IDL Library for Atomic Data of Ionized Nebulae

API Documentation for AtomNeb-idl

Contents

I Overview **5**

II API **9**

Directory: ./ 11

Overview 11
atomneb_get_aeff_collection_reference_citation.pro 11
atomneb_get_aeff_he_i_pfsd12_reference_citation.pro . . . 12
atomneb_get_aeff_n_ii_fsl13_reference_citation.pro 13
atomneb_get_aeff_o_ii_ssb17_reference_citation.pro 15
atomneb_get_aeff_ppb91_reference_citation.pro 16
atomneb_get_aeff_sh95_reference_citation.pro 17
atomneb_get_aij_reference_citation.pro 18
atomneb_get_elj_reference_citation.pro 20
atomneb_get_omij_reference_citation.pro 21
atomneb_list_aeff_collection_references.pro 22
atomneb_list_aeff_he_i_pfsd12_references.pro 23
atomneb_list_aeff_n_ii_fsl13_references.pro 24
atomneb_list_aeff_o_ii_ssb17_references.pro 25
atomneb_list_aeff_ppb91_references.pro 26
atomneb_list_aeff_sh95_references.pro 27
atomneb_list_aij_references.pro 29
atomneb_list_omij_references.pro 30
atomneb_read_aeff_collection.pro 31
atomneb_read_aeff_collection_list.pro 32
atomneb_read_aeff_collection_references.pro 33
atomneb_read_aeff_he_i_pfsd12.pro 34
atomneb_read_aeff_he_i_pfsd12_list.pro 35
atomneb_read_aeff_he_i_pfsd12_references.pro 36
atomneb_read_aeff_n_ii_fsl13.pro 37
atomneb_read_aeff_n_ii_fsl13_list.pro 38

atomneb_read_aeff_n_ii_fsl13_references.pro 39
atomneb_read_aeff_o_ii_ssb17.pro 40
atomneb_read_aeff_o_ii_ssb17_list.pro 41
atomneb_read_aeff_o_ii_ssb17_references.pro 42
atomneb_read_aeff_ppb91.pro 43
atomneb_read_aeff_ppb91_list.pro 44
atomneb_read_aeff_ppb91_references.pro 45
atomneb_read_aeff_sh95.pro 46
atomneb_read_aeff_sh95_list.pro 47
atomneb_read_aeff_sh95_references.pro 48
atomneb_read_aij.pro 49
atomneb_read_aij_list.pro 50
atomneb_read_aij_references.pro 51
atomneb_read_elj.pro 51
atomneb_read_elj_list.pro 53
atomneb_read_elj_references.pro 53
atomneb_read_omij.pro 54
atomneb_read_omij_list.pro 55
atomneb_read_omij_references.pro 56
atomneb_search_aeff_collection.pro 57
atomneb_search_aeff_he_i_pfsd12.pro 58
atomneb_search_aeff_n_ii_fsl13.pro 59
atomneb_search_aeff_o_ii_ssb17.pro 61
atomneb_search_aeff_ppb91.pro 62
atomneb_search_aeff_sh95.pro 63
atomneb_search_aij.pro 64
atomneb_search_omij.pro 65

Part I

Overview

Overview

AtomNeb-idl is an IDL library for reading atomic data from the `AtomNeb` database for *collisionally excited lines* and *recombination lines* commonly observed in ionized nebulae.

* `AtomNeb database for collisionally excited lines` contains energy levels (Ej), collision strengths (Îľij), and transition probabilities (Aij) of the most ions commonly observed in ionized nebulae:

- `Collection dataset` was compiled according to the atomic data used in pyNeb v1.0 and includes some improved atomic data from Cloudy v13.04 and the National Institute of Standards and Technology (NIST) Atomic Spectra Database, the CHIANTI atomic database.

- `Chianti52, Chianti60, and Chianti70 datasets` were compiled according to the atomic data used in the FORTRAN program MOCASSIN.

* `AtomNeb database for recombination lines` contains effective recombination coefficient (Îśeff), and Branching ratios (Br) of the most ions commonly observed in ionized nebulae.

- `RC Collection dataset` was compiled according to the atomic data used in the FORTRAN program MOCASSIN, and includes C II (Davey et al. 2000), N II (Escalante and Victor 1990), O II (Storey 1994; Liu et al. 1995), and Ne II ions (Kisielius et al. 1998).

- `SH95 Collection dataset` was compiled using hydrogenic ions for Z=1 to 8, namely H I, He II, Li III, Be IV, B V, C VI, N VII, and O VIII ions from Storey and Hummer (1995).

- `PPB91 Collection dataset` was compiled using the atomic data for H, He, C, N, O, Ne ions from Pequignot, Petitjean and Boisson (1991).

- `PFSD12 He I dataset` was compiled using effective He I recombination coefficients from Porter et al (2012) and (2013).

- `FSL13 N II dataset` was compiled using effective N II recombination coefficients from Fang, Storey and Liu (2011) and (2013).

- `SSB17 O II dataset` was compiled using effective O II recombination coefficients from Storey, Sochi and Bastin (2017).

Dependencies

* This package requires the following packages:

- The IDL Astronomy User's Library

* To get this package with all the dependent packages, you can simply use git command as follows:

```
git clone --recursive https://github.com/atomneb/AtomNeb-idl.git
```

GDL Installation

* The GNU Data Language (GDL) can be installed on

- Linux (Fedora):

```
sudo dnf install gdl
```

- Linux (Ubuntu):

```
sudo apt-get install gnudatalanguage
```

- OS X:

```
brew install gnudatalanguage
```

- Windows: using the GNU Data Language for Win32 (Unofficial Version) or compiling the GitHub source with Visual Studio 2015 as seen in appveyor.yml.

* To setup AtomNeb-idl in GDL, add its path to .gdl_startup in the home directory:

```
!PATH=!PATH + ':/home/AtomNeb-idl/pro/'
!PATH=!PATH + ':/home/AtomNeb-idl/externals/astron/pro/'
```

```
Set ``GDL_STARTUP`` in ``.bashrc`` (bash):

export GDL_STARTUP=~/.gdl_startup
```

or in .tcshrc (cshrc):

```
setenv GDL_STARTUP ~/.gdl_startup
```

* This package needs GDL version 0.9.8 or later.

IDL Installation

* To install AtomNeb-idl in IDL, add its path to your IDL path. For more information about the path management in IDL, read the IDL path management by Harris Geospatial or the IDL library installation by David Fanning.

* This package needs IDL version 7.1 or later.

