AtomNeb

Release 0.2.0

Ashkbiz Danehkar

Oct 13, 2020

CONTENTS:

1 AtomNeb API functions 1

Python Module Index 25

CHAPTER

ONE

ATOMNEB API FUNCTIONS

This module contains functions for Atomic Data of Ionized Nebulae

atomneb.**get_aeff_collection_reference_citation**(*atom_rc_file*, *atom*, *ion*, *br=None*, *reference=None*)

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').

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data-rc')
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_collection.fits')
>> atom='c'
>> ion='iii' # C III
>> citation = atomneb.get_aeff_collection_reference_citation(atom_rc_file,
↪ atom, ion)
>> print(citation)
   Davey, A. R., Storey, P. J. and Kisielius, R., Astron.Astrophys.Suppl.,
↪ 142, 85, 2000
```

Returns This function returns the Citation.

Return type str

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_collection.fits')
- **atom** (*str*) – atom name e.g. 'c'
- **ion** (*str*) – ionic level e.g 'iii'
- **br** (*boolean, optional*) – set for the branching ratios (Br)
- **reference** (*str, optional*) – set for the reference

atomneb.**get_aeff_he_i_pfsd12_reference_citation**(*atom_rc_file*, *atom*, *ion*, *reference=None*)

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').

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = ['atomic-data-rc']
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_he_ii_PFSD12.fits')
>> atom='he'
>> ion='ii' # He I
>> reference='PFSD13'
>> citation = atomneb.get_aeff_he_i_pfsd12_reference_citation(atom_rc_
↪file, atom, ion, reference=reference)
>> print(citation)
   Porter, R. L., Ferland, G. J., Storey, P. J. and Detisch, M. J., MNRAS,
↪ 433L, 89, 2013
```

Returns This function returns the Citation.

Return type str

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_he_ii_PFSD12.fits')
- **atom** (*str*) – atom name e.g. 'he'
- **ion** (*str*) – ionic level e.g 'ii'
- **reference** (*str, optional*) – set for the reference e.g. 'PFSD13'

atomneb.**get_aeff_n_ii_fsl13_reference_citation**(*atom_rc_file*, *atom*, *ion*, *reference=None*)

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').

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = ['atomic-data-rc']
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_n_iii_FSL13.fits')
>> atom='n'
>> ion='iii' # N II
>> reference='FSL13'
>> citation = atomneb.get_aeff_n_ii_fsl13_reference_citation(atom_rc_file,
↪ atom, ion)
>> print(citation)
   Fang X., Storey P.J., and Liu X.-W., R. 2011, Astron.Astrophys. 530,␣
↪A18; 2013, Astron.Astrophys. 550, C2
```

Returns This function returns the Citation.

Return type str

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_n_iii_FSL13.fits')
- **atom** (*str*) – atom name e.g. 'n'

- **ion** (*str*) – ionic level e.g 'iii'
- **reference** (*str, optional*) – set for the reference e.g. 'FSL13

atomneb.**get_aeff_o_ii_ssb17_reference_citation**(*atom_rc_file*, *atom*, *ion*, *reference=None*)

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').

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = ['atomic-data-rc']
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_o_iii_SSB17_orl_
↪case_b.fits')
>> atom='o'
>> ion='iii' # O II
>> reference='SSB17'
>> citation = atomneb.get_aeff_o_ii_ssb17_reference_citation(atom_rc_file,
↪ atom, ion)
>> print(citation)
   Storey, P.J., Sochi, T. and Bastin, R. 2017, MNRAS, 470, 379; VizieR␣
↪On-line Data Catalog: VI/150
```

Returns This function returns the Citation.

Return type str

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_o_iii_SSB17.fits')
- **atom** (*str*) – atom name e.g. 'o'
- **ion** (*str*) – ionic level e.g 'iii'
- **reference** (*str, optional*) – set for the reference e.g. 'SSB17

atomneb.**get_aeff_ppb91_reference_citation**(*atom_rc_file*, *atom*, *ion*, *reference=None*)

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').

