure to keep the Boss's Defense lower than Player's Attack at all times so they at least have a chance to beat him
- 7) Balance the health based on the Min Max Entries for this you will have to do a bit of math yourself based on the prior information given such as: Dmg Calculation and Attack Order
- 8) The Max Defense a player/boss can have is half their attack if this is higher than it will be capped out during calculation at 50%
- 9) Loot will get distributed evenly amongst all of the survivors at the end. In case no one survives then there is no loot to be distributed

!boss [JOIN PERMISSION]

Example	!boss
Response(1x)	{user} is trying to get a group of adventurers together to fight a boss!
	Type !boss to join him!

## **Events**



The Event System will allow the bot to automatically Greet/Shoutout the person of your choice and play a SFX if you wish. The system consists of two modes Join events and Speak events.

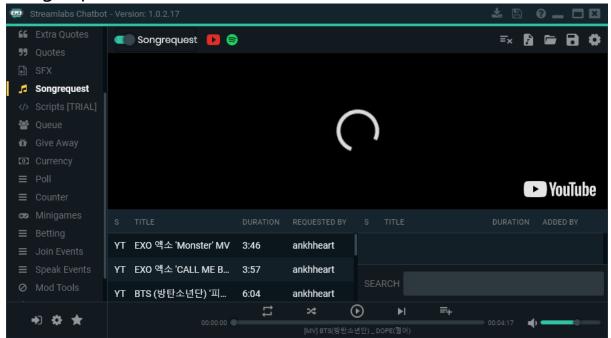
Join Events will perform its action when the person of your choice joins the channel. Then it will post its message and/or play its SFX.

Speak Events will perform its action when the person of your choice speaks in your channel for the first time. Then it will post its message and/or play its SFX.

In order for the bot to re-execute the events it has to be restarted. So the best thing is to restart it before a cast.

## Songrequest

!Songrequest (url/token)



The Song Request System allow you to create your own youtube playlist through the bot have them play whenever you want. Aside from that your viewers can request songs and spend currency to do so.

!Songrequest TY9cSlOhqTk

=/:0://-	1001.g. cquest 1.000.01.q. k
Response	{user}> The song Amv - [MEP] So Long Sentiment 720p has been added to the queue
!Skip	[SKIP PERMISSION]
Example	!Skip
Response	{user}> Your vote to skip has been successfully registered!
!Veto	[VETO PERMISSION]
Example	!Veto
Response	Amv - [MEP] So Long Sentiment 720p has been successfully skipped!
!Songblacklist	add (id) [EDITOR]
Example	!songblacklist add dQw4w9WgXcQ
Response	{user}> dQw4w9WgXcQ has been successfully Blacklisted!
!Songblacklist	remove (id) [EDITOR]
Example	!songblacklist remove dQw4w9WgXcQ
Response	{user}> dQw4w9WgXcQ has been successfully Un-Blacklisted!
!WrongSong	[EVERYONE]
Example	!WrongSong
Response	{user}, Successfully removed the last song you requested.
!Songlist	

https://streamlabs.com/<your name>#/chatbot/songlist

[REQUEST PERMISSION]

Example

Note

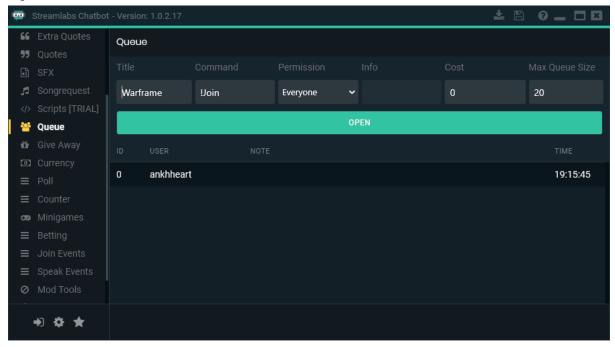
!Songlist

Create this yourself and point it to

!Volume (number) [EDITOR]

Example	!Volume 50
Response	{user}, Volume set to 50

## Queue

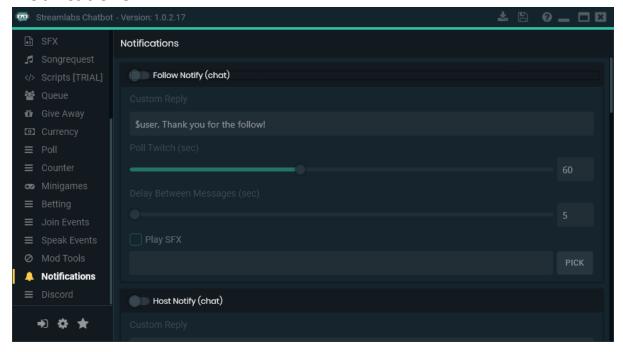


You can setup a Game Queue using this which allows your viewers to sign up to join you in a multiplayer game. You can have them spend currency to enter and you can even set it to Sub only in case you only want Subscribers to be able to sign up.

