PyHal

API Documentation

January 12, 2017

Contents

1 Package hal

1.1 Modules

1.2 Variables

Name	Description
___package___	**Value:** `None`

2 Package hal.files

2.1 Modules

- **models**: Main entities in files, such as documents, folders.
(Section 3, p. 8)

2.2 Variables

Name	Description
___package___	**Value:** `None`

3 Module hal.files.models

`Main entities in files, such as documents, folders.`

3.1 Variables

Name	Description
BAD_CHARS	**Value:** `['.', ':', '"', '\xe2\x80\x99', '&', '720p', '1080p', 'yi...`
RUSSIAN_CHARS	**Value:** `['\xd1\x88', '\xd0\xb0', '\xd0\xb1', '\xd0\xbb', '\xd0\xb...`
VIDEO_FORMAT	**Value:** `['.', '.3g2', '.3gp', '.amv', '.asf', '.avi', '.drc', '.f...`
ARCHIVE_FORMAT	**Value:** `['.7z', '.??_', '.?Q?', '.?Z?', '.a', '.ace', '.afa', '.a...`
SUBTITLE_FORMAT	**Value:** `['.srt', '.sub', '.sbv']`
TEXT_FORMAT	**Value:** `['.cnf', '.conf', '.cfg', '.chm', '.epub', '.log', '.asc'...`
IMAGE_FORMAT	**Value:** `['.ani', '.bmp', '.cal', '.fax', '.gif', '.img', '.jbg', ...`
AUDIO_FORMAT	**Value:** `['.3gp', '.aa', '.aac', '.aax', '.act', '.aiff', '.amr', ...`
PATH_SEPARATOR	**Value:** `'/'`
__package__	**Value:** `'hal.files'`

3.2 Class FileSystem

object
└ **hal.files.models.FileSystem**

Known Subclasses: hal.files.models.Directory, hal.files.models.Document, hal.files.models.MP3Song

3.2.1 Methods

__init__(*self*, *path*)

```
:param path: string
    Path to file
```

Overrides: object.__init__

fix_raw_path(*path*)

```
:param path: string
    Path to fix
:return: string
    Right path
```

remove_year(*name*)

```
:param name: string
    Name to edit
:return: string
    Given string bu with no years.
```

remove_brackets(*name*)

```
:param name: string
    Name to edit
:return: string
    Given string bu with no barckets.
```

extract_name_max_chars(*name*, *max_chars*=64, *blank*=' ')

```
:param name: string
    Name to edit
:param max_chars: int
    Maximum chars of new name
:param blank: string
    Char that represents the blank between words.
:return: string
    Name edited to contain at most max_chars (truncate to nearest word)
```

prettify(*name*, *bad_chars*=['.', ':', '"', '\xe2\x80\x99', '&', '720p', '1080p', 'yi..., *r*=' ')

```
:param name: string
    Name to edit
:param bad_chars: []
    List of bad strings to remove
:param r: string
    Default blanks in name.
:return: string
    Prettier name from given one: replace bad chars with good ones.
```

ls_dir(*path*, *include_hidden*=False)

```
:param path: string
    Path to directory to get list of files and folders
:param include_hidden: bool
    Whether to include hidden files in list.
:return: list
    List of paths in given directory.
```

ls_recurse(*path*, *include_hidden*=False)

```
:param path: string
    Path to directory to get list of files and folders
:param include_hidden: bool
    Whether to include hidden files in list.
:return: list
    List of paths in given directory recursively.
```

ls(*path*, *recurse*, *include_hidden*=False)

```
:param path: string
    Path to directory to get list of files and folders
:param recurse: bool
    Whether to recurse into subdirectories or not.
:param include_hidden: bool
    Whether to include hidden files in list.
:return: list
    List of paths in given directory recursively.
```

is_archive_mac(*self*)

```
:return: True iff document is an MACOSX archive.
```

is_russian(*self*)

```
:return: True iff document has a russian name.
```

trash(*self*)

```
:return: void
    Trash given file/folder
```

rename(*self*, *new_path*)

```
:param new_path: string
    New path to use
:return: void
    Rename to new path
```

