

Faculty: Science, Technology and Engineering

Course: **Python Language**

Program: Study Abroad in Engineering

Semester: 1 - Fall

ECTS credits: 3

Duration: 22,5 hours

Language of instruction: English

Instructor: Iban Ricart

Course Description

Learn the basics of the Python language, then deepen some specificities of the Python Language Python, and then discover different aspects and diversity of the Python ecosystem.

Prerequisites

Algorithms.

Have already programmed (any language).

Attendance and punctuality policies

Attendance is mandatory for all classes. Any presentation or activity missed due to student absences can only be rescheduled in cases of certified medical or family emergencies. If a student misses more than three classes in any course half a letter grade will be deducted from the final grade for each additional absence. Seven absences in 6 ECTS courses or four absences in 3 ECTS courses will result in a Fail grade. Notice that there is a minimum of 80% attendance.

Students will be marked ABSENT from any class if they arrive more than 20 minutes late. Students will not be permitted to enter the class unless the professor specifically accepts it. Even if the instructor allows students to join the class, they will still be marked as absent for that lesson.

Absences can be justified in the following cases: Death of a first-degree relative, serious illness of the student/ first-degree relative or obligation to attend legal affairs. In all these cases a document or receipt must be sent via e-mail to studyabroad@uvic.cat adding your professor in copy. Important! In case of injury/ illness of the student, a medical document

issued in Vic* needs to be provided.

*Medical documents accepted: physical doctor's notes which contain the hospital's stamp and signature in handwriting OR digital doctor's notes which contain the doctor's valid digital signature (a digital signature is valid when it shows the authentication of the person who signs and prevents the pdf to be modified after being signed).

Learning outcomes

1- Mastery of Python basics

2- Good understanding and a broad vision of the Python ecosystem

Method of presentation

- Lectures and discussions: Lectures with appropriate visual support provide the theoretical content of the sessions. Class discussions facilitate the students' ability to connect reading and lectures, analyzing or applying concepts.
- Class participation: Students are expected to participate in group activities and in the discussions based on the course readings and cases proposed.
- Home exercises: Students are expected to solve several exercises during the semester.

Required work and assessment methods

- Cases, reading and exercises (70%). Preparation, development and discussions related to exercises will be highly valuable for the success of the course both at individual and group level.
- Validation exams (30%). 2 Individual tests

Required work and assessment methods

Act	E1	E2	E3	E4	E5	E6	E7	E8	T1	T2
%	7	7	7	7	7	7	15	13	15	15

The minimum grade to pass is 5.

Retake exams and activities

The student must present the activities pending to delivery (E1 from E8). If the student passed the tests T1 and T2, it is not necessary to take the retake test. It is only mandatory to take the retake test if one or both tests were not passed.

The activities weight in the retake evaluation is the same as activities weight in the ordinary evaluation (see the percentages table), but the maximum grade is 5)

Contents

The content has been adapted for third year students to better match the Data Science and AI module taught jointly.

- Learn the basics of Python
- Python and IDEs: Anaconda, Spyder, Jupyter and Colab
- Deepen structures, classes, understanding
- Vectorization, numpy, matplotlib, pandas
- Learn by topic: Scientific computing
- Learn by topic: GUI
- Learn by topic: DataViz (plotly, folium, ..)
- Python frameworks: Django, Flask, ...

Unit One. Basics of Python and Environments

Week 1. Course presentation. Python language with Google Colab.

Week 2. Introduction to Python IDEs (E1. Install a Python Environment and Python Exercises).

Unit Two. Python structures and Object Oriented Programming.

Week 3. Python data structures.

Week 4. Object Oriented Programming with Python (E2. Structures and objects Python Exercises).

Unit Three. Vectorization, plots and data manipulation.

Week 5. Vectorization with Numpy. (E3. Numpy)

Week 6. Implementing plots with Matplotlib. (E4. Matplotlib)

Week 7. Using Pandas to manage data. (E5. Pandas)

Unit Four. Data Visualization with Python

Week 8. Data Visualization tools (Plotly, Folium ...) (E6. Data Visualization)

Unit Five. First individual test

week 9. Individual Test. Introduction to Python and structures. (T1)

Unit Six. GUI with Python

Week 10. Implementing GUIs with Python.

Week 11. Python project with GUI (E7. Python project with GUI)

Unit Seven. Introduction to Frameworks with Python

Week 12. Web Frameworks (Django and Flask)

Week 13. Multiplatform Frameworks. (PyQT). (E8. Mini Python Framework Project)

Unit Eight. Finalizing the course

Week 14. Revising contents.

Week 15. Final Test (T2).