!Join <note></note>	[EVE	RYONE]
Example	!Join AnkhHeart#4798	
Response	[None unless enabled under Settings -> Localization]	
!Queuelist	[EVE	RYONE]
Example	!QueueList	
Note	Create this yourself and point it to	
	https://streamlabs.com/ <your_name>#/chatbot/queue</your_name>	
!Queue Open <g< td=""><td>game&gt; [E</td><td>EDITOR]</td></g<>	game> [E	EDITOR]
Example	!Queue Open Warframe	
Response	A queue has opened up for: Warframe - Cost: 0 points - Type !join	
	(optional:Note) to join!	
!Queue Close	[8]	EDITOR]
Example	!Queue Close	
Response	The queue has been closed! You can no longer enter!	
!Queue Clear	[1	EDITOR]
Example	!Queue Clear	
Response	The Queue has been cleared!	
!Queue Pick <nu< td=""><td>umber&gt; [E</td><td>EDITOR]</td></nu<>	umber> [E	EDITOR]
Example	!Queue Pick 3	
Response	Next up: AnkhHeart, Castorr91, Must13	

# !Queue Random < number > [EDITOR] Example !Queue Random 3 Response Next up: FurRiffic, WellBrained, Ocgineer !Leave [EDITOR] Example !Leave Response AnkhHeart has left the queue.

## **Notifications**



You will find various in Chat Notifications here ranging from Follower, Host, Subscriber Notifications to GameWisp Notifications. You can customize each of these to your liking.

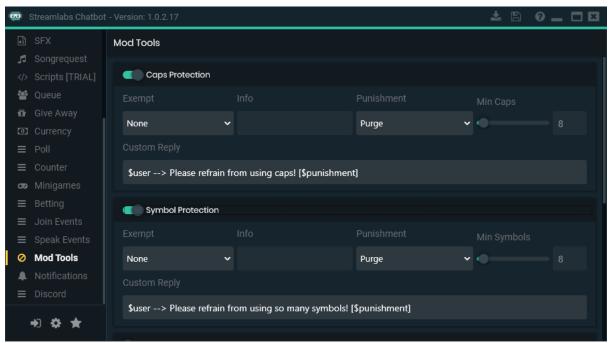
The Follow, Subscriber, Host & Streamlabs notifications require you to have your Streamlabs account connected.

The Cheer Notifications require you to have your Streamer Account connected.

The Extra Life Notifications require you to connect your Extra Life Parcipant ID.

The GameWisp Subscriber Notification is only available if you have connected your Gamewisp Account.

## **Mod Tools**

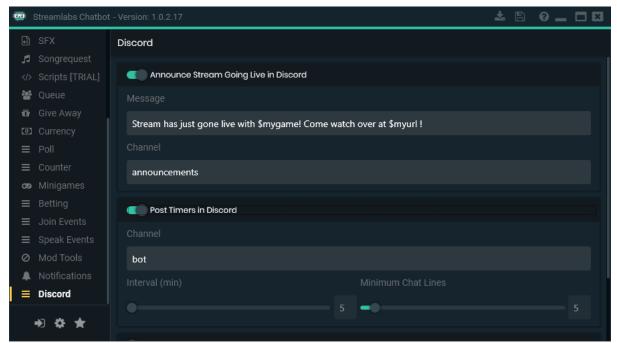


Using the Mod Tools you can have the bot punish viewers that post Links without permission, Spam Caps/Symbols or very offensive words/sentences.

Each of these can be fully customized. When it comes to Link Protection you can exempt certain websites from being punished.

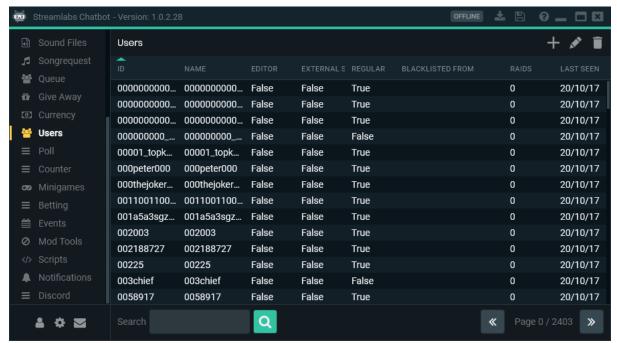
For the Word/Sentence Blacklist you can also make use wildcards such as \* or ?. More information about Wildcards can be found on the internet ex: <a href="https://en.wikipedia.org/wiki/Wildcard">https://en.wikipedia.org/wiki/Wildcard</a> character

## Discord



The Discord tab will allow you to activate specific functionality to work in Discord in regards to Timers, automatically assigning a role to everyone and even announcing when you go live.

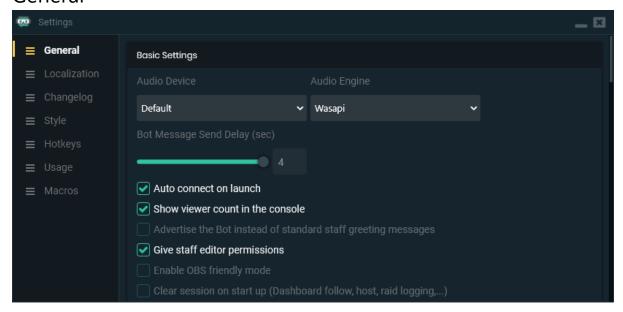
#### **Users**



In the Users tab you can see every user's id, name, whether they're an editor, external sub or regular. What they've been blacklisted from, how many times they've raided you and when they were last in your channel. From here you are also able to assign the Editor status to users whom you trust. They are then able to use Editor based chat commands to add, edit, remove commands, currency, start giveaways, etc...

