
Python

Dec 18, 2021

Contents:

1	PyOngc	1
1.1	Description	1
1.2	Usage	1
1.3	License	2
2	The ongc python module	3
2.1	Classes provided by ongc module	3
2.2	Functions provided by ongc module	3
2.2.1	Public Functions	3
2.2.2	Private Functions	3
2.3	Custom exceptions for PyOngc	3
3	The CLI interface	5
3.1	Synopsis	5
3.2	Description	5
3.3	Commands	5
3.3.1	Nearby	5
3.3.2	Neighbors	6
3.3.3	Search	6
3.3.4	Separation	7
3.3.5	Stats	7
3.3.6	View	7
3.4	Help	8
	Python Module Index	9
	Index	11

A python interface for accessing OpenNGC database data

1.1 Description

PyOngc provides a python module to access astronomical data about NGC and IC objects.

The data is based on OpenNGC database <https://github.com/mattiaverga/OpenNGC>.

It is composed by a python module and a command line interface named (guess what) ongc, which can be used to quickly see object details or build object lists based on several parameters.

PyOngc can pass data to PyEphem in a simple way to get ephemerides of NGC/IC objects: see the documentation about ongc.xephemFormat method.

1.2 Usage

```
>>> import pyongc
>>> DSOobject = pyongc.get("NGC7000")
>>> DSOobject.coords
array([[20.  , 59.  , 17.14],
       [44.  , 31.  , 43.6 ]])
```

Object data is easily available from command line also:

```
$ ongc view NGC7000 --details
+-----+
| Id: 13055      Name: NGC7000      Type: HII Ionized region      |
| R.A.: 20:59:17.14      Dec.: +44:31:43.6      Constellation: Cyg      |
| Common names:      |
|   North America Nebula      |
+-----+
| Major axis: 120.0'      Minor axis: 30.0'      Position angle: N/A      |
| B-mag: 4.0      V-mag: N/A      J-mag: N/A      H-mag: N/A      K-mag: N/A      |
|      |
+-----+
| Other identifiers:      |
|   C 020, LBN 373      |
+-----+
| OpenNGC notes:      |
|   B-Mag taken from LEDA      |
+-----+
```

The full documentation is available at <https://pyongc.readthedocs.io/en/latest/>.

1.3 License

PyOngc is licensed under MIT.

2.1 Classes provided by ongc module

2.2 Functions provided by ongc module

2.2.1 Public Functions

2.2.2 Private Functions

These functions are usually only useful internally.

2.3 Custom exceptions for PyOngc

Exceptions for PyONGC.

exception InvalidCoordinates (*text: Optional[str] = None*)

Raised when coordinates are not valid.

Maybe you're passing an object without registered coordinates (typically an *Unknown* object) to some function; or you input coordinates as text in a wrong format: to be recognized the input text must be in the format *HH:MM:SS.ss +/-DD:MM:SS.s*.

exception ObjectNotFound (*name: Optional[str] = None*)

Raised when a valid object identifier isn't found in the database.

The identifier is recognized to be part of one of the supported catalogs, but the object isn't in the database (or doesn't exist at all).

For example, *pyongc.Dso('NGC7000A')* is valid, but it doesn't exist.

exception UnknownIdentifier (*text: Optional[str] = None*)

Raised when input text can't be recognized as a valid object identifier.

You're asking for an identifier using the wrong format, or using an identifier which refers to a catalog not supported by PyOngc.

3.1 Synopsis

`ongc` COMMAND [OPTIONS] [ARGS]...

3.2 Description

`ongc` is the command line interface to used to browse ONGC database data. It can be used to query data about an object or to build up lists based on some parameters.

3.3 Commands

3.3.1 Nearby

The `nearby` command allows to list objects in proximity of given J2000 coordinates. If the output exceed 20 objects, the user can choose to view the list in a pager.

