Introduction to Programming in Python

Module 1: Course Introduction

Outline

Course Responsible

- Course responsible: Andrea Vandin
  - ★ a.vandin@santannapisa.it
  - ★ Tenure-track Assistant Professor in Computer Science at Institute of Economics & EMbeDS @ SSSA
  - ★ Formerly:
    - ▶ Associate Professor in Computer Science at DTU Technical University of Denmark
      - Most related teaching activity: responsible for 3 years of course 'Programming in C++ for non-computer scientists', 250 students
- Co-lecturer: Daniele Licari
  - ★ d.licari@santannapisa.it
  - ★ EMbeDS Data Engineer
  - ★ Great expert of Python

Course References & Material

- Webpages of the course:
  - ★ https://bit.ly/Intro2Python1920SSSA
    - ▶ Slides and examples from the lectures, further materials and links
  - ★ https://repl.it/student/classrooms/180817
    - ▶ Weekly coding assignments
- Suggested book: M. Lutz, Learning Python.
- Well-done tutorial: https://docs.python.org/3/tutorial/
- Software
  - ★ Python: https://www.python.org/
  - ★ Suggested Python editor: JupyterLab https://jupyter.org/
  - ★ Setup your machine: https://bit.ly/Intro2Python1920SSSA-setup

Tentative Course Description

This course will

introduce the students to the fundamental principles of structured programming with basic applications to data processing. Using Python as reference language, the course starts from basic notions of programming (variables, data types, collections, control & repetition structures, functions & modules), up to basic data processing functionalities (loading, manipulation, and visualization of CSV data).

Tentative Course Description

This course will

introduce the students to the fundamental principles of structured programming with basic applications to data processing. Using Python as reference language, the course starts from basic notions of programming (variables, data types, collections, control & repetition structures, functions & modules), up to basic data processing functionalities (loading, manipulation, and visualization of CSV data).

A student who has met the objectives of the course will

acquire a high-level understanding of the issues involved in computer programming, so to be able to make informed decisions. The student will be able to write simple Python programs of various nature, including those for reading, manipulating and visualizing data.

Tentative Learning Objectives

A student who has met the objectives of the course will be able to:

- select and use the correct data types for the problem at hand
- use variables and operations
- use and describe control and repetition strctures (if, loops)
- use and describe collections (lists, . . .)
- create and use functions, including recursive ones
- create and use classes with encapsulation and inheritance
- use libraries for File I/O, data manipulation, and data visualization
- use principles of structured program design and methods
- explain and apply the principles of abstract data types
- discuss Python-related issues in a clear and concise way, possibly using on-line platforms

Evaluation

- Weekly coding assignments
  - ★ To be handed in via *Repl.it* at https://repl.it/student/classrooms/180817
    - Automatic tests for your code and hints to fix bugs
    - Deadlines: before the following class
    - Feel free to contact us for support
  - ★ A fundamental learning tool of this course
- Oarl Exam - TBC
  - ★ You will solve a sort of bigger final assignment at home
  - ★ We will do an oral examination starting from your solutions
    - to the final assignment
    - to the weekly assignments
  - ★ Date: TBD

Tentative Lecture Plan

#	Date	Time	Topic
1	16/04	17:30-19:30	Course introduction
2	20/04	15:00-18:00	Data types & operations
3	27/04	15:00-18:00	Collections
-	04/05	–	*Break*
4	11/05	15:00-18:00	Control and Repetition structures
-	18/05	–	*Break*
5	25/05	15:00-18:00	Functions
6	01/06	15:00-18:00	Exceptions and OOP
7	08/06	15:00-18:00	Basic data manipulation & visualization
-	TBD	TBD	Exam

Further info

- No previous experience on computer programming required
- Previous experience in writing small programs is advantageous
- We might adjust the course level according to your expertise and feedback

Further info

- No previous experience on computer programming required
- Previous experience in writing small programs is advantageous
- We might adjust the course level according to your expertise and feedback
- You will never learn programming if you don't practice it!
  - ★ Therefore you have to regularly do all the assignments

