Introduction to programming using Python

Session 2

Matthieu Choplin

http://pythonguy.com/

Objectives of Session 2

- Remainder of Session 1: Quiz + exercices
- Controlling the flow of our programs
  - Conditional statements
  - The boolean type
  - While loop

Remainder of Session 1: Quiz (1)

- Which of the following are operators, and which are values?
    - *
    - 'hello'
    - -88.8
    - -
    - /
    - +
    - 5

Remainder of Session 1: Quiz (2)

- Which of the following is a variable, and which is a string?
  - spam
  - 'spam'

Remainder of Session 1: Quiz (3)

- Name three data types.

Remainder of Session 1: Quiz (4)

- What is an expression made up of? What do all expressions do?

Remainder of Session 1: Quiz (5)

- What does the variable bacon contain after the following code runs?

```
bacon = 20
bacon + 1
```

Remainder of Session 1: Quiz (6)

- Name 3 builtin functions and explain what they do

Remainder of Session 1: Quiz (7)

- How do you call a function?

Remainder of Session 1: Quiz (8)

- What function can you use to take a value from a user? For example, how can you ask the age of a user and store it into a variable?

Remainder of Session 1: Quiz (9)

- What function can you use to convert the age entered by the user and compute the years when he was born?

Remainder of Session 1: Quiz (10)

- Name three types of errors that we can get in a program

Exercise 1: converting a currency (static version)

Write a program that converts pounds into euros.

- The values can be hard coded for now (it means that the program will not be dynamic)
- Use comments
- Use variables
- Use print

👁 Show solution

Exercise 2: converting a currency (dynamic version)

- Write a program that ask the user what amount is to be converted in euros, convert it and display the result.
- Hint: we are going to need the function **input** and the function **float**

👁 Show solution

Exercise 3: computing the age of the user

- Ask a user to enter the year he was born, compute his age and tell him how old he will turn this current year.

👁 Show solution

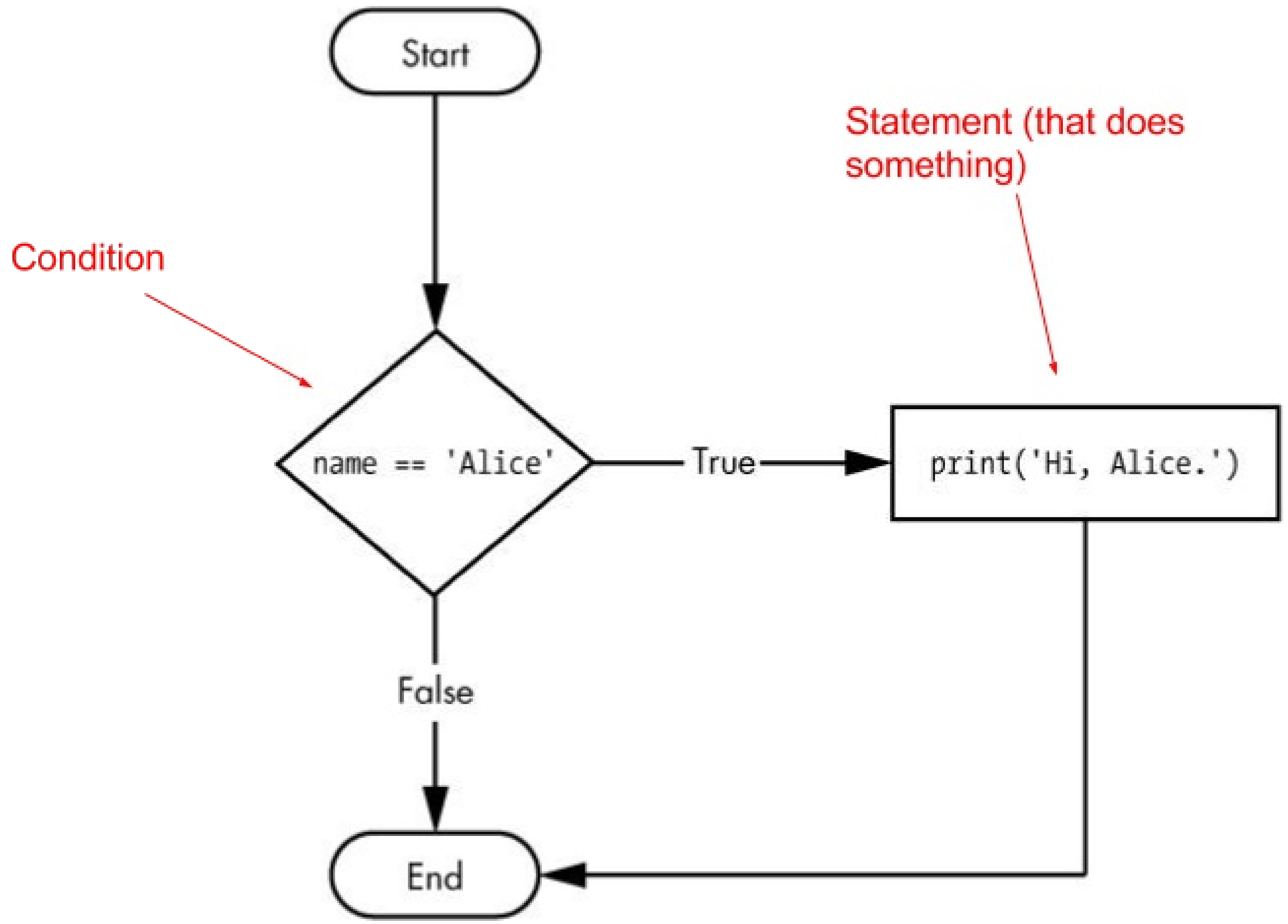
Exercise 4: fixing bugs

- Explain what this program is supposed to do and what the bug is here:

```
vat_rate = 20/100
car_price = 34500
car_price * (1 - vat_rate)
print("The gross value of the car is", car_price, "pounds")
```

