

Data Handling in Python

सीबीएसई पाठ्यक्रम पर आधारित
कक्षा -11

अध्याय -3



द्वारा:
संजीव भदौरिया
स्नातकोत्तर शिक्षक (संगणक विज्ञान)
के० वि० बाराबंकी (लखनऊ संभाग)

परिचय

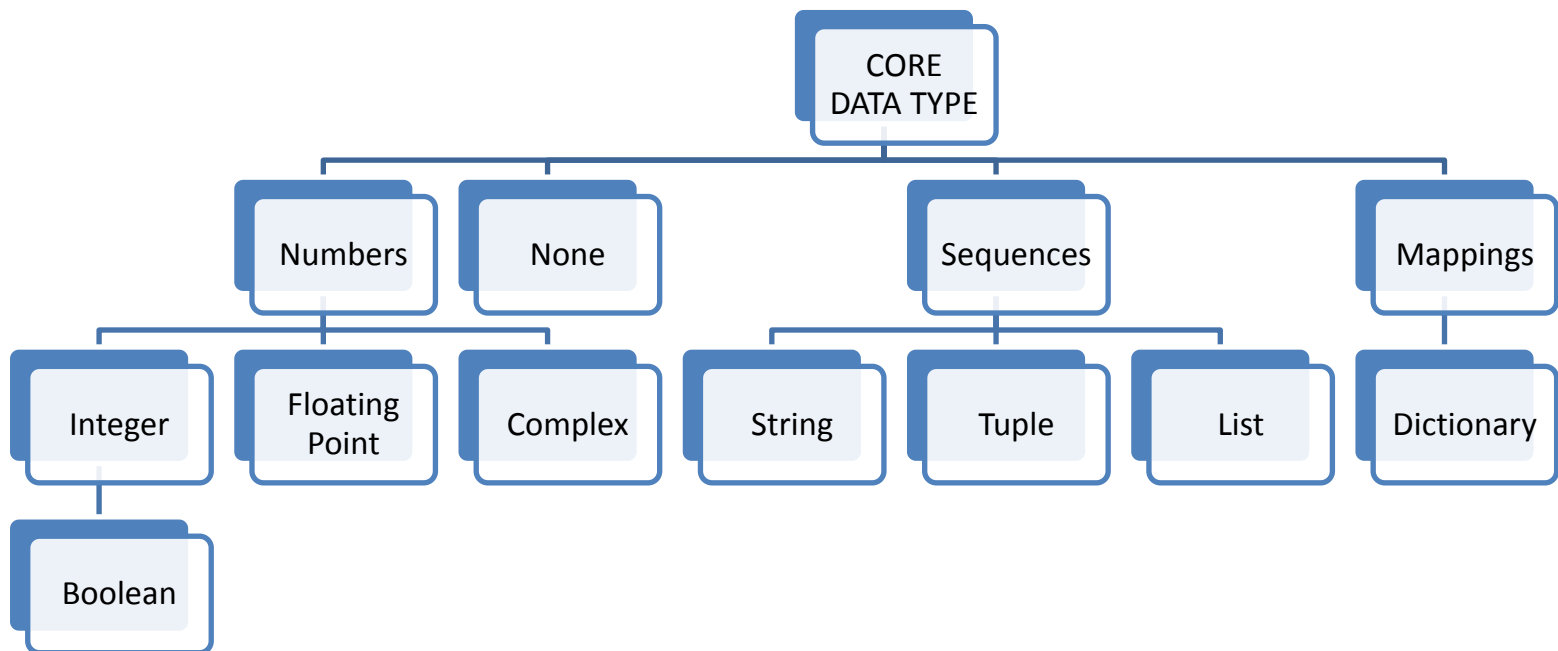
- इस अध्याय में हम data types, variables, operators, और expression के बारे में विस्तार से जानेंगे |
- पाइथन इसमें प्रयोग किये जाने वाले डाटा के प्रकार को संचालित करने के लिए data types का एक पहले से निर्धारित (predefined) संकलन रखता है
- पाइथन में आप किसी भी प्रकार का डाटा संग्रहीत कर सकते हैं |

