

## My Project

Generated by Doxygen 1.8.14



# Contents

<b>1</b>	<b>README</b>	<b>1</b>
<b>2</b>	<b>Class Index</b>	<b>3</b>
2.1	Class List . . . . .	3
<b>3</b>	<b>File Index</b>	<b>5</b>
3.1	File List . . . . .	5
<b>4</b>	<b>Class Documentation</b>	<b>7</b>
4.1	Comm.Comm Class Reference . . . . .	7
4.1.1	Detailed Description . . . . .	7
4.1.2	Member Function Documentation . . . . .	8
4.1.2.1	on_message_delete() . . . . .	8
4.1.2.2	on_message_edit() . . . . .	8
4.1.3	Member Data Documentation . . . . .	8
4.1.3.1	pass_context . . . . .	8
4.2	Image.Image Class Reference . . . . .	9
4.2.1	Detailed Description . . . . .	9
4.2.2	Member Data Documentation . . . . .	9
4.2.2.1	name . . . . .	9
4.3	Level.Level Class Reference . . . . .	10
4.3.1	Detailed Description . . . . .	10
4.3.2	Member Function Documentation . . . . .	10
4.3.2.1	add_experience() . . . . .	10
4.3.2.2	level_up() . . . . .	11
4.3.2.3	update_data() . . . . .	11
4.4	Misc.Misc Class Reference . . . . .	11
4.4.1	Detailed Description . . . . .	12
4.4.2	Member Function Documentation . . . . .	12
4.4.2.1	echo() . . . . .	12
4.4.2.2	owner() . . . . .	12
4.5	Music.Music Class Reference . . . . .	13
4.5.1	Detailed Description . . . . .	13
4.5.2	Member Data Documentation . . . . .	13
4.5.2.1	pass_context . . . . .	13

<b>5 File Documentation</b>	<b>15</b>
5.1 Bot.py File Reference . . . . .	15
5.1.1 Detailed Description . . . . .	16
5.1.2 Function Documentation . . . . .	16
5.1.2.1 load() . . . . .	16
5.1.2.2 on_member_join() . . . . .	16
5.1.2.3 on_message() . . . . .	16
5.1.2.4 unload() . . . . .	17
5.1.3 Variable Documentation . . . . .	17
5.1.3.1 pass_context . . . . .	17
5.2 Comm.py File Reference . . . . .	17
5.2.1 Detailed Description . . . . .	18
5.3 Image.py File Reference . . . . .	18
5.3.1 Detailed Description . . . . .	18
5.4 Level.py File Reference . . . . .	18
5.4.1 Detailed Description . . . . .	19
5.5 Misc.py File Reference . . . . .	19
5.5.1 Detailed Description . . . . .	19
5.6 Music.py File Reference . . . . .	19
5.6.1 Detailed Description . . . . .	20
<b>Index</b>	<b>21</b>

# Chapter 1

## README

### Steps to running bot

1. `python -m install -U discord.py`
  2. <https://wwwffmpeg.org/> -Link to path in windows envrionemnt, this is for playing youtube audio
  3. `python -m pip install -U youtube_dl`
  4. `python -m pip install -U imgurpython`
  5. Visit discord's developer's portal at <https://discordapp.com/developers/applications/>
  6. login and create an appliaction at
  7. Click bot tab on the left
  8. Click add bot
  9. Click OAuth2 tab on the left
  10. Click bot checkbox in scopes
  11. Copy link and paste to invite bot to server
  12. Replace token in [Bot.py](#). This is found in the bot tab under username.
- 
1. run [bot.py](#)



## Chapter 2

# Class Index

### 2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">Comm.Comm</a>	Class for implementing communication methods . . . . .	7
<a href="#">Image.Image</a>	Class for implementing image search methods . . . . .	9
<a href="#">Level.Level</a>	Class for implementing leveling system methods . . . . .	10
<a href="#">Misc.Misc</a>	Class for miscellaneous methods . . . . .	11
<a href="#">Music.Music</a>	Class for implementing a music player . . . . .	13





## Chapter 3

# File Index

### 3.1 File List

Here is a list of all documented files with brief descriptions:

