# COMP3002 – Alternative Computing Paradigms

## Lab 01 – Basic Python Programming (in Visual Studio)

## Thomas Wennekers, University of Plymouth

## Aims:

We have decided to use Python as the base language in AINT357 because it is one of the most widely used computer languages now and it did not feature very much in modules in previous years. Learning Python should be useful for your career.

We will also come across a diverse set of different concepts and languages during the module, like F#, Lisp, Prolog, Brainfuck, Python, Optical, Neural and Quantum Computing, and some more. If we would present each of these topics using a different setup, code or environment students could easily become confused. We therefore will almost exclusively use Python as the main language and Visual Studio as the main development environment; almost everything else will be developed from within this homogeneous setup. All of you know Visual Studio and many of you will know Python at least to some degree. This should make it easier to focus on the computational and theoretical concepts.

This lab provides basic level experience in Python programming. You should try to get an understanding of the basic level usage of Python quickly, ideally within the next two weeks, because much of the later code will be implemented in Python. We will even look into Python implementations of other computer languages like Lisp or Prolog, and we will use Python in the second half of the semester to simulate neural networks and quantum computers.

Advanced concepts of Python programming will be introduced as we need them. This applies especially to the functional programming constructs Python supports. The module will not provide a complete overview over Python. If you want to learn more, you can find many tutorials on the Web. This lab will make use of the tutorial at <https://www.python-course.eu/>

If you are already an experienced Python programmer you will not learn a lot of new things today; however, try to review the concepts and constructs mentioned in this lab tutorial.

## Python 3 tutorial

The material in this lab is by self-study. Please make yourself familiar with the Python 3 course website at <https://www.python-course.eu/>



Figure 1: Screenshot of Python 3 Course webpage.

## Python in Visual Studio

As mentioned earlier, the module will cover different programming languages and simulations of advanced computational systems all in Python. There are various Python environments around that could be used (and you are free to use them for your own experiments). However, the University’s IT services have setup a Python version embedded within Visual Studio that we will use in the labs. It is installed in the 2nd floor Babbage labs.

If you want to install the system on your own computer, you will have to do this from the University’s Software Centre, see Figure below:

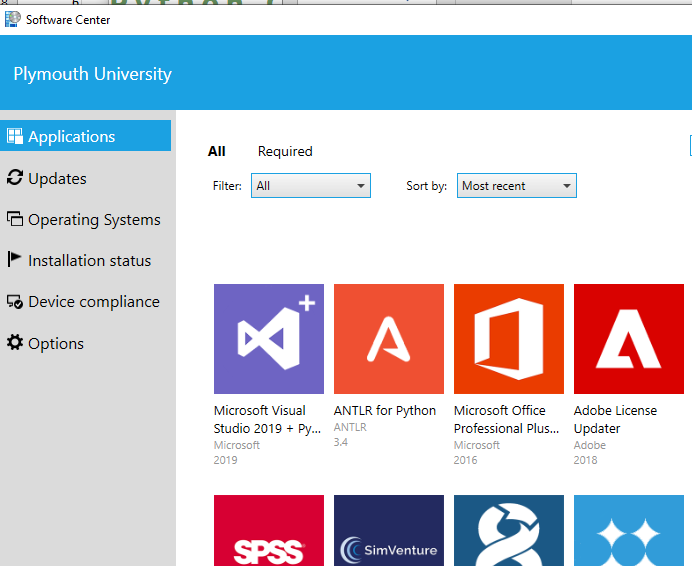


Figure 2: Screenshot of the University's Software Centre.

The software packages you would need are MS Visual Studio 2019 with Python and F# and ANTLR for Python. They are in the top left in the figure.

Unfortunately, the MS VS installation requires more than 100GB free space on your hard drive. It may not be possible to install this on a laptop. You could still use an existing VS installation and add Python manually (search the web, how to do this). Or use any Python version you are familiar with, Spyder, IDLE, etc. This should be Python 3, most code I test uses Python version 3.5; higher versions should work.

## How to start

You can start VS in the usual way in the Babbage labs. If you set up a project, choose “Python” and then “Python Application” Template. Give the project a proper name …

