PRG1

WEEK 2

NGEE ANN
SCHOOL OF INFOCOMM TECHNOLOGY

Operators and Interactivity

Programming I (PRG1)
Diploma in Information Technology
Diploma in Financial Informatics
Diploma in Information Security & Forensics
Year 1 (2018/19), Semester 1

Topics

1. More on Data Types

2. Introduction to Strings

3. Operators

4. Program Interactivity

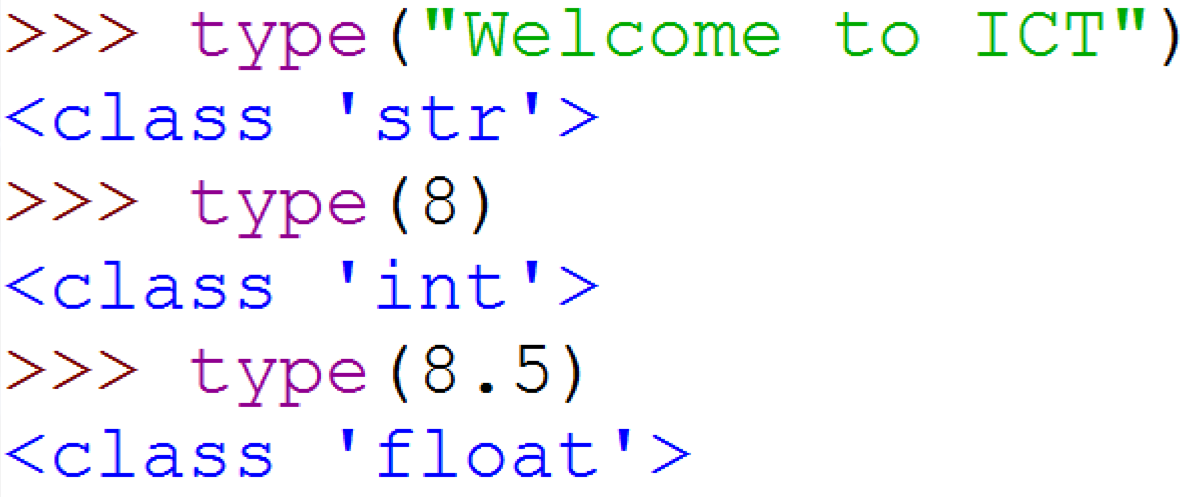
Values and data types

- A value is one of the fundamental things — like a letter or a number — that a program manipulates.
- These values are classified into different data types

Data Type	**Description**	**Example**
int	integer	46
float	floating point or decimal numbers	123.45
bool	boolean	True or False
str	sequence of characters	"A", 'Hello'

- Python has a function called *type* which can tell you the type of a value.

Values and data types



```
>>> type("Welcome to ICT")
<class 'str'>
>>> type(8)
<class 'int'>
>>> type(8.5)
<class 'float'>
```

- Not surprisingly, *strings* belong to the class str and *integers* belong to the class int and *float* belongs to float
- At this stage, you can treat the words class and type interchangeably.

Activity 1

Guess the type:

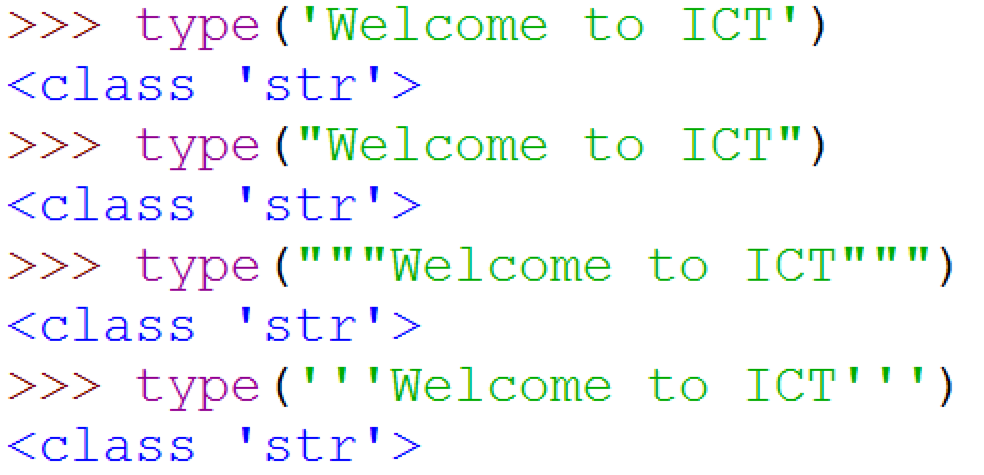
- >>> type(18.3)