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

3.2.2 Properties

Name	Description
Inherited from object	
___class___	

3.3 Class Document

object

hal.files.models.FileSystem

hal.files.models.Document

3.3.1 Methods

___init___(*self*, *path*)

```
:param path: string
    Path to file
```

Overrides: object.___init___

move_file_to_directory(*file_path*, *directory_path*)

```
:param file_path: string
    Path to file to move
:param directory_path: string
    Path to target directory where to move file
:return: void
    Move file to given directory
```

move_file_to_file(*old_path*, *new_path*)

```
:param old_path: string
    Old path of file to move
:param new_path: string
    New path (location) of file
:return: void
    Move file from old location to new one
```

write_data_to_file(*data*, *out_file*)

```
:param data: string
    Data to write to file.
:param out_file: string
    Path to output file.
:return: void
    Writes given data to given path file.
```

extract_name_extension(*file_name*)

```
:param file_name: string
    Name of file
:return: tuple string, string
    Name of file, extension of file
```

get_path_name(*self*)

```
:return: tuple string, string
    Name of path, name of file (or folder)
```

is_video(*self*)

```
:return: True iff document is a video.
```

is_subtitle(*self*)

```
:return: True iff document is a subtitle.
```

is_text(*self*)

```
:return: True iff document is a text file.
```

is_image(*self*)

```
:return: True iff document is an image.
```

is_audio(*self*)

```
:return: True iff document is an audio.
```

is_hidden(*self*)

```
:return: bool
    True iff path is hidden
```

Inherited from hal.files.models.FileSystem(Section 3.2)

extract_name_max_chars(), fix_raw_path(), is_archive_mac(), is_russian(), ls(), ls_dir(), ls_recurse(), prettify(), remove_brackets(), remove_year(), rename(), trash()

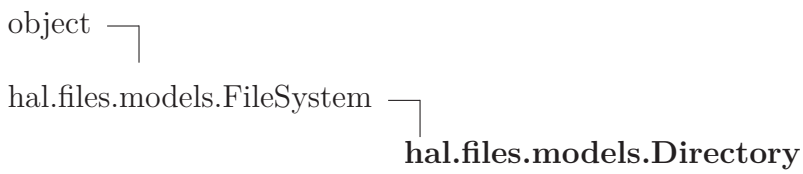
Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

3.3.2 Properties

Name	Description
Inherited from object	
___class___	

3.4 Class Directory



3.4.1 Methods

___init___(*self*, *path*)

```
:param path: string
    Path to file
```

Overrides: object.___init___

create_new(*path*)

```
:param path: string
    Path to directory to create
:return: void
    Creates new directory
```

get_path_name(*self*)

```
:return: tuple string, string
    Name of path, name of file (or folder)
```

is_empty(*self*)

```
:return: Bool
    True iff empty
```

Inherited from hal.files.models.FileSystem(Section 3.2)

extract_name_max_chars(), fix_raw_path(), is_archive_mac(), is_russian(), ls(), ls_dir(), ls_recurse(), prettify(), remove_brackets(), remove_year(), rename(), trash()

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

3.4.2 Properties

Name	Description
Inherited from object	
___class___	

3.5 Class MP3Song

object

hal.files.models.FileSystem

hal.files.models.MP3Song

`mp3 song`

3.5.1 Methods

___init___(*self*, *path*)

```
:param path: string
    Path to file
```

Overrides: object.___init___ extit(inherited documentation)

set_name(*self*, *name*)

set_artist(*self*, *artist*)

set_album(*self*, *album*)

set_nr_track(*self*, *nr_track*)

set_year(*self*, *year*)

Inherited from hal.files.models.FileSystem(Section 3.2)

extract_name_max_chars(), fix_raw_path(), is_archive_mac(), is_russian(), ls(), ls_dir(), ls_recurse(), prettify(), remove_brackets(), remove_year(), rename(), trash()

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

3.5.2 Properties

Name	Description
Inherited from object	
___class___	

4 Package hal.internet

4.1 Modules

- **engines**: Abstract search engines.
  (Section 5, p. 18)
- **parser**: Parse anything there is on the Internet.
  (Section 6, p. 20)
- **selenium**: Some utils methods for a selenium webdriver
  (Section 7, p. 22)
- **web**: Deal with webpages.
  (Section 8, p. 24)
- **youtube**: Get rss feed for youtube channel.
  (Section 9, p. 27)

4.2 Variables

Name	Description
___package___	**Value:** `None`

5 Module hal.internet.engines

`Abstract search engines.`

5.1 Class SearchEngineResult

object
└ **hal.internet.engines.SearchEngineResult**

5.1.1 Methods

___init___(*self*, *title*, *link*, *description*="")

`x.__init__(...) initializes x; see help(type(x)) for signature`

Overrides: object.___init___ extit(inherited documentation)

___str___(*self*)

`str(x)`

Overrides: object.___str___ extit(inherited documentation)