Project statistics

Directories:	1
.pro files:	52
.sav files:	0
Routines:	52
Lines:	1,078

Part II

API

Directory: ./

Overview

atomneb_get_aeff_collection_reference_citation.pro

ATOMNEB_GET_AEFF_COLLECTION_REFERENCE_CITATION

This function returns the reference citation for a recombination coefficient (Aeff) from the 2nd binary table extension of the FITS data file ('rc_collection.fits').

```
result = atomneb_get_aeff_collection_reference_citation(Atom_RC_file, atom, ion, /br,
   reference=string)
```

Returns

type=string. This function returns the Citation.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_collection.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'c'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Keywords

br IN TYPE=boolean
set for the branching ratios (Br), may not necessary

reference IN TYPE=string
set for the reference, not necessary

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_rc_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_collection.fits', root_dir=base_dir, subdir=data_rc_dir )
 IDL> atom='c'
 IDL> ion='iii' ; C III
 IDL> citation=atomneb_get_aeff_collection_reference_citation(Atom_RC_file, atom, ion)
 IDL> print, citation
    Davey, A. R., Storey, P. J. and Kisielius, R., Astron.Astrophys.Suppl., 142, 85, 2000
```

Author

Ashkbiz Danehkar

Copyright

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_get_aeff_he_i_pfsd12_reference_citation.pro

ATOMNEB_GET_AEFF_HE_I_PFSD12_REFERENCE_CITATION

This function returns the reference citation for a recombination coefficient (Aeff) from the 2nd binary table extension of the FITS data file ('rc_he_ii_PFSD12.fits').

```
result = atomneb_get_aeff_he_i_pfsd12_reference_citation(Atom_RC_file, atom, ion, reference
   =string)
```

Returns

type=string. This function returns the Citation.

Parameters

Atom_RC_file IN REQUIRED TYPE=string

the FITS data file name ('rc_he_ii_PFSD12.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'he'

ion IN REQUIRED TYPE=string
ionic level e.g 'ii'

Keywords

reference IN TYPE=string
set for the reference e.g. 'PFSD13', may not necessary

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_he_ii_PFSD12.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='he'
 IDL> ion='ii' ; He I
 IDL> reference='PFSD13'
 IDL> citation=atomneb_get_aeff_he_i_pfsd12_reference_citation(Atom_RC_file, atom, ion, reference=refere
 IDL> print, citation
    Porter, R. L., Ferland, G. J., Storey, P. J. and Detisch, M. J., MNRAS, 433L, 89, 2013
```

Author

Ashkbiz Danehkar

Copyright

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_get_aeff_n_ii_fsl13_reference_citation.pro

ATOMNEB_GET_AEFF_N_II_FSL13_REFERENCE_CITATION

This function returns the reference citation for a recombination coefficient (Aeff) from the 2nd binary table extension of the FITS data file ('rc_n_iii_FSL13.fits').

```
result = atomneb_get_aeff_n_ii_fsl13_reference_citation(Atom_RC_file, atom, ion, reference
   =string)
```

Returns

type=string. This function returns the Citation.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_n_iii_FSL13.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'n'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Keywords

reference IN TYPE=string
set for the reference e.g. 'FSL13', may not necessary

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_n_iii_FSL13.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='n'
 IDL> ion='iii' ; N II
 IDL> reference='FSL13'
 IDL> citation=atomneb_get_aeff_n_ii_fsl13_reference_citation(Atom_RC_file, atom, ion)
 IDL> print, citation
    Fang X., Storey P.J., and Liu X.-W., R. 2011, Astron.Astrophys. 530, A18; 2013, Astron.Astrophys. 55
```

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

03/07/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_get_aeff_o_ii_ssb17_reference_citation.pro

ATOMNEB_GET_AEFF_O_II_SSB17_REFERENCE_CITATION

This function returns the reference citation for a recombination coefficient (Aeff) from the 2nd binary table extension of the FITS data file ('rc_o_iii_SSB17.fits').

```
result = atomneb_get_aeff_o_ii_ssb17_reference_citation(Atom_RC_file, atom, ion, reference
   =string)
```

Returns

type=string. This function returns the Citation.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_o_iii_SSB17.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'o'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Keywords

reference IN TYPE=string
set for the reference e.g. 'SSB17', may not necessary

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_o_iii_SSB17.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='o'
 IDL> ion='iii' ; O II
 IDL> reference='SSB17'
 IDL> citation=atomneb_get_aeff_o_ii_ssb17_reference_citation(Atom_RC_file, atom, ion)
 IDL> print, citation
    Storey, P.J., Sochi, T. and Bastin, R. 2017, MNRAS, 470, 379; VizieR On-line Data Catalog: VI/150
```

Author

Ashkbiz Danehkar

Copyright

History

03/07/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_get_aeff_ppb91_reference_citation.pro

ATOMNEB_GET_AEFF_PPB91_REFERENCE_CITATION

This function returns the reference citation for a recombination coefficient (Aeff) from the 2nd binary table extension of the FITS data file ('rc_PPB91.fits').

```
result = atomneb_get_aeff_ppb91_reference_citation(Atom_RC_file, atom, ion, reference=
   string)
```

Returns

type=string. This function returns the Citation.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_PPB91.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'c'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Keywords

reference IN TYPE=string
set for the reference, not necessary

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_PPB91.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='c'
 IDL> ion='iii'
```

```
IDL> citation=atomneb_get_aeff_ppb91_reference_citation(Atom_RC_file, atom, ion)
IDL> print, citation
   Pequignot, D., Petitjean, P. and Boisson, C. Astron.Astrophys., 251, 680, 1991
```

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_get_aeff_sh95_reference_citation.pro

ATOMNEB_GET_AEFF_SH95_REFERENCE_CITATION

This function returns the reference citation for a recombination coefficient (Aeff) from the 2nd binary table extension of the FITS data file ('rc_SH95.fits').

```
result = atomneb_get_aeff_sh95_reference_citation(Atom_RC_file, atom, ion, reference=
   string, case1=string)
```

Returns

type=string. This function returns the Citation.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_SH95.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'h'

ion IN REQUIRED TYPE=string
ionic level e.g 'ii'

Keywords

reference IN TYPE=string
set for the reference, not necessary

case1 IN TYPE=string
set for the case 'a' or 'b', defualt 'b'

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_SH95.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='h'
 IDL> ion='ii' ; H I
 IDL> citation=atomneb_get_aeff_sh95_reference_citation(Atom_RC_file, atom, ion)
 IDL> print, citation
    Storey, P. J. and Hummer, D. G., MNRAS, 272, 41S, 1995
```

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_get_aij_reference_citation.pro

ATOMNEB_GET_AIJ_REFERENCE_CITATION

This function returns the reference citation for a transition probability (Aij) from the 2nd binary table extension of the FITS data file ('AtoAij.fits')

```
result = atomneb_get_aij_reference_citation(Atom_Aij_file, atom, ion, reference)
```

Returns

type=string. This function returns the Citation.

Parameters

Atom_Aij_file IN REQUIRED TYPE=string
the FITS data file name ('AtoAij.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'o'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

reference IN TYPE=string
set for the reference e.g. 'FFT04'

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data', 'collection']
 IDL> Atom_Aij_file = filepath('AtomAij.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='o'
 IDL> ion='iii'
 IDL> reference='FFT04'
 IDL> citation=atomneb_get_aij_reference_citation(Atom_Aij_file, atom, ion, reference)
 IDL> print,citation
    Froese Fischer et al 2004, ADNDT 87, 1
```

Author

Ashkbiz Danehkar

Copyright

History

24/12/2015, IDL code by A. Danehkar

Version

0.0.1

atomneb_get_elj_reference_citation.pro

ATOMNEB_GET_ELJ_REFERENCE_CITATION

This function returns the reference citation for energy levels (Ej) from the 2nd binary table extension of the FITS data file ('Atom-Elj.fits').