- >>> type("18")

- >>> type("18.3")

More on Strings

– Strings in Python can be enclosed in either single quotes (') or double quotes ("), or three of each (''') or (""")



```
>>> type('Welcome to ICT')
<class 'str'>
>>> type("Welcome to ICT")
<class 'str'>
>>> type("""Welcome to ICT""")
<class 'str'>
>>> type('''Welcome to ICT''')
<class 'str'>
```

Operators

- Operators are symbols in programming languages to perform processing actions
  - Arithmetic Operators
  - Assignment Operator
  - Relational Operators – Conditional Operators
  - Logical Operators – Conditional Operators

Activity 2

- In what order would you solve this mathematical equation?

$$E_{wg} = 6.112 \text{ x } e^{(17.67 \text{x} T_w)/(T_w+243.5)}$$

Where,

T_w = Wet Bulb Temperature
e = Vapor Pressure

Arithmetic Operators

Operators		Example	Given y=5; z=3
+	addition	x = y + z	
-	subtraction	x = y – z	
*	multiplication	x = y * z	
/	division	x = y / z	
%	modulus/remainder	x = y % z	
//	floor division	x = y // z	
**	exponentiation	x = y ** z	

Order of Arithmetic Operators

- When more than one operator appears in an expression, the order of evaluation depends on the rules of precedence.
- Order of precedence (highest to lowest) is:

Operator	Description	
( )	Parentheses (grouping)	
**	Exponentiation (raise to a power)	*right to left*
* / // %	Multiplication, division, floor division, modulus	*left to right*
+ -	Addition, subtraction	*left to right*
=	Assignment	

Mathematical Formulae to Python

	Mathematical formulae	Python Statement
1.	$y = ax + b$	
2.	$y = (a + b)(a - b)$	
3.	$y = 2[(a + b)(a - b)] - x$	
4.	$y = 1 - \frac{2a}{3b}$	
5.	$a = -b$	

Assignment Operator =

- Assigns value to variable
- Assignment is from right to left
  - E.g.
  - What is the value of k after each line?

```
k = 6
k = k + 1
```

  - adds 1 to value of variable **k**
  - associates new value to the variable **k**

Assignment Statements

- Simple Assignment
- *<variable> = <expr>*
  variable is an identifier, expr is an expression
- The expression on the RHS is evaluated to produce a value which is then associated with the variable named on the LHS.

Examples of Assignment Statement

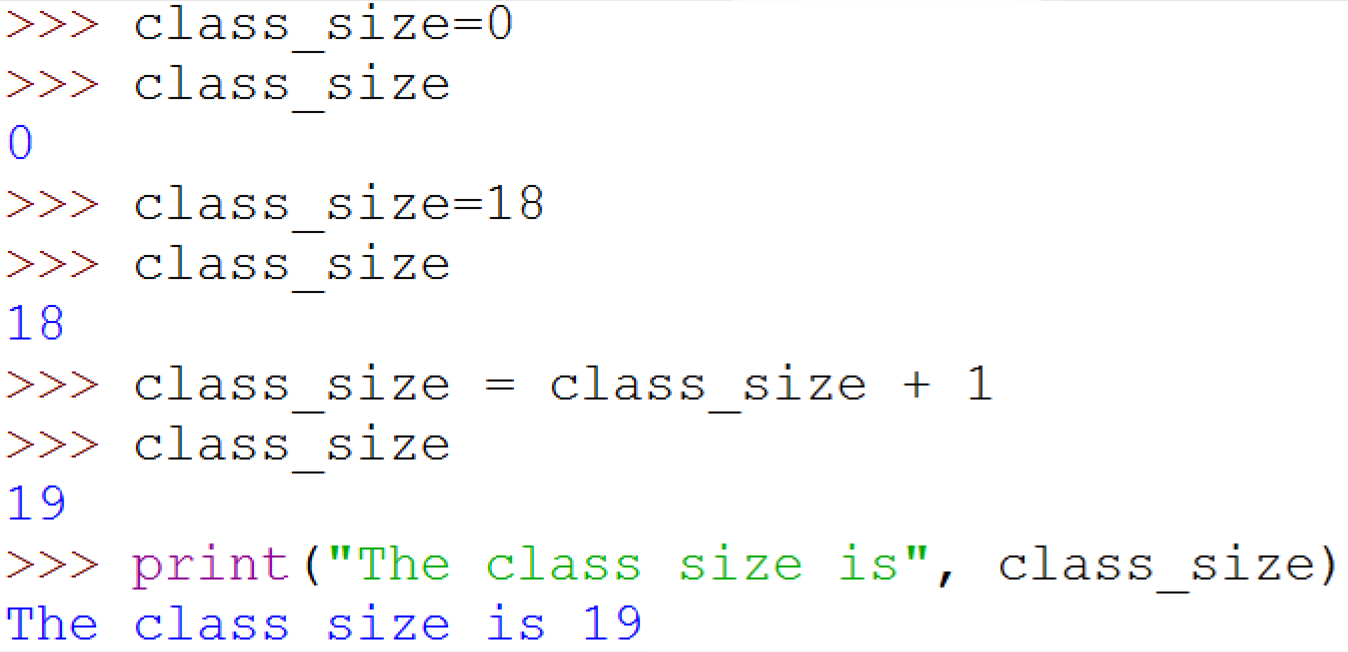
```
x = 3.9 * y * (1-z)

fahrenheit = 9/5 * celsius + 32

x = 5
```