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___subclasshook___()

5.1.2 Properties

Name	Description
Inherited from object	
___class___	

5.2 Class SearchEngine

object
└ **hal.internet.engines.SearchEngine**

5.2.1 Methods

___init___(*self*, *url*, *blank_replace*="+")

```
:param url: string
    Url of search engine used in all query.
:param blank_replace:
    Every search engine has to replace blanks in query
```

Overrides: object.___init___

parse_query(*self*, *query*)

```
:param query: string
    Query to search engine.
:return: string
    Parse given query in order to meet search criteria of search engine.
```

get_search_page(*self*, *query*)

```
:param query: string
    Query to search engine.
:return: string
    Get HTML source of search page of given query.
```

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

5.2.2 Properties

Name	Description
Inherited from object	
___class___	

6 Module hal.internet.parser

```
Parse anything there is on the Internet.
```

6.1 Functions

is_string_well_formatted(*string*)

```
:param string: string
    String to parse
:return: bool
    True iff string is good formatted
```

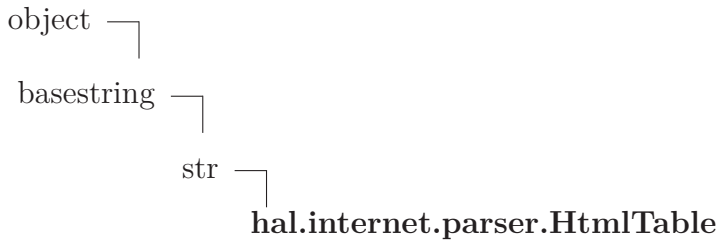
html_stripper(*string*)

```
:param string: string
    String to parse
:return: string
    Given string with raw HTML elements removed
```

6.2 Variables

Name	Description
__package__	**Value:** `'hal.internet'`

6.3 Class HtmlTable



6.3.1 Methods

___init___(*self*, *html_source*)

```
:param html_source: string
    Html source of table
```

Overrides: object.___init___

parse(*self*)

```
:return: list of list
    List of list of values in table
```

Inherited from str

___add___(), ___contains___(), ___eq___(), ___format___(), ___ge___(), ___getattribute___(), ___getitem___(), ___getnewargs___(), ___getslice___(), ___gt___(), ___hash___(), ___le___(), ___len___(), ___lt___(), ___mod___(), ___mul___(), ___ne___(), ___new___(), ___repr___(), ___rmod___(), ___rmul___(), ___sizeof___(), ___str___(), capitalize(), center(), count(), decode(), encode(), endswith(), expandtabs(), find(), format(), index(), isalnum(), isalpha(), isdigit(), islower(), isspace(), istitle(), isupper(), join(), ljust(), lower(), lstrip(), partition(), replace(), rfind(), rindex(), rjust(), rpartition(), rsplit(), rstrip(), split(), splitlines(), startswith(), strip(), swapcase(), title(), translate(), upper(), zfill()

Inherited from object

___delattr___(), ___reduce___(), ___reduce_ex___(), ___setattr___(), ___subclasshook___()

6.3.2 Properties

Name	Description
Inherited from object	
___class___	

7 Module hal.internet.selenium

Some utils methods for a selenium webdriver

7.1 Variables

Name	Description
___package___	**Value:** None

7.2 Class SeleniumForm

Great and simple static methods to deal with selenium webdrivers.

7.2.1 Methods

fill_form_field(*browser*, *field_name*, *field_value*)

```
:param browser: webdriver
    Browser to use to submit form.
:param field_name :string
    Name of field to fill
:param field_value: string
    Value with which to fill field.
:return: void
    Fill given field wiht given value.
```

fill_login_form(*browser*, *username*, *username_field*, *userpassword*, *userpassword_field*)

```
:param browser: webdriver
    Browser to use to submit form.
:param username: string
    Username of user to login.
:param username_field: string
    Name of field to fill with username.
:param userpassword: string
    Password of user to login.
:param userpassword_field: string
    Name of field to fill with userpassword.
:return: void
    Form filled with given information.
```

submit_form(*browser*, *button_name*)

```
:param browser: webdriver
    Browser to use to submit form.
:param button_name: string
    Name of button to press to submit form
:return: void
    Submit form.
```

8 Module hal.internet.web

Deal with webpages.

