Tokens

In Python, a token is the smallest unit of the source code that the Python interpreter recognizes during the process of lexical analysis (the first step in code compilation or interpretation). Each token denotes a meaningful element in Python, such as

1. Keywords.
2. Punctuation/delimiters.
3. Identifiers.
4. Operators.
5. Literals.

Keywords

In Python, a **keyword** is a **reserved word** that has a **specific meaning** and purpose in the language. These words are part of Python's **syntax** and **cannot be used as variable names**, function names, or identifiers.

Characteristics of Keywords:

- Predefined by the Python language.
- Case-sensitive (e.g., `True` is different from `true`).
- Cannot be used for anything other than their intended purpose.

Examples:

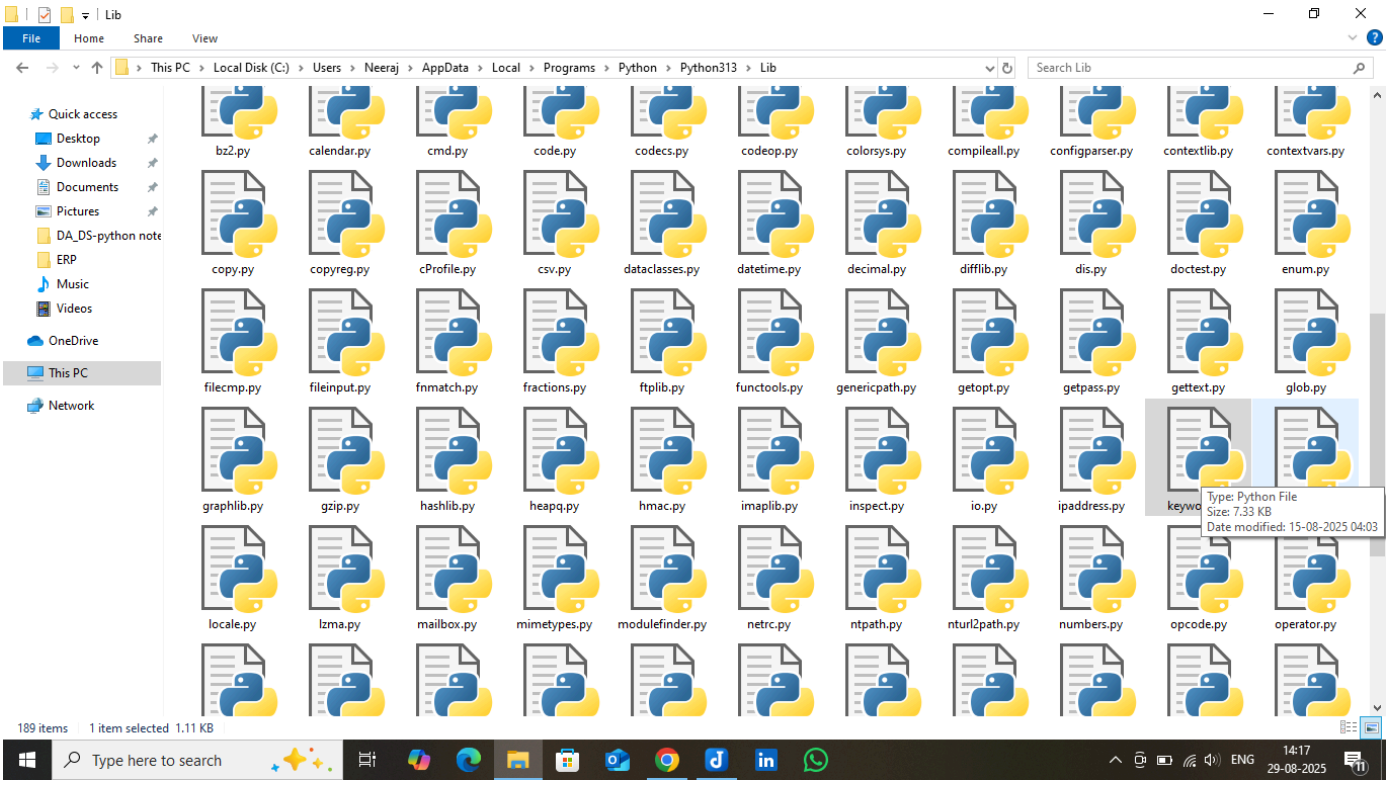
Keyword	Purpose
`if`	Used for conditional statements
`else`	Defines the alternative block of `if`
`elif`	Else if – additional condition checks
`for`	Used for looping
`while`	Used for creating loops
`def`	Defines a function
`return`	Returns a value from a function
`class`	Defines a class
`try`	Starts a try block for exception handling
`except`	Catches exceptions

Keyword	Purpose
`import`	Imports modules
`from`	Imports specific parts of a module
`pass`	A placeholder that does nothing
`break`	Exits the current loop
`continue`	Skips to the next iteration in a loop
`True`	Boolean true value
`False`	Boolean false value
`None`	Represents the absence of a value

keyword.py is already present in a python source code
`path C:\Users\Neeraj\AppData\Local\Programs\Python\Python313\Lib`
`keyword.py`

```
"""Keywords (from "Grammar/python.gram")

This file is automatically generated; please don't muck it up!

To update the symbols in this file, 'cd' to the top directory of
the python source tree and run:

    PYTHONPATH=Tools/peg_generator python3 -m pegen.keywordgen \
        Grammar/python.gram \
        Grammar/Tokens \
        Lib/keyword.py

Alternatively, you can run 'make regen-keyword'.
"""

__all__ = ["iskeyword", "issoftkeyword", "kwlist", "softkwlist"]

kwlist = [
    'False',
    'None',
    'True',
    'and',
    'as',
    'assert',
    'async',
    'await',
```

```
    'break',
    'class',
    'continue',
    'def',
    'del',
    'elif',
    'else',
    'except',
    'finally',
    'for',
    'from',
    'global',
    'if',
    'import',
    'in',
    'is',
    'lambda',
    'nonlocal',
    'not',
    'or',
    'pass',
    'raise',
    'return',
    'try',
    'while',
    'with',
    'yield'
]

softkwlist = [
    '_',
    'case',
    'match',
    'type'
]

iskeyword = frozenset(kwlist).__contains__
issoftkeyword = frozenset(softkwlist).__contains__
```



if you want to get all the pre-define keywords and soft keywords then you try below mention code:---

```
import keyword

all_keys = keyword.kwlist
print(all_keys)

all_soft_keys = keyword.softkwlist
print(all_soft_keys)

# output
# all_keywords
['False', 'None', 'True', 'and', 'as', 'assert', 'async',
 'await', 'break', 'class', 'continue', 'def', 'del',
 'elif', 'else', 'except', 'finally', 'for', 'from', 'global',
 'if', 'import', 'in', 'is', 'lambda',
 'nonlocal', 'not', 'or', 'pass', 'raise', 'return', 'try',
 'while', 'with', 'yield']

# all soft_keywords
['_', 'case', 'match', 'type']
```