# Settings

#### General



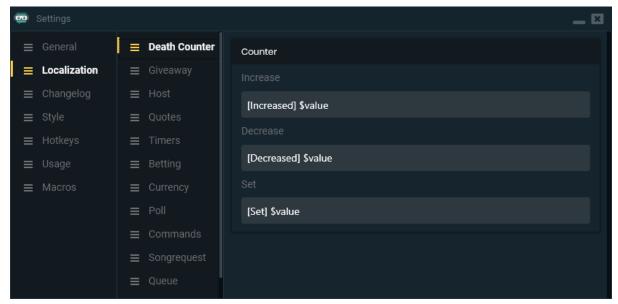
In the General Settings you will find some basic functionality such as the delay between bot messages, Auto connecting all your accounts when you launch it, etc...

Aside from this you can also setup a !raider command that can be used by Mods to save out a list of users that have raided your channel. These users will also get added to the Data.xlsx file in your Cloud folder if you have set that up.

#### !Reg Add/Remove (Name) [EDITOR] !Reg Add AnkhHeart / !Reg Remove AnkhHeart Example Response AnkhHeart has become a Regular! / AnkhHeart is no longer a Regular. !Sub Add/Remove (Name) [EDITOR] Example !Sub Add AnkhHeart / !Sub Remove AnkhHeart Response AnkhHeart has become a Subscriber! / AnkhHeart is no longer a Subscriber. !Blacklist Add/Remove (Name) [EDITOR] !Blacklist Add AnkhHeart / !Blacklist Remove AnkhHeart Example AnkhHeart has been Blacklisted! / AnkhHeart has been removed from the Response Blacklist.

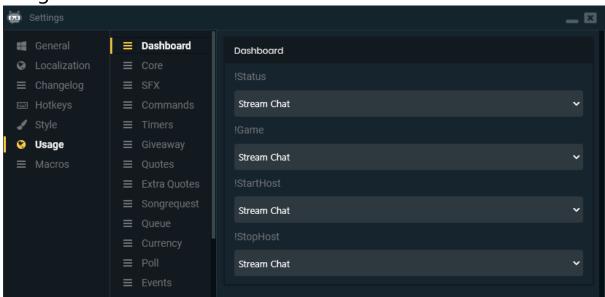
©Streamlabs.com

## Localization



Within the Localization you have the ability to change any of the default responses though try to keep most of the \$parameters unless you really don't want them to be there.

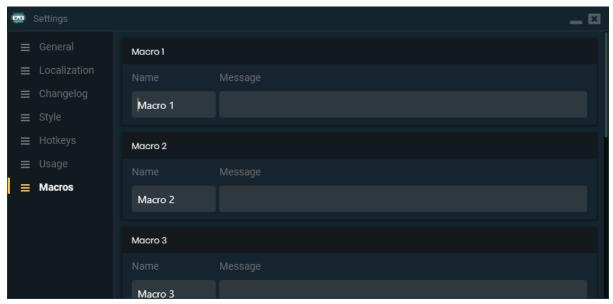
## Usage



Within the Usage you can change where and how commands can be used. Whether it be Chat, Whispers or Discord or everywhere.

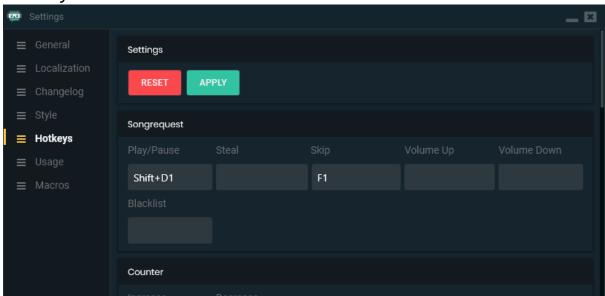
In the Core you can determine if users have to be in your Channel to use Whisper commands with the bot or not.

#### Macros



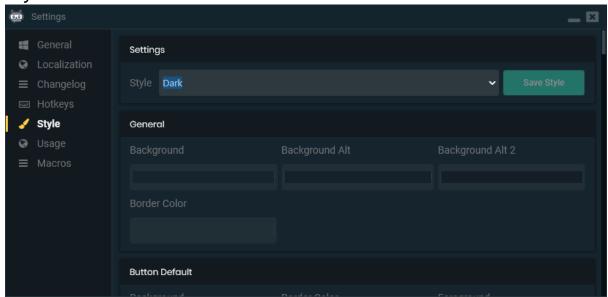
This is where you are able to setup Macros for the 7 buttons in the Console UI. It's a simple way to setup a few buttons to automatically post commands / messages in chat for you when you click them.

## Hotkeys



Using the Hotkeys you can set up Global Hotkeys for various actions within the bot such as Pausing a song, adding it to your playlist, skipping, increasing the volume,... Aside from this you're also able to setup Hotkeys for the previously Mentioned Macros so you don't have to click the buttons yourself. You can just hit your hotkey and it will execute them for you. Also don't forget to click Apply at the bottom to save your Hotkeys

## Style



In case you're unhappy with the colours of Streamlabs Chatbot's UI then this is where you would go. You can change every single colour here though you have to set the Style to Custom. Once you're done with your changes and you like the result simply click on Save Custom Style and it will be saved.