8.1 Functions

is_url(*candidate_url*)

```
:param candidate_url: str
    Possible url to check for url
:return: bool
    True iff candidate is a valid url
```

download_url(*url*, *local_file*)

```
:param url: string
    Url to download
:param local_file: string
    Save url as this path
:return: void
    Download link to local file
```

8.2 Variables

Name	Description
CHROME_USER_AGENT	**Value:** `["Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US) AppleWe...`
URL_VALID_REGEX	**Value:** `re.compile(r"^(?:http|ftp)s?://" r"(?:(?:[A-Z0-9](?:[A-Z0...`

8.3 Class Webpage

object
└ **hal.internet.web.Webpage**

representation of URL (web page)

8.3.1 Methods

___init___(*self*, *url*, *using_tor*=`False`)

```
:param url: string
    Url of webpage
:param using_tor: bool
    Whether using tor or not to fetch source page
```

Overrides: object.___init___

parse_url(*raw_url*)

```
:param raw_url: url to parse
:return: parses correctly url
```

get_scheme(*self*)

```
:return: get scheme (HTTP, HTTPS, FTP ..) from given url
```

get_hostname(*self*)

```
:return: extract hostname from given url
```

get_domain(*self*)

```
:return: get domain from given url
```

get_html_source(*self*, *tor*=`False`)

```
:return: void
    Saves HTML source of webpage
```

get_links(*self*, *recall*, *timeout*)

```
:param recall: max time to attempt to fetch url
:param timeout: max time (s) to wait for web_page response
:return: array of out_links
```

open_in_browser(*self*, *times*)

```
:param times: int
    Times to open webpage in browser
:return: void
    Open a wendrive and go to webpage
```

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

8.3.2 Properties

Name	Description
Inherited from object	
___class___	

9 Module hal.internet.youtube

Get rss feed for youtube channel.

9.1 Functions

get_channel_page(*channel_name*, *youtube_channel_url*="https://www.youtube.com/user/")

```
@param channel_name: string
    name of channel.
@param youtube_channel_url: string
    base url of youtube channels.
@return string
    source page of youtube channel.
```

get_channel_id(*channel_name*, *channel_id_field*="data-channel-external-id")

```
@param channel_name: string
    channel_name name of channel.
@param channel_id_field: string
    default field to get channel id.
@return string
    id of youtube channel.
```

get_channel_feed_url(*channel_name*, *base_feed_url*="https://www.youtube.com/feeds/videos.xml?channel_id=")

```
@param channel_name: string
    channel_name name of channel.
@param base_feed_url: string
    default base url for rss feed of youtube channels.
@return string
    rss url feed of youtube channel.
```

10 Package hal.maths

`MATHS: important and scalable math functions`

10.1 Modules

- **crypt**: Perform fast hash, encryption and calculations related to cryptography.
  (Section 11, p. 29)
- **maths**: A few elegant and powerful mathematical functions.
  (Section 12, p. 41)
- **plotter**: Show elegant plots in any dimension.
  (Section 13, p. 44)

10.2 Variables

Name	Description
___package___	**Value:** `None`

11 Module hal.maths.crypt

`Perform fast hash, encryption and calculations related to cryptography.`

11.1 Variables

Name	Description
___package___	**Value:** `'hal.maths'`

11.2 Class MD5

object
└ **hal.maths.crypt.MD5**

`md5 hash`

11.2.1 Methods

___init___(*self*, *string*)

`x.__init__(...) initializes x; see help(type(x)) for signature`

Overrides: object.___init___ extit(inherited documentation)

hash(*self*)

`:return: hash plaintext`

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

11.2.2 Properties

Name	Description
Inherited from object	
___class___	

11.3 Class MD6

object
└ **hal.maths.crypt.MD6**

`md6 hash`

11.3.1 Methods

___init___(*self*, *string*, *size*)

`x.__init__(...) initializes x; see help(type(x)) for signature`

Overrides: object.___init___ extit(inherited documentation)

hash(*self*)

`:return: return md6 hash`

hex(*self*, *data*, *size*)

```
:param data: plaintext
:param size: bytes
:return: hex representation
```

raw(*self*, *data*, *size*)

```
:param data: plaintext
:param size: bytes
:return: raw representation
```

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

11.3.2 Properties

Name	Description
Inherited from object	
__class__	

11.3.3 Class Variables

Name	Description
ALLOWED_SIZE	**Value:** `[64, 128, 224, 256, 384, 512]`

11.4 Class SHA

object
└ **hal.maths.crypt.SHA**

`general SHA hash`

11.4.1 Methods

__init__(*self*, *string*, *size*, *salt*=`None`)

`x.__init__(...) initializes x; see help(type(x)) for signature`

Overrides: object.__init__ extit(inherited documentation)

hash(*self*)

