Python Programming

Dr. Raj Gaurang Tiwari

Department of Computer Science and Engineering
Chitkara University, Punjab

Contents

CHITKARA UNIVERSITY

- Data Types and Values
- Expression
- Operators
- Variables

Data Types and Values

- Basic data types
  - int

    1, 2, -1, -2
  - float

    1.0, 2.3, -5.7
  - string

    ‘apple’, “banana”
  - Boolean

    True, False

Expressions

CHITKARA UNIVERSITY

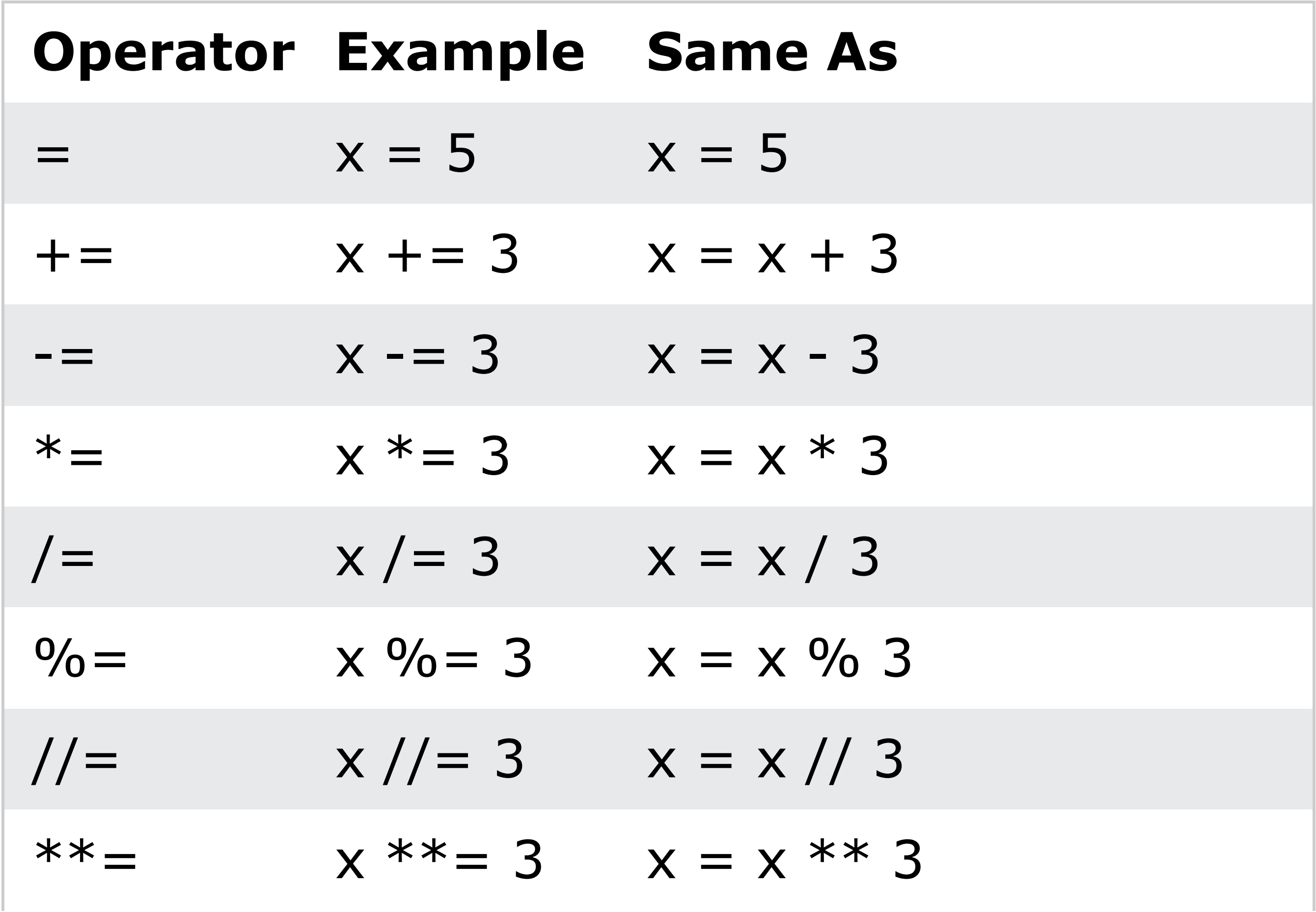
- Combine values using operators
- Know which operators are valid for which data types
- Expressions can be evaluated
- They evaluate to a single value
- Order is important during evaluation
  - known as precedence
- Operators: +, -, *, / , **, //, %

