Practical Programming
in Python

Inspired by 'Practical Programming' by Paul Gries, Jennifer Campbell, Jason Montojo

Introduction
Getting to Know Each Other, The Plan & Getting Ready

Who are we?, Who are you?, What do you need?, What do you expect?, Why do you want to program?, Do you?, What do you know already?, What we are going to do.

Kurt Rinnert, Kate Shaw

Physics Without Frontiers

The Abdus Salam International Centre for Theoretical Physics

Abstract

"If you fail to prepare you are preparing to fail."
– Anonymous

First, let's introduce ourselves.
We (all of us) will need to spent some time to make sure everyone is technically ready to go.
We'll present our plan & detailed outline for the course.
We'd like to get an impression of what you expect, what you know and what you want to learn.
Then we'll discuss possible changes to the plan to best meet your needs.

Overview

- Who we are
- Who you are
- Getting the work environment ready
- The plan
- Your expectations & requirements
- Discussion of amendments to the plan

After this, we are good to go on the journey.

The Work Environment

- We provide a virtual machine with an Xubuntu installation
- This comes with
  - a Python 3 environment with everything needed for the course
  - an Integrated Development Environment (IDE), especially designed for teaching
  - all the files (slides & exercises) relevant to the course
  - a Linux developer & desktop install for those who want to explore
- This should work on Linux/OSX/Windows

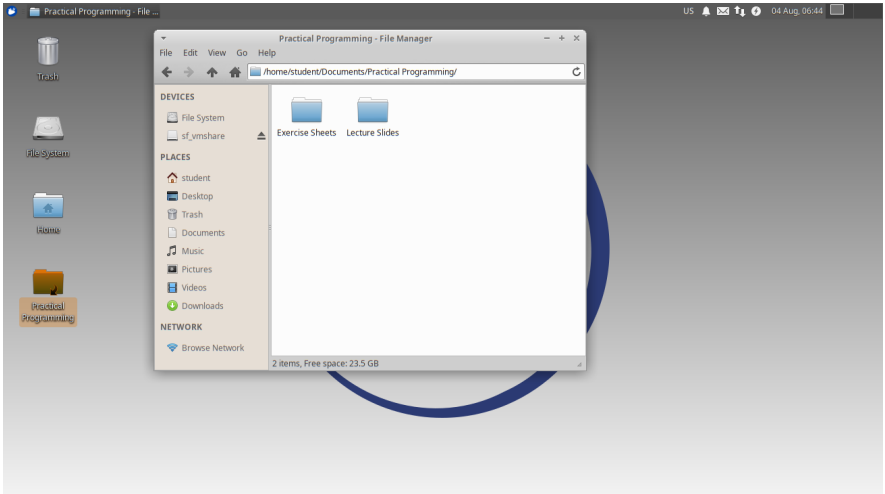
We *strongly* recommend you use this. If you insist otherwise, we *just might* support you.

Let’s get ready!

After Starting the VM

You can customize the look & feel, if necessary.

Finding the Course Files

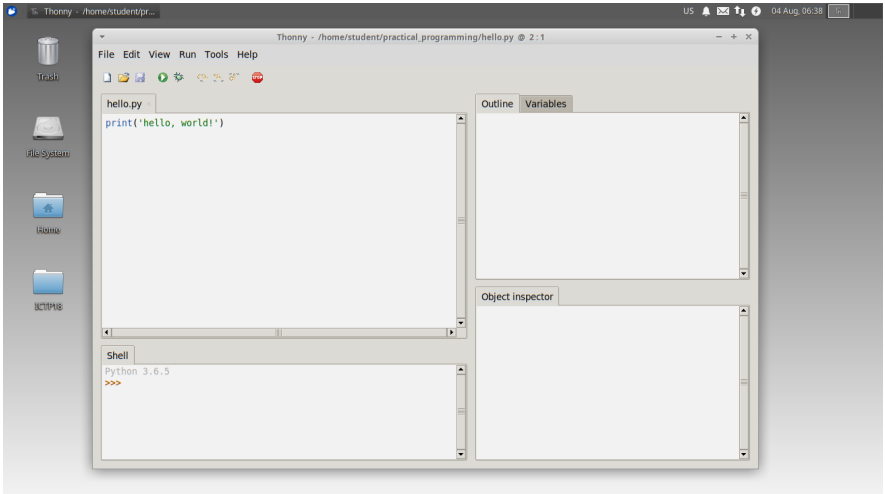


This folder contains all slides & excersise sheets.

Starting the Thonny IDE

In general, you start applications from this menu.

The Thonny IDE



This is a special version adapted for this course. You'll spend a lot of time using this.

Everyone good to go?

The Plan

- This is about *practical* programming
- It is *not* about a programming language…
- …although, of course, we have to use one
- We will
  - use python, in particular version 3
  - introduce basic & advanced concepts
  - explain how these are represented in the python virtual machine
  - enable you to write useful programs, including graphical user interfaces (GUIs)
- The course is a mixture of lectures/hands-on sessions & exercises
- There is *no grading*, but a lot of immediate feed back
- We'll finish with a fun programming challenge

The aim is that you'll be able to write the programs *you* need.

Course Outline

Introduction

Lecture 1: What's Programming

Lecture 2: Hello, Python

Lecture 3: Designing and Using Functions

Lecture 4: Working with Text

Lecture 5: Making Choices

Lecture 6: A Modular Approach to Program Organization

Lecture 7: Using Methods

Lecture 8: Storing Collections of Data Using Lists

Lecture 9: Repeating Code Using Loops

Lecture 10: Reading and Writing Files

Lecture 11: Storing Data Using Other Collection Types

Lecture 12: Designing Algorithms

Lecture 13: Searching and Sorting

Lecture 14: Object-Oriented Programming

Lecture 15: Testing and Debugging

Lecture 16: Creating Graphical User Interfaces

The Challenge: Droids!

We can shift emphasis and add items according to your needs. (Within limits).

Please consider the questions on the first exercise sheet
for a few minutes.

Then we discuss possible modifications/additions to the course.