Teaching London Computing

Programming for GCSE Topic 3.1: If Statements in Python

Queen Mary University of London

KING'S College LONDON

COMPUTING AT SCHOOL
EDUCATE · ENGAGE · ENCOURAGE

Department for Education

SUPPORTED BY
MAYOR OF LONDON

NETWORK OF EXCELLENCE COMPUTER SCIENCE TEACHING
UNIVERSITY PARTNER

William Marsh
School of Electronic Engineering and Computer Science
Queen Mary University of London

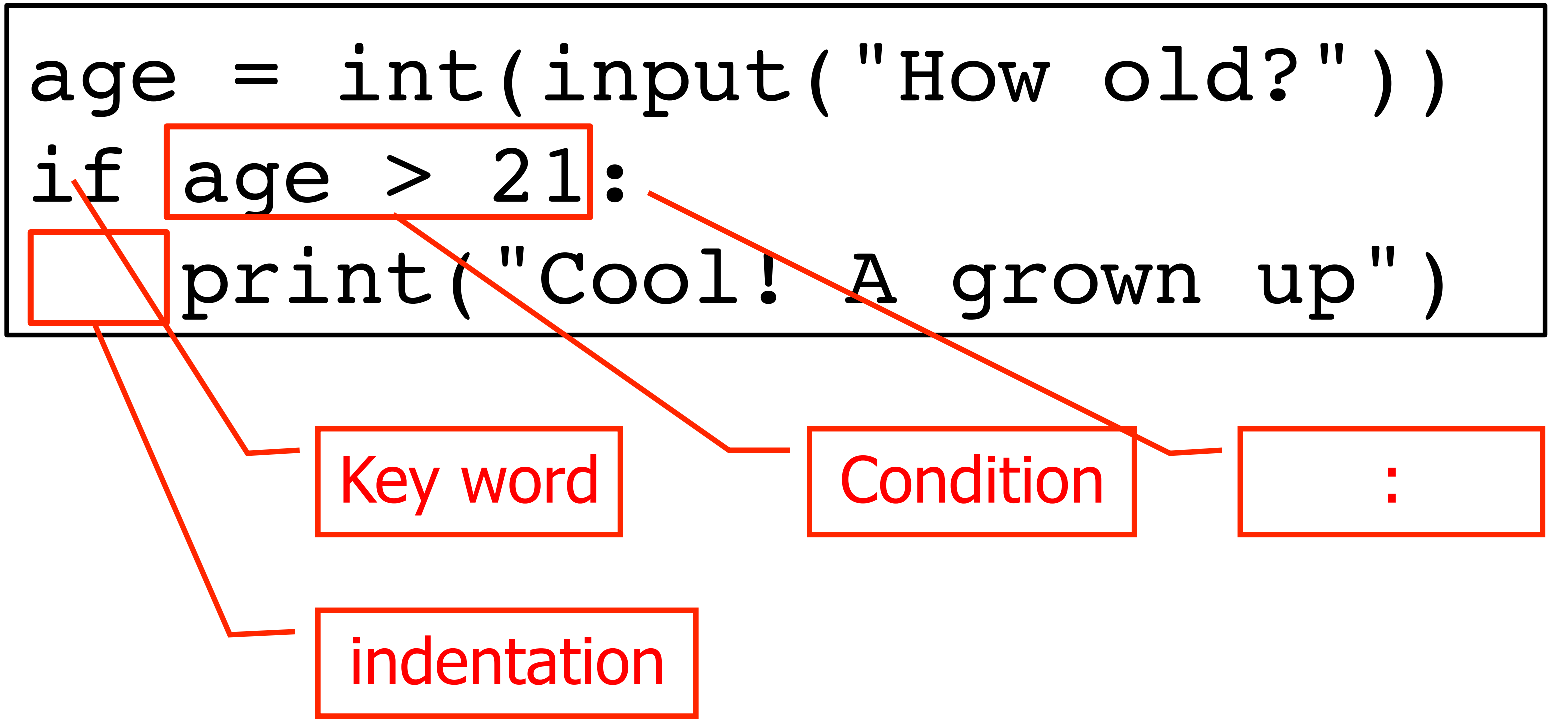
Aims

- If statement basics
- Visualising ‘If’
  - Language
  - Boxes
  - Scratch
- Comparisons
  - Boolean expressions
- More complex ‘If’ statements