`:return: hash of given size`

hash_sha1(*self*)

`:return: sha1 hash`

hash_sha224(*self*)

`:return: sha224 hash`

hash_sha256(*self*)

```
:return: sha256 hash
```

hash_sha384(*self*)

```
:return: sha384 hash
```

hash_sha512(*self*)

```
:return: sha512 hash
```

hash_shasalted(*self*)

```
:return: sha512 hash
```

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

11.4.2 Properties

Name	Description
Inherited from object	
___class___	

11.4.3 Class Variables

Name	Description
ALLOWED_SIZE	**Value:** `[1, 224, 256, 384, 512]`

11.5 Class DES

object
└ **hal.maths.crypt.DES**

`DES hash`

11.5.1 Methods

___init___(*self*, *string*, *key*, *size*)

`x.__init__(...) initializes x; see help(type(x)) for signature`

Overrides: object.___init___ extit(inherited documentation)

hash(*self*)

`:return: hash of given size`

hash_des(*self*)

`:return: des hash`

hash_des3(*self*)

`:return: des3 hash`

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

11.5.2 Properties

Name	Description
Inherited from object	
___class___	

11.5.3 Class Variables

Name	Description
ALLOWED_SIZE	**Value:** `[1, 3]`

11.6 Class ARC

object
└ **hal.maths.crypt.ARC**

`ARC hash`

11.6.1 Methods

___init___(*self*, *string*, *key*, *size*)

`x.__init__(...) initializes x; see help(type(x)) for signature`

Overrides: object.___init___ extit(inherited documentation)

hash(*self*)

`:return: hash of given size`

hash_ar2(*self*)

`:return: des hash`

hash_arc4(*self*)

`:return: des3 hash`

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

11.6.2 Properties

Name	Description
Inherited from object	
___class___	

11.6.3 Class Variables

Name	Description
ALLOWED_SIZE	**Value:** `[2, 4]`

11.7 Class AES

object
└
hal.maths.crypt.AES

`aes hash`

11.7.1 Methods

___init___(*self*, *string*, *key*)

`x.__init__(...) initializes x; see help(type(x)) for signature`

Overrides: object.___init___ extit(inherited documentation)

hash(*self*)

`:return: hash plaintext`

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

11.7.2 Properties

Name	Description
Inherited from object	

continued on next page

Name	Description
___class___	

11.8 Class HMAC

object
└
hal.maths.crypt.HMAC

`hmac hash`

11.8.1 Methods

___init___(*self*, *string*, *key*)

`x.__init__(...) initializes x; see help(type(x)) for signature`

Overrides: object.___init___ extit(inherited documentation)

hash(*self*)

`:return: hash plaintext`

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

11.8.2 Properties

Name	Description
Inherited from object	
___class___	

11.9 Class BLOWFISH

object
└
hal.maths.crypt.BLOWFISH

`blowfish hash`

11.9.1 Methods

___init___(*self*, *string*, *key*)

`x.__init__(...) initializes x; see help(type(x)) for signature`

Overrides: object.___init___ extit(inherited documentation)

hash(*self*)

`:return: hash plaintext`

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

11.9.2 Properties

Name	Description
Inherited from object	
___class___	

11.10 Class IDEA

object
└ **hal.maths.crypt.IDEA**

`IDEA hash`

11.10.1 Methods

___init___(*self*, *string*, *key*)

`x.__init__(...) initializes x; see help(type(x)) for signature`

Overrides: object.___init___ extit(inherited documentation)

hash(*self*)

```
:return: IDEA hash
```

change_key(*self*, *key*)

```
:param key: new key
:return: change key
```

encrypt(*self*)

```
:return: encrypt with key
```

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

11.10.2 Properties

Name	Description
Inherited from object	
___class___	

11.11 Class CAST128

object
└ **hal.maths.crypt.CAST128**

```
CAST 128 hash
```

11.11.1 Methods

___init___(*self*, *string*, *key*)

`x.__init__(...) initializes x; see help(type(x)) for signature`

Overrides: object.___init___ extit(inherited documentation)

encrypt(*self*)

decrypt(*self*)

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

11.11.2 Properties

Name	Description
Inherited from object	
___class___	

11.12 Class Dsa

object
└ **hal.maths.crypt.Dsa**

`dsa hash`

11.12.1 Methods

___init___(*self*, *string*)

`x.__init__(...) initializes x; see help(type(x)) for signature`

Overrides: object.___init___ extit(inherited documentation)

hash(*self*)

```
:return: hash plaintext
```

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

11.12.2 Properties

Name	Description
Inherited from object	
___class___	

12 Module hal.maths.maths

A few elegant and powerful mathematical functions.