Operators

Python divides the operators in the following groups:

- Arithmetic operators
- Assignment operators
- Comparison operators
- Logical operators
- Identity operators
- Membership operators
- Bitwise operators

Python Assignment Operators



Operator	Example	Same As
=	x = 5	x = 5
+=	x += 3	x = x + 3
-=	x -= 3	x = x - 3
*=	x *= 3	x = x * 3
/=	x /= 3	x = x / 3
%=	x %= 3	x = x % 3
//=	x //= 3	x = x // 3
**=	x **= 3	x = x ** 3

Arithmetic Operators

Operator	Name	Example	
+	Addition	x + y	
-	Subtraction	x - y	
*	Multiplication	x * y	
/	Division	x / y	
%	Modulus	x % y	
**	Exponentiation	x ** y	
//	Floor division	x // y	

Comparison Operators

CHITKARA UNIVERSITY

Comparison operators are used to compare two values:

Operator	Name	Example
==	Equal	x == y
!=	Not equal	x != y
>	Greater than	x > y
<	Less than	x < y
>=	Greater than or equal to	x >= y
<=	Less than or equal to	x <= y

Logical Operators

Logical operators are used to combine conditional statements:

Operator	Description	Example
and	Returns True if both statements are true	x < 5 and x < 10
or	Returns True if one of the statements is true	x < 5 or x < 4
not	Reverse the result, returns False if the result is true	not(x < 5 and x < 10)

Identity Operators

Operator	Description	Example
is	Returns True if both variables are the same object	x is y
is not	Returns True if both variables are not the same object	x is not y

Membership Operators

Operator	Description	Example
in	Returns True if a sequence with the specified value is present in the object	x in y
not in	Returns True if a sequence with the specified value is not present in the object	x not in y

Expressions

- 2 + 3 * 5 - 8 / 4 = ?
- (2 + 3)*5 - (8/4)
- ‘Alice’ + ‘Bob’
- ‘Alice’ * 5
- 5 > 3 = ?
- (5 > 3) and (4 > 5) = ?

Variables

- Refer to values by names
- Variables store/point to values
- Assignment
    - x = 3
    - y = x + 5
- Left side of is a variable name
- Right side is an expression

Variable Names

CHITKARA
UNIVERSITY

- A single word - no spaces
- Can contain letters, digits, underscore
- Cannot start with a number
- Case-sensitive
- Names describe what they contain

balance
current_balance
‘hello’
$abc
_abc
_hello

currentBalance
1current
“hello”
%abc
a_%^sh _
a123

Variables and Constant in Python

- A variable is a named location used to store data in the memory.

```
number = 10
number = 1.1

website = "apple.com"
print(website)
```

Assigning multiple values to multiple variables

```
a, b, c = 5, 3.2, "Hello"
print (a)
print (b)
print (c)
```

Variables and Constant in Python

same value to multiple variables

```
x = y = z = "same"
print (x)
print (y)
print (z)
```