## ChangeLogs

This is where you will be to read up on the Changelogs in case you didn't read them prior to updating to a newer version.

# **Permission Levels**

+ a			
Description	Everyone		
+ r			
Description	Regular		
+ s			
Description	Subscriber		
+gw			
Description	GameWisp Subscriber		
+ m			
Description	Moderator		
+ e			
Description	Editor		
+i			
Description	Invisible		
+u(name) ex: +	u(ankhheart)		
Description	User_Specific: AnkhHeart		
+r(MinRank) ex	: +r(Lion)		
Description	Min_Rank: Lion		
+p(MinPoints) ex: +p(1000)			
Description	Min_Points: 1000		
+h(MinHours) ex: +h(13)			
Description	Min_Hours: 13		

# Usage Levels

sc	
Description	Stream Chat
SW	
Description	Stream Whisper
SB	
Description	Stream Both
DC	
Description	Discord Chat
DW	
Description	Discord Whisper
DB	
Description	Discord Both
СВ	
Description	Chat Both
WB	
Description	Whisper Both
Α	
Description	All

# **Parameters**

## **Basic Parameters**

\$desc(descript)	ion)
------------------	------

	· ,						
Description	Special parameter that can be placed on the first line of a command to sync a custom description to the web						
Example	\$desc(This command does an API call somewhere!) \$readapi(https://randomapi.com/thing)						
\$userid							
Description	Displays the user's id, in case of Twitch it's the user's name in lower case characters. Make sure to use \$userid when using \$addpoints, \$removepoints, \$givepoints parameters.						
Example	/me steals a cookie from \$userid   !test						
Response	Bot steals a cookie from ankhheart						
\$username							
Description	Displays the user's display name. Make use of this parameter when you just want to output a good looking version of their name to chat.						
Example	/me steals a cookie from \$username   !test						
Response	Bot steals a cookie from AnkhHeart						
, \$targetid							
Description	Displays the target's id, in case of Twitch it's the target's name in lower case characters. Make sure to use \$targetid when using \$addpoints, \$removepoints, \$givepoints parameters.						
Example	/me kicks \$targetid in the face!   !kick AnkhHeart						
Response	Bot kicks ankhheart in the face!						
\$targetname							
Description	Displays the target's display name. Make use of this parameter when you just want to output a good looking version of their name to chat.						
Example	/me kicks \$targetid in the face!   !kick ankhheart						
Response	Bot kicks AnkhHeart in the face!						
\$touserid							
Description	Displays the target's or user's id, in case of Twitch it's the target's or user's name in lower case characters. Make sure to use \$touserid when using \$addpoints, \$removepoints, \$givepoints parameters.						
Example	/me kicks \$touserid in the face!   !kick Chair or !kick						
Response	Bot kicks chair in the face! or Bot kicks ankhheart in the face						
\$tousername							
Description	Displays the target's or user's display name. Make use of this parameter when you just want to output a good looking version of their name to chat.						
Example	/me kicks \$tousername in the face!   !kick thunderceptor11 or !kick						
Response	Bot kicks Thunderceptor11 in the face! or Bot kicks AnkhHeart in the face!						
\$randuserid	·						
Description	Displays a random user that has spoken in chat recently. In case of Twitch it's the random user's name in lower case characters. Make sure to use						

©Streamlabs.com

	\$randuserid when using \$addpoints, \$removepoints, \$givepoints						
	parameters.						
Example	/me gives \$randuserid a hug!   !hug						
Response	Bot gives ankhheart a hug!						
\$randusernam	e e						
Description	Displays a random user that has spoken in chat recently. Make use of this parameter when you just want to output a good looking version of their name to chat.						
Example	/me gives \$randusername a hug!   !hug						
Response	Bot gives AnkhHeart a hug!						
\$botname							
Description	Displays the bot's name						
Example	Hello I am \$botname   !name						
Response	Bot: Hello I am StreamlabsBot!						
\$msg							
Description	Displays the text after the command						
Example	\$username rolls a \$randnum(1,21) for \$msg   !msg   wish   had 9 lives!						
Response	Bot: AnkhHeart rolls a 18 for I wish I had 9 lives!						
\$mychannel							
Description	This will be replaced by the channelname where the bot is connected						
Example	Connected to \$mychannel  !mychannel						
Response	Bot: Connected to AnkhHeart						
\$dummyormsg							
Description	This will get replaced by anything behind the command. If there is nothing it be cleared from the response message.						
Example	http://api.com/\$dummyormsg   !test Cats or !test						
Response	Bot: http://api.com/Cats or http://api.com						
\$dummy							
Description	This is a Dummy that will not post the message if there is nothing behind the command						
Example	\$dummy \$readrandline(C:\Users\Ankh\Blah.txt)   !8ball Am I green?						
Response	Bot: Perhaps?!						
\$arg1 to \$arg9	9						
Description	\$arg1 will give you the first word after the command and \$arg9 the ninth. If these parameters are in the command it expects them to be there if they are not entered the command will not post.						
Example	/me hugs says \$arg1 \$arg2!   !argtest Hi Meow? Cookies?						
Response	Bot Hi Meow?!						
\$argl1 to \$arg	19						
Description	\$argl1 will give you the first word after the command and \$argl9 the ninth but all in lower case. If these parameters are in the command it expects them to be there if they are not entered the command will not post.						
Example	/me hugs says \$arg1 \$arg2!   !argtest Hi Meow? Cookies?						
Pasnansa	Pot hi moow?						