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = ['atomic-data-rc']
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_PPB91.fits')
>> atom='c'
>> ion='iii'
>> citation = atomneb.get_aeff_ppb91_reference_citation(atom_rc_file,␣
↪atom, ion)
>> print(citation)
   Pequignot, D., Petitjean, P. and Boisson, C. Astron.Astrophys., 251,␣
↪680, 1991
```

(continues on next page)

(continued from previous page)

Returns This function returns the Citation.

Return type str

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_PPB91.fits')
- **atom** (*str*) – atom name e.g. 'c'
- **ion** (*str*) – ionic level e.g 'iii'
- **reference** (*str, optional*) – set for the reference

atomneb.**get_aeff_sh95_reference_citation**(*atom_rc_file*, *atom*, *ion*, *reference=None*, *case1=None*)

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').

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = ['atomic-data-rc']
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_SH95.fits')
>> atom='h'
>> ion='ii' # H I
>> citation = atomneb.get_aeff_sh95_reference_citation(atom_rc_file, atom,
↪ ion)
>> print(citation)
   Storey, P. J. and Hummer, D. G., MNRAS, 272, 41S, 1995
```

Returns This function returns the Citation.

Return type str

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_SH95.fits')
- **atom** (*str*) – atom name e.g. 'h'
- **ion** (*str*) – ionic level e.g 'ii'
- **reference** (*str, optional*) – set for the reference
- **case1** (*str, optional*) – set for the case 'a' or 'b', defualt 'b'

atomneb.**get_aij_reference_citation**(*atom_aij_file*, *atom*, *ion*, *reference*)

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

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data', 'collection')
>> atom_aij_file = os.path.join(base_dir,data_dir, 'AtomAij.fits')
>> atom='o'
>> ion='iii'
>> reference='FFT04'
>> citation = atomneb.get_aij_reference_citation(atom_aij_file, atom, ion,
↪ reference)
>> print(citation)
   Froese Fischer et al 2004, ADNDT 87, 1
```

Returns This function returns the Citation.

Return type str

Parameters

- **Atom_Aij_file** (*str*) – the FITS data file name ('AtoAij.fits')
- **atom** (*str*) – atom name e.g. 'o'
- **ion** (*str*) – ionic level e.g 'iii'
- **reference** (*str*) – set for the reference e.g. 'FFT04'

atomneb.**get_elj_reference_citation**(*atom_elj_file*, *reference*)

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

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data', 'collection')
>> atom_elj_file = os.path.join(base_dir,data_dir, 'AtomElj.fits')
>> reference='L7288'
>> citation=atomneb.get_elj_reference_citation(atom_elj_file, reference)
>> print(citation)
   C. E. Moore, in CRC Series in Evaluated Data in Atomic Physics, 339 pp.
↪ (CRC Press, Boca Raton, FL, 1993)
```

Returns This function returns the Citation.

Return type str

Parameters

- **atom_elj_file** (*str*) – the FITS data file name ('AtomElj.fits')
- **reference** (*str*) – set for the reference e.g. 'L7288'

atomneb.**get_omij_reference_citation**(*atom_omij_file*, *atom*, *ion*, *reference*)

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

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data', 'collection')
>> atom_omij_file = os.path.join(base_dir,data_dir, 'AtomOmij.fits')
>> atom='o'
>> ion='iii'
>> reference='SSB14'
>> citation = atomneb.get_omij_reference_citation(atom_omij_file, atom,␣
↪ion, reference)
>> print(citation)
   Storey, P. J., Sochi, T., and Badnell, N. R. 2014, Astron.Astrophys.,␣
↪441, 3028
```

Returns This function returns the Citation.

Return type str

Parameters

- **atom_omij_file** (*str*) – the FITS data file name ('AtomOmij.fits')
- **atom** (*str*) – atom name e.g. 'o'
- **ion** (*str*) – ionic level e.g 'iii'
- **reference** (*str*) – set for the reference e.g. 'SSB14'

atomneb.**list_aeff_collection_references**(*atom_rc_file*, *atom*, *ion*, *br=None*)

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').

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data-rc')
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_collection.fits')
>> atom='c'
>> ion='iii' # C III
>> list_cii_aeff_reference = atomneb.list_aeff_collection_references(atom_
↪rc_file, atom, ion)
>> print(list_cii_aeff_reference)
```

Returns This function returns the references.

Return type an array of strings

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_collection.fits')
- **atom** (*str*) – atom name e.g. 'c'
- **ion** (*str*) – ionic level e.g 'iii'

- **br** (*boolean, optional*) – set for the branching ratios (Br)

atomneb.**list_aeff_he_i_pfsd12_references**(*atom_rc_file*, *atom*, *ion*)

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').

