Python Fundamentals

Introduction

Python Character Set

A set of valid characters recognized by python. Python uses the traditional ASCII character set. The latest version recognizes the Unicode character set. The ASCII character set is a subset of the Unicode character set

Letters :– A-Z,a-z

Digits :– 0-9

Special symbols :– Special symbol available over keyboard

White spaces:– blank space,tab,carriage return,new line, form feed

Other characters:- Unicode

Indentation

- Indentation refers to the spaces applied at the beginning of a code line.
- In other programming languages the indentation in code is for readability only, where as the indentation in Python is very important.
- Python uses indentation to indicate a block of code or used in block of codes.
- E.g.1 if 3 > 2: print(“Three is greater than two!") //syntax error due to not indented

Comments

A **comment** is text that doesn't affect the outcome of a code, it is just a piece of text to let someone know what you have done in a program or what is being done in a block of code.

It is readable for programmer(a person who is writing the code)

, but ignored by python interpreter.

TYPES of comments :

i. **Single line comment**: Which begins with # (hash)sign.

i. **Multi line comment (docstring):** either write multiple line beginning with # sign or use triple quoted multiple line.

E.g.

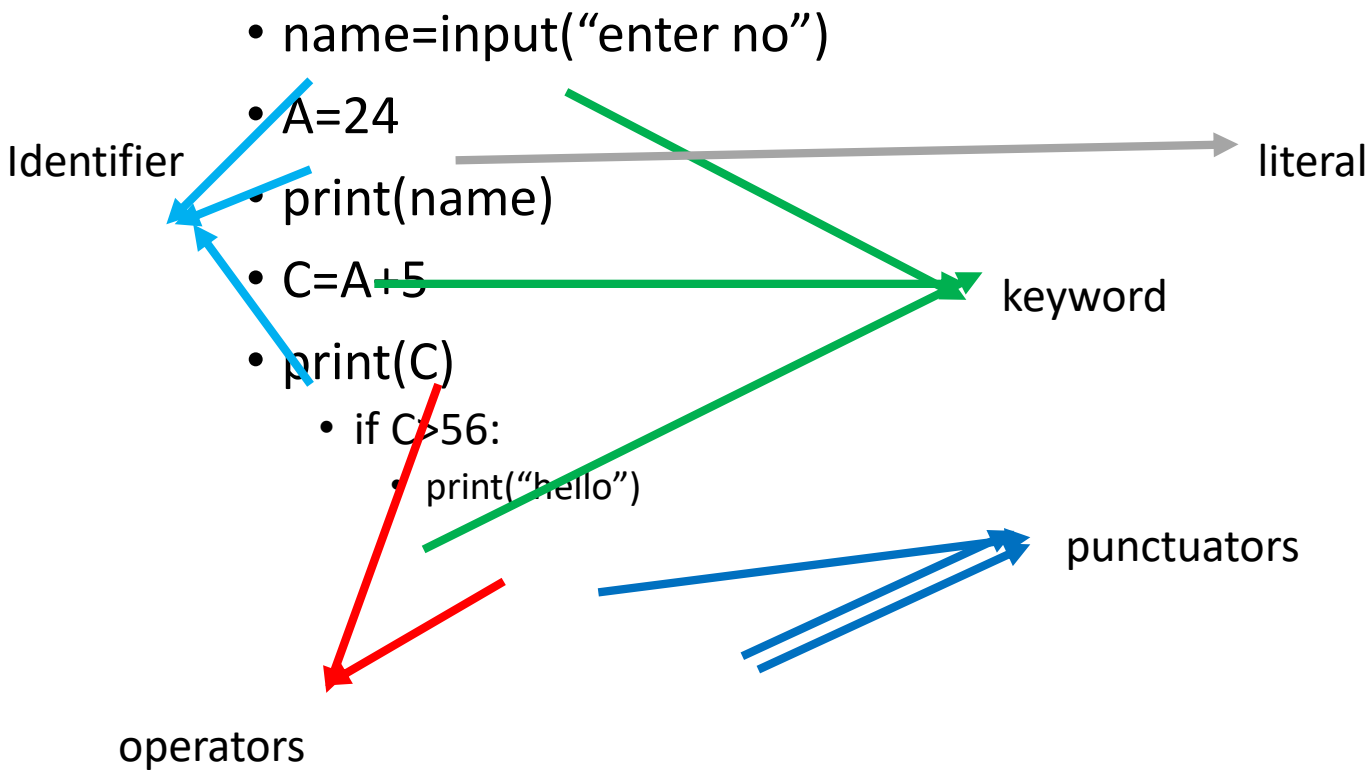
```
‘’’this is my
first

python multiline comment ‘’’
```

Token

Smallest individual unit in a program is known as token.

