Fundamentals of the Python:

Token is nothing but smallest individual component in the program

Python Tokens Are :

1.Character SET

Set of characters which are supported by Python Language interpreter

* Python will support Unicode char SET
* Unicode = ASCII and Non ASCII
* Non Ascii = Other Language Characters [National | International]
* Range : 0 to 65535
* We can develop Language Friendly Application

2.Variables

* It is a space to store the data or
* It is named container which enable you to store the data temporally during the program execution

Java :

Syn:[modifiers]b<datatype>b<identifiers>[=value];

Actually here b is nothing a space

Python:

Syn:<identifier>=<value>[;]

```
eno=10
ename="Ramesh";
```

3.Datatypes

4.Operators

5.Identifiers

Are nothing but all the names which are declared by us for our programming requirements, Such as : Variable names, Function names, Class names.....

Rules :

1. It must starts with an alphabet or _
2. It may be in Lower | Upper | Mixed cases
3. No Limit in the length of identifiers
4. It May Have digits
   Eg: e00no; ena98me; [valid]
5. It may have a Special character [i.e. : _]
6. It should not be Python Keyword

6.Keywords

* These are nothing but reserved words
* Every keyword is having its importance in the program
* The meaning of the keyword can't be changed
* To know the keywords existed in the python then we have to use kwlist [predefined Variable of type <class 'list'>] Existed in keyword module

```
import keyword
keyword.kwlist   [shell and idle]
In script Mode --> print(keyword.kwlist)
```

```
>>> import keyword
>>> keyword.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']

>>> type(keyword.kwlist)
<class 'list'>
>>> len(keyword.kwlist)   #35
```

iskeyword() :
It returns True if the given String is a keyword else it will return False

```
Eg:   >>> keyword.iskeyword('Roja') #False
      >>>keyword.iskeyword('as')      #True
```