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = ['atomic-data-rc']
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_he_ii_PFSD12.fits')
>> atom='he'
>> ion='ii' # He I
>> list_hei_aeff_reference = atomneb.list_aeff_he_i_pfsd12_
↪references(atom_rc_file, atom, ion)
>> print(list_hei_aeff_reference)
   PFSD12 PFSD13
```

Returns This function returns the references.

Return type an array of strings

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_he_ii_PFSD12.fits')
- **atom** (*str*) – atom name e.g. 'he'
- **ion** (*str*) – ionic level e.g 'ii'

atomneb.**list_aeff_n_ii_fsl13_references**(*atom_rc_file*, *atom*, *ion*)

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').

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = ['atomic-data-rc']
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_n_iii_FSL13.fits')
>> atom='n'
>> ion='iii' # N II
>> list_nii_aeff_reference = atomneb.list_aeff_n_ii_fsl13_references(atom_
↪rc_file, atom, ion)
>> print(list_nii_aeff_reference)
```

Returns This function returns the references.

Return type an array of strings

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_n_iii_FSL13.fits')
- **atom** (*str*) – atom name e.g. 'n'

- **ion** (*str*) – ionic level e.g 'iii'

atomneb.**list_aeff_o_ii_ssb17_references**(*atom_rc_file*, *atom*, *ion*)

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').

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = ['atomic-data-rc']
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_o_iii_SSB17_orl_
↪case_b.fits')
>> atom='o'
>> ion='iii' # O II
>> list_oii_aeff_reference = atomneb.list_aeff_o_ii_ssb17_references(atom_
↪rc_file, atom, ion)
>> print(list_oii_aeff_reference)
```

Returns This function returns the references.

Return type an array of strings

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_o_iii_SSB17.fits')
- **atom** (*str*) – atom name e.g. 'o'
- **ion** (*str*) – ionic level e.g 'iii'

atomneb.**list_aeff_ppb91_references**(*atom_rc_file*, *atom*, *ion*)

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').

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = ['atomic-data-rc']
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_PPB91.fits')
>> atom='c'
>> ion='iii'
>> list_cii_aeff_reference = atomneb.list_aeff_ppb91_references(atom_rc_
↪file, atom, ion)
>> print(list_cii_aeff_reference)
```

Returns This function returns the references.

Return type an array of strings

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_PPB91.fits')
- **atom** (*str*) – atom name e.g. 'c'

- **ion** (*str*) – ionic level e.g 'iii'

atomneb.**list_aeff_sh95_references**(*atom_rc_file*, *atom*, *ion*)

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').

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = ['atomic-data-rc']
>> atom_rc_file = os.path.join(base_dir,data_dir,'rc_SH95.fits')
>> atom='h'
>> ion='ii' # H I
>> list_hi_aeff_reference = atomneb.list_aeff_sh95_references(atom_rc_
↪file, atom, ion)
>> print(list_hi_aeff_reference)
```

Returns This function returns the references.

Return type an array of strings

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_SH95.fits')
- **atom** (*str*) – atom name e.g. 'h'
- **ion** (*str*) – ionic level e.g 'ii'

atomneb.**list_aij_references**(*atom_aij_file*, *atom*, *ion*)

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').

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data', 'collection')
>> atom_aij_file = os.path.join(base_dir,data_dir, 'AtomAij.fits')
>> atom='o'
>> ion='iii'
>> list_oiii_aij_reference = atomneb.list_aij_references(atom_aij_file,␣
↪atom, ion)
>> print(list_oiii_aij_reference)
   FFT04-SZ00 FFT04 GMZ97-WFD96 SZ00-WFD96
```

Returns This function returns the references.

Return type an array of data

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('AtoAij.fits')
- **atom** (*str*) – atom name e.g. 'o'

- **ion** (*str*) – ionic level e.g 'iii'

atomneb.**list_omij_references**(*atom_omij_file*, *atom*, *ion*)

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').

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data', 'collection')
>> atom_omij_file = os.path.join(base_dir,data_dir, 'AtomOmij.fits')
>> atom='o'
>> ion='iii'
>> list_oiii_omij_reference = atomneb.list_omij_references(atom_omij_file,
↪ atom, ion)
>> print(list_oiii_omij_reference)
   AK99 LB94 Pal12-AK99 SSB14
```

Returns This function returns the references.

Return type str

Parameters

- **atom_omij_file** (*str*) – the FITS data file name ('AtomOmij.fits')
- **atom** (*str*) – atom name e.g. 'o'
- **ion** (*str*) – ionic level e.g 'iii'

atomneb.**read_aeff_collection**(*atom_rc_file*, *atom*, *ion*, *br=None*, *reference=None*)

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

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data-rc')
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_collection.fits')
>> atom='c'
>> ion='iii' # C III
>> cii_rc_data = atomneb.read_aeff_collection(atom_rc_file, atom, ion)
>> n_line = len(cii_rc_data.wavelength)
>> for i in range(0, n_line):
>>       print(cii_rc_data.wavelength[i], cii_rc_data.a[i],
>>             cii_rc_data.b[i], cii_rc_data.c[i],
>>             cii_rc_data.d[i], cii_rc_data.f[i])
   914.00000      0.69280000     0.021400000    -0.016300000     -0.
↪24310000     -0.88000000
   962.00000       1.0998000   -0.0042000000    -0.027900000     -0.
↪22940000     -0.96560000
   ...
```