```
result = atomneb_get_elj_reference_citation(Atom_Elj_file, reference)
```

Returns

type=string. This function returns the Citation.

Parameters

Atom_Elj_file IN REQUIRED TYPE=string
the FITS data file name ('AtomElj.fits')

reference IN TYPE=string
set for the reference e.g. 'L7288'

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data', 'collection']
 IDL> Atom_Elj_file = filepath('AtomElj.fits', root_dir=base_dir, subdir=data_dir )
 IDL> reference='L7288'
 IDL> citation=atomneb_get_elj_reference_citation(Atom_Elj_file, reference)
 IDL> print,citation
    C. E. Moore, in CRC Series in Evaluated Data in Atomic Physics, 339 pp. (CRC Press, Boca Raton, FL,
```

Author

Ashkbiz Danehkar

Copyright

History

24/12/2015, IDL code by A. Danehkar

Version

0.0.1

atomneb_get_omij_reference_citation.pro

ATOMNEB_GET_OMIJ_REFERENCE_CITATION

This function returns the reference citation for collision strengths (Omega_ij) from the 2nd binary table extension of the FITS data file ('AtomOmij.fits').

```
result = atomneb_get_omij_reference_citation(Atom_Omij_file, atom, ion, reference)
```

Returns

type=string. This function returns the Citation.

Parameters

Atom_Omij_file IN REQUIRED TYPE=string
: the FITS data file name ('AtomOmij.fits')

atom IN REQUIRED TYPE=string
: atom name e.g. 'o'

ion IN REQUIRED TYPE=string
: ionic level e.g 'iii'

reference IN TYPE=string
: set for the reference e.g. 'SSB14'

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data', 'collection']
 IDL> Atom_Omij_file = filepath('AtomOmij.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='o'
 IDL> ion='iii'
 IDL> reference='SSB14'
 IDL> citation=atomneb_get_omij_reference_citation(Atom_Omij_file, atom, ion, reference)
 IDL> print,citation
    Storey, P. J., Sochi, T., and Badnell, N. R. 2014, Astron.Astrophys., 441, 3028
```

Author

Ashkbiz Danehkar

Copyright

History

24/12/2015, IDL code by A. Danehkar

Version

0.0.1

atomneb_list_aeff_collection_references.pro

ATOMNEB_LIST_AEFF_COLLECTION_REFERENCES

This function returns a list for all references of recombination coefficients (Aeff) for given element and ionic level from the FITS data file ('rc_collection.fits').

```
result = atomneb_list_aeff_collection_references(Atom_RC_file, atom, ion, /br)
```

Returns

type=an array of strings. This function returns the references.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_collection.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'c'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Keywords

br IN TYPE=boolean
set for the branching ratios (Br), may not necessary

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_rc_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_collection.fits', root_dir=base_dir, subdir=data_rc_dir )
 IDL> atom='c'
 IDL> ion='iii' ; C III
 IDL> list_cii_aeff_reference=atomneb_list_aeff_collection_references(Atom_RC_file, atom, ion)
 IDL> print,list_cii_aeff_reference
```

Author

Ashkbiz Danehkar

Copyright

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_list_aeff_he_i_pfsd12_references.pro

ATOMNEB_LIST_AEFF_HE_I_PFSD12_REFERENCES

This function returns a list for all references of recombination coefficients (Aeff) for given element and ionic level from the FITS data file ('rc_he_ii_PFSD12.fits').

```
result = atomneb_list_aeff_he_i_pfsd12_references(Atom_RC_file, atom, ion)
```

Returns

type=an array of strings. This function returns the references.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_he_ii_PFSD12.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'he'

ion IN REQUIRED TYPE=string
ionic level e.g 'ii'

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_he_ii_PFSD12.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='he'
```

```
IDL> ion='ii' ; He I
IDL> list_hei_aeff_reference=atomneb_list_aeff_he_i_pfsd12_references(Atom_RC_file, atom, ion)
IDL> print,list_hei_aeff_reference
   PFSD12 PFSD13
```

Author

Ashkbiz Danehkar

Copyright

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_list_aeff_n_ii_fsl13_references.pro

ATOMNEB_LIST_AEFF_N_II_FSL13_REFERENCES

This function returns a list for all references of recombination coefficients (Aeff) for given element and ionic level from the FITS data file ('rc_n_iii_FSL13.fits').

```
result = atomneb_list_aeff_n_ii_fsl13_references(Atom_RC_file, atom, ion)
```

Returns

type=an array of strings. This function returns the references.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_n_iii_FSL13.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'n'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_n_iii_FSL13.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='n'
 IDL> ion='iii' ; N II
 IDL> list_nii_aeff_reference=atomneb_list_aeff_n_ii_fsl13_references(Atom_RC_file, atom, ion)
 IDL> print,list_nii_aeff_reference
```

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

03/07/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_list_aeff_o_ii_ssb17_references.pro

ATOMNEB_LIST_AEFF_O_II_SSB17_REFERENCES

This function returns a list for all references of recombination coefficients (Aeff) for given element and ionic level from the FITS data file ('rc_o_iii_SSB17.fits').

```
result = atomneb_list_aeff_o_ii_ssb17_references(Atom_RC_file, atom, ion)
```

Returns

type=an array of strings. This function returns the references.

Parameters

Atom_RC_file IN REQUIRED TYPE=string

the FITS data file name ('rc_o_iii_SSB17.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'o'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_o_iii_SSB17.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='o'
 IDL> ion='iii' ; O II
 IDL> list_oii_aeff_reference=atomneb_list_aeff_o_ii_ssb17_references(Atom_RC_file, atom, ion)
 IDL> print,list_oii_aeff_reference
```

Author

Ashkbiz Danehkar

Copyright

History

03/07/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_list_aeff_ppb91_references.pro

ATOMNEB_LIST_AEFF_PPB91_REFERENCES

This function returns a list for all references of recombination coefficients (Aeff) for given element and ionic level from the FITS data file ('rc_PPB91.fits').

```
result = atomneb_list_aeff_ppb91_references(Atom_RC_file, atom, ion)
```

Returns

type=an array of strings. This function returns the references.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_PPB91.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'c'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_PPB91.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='c'
 IDL> ion='iii'
 IDL> list_cii_aeff_reference=atomneb_list_aeff_ppb91_references(Atom_RC_file, atom, ion)
 IDL> print,list_cii_aeff_reference
```

Author

Ashkbiz Danehkar

Copyright

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_list_aeff_sh95_references.pro

ATOMNEB_LIST_AEFF_SH95_REFERENCES

This function returns a list for all references of recombination coefficients (Aeff) for given element and ionic level from the FITS data file ('rc_SH95.fits').

```
result = atomneb_list_aeff_sh95_references(Atom_RC_file, atom, ion)
```

Returns

type=an array of strings. This function returns the references.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_SH95.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'h'

ion IN REQUIRED TYPE=string
ionic level e.g 'ii'

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_SH95.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='h'
 IDL> ion='ii' ; H I
 IDL> list_hi_aeff_reference=atomneb_list_aeff_sh95_references(Atom_RC_file, atom, ion)
 IDL> print,list_hi_aeff_reference
```

Author

Ashkbiz Danehkar

Copyright

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_list_aij_references.pro

ATOMNEB_LIST_AIJ_REFERENCES

This function returns a list for all references of transition probabilities (Aij) for given element and ionic level from the FITS data file ('AtoAij.fits').