12.1 Functions

get_prime(*bits*)

```
:param bits: size of number to generate (bits)
:return: prime number of given size
```

blumblumshub(*seed*, *amount*, *prime0*, *prime1*)

```
:param seed: seeder
:param amount: amount of number to generate
:param prime0: one prime number
:param prime1: the second prime number
:return: pseudo-number generator
```

12.2 Variables

Name	Description
__package__	**Value:** `'hal.maths'`

12.3 Class Integer

object
└ **hal.maths.maths.Integer**

12.3.1 Methods

__init__(*self*, *string*)

`x.__init__(...) initializes x; see help(type(x)) for signature`

Overrides: object.__init__ extit(inherited documentation)

is_probably_prime(*self*)

```
:return: test with miller-rabin
```

test_miller_rabin(*self*, *precision*)

```
:param precision: number of rounds to perform (higher -> better precision)
:return: True iff probably prime
```

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

12.3.2 Properties

Name	Description
Inherited from object	
___class___	

12.3.3 Class Variables

Name	Description
LOW_PRIMES	**Value:** `[2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, ...`

12.4 Class EightQueen

object
└ **hal.maths.maths.EightQueen**

```
8 queen problem solver
```

12.4.1 Methods

___init___(*self*, *board_size*)

`x.__init__(...) initializes x; see help(type(x)) for signature`

Overrides: object.___init___ extit(inherited documentation)

under_attack(*self*, *col*, *queens*)

solve(*self*, *n*)

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

12.4.2 Properties

Name	Description
Inherited from object	
___class___	

13 Module hal.maths.plotter

Show elegant plots in any dimension.

13.1 Class Plot2d

object
└ **hal.maths.plotter.Plot2d**

2d plot

13.1.1 Methods

scatter(*vectorx*, *vectory*)

```
:param vectorx: vector in x axis
:param vectory: vector in y axis
:return: 2d scatter plot
```

param(*self*, *functionx*, *functiony*, *min*, *max*, *points*)

```
:param functionx: function in x value
:param functiony: function in y value
::param min: minimum value
:param max: maximum value
:param points: number of points to display
:return: 2d parametric graph of given function from min to max
```

plot(*self*, *function*, *min*, *max*, *points*)

```
:param function: function to plot
:param min: minimum value
:param max: maximum value
:param points: number of points
:return: plot 2d function
```

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___init___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

13.1.2 Properties

Name	Description
Inherited from object	
___class___	

13.2 Class Plot3d

object
└
hal.maths.plotter.Plot3d

13.2.1 Methods

scatter(*vectorx*, *vectory*, *vectorz*)

```
:param vectorx: vector in x axis
:param vectory: vector in y axis
:param vectorz: vector in z axis
:return: plot 3d scattered points
```

param(*self*, *functionx*, *functiony*, *functionz*, *min*, *max*, *points*)

```
:param functionx: function in x
:param functiony: function in y
:param functionz: function in z
:param min: minimum
:param max: maximum
:param points: number of points
:return: 3d parametric graph of given function from min to max
```

plot(*self*, *function*, *minx*, *maxx*, *pointsx*, *miny*, *maxy*, *pointsy*)

```
:param function: function to plot
:param minx: minimum of x-values
:param maxx: maximum of x-values
:param pointsx: points in x axis
:param miny: minimum of y-values
:param maxy: maximum of y-values
:param pointsy: points in y axis
:return: plot 3d function
```

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___init___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

13.2.2 Properties

Name	Description
Inherited from object	
___class___	

13.3 Class Plot4d

object
└ **hal.maths.plotter.Plot4d**

13.3.1 Methods

scatter(*vectorx*, *vectory*, *vectorz*, *vectorw*)

```
:param vectorx: vector in x axis
:param vectory: vector in y axis
:param vectorz: vector in z axis
:param vectorw: vector in w axis
:return: plot 4d scattered points
```

param(*self*, *functionx*, *functiony*, *functionz*, *functionw*, *min*, *max*, *points*)

```
:param functionx: function in x
:param functiony: function in y
:param functionz: function in z
:param functionw: function in w
:param min: minimum
:param max: maximum
:param points: number of points
:return: 4d parametric graph of given function from min to max
```

plot(*self*, *function*, *minx*, *maxx*, *miny*, *maxy*, *minz*, *maxz*, *precision*, *kind*)

```
:param function: function to plot
:param minx: minimum of x-values
:param maxx: maximum of x-values
:param miny: minimum of y-values
:param maxy: maximum of y-values
:param minz: minimum of z-values
:param maxz: maximum of z-values
:param precision: precision
:param kind: slice: x cont -> 3d plot with y,z variables in plane and w as "z"-axis
             contour: x cont -> 3d plot with y,z variables in plane and w colored
:return: plot 4d function
```