©Streamlabs.com

Bot hi meow?!

Expects a valid integer

Response | | \$num1 to \$num9

Description

Example	/me hugs \$targetname \$num2 times!   !hug ankhheart 10					
Response	Bot hugs AnkhHeart 10 times!					
\$randnum(max)	or \$randnum(min,max)					
Description	Displays a random number in a specified range					
Example	/me rolls a \$randnum(1,7)!   !roll					
Response	Bot rolls a 3!					
\$randquote						
Description	Displays a random quote					
Example	\$randquote   !randquote					
Response	Bot: I am not a cat! – AnkhHeart [Thief] [01/01/2015] \$randextra					
\$randextra						
Description	Displays a random value from the extra quotes					
Example	\$randextra   !randgif					
Response	Bot: http://randomURL.com/gif12.gif					
\$quotes						
Description	Displays the amount of quotes					
Example	There are \$quotes quotes.   !quotes					
Response	Bot: There are 123 quotes.					
\$maxquotes						
Description	Displays the highest number quote					
Example	There are \$quotes quotes. Ranging from 0 to \$maxquotes.   !quotes					
Response	Bot: There are 123 quotes. Ranging from 0 to 122.					
\$count						
Description	Counts amount of times a command has been used					
Example	/me has \$count jars of salt.   !count					
Response	Bot has 3 jars of salt.   Bot has 4 jars of salt.   Bot has 5 jars of salt   etc					
\$checkcount(co						
Description	Displays the count of a specific command					
Example	Cookie Count: \$checkcount(!cookie)   !check					
Response	Bot: Cookie Count: 10					
	mCommandsPerPage)					
Description	Displays a list of all available commands for the user					
Example	Commands: \$commands(3)   !commands or !commands (PageNumber)					
Response	Bot: Commands: !Cookie, !Slap, !Caster [Page 0/2]					
\$queuepos(targ	etid)					
Description	This will display the target's position in the queue					
Example	\$username you are in Position \$queuepos(\$userid)   !MyPos					
Response	AnkhHeart you are in Position 1					
\$queue(amount						
Description	This will display the first X amount of people in the queue					
Example	Next Up in Queue: \$queue(3)   !NextUp					
Response	Next Up in Queue: #0 mohammedbaraax1, #1 ankhheart, #2					
	gamegooru21					
\$timers(NumTir						

52 ©Streamlabs.com

Timers: \$timers(3) | !timers or !timers (PageNumber)

Displays a list of all available Timers

Description

Example

Response	Bot: Timers: !ctt, !twitter, !youtube [Page 0/1]						
\$date							
Description	Displays the Date based on the format under Quote Settings						
Example	Currently it is \$date						
Response	Bot: Currently it is 08/09/2015						
\$sfx(NumSFXPe							
Description	Displays a list of all available SFX for the user						
Example	SFX: \$sfx(3)   !sfx or !sfx (PageNumber)						
Response	Bot: SFX: !scream, !pika, !morph [Page 0/0]						
\$time							
Description	Displays the caster's time						
Example	Currently it is \$time over at AnkhHeart's part of the world.						
Response	Bot: Currently it is 10:20 PM over at AnkhHeart's part of the world.						
\$currencyname							
Description	Displays currencyname						
Example	In this channel you can collect \$currencyname   !currency						
Response	Bot: In this channel you can collect Cookies!						
\$currentsong a	nd \$requestedby						
Description	Return the current song that is being played through songrequest						
Example	Current Song: \$currentsong – Requested By \$requestedby !currentsong						
Response	Bot: Current song: ONE MORE FIGHT – Requested By AnkhHeart						
\$nextsong and	\$nextrequestedby						
Description	Return the current song that is next in queue						
Example	Next Song: \$nextsong – Requested By \$nextrequestedby !nextsong						
Response	Bot: Next song: ONE MORE FIGHT – Requested By AnkhHeart						
\$countdown(12	2:00 AM) or \$countdown(04/05/2015 12:00 AM)						
Description	Allows you to start a countdown from the current time to the set						
	time/date						
Example	\$countdown(04/05/2015 12:00 AM)   !sleep						
Response	Bot: 1 day 2 hours 48 minutes 36 seconds						
\$countup(12:00	O AM) or \$countup(04/05/2015 12:00 AM)						
Description	Allows you to set a start date for when the bot should start counting						
Example	\$countup(07/03/2016 12:00 AM)   !UsingStreamlabs Chatbot						
Response	Bot: 1 day 2 hours 48 minutes 36 seconds						
\$math[MathFu	nction]						
Description	Allows you to perform math functions inside of Streamlabs Chatbot						
Example	\$math[10+5/2]   !Math						
Response	Bot: 12						
\$extralifegoal							
Description	Grabs your Extra Life goal						
Example	\$extralifegoal   !goal						
Response	Bot: 5000						
\$extralifeamou	nt						
Description	Grabs the amount you currently raised for Extra Life						
Example	\$extralifeamount   !amount						
Response	Bot: 100						