Assignment Statements

– Variables can be reassigned as many times as you need during the execution of your program!



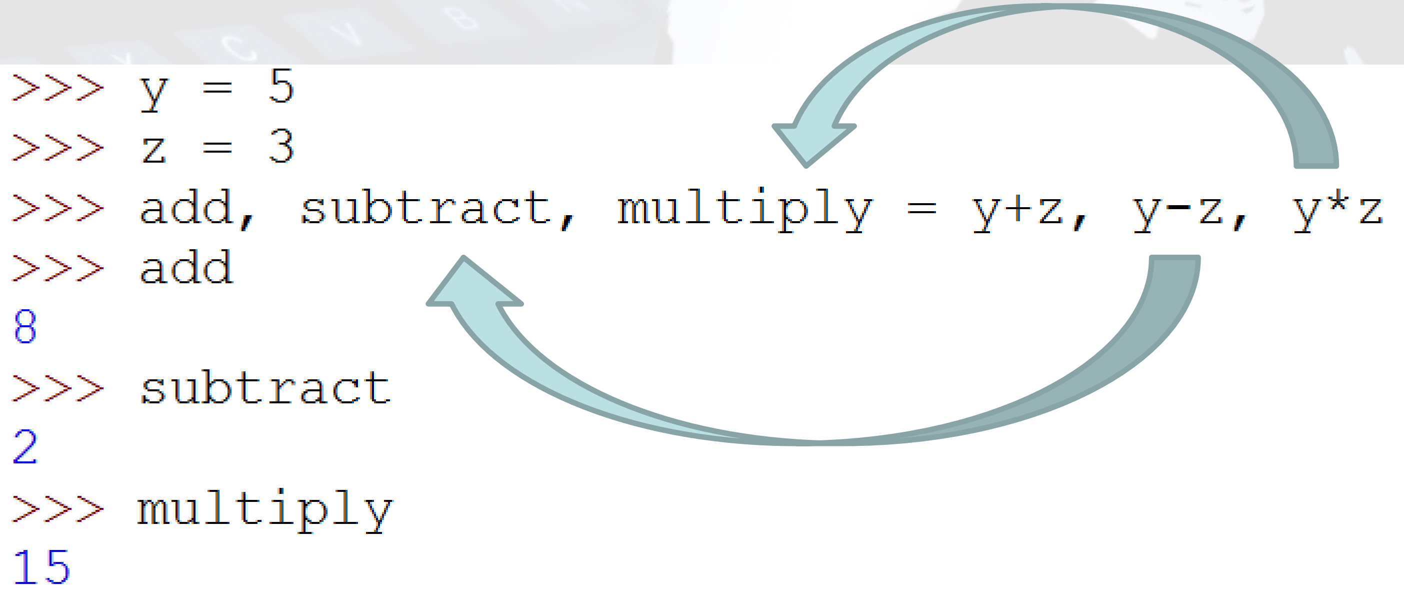
```
>>> class_size=0
>>> class_size
0
>>> class_size=18
>>> class_size
18
>>> class_size = class_size + 1
>>> class_size
19
>>> print("The class size is", class_size)
The class size is 19
```