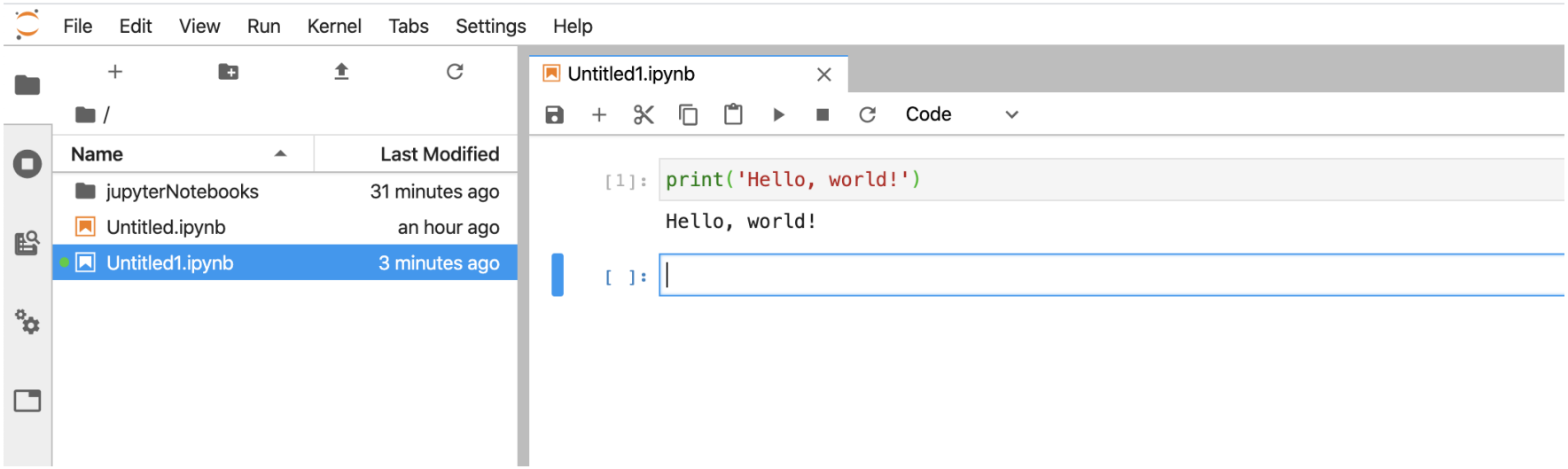
Ideas for an Effective Course

Live Programming & Assignments

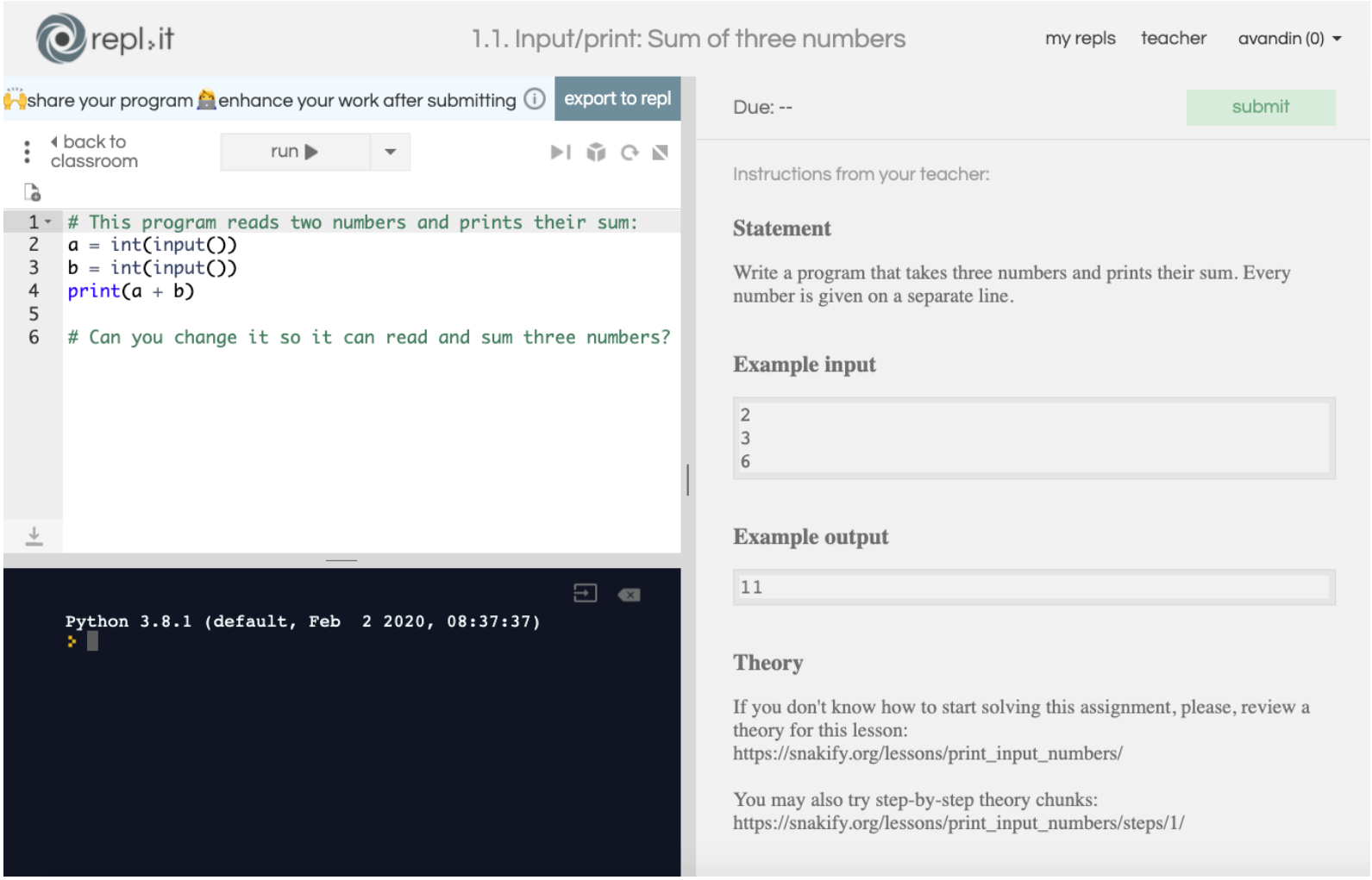
We have blocks of 3 hours.

- First part:
  Intro to week's topics & Live programming
  - ★ Not many slides
  - ★ Instead: we develop a few example programs
    - ▶ Please have your laptop ready!
      https://bit.ly/Intro2Python1920SSSA-setup
    - ▶ You find code in advance here
      https://bit.ly/Intro2Python1920SSSA-slides-code
- Second part:
  You consolidate your understanding working on the assignments
  - ★ Begin working on the assignments with our live support if needed
  - ★ Complete them offline before next class. Contact us if needed

Live Programming



Repl.it



- First time visit: https://repl.it/classroom/invite/olYrkRa
- After that: https://repl.it/student/classrooms/180817

Repl.it

- A Repl.it classroom is a collection of assignments with autograding functionalities
- Yo will find have a series of assignments.
- Your dashboard will be an ordered list of assignments.

Outline

What is a program?

- A sequence of code instructions to control a machine
  - ★ Input/output
  - ★ Mathematical operations
  - ★ Conditional and repetitive executions
- A recipe to instruct a machine to execute instructions.
  - ★ We can't use a *natural language*.
  - ★ We need a **programming language**

Programming languages

The 7 Most In-Demand Programming Languages of 2019

March 15, 2019

Aspiring developers need to know what languages to learn; they need to select the right education and work on a skill set that will impress future employers and land their dream job. So what are the top programming languages? And what is the best one to learn? We've compiled a list for you that highlights the most in-demand programming languages based off current job postings on the market.