Simple ‘If’ Statements

Simple 'If' Statement

- Change the statements in a program



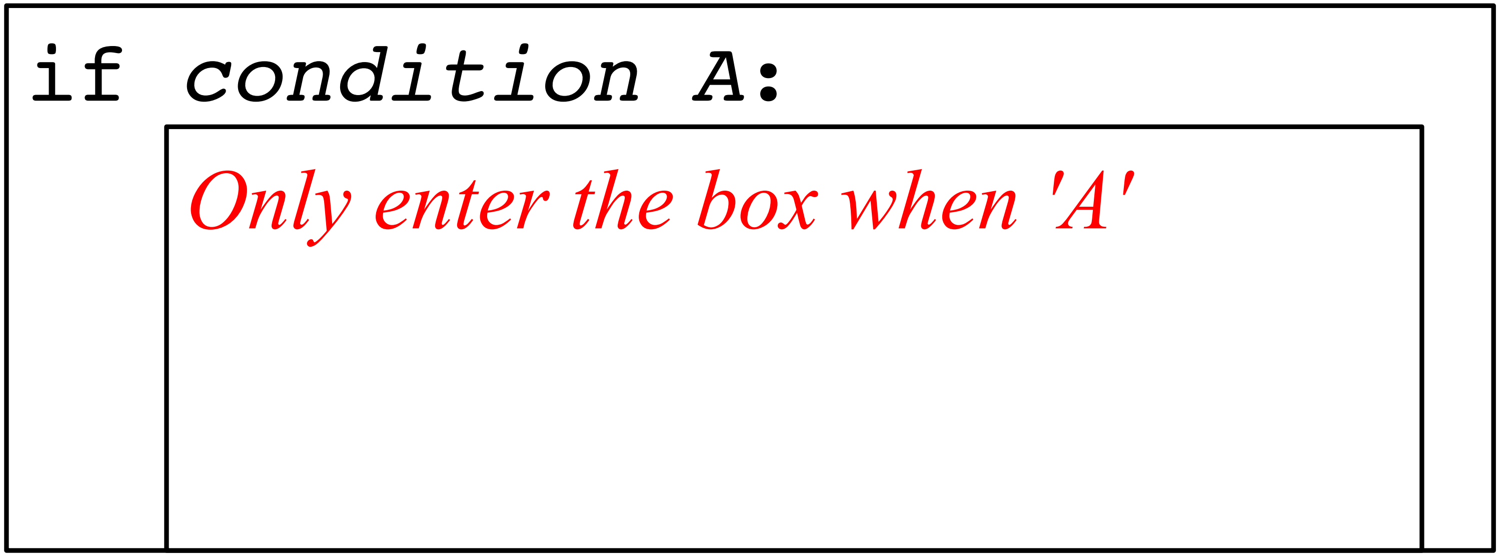
```
age = int(input("How old?"))
if age > 21:
    print("Cool! A grown up")
```

Visualising ‘If’