<a href="#">Bot.py</a>	Initializes bot client and module loading . . . . .	15
<a href="#">Comm.py</a>	Implements methods for communication in Discord chat bot . . . . .	17
<a href="#">Image.py</a>	Implements methods for Imgur image search in Discord chat bot . . . . .	18
<a href="#">Level.py</a>	Implements methods for leveling system in Discord chat bot. Only provides functionality . . . .	18
<a href="#">Misc.py</a>	Miscellaneous functions for bot. Does not have real application to chatbot . . . . .	19
<a href="#">Music.py</a>	Implements methods for a music play in Discord chat bot . . . . .	19



## Chapter 4

# Class Documentation

### 4.1 Comm.Comm Class Reference

Class for implementing communication methods.

#### Public Member Functions

- `def __init__ (self, client)`
- `def on_message_delete (self, message)`  
*Bot notifies community server when message is deleted.*
- `def on_message_edit (self, before, after)`  
*Bot notifies community server when message is edited.*
- `def on_reaction_add (self, reaction, user)`
- `def clear (self, ctx, amount=5)`

#### Public Attributes

- `client`
  - `ONDELETE`
  - `ONEDIT`
- Displays a particular news article fomr news.com.au about blackholes.*

#### Static Public Attributes

- `int ONDELETE = 0`
- `int ONEDIT = 0`
- `pass_context`  
*Clears messages in channel.*

#### 4.1.1 Detailed Description

Class for implementing communication methods.

## 4.1.2 Member Function Documentation

### 4.1.2.1 on\_message\_delete()

```
def Comm.Comm.on_message_delete (
    self,
    message )
```

Bot notifies community server when message is deleted.

Prints 'Message deleted by <user>' into server

#### Parameters

<i>message</i>	Message that was typed into Discord server
----------------	--

### 4.1.2.2 on\_message\_edit()

```
def Comm.Comm.on_message_edit (
    self,
    before,
    after )
```

Bot notifies community server when message is edited.

Prints 'Message edited by <user>' into server

#### Parameters

<i>message</i>	Message that was typed into Discord server
----------------	--

## 4.1.3 Member Data Documentation

### 4.1.3.1 pass\_context

```
Comm.Comm.pass_context [static]
```

Clears messages in channel.

Clears <amount> messages in channel. Default is 5 messages after command unless specified. Called by !clears [int]

## Parameters

<i>ctx</i>	Discord chat context
<i>amount</i>	Number of message to delete. Default is 5.

The documentation for this class was generated from the following file:

- [Comm.py](#)

## 4.2 Image.Image Class Reference

Class for implementing image search methods.

### Public Member Functions

- `def \_\_init\_\_ (self, client)`  
*Initializes image module.*
- `def imgurService (self, ctx)`

### Public Attributes

- **client**
- **imgur**

### Static Public Attributes

- [name](#)  
*Helper function for image search.*
- **no\_pm**
- **True**
- **pass\_context**

#### 4.2.1 Detailed Description

Class for implementing image search methods.

#### 4.2.2 Member Data Documentation

##### 4.2.2.1 name

```
Image.Image.name [static]
```

Helper function for image search.

#### Parameters

<code>ctx</code>	Discord context
------------------	-----------------

The documentation for this class was generated from the following file:

- [Image.py](#)

## 4.3 Level.Level Class Reference

Class for implementing leveling system methods.

### Public Member Functions

- `def __init__ (self, client)`  
*Initializes level module.*
- `def update_data (self, users, member)`  
*Creates a member expereince profile.*
- `def add_experience (self, users, member, exp)`  
*Adds experience to a member expereince profile.*
- `def level_up (self, users, member, channel)`  
*Congradulates member for reaching next level in experience profile.*

### Public Attributes

- `client`

### 4.3.1 Detailed Description

Class for implementing leveling system methods.

### 4.3.2 Member Function Documentation

#### 4.3.2.1 add\_experience()

```
def Level.Level.add_experience (  
    self,  
    users,  
    member,  
    exp )
```

Adds experience to a member expereince profile.

