Hertie School

Master of International Affairs/Master of Public Policy
Fall Semester 2020

Course Syllabus, Version 08.07.2020

GRAD-E1326: Python Programming for Data Scientists

Concentration: Policy Analysis
Hannah Béchara

1. General information

Class time	Group A: Tues, 14-16h Group B: Thu, 12-14h
Course Format:	This course uses a "flipped classroom" format and combines 50 minutes of pre-recorded material (audio or video) with a 50-minute interactive seminar. Students will use the pre-recorded material to prepare for the seminar. The seminar is taught onsite at the Hertie School, or online via the platform Clickmeeting, depending upon your location. For those attending the online seminar, Clickmeeting allows for interactive, participatory seminar style teaching.
Instructor	Hannah Béchara
Instructor's office	3.14
Instructor's e-mail	bechara@hertie-school.org
Instructor's phone number	252
Assistant	TBA
Instructor's Office Hours	Tuesday 16:00 → 17:00

Link to Module Handbook **MIA** and **MPP**
Link to **Study, Examination and Admission Rules**

Instructor Information:
Hannah is an NLP post-doc who inadvertently found herself hired by Hertie's Data Science lab. In between training neural networks and support vector machines, Hannah occasionally teaches programming classes in Python, the programming language for winners. For reasons yet unclear, the University of Wolverhampton decided to award Hannah a PhD in Computer Science.

2. Course Contents and Learning Objectives

Course contents:
This Python for Data Scientists course leads the students from the basics of writing and running Python scripts to more advanced features such object-oriented programming and data structures.

Main learning objectives:
Master the fundamentals of writing Python scripts
Learn core scripting elements such as variables and flow control structures
Discover how to work with lists and sequence data

Write Python functions to facilitate code reuse
Use Python to read and write files
Learn the fundamentals of object oriented programmed, test-based programming and algorithm analysis.

Target group:

This course is designed for beginners with **no previous programming experience** who are interested in learning Python for Data Science. It is especially recommended for students who plan to take next semester's Machine Learning or Deep Learning course.

Teaching style:
Classes will be highly interactive, focusing on hands-on experience and allowing students to test out everything they have learned in class. Each 2-hour session will include a lecture that introduces some core concepts, a classroom activity that allows students to try out what they learned and solve a problem, and a discussion of the solutions to the problem.

Prerequisites:
None to take this course.
But this course is a prerequisite for Machine Learning and NLP with Deep Learning.

3. Grading and Assignments

General guidelines for instructors

The students will be assessed based on 3 assignments and 1 end-of-term project that test their problem-solving skills as well as their grasp of the core Python concepts discussed in that week's lecture. Group work is accepted.

Composition of Final Grade:

Assignment 1: Project Pitch	Deadline: Week 4	Submit via: Class Presentations	20%
Assignment 2: Midterm Report	Deadline: Week 6	Submit via Moodle.	20%
Assignment 3: Final Report	Deadline: Week 12	Submit via Moodle	40%
Final Presentations	Deadline Week 12	Class Presentations	10%
Participation grade (if applicable)			10%

The assessment for the course consists of a project, presentation and participation. The research project must be done in teams of 2-4 (individual submissions will not be accepted for the project).
The aim of the assessments is three-fold. First, it will provide you with the opportunity to apply the concepts learned in this class creatively, which helps you with understanding material more deeply. Second, designing and working on a unique project in a team which is something that you will encounter, if you haven't already, in the workplace, and the project helps you prepare for that. Third,

along with the opportunity to practice and the satisfaction of working creatively, students can use this project to enhance their portfolio or resume.

Note about grading. You will be graded on the quality of your code as well as the efficiency and tidiness. Additionally, you are expected to document your code and to clearly mark any code that is not your own. Smelly code will be penalised.
You must include a link to a GitHub repository containing the code of your project. Your repository must be viewable to the instructor by the submission deadline. If your repository is private, make it accessible to us (GitHub ID *hbechara*). If your repository is not visible to us, your assignment will not be considered complete, so if you are worried please submit well in advance of the deadline so we can confirm the repository is visible. Furthermore, we will assess individual contribution to the team, should such an issue arise, based on the frequency and quality of GitHub commits in your project repository, so make sure you start the repository as the very first stage of your project.

