EVE Market viewer

0.01

Generated by Doxygen 1.9.5

Chapter 1

This is the <strong>HOMEPAGE</strong>.

Refer to `Markdown` for how to write markdown files.

1.1 Quick Start Notes:

1. Add images to the *images* folder if the file is referencing an image.

Chapter 2

# Program usage

2.1 Commands

```
authorize
```

- Used to authoriza a character

```
logout
```

- Removes the current character authorization

```
set region "{region_name}"
```

- Sets the default region to the given region

```
set station "{station_name}"
```

- Sets the default station the the given station

```
orders "{type_name}" ["{region_name}"] [-a] [-de] [-d] [-v] [-p]
```

- Displays current buy and sell orders for type {type_name} in region {region_name}, or in the default one if no region name is provided.
- Sell orders are by default price ascending, buy orders price price descending.
- Additional options: -a : Sort type ascending -de : Sort type descending -d : Sort by date -v : Sort by volume remaining -p : Sort by price

```
wallet
```

- Displays the current account balance of the authorized character transactions
- Displays the transaction history of the authorized character

```
my_orders [-a] [-de] [-d] [-v] [-p]
```

- Displays the active market orders of the authorized character
- Additional options: -a : Sort type ascending -de : Sort type descending -d : Sort by date -v : Sort by volume remaining -p : Sort by price

```
my_order_history [-a] [-de] [-d] [-v] [-p]
```

- Displays all completed market orders of the authorized character
- Additional options: -a : Sort type ascending -de : Sort type descending -d : Sort by date -v : Sort by volume remaining -p : Sort by price

```
info "{type_name}"
```

- Displays information about the given type history "{type_name}" ["{region_name}"]
- Displays price and volume history of type {type_name} in region {region_name}, or the default region if no region name specified.

```
assets
```

- Displays all assets of the authorized character contracts page_number ["{region_name}"]
- Displays page page_number of public contracts (or all contracts available to the authorized character) in the given region, or the default one if no region name is provided.

```
my_contracts
```

- Displays all contracts of the authorized character.

```
exit
```

- Exits the program.

```
help
```

- Displays the help

Chapter 3

Namespace Index

3.1 Package List

Here are the packages with brief descriptions (if available):

Chapter 4

Class Index

4.1 Class List

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

Chapter 5

Namespace Documentation

5.1 eve_market Namespace Reference

Classes

- class ContractInterface

  Main interaction point for the Contract part of ESI.
- class MainEsiInterface

  Main interface for communicating with the ESI API. Holds references to specialized interfaces, has some helper functions and takes care of authentication of the user.
- class MarketInterface

  Class that handles the interaction with the Market part of ESI, as well ass the handling of assets.
- class Printer

  Class that handles printing of various data.
- class Program
- class UniverseInterface

  Class that handles the interaction with the Universe part of ESI, and provides various helpful functions to other parts of the program.

Chapter 6

Class Documentation

6.1 eve_market.ContractInterface Class Reference

Main interaction point for the Contract part of ESI.

Public Member Functions

- ContractInterface (MainEsiInterface @interface, TextWriter textWriter)

  Basic constructor. Only assigns the instances from arguments to fields.
- void HandleContracts (string[ ] tokens)

  Handles the "contracts" command. Displays a given page of contracts in the input region. If no region is given, the default region is used.
- void HandleMyContracts (string[ ] tokens)

  Handles the "my_contracts" command. Displays all contracts of the authorized character.
- List< List< string > > getContractItemString (long contractId, bool isPublic=false)

  Helper function. For a given contract, constructs a string of the items in the given contract, with a specific formatting. If listPublic is false, it is assumed that the contract is part of the authorized characters contracts, otherwise it's public.
- List< string > GetBidStrings (int contractId)

  Helper function. Constructs a string containing all the bids from of the given auction contract. Returns an empty lists if no bids are found.

Public Attributes

- TextWriter **output**

  Text output stream
- MainEsiInterface **mainInterface**

  Reference to a MainInterface instance

6.1.1 Detailed Description

Main interaction point for the Contract part of ESI.

6.1.2 Constructor & Destructor Documentation

6.1.2.1 ContractInterface()

```
eve_market.ContractInterface.ContractInterface (
             MainEsiInterface @ interface,
             TextWriter textWriter )
```

Basic constructor. Only assigns the instances from arguments to fields.