Returns 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},

Return type an array of data

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_collection.fits')
- **atom** (*str*) – atom name e.g. 'c'
- **ion** (*str*) – ionic level e.g 'iii'
- **br** (*boolean, optional*) – set for the branching ratios (Br)
- **reference** (*str, optional*) – set for the reference

atomneb.**read_aeff_collection_list**(*atom_rc_file*)

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

Returns This function returns the aeff_data_list: { Aeff_Data:", Extension:0.0}

Return type an array of data

Parameters **atom_rc_file** (*str*) – the FITS data file name ('rc_collection.fits')

atomneb.**read_aeff_collection_references**(*atom_rc_file*)

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

Returns This function returns the aeff_data_reference: { Reference:", Citation:"}

Return type an array of data

Parameters **atom_rc_file** (*str*) – the FITS data file name ('rc_collection.fits')

atomneb.**read_aeff_he_i_pfsd12**(*atom_rc_file*, *atom*, *ion*, *wavelength=None*, *reference=None*)

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

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data-rc')
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_he_ii_PFSD12.fits')
>> atom='he'
>> ion='ii' # He I
>> hei_rc_data = atomneb.read_aeff_he_i_pfsd12(atom_rc_file, atom, ion)
>> hei_rc_data_wave = atomneb.read_aeff_he_i_pfsd12(atom_rc_file, atom,␣
↪ion, wavelength=True)
```

(continues on next page)

(continued from previous page)

```
>> print(hei_rc_data.aeff[0])
   5000.0000       10.000000      -25.379540      -25.058970      -25.
↪948440      ...
>> n_line = len(hei_rc_data_wave.wavelength)
>> for i in range(0, n_line):
>>     print(hei_rc_data_wave.wavelength[i],
>>           hei_rc_data_wave.lowerterm[i], hei_rc_data_wave.upperterm[i])
   2945.00005p^{3}P2s^{3}S
   3188.00004p^{3}P2s^{3}S
   3614.00005p^{1}P2s^{1}S
   ...
```

Returns This function returns the effective recombination coefficients.

Return type an array of data

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_he_ii_PFSD12.fits')
- **atom** (*str*) – atom name e.g. 'he'
- **ion** (*str*) – ionic level e.g 'ii'
- **wavelength** (*boolean, optional*) – set for returning the wavelengths
- **reference** (*str, optional*) – set for the reference

atomneb.**read_aeff_he_i_pfsd12_list**(*atom_rc_file*)

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')

Returns This function returns the aeff_data_list: { Aeff_Data:", Extension:0.0}

Return type an array of data

Parameters **atom_rc_file** (*str*) – the FITS data file name ('rc_he_ii_PFSD12.fits')

atomneb.**read_aeff_he_i_pfsd12_references**(*atom_rc_file*)

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').

Returns This function returns the aeff_data_reference:{ Reference:", Citation:"}

Return type an array of data

Parameters **atom_rc_file** (*str*) – the FITS data file name ('rc_he_ii_PFSD12.fits')

atomneb.**read_aeff_n_ii_fsl13**(*atom_rc_file*, *atom*, *ion*, *wavelength_range*, *wavelength=None*, *reference=None*)

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

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data-rc')
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_n_iii_FSL13.fits')
>> atom='n'
>> ion='iii' # N II
>> wavelength_range=[4400.0, 7100.0]
>> nii_rc_data = atomneb.read_aeff_n_ii_fsl13(atom_rc_file, atom, ion,␣
↪wavelength_range)
>> nii_rc_data_wave = atomneb.read_aeff_n_ii_fsl13(atom_rc_file, atom,␣
↪ion, wavelength_range, wavelength=True)
>> print(nii_rc_data.aeff[0])
   255.000      79.5000      47.3000      12.5000      ...
>> n_line = len(nii_rc_data_wave.wavelength)
>> for i in range(0, n_line):
>>     print(nii_rc_data_wave.wavelength[i], nii_rc_data_wave.tr[i], nii_
↪rc_data_wave.trans[i])
   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
   ...
```

Returns This function returns the effective recombination coefficients.