```
result = atomneb_list_aij_references(Atom_Aij_file, atom, ion)
```

Returns

type=an array of strings. This function returns the references.

Parameters

Atom_Aij_file

atom IN REQUIRED TYPE=string
atom name e.g. 'o'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data', 'collection']
 IDL> Atom_Aij_file = filepath('AtomAij.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='o'
 IDL> ion='iii'
 IDL> list_oiii_aij_reference=atomneb_list_aij_references(Atom_Aij_file, atom, ion)
 IDL> print,list_oiii_aij_reference
    FFT04-SZ00 FFT04 GMZ97-WFD96 SZ00-WFD96
```

Author

Ashkbiz Danehkar

Copyright

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_list_omij_references.pro

ATOMNEB_LIST_OMIJ_REFERENCES

This function returns a list for all references of collision strengths (Omega_ij) for given element and ionic level from the FITS data file ('AtomOmij.fits').

```
result = atomneb_list_omij_references(Atom_Omij_file, atom, ion)
```

Returns

type=an array of strings. This function returns the references.

Parameters

Atom_Omij_file IN REQUIRED TYPE=string
the FITS data file name ('AtomOmij.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'o'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data', 'collection']
 IDL> Atom_Omij_file = filepath('AtomOmij.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='o'
 IDL> ion='iii'
 IDL> list_oiii_omij_reference=atomneb_list_omij_references(Atom_Omij_file, atom, ion)
 IDL> print,list_oiii_omij_reference
    AK99 LB94 Pal12-AK99 SSB14
```

Author

Ashkbiz Danehkar

Copyright

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_collection.pro

ATOMNEB_READ_AEFF_COLLECTION

This function returns the effective recombination coefficients
(Aeff) from the table extensions of the FITS data file ('rc_collection.fits').

```
result = atomneb_read_aeff_collection(Atom_RC_file, atom, ion, /br, reference=string)
```

Returns

type=an array of data. This function returns the effective recombination coefficients. aeff_data (c_iii_aeff) { Wavelength:0.0, a: 0.0, b: 0.0, c: 0.0, d: 0.0, f: 0.0}

aeff_data (n_iii_aeff) { a: 0.0, b: 0.0, c: 0.0}

aeff_data (n_iii_br) {Wavelength: 0.0, BR: 0.0, $ g1:0, g2:0, Mult1:", LowerTerm:", UpperTerm:" }

aeff_data (o_iii_aeff) {Term: ", Case1: ", a2: 0.0, a4: 0.0, a5: 0.0, a6: 0.0, b: 0.0, c: 0.0, d: 0.0}

aeff_data (o_iii_br) {Wavelength:double(0.0), Br_A: 0.0, Br_B: 0.0, Br_C: 0.0, g1: 0, g2: 0, Mult1: ", LowerTerm: ", UpperTerm: "}

aeff_data (ne_iii_aeff) {Wavelength:0.0, a: 0.0, b: 0.0, c: 0.0, d: 0.0, f: 0.0, br: 0.0}

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_collection.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'c'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Keywords

br IN TYPE=boolean
set for the branching ratios (Br), may not necessary

reference IN TYPE=string
set for the reference, not necessary

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_rc_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_collection.fits', root_dir=base_dir, subdir=data_rc_dir )
 IDL> atom='c'
 IDL> ion='iii' ; C III
 IDL> cii_rc_data=atomneb_read_aeff_collection(Atom_RC_file, atom, ion)
 IDL> temp=size(cii_rc_data.Wavelength,/DIMENSIONS)
 IDL> n_line=temp[0]
 IDL> for i=0,n_line-1 do print,cii_rc_data[i].Wavelength, cii_rc_data[i].a, $
 IDL>                          cii_rc_data[i].b, cii_rc_data[i].c, $
 IDL>                          cii_rc_data[i].d, cii_rc_data[i].f
    914.00000      0.69280000     0.021400000    -0.016300000     -0.24310000     -0.88000000
    962.00000       1.0998000   -0.0042000000    -0.027900000     -0.22940000     -0.96560000
    ...
```

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_collection_list.pro

ATOMNEB_READ_AEFF_COLLECTION_LIST

private

This function returns the list of effective recombination coefficients (Aeff) from the 1st binary table extension of the FITS data file ('rc_collection.fits')

```
result = atomneb_read_aeff_collection_list(Atom_RC_file)
```

Returns

type=an array of data. This function returns the aeff_data_list:
{ Aeff_Data:'', Extention:0.0}

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_collection.fits')

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_collection_references.pro

ATOMNEB_READ_AEFF_COLLECTION_REFERENCES

private

This function returns the reference list of recombination coefficients (Aeff) from the 2nd binary table extension of the FITS data file ('rc_collection.fits').

```
result = atomneb_read_aeff_collection_references(Atom_RC_file)
```

Returns

type=an array of data. This function returns the aeff_data_reference:
{ Reference:'', Citation:''}

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_collection.fits')

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_he_i_pfsd12.pro

ATOMNEB_READ_AEFF_HE_I_PFSD12

This function returns the effective recombination coefficients
(Aeff) from the table extensions of the FITS data file ('rc_he_ii_PFSD12.fits').

```
result = atomneb_read_aeff_he_i_pfsd12(Atom_RC_file, atom, ion, /wavelength, reference=
    string)
```

Returns

type=an array of data. This function returns the effective
recombination coefficients.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_he_ii_PFSD12.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'he'

ion IN REQUIRED TYPE=string
ionic level e.g 'ii'

Keywords

wavelength IN TYPE=boolean
set for returning the wavelengths

reference IN TYPE=string
set for the reference, not necessary

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_rc_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_he_ii_PFSD12.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='he'
 IDL> ion='ii' ; He I
 IDL> hei_rc_data=atomneb_read_aeff_he_i_pfsd12(Atom_RC_file, atom, ion)
 IDL> hei_rc_data_wave=atomneb_read_aeff_he_i_pfsd12(Atom_RC_file, atom, ion, /wavelength)
 IDL> print,hei_rc_data[0].Aeff
    5000.0000       10.000000      -25.379540      -25.058970      -25.948440      ...
 IDL> temp=size(hei_rc_data_wave.Wavelength,/DIMENSIONS)
 IDL> n_line=temp[0]
 IDL> for i=0,n_line-1 do print,hei_rc_data_wave[i].Wavelength, hei_rc_data_wave[i].LowerTerm, hei_rc_da
    2945.00005p^{3}P2s^{3}S
    3188.00004p^{3}P2s^{3}S
    3614.00005p^{1}P2s^{1}S
    ...
```

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_he_i_pfsd12_list.pro

ATOMNEB_READ_AEFF_HE_I_PFSD12_LIST

private

This function returns the list of effective recombination coefficients (Aeff) from the 1st binary table extension of the FITS data file ('rc_he_ii_PFSD12.fits')

```
result = atomneb_read_aeff_he_i_pfsd12_list(Atom_RC_file)
```

Returns

type=an array of data. This function returns the aeff_data_list: { Aeff_Data:'', Extention:0.0}

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_he_ii_PFSD12.fits')

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_he_i_pfsd12_references.pro

ATOMNEB_READ_AEFF_HE_I_PFSD12_REFERENCES

private

This function returns the reference list of recombination coefficients (Aeff) from the 2nd binary table extension of the FITS data file ('rc_he_ii_PFSD12.fits').