Indentation versus brackets

Boxes

- An 'if' statement has an inside

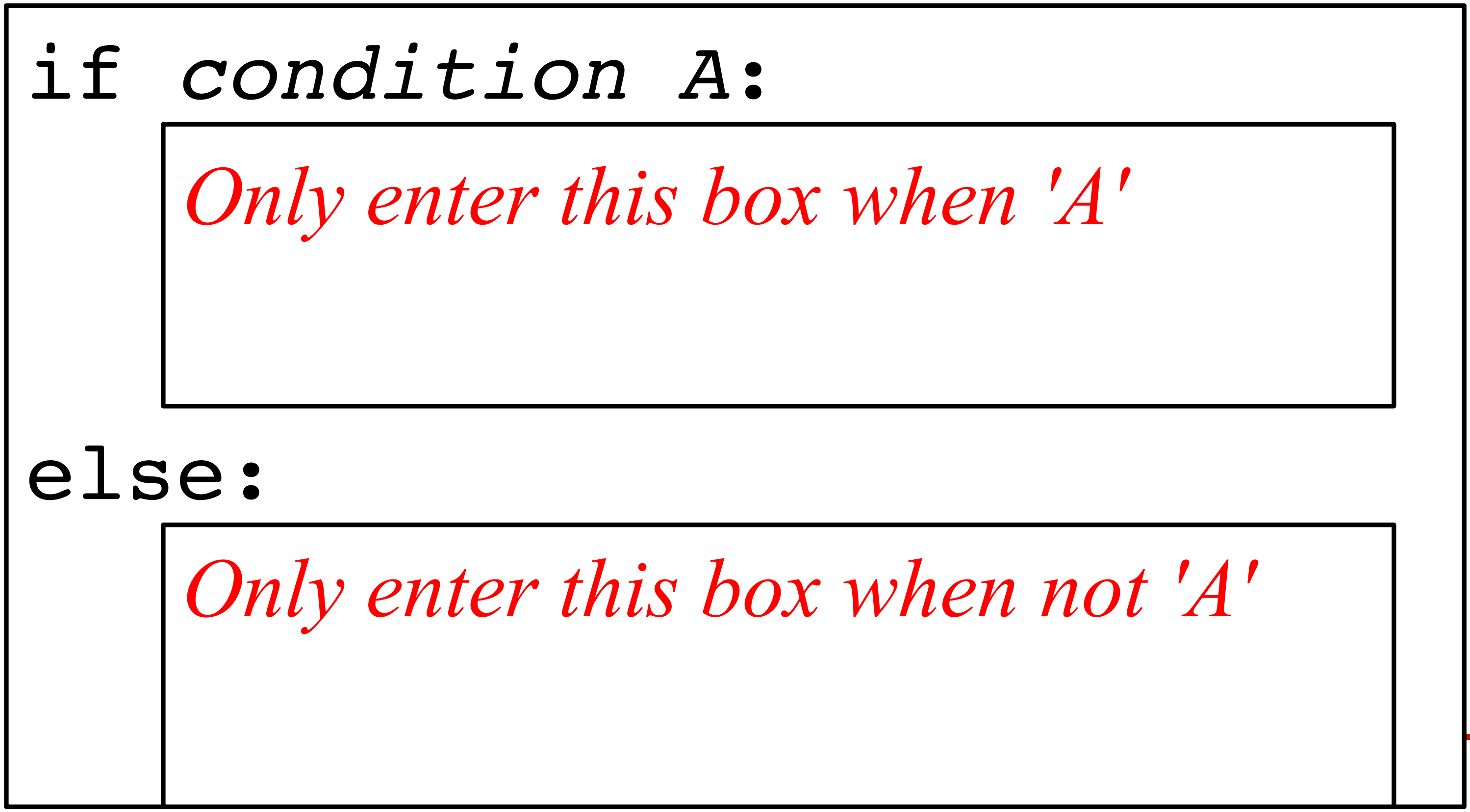


```
if condition A:
```

Only enter the box when 'A'