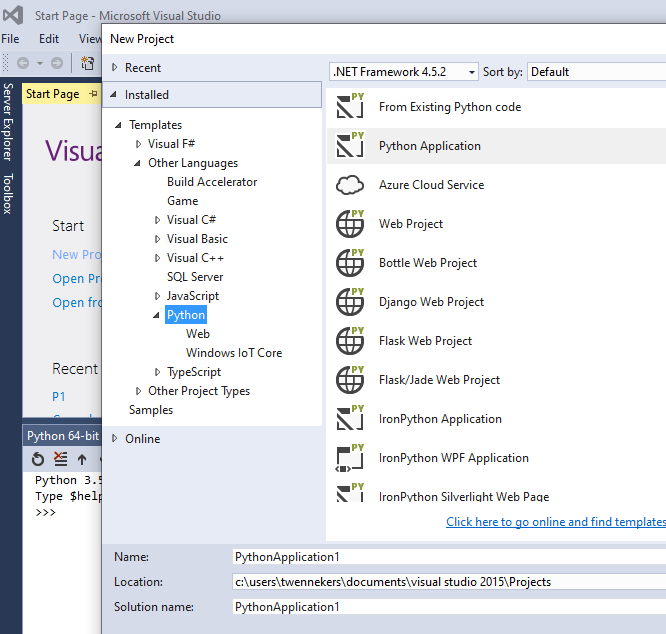


Figure : Creating a Python Project.

When the Project is created you will see the VS development environment, see Figure 4.

The bottom left window in Fig.4 is Python’s interactive shell, a command interpreter were you can type Python statements and commands for the system to execute them.

The upper left are is for code and other files as usual in VS. Note that your setup may look different. You will probably also use a different version of VS (the version in the figure is VS 2015) and Python (here Version 3.5).

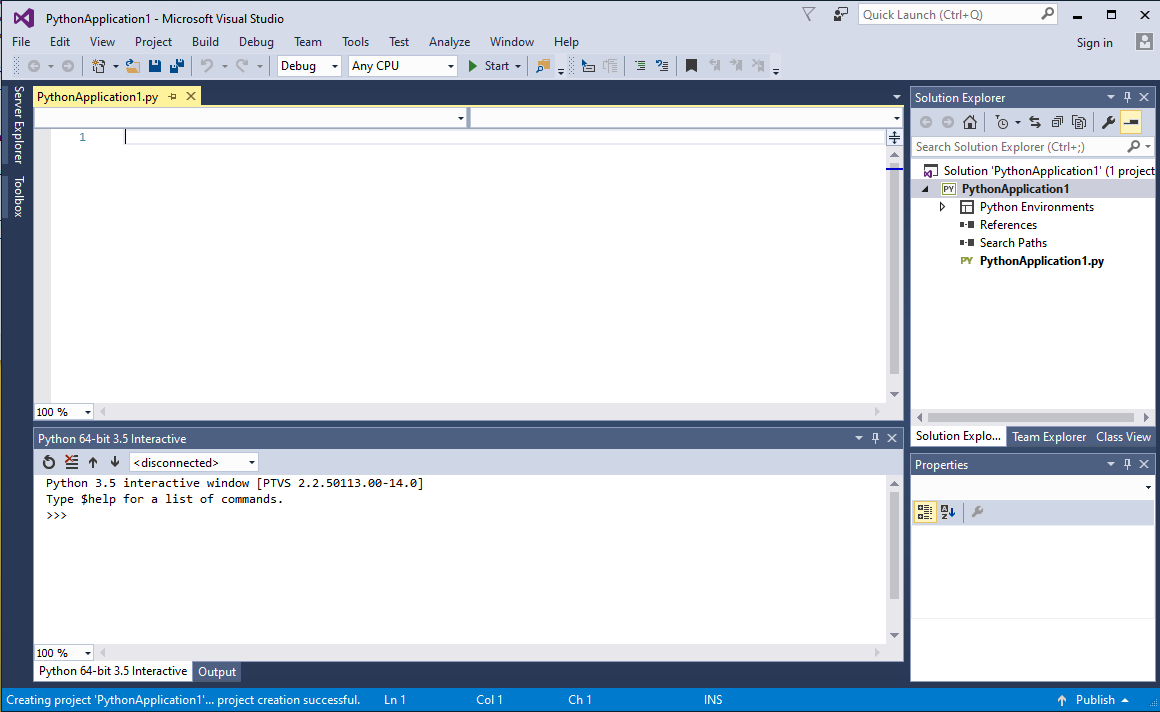


Figure : MS VS Development Environment with Python.

## Tasks

We have not developed our own Python introduction, but rely on the Python 3 course at <https://www.python-course.eu/> Please make yourself acquainted today with the concepts in the list below. Go through the respective sections of the course and check the constructs out in Visual Studio right away. If you need more time than 2 hours, please proceed during the week. In future lectures we will basically assume that you are fluent in the concepts below.

* Interpreter / Shell
* Program/Script structure, especially block indentation rules
* Types and Variables
* Interactive I/O (to/from console)
* Arithmetic and Logical Expressions
* For and while loop
* Conditional statements
* Functions/procedures with parameters and return values
* Lists, arrays, dictionaries
* I/O to/from files
* Modules
* Classes