periodic Documentation

Release 0

Luis Naranjo

May 26, 2012

CONTENTS

Periodic is an open source simple python API/command line script for the periodic table.

Developed by Luis Naranjo <luisnaranjo733@hotmail.com>

```
>>> import periodic
>>> element = periodic.element(12)
>>> attributes = ['atomic', 'symbol', 'name', 'mass']
>>> for attribute in attributes:
...     print getattr(element, attribute)
...
12
Mg
Magnesium
24.305
```

CHAPTER

ONE

INSTALLATION

If you haven't installed pip yet, here is an excellent guide on how to do so (in the 'Properly Install Python' section).

After you have the that all set up, you can run the following command:

```
>>> pip install periodic
```

or if you are on a Linux or Mac OS,

```
>>> sudo pip install periodic
```

CHAPTER

TWO

RETRIEVE ELEMENT AS AN OBJECT

```
>>> from periodic.table import element
>>> hydrogen = element('hydrogen')
>>> hydrogen.mass
1.0079
>>> hydrogen.symbol
'H'
```

class periodic.table.**element**

Takes periodic data as input, and returns the correct element.

Class arguments

The input for elements can be any of the following, and is case *insensitive*.

- element name (example: hydrogen) - **STRING**
- element symbol (example: H) - **STRING**
- atomic number (example: 1) - **INTEGER**
- atomic mass (example: 1.0079) - **FLOATING POINT**

Returns

Returns an Element object, or None

Element attributes

- symbol
- name
- mass
- atomic
- charge
- type

CHAPTER

THREE

PERIODIC MASS VARIABLES

Periodic also includes 'loose' atomic mass variables, for quick calculations.

```
>>> from periodic.mass import *
>>> H
1.0079
>>> H+Cl
36.4609
```

CHAPTER

FOUR

ADVANCED DATABASE QUERIES

Periodic relies on sqlalchemy for storage. Periodic leaves the "session" variable exposed, from sqlalchemy.

You can use this to make useful database queries.

Here are a few examples to get your ideas flowing:

1. If you wanted to show the first three elements, ordered by their symbols you can do something like this:

```
>>> from periodic.table import session, Element
>>> session.query(Element).order_by(Element.symbol).all()[:3]  #
[<Element('Ac', '89')>, <Element('Ag', '47')>, <Element('Al', '13')>]
```

2. show the four heaviest elements in the periodic table (ordered by atomic mass)

```
>>> from periodic.table import session, Element
>>> session.query(Element).order_by(Element.mass).all()[-4:]  # Show the four heaviest elements in t
[<Element('Uup', '115')>, <Element('Uuq', '114')>, <Element('Uuh', '116')>, <Element('Uuo', '118')>]
```

Refer to the sqlalchemy documentation for more information on using that ORM.

CHAPTER

FIVE

ASCII PERIODIC TABLE

There is also a nice ascii periodic table available:

```
>>> import periodic
>>> print periodic.table
  -----                                                                 -----
1 | H |                                                               |He |
  |---+----                                         --------------------+---|
2 |Li |Be |                                         | B | C | N | O | F |Ne |
  |---+---|                                         |---+---+---+---+---+---|
3 |Na |Mg |3B  4B  5B  6B  7B |    8B     |1B  2B |Al |Si | P | S |Cl |Ar |
  |---+---+---------------------------------------+---+---+---+---+---+---|
4 | K |Ca |Sc |Ti | V |Cr |Mn |Fe |Co |Ni |Cu |Zn |Ga |Ge |As |Se |Br |Kr |
  |---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---|
5 |Rb |Sr | Y |Zr |Nb |Mo |Tc |Ru |Rh |Pd |Ag |Cd |In |Sn |Sb |Te | I |Xe |
  |---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---|
6 |Cs |Ba |LAN|Hf |Ta | W |Re |Os |Ir |Pt |Au |Hg |Tl |Pb |Bi |Po |At |Rn |
  |---+---+---+-------------------------------------------------------------
7 |Fr |Ra |ACT|
  -------------
              ---------------------------------------------------------------
   Lanthanide |La |Ce |Pr |Nd |Pm |Sm |Eu |Gd |Tb |Dy |Ho |Er |Tm |Yb |Lu |
              |---+---+---+---+---+---+---+---+---+---+---+---+---+---+---|
   Actinide   |Ac |Th |Pa | U |Np |Pu |Am |Cm |Bk |Cf |Es |Fm |Md |No |Lw |
              ---------------------------------------------------------------
```

CHAPTER

SIX

INTERACTIVE SHELL (API)

Invoking the interactive shell from python is as easy as

```
>>> import periodic
>>> periodic.interactive_shell()
```

CHAPTER

SEVEN

INTERACTIVE SHELL (CONSOLE SCRIPT)

Periodic provides an originally named console script called ‘periodic’.

It’s usage for now is limited to periodic table reference.

In the future, it will be able to do with elements!

```
$ periodic
Enter any of the following periodic values of the element you are looking for:
        ['atomic', 'symbol', 'name', 'mass']

Use ^C or type 'exit' to exit.
================================================================================
> 12
atomic: 12
symbol: Mg
name: Magnesium
mass: 24.305
================================================================================
> uranium
atomic: 92
symbol: U
name: Uranium
mass: 238.02891
================================================================================
> H
atomic: 1
symbol: H
name: Hydrogen
mass: 1.00794
================================================================================
> 15.9994
atomic: 8
symbol: O
name: Oxygen
mass: 15.9994
================================================================================
```

PYTHON MODULE INDEX

p