



Curtin University



# Python Objects

ISYS2001, School of Marketing and Management

## **ELECTRONIC WARNING NOTICE FOR COPYRIGHT STATUTORY LICENCES**

### **WARNING**

This material has been reproduced and communicated to you by or on behalf of **Curtin University** in accordance with section 113P of the *Copyright Act 1968 (the Act)*

The material in this communication may be subject to copyright under the Act. Any further reproduction or communication of this material by you may be the subject of copyright protection under the Act.

Do not remove this notice.

I acknowledge the traditional custodians of the land on which I work and live, and recognise their continuing connection to land, water and community. I pay respect to elders past, present and emerging.

# Today

- Break a problems down into simple actions
- Actions carry out simple tasks
- A set of actions is called an algorithm
- Algorithms expressed as Pseudocode
- Coding is translating an algorithm to a program
- Six operations a computer can do

“The biggest mistake I see new programmers make is focusing on learning syntax instead of learning how to solve problems.”

— V. Anton Spraul, Think Like a Programmer: An Introduction to Creative Problem solving

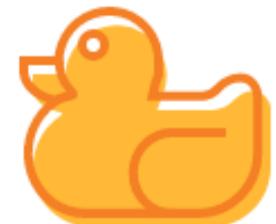
# Problem Solving Methodology

- State the problem clearly
- Describe the input and output
- Work a simple example by hand
- Develop an algorithm (and convert to Python)
- Test solution with a variety of data

“If you can’t explain something in simple terms, you don’t understand it.”

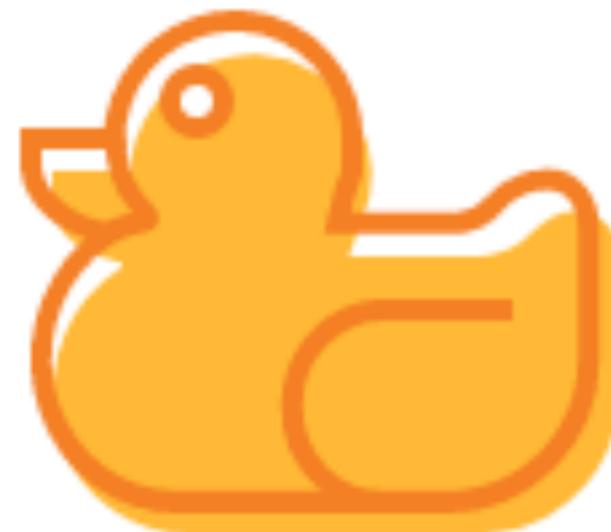
— Richard Feynman

Can I help?



# Rubber Duck

Can I help?



# Psuedocode

- List the steps
- Use short phrases (make it easier to convert)
- No Syntax or rules
- Not a language → (not compiled or executed)
- Communication tool
- Logic of the solution
- Translate into a programming language

# Recall

Pick up dictionary

Open to middle of dictionary

Look at page

If word is on page

    Read definition

Else if word is earlier in book

    Open to middle of left half of book

    Go back to line 3

Else if word is later in book

    Open to middle of right half of book

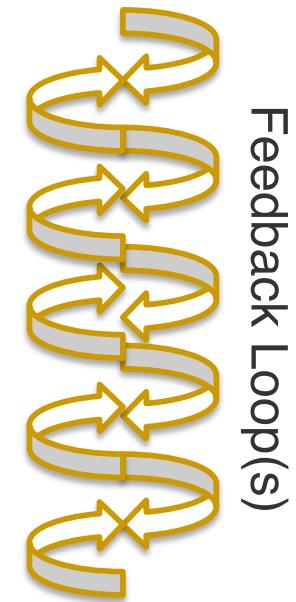
    Go back to line 3

Else

    Quit

# Problem Solving Methodology

- State the problem clearly
- Describe the input and output
- Work a simple example by hand
- Develop an algorithm (and convert to Python)
- Test solution with a variety of data



Computer can...