1. **Keywords**
2. **Identifiers**
3. **Literals**
4. **Operators**
5. **punctuators**

A Sample python program



Keywords

Reserve word / predefined words of the compiler/interpreter which can't be used as identifier.

and	exec	Not	for
as	finally	or	chr
assert	for	pass	input
break	from	print	ord
class	global	raise	len
continue	if	return	float
def	import	try	seek
del	in	while	open
elif	is	with	close
else	lambda	yield	flush
except	range	remove	nonlocal

Identifiers

A Python identifier is a name used to identify a variable, function, class, module or other object.

1. Variable names must only be a non-keyword word with no space in between.
2. Variable name must be made up of only letters,numbers and underscore(_)
3. Identifier /variable names can not begin with number.
4. An identifier starts with a letter A to Z or a to z or an underscore (_) followed by zero or more letters, underscores and digits (0 to 9).
5. Python does not allow special characters
6. Identifier must not be a keyword of Python.
7. Python is a case sensitive programming language.

Identifiers

Python is a case sensitive programming language.

Thus, **Rollnumber** and rollnumber are two different identifiers in Python.

Some valid identifiers : Mybook, file123, z2td, date_2, _no

Some invalid identifier : 2rno, break, my.book, data-cs

Literals

Literals in Python can be defined as number, text, or other data that represent values to be stored in variables. In other words these are the data items that have a fixed or constant values:

Types of Literals:

1. **String literals**
2. **Numeric literals**
3. **Boolean literals**
4. **Special literals**

String Literals

A string literals is a sequence of characters surrounded by double quotes (single,double or triple quotes)

- Single line strings
- Multiline strings

Single line strings- it must terminate in online i.e the closing quotes should be same as that of the opening quotes

e.g. name=“amita”

Multi line strings- these are string spread across multiple lines. With single and double quotes, each line other that the concluding line has an end character as \(backslash) but with triple quotes no backslash needed at the end of intermediate lines

e.g. name=“amita\
kumar”

name=“’’amita
kumar ‘’’

Literals

Escape sequence- In string we can include non-graphic characters through escape sequences

Escape Sequence	Description
****	Backslash (\)
\'	Single quote (')
\"	Double quote (")
\a	ASCII Bell (BEL)
\b	ASCII Backspace (BS)
\f	ASCII Formfeed (FF)
\n	ASCII Linefeed (LF)
\r	ASCII Carriage Return (CR)
\t	ASCII Horizontal Tab (TAB)
\v	ASCII Vertical Tab (VT)
\ooo	Character with octal value ooo
\xhh	Character with hex value hh

Numeric Literals

int(signed integers)-positive or negative whole numbers with no decimal point

- Decimal form-an integer beginning with digits 1-9 e.g. 245,540
- Octal form-An integer beginning with 0o (‘zero followed by letter o) e.g.0o35, 0o45
- Hexadecimal form- an integer beginning wit 0x ((‘zero followed by letter x).
  E.g 0x35,0x45 (digits 0-9 and A-F)

Floating point Literals (real literals)-number with fractional part
e.g.-12.5, 78.7 etc and exponent form 0.17E5, 3.3 E2 etc

Complex Number Literals-Numbers in the form of a+bi where i is the sqrt(-1) which is imaginary.
a is real part of number and b is imaginary part

Boolean Literals

A Boolean Literal has two Boolean values true and false

Special Literals: None

Python has one special literal which is None.
It is used to indicate absence of value.