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___init___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

13.3.2 Properties

Name	Description
Inherited from object	
___class___	

14 Package hal.ml

14.1 Modules

- **data** *(Section 15, p. 49)*
  - **parser**: Parsers for raw databases.
    (Section 16, p. 50)
- **features**: Collection of methods to find weights of features and select the best ones.
  (Section 17, p. 52)
- **models** *(Section 18, p. 53)*
  - **classification**: Prediction methods based on classification algorithms.
    (Section 19, p. 54)
  - **pipelined**: Prediction methods based on multiple models mixed up.
    (Section 20, p. 55)
  - **regression**: Prediction methods based on regression algorithms.
    (Section 21, p. 56)
  - **time_series**: Multi-purpose prediction methods to be used in time-series.
    (Section 22, p. 57)
- **predict**: " General model to make prediction about everything.
  (Section 23, p. 58)
- **utils**: Various tools and utilities to deal with database and machine learning.
  (Section 24, p. 59)

14.2 Variables

Name	Description
___package___	**Value:** `None`

15 Package hal.ml.data

15.1 Modules

- **parser**: Parsers for raw databases.
  (Section 16, p. 50)

15.2 Variables

Name	Description
___package___	**Value:** `None`

16 Module hal.ml.data.parser

`Parsers for raw databases.`

16.1 Variables

Name	Description
___package___	**Value:** `None`

16.2 Class Parser

object
└ **hal.ml.data.parser.Parser**

Known Subclasses: hal.ml.data.parser.CSVParser

16.2.1 Methods

___init___(*self*, *database_file*)

`:param database_file: a raw .csv file that contains any data about anything`

Overrides: object.___init___

get_lines(*self*)

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

16.2.2 Properties

Name	Description
Inherited from object	
___class___	

16.3 Class CSVParser

object
hal.ml.data.parser.Parser
hal.ml.data.parser.CSVParser

16.3.1 Methods

___init___(*self*, *database_file*)

```
:param database_file: a raw .csv file that contains any data about anything
```

Overrides: object.___init___

parse_data(*self*)

```
store values in array, store lines in array; the result is a 2D matrix
```

Inherited from hal.ml.data.parser.Parser(Section 16.2)

get_lines()

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

16.3.2 Properties

Name	Description
Inherited from object	
___class___	

17 Module hal.ml.features

`Collection of methods to find weights of features and select the best ones.`

17.1 Functions

select_k_best(x, y, k)

`select k best features in dataset`

get_best_features(x, y)

`finds the optimal number of features`

get_features(x, y, $n\_features\_to\_select$)

`finds the optimal features`

17.2 Variables

Name	Description
___package___	**Value:** `'hal.ml'`

18 Package hal.ml.models

18.1 Modules

- **classification**: Prediction methods based on classification algorithms.
  (Section 19, p. 54)
- **pipelined**: Prediction methods based on multiple models mixed up.
  (Section 20, p. 55)
- **regression**: Prediction methods based on regression algorithms.
  (Section 21, p. 56)
- **time_series**: Multi-purpose prediction methods to be used in time-series.
  (Section 22, p. 57)

18.2 Variables

Name	Description
__package__	**Value:** `None`

19 Module hal.ml.models.classification

`Prediction methods based on classification algorithms.`

19.1 Functions

extra_trees_classifier()

random_forest()

knn()

`very fast and slightly more accurate than AdaBoost`

ada_boost()

`fast, accurate but too uncertainty`

bayes_gauss()

`slower than svr but equally accuarte`

bayes_bernoulli()

19.2 Variables

Name	Description
___package___	**Value:** `'hal.ml.models'`

20 Module hal.ml.models.pipelined

`Prediction methods based on multiple models mixed up.`

20.1 Functions

logistic_rbm()

anova_svm()

20.2 Variables

Name	Description
___package___	**Value:** `'hal.ml.models'`

21 Module hal.ml.models.regression

`Prediction methods based on regression algorithms.`

21.1 Functions

support_vector_machine()

`super fast and precise`

logistic_regression()

21.2 Variables

Name	Description
___package___	**Value:** `'hal.ml.models'`

22 Module hal.ml.models.time_series

Multi-purpose prediction methods to be used in time-series.