Parameters

interface	Reference to an (already created) MainEsiInterface instance
textWriter	Text output stream

6.1.3 Member Function Documentation

6.1.3.1 GetBidStrings()

```
List< string > eve_market.ContractInterface.GetBidStrings (
             int contractId )
```

Helper function. Constructs a string containing all the bids from of the given auction contract. Returns an empty lists if no bids are found.

Parameters

contract↩ Id	Auction contract ID

Returns

List of bid strings

6.1.3.2 getContractItemString()

```
List< List< string > > eve_market.ContractInterface.getContractItemString (
             long contractId,
             bool isPublic = false )
```

Helper function. For a given contract, constructs a string of the items in the given contract, with a specific formatting. If listPublic is false, it is assumed that the contract is part of the authorized characters contracts, otherwise it's public.

Parameters

contract↵ Id	ID of the contract
isPublic	Whether the contract is a public or character one

Returns

List of item info strings

6.1.3.3 HandleContracts()

```
void eve_market.ContractInterface.HandleContracts (
            string[] tokens )
```

Handles the "contracts" command. Displays a given page of contracts in the input region. If no region is given, the default region is used.

Parameters

tokens	Command line tokens

6.1.3.4 HandleMyContracts()

```
void eve_market.ContractInterface.HandleMyContracts (
            string[] tokens )
```

Handles the "my_contracts" command. Displays all contracts of the authorized character.

Parameters

tokens	Command line tokens (included for uniformity of API, not used)

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

- C:/Users/Samo/Repos/eve_market/ContractInterface.cs

6.2 eve_market.MainEsiInterface Class Reference

Main interface for communicating with the ESI API. Holds references to specialized interfaces, has some helper functions and takes care of authentication of the user.

Public Member Functions

- MainEsiInterface (TextWriter writer)

  Basic contstructor, creates the instances of other interfaces and the Printer, and also creates the EsiClient used for communication with ESI. If there is an auth token from a previous session, uses it again.
- bool IsIdField (string field)

  Helper function which returns true if a given json field is an ID, and thus the corresponding name has to be resolved.
- string GenerateStateString (int length=10)

  Generates a state string of given length, used for OAuth.
- void HandleHelp (string[ ] tokens)

  Handles the "help" command. Prints the text inside the help.txt file.
- string StringFromSlice (string[ ] tokens, int offset, int size, char sep=' ')

  Helper function which creates a string from a given slice of the "tokens" string array.
- bool CheckAuthorization ()

  Checks whether a character is authorized. Returns the value of IsAuthorized an
- async void HandleLogout (string[ ] tokens)

  Logs the authorized character out (if there is any)
- async void HandleAuthorize (string[ ] tokens)

  Handles the authorization of a user. Takes care of the OAuth authorization process, and refers the end-user to the auth page.

Public Attributes

- EsiClient **Client**

  ESI API Client instance
- TextWriter **Output**

  Stream to which output will be written
- MarketInterface **marketInterface**

  MarketInterface instance, used for accessing the Market portion of ESI
- UniverseInterface **universeInterface**

  UniverseInterface instance, used for accessing the Universe portoin of ESI
- ContractInterface **contractInterface**

  ContractInterface instance, used for accessing the Universe portion of ESI
- Printer **printer**

  Printer instance, used for nice printing of various data gathered from ESI

Properties

- bool **IsAuthorized** `[get]`

  Bool which is set to true if a character is authorized
- SsoToken **AuthToken** `[set]`

  Auth Token obtained from the server
- AuthorizedCharacterData **AuthData** `[get, set]`

  Information about the authorized character

6.2.1 Detailed Description

Main interface for communicating with the ESI API. Holds references to specialized interfaces, has some helper functions and takes care of authentication of the user.

6.2.2 Constructor & Destructor Documentation

6.2.2.1 MainEsiInterface()

```
eve_market.MainEsiInterface.MainEsiInterface (
             TextWriter writer )
```

Basic contstructor, creates the instances of other interfaces and the Printer, and also creates the EsiClient used for communication with ESI. If there is an auth token from a previous session, uses it again.