DATA TYPES

- डाटा किसी भी प्रकार का हो सकता है जैसे - character, integer, real, string इत्यादि |
- पाइथन में “ “ के अंतर्गत लिखा गया कोई भी डाटा string डाटा होता है |
- कोई भी सम्पूर्ण संख्या integer डाटा होती है |
- कोई भी अपूर्ण संख्या अर्थात भिन्न या दशमलव (.) के साथ कोई भी संख्या real डाटा होती है |
- True या False डाटा boolean को प्रदर्शित करती है
- पाइथन निम्न core data types का समर्थन करती है
 - I. Numbers (int जैसे 10, 5) (float जैसे 3.5, 302.24) (complex जैसे 3+5i)
 - II. String (जैसे “pankaj”, ‘pankaj’, ‘a’, “a”)
 - III. List जैसे [3,4,5,“pankaj”] इसके अवयव परिवर्तनीय (Mutable) हैं |
 - IV. Tuple जैसे (3,4,5,“pankaj”) इसके अवयव अपरिवर्तनीय (immutable) हैं |
 - V. Dictionary जैसे {‘a’:1, ‘e’:2, ‘l’:3, ‘o’:4, ‘u’:5} जहाँ a,e,i,o,u key हैं और 1,2,3,4,5 क्रमशः उनकी values

CORE DATA TYPES

निम्न चित्र आपको पाइथन के Core Data Type को समझने में मदद करेगा |



Mutable और Immutable Types

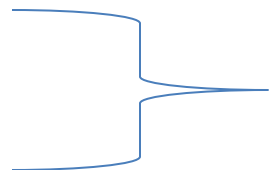
- पाइथन में डाटा objects को दो प्रकारों में वर्गीकृत किया जा सकता है -
 - Mutable (Changeable)
 - Immutable (Non-Changeable)

निम्न निर्देशों को ध्यानपूर्वक पढ़िए -

p = 10

q = p

r = 10



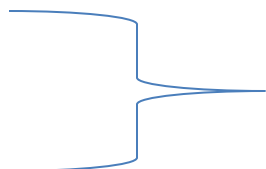
ये तीनों 10, 10, 10 को इंगित करेंगे।

अब इनमें परिवर्तन करने की कोशिश करते हैं -

p = 17

r = 7

q = 9



क्या ये values वास्तव में परिवर्तित हुईं ?

तो उत्तर है नहीं।

ऐसा इसलिए क्योंकि यहाँ values ऑब्जेक्ट हैं और p, q, r उन objects के reference name। इसको समझने के लिए हम अगली स्लाइड पर चलते हैं।

Variables और Values

यहाँ पर एक समझने वाली बहुत महत्वपूर्ण बात ये है जो अन्य भाषाओं में नहीं होती कि

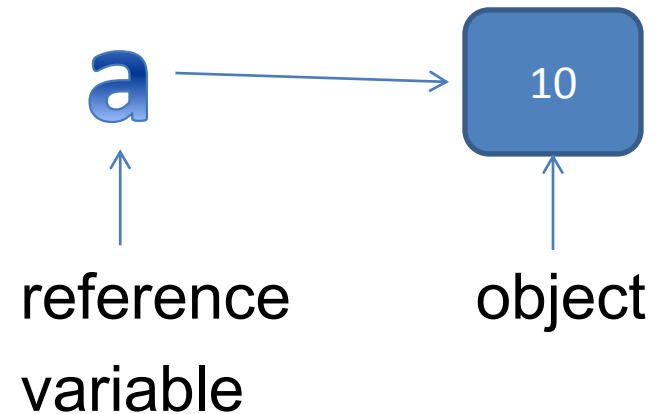
- पाइथन में values एक प्रकार के object होते हैं
- और variable name उन objects के reference name होते हैं |

इसको अब हम अच्छे से समझते हैं माना हम a variable में 10 assign करते हैं |

a = 10

यहाँ value 10 एक ऑब्जेक्ट है और a उस ऑब्जेक्ट का रिफरेन्स नाम है |

अब अगर इसी के आगे एक और variable b की वैल्यू 10 करें तो b भी इसी ऑब्जेक्ट को refer करेगा |



