Package dns :: Module tokenizer

[hide private]
[frames] | no frames]

Source Code for Module dns.tokenizer

```
# Copyright (C) 2003-2007, 2009-2011 Nominum, Inc.
#
# Permission to use, copy, modify, and distribute this software and its
# documentation for any purpose with or without fee is hereby granted,
# provided that the above copyright notice and this permission notice
# appear in all copies.
#
# THE SOFTWARE IS PROVIDED "AS IS" AND NOMINUM DISCLAIMS ALL WARRANTIES
# WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
# MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL NOMINUM BE LIABLE FOR
# ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
# WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
# ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT
# OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

"""Tokenize DNS master file format"""

from io import StringIO
import sys

import dns.exception
import dns.name
import dns.ttl
from ._compat import long, text_type, binary_type

_DELIMITERS = {
    ' ': True,
    '\t': True,
    '\n': True,
    ';': True,
    '(': True,
    ')': True,
    '"': True}

_QUOTING_DELIMITERS = {'"': True}

EOF = 0
EOL = 1
WHITESPACE = 2
IDENTIFIER = 3
QUOTED_STRING = 4
COMMENT = 5
DELIMITER = 6


class UngetBufferFull(dns.exception.DNSException):

    """An attempt was made to unget a token when the unget buffer was full."""


class Token(object):

    """A DNS master file format token.

    @ivar ttype: The token type
    @type ttype: int
    @ivar value: The token value
    @type value: string
    @ivar has_escape: Does the token value contain escapes?
    @type has_escape: bool
    """

    def __init__(self, ttype, value='', has_escape=False):
        ...

    def is_eof(self):
        return self.ttype == EOF

    def is_eol(self):
        return self.ttype == EOL

    def is_whitespace(self):
        return self.ttype == WHITESPACE

    def is_identifier(self):
        return self.ttype == IDENTIFIER

    def is_quoted_string(self):
        return self.ttype == QUOTED_STRING

    def is_comment(self):
        return self.ttype == COMMENT

    def is_delimiter(self):
        return self.ttype == DELIMITER

    def is_eol_or_eof(self):
        return self.ttype == EOL or self.ttype == EOF

    def __eq__(self, other):
        ...

    def __ne__(self, other):
        ...

    def __str__(self):
        return '%d "%s"' % (self.ttype, self.value)

    def unescape(self):
        ...

    # compatibility for old-style tuple tokens

    def __len__(self):
        return 2

    def __iter__(self):
        return iter((self.ttype, self.value))

    def __getitem__(self, i):
        ...


class Tokenizer(object):
    ...
```

Home **Trees** **Indices** **Help** **dnspython**

Generated by Epydoc 3.0.1 on Fri Sep 30 08:16:44 2016 http://epydoc.sourceforge.net