Parameters

writer	Output text stream

6.2.3 Member Function Documentation

6.2.3.1 CheckAuthorization()

```
bool eve_market.MainEsiInterface.CheckAuthorization ( )
```

Checks whether a character is authorized. Returns the value of IsAuthorized an

Returns

6.2.3.2 GenerateStateString()

```
string eve_market.MainEsiInterface.GenerateStateString (
             int length = 10 )
```

Generates a state string of given length, used for OAuth.

Parameters

length	length of the state string

Returns

State string

6.2.3.3 HandleAuthorize()

```
async void eve_market.MainEsiInterface.HandleAuthorize (
             string[] tokens )
```

Handles the authorization of a user. Takes care of the OAuth authorization process, and refers the end-user to the auth page.

Parameters

tokens	Command line tokens. Included for the uniformity of API

6.2.3.4 HandleHelp()

```
void eve_market.MainEsiInterface.HandleHelp (
             string[] tokens )
```

Handles the "help" command. Prints the text inside the help.txt file.

Parameters

tokens	Command line tokens. Only included for uniformity of API

6.2.3.5 HandleLogout()

```
async void eve_market.MainEsiInterface.HandleLogout (
             string[] tokens )
```

Logs the authorized character out (if there is any)

Parameters

tokens	Command line tokens. Only included for the uniformity of API

6.2.3.6 IsIdField()

```
bool eve_market.MainEsiInterface.IsIdField (
             string field )
```

Helper function which returns true if a given json field is an ID, and thus the corresponding name has to be resolved.

Parameters

field	Name of the field

Returns

True if

6.2.3.7 StringFromSlice()

```
string eve_market.MainEsiInterface.StringFromSlice (
             string[] tokens,
             int offset,
             int size,
             char sep = ' ' )
```

Helper function which creates a string from a given slice of the "tokens" string array.

Parameters

tokens	Array of strings
offset	Beginning of the slice
size	Length of the slice
sep	Separator to be added between tokens

Returns

String of joined tokens starting at "offset" with a length of "size", separated by "sep"

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

- C:/Users/Samo/Repos/eve_market/MainEsiInterface.cs

6.3 eve_market.MarketInterface Class Reference

Class that handles the interaction with the Market part of ESI, as well ass the handling of assets.

Public Types

- enum SortOrder

  Enum choosing the sort order of various items
- enum SortBy

  Enum for choosing the field by which orders should be sorted.

Public Member Functions

- MarketInterface (MainEsiInterface @interface, TextWriter textWriter)

  Basic constructor. Only assigns the instances from arguments to fields.
- void HandleDefaults (string[ ] tokens)

  Handles the "set" command. Sets default values for region or station used when displaying market listings.
- void HandleTransactions (string[ ] tokens)

  Handles the "transactions" command. Displays all of the recent transactions of the authorized characters
- void SortOrders (List< Order > orders, SortBy sortByType, SortOrder sortOrder)

  Sorts orders based on various fields defined by SortBy enum and in ascending or descending order
- void SortItems (List< Item > items, SortBy sortByType, SortOrder sortOrder)

  Sorts a list of items according to the given field in the given order
- void HandleHistory (string[ ] tokens)

  Prints type history in a given region.
- void HandleAssets (string[ ] tokens, SortBy sortByType=SortBy.Station, SortOrder sortOrder=SortOrder.↩ Descending)

  Handles the "assets" command. Displays all assets of the authorized character as a table.
- void HandleOrders (string[ ] tokens, SortBy sortByType=SortBy.Price, SortOrder sortOrder=SortOrder.↩ Descending)

  Handles the "orders" command. Finds all orders of the given type in the given location. If not location is given, the default region will be used
- void HandleMyOrders (string[ ] tokens, SortBy sortByType=SortBy.Date, SortOrder sortOrder=SortOrder.↩ Descending)

  Handles the "my_orders" command. Gathers orders from the API, sorts them and prints them out to output
- void HandleOrderHistory (string[ ] tokens, SortBy sortByType=SortBy.Date, SortOrder sortOrder=Sort↩ Order.Descending)

  Handles the "my_order_history" command. Gathers the order history of a given character, potentially sorts it and prints it out.
- async void HandleWallet (string[ ] tokens)

  Gathers information about the authorized characters' wallet and prints it out to output

Public Attributes

- TextWriter **output**

  Text output stream
- MainEsiInterface **mainInterface**

  Reference to a MainEsiInterface instance
- long **defaultRegionId** = 10000002

  Id of the default region ("The Forge" currently)
- long **defaultStationId** = 60003760

  Id of the default station ("Jita IV - Moon 4 - Caldari Navy Assembly Plant")

6.3.1 Detailed Description

Class that handles the interaction with the Market part of ESI, as well ass the handling of assets.