Here are the Top 7 programming languages with most job posting on Indeed as of January 2019:

- Java – 65,986 jobs
- Python – 61,818 jobs
- Javascript – 38,018 jobs
- C++ – 36,798 jobs
- C# – 27,521 jobs
- PHP – 16,890 jobs
- PERL – 13, 727 jobs

www.codingdojo.com/blog/the-7-most-in-demand-programming-languages-of-2019

The Python Programming language

- High-level: almost human readable. Abstracts from hardware
- Beginner-friendly: streamlined syntax. Get easily to write simple programs
- Free, open-source and multi-platform
- Developed since the 90s, therefore it has
  - ★ A wide community
  - ★ Many predefined software modules

Python programs

- A sequence of python instructions to control a machine
- Python supports the most common programming styles
  - ★ Imperative: Statements are executed in sequence changing the state of the program (the variables)
  - ★ Procedural: The program is structured in reusable units named functions
  - ★ Object-oriented: The program is structured as a collection of interacting objects that send messages to each other.
  - ★ Functional: Statements are not written/executed as an ordered sequence of instructions. A computation is treated as the evaluation of a mathematical function.

Variables

Basic abstraction to represent units of data

A variable has a name and a value

- Names can contain any letter, number, or the underscore _

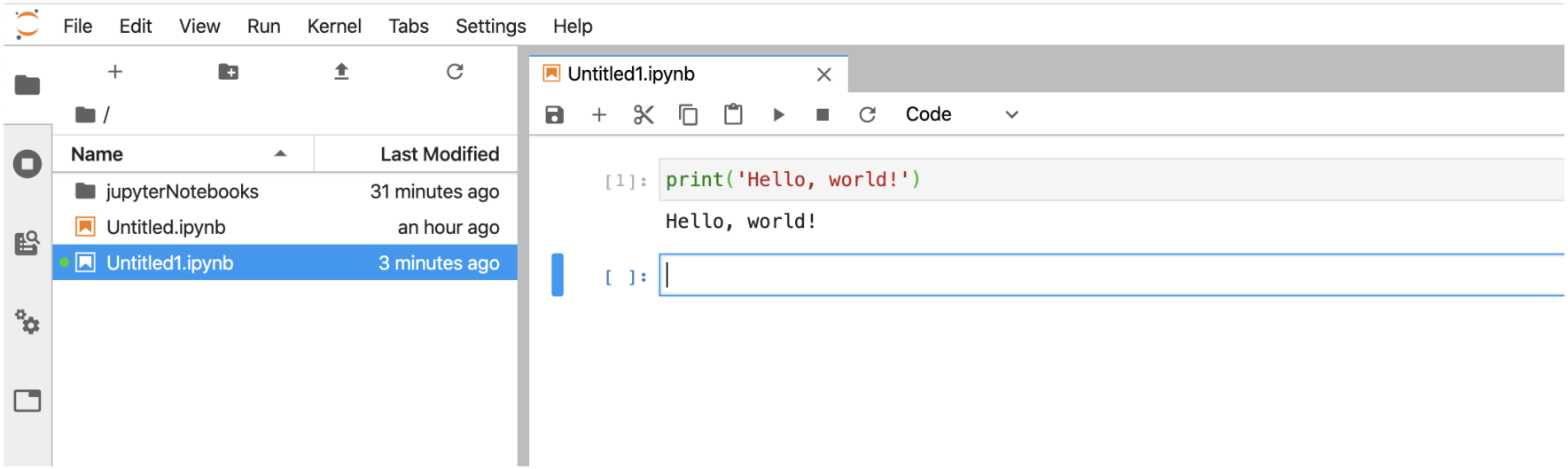
False	await	else	import	pass
None	break	except	in	raise
True	class	finally	is	return
and	continue	for	lambda	try
as	def	from	nonlocal	while
assert	del	global	not	with
async	elif	if	or	yield

- ★ Cannot start with numbers
- ★ Cannot be a keyword
- ★ Names are case-sensitive

- We assign/update values to variables using assignment statements

```
monthNumber=3
monthName="April"
print("The number of",monthName,"is",monthNumber)
monthNumber=4
print("The number of",monthName,"is",monthNumber)
```

Live Programming



Configure your machine

If you have not done it yet

Follow the instructions in
https://bit.ly/Intro2Python1920SSSA-setup