```
result = atomneb_read_aeff_he_i_pfsd12_references(Atom_RC_file)
```

Returns

type=an array of data. This function returns the aeff_data_reference: { Reference:'', Citation:''}

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_he_ii_PFSD12.fits')

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_n_ii_fsl13.pro

ATOMNEB_READ_AEFF_N_II_FSL13

This function returns the effective recombination coefficients
(Aeff) from the table extensions of the FITS data file ('rc_n_iii_FSL13.fits').

```
result = atomneb_read_aeff_n_ii_fsl13(Atom_RC_file, atom, ion, wavelength_range, /wavelength
   , reference=string)
```

Returns

type=an array of data. This function returns the effective
recombination coefficients.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_n_iii_FSL13.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'n'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

wavelength_range IN REQUIRED TYPE=array
wavelength range e.g. [4400.0, 7100.0]

Keywords

wavelength IN TYPE=boolean
set for returning the wavelengths

reference IN TYPE=string
set for the reference, not necessary

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_rc_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_n_iii_FSL13.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='n'
 IDL> ion='iii' ; N II
 IDL> wavelength_range=[4400.0, 7100.0]
 IDL> nii_rc_data=atomneb_read_aeff_n_ii_fsl13(Atom_RC_file, atom, ion, wavelength_range)
 IDL> nii_rc_data_wave=atomneb_read_aeff_n_ii_fsl13(Atom_RC_file, atom, ion, wavelength_range, /waveleng
 IDL> print,nii_rc_data[0].Aeff
    255.000      79.5000      47.3000      12.5000      ...
 IDL> temp=size(nii_rc_data_wave.Wavelength,/DIMENSIONS)
 IDL> n_line=temp[0]
 IDL> for i=0,n_line-1 do print,nii_rc_data_wave[i].Wavelength, nii_rc_data_wave[i].Tr, nii_rc_data_wave
    6413.236g - 4f2p6g G[9/2]o4 - 2p4f F[7/2]e3
    6556.326g - 4f2p6g G[9/2]o5 - 2p4f G[7/2]e4
    6456.976g - 4f2p6g G[9/2]o5 - 2p4f F[7/2]e4
    ...
```

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

03/07/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_n_ii_fsl13_list.pro

ATOMNEB_READ_AEFF_N_II_FSL13_LIST

private

This function returns the list of effective recombination coefficients (Aeff) from the 1st binary table extension of the FITS data file ('rc_n_iii_FSL13.fits')

```
result = atomneb_read_aeff_n_ii_fsl13_list(Atom_RC_file)
```

Returns

type=an array of data. This function returns the aeff_data_list:
{Aeff_Data:'', Extention:0, $ IND:long(0), Wavelength: float(0.0),
$ Tr:'', Trans: '', T_X: ''}

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_n_iii_FSL13.fits')

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

03/07/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_n_ii_fsl13_references.pro

ATOMNEB_READ_AEFF_N_II_FSL13_REFERENCES

private

This function returns the reference list of recombination coefficients (Aeff) from the 2nd binary table extension of the FITS data file ('rc_n_iii_FSL13.fits').

```
result = atomneb_read_aeff_n_ii_fsl13_references(Atom_RC_file)
```

Returns

type=an array of data. This function returns the aeff_data_reference:
{ Reference:'', Citation:''}

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_n_iii_FSL13.fits')

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

03/07/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_o_ii_ssb17.pro

ATOMNEB_READ_AEFF_O_II_SSB17

This function returns the effective recombination coefficients
(Aeff) from the table extensions of the FITS data file ('rc_o_iii_SSB17.fits').

```
result = atomneb_read_aeff_o_ii_ssb17(Atom_RC_file, atom, ion, case1, wavelength_range, /
   wavelength, reference=string)
```

Returns

type=an array of data. This function returns the effective
recombination coefficients.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_o_iii_SSB17.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'o'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

case1 IN TYPE=string
set for the case 'a' or 'b', defualt 'b'

wavelength_range IN REQUIRED TYPE=array
wavelength range e.g. [5320.0, 5330.0]

Keywords

wavelength IN TYPE=boolean
set for returning the wavelengths

reference IN TYPE=string
set for the reference, not necessary

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_rc_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_o_iii_SSB17.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='o'
 IDL> ion='iii' ; O II
 IDL> case1='B'
 IDL> wavelength_range=[5320.0, 5330.0]
 IDL> oii_rc_data=atomneb_read_aeff_o_ii_ssb17(Atom_RC_file, atom, ion, case1, wavelength_range)
 IDL> oii_rc_data_wave=atomneb_read_aeff_o_ii_ssb17(Atom_RC_file, atom, ion, case1, wavelength_range, /w
 IDL> print,oii_rc_data[0].Aeff
    1.64100e-30  1.60000e-30  1.56400e-30  1.54100e-30 ...
 IDL> temp=size(oii_rc_data_wave.Wavelength,/DIMENSIONS)
 IDL> n_line=temp[0]
 IDL> for i=0,n_line-1 do print,oii_rc_data_wave[i].Wavelength, oii_rc_data_wave[i].lower_term, oii_rc_d
    5327.172s22p2(1S)3p 2Po
    5325.422s22p2(1S)3p 2Po
    5327.182s22p2(1D)3d 2Ge
    ...
```

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

03/07/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_o_ii_ssb17_list.pro

ATOMNEB_READ_AEFF_O_II_SSB17_LIST

private

This function returns the list of effective recombination coefficients (Aeff) from the 1st binary table extension of the FITS data file ('rc_o_iii_SSB17.fits')

```
result = atomneb_read_aeff_o_ii_ssb17_list(Atom_RC_file)
```

Returns

type=an array of data. This function returns the aeff_data_list: {Aeff_Data:'', Extention:0, $ IND:long(0), Wavelength: float(0.0), $ Case1:'', lower_term: '', upper_term: ''}

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_o_iii_SSB17.fits')

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

03/07/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_o_ii_ssb17_references.pro

ATOMNEB_READ_AEFF_O_II_SSB17_REFERENCES

private

This function returns the reference list of recombination coefficients (Aeff) from the 2nd binary table extension of the FITS data file ('rc_o_iii_SSB17.fits').

```
result = atomneb_read_aeff_o_ii_ssb17_references(Atom_RC_file)
```

Returns

type=an array of data. This function returns the aeff_data_reference: { Reference:'', Citation:''}

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_o_iii_SSB17.fits')

Author

Ashkbiz Danehkar

Copyright

History

03/07/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_ppb91.pro

ATOMNEB_READ_AEFF_PPB91

This function returns the effective recombination coefficients
(Aeff) from the table extensions of the FITS data file ('rc_PPB91.fits').