6.3.2 Constructor & Destructor Documentation

6.3.2.1 MarketInterface()

```
eve_market.MarketInterface.MarketInterface (
            MainEsiInterface @ interface,
            TextWriter textWriter )
```

Basic constructor. Only assigns the instances from arguments to fields.

Parameters

interface	Reference to an (already created) MainEsiInterface instance
textWriter	Text output stream

6.3.3 Member Function Documentation

6.3.3.1 HandleAssets()

```
void eve_market.MarketInterface.HandleAssets (
            string[] tokens,
            SortBy sortByType = SortBy.Station,
            SortOrder sortOrder = SortOrder.Descending )
```

Handles the "assets" command. Displays all assets of the authorized character as a table.

Parameters

tokens	Command line tokens
sortByType	Which field to sort by
sortOrder	Ascending/Descending

6.3.3.2 HandleDefaults()

```
void eve_market.MarketInterface.HandleDefaults (
            string[] tokens )
```

Handles the "set" command. Sets default values for region or station used when displaying market listings.

Parameters

tokens	Tokens from the input, including the original command

6.3.3.3 HandleHistory()

```
void eve_market.MarketInterface.HandleHistory (
             string[] tokens )
```

Prints type history in a given region.

Parameters

tokens	Command line tokens

6.3.3.4 HandleMyOrders()

```
void eve_market.MarketInterface.HandleMyOrders (
             string[] tokens,
             SortBy sortByType = SortBy.Date,
             SortOrder sortOrder = SortOrder.Descending )
```

Handles the "my_orders" command. Gathers orders from the API, sorts them and prints them out to output

Parameters

tokens	Contains tokens from the command line.
sortByType	Enum which determines the field by which to sort the orders
sortOrder	Enum for ascending/descending sort order

6.3.3.5 HandleOrderHistory()

```
void eve_market.MarketInterface.HandleOrderHistory (
             string[] tokens,
             SortBy sortByType = SortBy.Date,
             SortOrder sortOrder = SortOrder.Descending )
```

Handles the "my_order_history" command. Gathers the order history of a given character, potentially sorts it and prints it out.

Parameters

tokens	
sortByType	
sortOrder	

6.3.3.6 HandleOrders()

```
void eve_market.MarketInterface.HandleOrders (
             string[] tokens,
             SortBy sortByType = SortBy.Price,
             SortOrder sortOrder = SortOrder.Descending )
```

Handles the "orders" command. Finds all orders of the given type in the given location. If not location is given, the default region will be used

Parameters

tokens	
sortByType	
sortOrder	

6.3.3.7 HandleTransactions()

```
void eve_market.MarketInterface.HandleTransactions (
             string[] tokens )
```

Handles the "transactions" command. Displays all of the recent transactions of the authorized characters

Parameters

tokens	Command line tokens (included for uniformity of API, not used)

6.3.3.8 HandleWallet()

```
async void eve_market.MarketInterface.HandleWallet (
             string[] tokens )
```

Gathers information about the authorized characters' wallet and prints it out to output

Parameters

tokens	Included for the uniformity of API, not used

6.3.3.9 SortItems()

```
void eve_market.MarketInterface.SortItems (
             List< Item > items,
             SortBy sortByType,
             SortOrder sortOrder )
```

Sorts a list of items according to the given field in the given order

Parameters

items	List of items to sort
sortByType	Field to sort by
sortOrder	Order to sort by

6.3.3.10 SortOrders()

```
void eve_market.MarketInterface.SortOrders (
            List< Order > orders,
            SortBy sortByType,
            SortOrder sortOrder )
```

Sorts orders based on various fields defined by SortBy enum and in ascending or descending order

Parameters

orders	
sortByType	
sortOrder	

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

- C:/Users/Samo/Repos/eve_market/MarketInterface.cs

6.4 eve_market.Printer Class Reference

Class that handles printing of various data.