Variables और Values

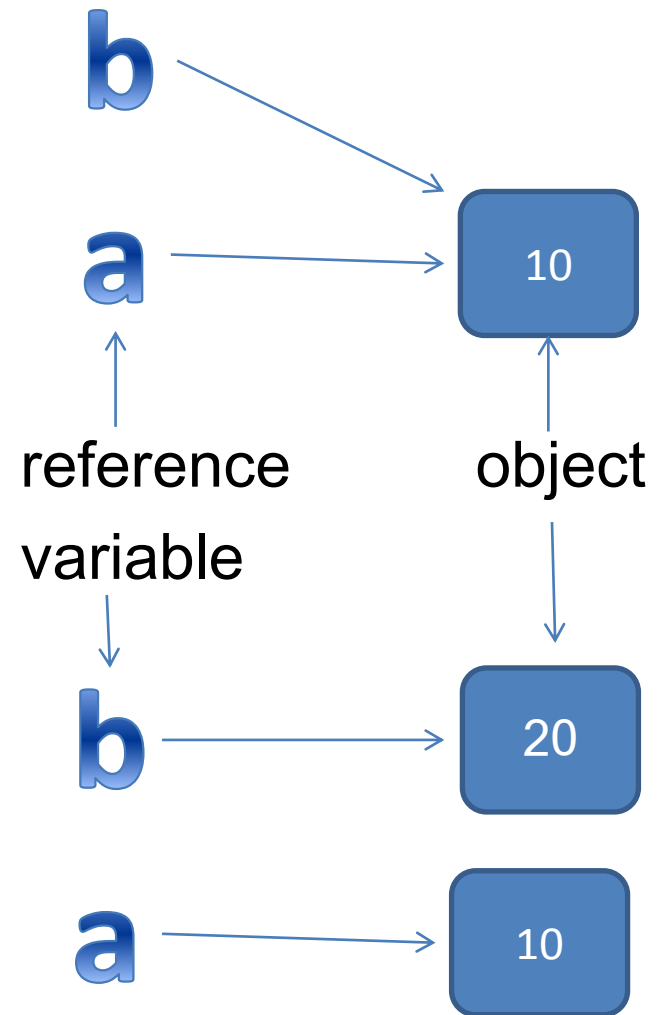
अब अगर इसी के आगे एक और variable **b** की वैल्यू 10 करें तो **b** भी इसी ऑब्जेक्ट को refer करेगा | देखिये कैसे -

b=10

यहाँ हमारे पास variable 2 हैं लेकिन उनका लोकेशन एक ही है | अब यदि हम **b** की वैल्यू में बदलाव करेंगे अर्थात्

b=20

तब एक नया ऑब्जेक्ट बनेगा 20 नयी लोकेशन के साथ और इस ऑब्जेक्ट को refer करेगा **b**



Mutable और Immutable Types

- अतः पाइथन में डाटा objects को दो प्रकारों में वर्गीकृत किया जा सकता है
 - Mutable (Changeable)
 - इसके अंतर्गत lists, dictionaries और sets आते हैं
 - Immutable (Non-Changeable)
 - इसके अंतर्गत integers, floats, Booleans, strings और tuples आते हैं

Variable Internals

The Type of an Object

- निम्न कमांड पर ध्यान दें

```
>>> a=4
```

```
>>> type(4)
```

```
<class 'int'>
```

यहाँ 4 एक ऑब्जेक्ट है और उसका class है int

```
>>> type(a)
```

```
<class 'int'>
```

यहाँ a जिस ऑब्जेक्ट को refer कर रहा है उसका class बता रहा है int

Variable Internals

The Value of an Object

- निम्न कमांड पर ध्यान दें

```
>>> print(4)
```

```
4
```

```
>>> print(a)
```

```
4
```

यहाँ print() के द्वारा value output आरहा है

The ID of an Object

- निम्न कमांड पर ध्यान दें

```
>>> id(4)
```

```
1817668720
```

```
>>> id(a)
```

```
1817668720
```

```
>>> |
```