```
result = atomneb_read_aeff_ppb91(Atom_RC_file, atom, ion, reference=string)
```

Returns

type=an array of data. This function returns the effective
recombination coefficients: { Ion: ' ' Case1:'' Wavelength:0.0,
a: 0.0, b: 0.0, c: 0.0, d: 0.0, br: 0.0, y: 0.0}

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_PPB91.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'c'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Keywords

reference IN TYPE=string
set for the reference, not necessary

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_rc_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_PPB91.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='c'
 IDL> ion='iii' ; C II
 IDL> cii_rc_data=atomneb_read_aeff_ppb91(Atom_RC_file, atom, ion)
 IDL> temp=size(cii_rc_data.Wavelength,/DIMENSIONS)
 IDL> n_line=temp[0]
 IDL> for i=0,n_line-1 do print,cii_rc_data[i].Ion,cii_rc_data[i].Case1, $
 IDL>                          cii_rc_data[i].Wavelength, cii_rc_data[i].a, $
 IDL>                          cii_rc_data[i].b, cii_rc_data[i].c, $
 IDL>                          cii_rc_data[i].d, cii_rc_data[i].br, $
 IDL>                          cii_rc_data[i].Q, cii_rc_data[i].y
    C2+A       9903.4600      0.69700000     -0.78400000      ...
    C2+A       4267.1500       1.0110000     -0.75400000      ...
    ...
```

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_ppb91_list.pro

ATOMNEB_READ_AEFF_PPB91_LIST

private

This function returns the list of effective recombination coefficients (Aeff) from the 1st binary table extension of the FITS data file ('rc_PPB91.fits')

```
result = atomneb_read_aeff_ppb91_list(Atom_RC_file)
```

Returns

type=an array of data. This function returns the aeff_data_list: { Aeff_Data:'', Extention:0.0}

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_PPB91.fits')

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_ppb91_references.pro

ATOMNEB_READ_AEFF_PPB91_REFERENCES

private

This function returns the reference list of recombination coefficients (Aeff) from the 2nd binary table extension of the FITS data file ('rc_PPB91.fits').

```
result = atomneb_read_aeff_ppb91_references(Atom_RC_file)
```

Returns

type=an array of data. This function returns the aeff_data_reference: { Reference:'', Citation:''}

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_PPB91.fits')

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_sh95.pro

ATOMNEB_READ_AEFF_SH95

This function returns the effective recombination coefficients
(Aeff) from the table extensions of the FITS data file ('rc_SH95.fits').

```
result = atomneb_read_aeff_sh95(Atom_RC_file, atom, ion, reference=string, case1=string)
```

Returns

type=an array of data. This function returns the effective
recombination coefficients.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_SH95.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'h'

ion IN REQUIRED TYPE=string
ionic level e.g 'ii'

Keywords

reference IN TYPE=string
set for the reference, not necessary

case1 IN TYPE=string
set for the case 'a' or 'b', defualt 'b'

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_rc_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_SH95.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='h'
 IDL> ion='ii' ; H I
 IDL> hi_rc_data=atomneb_read_aeff_sh95(Atom_RC_file, atom, ion)
 IDL> print,hi_rc_data[0].Aeff
```

```
100.00000       500.00000       0.0000000   4.2140000e-27    1.7560000e-27  ...
...
```

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_sh95_list.pro

ATOMNEB_READ_AEFF_SH95_LIST

private

This function returns the list of effective recombination coefficients (Aeff) from the 1st binary table extension of the FITS data file ('rc_SH95.fits')

```
result = atomneb_read_aeff_sh95_list(Atom_RC_file)
```

Returns

type=an array of data. This function returns the aeff_data_list: { Aeff_Data:'', Extention:0.0}

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_SH95.fits')

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aeff_sh95_references.pro

ATOMNEB_READ_AEFF_SH95_REFERENCES

private

This function returns the reference list of recombination coefficients (Aeff) from the 2nd binary table extension of the FITS data file ('rc_SH95.fits').

```
result = atomneb_read_aeff_sh95_references(Atom_RC_file)
```

Returns

type=an array of data. This function returns the aeff_data_reference: { Reference:'', Citation:''}

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_SH95.fits')

Author

Ashkbiz Danehkar

Copyright

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aij.pro

ATOMNEB_READ_AIJ

This function returns the transition probabilities (Aij) from the table extensions of the FITS data file ('AtomAij.fits').

```
result = atomneb_read_aij(Atom_Aij_file, atom, ion, reference=string, level_num=string)
```

Returns

type=an array of data. This function returns the aij_data: { Aij:dblarr(n_level,n_level) }.

Parameters

Atom_Aij_file

atom IN REQUIRED TYPE=string
atom name e.g. 'o'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Keywords

reference IN TYPE=string
set for the reference, not necessary

level_num IN TYPE=string
set for the maximum level number.

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data', 'collection']
 IDL> Atom_Aij_file = filepath('AtomAij.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='o'
 IDL> ion='iii'
 IDL> reference='FFT04'
 IDL> oiii_aij_data=atomneb_read_aij(Atom_Aij_file, atom, ion, reference=reference)
 IDL> print,oiii_aij_data.Aij[*,*]
    0.0000000   2.5960000e-05   3.0300000e-11   2.3220000e-06       0.0000000    0.0021910000
    0.0000000       0.0000000   9.6320000e-05    0.0069510000      0.22550000       230.80000
    0.0000000       0.0000000       0.0000000     0.020290000   0.00069980000       576.50000
    0.0000000       0.0000000       0.0000000       0.0000000       1.6850000    0.0057770000
    0.0000000       0.0000000       0.0000000       0.0000000       0.0000000   3.7600000e-11
    0.0000000       0.0000000       0.0000000       0.0000000       0.0000000       0.0000000
```

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

24/12/2015, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aij_list.pro

ATOMNEB_READ_AIJ_LIST

private

This function returns the list of transition probabilities (Aij) from the 1st binary table extension of the FITS data file ('AtomAij.fits').

```
result = atomneb_read_aij_list(Atom_Aij_file)
```

Returns

type=an array of data. This function returns the aij_data_list: { Aij_Data:", Extention:0.0}

Parameters

Atom_Aij_file IN REQUIRED TYPE=string
the FITS data file name ('AtomAij.fits')

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

24/12/2015, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_aij_references.pro

ATOMNEB_READ_AIJ_REFERENCES

private

This function returns the reference list of transition probabilities (Aij) from the 1nd binary table extension of the FITS data file ('AtomAij.fits').

```
result = atomneb_read_aij_references(Atom_Aij_file)
```

Returns

type=an array of data. This function returns the aij_data_reference: { Reference:'', Citation:''}

Parameters

Atom_Aij_file IN REQUIRED TYPE=string
the FITS data file name ('AtomAij.fits')

Author

Ashkbiz Danehkar

Copyright

History

24/12/2015, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_elj.pro

ATOMNEB_READ_ELJ

This function returns the energy levels (Ej) from the table extensions of the FITS data file ('AtomElj.fits').

```
result = atomneb_read_elj(Atom_Elj_file, atom, ion, level_num=string)
```

Returns

type=an array of data. This function returns the elj_data: { Configuration:'', Term:'', J:'', J_v:0.0, Ej:0.0, Reference:''}.