Public Member Functions

- Printer (MainEsiInterface esiInterface, TextWriter writer)

    Basic constructor. Only assigns the instances from arguments to fields.
- void PrintLine (int length)

    Prints a line of a given length to the output stream.
- void PrintLine (List< string > fields, int width)

    Prints a line that adapts to the number of fields, and the width corresponding to one field.
- void PrintTableHeader (List< string > fields, int width)

    Prints the header of a table, which contains the given fields, with a cell width of "width".
- void PrintContracts (List< ESI.NET.Models.Contracts.Contract > contracts, bool listPublic=false)

    Prints the contracts from arguments. Warning, contracts have a lot of variations and this function is quite extensive in its printing. Use listPublic if you're trying to print public contracts.
- void PrintObjList< T > (List< T > objList, int width, int rows, List< string > fields, List< string > fieldDesc)

    Prints a list of json-convertible objects as a table, with a given cell width and number of rows. Only prints the fields named in the "fields". If a field is an ID field, also resolves it's name before printing.

Public Attributes

- TextWriter **output**

  Text output stream
- MainEsiInterface **mainEsiInterface**

  Reference to a MainIntreface instance.

6.4.1 Detailed Description

Class that handles printing of various data.

6.4.2 Constructor & Destructor Documentation

6.4.2.1 Printer()

```
eve_market.Printer.Printer (
            MainEsiInterface esiInterface,
            TextWriter writer )
```

Basic constructor. Only assigns the instances from arguments to fields.

Parameters

esiInterface	Reference to an (already created) MainEsiInterface instance
writer	Text output stream

6.4.3 Member Function Documentation

6.4.3.1 PrintContracts()

```
void eve_market.Printer.PrintContracts (
            List< ESI.NET.Models.Contracts.Contract > contracts,
            bool listPublic = false )
```

Prints the contracts from arguments. Warning, contracts have a lot of variations and this function is quite extensive in its printing. Use listPublic if you're trying to print public contracts.

Parameters

contracts	List of contracts to print
listPublic	Whether public contract is being printed

6.4.3.2 PrintLine() [1/2]

```
void eve_market.Printer.PrintLine (
             int length )
```

Prints a line of a given length to the output stream.

Parameters

length	Length of the line

6.4.3.3 PrintLine() [2/2]

```
void eve_market.Printer.PrintLine (
             List< string > fields,
             int width )
```

Prints a line that adapts to the number of fields, and the width corresponding to one field.

Parameters

fields	List of fields
width	Width for one field

6.4.3.4 PrintObjList< T >()

```
void eve_market.Printer.PrintObjList< T > (
             List< T > objList,
             int width,
             int rows,
             List< string > fields,
             List< string > fieldDesc )
```

Prints a list of json-convertible objects as a table, with a given cell width and number of rows. Only prints the fields named in the "fields". If a field is an ID field, also resolves it's name before printing.

Template Parameters

T	JObject-convertible type

Parameters

Parameters

objList	List of items to print
width	Table cell width
rows	Number of rows. If it's less than the length of "objList", prints only the first "rows" items.
fields	Keys of the fields to print.

6.4.3.5 PrintTableHeader()

```
void eve_market.Printer.PrintTableHeader (
            List< string > fields,
            int width )
```

Prints the header of a table, which contains the given fields, with a cell width of "width".

Parameters

fields	List of field names
width	Width of a cell in the table

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

- C:/Users/Samo/Repos/eve_market/Printer.cs

6.5 eve_market.Program Class Reference

Static Public Member Functions

- static void ParseInput (TextReader streamReader, TextWriter writer, MainEsiInterface apiInterface)

    Simple entry point function to the program. Turns the command line input into tokens and calls a function based on the value of the first token.

6.5.1 Member Function Documentation

6.5.1.1 ParseInput()

```
static void eve_market.Program.ParseInput (
            TextReader streamReader,
            TextWriter writer,
            MainEsiInterface apiInterface )  [static]
```

Simple entry point function to the program. Turns the command line input into tokens and calls a function based on the value of the first token.

Parameters

streamReader	Input stream
writer	Output stream
apiInterface	MainEsiInterface instance

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

- C:/Users/Samo/Repos/eve_market/Program.cs

6.6 eve_market.UniverseInterface Class Reference

Class that handles the interaction with the Universe part of ESI, and provides various helpful functions to other parts of the program.