## Parameters

<i>users</i>	JSON Dictionary of members
<i>member</i>	Member that is being updated
<i>exp</i>	Number of experience awarded to member

## 4.3.2.2 level\_up()

```
def Level.Level.level_up (
    self,
    users,
    member,
    channel )
```

Congradulates member for reaching next level in experience profile.

## Parameters

<i>users</i>	JSON Dictionary of members
<i>member</i>	Member that is being updated
<i>channel</i>	Where bot outputs message

## 4.3.2.3 update\_data()

```
def Level.Level.update_data (
    self,
    users,
    member )
```

Creates a member expereince profile.

## Parameters

<i>users</i>	JSON Dictionary of members
<i>member</i>	Member that is being updated

The documentation for this class was generated from the following file:

- [Level.py](#)

## 4.4 Misc.Misc Class Reference

Class for miscellaneous methods.

## Public Member Functions

- `def \_\_init\_\_ (self, client)`  
*Initializes miscellaneous module.*
- `def owner (self)`  
*Bot prints the owner of the bot created.*
- `def echo (self, args)`  
*Echo message in channel.*

## Public Attributes

- **client**

### 4.4.1 Detailed Description

Class for miscellaneous methods.

### 4.4.2 Member Function Documentation

#### 4.4.2.1 `echo()`

```
def Misc.Misc.echo (
    self,
    args )
```

Echo message in channel.

##### Parameters

<code>args</code>	String message to be echoed
-------------------	-----------------------------

#### 4.4.2.2 `owner()`

```
def Misc.Misc.owner (
    self )
```

Bot prints the owner of the bot created.

Prints 'SE3XA3'

The documentation for this class was generated from the following file:

- [Misc.py](#)



## 4.5 Music.Music Class Reference

Class for implementing a music player.

### Public Member Functions

- `def \_\_init\_\_ (self, client)`  
*Initializes music module.*
- `def join (self, ctx)`

### Public Attributes

- **client**

### Static Public Attributes

- dictionary **players** = {}
- dictionary **queues** = {}
- [pass\\_context](#)  
*Bot joins channel user is in.*

#### 4.5.1 Detailed Description

Class for implementing a music player.

#### 4.5.2 Member Data Documentation

##### 4.5.2.1 `pass_context`

`Music.Music.pass_context` [static]

Bot joins channel user is in.

Called by !join

##### Parameters

<code>ctx</code>	Discord chat context
------------------	----------------------

The documentation for this class was generated from the following file:

- [Music.py](#)



## Chapter 5

# File Documentation

### 5.1 Bot.py File Reference

Initializes bot client and module loading.

#### Functions

- def `Bot.on_ready ()`  
*Detects when bot is ready and outputs into terminal.*
- def `Bot.on_message (message)`  
*Displays messages sent in Discord server into terminal.*
- def `Bot.on_member_join (member)`  
*Assigns role and level to new member joining server.*
- def `Bot.unload (extension)`  
*Manually loads module(Cog)*
- def `Bot.load (extension)`  
*Manually loads module(Cog)*
- def `Bot.helps (ctx)`

#### Variables

- string `Bot.TOKEN` = 'NTE2Mzc4ODU4Mzg0MTMwMDQ4.Dtyy6w.Gs8YYm6ygzMPJfEyoPfvie7MvFk'
- `Bot.mod` = `os.path.dirname(os.path.abspath(__file__))`
- `Bot.client` = `commands.Bot(command_prefix = '!')`
- `Bot.level` = `Level.Level(client)`
- list `Bot.extensions` = ['Music', 'Comm', 'Misc', 'Image']
- `Bot.pass_context`  
*Bot privately messages user a list of commands the bot supports.*

### 5.1.1 Detailed Description

Initializes bot client and module loading.

#### Author

Jason Tsui

#### Date

9/11/2018

### 5.1.2 Function Documentation

#### 5.1.2.1 load()

```
def Bot.load (
    extension )
```

Manually loads module(Cog)

#### Parameters

<i>extension</i>	name of module to be imported
------------------	-------------------------------

#### 5.1.2.2 on\_member\_join()

```
def Bot.on_member_join (
    member )
```

Assigns role and level to new member joining server.