Return type an array of data

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_n_iii_FSL13.fits')
- **atom** (*str*) – atom name e.g. 'n'
- **ion** (*str*) – ionic level e.g 'iii'
- **wavelength** (*boolean, optional*) – set for returning the wavelengths
- **reference** (*str, optional*) – set for the reference

atomneb.**read_aeff_n_ii_fsl13_list**(*atom_rc_file*)

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')

Returns This function returns the aeff_data_list: {Aeff_Data:", Extension:0, IND:long(0), Wavelength: float(0.0), Tr:", Trans: '', T_X: ''}

Return type an array of data

Parameters **atom_rc_file** (*str*) – the FITS data file name ('rc_n_iii_FSL13.fits')

atomneb.**read_aeff_n_ii_fsl13_references**(*atom_rc_file*)

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').

Returns This function returns the aeff_data_reference: { Reference:", Citation:"}

Return type an array of data

Parameters **atom_rc_file** (*str*) – the FITS data file name ('rc_n_iii_FSL13.fits')

atomneb.**read_aeff_o_ii_ssb17**(*atom_rc_file*, *atom*, *ion*, *case1*, *wavelength_range*, *wavelength=None*, *reference=None*)

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

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data-rc')
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_o_iii_SSB17_orl_
↪case_b.fits')
>> atom='o'
>> ion='iii' # O II
>> case1='B'
>> wavelength_range=[5320.0, 5330.0]
>> oii_rc_data = atomneb.read_aeff_o_ii_ssb17(atom_rc_file, atom, ion,␣
↪case1, wavelength_range)
>> oii_rc_data_wave = atomneb.read_aeff_o_ii_ssb17(atom_rc_file, atom,␣
↪ion,
>>                                                 case1, wavelength_
↪range, wavelength=True)
>> print(oii_rc_data.aeff[0])
   1.64100e-30  1.60000e-30  1.56400e-30  1.54100e-30 ...
>> n_line = len(oii_rc_data_wave.wavelength)
>> for i in range(0, n_line):
>>      print(oii_rc_data_wave.wavelength[i], oii_rc_data_wave.lower_
↪term[i], oii_rc_data_wave.upper_term[i])
   5327.172s22p2(1S)3p 2Po
   5325.422s22p2(1S)3p 2Po
   5327.182s22p2(1D)3d 2Ge
   ...
```

Returns This function returns the effective recombination coefficients.

Return type an array of data

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_o_iii_SSB17.fits')
- **atom** (*str*) – atom name e.g. 'o'
- **ion** (*str*) – ionic level e.g 'iii'
- **case1** (*str*) – set for the case 'a' or 'b', defualt 'b'
- **wavelength_range** (*array*) – wavelength range e.g. [5320.0, 5330.0]
- **wavelength** (*boolean, optional*) – set for returning the wavelengths
- **reference** (*string, optional*) – set for the reference

atomneb.**read_aeff_o_ii_ssb17_list**(*atom_rc_file*)

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')

Returns This function returns the aeff_data_list: {Aeff_Data:", Extension:0, IND:long(0), Wavelength: float(0.0), Case1:", lower_term: '', upper_term: ''}

Return type an array of data

Parameters **atom_rc_file** (*str*) – the FITS data file name ('rc_o_iii_SSB17.fits')

atomneb.**read_aeff_o_ii_ssb17_references**(*atom_rc_file*)

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').

Returns This function returns the aeff_data_reference: { Reference:", Citation:"}

Return type an array of data

Parameters **atom_rc_file** (*str*) – the FITS data file name ('rc_o_iii_SSB17.fits')

atomneb.**read_aeff_ppb91**(*atom_rc_file*, *atom*, *ion*, *reference=None*)

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

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data-rc')
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_PPB91.fits')
>> atom='c'
>> ion='iii' # C II
>> cii_rc_data = atomneb.read_aeff_ppb91(atom_rc_file, atom, ion)
>> n_line = len(cii_rc_data.wavelength)
>> for i in range(0, n_line):
>>           print(cii_rc_data.ion[i], cii_rc_data.case1[i], cii_rc_data.
↪wavelength[i],
>>                 cii_rc_data.a[i], cii_rc_data.b[i], cii_rc_data.c[i],
>>                 cii_rc_data.d[i], cii_rc_data.br[i], cii_rc_data.q[i],␣
↪cii_rc_data.y[i])
   C2+A       9903.4600      0.69700000     -0.78400000      ...
   C2+A       4267.1500       1.0110000     -0.75400000      ...
   ...
```