Public Member Functions

- UniverseInterface (MainEsiInterface @interface, TextWriter textWriter)

  Basic constructor. Only assigns the instances from arguments to fields.
- void PrintItemInfo (string itemName)

  Finds the closest matching item and prints its description
- void HandleInfo (string[ ] tokens)

  HAndles the "info" command.
- long[ ] SearchName (string query, SearchCategory category)

  Find names of types/objects which containg a given string query and returns the list of matching ids
- long FindRegion (long id, SearchCategory type)

  Finds the region of a given object lower in the hierarchy
- long NameToId (string name, SearchCategory category)

  For a given name of object and it's type, finds its ID. Doesn't require exact names, will find the closest match and returns its ID If no matches are found, returns -1
- void ContractIdToName (List< long > ids, string fieldName)

  Function that resolves the ids in a given contract, categories are determined based on the name of the field to which these ids belong. Results are cached and are available later from the IdToName() function.
- List< string > IdToName (List< long > ids)

  Finds the names corresponding to these IDs, caches them and returns them as a list. Make sure all IDs are unique and of the same type.It is better to later query the IdToName(long id) method for the results, instead of using this list.
- string IdToName (long id)

  Overload for singular ids. If id is in the cache, returns it's cached value. Otherwise calls IdToName() with a List of length 1 containing this id.

Public Attributes

- TextWriter **output**

  Text output stream
- MainEsiInterface **mainInterface**

  Reference to a MainEsiInterface instance

6.6.1 Detailed Description

Class that handles the interaction with the Universe part of ESI, and provides various helpful functions to other parts of the program.

6.6.2 Constructor & Destructor Documentation

6.6.2.1 UniverseInterface()

```
eve_market.UniverseInterface.UniverseInterface (
             MainEsiInterface @ interface,
             TextWriter textWriter )
```

Basic constructor. Only assigns the instances from arguments to fields.

Parameters

interface	Reference to an (already created) MainEsiInterface instance
textWriter	Text output stream

6.6.3 Member Function Documentation

6.6.3.1 ContractIdToName()

```
void eve_market.UniverseInterface.ContractIdToName (
             List< long > ids,
             string fieldName )
```

Function that resolves the ids in a given contract, categories are determined based on the name of the field to which these ids belong. Results are cached and are available later from the IdToName() function.

Parameters

ids	List of ids to resolve
fieldName	Name of the field to which these ids belong

6.6.3.2 FindRegion()

```
long eve_market.UniverseInterface.FindRegion (
             long id,
             SearchCategory type )
```

Finds the region of a given object lower in the hierarchy

Parameters

id	ID of the object
type	Type of the object

Returns

ID of the region to which this object belongs

6.6.3.3 HandleInfo()

```
void eve_market.UniverseInterface.HandleInfo (
             string[] tokens )
```

HAndles the "info" command.

Parameters

tokens	

6.6.3.4 IdToName() [1/2]

```
List< string > eve_market.UniverseInterface.IdToName (
             List< long > ids )
```

Finds the names corresponding to these IDs, caches them and returns them as a list. Make sure all IDs are unique and of the same type.It is better to later query the IdToName(long id) method for the results, instead of using this list.

Parameters

ids	List of ids to resolve

Returns

Name corresponding to the given ID

6.6.3.5 IdToName() [2/2]

```
string eve_market.UniverseInterface.IdToName (
             long id )
```

Overload for singular ids. If id is in the cache, returns it's cached value. Otherwise calls IdToName() with a List of length 1 containing this id.

Parameters

id	Id to resolve

Returns

Resolved name

6.6.3.6 NameToId()

```
long eve_market.UniverseInterface.NameToId (
             string name,
             SearchCategory category )
```

For a given name of object and it's type, finds its ID. Doesn't require exact names, will find the closest match and returns its ID If no matches are found, returns -1

Parameters

name	Name of the object
category	Enum describing the category of the object

Returns

ID of the given query type/object

6.6.3.7 PrintItemInfo()

```
void eve_market.UniverseInterface.PrintItemInfo (
             string itemName )
```

Finds the closest matching item and prints its description

Parameters

itemName	Command line tokens

6.6.3.8 SearchName()

```
long[] eve_market.UniverseInterface.SearchName (
            string query,
            SearchCategory category )
```

Find names of types/objects which containg a given string query and returns the list of matching ids

Parameters

query	
category	Enum describing the category of the object

Returns

A list of ids of the matches

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

- C:/Users/Samo/Repos/eve_market/UniverseInterface.cs

Index