The coordinates must be expressed in the form *HH:MM:SS(.SS) +/-DD:MM:SS(.S)*

`ongc nearby [options] RA DEC`

RA

Right Ascension in the form *HH:MM:SS(.SS)*

DEC

Declination in the form *+/-DD:MM:SS(.S)*

`--radius INTEGER`

It's the radius of the search, expressed in arcmin. If it's not provided, the search is made with a default value of 60'.

```
--catalog [all|NGC|IC]
```

Allows to list only objects from NGC or IC catalog. By default the search will output all objects.

3.3.2 Neighbors

The `neighbors` command allows to list objects in proximity of another object. If the output exceeds 20 objects, the user can choose to view the list in a pager.

```
ongc neighbors [options] NAME
```

NAME

The identifier of the object. It can be the main identifier (NGC or IC id) or one of the alternative identifier.

```
--radius INTEGER
```

It's the radius of the search, expressed in arcmin. If it's not provided, the search is made with a default value of 30'.

```
--catalog [all|NGC|IC]
```

Allows to list only objects from NGC or IC catalog. By default the search will output all objects.

3.3.3 Search

The `search` command allows to build up a list of objects with specific parameters. The different options can be combined to perform a very granular search.

If the output exceeds 20 objects, the user can choose to view the list in a pager.

```
ongc search [options]
```

```
--catalog [all|NGC|IC]
```

Allows to list only objects from NGC or IC catalog. By default the search will output all objects.

```
--type TEXT
```

Allows to list only objects of specific kinds. It accepts multiple comma separated values of the object types listed in the [OpenNGC guide](#).

```
--constellation TEXT
```

Allows to list only objects within some constellations boundaries. It accepts multiple comma separated values. Use IAU 3-letter form.

```
--minsize FLOAT
```

Allows to list only objects with a major axis greater or equal than the specified value expressed in arcmin.

```
--maxsize FLOAT
```

Allows to list only objects with a major axis lower than the specified value expressed in arcmin. It will also include all objects with an unknown major axis value.

```
--uptobmag FLOAT
```

Allows to list only objects with B-Mag brighter than the specified value.

```
--uptovmag FLOAT
```

Allows to list only objects with V-Mag brighter than the specified value.

```
--minra TEXT
```

Allows to list only objects with a Right Ascension greater than the specified value. It accepts an input in the form *HH:MM:SS(.SS)*

```
--maxra TEXT
```

Allows to list only objects with a Right Ascension lower than the specified value. It accepts an input in the form *HH:MM:SS(.SS)*

```
--mindec TEXT
```

Allows to list only objects with a Declination greater than the specified value. It accepts an input in the form *+/-DD:MM:SS(.S)*

```
--maxdec TEXT
```

Allows to list only objects with a Declination lower than the specified value. It accepts an input in the form *+/-DD:MM:SS(.S)*

```
-n, --named TEXT
```

Allows to search an object by its common name (or part of it).

```
-N, --withname
```

Allows to list only objects with a common name.

```
-O, --out_file FILENAME
```

Outputs the results to a file rather than to the terminal.

3.3.4 Separation

The `separation` command returns the apparent angular separation between two objects.

```
ongc separation OBJ1 OBJ2
```

OBJ1

The identifier of the object. It can be the main identifier (NGC or IC id) or one of the alternative identifier.

OBJ2

The identifier of the object. It can be the main identifier (NGC or IC id) or one of the alternative identifier.

3.3.5 Stats

The `stats` command shows some information about the database in use.

```
ongc stats
```

3.3.6 View

The `view` command allows to gather information about a specific object.

```
ongc view [options] NAME
```

Without options, prints a brief description of the object composed by the main identifier used in ONGC database, the object type and the constellation where the object is located.

NAME

The identifier of the object. It can be the main identifier (NGC or IC id) or one of the alternative identifier.

-D, --details

Prints all the available information about the object.

The output is rendered in a table suited to be viewed in a 80cols terminal.

3.4 Help

If you find bugs in ongc (or in the man page), please feel free to file a bug report or a pull request:

<https://github.com/mattiaverga/PyOngc>

e

`exceptions`, 3

E

exceptions (*module*), 3

I

InvalidCoordinates, 3

O

ObjectNotFound, 3

U

UnknownIdentifier, 3