Returns 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}

Return type an array of data

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_PPB91.fits')
- **atom** (*str*) – atom name e.g. 'c'
- **ion** (*str*) – ionic level e.g 'iii'
- **reference** (*str, optional*) – set for the reference

atomneb.**read_aeff_ppb91_list**(*atom_rc_file*)

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

Returns This function returns the aeff_data_list: { Aeff_Data:", Extension:0.0}

Return type an array of data

Parameters **atom_rc_file** (*str*) – the FITS data file name ('rc_PPB91.fits')

atomneb.**read_aeff_ppb91_references**(*atom_rc_file*)

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

Returns This function returns the aeff_data_reference: { Reference:", Citation:"}

Return type an array of data

Parameters **atom_rc_file** (*str*) – the FITS data file name ('rc_PPB91.fits')

atomneb.**read_aeff_sh95**(*atom_rc_file*, *atom*, *ion*, *reference=None*, *case1=None*)

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

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data-rc')
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_SH95.fits')
>> atom='h'
>> ion='ii' # H I
>> hi_rc_data = atomneb.read_aeff_sh95(atom_rc_file, atom, ion)
>> print(hi_rc_data.aeff[0])
   100.00000       500.00000       0.0000000   4.2140000e-27    1.
↪7560000e-27  ...
   ...
```

Returns This function returns the effective recombination coefficients.

Return type an array of data

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_SH95.fits')
- **atom** (*str*) – atom name e.g. 'h'
- **ion** (*str*) – ionic level e.g 'ii'
- **reference** (*str, optional*) – set for the reference
- **case1** (*boolean, optional*) – set for the case 'a' or 'b', defualt 'b'

atomneb.**read_aeff_sh95_list**(*atom_rc_file*)

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

Returns This function returns the aeff_data_list: { Aeff_Data:", Extension:0.0}

Return type an array of data

Parameters **atom_rc_file** (*str*) – the FITS data file name ('rc_SH95.fits')

atomneb.**read_aeff_sh95_references**(*atom_rc_file*)

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

Returns This function returns the aeff_data_reference: { Reference:", Citation:"}

Return type an array of data

Parameters **atom_rc_file** (*str*) – the FITS data file name ('rc_SH95.fits')

atomneb.**read_aij**(*atom_aij_file*, *atom*, *ion*, *reference=None*, *level_num=None*)

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

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data', 'collection')
>> atom_aij_file = os.path.join(base_dir,data_dir, 'AtomAij.fits')
>> atom='o'
>> ion='iii'
>> reference='FFT04'
>> oiii_aij_data = atomneb.read_aij(atom_aij_file, atom, ion, reference)
>> 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
```

Returns This function returns the aij_data: { Aij:dblarr(n_level,n_level) }.

Return type an array of data

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('AtoAij.fits')
- **atom** (*str*) – atom name e.g. 'o'
- **ion** (*str*) – ionic level e.g 'iii'
- **reference** (*str, optional*) – set for the reference
- **level_num** (*str, optional*) – set for the maximum level number

atomneb.**read_aij_list**(*atom_aij_file*)

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

Returns This function returns the aij_data_list: { Aij_Data:", Extension:0.0}

Return type an array of data

Parameters **Atom_Aij_file** (*str*) – the FITS data file name ('AtomAij.fits')

atomneb.**read_aij_references**(*atom_aij_file*)

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

Returns This function returns the aij_data_reference: { Reference:", Citation:"}

Return type an array of data

Parameters **Atom_Aij_file** (*str*) – the FITS data file name ('AtomAij.fits')

atomneb.**read_elj**(*atom_elj_file*, *atom*, *ion*, *level_num=None*)