Literals

Example of String Literals in Python
name = ‘Johni’ , fname =“johny”

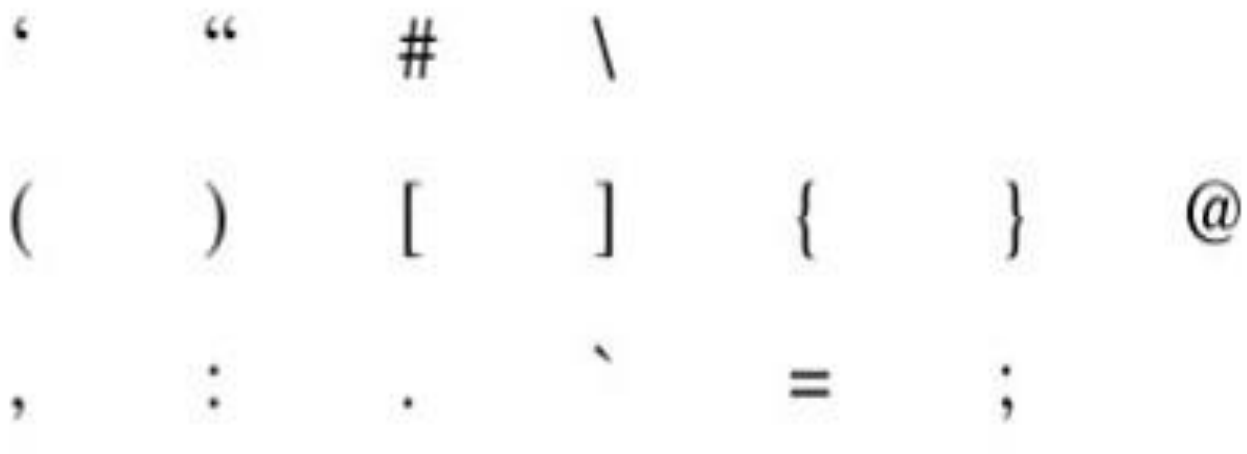
Example of Integer Literals in Python(numeric literal)
age = 22

Example of Float Literals in Python(numeric literal)
height = 6.2

Example of Special Literals in Python
name = None

Punctuators

These are the symbols that are use in programming language to organize sentence structure or to implement the grammatical and structure of a Syntax. Following are the python punctuators.



Variables

- Variable is a name given to a memory location.
- A variable can consider as a container which holds value.
- Python is a type infer language that means you don't need to specify the datatype of variable.
- Python automatically get variable datatype depending upon the value assigned to the variable.

Variables

- Syntax to declare a variable:

  variable_name= value

- **Assigning singleValues To Variable**
- name = ‘python' # String Data Type
- **num** = None # **a variable without value**

```
a = 23        # Integer
b = 6.2       # Float
sum = a + b
print (sum)
```

Variables

Multiple Assignment:

assign a single value to many variables

- **Single value to multiple variable**

  a = b = c = 1

- **Assigning value to multiple variable**

  1. a,b = 1,2 # assigning value to multiple variable

  2. a,b = b,a # value of a and b is swaped

Dynamic typing

Data type of a variable depend/change upon the value assigned to a variable on each next statement.

```
X = 25              # integer type
X = “python”        # x variable data type change to string on just next line
```

Now programmer should be aware that not to write like this:

```
Y = X / 5  # error !! String cannot be divided
```

Input and Output

print() Function In Python is used to print output on the screen.

Syntax of Print Function

print(expression/variable)

e.g.

```
print(122)
```

Output :-

122

```
print('hello India')
```

Output :-

hello India

```
print('Computer','Science')
print('Computer','Science',sep=' & ')
print('Computer','Science',sep=' & ',end='.')
```

Output :-

Computer Science
Computer & Science
Computer & Science.

Input From Keyboard

variable = input(< Prompt to display>)

e.g. name= input('What is your name:')

The input () function always returns a value of string type .

If you enter integer value it will be treated as string .

Example:

age= int(input('What is your age:'))

type(age) =>> int

type()-type function is used to find the data "type" of the variable/object

Input and Output

```
var1=‘Computer Science'
var2=‘Informatics Practices'
print(var1,' and ',var2,' )
```

Output :-

Computer Science and Informatics Practices

e.g.

```
age = int(raw_input(‘enter your age’))
percentage = float(raw_input(‘enter percentage’))
```

input() Function In Python allows a user to give input to a program from a keyboard but returns the value accordingly.

e.g.

```
age = int(input(‘enter your age’))
C = age+2 #will not produce any error
```

NOTE : input() function always enter string value in python . There is need of int(),float() function can be used for data conversion.