

How to use the *Pythonista* app
to learn the *Python*
programming language



1. Printing text


Press  and create an `Empty Script`. Type in these commands and run them :

```
print 'Hello world.'  
print 'I can print text.'  
print '\n'  
print 'Why did the beach cry?'  
print 'Because the seaweed!'
```

The **print** command prints text on screen.

This prints a blank line.

Edit and improve:

- Can you change the joke to make it better?
Remember to press  to check your program works!

Key vocabulary

Program – A sequence of commands which a computer follows.

Run - Carrying out the commands in a program.


2. Solving calculations

Press  and create an [Empty Script](#). Type in these commands and run them :

```
print 50 + 50
print 50 - 25
print 50 * 10
print 50 / 5
print (3 * 6) + 2
print (8 + 7) / 3
print (20 - 10) * 5
```

The answers to these calculations will be printed when the program is run.

Edit and improve:

- Can you change the numbers in the calculations? Remember to press  to check your program works!

Key vocabulary

Testing - Trying out a program to check if it works as expected.
Debugging - Finding and correcting mistakes in a program's code.

3. Text variables

Press  and create an `Empty Script`. Type in these commands and run them :

This is a variable – something that can change.

```
name = 'Molly'
print 'Hello', name

food = 'chocolate'
Print 'I like to eat', food

team = 'Manchester United'
print 'I support', team

sport = swimming
print sport, 'is fun'
```

You always put text in inverted commas!

This prints what the variable is set as.

Edit and improve:

- Change the text each variable is set as. Remember the inverted commas!

Key vocabulary

Variable – A value that can be stored and used in a program.

4. Inputting text

Press  and create an `Empty Script`. Type in these commands and run them :

This is a variable – something that can change.

The `raw_input` command makes the user type in what they want.

```
name = raw_input('What is your name?')
print 'Hello', name

age = raw_input('What is your age?')
print 'You are', age

town = raw_input('Where do you live?')
print 'You live in', town

subject = raw_input('What subject do you like?')
print subject, 'is your favourite'
```

5. Inputting numbers

Press  and create an [Empty Script](#). Type in these commands and run them :

This is a variable – something that can change.

The `float` command tells the computer the user is typing a number.

```
number = float(raw_input('Type a whole number.'))
answer = number * 8
print answer
```

```
number2 = float(raw_input('Type another whole number.'))
answer = number + number 2
print answer
```

Edit and improve:

- Change this symbol to do different calculations.

6. Random numbers

Press  and create an [Empty Script](#). Type in these commands and run them :

This sets a variable as a random number.

What happens if you change the 10 to a smaller number or the 20 to a bigger number?

```
import random

number = random.randrange(10,20,1)
print number
print number + 10
print number * 10
```

Edit and improve:

- Change these symbols and numbers to do different calculations with the random number.

7. Programs with a purpose

Press  and create an **Empty Script**. Type in these commands and run them :

Variable

User input

This works out what % you got in a test.

```
score = float(raw_input('Type your score.'))
total = float(raw_input('Type the total possible.'))
percent = score / total * 100
print 'Your percentage is', percent
```

This works out the area of a rectangle.

```
width = float(raw_input('Type the rectangle width.'))
length = float(raw_input('Type the rectangle length.'))
area = length * width
print 'The area is', area
```

Programming challenge:

Create a program that calculates the perimeter of a rectangle by adding together its two lengths and two widths, inputted by the user.

8. Lists

Press  and create an `Empty Script`. Type in these commands and run them :

```
import random

colours = ['red', 'green']
animals = ['lions', 'bears']

print 'My rainbow zoo has:'

colour = random.choice(colours)
animal = random.choice(animals)
print colour, animal
```

These two variables store lists of colours and animals. Remember your inverted commas!

This picks a random colour and animal and prints it.

Key vocabulary

List – A set of values

Edit and improve:

- Put more items in the lists to make the rainbow zoo more fun!

9. Functions

Press  and create an `Empty Script`. Type in these commands and run them :

Key vocabulary

Function – A sub-program which can be called (run) later using its name.

This function is named `cointoss`.

```
import random

def cointoss():
    options = ['heads', 'tails']
    result = random.choice(options)
    print result
```

```
cointoss()
cointoss()
cointoss()
cointoss()
cointoss()
```

This is a function – a set of commands with a name that does something (tosses a coin).

Programming challenge:

Change this function to roll a dice instead. Change its name from `cointoss` to `roll`.

Then change the options to `[1, 2, 3, 4, 5, 6]`

10. Conditional (if) statements

Press  and create an `Empty Script`. Type in these commands and run them :

This program prints 'Correct' if the user answers correctly, else prints 'Wrong' if they answer incorrectly.

```
answer = raw_input('Do cats bark? ')
if answer == 'no':
    print 'Correct'
else:
    print 'Wrong'
```

REMEMEBR THE COLON!

THIS MUST BE TABBED IN!

== means 'is the same as'

Edit and improve:

- Change the question being asked (and the answer too, if needed).

11. OR statements

Press  and create an [Empty Script](#). Type in these commands and run them :

```
answer = raw_input('Is it dark at night? ')
if answer == 'yes' or answer == 'YES':
    print 'Correct'
else:
    print 'Wrong'
```

The **or** command lets the user input different answers but still get the question correct.

Edit and improve:

- Change the question being asked (and the answer too, if needed).

Key vocabulary

Conditional (IF) statement – Decides which commands to run depending on whether certain things (conditions) are true or false.

12. Score calculators

Press  and create an `Empty Script`. Type in these commands and run them :

```
score = 0

answer = raw_input('Is it grass green? ')
if answer == 'yes' or answer == 'YES':
    print 'Correct'
    score = score + 1
else:
    print 'Wrong'

answer = raw_input('What is 3 + 3? ')
if answer == '6' or answer == 'six':
    print 'Correct'
    score = score + 1
else:
    print 'Wrong'

print 'Your score is', score
```

This variable sets the score to 0 at the start.

This adds 1 to the score if the user answers correctly.

The quiz uses conditional (if) statements to print if the user answers correctly or not.

Programming challenge:

Create your own quiz that prints the player's score at the end.

13. While loops

Press  and create an `Empty Script`. Type in these commands and run them :

```
password = 'fish'
guess = ''

while (password != guess):
    guess = raw_input('Enter password: ')
    if password == guess:
        print 'Correct'
    else:
        print 'Try again'
```

This variable sets password as 'fish'.

This while loop keeps repeating while the guess is wrong, until the guess is correct.

A conditional (if) statement prints if the guess is correct or not.

Key vocabulary

While loop – Commands in a while loop keep repeating until a condition is met (e.g. the correct password is inputted).