©Streamlabs.com

## **Currency Parameters**

\$	р	o	i	n	t	s
~	r	•	•	••	•	_

\$points	
Description	Displays the num of points of the user or target
Example	\$username has \$points \$currencyname   !cookies or !cookies ankhheart
Response	Bot: AnkhHeart has 1234 Cookies!
\$pointstext	
Description	Displays the num of points of the user or target nicely formatted
Example	\$username has \$pointstext \$currencyname   !cookies or !cookies ankhheart
Response	Bot: AnkhHeart has 1,234 Cookies!
\$raids	
Description	Displays amount of times the user or target has raided the channel
Example	\$username has raided the channel \$raids time(s) so far!   !raids or !raids AnkhHeart
Response	Bot: AnkhHeart has raided the channel 3 time(s) so far!
\$rank	
Description	Displays the users rank
Example	\$username is Rank: \$rank   !rank or !rank AnkhHeart
Response	Bot: AnkhHeart is Rank: Ninja Kitty
\$hours	
Description	Displays amount of hours the user has been in the stream for
Example	\$username spent \$hours in the stream!   !hrs
Response	Bot: AnkhHeart spent 10.5 hrs in the stream!
\$level	
Description	Displays the users level
Example	\$username is Level \$level!   !Lvl
Response	Bot: AnkhHeart is Level 10
\$toppoints(nu	m)
Description	Displays top X amount of users based on points (Except Caster & Bots)
Example	Top 3: \$toppoints(3) } !top3
Response	Bot: Top 3: #1 Promouse(10000), #2 Gamegooru21(9999), #3 EdeMonster(9998)
\$tophours(nun	1)
Description	Displays top X amount of users based on hours(Except Caster & Bots)
Example	Top 2: \$tophours(2)   !top2
Response	Bot: Top 2: #1 KrystalRayne(123 Hrs), #2 Pixelmonkey (120 Hrs)
\$pointspos	
Description	Displays the users position in the ranking based on amount of points
Example	\$username is ranked #\$pointspos   !mypos
Response	Bot: AnkhHeart is ranked #1
\$hourspos	
Description	Displays the users position in the ranking based on amount of hours
Example	\$username is ranked #\$hourspos   !hrspos
Response	Bot: AnkhHeart is ranked #2
\$nxtrankreq	
Description	Displays the amount of points/hours the user requires for his next rank

Example	\$username, You need \$nxtrankreq points to become a \$nxtrank!
Response	Bot: AnkhHeart, You need 13 points to become a Ninja Kitty!
\$nxtrank	
Description	Displays the next rank that the user can achieve
Example	\$username, The next rank that you can achieve is \$nxtrank!
Response	Bot: AnkhHeart, The next rank that you can achieve is Ninja Kitty!
\$addpoints("ta	rgetid","min","max","succeed","fail")
Description	Adds points to a certain user and sends a message upon succeeding / failing
Example	\$addpoints("ankhheart","10","50","ankhheart Got \$value points","Failed to give points!")   !addpoints
Response	Bot: AnkhHeart got 25 points
\$removepoints	("targetid","min","max","succeed","fail","forceremove true or false")
Description	Removes points from a certain user and sends a message upon
	succeeding/failing.
	Force remove(true/false) removes points even if the user doesn't have
	enough.
Example	\$removepoints("ankhheart","10","100","Removed \$value points from
	ankhheart.","Unable To remove \$value points from ankhheart!","false")   !removepoints
Response	Bot: Removed 85 points from ankhheart.
\$givepoints("fi	romid","toid","num","succeed","fail","forcegive true or false")
Description	Gives points from one person to another.
Example	\$givepoints("\$userid","\$targetid","50","\$username gave \$value points to
	\$targetname","\$username didn't have enough points to give to
	\$targetname!","false")   !give gamegooru21
Response	Bot: AnkhHeart gave 50 points to Gamegooru21
\$value [Only W	orks inside of \$addpoints, \$givepoints or \$removepoints]
Description	Gets replaced with the random value between min & max
Example	\$givepoints("\$userid","\$targetid","50","\$username gave \$value points to
	\$targetname","\$username didn't have enough points to give to
	\$targetname!","false")  !give gamegooru21
_	D : 4 1111 : 50

\$newbalance(targetid) [Only Works inside of \$addpoints, \$givepoints or \$removepoints]

Bot: AnkhHeart gave 50 gamegooru21

fine in a management of the ma	
Description	Gets replaced with the remaining balance after a \$removepoints, \$addpoints or \$givepoints transaction
Example	\$givepoints("\$userid","\$targetid","\$0","\$username gave \$value points to \$targetname. \$targetname: \$newbalance(\$targetid) points remaining.","fail","false")  !give gamegooru21
Response	Bot: AnkhHeart gave 50 gamegooru21. AnkhHeart 50 remaining

## Twitch API Parameters

#### **User Channel Data**

#### \$userurl

Response

yuseruri	
Description	Displays the user's twitch channel URL
Example	\$user's twitch channel is: \$userurl   !userurl

Response	Bot: AnkhHeart's twitch channel is: http://twitch.tv/AnkhHeart
response	bot. 7 than reares eviter charmer is. http://twiteri.ev/7 than reare

#### \$usergame

Description	Displays the user's last played/current game
Example	\$user's was/is playing: \$usergame   !usergame
Response	Bot: AnkhHeart was/is playing: Bloodborne

