BLIBSPACE EDU

Paul Martin

WHAT IS BLIBSPACE?

blibSpace is a website, designed around the concept of cooperative programming and development. The website *blibspace.com* has multiple subordinate sectors: one for schools and education (edu and preblib) and two for the general public (hub.blibspace.com and howto.blibspace.com).

The one for the schools can be accessed by '[school name].blibspace.com' for example 'isbremen.blibspace.com'. On the login page each student has to input his username (first letter of given name + surname). For example pmartin.
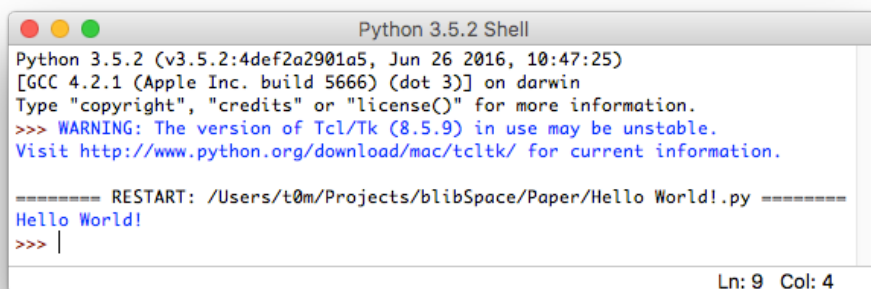
To be able to use the website, students must already have some knowledge about programming. If this is not the case, just follow the *Introduction to Programming*.

INTRODUCTION TO PROGRAMMING
What is Programming?

Programming is the process that is done to create computer programs. These programs contain instructions, telling the computer what it should do once they are executed.
One example is a program called 'Hello World!'. It contains one instruction: output the text 'Hello World!'. When you execute it, a dialogue box appears, containing the text 'Hello World!'.

The instructions can be written down in different styles (called languages). This can be compared to the real world example of how the english phrase "I am happy" can be expressed in many different ways:

Spanish: "Yo soy feliz"

French: "Je suis content"

English: "I am happy"

German: "Ich bin fröhlich"

Some programming languages are very similar to one another while they can differ to others quite a bit.

Why are there so many different languages?

This is simply the case because they specialise in different things. For example, some languages specialise in: * Being able to run quickly * Simple to learn by beginners * Simplifying commands to enable programmers to work quickly

Matt Sherman wrote an interesting article on this topic that is on-point: https://stackoverflow.blog/2015/07/why-are-there-so-many-programming-languages/
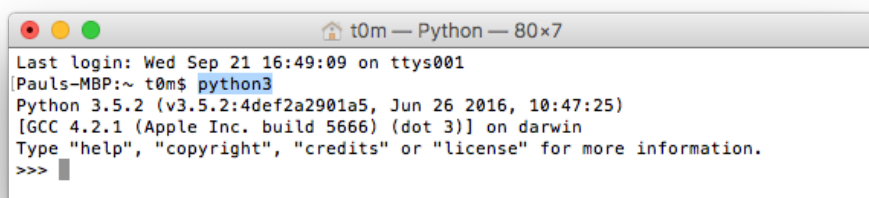
We will be working with **Python**

Python

What is Python?

Python is a programming language that is easy to learn but also frequently used by more experienced developers due to it's great variety of possibilities.

Python in a Terminal

Open up the program Python3

```
Last login: Wed Sep 21 16:49:09 on ttys001
[Pauls-MBP:~ t0m$ python3
Python 3.5.2 (v3.5.2:4def2a2901a5, Jun 26 2016, 10:47:25)
[GCC 4.2.1 (Apple Inc. build 5666) (dot 3)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

A command line with three arrows will show up `>>>`
Type `print("Hello World")`
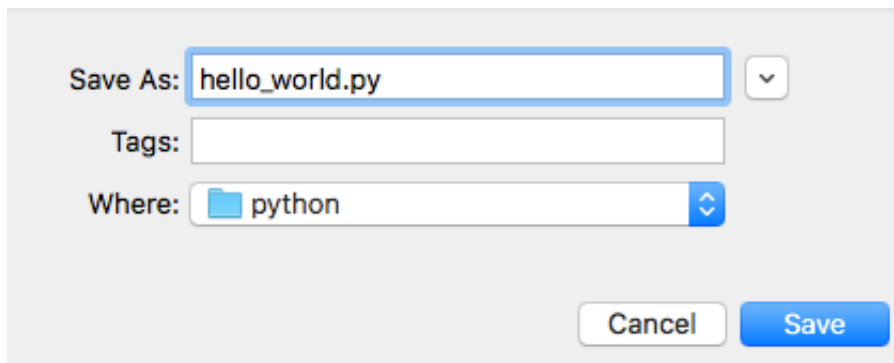The terminal will output:

Hello World!

This is basically what the program 'Hello World!' does.
You have just used your first command `print()` to output a text.
Close the terminal-window once you are done.

Writing a Python Program

Create a new text file with the text-editor of your choice (I suggest using <u>Atom</u> or <u>SublimeText</u>) and name it 'hello_world.py'. The ending '.py' tells the computer that the file is a Python program.

The file you just created will contain the code for your first Computerprogramm.
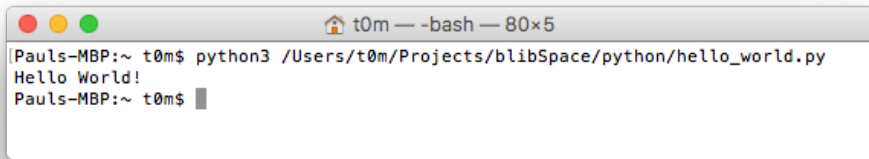Just write down the command `print()` containing some text, to output the text entered.
The file should contain `print("Hello World!")` (or some other text you chose instead of Hello World!).
Save the file and open a new terminal.
Find the file you created (hello_world.py) in Finder (macOS) or FileExplorer (Windows).
Simply drag the file onto the icon of Python and see what happens.

The terminal window should look similar to this:

```
● ● ●                    🏠 t0m — -bash — 80×5
[Pauls-MBP:~ t0m$ python3 /Users/t0m/Projects/blibSpace/python/hello_world.py  ]
Hello World!
Pauls-MBP:~ t0m$ ▌
```

Learning Python

Basics

Go to the chapter called PreBlib on <u>blibspace.com</u>. There you will find the stages
one to eight. I developed these stages to teach the basics of python.
Each one focuses on a different topic. A stage consists of an introduction to the
topic, some examples and a few tasks. To solve the tasks, you enter your code
below the comments (marked with #). To debug (remove programming errors) or
test your program, just execute it.

Once you are done with the program and have learnt the basics, you can move on
to your school's sector: *[school name].blibspace.com*