Boxes

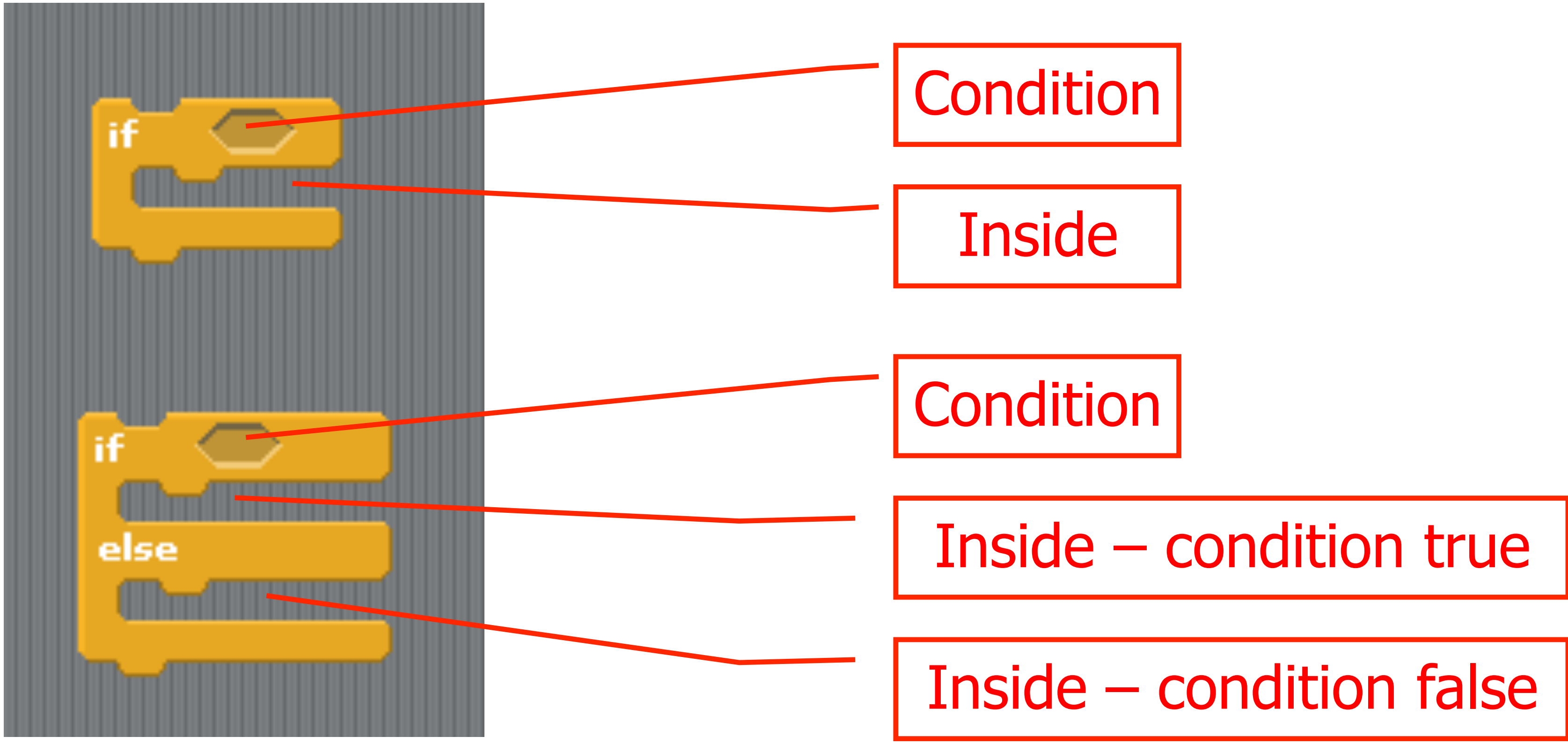
- An 'if' statement has an inside



```
if condition A:
    Only enter this box when 'A'
else:
    Only enter this box when not 'A'
```