Store Values

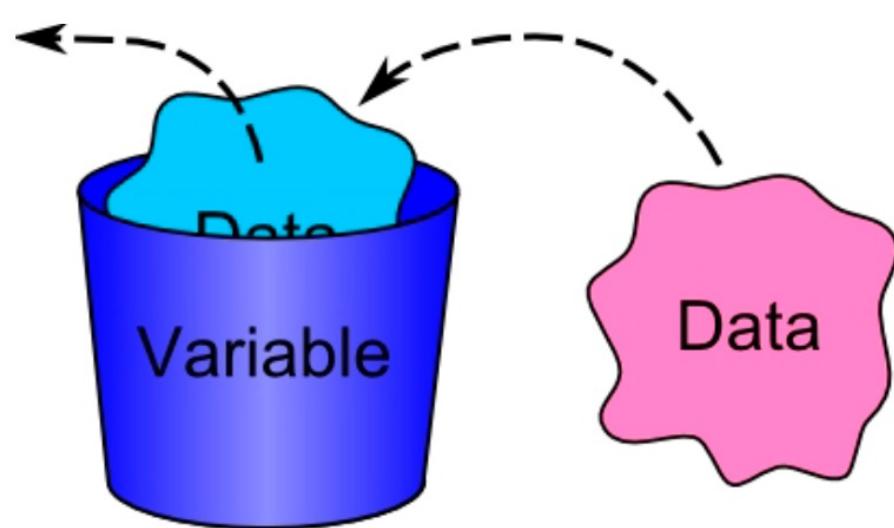
Perform Operations

# What are values?

- Single Value
  - Integer
  - Float
  - Character
  - Boolean
- Data Structure
  - Lists
  - Strings
  - Files

<b>Data Type</b>	<b>Python Expressions</b>	<b>Description</b>	<b>Example</b>
integer	int	A whole number	143267
decimal	float	A floating point number	3.13159
string	str	Set of characters in quotes	"Hello"
boolean	bool	anything evaluates true/false	True
List	list	An ordered list of values	[1,4,5]

A ***variable*** is the name given to a part of the computers memory, designed to store a particular data item.



# Variable Names

- Rules

- Only letters, numbers, and the underscore character (\_)

- No spaces

- No quotes

- Can't start with a number

- Can't be a keyword import

- Conventions

- variable names should be meaningful

- separate words with underscores (snake\_case)

# Only Six Operations

Receive Information

Output Information

Perform Arithmetic

Assign a value to a variable

Compare to variables

Repeat a group of actions.

```
input("What is your name?")
```

```
print("Hello, world")
```

Operation	Description	Example	Result
+	Addition	3 + 2.2	5.2
-	Subtraction	5 - 2	3
*	Multiplication	3 * 8	24
**	Raise to the power of	3 ** 2	9
/	Division	5/2	2.5
//	Integer Division	5//2	2
%	Modulo	5%2	1

```
name = input("What is your name?")
```

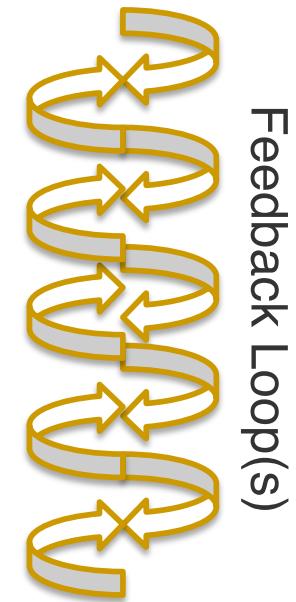
```
if first_number > second_number:  
    print(first_number)  
  
else:  
    print(second_number)
```

Operation	Description	Example	Result
<	Less than	3 < 2	False
>	Greater than	3 > 2	True
<=	Less than or equal	3 <= 2	False
>=	greater than or equal	3 >= 2	True
==	equal to	3 == 2	False
!=	not equal to	3 != 2	True
not	not	not(3 == 2)	True

```
for number in [10,9,8,7,6,5,4,3,2,1]:  
    print(number)  
print("Blast Off!")
```

# Problem Solving Methodology

- State the problem clearly
- Describe the input and output
- Work a simple example by hand
- Develop an algorithm (and convert to Python)
- Test solution with a variety of data



Make the steps in your pseudocode to be one of the six operations, then translating from pseudocode to a programming language becomes simple.

# Can you

- Break a problems down into simple actions
- Actions carry out simple tasks
- A set of actions is called an algorithm
- Algorithms expressed as Pseudocode
- Coding is translating an algorithm to a program
- Six operations a computer can do