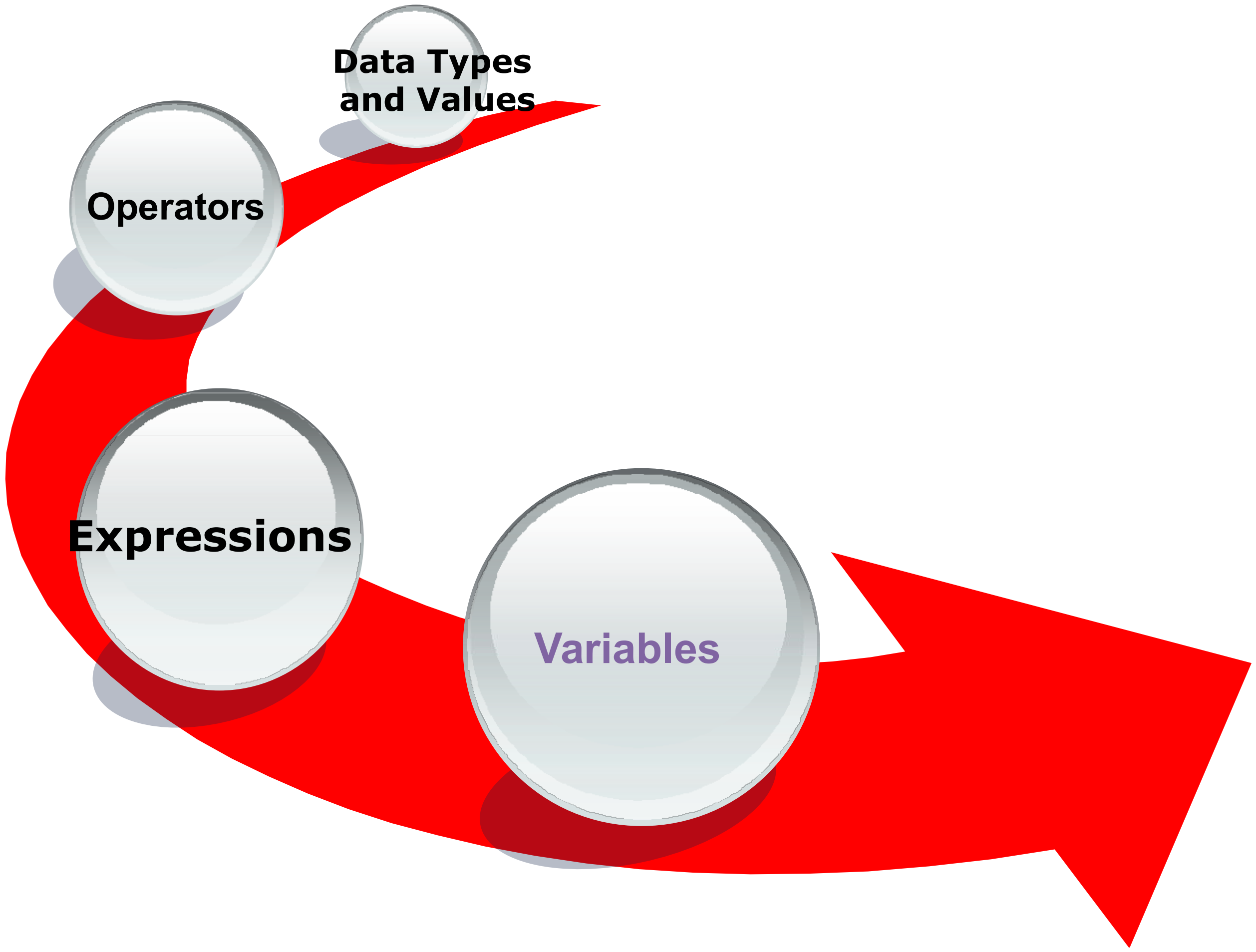
A Code Sample

```
x = 34 - 23                 # A comment.
y = “Hello”                 # Another one.
z = 3.45
if z == 3.45 or y == “Hello”:
    x = x + 1
    y = y + “ World”        # String concat.
print x
print y
```

Enough to Understand the Code

- Indentation matters to code meaning
  - Block structure indicated by indentation
- First assignment to a variable creates it
  - Variable types don't need to be declared.
  - Python figures out the variable types on its own.
- Assignment is `=` and comparison is `==`
- For numbers `+ - * / %` are as expected
  - Special use of `+` for string concatenation
- Logical operators are words (`and, or, not`) *not* symbols
- The basic printing command is `print`

Summery



Thank You !