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

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data', 'collection')
>> atom_elj_file = os.path.join(base_dir,data_dir, 'AtomElj.fits')
>> atom='o'
>> ion='iii'
>> oiii_elj_data=atomneb.read_elj(atom_elj_file, atom, ion, level_num=6)
>> print(np.asarray(oiii_elj_data.j_v))
   0.00000      1.00000      2.00000      2.00000      0.00000      2.
↪00000
>> print(np.asarray(oiii_elj_data.ej))
   0.0000000       113.17800       306.17400       20273.270       43185.
↪740       60324.790
```

Returns This function returns the elj_data:{ Configuration:", Term:", J:", J_v:0.0, Ej:0.0, Reference:"}.

Return type an array of data

Parameters

- **atom_elj_file** (*str*) – the FITS data file name ('AtomElj.fits')
- **atom** (*str*) – atom name e.g. 'o'
- **ion** (*str*) – ionic level e.g 'iii'
- **level_num** (*int, optional*) – set for the maximum level number.

atomneb.**read_elj_list**(*atom_elj_file*)

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

Returns This function returns the elj_data_list: { Elj_Data:", Extension:0.0}

Return type an array of data

Parameters **atom_elj_file** (*str*) – the FITS data file name ('AtomElj.fits')

atomneb.**read_elj_references**(*atom_elj_file*)

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

Returns This function returns the aij_data_reference: { Reference:", Citation:"}

Return type an array of data

Parameters **atom_elj_file** (*str*) – the FITS data file name ('AtomElj.fits')

atomneb.**read_omij**(*atom_omij_file*, *atom*, *ion*, *reference=None*, *level_num=None*)

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

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data', 'collection')
>> atom_omij_file = os.path.join(base_dir,data_dir, 'AtomOmij.fits')
>> atom='o'
>> ion='iii'
>> reference='SSB14'
>> oiii_omij_data=atomneb.read_omij(atom_omij_file, atom, ion,␣
↪reference=reference)
>> print(np.asarray(oiii_omij_data.level1))
   0       1       1       1       1       2       2       2       3␣
↪ 3       4
>> print(np.asarray(oiii_omij_data.level2))
   0       2       3       4       5       3       4       5       4␣
↪ 5       5
>> print(np.asarray(oiii_omij_data.strength)[0])
   100.00000       125.89254       158.48932       199.52623       251.
↪18864       ...
```

Returns This function returns the omij_data: { level1:0, level2:0, strength:array(temp_steps)}.

Return type an array of data

Params

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'

Parameters

- **reference** (*str, optional*) – set for the reference e.g. 'SSB14'
- **level_num** (*int, optional*) – set for the maximum level number.

atomneb.**read_omij_list**(*atom_omij_file*)

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

Returns This function returns the omij_data_list: { Omij_Data:", Extension:0.0}

Return type an array of data

Parameters **atom_omij_file** (*str*) – the FITS data file name ('AtomOmij.fits')

atomneb.**read_omij_references**(*atom_omij_file*)

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

Returns his function returns the aij_data_reference: { Reference:", Citation:"}

Return type an array of data

Parameters **atom_omij_file** (*str*) – the FITS data file name ('AtomOmij.fits')

atomneb.**search_aeff_collection**(*atom_rc_file*, *atom*, *ion*, *br=None*)

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.

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data-rc')
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_collection.fits')
>> atom='c'
>> ion='iii' # C III
>> list_cii_aeff_data = atomneb.search_aeff_collection(atom_rc_file, atom,
↪ ion)
>> print(list_cii_aeff_data)
   c_iii_aeff
```

Returns This function returns the Aeff_Data.

Return type an array of data

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_collection.fits')
- **atom** (*str*) – atom name e.g. 'c'
- **ion** (*str*) – ionic level e.g 'iii'
- **br** (*boolean, optional*) – set for the branching ratios (Br), may not necessary

atomneb.**search_aeff_he_i_pfsd12**(*atom_rc_file*, *atom*, *ion*)

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.

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data-rc')
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_he_ii_PFSD12.fits')
>> atom='he'
>> ion='ii' # He I
>> list_hei_aeff_data = atomneb.search_aeff_he_i_pfsd12(atom_rc_file,␣
↪atom, ion)
>> print(list_hei_aeff_data)
   he_ii_aeff_PFSD12 he_ii_aeff_PFSD13
```

Returns This function returns the Aeff_Data.

Return type an array of data

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_he_ii_PFSD12.fits')
- **atom** (*str*) – atom name e.g. 'he'
- **ion** (*str*) – ionic level e.g 'ii'

atomneb.**search_aeff_n_ii_fsl13**(*atom_rc_file*, *atom*, *ion*, *wavelength*)

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.

