32.5. `token` — Constants used with Python parse trees

Source code: Lib/token.py

This module provides constants which represent the numeric values of leaf nodes of the parse tree (terminal tokens). Refer to the file `Grammar/Grammar` in the Python distribution for the definitions of the names in the context of the language grammar. The specific numeric values which the names map to may change between Python versions.

The module also provides a mapping from numeric codes to names and some functions. The functions mirror definitions in the Python C header files.

token.**tok_name**
> Dictionary mapping the numeric values of the constants defined in this module back to name strings, allowing more human-readable representation of parse trees to be generated.

token.**ISTERMINAL**(*x*)
> Return true for terminal token values.

token.**ISNONTERMINAL**(*x*)
> Return true for non-terminal token values.

token.**ISEOF**(*x*)
> Return true if *x* is the marker indicating the end of input.

The token constants are:

token.**ENDMARKER**
token.**NAME**
token.**NUMBER**
token.**STRING**
token.**NEWLINE**
token.**INDENT**
token.**DEDENT**
token.**LPAR**
token.**RPAR**
token.**LSQB**
token.**RSQB**
token.**COLON**

```
token.COMMA
token.SEMI
token.PLUS
token.MINUS
token.STAR
token.SLASH
token.VBAR
token.AMPER
token.LESS
token.GREATER
token.EQUAL
token.DOT
token.PERCENT
token.LBRACE
token.RBRACE
token.EQEQUAL
token.NOTEQUAL
token.LESSEQUAL
token.GREATEREQUAL
token.TILDE
token.CIRCUMFLEX
token.LEFTSHIFT
token.RIGHTSHIFT
token.DOUBLESTAR
token.PLUSEQUAL
token.MINEQUAL
token.STAREQUAL
token.SLASHEQUAL
token.PERCENTEQUAL
token.AMPEREQUAL
token.VBAREQUAL
token.CIRCUMFLEXEQUAL
token.LEFTSHIFTEQUAL
token.RIGHTSHIFTEQUAL
token.DOUBLESTAREQUAL
token.DOUBLESLASH
token.DOUBLESLASHEQUAL
token.AT
token.ATEQUAL
token.RARROW
token.ELLIPSIS
token.OP
token.AWAIT
```

`token.`**`ASYNC`**

`token.`**`ERRORTOKEN`**

`token.`**`N_TOKENS`**

`token.`**`NT_OFFSET`**

Changed in version 3.5: Added `AWAIT` and `ASYNC` tokens. Starting with Python 3.7, "async" and "await" will be tokenized as `NAME` tokens, and `AWAIT` and `ASYNC` will be removed.