#### Parameters

<i>member</i>	Member that is joining the server
---------------	-----------------------------------

#### 5.1.2.3 on\_message()

```
def Bot.on_message (
    message )
```

Displays messages sent in Discord server into terminal.

**Parameters**

<code>message</code>	Message typed into Discord server
----------------------	-----------------------------------

**5.1.2.4 unload()**

```
def Bot.unload (
    extension )
```

Manually loads module(Cog)

**Parameters**

<code>extension</code>	name of module to be removed
------------------------	------------------------------

**5.1.3 Variable Documentation****5.1.3.1 pass\_context**

```
Bot.pass_context
```

Bot privately messages user a list of commands the bot supports.

!help

**5.2 Comm.py File Reference**

Implements methods for communication in Discord chat bot.

**Classes**

- class [Comm.Comm](#)  
*Class for implementing communication methods.*

**Functions**

- def **Comm.setup** (client)

### 5.2.1 Detailed Description

Implements methods for communication in Discord chat bot.

#### Author

Jason Tsui

#### Date

11/14/2018

## 5.3 Image.py File Reference

Implements methods for Imgur image search in Discord chat bot.

### Classes

- class [Image.Image](#)  
*Class for implementing image search methods.*

### Functions

- def **Image.setup** (client)

### 5.3.1 Detailed Description

Implements methods for Imgur image search in Discord chat bot.

#### Author

Abdul Elrahwan

#### Date

11/14/2018

## 5.4 Level.py File Reference

Implements methods for leveling system in Discord chat bot. Only provides functionality.

### Classes

- class [Level.Level](#)  
*Class for implementing leveling system methods.*

## Functions

- def **Level.setup** (client)

### 5.4.1 Detailed Description

Implements methods for leveling system in Discord chat bot. Only provides functionality.

#### Author

Jason Tsui

#### Date

11/10/2018

## 5.5 Misc.py File Reference

Miscellaneous functions for bot. Does not have real application to chatbot.

## Classes

- class [Misc.Misc](#)  
*Class for miscellaneous methods.*

## Functions

- def **Misc.setup** (client)

### 5.5.1 Detailed Description

Miscellaneous functions for bot. Does not have real application to chatbot.

#### Author

Jason Tsui

#### Date

11/01/2018

## 5.6 Music.py File Reference

Implements methods for a music play in Discord chat bot.

## Classes

- class [Music.Music](#)  
*Class for implementing a music player.*

## Functions

- def **Music.setup** (client)

### 5.6.1 Detailed Description

Implements methods for a music play in Discord chat bot.

#### Author

Jason Tsui

#### Date

11/09/2018



# Index

- add\_experience
  - Level::Level, 10
- Bot.py, 15
  - load, 16
  - on\_member\_join, 16
  - on\_message, 16
  - pass\_context, 17
  - unload, 17
- Comm.Comm, 7
- Comm.py, 17
- Comm::Comm
  - on\_message\_delete, 8
  - on\_message\_edit, 8
  - pass\_context, 8
- echo
  - Misc::Misc, 12
- Image.Image, 9
- Image.py, 18
- Image::Image
  - name, 9
- Level.Level, 10
- Level.py, 18
- Level::Level
  - add\_experience, 10
  - level\_up, 11
  - update\_data, 11
- level\_up
  - Level::Level, 11
- load
  - Bot.py, 16
- Misc.Misc, 11
- Misc.py, 19
- Misc::Misc
  - echo, 12
  - owner, 12
- Music.Music, 13
- Music.py, 19
- Music::Music
  - pass\_context, 13
- name
  - Image::Image, 9
- on\_member\_join
  - Bot.py, 16
- on\_message
  - Bot.py, 16
- on\_message\_delete
  - Comm::Comm, 8
- on\_message\_edit
  - Comm::Comm, 8
- owner
  - Misc::Misc, 12
- pass\_context
  - Bot.py, 17
  - Comm::Comm, 8
  - Music::Music, 13
- unload
  - Bot.py, 17
- update\_data
  - Level::Level, 11