

# Python Fundamental

सीबीएसई पाठ्यक्रम पर आधारित  
कक्षा -11

## अध्याय -२

द्वारा:

संजीव भदौरिया

स्नातकोत्तर शिक्षक (संगणक विज्ञान )

के० वि० बाराबंकी (लखनऊ संभाग)

# Python Introduction

- यदि हमें input, Process और output सम्पन्न करना है तो हमें program लिखना ही होगा ।
- Program इंस्ट्रक्शन का ऐसा समूह होता है जिसके द्वारा प्रोसेसिंग को नियंत्रित किया जाता है ।
- अन्य शब्दों में प्रोसेसिंग का आधार program होता है ।
- इस अध्याय में हम प्रोग्रामिंग के सारे अवयव (elements) जैसे – character set, token, expressions, statements, input और output के बारे में जानेंगे ।

# Python Character Set

- Character Set किसी भी भाषा में अक्षरों अथवा संकेतों का वह संकलन है जिसे भाषा पहचान सकती है ।
- एक character set के अंतर्गत अक्षर, संकेत, व्यंजन, अंक, या अन्य कोई सांकेतिक अक्षर आसकता है ।
  - Letters: A-Z, a-z
  - Digits: 0-9
  - Special Symbols: `_`, `+`, `-`, `*`, `/`, `(`, `)`, `{`, `}` . . . इत्यादि ।
  - White Spaces: blank space, tab, carriage return, newline, formfeed इत्यादि।
  - Other characters: पाइथन सभी प्रकार के ASCII तथा UNICODE अक्षरों को प्रोसेस कर सकता है ।

# Tokens

- Token किसी भी कंप्यूटर भाषा की सबसे छोटी इकाई(smallest unit) होती है | इसे Lexical Unit भी कहते हैं | जो निम्न प्रकार के होते हैं -
  - i. Keywords
  - ii. Identifiers (Names)
  - iii. Literals
  - iv. Operators
  - v. Punctuators
- अबी इनके बारे में हम एक एक करके चर्चा करेंगे . . .

# Keywords

- **Keywords** वे शब्द होते हैं जो कम्पाइलर के लिए एक विशेष अर्थ या सन्देश भेजते हैं ।
- इन्हें विशेष कार्यों हेतु आरक्षित रखा गया है ।
- इन्हें सामान्य नामों अथवा **identifiers** के तौर पर प्रयोग नहीं कर सकते हैं ।
- पाइथन भाषा में निम्नलिखित **keywords** होते हैं ।

False	class	finally	is	return
None	continue	for	lambda	try
True	def	from	nonlocal	while
and	del	global	not	with
as	elif	if	or	yield
assert	else	import	pass	
break	except	in	raise	

# Identifiers

- ये किसी भी program के निर्माणकर्ता होते हैं तथा program के विभिन्न भागों को नामित करने के लिए इनका प्रयोग करते हैं, जैसे - variable, objects, classes, functions इत्यादि के नाम देने में |
- एक identifier अक्षरों और अंको का सार्थक मिश्रण होता है जो किसी का नाम हो सकता है |
- किसी भी identifier का पहला अक्षर कोई letter ही होगा या अंडरस्कोर ( \_ ) होगा |
- बड़े अक्षर (upper case) तथा छोटे अक्षर (lower case) अलग अलग होते हैं |
- 0 से लेकर 9 तक के सारे अंक इसका हिस्सा हो सकते हैं सिर्फ पहले अक्षर को छोड़कर |
- Identifier की लम्बाई असीमित है तथा केस सेंसिटिव है |
- किसी भी keyword को identifier के तौर पर नहीं लिया जा सकता |
- किसी भी identifier में कोई भी विशेष सांकेतिक अक्षर की नहीं लिया जा सकता सिर्फ अंडरस्कोर ( \_ ) को ही लिया जा सकता है |
- कुछ मान्य identifier –
  - Myfile, Date9\_7\_17, Z2T0Z9, \_DS, \_CHK FILE13 इत्यादि |
- कुछ अमान्य identifier –
  - DATA-REC, 29COLOR, break, My.File इत्यादि |