#### **\$userstatus**

Description	Displays the user's stream title
Example	\$user's Stream title is: \$userstatus   !userstatus
Response	Bot: AnkhHeart's Stream title is: [720p] Fable: The Lost Chapters [PC]

## **Target Channel Data**

#### \$url

Description	Displays the target's twitch channel URL
Example	\$targetname can be found streaming at: \$url   !url AnkhHeart
Response	Bot: AnkhHeart can be found streaming at: http://twitch.tv/AnkhHeart

## \$game

Description	Displays the target's current/last played game
Example	\$targetname has last played: \$game   !game AnkhHeart
Response	Bot: AnkhHeart has last played: Bloodborne

#### \$status

Description	Displays the target's stream title
Example	\$targetname Stream title is: \$title   !title AnkhHeart
Response	Bot: AnkhHeart Stream title is: [720p] Fable: The Lost Chapters [PC]

## My Channel Data

## \$myurl

Description	Displays the twitch channel URL for your stream
Example	My channel is: \$myurl.   !myurl
Response	Bot: My channel is: http://twitch.tv/AnkhHeart

#### \$mygame

Description	Displays the game you are currently playing
Example	I am playing: \$mygame   !currentgame
Response	Bot: I am playing: Bloodborne

#### \$mystatus

Description	Displays your stream title
Example	Status: \$mystatus   !mystatus
Response	Bot: Status: [720p] Fable: The Lost Chapters [PC]

#### **\$uptime**

Description	Displays for how long the stream has been Live
Example	The stream has been live for: \$uptime   !uptime
Response	Bot: The stream has been live for: 1 hour 25 minutes 58 seconds

## Sub / Follower Counts

#### \$followercount

T	* · · · · · · · · · · · · · · · · · · ·	
Description	Displays your streams follower count	
Example	AnkhHeart has \$followercount followers! !followercount	

Response	Bot: AnkhHeart has 1070 followers!	
\$subcount		
Description	Displays your streams sub count	
Example	AnkhHeart has \$subcount subs! !subcount	
Response	Bot: AnkhHeart has 0 subs!	
\$gwsubcount		
Description	Displays your gamewisp sub count	
Example	AnkhHeart has \$gwsubcount subs! !gwsubcount	
Response	Bot: AnkhHeart has 0 gamewisp subs!	

#### Miscellaneous

#### \$currenthosts(NumHostsPerPage)

+ ca c	tuminosts: en age,
Description	Displays everyone that is currently hosting the stream (Only works when you're live)
Example	Current Hosts: \$currenthosts(2)
Response	Bot: Current Hosts: EdeMonster, Promouse [Page 0/1]

#### \$setgame(game) and \$settitle(title)

Description	Allows you to set the game & title through a command and create presets
	for certain games
Example	\$setgame(Dungeon Defenders II) \$settitle(MMO Mornings)  !dd2
Response	No response as the bot will simply update the game & title

## File Reading Parameters

#### \$readline(FileLocation)

Description	Reads the first line of the document
Example	<pre>\$readline(C:\test.txt)   !currentsong</pre>
Response	Bot: Currently playing: Popskyy - Rize Up

#### readrandline(FileLocation)

Description	Reads a random line from the file
Example	/me slaps \$randusername with a \$readrandline(C:\test2.txt)!   !slap
Response	Bot slaps AnkhHeart with a Tuna!   Bot slaps AnkhHeart with a Brick!   etc

#### \$readspecificline(FileLocation,LineNum)

Description	Reads a specific line from the file (Starts from 0)
Example	/me slaps \$randusername with a \$readspecificline(C:\test2.txt,3)!   !slap
Response	Bot slaps AnkhHeart with a Shovel!

## **Custom API Reading Parameter**

#### \$readapi(URL)

Description	Displays the text on the URL's page. Max 500 characters
Example	\$readapi(https://nightdev.com/hosted/followers.php?channel=ankhheart &limit=5)
Response	Bot: 1. BensGaming808, 2. Gamakuro, 3. GENERAL_XROS, 4. wulleybully, 5. NorQuel

## Save File Parameters

\$savetofile("FileLocation","Text")

\$savetofile("FileLocation","Text","SucceedMsg","FailMsg")

Description	Adds to the end of the file
Example	\$savetofile("C:\test.txt","\$msg","Succeeded :D","Failed! ")   !save I am a cat
Response	Bot: Succeeded!

#### \$overwritefile("FileLocation","Text")

#### \$overwritefile("FileLocation","Text","SucceedMsg","FailMsg")

Description	Overwrites all the data in the .txt file with the added text
Example	\$overwritefile("C:\test.txt","\$msg","Succeeded :D","Failed! ")   !save I am a
	cat
Response	Bot: Succeeded!

## Miscellaneous Parameters

#### \$tier

Description	Only usable in the GameWisp Chat Notification	
Example	\$username just subbed Tier: \$tier!	
Response	Bot: AnkhHeart just subbed Tier: 1!	

#### \$months

Description	Only usable in the GameWisp Chat Resub / Twitch Resub Notification	
Example	\$username just subbed for \$months months in a row!	
Response	Bot: AnkhHeart just subbed for 3 months in a row!	

#### \$reward

Description	Only usable in the GameWisp Chat Notification	
Example	\$username just subbed for \$months months in a row and received \$reward points!	
Response	Bot: AnkhHeart just subbed for 3 months in a row and received 10,000 points!	

