

# Types of Data

Programming and Algorithms

Lecture by  
Dr Daniil Osudin

```
n = 3
for i in range(1,n+1):
    print("Hello World!")
```

Hello World!  
Hello World!  
Hello World!



# What will we Cover?

- Overview of what data is
- Introduction to different types of data
- Using of variables to store data

## Data:

- Collection of facts or statistics
- Input to a computer program
- Output of a computer program

```
# Example to check type:  
print(type(12.3))  
<class 'float'>
```

## Data types:

- Consider how the program should handle the data
- Determine what operations can be performed on the data
- Decision on how to store the data on a computer

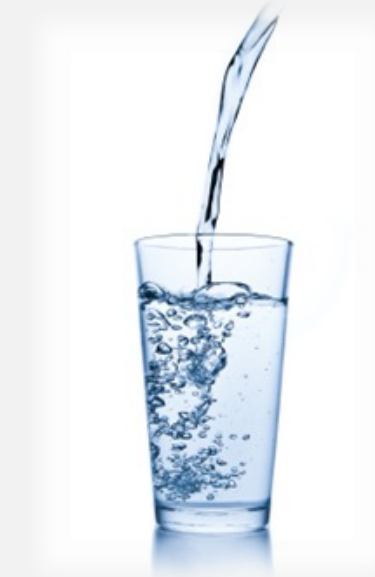
# Examples of Data



- Number of:
  - Screens
  - Keyboards
  - PCs
  - Tables
  - Chairs
- Computer lab room size
- Colour of
  - Walls
  - Floor

# Numeric Data Types

- Integer (or int)
  - Positive or negative whole numbers and zero
  - Examples
    - 13 screens
    - 2 tables
    - 1 window
    - 1 glass of water
- Floating-point (or float)
  - Positive or negative numbers with a decimal point
  - Examples
    - 22.75 m<sup>2</sup> is the Area of the room
    - 0.25 litres of water



# Other Data Types

- String
  - Sequence of characters
  - Enclosed in single or double quotes: ' ', ""
  - Examples
    - Room number: 'C203'
    - Username: "abcd012"
    - Name: "Daniil"
- Boolean
  - Only two values: True, False
  - Results from an evaluation of a statement to true or false
  - Expression 'a is equal to b' will be:
    - True if a and b are equal
    - False if a and b are not equal

# Data Structures

- Used to organise, manage, retrieve and store data
- Covered later in the module:
  - Lists
  - Tuples
  - Sets
  - Dictionaries

## Examples

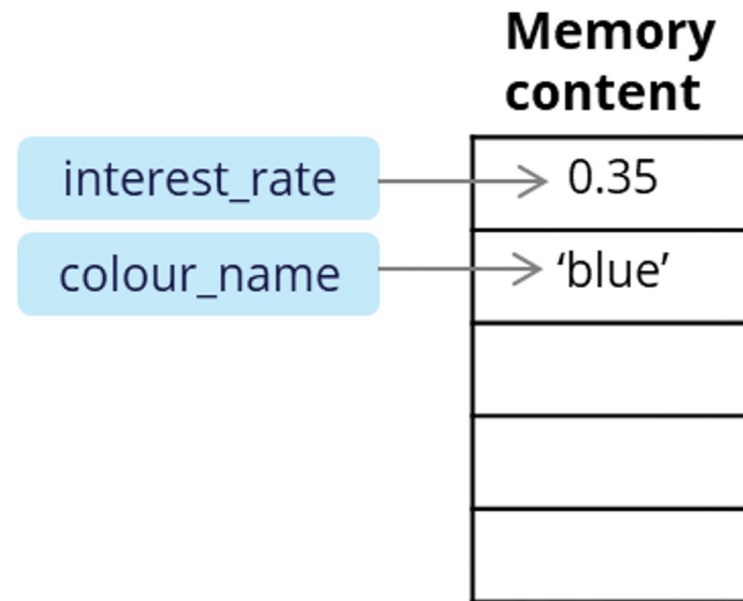
- Shopping list: a collection of strings
- Coordinates on a map: a pair of floats for latitude and longitude
- List of Contacts: a collection of accounts (name, number)



# Variables I

A **variable** is a named area in computer's memory to hold data

- Value may change during the program execution
- Created using an assignment statement