22.1 Functions

test_stationarity(*timeseries*)

arma(*dates*, *values*, *start*=None, *end*=None, *plot*=False)

```
Predict days values using ARMA algorithm.
:param dates: list of str date
:param values: list of float values
:param start: start predicting in this day
:param end: end of prediction
:param plot: whether to plot or not values in graph
```

arima(*dates*, *values*, *start*=None, *end*=None)

```
Predict days values using ARIMA algorithm.
:param dates: list of str date
:param values: list of float values
:param start: start predicting in this day
:param end: end of prediction
```

var(*dates*, *values*, *start*=None, *end*=None)

```
Predict days values using ARIMA algorithm.
:param dates: list of str date
:param values: list of float values
:param start: start predicting in this day
:param end: end of prediction
```

dynamic_var(*dates*, *values*, *start*=None, *end*=None)

```
Predict days values using ARIMA algorithm.
:param dates: list of str date
:param values: list of float values
:param start: start predicting in this day
:param end: end of prediction
```

23 Module hal.ml.predict

```
" General model to make prediction about everything.
```

23.1 Variables

Name	Description
___package___	**Value:** `'hal.ml'`

23.2 Class BasePrediction

object
└ **hal.ml.predict.BasePrediction**

23.2.1 Methods

___init___(*self*, *model*, *rounds*)

`x.__init__(...) initializes x; see help(type(x)) for signature`

Overrides: object.___init___ extit(inherited documentation)

train(*self*, *x*, *y*)

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

23.2.2 Properties

Name	Description
Inherited from object	
___class___	

24 Module hal.ml.utils

Various tools and utilities to deal with database and machine learning.

24.1 Functions

precision(*matrix*)

```
Calcualtes accuarcy on database

:param matrix: 2x2 matrix that looks like
True Positive  - False Negative
     |         -       |
False Positive - True Negative
```

recall(*matrix*)

```
Calcualtes recall on database

:param matrix: 2x2 matrix that looks like
True Positive  - False Negative
     |         -       |
False Positive - True Negative
```

tn_rate(*matrix*)

```
Calcualtes true negative rate on database

:param matrix: 2x2 matrix that looks like
True Positive  - False Negative
     |         -       |
False Positive - True Negative
```

accuracy(*matrix*)

```
Calcualtes recall on database

:param matrix: 2x2 matrix that looks like
True Positive  - False Negative
     |         -       |
False Positive - True Negative
```

f1_score(*matrix*)

```
Calcualtes f1 score on database

:param matrix: 2x2 matrix that looks like
True Positive  - False Negative
     |         -       |
False Positive - True Negative
```

pearson(*x*, *y*)

```
Pearson coefficient of arrays
```

show_correlation_matrix(*feature_list*, *correlation_matrix*)

```
Show the given correlation matrix as image
```

25 Package hal.profile

25.1 Modules

- **performance**: Perform benchmarks and tests on your PC.
(Section 26, p. 62)

25.2 Variables

Name	Description
___package___	**Value:** `None`

26 Module hal.profile.performance

`Perform benchmarks and tests on your PC.`

26.1 Variables

Name	Description
___package___	**Value:** `'hal.profile'`

26.2 Class EightQueenTest

object
└ **hal.profile.performance.EightQueenTest**

`test CPU by solving eight-queen problem`

26.2.1 Methods

___init___(*self*, *size*)

`x.__init__(...) initializes x; see help(type(x)) for signature`

Overrides: object.___init___ extit(inherited documentation)

welcome()

`:return: introduce script`

introduction()

`:return: introduce 8 queen problem`

run(*self*)

Inherited from object

___delattr___(), ___format___(), ___getattribute___(), ___hash___(), ___new___(), ___reduce___(), ___reduce_ex___(), ___repr___(), ___setattr___(), ___sizeof___(), ___str___(), ___subclasshook___()

26.2.2 Properties

Name	Description
Inherited from object	
___class___	

27 Package hal.wrappers

27.1 Modules

- **methods**: Typical (and useful) function wrappers
(Section 28, p. 65)

27.2 Variables

Name	Description
___package___	**Value:** `None`

28 Module hal.wrappers.methods

Typical (and useful) function wrappers

28.1 Functions

handle_exceptions(*function*)

```
:param function: callback function
    function to wrap
:return: callback function return type
    wraps callback function
```

Index