If in Scratch

- Scratch blocks make the ‘inside’ idea obvious



Condition

Inside

Condition

Inside – condition true

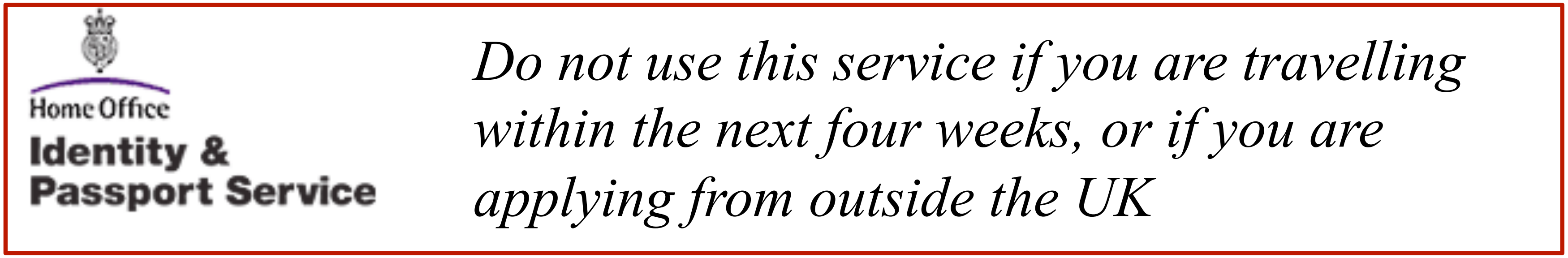
Inside – condition false

Other Languages

- Java uses
  - ( ) – condition in brackets; no colon
  - '{' – start of box
  - '}' – end of box
- Pascal
  - THEN instead of the colon
  - BEGIN – start of box
  - END – end of box

'If' Language

- Many instructions have 'if' language



Home Office
Identity & Passport Service

Do not use this service if you are travelling within the next four weeks, or if you are applying from outside the UK

```
if travelling in next 4 weeks:
   print("Do not use this service")
elif applying from outside UK:
   print("Do not use this service")
else:
   print("Ok; use this service")
```

Comparisons

True and false expression – used in 'if'

True and False Expressions

- Comparisons
- Result can be 'True' or 'False'

Operator	Meaning
`x > y`	x is greater than y
`x < y`	x is less than y
`x == y`	x is equals to y
`x != y`	x is NOT equal to y
`x <= y`	x is less than or equal to y
`x >= y`	x is greater than or equal to y

Examples

- What are the values of the following?

Expression	True or False?
`10 > 5`	
`2 + 3 < 10`	
`10 != 11`	
`10 <= 10`	
`10 >= 11`	

Comparison of Strings

- Comparison operators work for strings
- Look at the following examples

```
>>> "David" < "David Cameron"
True
>>> "Dave" < "David"
True
>>> "Tom" < "Thomas"
False
>>> "Bill" < "William"
True
>>> "AAA" < "AAB"
True
>>> "AAA" < "aaa"
True
>>> "aaa" < "AAA"
False
```

Using ASCII ordering, letter by letter

'=' and '=='

- Do not confuse
- Assignment
  - Operator =
  - Set variable to value (of expression)
- Equality
  - Operator ==
  - Compare two expressions

More Complex Conditional Statements

Else and Else If

- 'If' statement with an alternative

```
if condition:
    statement 1 – when condition true
else:
    statement 2 – when condition false
```

```
if condition A:
    statement 1 – when condition A true
elif condition B:
    statement 2 – when A false and B true
else:
    statement 3 – when both A and B false
```

Compare these programs:

```
age = input("How old are you?")
if age < 22:
   print("Ah, youth!")
if age >= 22:
   print("Cool! A grown up")
```

Equivalent

```
age = input("How old are you?")
if age < 22:
   print("Ah, youth!")
else:
   print("Cool! A grown up")
```

Teaching Issue

- Strong understanding of statements within statements
  - Needed for loops too
- Only 'if' and 'if … else' are essential
- Boolean expression: true and false as values

Summary

- 'If' allows statements in the program to vary
- Comparison operator create 'conditions' (also called 'boolean expressions')
- Python uses 'indentation' to inside the 'if' from outside
- 'Inside' versus 'outside' essential for next topic: loops