Parameters

Atom_Elj_file IN REQUIRED TYPE=string
the FITS data file name ('AtomElj.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'o'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Keywords

level_num IN TYPE=string
set for the maximum level number.

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data', 'collection']
 IDL> Atom_Elj_file = filepath('AtomElj.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='o'
 IDL> ion='iii'
 IDL> oiii_elj_data=atomneb_read_elj(Atom_Elj_file, atom, ion, level_num=6)
 IDL> print,oiii_elj_data.J_v
    0.00000      1.00000      2.00000      2.00000      0.00000      2.00000
 IDL> print,oiii_elj_data.Ej
    0.0000000       113.17800       306.17400       20273.270       43185.740       60324.790
```

Author

Ashkbiz Danehkar

Copyright

History

24/12/2015, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_elj_list.pro

ATOMNEB_READ_ELJ_LIST

private

This function returns the list of energy levels (Ej) from the 1st binary table extension of the FITS data file ('AtomElj.fits')

```
result = atomneb_read_elj_list(Atom_Elj_file)
```

Returns

type=an array of data. This function returns the elj_data_list: { Elj_Data:'', Extention:0.0}

Parameters

Atom_Elj_file IN REQUIRED TYPE=string
the FITS data file name ('AtomElj.fits')

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

24/12/2015, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_elj_references.pro

ATOMNEB_READ_ELJ_REFERENCES

private

This function returns the reference list of energy levels (Ej) from the 2nd binary table extension of the FITS data file ('AtomElj.fits').

```
result = atomneb_read_elj_references(Atom_Elj_file)
```

Returns

type=an array of data. This function returns the aij_data_reference: { Reference:'', Citation:''}

Parameters

Atom_Elj_file IN REQUIRED TYPE=string
the FITS data file name ('AtomElj.fits')

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

24/12/2015, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_omij.pro

ATOMNEB_READ_OMIJ

This function returns the collision strengths (omega_ij) from the
table extensions of the FITS data file ('AtomOmij.fits').

```
result = atomneb_read_omij(Atom_Omij_file, atom, ion, reference=string, level_num=string)
```

Returns

type=an array of data. This function returns the omij_data: {
level1:0, level2:0, strength:dblarr(temp_steps)}.

Parameters

Atom_Omij_file IN REQUIRED TYPE=string
the FITS data file name ('AtomOmij.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'o'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Keywords

reference IN TYPE=string
set for the reference e.g. 'SSB14'

level_num IN TYPE=string
set for the maximum level number.

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data', 'collection']
 IDL> Atom_Omij_file = filepath('AtomOmij.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='o'
 IDL> ion='iii'
 IDL> reference='SSB14'
 IDL> oiii_omij_data=atomneb_read_omij(Atom_Omij_file, atom, ion, reference=reference)
 IDL> print,oiii_omij_data.level1
    0       1       1       1       1       2       2       2       3       3       4
 IDL> print,oiii_omij_data.level2
    0       2       3       4       5       3       4       5       4       5       5
 IDL> print,oiii_omij_data[0].strength
    100.00000       125.89254       158.48932       199.52623       251.18864       ...
```

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

24/12/2015, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_omij_list.pro

ATOMNEB_READ_OMIJ_LIST

private

This function returns the list of collision strengths (omega_ij) from the 1st binary table extension of the FITS data file ('Atom-Omij.fits').

```
result = atomneb_read_omij_list(Atom_Omij_file)
```

Returns

type=an array of data. This function returns the omij_data_list:
{ Omij_Data:'', Extention:0.0}

Parameters

Atom_Omij_file IN REQUIRED TYPE=string
the FITS data file name ('AtomOmij.fits')

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

24/12/2015, IDL code by A. Danehkar

Version

0.0.1

atomneb_read_omij_references.pro

ATOMNEB_READ_OMIJ_REFERENCES

private

This function returns the reference list of collision strengths
(omega_ij) from the 2nd binary table extension of the FITS data
file ('AtomOmij.fits').

```
result = atomneb_read_omij_references(Atom_Omij_file)
```

Returns

type=an array of data. This function returns the aij_data_reference:
{ Reference:'', Citation:''}

Parameters

Atom_Omij_file IN REQUIRED TYPE=string
the FITS data file name ('AtomOmij.fits')

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

24/12/2015, IDL code by A. Danehkar

Version

0.0.1

atomneb_search_aeff_collection.pro

ATOMNEB_SEARCH_AEFF_COLLECTION

This function searches effective recombination coefficients (Aeff)
for given element and ionic levels in the FITS data file ('rc_collection.fits'),
and returns the data entry.

```
result = atomneb_search_aeff_collection(Atom_RC_file, atom, ion, /br)
```

Returns

type=array of data. This function returns the Aeff_Data.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_collection.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'c'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Keywords

br IN TYPE=boolean
set for the branching ratios (Br), may not necessary

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_rc_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_collection.fits', root_dir=base_dir, subdir=data_rc_dir )
 IDL> atom='c'
 IDL> ion='iii' ; C III
 IDL> list_cii_aeff_data=atomneb_search_aeff_collection(Atom_RC_file, atom, ion)
 IDL> print,list_cii_aeff_data
    c_iii_aeff
```

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_search_aeff_he_i_pfsd12.pro

ATOMNEB_SEARCH_AEFF_HE_I_PFSD12

This function searches effective recombination coefficients (Aeff)
for given element and ionic levels in the FITS data file ('rec_he_ii_PFSD12.fits'),
and returns the data entry.

```
result = atomneb_search_aeff_he_i_pfsd12(Atom_RC_file, atom, ion)
```

Returns

type=array of data. This function returns the Aeff_Data.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_he_ii_PFSD12.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'he'

ion IN REQUIRED TYPE=string
ionic level e.g 'ii'

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_rc_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_he_ii_PFSD12.fits', root_dir=base_dir, subdir=data_rc_dir )
 IDL> atom='he'
 IDL> ion='ii' ; He I
 IDL> list_hei_aeff_data=atomneb_search_aeff_he_i_pfsd12(Atom_RC_file, atom, ion)
 IDL> print,list_hei_aeff_data
    he_ii_aeff_PFSD12 he_ii_aeff_PFSD13
```

Author

Ashkbiz Danehkar

Copyright

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_search_aeff_n_ii_fsl13.pro

ATOMNEB_SEARCH_AEFF_N_II_FSL13

This function searches effective recombination coefficients (Aeff)
for given element and ionic levels in the FITS data file ('rc_n_iii_FSL13.fits'),
and returns the data entry.

```
result = atomneb_search_aeff_n_ii_fsl13(Atom_RC_file, atom, ion, wavelength)
```