यहाँ वैल्यू 4 और variable a की id एक जैसी ही आरही है इसका अर्थ ये है की 4 एक ऑब्जेक्ट है और a उस ऑब्जेक्ट को refer कर रहा है इसी कारण a की भी वही id आरही है।

Operators

- किसी भी operand (identifier) के ऊपर होने वाले operation (कार्य) अथवा specific task को दर्शाने वाला सांकेतिक अक्षर operator कहलाता है जैसे + , - , > , < इत्यादि |
- पाइथन में निम्न प्रकार के operators होते हैं -
 - I. Arithmetic Operator
 - II. Relation Operator
 - III. Identity Operators
 - IV. Logical Operators
 - V. Bitwise Operators
 - VI. Membership Operators

Arithmetic Operators

- Python में binary arithmetic operator निम्न प्रकार के होते हैं -
 - जोड़ने के लिए + जैसे $2+3$ का मान 5 आयेगा |
 - घटाने के लिए - जैसे $2-3$ का मान -1 आयेगा |
 - गुणा करने के लिए * जैसे $2*3$ का मान 6 आयेगा |
 - भाग देने के लिए / इसका उत्तर फ़ैक्शन में आता है जैसे $13/2$ का मान आयेगा 6.5 |
 - भागफल के लिए // इसका उत्तर भागफल जो की पूर्ण संख्या में आता है जैसे $13//2$ का मान आयेगा 6 |
 - शेषफल के लिए % इसका उत्तर शेषफल जो की पूर्ण संख्या में आता है जैसे $13/2$ का मान आयेगा 1 |
 - घात मान के लिए ** इसका उत्तर घातांक का मान आता है जैसे 2^3 का मान 8 आयेगा |

Assignment Operators

- Python में assignment operator निम्न प्रकार के होते हैं -
 - = $a=10$ करने पर a में 10 assign हो जाता है |
 - += $a+=5$ का अर्थ $a=a+5$ है
 - -= $a-=5$ का अर्थ $a=a-5$ है
 - *= $a*=5$ का अर्थ $a=a*5$ है
 - /= $a/=5$ का अर्थ $a=a/5$ है
 - //= $a//=5$ का अर्थ $a=a//5$ है
 - %= $a%=5$ का अर्थ $a=a\%5$ है
 - **= $a**=5$ का अर्थ $a=a**5$ है

Relational Operators

- Python में Relational operator का प्रयोग समानता को check करने के लिए किया जाता है तथा ये expression True अथवा False उत्तर देते हैं Relational Operator निम्न प्रकार के होते हैं -

- | | | |
|------|---------------------------|---------------|
| • < | Less Than | जैसे $a < b$ |
| • > | Greater Than | जैसे $a > b$ |
| • <= | Less Than and Equal to | जैसे $a <= b$ |
| • >= | Greater Than and Equal to | जैसे $a >= b$ |
| • == | Equal to | जैसे $a == b$ |
| • != | not Equal to | जैसे $a != b$ |

Identity Operators

- Python में Identity operator का प्रयोग भी समानता को check करने के लिए किया जाता है तथा ये expression True अथवा False उत्तर देते हैं Identity Operator निम्न प्रकार के होते हैं -
 - “is” operator यदि a=5 और b=5 तो a is b का उत्तर True होगा
 - “is not” operator यदि a=5 और b=5 तो a is not b का उत्तर False होगा
- परन्तु Relational Operator (==) और Identity operator (is) में असमानता है strings के केस में जिन्हें हम बाद के अध्याय में देखेंगे |

Logical Operators

- Python में binary logical operator 2 प्रकार के हैं -
 - or operator
 - » यदि $a = \text{True}$ और $b = \text{False}$ तो $a \text{ or } b$ *True* return करेगा |
 - and operator
 - » यदि $a = \text{True}$ और $b = \text{False}$ तो $a \text{ and } b$ *False* return करेगा |
- Python में Unary logical operator एक ही प्रकार का है –
 - not operator
 - यदि $a = \text{True}$ तो $\text{not } a$ *False* return करेगा |

Operator Associativity

- Python में किसी स्टेटमेंट में या expression में एक साथ कई operator लगे हों तो operator associativity हमेशा left-to-right रहेगी |

```
>>> 7*8/5//2
5.0
```