Assignment Details

Assignment 1

- The main purpose of the project proposal is to receive feedback from the instructor regarding whether your project is feasible and whether it is within the scope of this class. Also, the project proposal offers a chance to receive useful feedback and suggestions on your project. The goal is for you to propose the project, motivate its rationale, and assess its potential to contribute new knowledge by situating it within related literature in the scientific community.
- Students will also have to prepare a 5 minute pitch for the class. The aim of this presentation is to prepare you for the final presentation, and also get feedback from other students, as well as get an idea of what your fellow students are working on.
- For the project, you will be working in a team consisting of 2-4 students. If you have any concerns about working with someone in your group, please discuss it with the instructor.
- After you have received feedback from the instructor and your project proposal has been graded, you are advised to stick to the project outline in the proposal as closely as possible.

Assignment 2

By the middle of the course, students should present their initial progress report. This should include an early working demo of your project with some complete features. This serves as a project milestone. The milestone should help you make progress on your project and receive feedback.

Assignment 3

You must include a link to a GitHub repository containing full working code of your project, including JUnit tests and full documentation.

Class Presentation

At the end of the semester, teams will produce poster/video presentation of their work to the class and broader Hertie community. Detailed description of the presentation task and marking rubric will be made available on Moodle. Selected projects will be displayed on the lab's website.

Participation grade
We appreciate everyone being actively involved in the class. **For full** participation credit, we expect you to contribute relevant questions and ideas to the online class forum on Piazza, and answer questions from your peers. The top ~5 contributors of endorsed answers to Piazza will get 10%; others will get credit in proportion to the participation of the ~5th person. Use your real name and your Hertie email address for participation credit.

Late submission of assignments: For each day the assignment is turned in late, the grade will be reduced by 10% (e.g. submission two days after the deadline would result in 20% grade deduction).

Attendance: Students are expected to be present and prepared for every class session. Active participation during lectures and seminar discussions is essential. If unavoidable circumstances arise which prevent attendance or preparation, the instructor should be advised by email with as much advance notice as possible. Please note that students cannot miss more than two out of 12 course sessions. For further information please consult the Examination Rules §10.

Academic Integrity: The Hertie School is committed to the standards of good academic and ethical conduct. Any violation of these standards shall be subject to disciplinary action. Plagiarism, deceitful actions as well as free-riding in group work are not tolerated. See Examination Rules §16.

Compensation for Disadvantages: If a student furnishes evidence that he or she is not able to take an examination as required in whole or in part due to disability or permanent illness, the Examination Committee may upon written request approve learning accommodation(s). In this respect, the submission of adequate certificates may be required. See Examination Rules §14.

Extenuating circumstances: An extension can be granted due to extenuating circumstances (i.e., for reasons like illness, personal loss or hardship, or caring duties). In such cases, please contact the course instructors and the Examination Office *in advance* of the deadline.

4. General Readings

Required:
Michael Dawson. Python for the Absolute Beginner

Recommended:

Steven Bird, Ewan Klein and Edward Loper. Natural Language Processing with Python

Aditya Y. Bhargava, Grokking Algorithms: An illustrated guide for programmers and other curious people

Roy Osherove, The Art of Unit Testing

Optional:

https://greenteapress.com/wp/think-python-2e/

https://en.wikibooks.org/wiki/Non-Programmer%27s_Tutorial_for_Python_3

https://www.youtube.com/playlist?list=PLlrxDoHtieHhS8VzuMCfQD4uJ9yne1mE6

5. Session Overview

If your class falls on a public holiday, or if you are unavailable for one regular session, please contact the Curricular Affairs team at curricular-affairs@hertie-school.org to reschedule.

Session	Session Date	Session Title
1	Group A: 08.09.2020 Group B: 10.09.2020	Getting Started with Python: Working with Data
2	Group A: 15.09.2020 Group B: 17.09.2020	Flow Control
3	Group A: 22.09.2020 Group B: 24.09.2020	Data Structures
4	Group A: 29.09.2020 Group B: 01.10.2020	Functions
5	Group A: 06.10.2020 Group B: 08.10.2020	Project Pitches
6	Group A: 13.10.2020 Group B: 15.10.2020	Packages, Modules and Files
Mid-term Exam Week: 19.10 - 23.10.2020 – no class		
7	Group A: 27.10.2020 Group B: 29.10.2020	Object Oriented Programming I
8	Group A: 03.11.2020 Group B: 05.11.2020	Object Oriented Programming II

9	**Group A: 10.11.2020 Group B: 12.11.2020**	**OOP Case Study**
10	**Group A: 17.11.2020 Group B: 19.11.2020**	**Unit Tests and Documentation**
11	**Group A: 24.11.2020 Group B: 26.11.2020**	**Algorithm Analysis and Big-O Notation**
12	**Group A: - 01.12.2020 Group B: 03.12.2020**	**Project Presentations**
Final Exam Week: 14.12 - 18.12.2020 – no class		

6. Course Sessions and Readings

All readings will be accessible on the Moodle course site before semester start. In the case that there is a change in readings, students will be notified by email.