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data-rc')
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_n_iii_FSL13.fits')
>> atom='n'
>> ion='iii' # N II
>> wavelength=5679.56
>> list_nii_aeff_data = atomneb.search_aeff_n_ii_fsl13(atom_rc_file, atom,
↪ ion, wavelength)
>> print(np.asarray(list_nii_aeff_data.wavelength))
   5679.56
>> print(np.asarray(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
```

Returns This function returns the Aeff_Data.

Return type an array of data

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_n_iii_FSL13.fits')
- **atom** (*str*) – atom name e.g. 'n'
- **ion** (*str*) – ionic level e.g 'iii'
- **wavelength** (*int*) – set the wavelengths
- **reference** (*str, optional*) – set for the reference

atomneb.**search_aeff_o_ii_ssb17**(*atom_rc_file*, *atom*, *ion*, *case1*, *wavelength*)

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.

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data-rc')
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_o_iii_SSB17_orl_
↪case_b.fits')
>> atom='o'
>> ion='iii' # O II
>> case1='B'
>> wavelength=5325.42
>> list_oii_aeff_data = atomneb.search_aeff_o_ii_ssb17(atom_rc_file, atom,
↪ ion, case1, wavelength)
>> print(np.asarray(list_oii_aeff_data.wavelength))
   5325.42
>> print(np.asarray(list_oii_aeff_data.aeff))
   3.41800e-32  3.33300e-32  3.25700e-32  3.20900e-32  3.16800e-32 ...
```

Returns This function returns the Aeff_Data.

Return type an array of data

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_o_iii_SSB17.fits')
- **atom** (*str*) – atom name e.g. 'o'
- **ion** (*str*) – ionic level e.g 'iii'
- **case1** (*str*) – set for the case 'a' or 'b', defualt 'b'
- **wavelength** (*float*) – set the wavelengths

atomneb.**search_aeff_ppb91**(*atom_rc_file*, *atom*, *ion*)

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.

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data-rc')
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_PPB91.fits')
>> atom='c'
>> ion='iii'
>> list_cii_aeff_data = atomneb.search_aeff_ppb91(atom_rc_file, atom, ion)
>> print(list_cii_aeff_data)
   c_iii_aeff
```

Returns This function returns the Aeff_Data.

Return type an array of data

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_PPB91.fits')
- **atom** (*str*) – atom name e.g. 'c'
- **ion** (*str*) – ionic level e.g 'iii'

atomneb.**search_aeff_sh95**(*atom_rc_file*, *atom*, *ion*)

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.

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data-rc')
>> atom_rc_file = os.path.join(base_dir,data_dir, 'rc_SH95.fits')
>> atom='h'
>> ion='ii' # H I
>> list_hi_aeff_data = atomneb.search_aeff_sh95(atom_rc_file, atom, ion)
>> print(list_hi_aeff_data)
   h_ii_aeff_a h_ii_aeff_b
```

Returns This function returns the Aeff_Data.

Return type an array of data

Parameters

- **atom_rc_file** (*str*) – the FITS data file name ('rc_SH95.fits')
- **atom** (*str*) – atom name e.g. 'h'
- **ion** (*str*) – ionic level e.g 'ii'

atomneb.**search_aij**(*atom_aij_file*, *atom*, *ion*)

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

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data', 'collection')
>> atom_aij_file = os.path.join(base_dir,data_dir, 'AtomAij.fits')
>> atom='o'
>> ion='iii'
>> list_oiii_aij_data = atomneb.search_aij(atom_aij_file, atom, ion)
>> print(list_oiii_aij_data)
   o_iii_aij_FFT04-SZ00 o_iii_aij_FFT04 o_iii_aij_GMZ97-WFD96 o_iii_aij_
↪SZ00-WFD96
```

Returns This function returns the Aij_Data.

Return type an array of data

Parameters

- **Atom_Aij_file** (*str*) – the FITS data file name ('AtomAij.fits')
- **atom** (*str*) – atom name e.g. 'o'
- **ion** (*str*) – ionic level e.g 'iii

atomneb.**search_omij**(*atom_omij_file*, *atom*, *ion*)

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.

For example:

```
>> import atomneb
>> import numpy as np
>> import os
>> base_dir = os.getcwd()
>> data_dir = os.path.join('..','atomic-data', 'collection')
>> atom_omij_file = os.path.join(base_dir,data_dir, 'AtomOmij.fits')
>> atom='o'
>> ion='iii'
>> list_oiii_omij_data = atomneb.search_omij(atom_omij_file, atom, ion)
>> print(list_oiii_omij_data)
   o_iii_omij_AK99 o_iii_omij_LB94 o_iii_omij_Pal12-AK99 o_iii_omij_SSB14
```

Returns This function returns the Omij_Data.

Return type an array of data

Parameters

- **atom_omij_file** (*str*) – the FITS data file name ('AtomOmij.fits')
- **atom** (*str*) – atom name e.g. 'o'
- **ion** (*str*) – ionic level e.g 'iii'

PYTHON MODULE INDEX

a

atomneb, 1