# Literals / Values

- Literals को प्रायः निश्चित मान (Constant Values ) कहा जाता है |
- पाइथन कई प्रकार के literals को रखने के लिए अनुमति देता है -
  - String literals - “Pankaj”
  - Numeric literals – 10, 13.5, 3+5i
  - Boolean literals – True या False
  - Special Literal *None*
  - Literal collections

# String Literals

- String Literal एक प्रकार का characters का एक क्रमिक समूह होता है जोकि कोट्स (“ “ या ‘ ‘) से घिरा रहता है |
- पाइथन में string 2 प्रकार का होता है -
  - Single लाइन string
    - Text = “Hello World” या Text = ‘Hello World’
  - Multi लाइन string
    - Text = ‘hello \ world’ या Text = “hello word”



# Numeric Literals

- Numeric values तीन प्रकार की हो सकती हैं -
  - int (signed integers)
    - Decimal Integer Literals – 10, 17, 210 इत्यादि |
    - Octal Integer Literals - 0o17, 0o217 इत्यादि |
    - Hexadecimal Integer Literals – 0x14, 0x2A4, 0xABD इत्यादि |
  - float (floating point real value)
    - Fractional Form – 2.0, 17.5 -13.5, -.00015 इत्यादि |
    - Exponent Form - -1.7E+8, .25E-4 इत्यादि |
  - complex (complex numbers)
    - 3+5i इत्यादि |

# Boolean Literals

- ये 2 ही प्रकार के मान होते हैं True या False
  - A= True
  - B=False

# Special Literals

- यह None होता जिसका मतलब होता है कुछ नहीं |
  - X = None

# Operators

- Operator वह सांकेतिक अक्षर (Symbol) होते हैं जिनको किसी identifier पर प्रयोग करते हैं तो यह compute करके गणना करने में सक्षम होते हैं | ये identifier पर किसी न किसी कार्य को इंगित करते हैं |
- अतः किसी भी ऑपरेटर को कोई न कोई operand चाहिए ताकि वह उनके ऊपर गणना कर सके |

उदाहरण :