Simultaneous Assignment

– Several values can be calculated at the same time

– <var>, <var>, … = <expr>, <expr>, …

– Evaluate the expressions in the RHS and assign them to the variables on the LHS



```
>>> y = 5
>>> z = 3
>>> add, subtract, multiply = y+z, y-z, y*z
>>> add
8
>>> subtract
2
>>> multiply
15
```

Activity

– Given x = 5, y = 3, how would you swap the values in x and y?

Relational Operators

– *Relational* operators compares the two operands and produce a *Boolean* value of either *True* or *False*.

Operator	Description
==	equal
!=	not equal
>	greater than
>=	greater than or equal
<	less than
<=	less than or equal

Activity 3

- Assume x = 4, y = 3, z = 2

Boolean Expression	Result
x < y + z	
y == 2 * x + 3	
z <= x - y	
z > x % z	
x != y	

Logical Operators

- *Logical* operators allow us to build more complex Boolean expressions from simpler Boolean expressions.

Logical Operator	Description
and	produces True only if both Boolean expressions are true
or	produces True if either of the Boolean expressions is true
not	Negates a Boolean value

Order of Operators

Operator Precedence, from highest to lowest:

Operator	Description	
( )	Parentheses (grouping)	
**	Exponentiation (raise to a power)	*right to left*
* / // %	Multiplication, division, floor division, modulus	*left to right*
+ -	Addition, subtraction	*left to right*
<, <=, >, >=, !=, ==	Relational operators	
not x	Boolean NOT	
and	Boolean AND	
or	Boolean OR	

Operators - Example

- Assume x = 1, y = 4, z = 14

Boolean Expression	Result
x <= 1 and y == 3	
x <= 1 or y == 3	
not (x > 1)	
not x > 1	
not (x <= 1 or y == 3)	
x <= 1 or y > 1 and z < 1	
(x <= 1 or y > 1) and z < 1	

Type conversion

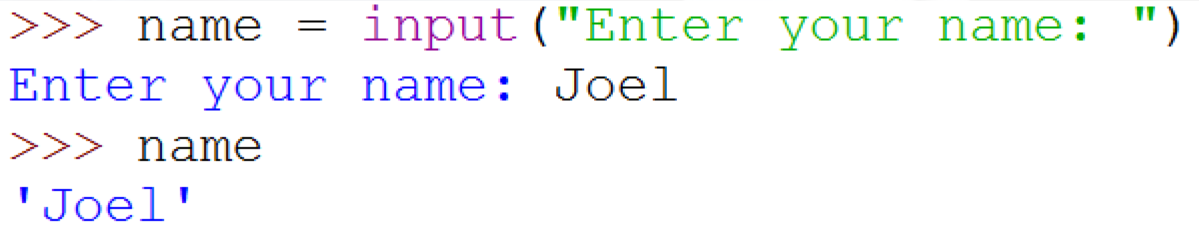
- int(), float(), str(), and bool() convert to integer, floating point, string, and boolean (True or False) types, respectively

Example	Output
print(4.0/2.0)	2.0
print(int(4)/int(2))	2.0
print(float(4)/float(2))	2.0
print(int(3.1415926))	3
print(str(3.1415926))	3.1415926
print(bool(1))	True
print(bool(0))	False

Program Interactivity

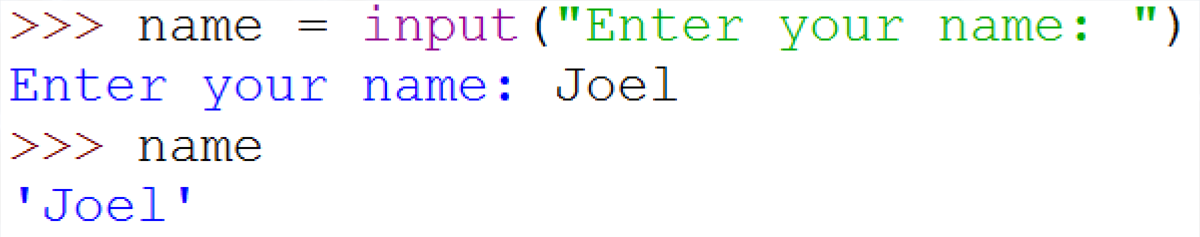
- There is a built-in function in Python for getting input from the user.
- The purpose of an input statement is to get input from the user and store it into a variable.