# Variables II

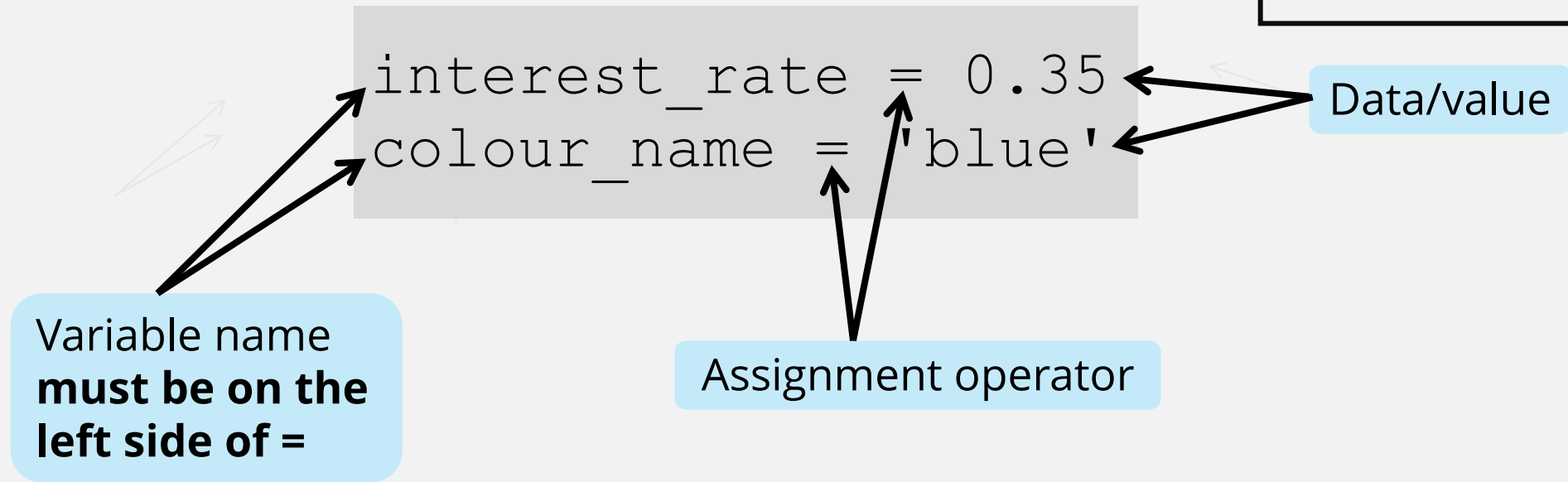
## Naming rules for variables

- Cannot begin with a number
- Contains no spaces
- Contains letters, numbers and underscore \_ only
- Cannot be a keyword

## Naming conventions for variables

- Use all lower-case letters
- Use names that indicate the purpose of the data
- If using more than one word, combine with an underscore

# Assignment Statement I



# Assignment Statement II

Multiple assignments in a single line:

```
name, surname = 'John', 'Green'
```

Same as:

```
name = 'John'  
surname = 'Green'
```

# Assignment Statement III

Multiple assignments of different data types:

```
pet, total, radius = 'dog', 50, 2.5
```

Same as:

```
pet = 'dog'  
total = 50  
radius = 2.5
```

String

Integer

Float

# Common Errors in Python I

Incomplete statement:

```
x =
```

```
File "C:\Users\danos\AppData\Local\Temp\ipykernel_2860\3379902148.py", line 1
```

```
x =
```

```
^
```

```
SyntaxError: invalid syntax
```

Error location

Likely cause of the error

Error type

Error description

# Common Errors in Python II

Incomplete statement:

```
= 3
```

```
File "C:\Users\danos\AppData\Local\Temp\ipykernel_2860\406  
8146745.py", line 1
```

```
= 3
```

```
^
```

```
SyntaxError: invalid syntax
```

Error location

Likely cause of  
the error

Error type

Error description

# Common Errors in Python III

Using pre-built function as a name:

```
if = 3
```

```
File "C:\Users\danos\AppData\Local\Temp\ipykernel_2860\3577728022.py", line 1
```

```
if = 3
```

```
^
```

```
SyntaxError: invalid syntax
```

Error location

Likely cause of the error

Error type

Error description

**Note:** 'if' function will be covered later in the course



# Try It Yourself

Enter and run the following statements in the python environment:

```
x = 1  
print(x)
```

```
truth = True  
print(truth)
```

```
word = 'bird'  
print(word)
```

```
x = 1  
print(type(x))
```

```
truth = True  
print(type(truth))
```

```
word = 'bird'  
print(type(word))
```