$$c = a + b$$

यहाँ a, b, c operand हैं तथा = और + operators हैं जिनका कार्य भिन्न भिन्न है |

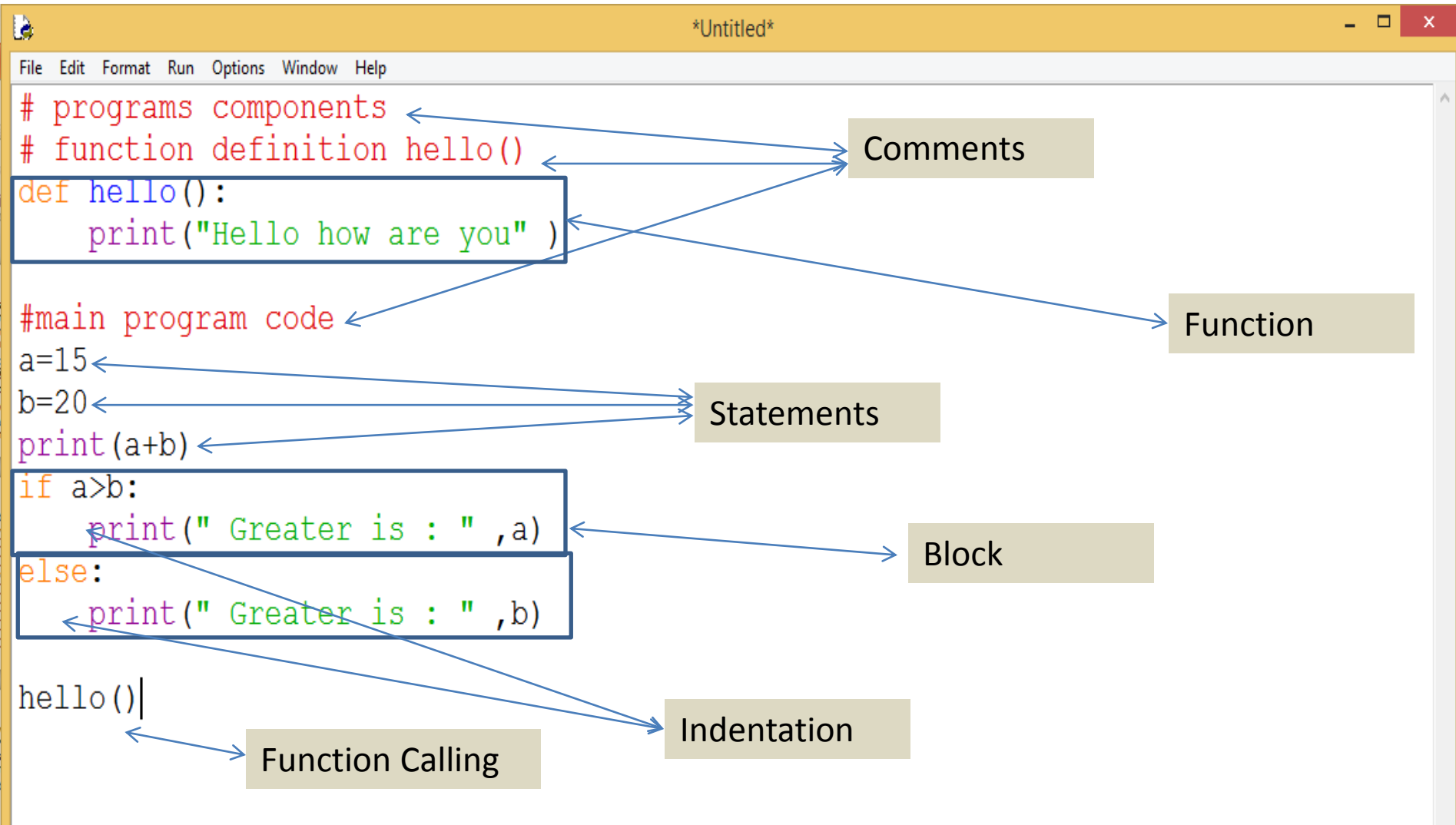
# Types of Operators

- Python में operators निम्न प्रकार के होते हैं -
  - Unary Operator
    - Unary plus (+)
    - Unary Minus (-)
    - Bitwise complement (~)
    - Logical Negation (not)
  - Binary Operator
    - Arithmetic operator (+, -, \*, /, %, \*\*, //)
    - Relational Operator(<, >, <=, >=, ==, != )
    - Logical Operator (and, or)
    - Assignment Operator (=, /=, +=, -=, \*=, %=, \*\*=, //=)
    - Bitwise Operator (& *bitwise and*, ^ *bitwise xor*, | *bitwise or*)
    - Shift operator (<< *shift left*, >> *shift right*)
    - Identity Operator (is, is not)
    - Membership Operator (in, not in)

# Punctuators

- पाइथन में punctuators का प्रयोग program के स्ट्रक्चर को बनाये रखने के लिए तथा निर्देशों (Instructions) एवं कथनों (Statements)के मध्य तालमेल बनाये रखने के लिए किया जाता है
- Python में Punctuators निम्न प्रकार के होते हैं -  
' , " , # , \ , ( , ) , [ , ] , { , } , @ , . , , : , .. ` , =

# A Python Program Structure



# A Python Program Structure

- जैसा की हमने पिछली स्लाइड में देखा की एक program में निम्न compnents होते हैं -
  - Expressions जैसे  $a+b$ ,  $a>b$  इत्यादि
  - Statements जैसे  $a=10$ ,  $c=a+b$  इत्यादि
  - Comments जैसे # से शुरू होने वाले वाक्य
  - Function जैसे def keyword से शुरू होने वाले ब्लॉक
  - Blocks और indentation जैसे if और else के ब्लॉक

इनके बारे में विस्तार से आगे पढ़ेंगे

# धन्यवाद

- कृपया हमारे और अधिक पाठ्य सामग्री हेतु इस ब्लॉग पर अनुसरित करें -
- [www.pythontrends.wordpress.com](http://www.pythontrends.wordpress.com)