<variable> = input(<prompt>)



```
>>> name = input("Enter your name: ")
Enter your name: Joel
>>> name
'Joel'
```

Assigning Input



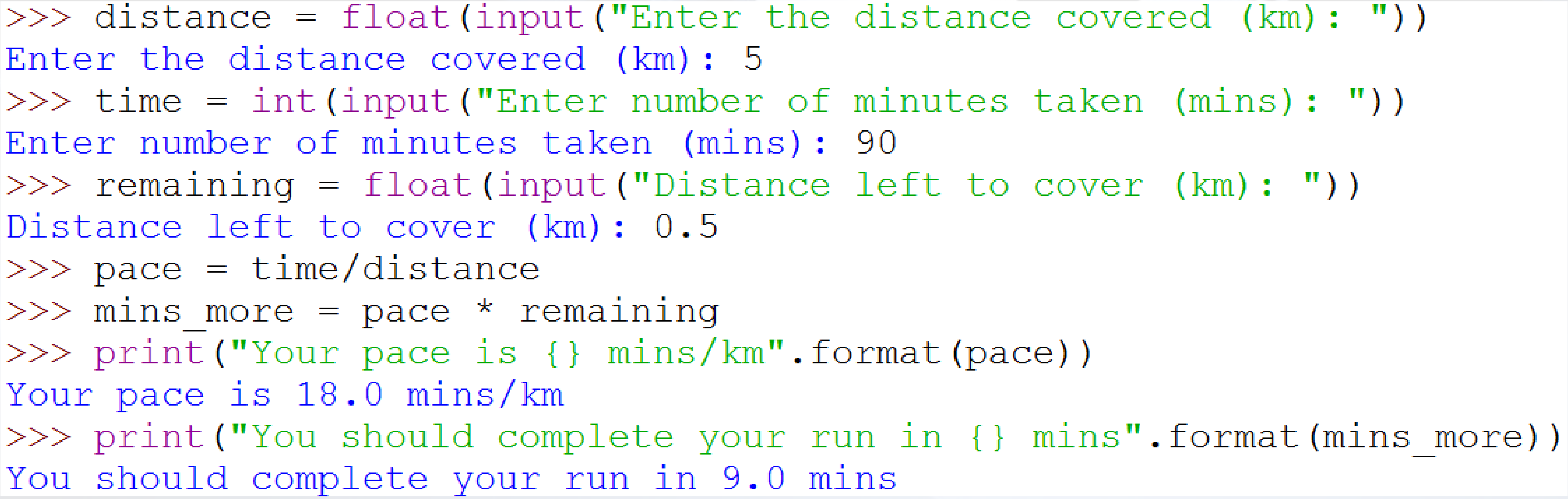
```
>>> name = input("Enter your name: ")
Enter your name: Joel
>>> name
'Joel'
```

- First the prompt is printed
- The input part waits for the user to enter a value and press <enter>
- The expression that was entered is captured as a string of characters. The value is assigned to the variable.

Assigning Input

- You **need** to convert the string to int or float in order to use the numeric value.

Pace (min/km)= Time (mins) / Distance (km)



```
>>> distance = float(input("Enter the distance covered (km): "))
Enter the distance covered (km): 5
>>> time = int(input("Enter number of minutes taken (mins): "))
Enter number of minutes taken (mins): 90
>>> remaining = float(input("Distance left to cover (km): "))
Distance left to cover (km): 0.5
>>> pace = time/distance
>>> mins_more = pace * remaining
>>> print("Your pace is {} mins/km".format(pace))
Your pace is 18.0 mins/km
>>> print("You should complete your run in {} mins".format(mins_more))
You should complete your run in 9.0 mins
```

Reading Reference

- How to Think Like a Computer Scientist: Learning with Python 3
  - Chapter 2

    http://openbookproject.net/thinkcs/python/english3e/variables_expressions_statements.html

Summary

- Type function to find out the data type
- Operators
- Program Interactivity