- इस उदहारण में पहले $7*8$ होगा अर्थात 56, फिर इस 56 में 5 का भाग लगेगा अर्थात 11.2, तब 11.2 में 2 का पुनः // के द्वारा भाग लगेगा और उत्तर आयेगा |

सिर्फ ****** के सन्दर्भ में right-to-left associativity रहेगी |

```
>>> 3**3**2
19683
```

इस उदहारण में $3**(3**2)$ के अनुसार गणना होगी |

Expressions

- Python में expression निम्न प्रकार के होते हैं -
 - Arithmetic Expressions जैसे $a+b$, $5-4$ इत्यादि |
 - Relational Expressions जैसे $a>b$, $a==b$ इत्यादि |
 - Logical Expressions जैसे $a>b$ and $a>c$, a or b इत्यादि |
 - String Expressions जैसे “Pankaj” + “Kumar” इत्यादि |

Type Casting

- जैसा की हम जानते हैं कि यदि पाइथन के expression में datatypes mixed रूप में रहते हैं तो पाइथन कुछ operand का datatype आंतरिक रूप से बदल देता है | इस प्रक्रिया को implicit type conversion कहते हैं |
- एक और विकल्प हमारे पास explicit type conversion के रूप में हमारे पास होता है जिसके लिए हम निम्न प्रारूप अपनाते हैं –
- <datatype> (identifier)

जैसे – a="4"

b=int(a)

अथवा यदि हमारे पास a=5 और b=10.5 है तो हम a को float में बदल सकते हैं | d=float(a) करके |

– पाइथन में data conversion फंक्शन निम्न हैं

(1) int () (2) float() (3) complex() (4) str()

(5) bool()

Working with math Module of Python

- Math से सम्बंधित समस्त कार्यों के लिए पाइथन math module प्रदान करता है जिसको प्रयोग करने के लिए हमें अपने program में सबसे पहले निम्न निर्देश लिखना होता है - **import math**

```
import math
```

```
a=25
```

```
print(math.sqrt(a))
```

इस program का उत्तर होगा –

5.0

```
>>> import math
```

```
>>> dir (math)
```

```
['__doc__', '__loader__', '__name__', '__package__', '__spec__', 'acos', 'acosh', 'asin', 'asinh', 'atan', 'atan2', 'atanh', 'ceil', 'copysign', 'cos', 'cosh', 'degrees', 'e', 'erf', 'erfc', 'exp', 'expm1', 'fabs', 'factorial', 'floor', 'fmod', 'frexp', 'fsum', 'gamma', 'gcd', 'hypot', 'inf', 'isclose', 'isfinite', 'isinf', 'isnan', 'ldexp', 'lgamma', 'log', 'log10', 'log1p', 'log2', 'modf', 'nan', 'pi', 'pow', 'radians', 'sin', 'sinh', 'sqrt', 'tan', 'tanh', 'tau', 'trunc']
```

```
>>>
```

किसी module में ये पता करना हो की कितने फंक्शन हैं तो निम्न कमांड दीजिये -

```
>>>dir (math)
```

Python में Input लेना

- Python में input लेने के लिए input () फंक्शन होता है जो की string के रूप में input लेता है फिर उसको आवश्यकता के अनुसार type cast करते हैं | जैसे किसी बेलन (cylinder) का आयतन (Volume) निकलने हेतु program निम्न है -

```
#Program to calculate vloume of a cone
radius=int(input("Enter the radius of the Cylinder : "))
height=int(input("Enter the height of the Cylinder : "))
volume = 3.14* radius*radius*height
print("The Volume of the cylinder is : ",volume)
|
```

- इसका output निम्नवत आएगा |

```
>>>
RESTART: C:/Users/KVBBKServer/AppData/Local/Programs/Python/Python36/VolOfCyl
inder.py
Enter the radius of the Cylinder : 10
Enter the height of the Cylinder : 5
The Volume of the cylinder is : 1570.0
>>> |
```

धन्यवाद

- अधिक सामग्री के लिए निम्न लिंक पर क्लिक करें –
- www.pythontrends.wordpress.com