Controlling the flow of our programs

We can represent the flow of execution with a flow chart



Structure of a simple if statement

Pseudo code:

```
if condition:
    # statement (mind the indentation)
```

Example, representation of the flow chart example in python code:

```python
if name=='Alice':
    print('Hi Alice')
```

The two-way if statement

Pseudo code:

```
if condition:
    # statement (mind the indentation)
else:
    # statement executed when the condition is False
```

Example, representation of the flow chart example in python code with an else statement:

```
if name=='Alice':
    print('Hi Alice')
else:
    print('Hi')
```

Multiple Alternative if Statements

The naive way

```
if condition:
    # statement (mind the indentation)
else:
    if condition:
        # statement executed when
        # the previous condition is False
    else:
        # statement executed when none of
        # the previous condition is verified
```

Multiple Alternative if Statements

The better way, the pythonc way

```
if condition:
    # statement (mind the indentation)
elif condition:
    # statement executed when
    # the previous condition is False
elif condition:
    # statement executed when none of
    # the previous condition is verified
else:
    # executed when all conditions are False
```

Value of the condition

The program will execute the statement only if the condition is verified. Only if the condition is True.

The condition is actually a **boolean**.

The Boolean Type

- It has only 2 possible values: **True** or **False**. Notice that they are both capitalized, which is important because Python is case sensitive
- It is often obtained as a result of a comparison expression.

The Comparison Operators

Operator	Meaning
<	less than
<=	less than or equal
>	greater than
>=	greater than or equal
==	equal to
!=	not equal to

Examples

```
'hello' == 'hello'
'hello' == 'Hello'
'dog' != 'cat'
True == True
True != False
42 == 42.0
42 == '42'
```

Difference between '==' and '='

- The sign **=** is the sign of **assignment**, it is used for assigning a value to a variable
- The sign **==** is the sign of **comparison**, it compares 2 values and return a boolean (True or False)

Exercise: password

Create a program that ask the user for a password.

- Have the password defined in "clear" in your program, in a variable called "PASSWORD"
- Use input() to receive the password entered by the user
- If the word entered by the user matches the password, display "Access Granted", else, "Forbidden"

Solution: password

👁 Show solution

Truth tables

Show every possible result of a Boolean operator.

The **and** Operator's Truth Table

Expression	**Evaluates to...**
True and True	True
True and False	False
False and True	False
False and False	False

The **or** Operator's Truth Table

Expression	**Evaluates to…**
True or True	True
True or False	True
False or True	True
False or False	False

The **not** Operator

It operates on only one Boolean value (or expression). The not operator simply evaluates to the opposite Boolean value.

```
not True
not not True
not 1 != 2
```

Exercise: password and login

Create a program that ask the user for a login and password.

- Have the password "PASSWORD" AND login "LOGIN" defined in "clear" in your program, in variables
- Use input() to receive the password and login entered by the user
- If login and password match the values of your PASSWORD and LOGIN, display "Access Granted", else, "Forbidden"

Solution: password and login

👁 Show solution

Exercise: check number divisor

Write a program that prompts the user to enter an integer. If the number is a multiple of 5, print HiFive. If the number is divisible by 2, print HiEven.

- Use input() take the user input
- Use int() to convert the value return by input into an integer
- Use % to see if a number x is divisible by an other number y, if x%y returns 0, then x is divisible by y
- Use print()

Solution: control flow

👁 Show solution

Exercise: grading students

Write a program that is going to give the grade of a student according to the score obtained.

- Display 'A' if the score is greater than 90
- Display 'B' if the score is between 80 and 90
- Display 'C' if the score is between 70 and 80
- Display 'D' if the score is between 60 and 70
- Display 'F' if the score is lower than 60

Solution: grading students

👁 Show solution

Exercise: determining a leap year

This program first prompts the user to enter a year as an int value and checks if it is a leap year.

A year is a leap year if it is divisible by 4 but not by 100, or it is divisible by 400.

- Use input() to take the user input (the year, i.e. 2016) and convert it with int()
- Use % to see if a number x is divisible by an other number y, if x%y returns 0, then x is divisible by y
- Check if the year is divisible by 4 AND not divisible by 100
- OR check if the year is divisible by 400.
- Use print()

Solution: determining a leap year

- 👁 Complete solution

Solution optimized: determining a leap year

- Condition to use
- Complete solution

Exercise: Chinese Zodiac sign

Now let us write a program to find out the Chinese Zodiac sign for a given year. The Chinese Zodiac sign is based on a **12-year cycle**, each year being represented by an animal: rat, ox, tiger, rabbit, dragon, snake, horse, sheep, monkey, rooster, dog, and pig, in this cycle

👁 Hint 1

👁 Hint 2

Exercise: Chinese Zodiac sign

Year	Zodiac sign
0	monkey
1	rooster
2	dog
3	pig
4	rat
5	ox
6	tiger
7	rabbit
8	dragon
9	snake
10	horse
11	sheep

👁 Complete solution