Returns

type=array of data. This function returns the Aeff_Data.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_n_iii_FSL13.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'n'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

wavelength

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_rc_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_n_iii_FSL13.fits', root_dir=base_dir, subdir=data_rc_dir )
 IDL> atom='n'
 IDL> ion='iii' ; N II
 IDL> wavelength=5679.56
 IDL> list_nii_aeff_data=atomneb_search_aeff_n_ii_fsl13(Atom_RC_file, atom, ion, wavelength)
 IDL> print,list_nii_aeff_data.Wavelength
    5679.56
 IDL> print,list_nii_aeff_data.Aeff
    7810.00      1780.00      850.000      151.000      74.4000      53.1000      47.4000
    7370.00      1700.00      886.000      206.000      110.000      80.1000      70.8000
    7730.00      1680.00      900.000      239.000      138.000      103.000      92.9000
    8520.00      1710.00      905.000      244.000      142.000      107.000      97.0000
```

Author

Ashkbiz Danehkar

Copyright

History

03/07/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_search_aeff_o_ii_ssb17.pro

ATOMNEB_SEARCH_AEFF_O_II_SSB17

This function searches effective recombination coefficients (Aeff)
for given element and ionic levels in the FITS data file ('rc_o_iii_SSB17.fits'),
and returns the data entry.

```
result = atomneb_search_aeff_o_ii_ssb17(Atom_RC_file, atom, ion, case1, wavelength)
```

Returns

type=array of data. This function returns the Aeff_Data.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
: the FITS data file name ('rc_o_iii_SSB17.fits')

atom IN REQUIRED TYPE=string
: atom name e.g. 'o'

ion IN REQUIRED TYPE=string
: ionic level e.g 'iii'

case1 IN TYPE=string
: set for the case 'a' or 'b', defualt 'b'

wavelength IN TYPE=float
: set the wavelengths

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_rc_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_o_iii_SSB17.fits', root_dir=base_dir, subdir=data_rc_dir )
 IDL> atom='o'
 IDL> ion='iii' ; O II
 IDL> case1='B'
 IDL> wavelength=5325.42
 IDL> list_oii_aeff_data=atomneb_search_aeff_o_ii_ssb17(Atom_RC_file, atom, ion, case1, wavelength)
 IDL> print,list_oii_aeff_data.Wavelength
    5325.42
 IDL> print,list_oii_aeff_data.Aeff
    3.41800e-32  3.33300e-32  3.25700e-32  3.20900e-32  3.16800e-32 ...
```

Author

Ashkbiz Danehkar

Copyright

History

03/07/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_search_aeff_ppb91.pro

ATOMNEB_SEARCH_AEFF_PPB91

This function searches effective recombination coefficients (Aeff)
for given element and ionic levels in the FITS data file ('rec_PPB91.fits'),
and returns the data entry.

```
result = atomneb_search_aeff_ppb91(Atom_RC_file, atom, ion)
```

Returns

type=array of data. This function returns the Aeff_Data.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
: the FITS data file name ('rc_PPB91.fits')

atom IN REQUIRED TYPE=string
: atom name e.g. 'c'

ion IN REQUIRED TYPE=string
: ionic level e.g 'iii'

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_rc_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_PPB91.fits', root_dir=base_dir, subdir=data_rc_dir )
 IDL> atom='c'
 IDL> ion='iii'
 IDL> list_cii_aeff_data=atomneb_search_aeff_ppb91(Atom_RC_file, atom, ion)
 IDL> print,list_cii_aeff_data
    c_iii_aeff
```

Author

Ashkbiz Danehkar

Copyright

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_search_aeff_sh95.pro

ATOMNEB_SEARCH_AEFF_SH95

This function searches effective recombination coefficients (Aeff)
for given element and ionic levels in the FITS data file ('rec_SH95.fits'),
and returns the data entry.

```
result = atomneb_search_aeff_sh95(Atom_RC_file, atom, ion)
```

Returns

type=array of data. This function returns the Aeff_Data.

Parameters

Atom_RC_file IN REQUIRED TYPE=string
the FITS data file name ('rc_SH95.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'h'

ion IN REQUIRED TYPE=string
ionic level e.g 'ii'

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_rc_dir = ['atomic-data-rc']
 IDL> Atom_RC_file= filepath('rc_SH95.fits', root_dir=base_dir, subdir=data_rc_dir )
 IDL> atom='h'
 IDL> ion='ii' ; H I
```

```
IDL> list_hi_aeff_data=atomneb_search_aeff_sh95(Atom_RC_file, atom, ion)
IDL> print,list_hi_aeff_data
   h_ii_aeff_a h_ii_aeff_b
```

Author

Ashkbiz Danehkar

Copyright

History

15/01/2017, IDL code by A. Danehkar

Version

0.0.1

atomneb_search_aij.pro

ATOMNEB_SEARCH_AIJ

This function searches transition probabilities (Aij) for given element and ionic levels in the FITS data file ('AtomAij.fits'), and returns the data entry.

```
result = atomneb_search_aij(Atom_Aij_file, atom, ion)
```

Returns

type=array of data. This function returns the Aij_Data.

Parameters

Atom_Aij_file IN REQUIRED TYPE=string
the FITS data file name ('AtoAij.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'o'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data', 'collection']
 IDL> Atom_Aij_file = filepath('AtomAij.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='o'
 IDL> ion='iii'
 IDL> list_oiii_aij_data=atomneb_search_aij(Atom_Aij_file, atom, ion)
 IDL> print,list_oiii_aij_data
    o_iii_aij_FFT04-SZ00 o_iii_aij_FFT04 o_iii_aij_GMZ97-WFD96 o_iii_aij_SZ00-WFD96
```

Author

Ashkbiz Danehkar

Copyright

This library is released under a GNU General Public License.

History

24/12/2015, IDL code by A. Danehkar

Version

0.0.1

atomneb_search_omij.pro

ATOMNEB_SEARCH_OMIJ

This function searches collision strengths (omega_ij) for given element and ionic levels in the FITS data file ('AtomOmij.fits'), and returns the data entry.

```
result = atomneb_search_omij(Atom_Omij_file, atom, ion)
```

Returns

type=array of data. This function returns the Omij_Data.

Parameters

Atom_Omij_file IN REQUIRED TYPE=string
the FITS data file name ('AtomOmij.fits')

atom IN REQUIRED TYPE=string
atom name e.g. 'o'

ion IN REQUIRED TYPE=string
ionic level e.g 'iii'

Examples

For example:

```
IDL> base_dir = file_dirname(file_dirname((routine_info('$MAIN$', /source)).path))
 IDL> data_dir = ['atomic-data', 'collection']
 IDL> Atom_Omij_file = filepath('AtomOmij.fits', root_dir=base_dir, subdir=data_dir )
 IDL> atom='o'
 IDL> ion='iii'
 IDL> list_oiii_omij_data=atomneb_search_omij(Atom_Omij_file, atom, ion)
 IDL> print,list_oiii_omij_data
    o_iii_omij_AK99 o_iii_omij_LB94 o_iii_omij_Pal12-AK99 o_iii_omij_SSB14
```

Author

Ashkbiz Danehkar

Copyright

History

24/12/2015, IDL code by A. Danehkar

Version

0.0.1