#### \$bits / \$totalbits

Description	Only usable in the Cheer Chat Notification	
Example	\$username just cheered \$bits bits for a total of \$totalbits bits!	
Response	Bot: AnkhHeart just cheered 100 bits for a total of 1234 bits!	

#### \$donationmsg

Description	Only usable in the Streamlabs Chat Notification	
Example	\$username just donated \$amount USD! Message: \$donationmsg	
Response	Bot: AnkhHeart just donated 10 USD! Message: Harro <3	

#### **\$viewers**

Description	Only usable in the Host Chat Notification	
Example	\$username just hosted you for \$viewers viewer(s)!	
Response	Bot: AnkhHeart just hosted you for 10 viewer(s)!	

# **Generated Text Files**

The bot automatically generates text files that can be used to display information on stream. These files can be found in the Bot's Install Directory -> Services -> Twitch -> Files folder. If you don't remember where you installed the bot just Right click on its shortcut and select

Open File Location. If this leads you to the Startup folder instead do it once more on the shortcut there and eventually you will end up in the Bot's install Directory.

#### Follower Amount (per session)

•	
File	AmountOfFollowers.txt
New Followe	r List (per session)
File	Followers.txt
Last Follower	
File	RecentFollower.txt

## GameWisp Sub Amount (per session)

RecentGameWispSub.txt

File	AmountOfGameWispSubs.txt	
New GameWisp Subs (per session)		
File	GameWispSubs.txt	
Last GameWisp Sub		

#### Host Amount (per session)

File	AmountOfHosts.txt
New Hosts (	per session)
File	Hosts.txt
Last Host	
File	RecentHost txt

#### Twitch Sub Amount (per session)

File	AmountOfSubs.txt	
New Subs (per session)		
File	Subs.txt	

#### Last Sub

File

File

File	RecentSub.txt	
------	---------------	--

#### **Current Song**

Requested By	
File	RequestedBy.txt

#### Complete Current Song + Requested By

CurrentSong.txt

File	CurrentlyPlaying.txt	

#### **Death Counter**

File	Deaths.txt
------	------------

#### **Recent Donator**

File	Streamlabs_Recent_Donator.txt

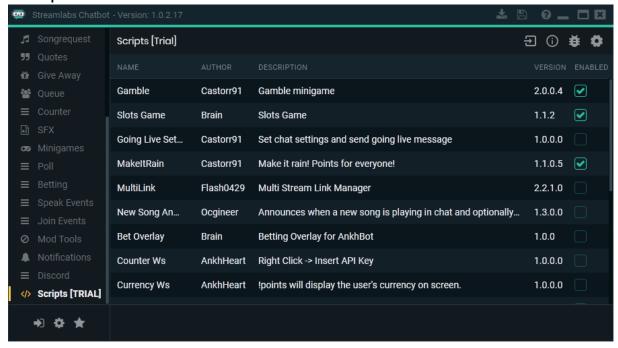
#### Extra Life (Raised / Goal)

•		
File	ExtraLife.txt	
Extra Life Team (Raised / Goal)		
File	ExtraLife_Team.txt	
Extra Life Donations (per Session)		
File	ExtraLife_Donators.txt	
Extra Life Last Donator		
File	ExtraLife_Recent_Donator.txt	

©Streamlabs.com

# Streamlabs Chatbot Python Scripting

## Setup



In order to properly use Python Scripts within Streamlabs Chatbot you need to have Python 2.7 installed: <a href="https://www.python.org/ftp/python/2.7.13/python-2.7.13.msi">https://www.python.org/ftp/python/2.7.13/python-2.7.13.msi</a>

Once you've done that click on the Settings Icon inside of the Scripts tab -> Pick Folder and navigate to your C:\Python27\Lib folder on your System. This can vary from the one that I am referring to since it all depends on where you installed Python. Once you've found the Lib folder select it and hit Save.

Afterwards you can right click within the Scripts View and reload any scripts. At the bottom of the page you will see Errors being logged from the Python Scripts that you're trying to load in case they are not considered valid or have bugs in them.

#### **Importing**

Importing a script is simple. Simply click the Import Button in the Scripts Tab, Navigate to the Zip File and Open it. Afterwards the bot will import the script for you and reload your scripts so it's ready to go.

#### Assigning API Key to Script

Whenever you right click on a script you're able to select `Insert Api Key` which will put a small .js file in the script's folder that will give it access to connect to the chatbot's websocket server.



The Api Key in question can be refreshed whenever you wish. Though in this case you will need to update each client.js file so it uses the appropriate key. This key is used as a password so only clients which you have granted access may connect to the server.

#### Creating your own script?

If you want to create your own scripts please checkout the Chatbot boilerplate over at: Wiki: <a href="https://github.com/AnkhHeart/Streamlabs-Chatbot-Python-Boilerplate/wiki">https://github.com/AnkhHeart/Streamlabs-Chatbot-Python-Boilerplate</a> Code: <a href="https://github.com/AnkhHeart/Streamlabs-Chatbot-Python-Boilerplate">https://github.com/AnkhHeart/Streamlabs-Chatbot-Python-Boilerplate</a>

## **FAQ**

For the FAQ please check the website!

http://www.Streamlabs Chatbot.com/faq/