Required readings are to be read and analysed thoroughly. Optional readings are intended to broaden your knowledge in the respective area and it is highly recommended to at least skim them.

Session 1: Getting Started with Python	
Learning Objective	Learning to use Google Collab and Jupyter Notebooks. Learning about data types, variables, standard input and output with Python. Learning about Integers, Floats and Strings and operators. Learning to open and read files, write to files, and handle exceptions Learn the structure of Python Packages and how to import modules from packages in Python. Writing our first Python program.
Required Readings	Chapter 1 &2: Python Programming for the Absolute Beginner by Michael Dawson

Optional Readings	https://realpython.com/python-introduction/ https://www.programiz.com/python-programming/variables-datatypes https://www.w3resource.com/python/python-variable.php

Session 2: Data Structures	
Learning Objective	Lists, tuples, and dictionaries. More advanced data structures and working with strings.
Required Readings	Chapter 4: Python Programming for the Absolute Beginner by Michael Dawson
Optional Readings	https://realpython.com/lessons/lists-tuples-python-overview/ https://realpython.com/python-dicts/

Session 3: Flow Control	
Learning Objective	Learning about conditionals, loops etc.
Required Readings	Chapter 3: Python Programming for the Absolute Beginner by Michael Dawson
Optional Readings	http://www.python-course.eu/python3_conditional_statements.php http://www.python-course.eu/python3_loops.php http://www.python-course.eu/python3_for_loop.php
Session 4: Functions	
Learning Objective	To understand the use of functions, and basic principles like encapsulation and recursion. Learning to reuse functions, global variables. This session will also include a discussion on class projects.
Required Readings	Chapter 6: Python Programming for the Absolute Beginner by Michael Dawson
Optional Readings	https://en.wikibooks.org/wiki/Non-Programmer%27s_Tutorial_for_Python_3/Defining_Functions
Session 5: Project Pitches	

Learning Objective	During this session, each team will be given 5 minutes to pitch their project proposal to the class. Pitches should be pre-recorded.
Required Readings	None
Optional Readings	None

Session 6: Packages, Modules and Files	
Learning Objective	Learning to write to and read from files. Importing packages and modules, packaging your own modules and looking over some useful python libraries.
Required Readings	
Optional Readings	

Mid-term Exam Week: 19 – 23.10.2020 – no class

Session 7: Object Oriented Programming Part I	
Learning Objective	Learning about object oriented programming: classes, objects, attributes and methods
Required Readings	Chapter 8: Python Programming for the Absolute Beginner by Michael Dawson
Optional Readings	http://greenteapress.com/thinkpython2/html/thinkpython2018.html http://greenteapress.com/thinkpython2/html/thinkpython2019.html

Session 8: Object Oriented Programming Part II	
Learning Objective	Digging deeper into object oriented programming concepts with inheritance and composition
Required Readings	Chapter 8: Python Programming for the Absolute Beginner by Michael Dawson
Optional Readings	

Session 9: OOP Case Study	
Learning Objective	During this session, we will learn to apply the more complex OOP concepts
Required Readings	Chapter 8: Python Programming for the Absolute Beginner by Michael Dawson

Optional Readings	

Session 10: Documentation and Testing	
Learning Objective	Learning Unit-testing as part of development. Understanding the importance of test-driven programming and writing unit tests
Required Readings	
Optional Readings	https://docs.python.org/2/library/unittest.html

Session 11: Analysis of Algorithms	
Learning Objective	Learning the basics of algorithm analysis, performance and requirements.
Required Readings	
Optional Readings	http://greenteapress.com/thinkpython2/html/thinkpython2022.html.

Session 12: Project Presentations	
Learning Objective	Students will have a chance to present projects and demos to the class.
Required Readings	
Optional Readings	

Final Exam Week: 14 - 18.12.2020 – no class