



COURSE DESCRIPTION CARD - SYLLABUS

Course name

Object Oriented Programming [S2Elmob1-SSP>PO2]

Course

Field of study

Electromobility

Year/Semester

2/3

Area of study (specialization)

Car Onboard Systems

Profile of study

general academic

Level of study

second-cycle

Course offered in

Polish

Form of study

full-time

Requirements

compulsory

Number of hours

Lecture

0

Laboratory classes

0

Other

0

Tutorials

0

Projects/seminars

15

Number of credit points

1,00

Coordinators

dr inż. Stanisław Mikulski

stanislaw.mikulski@put.poznan.pl

dr inż. Damian Burzyński

damian.burzynski@put.poznan.pl

Lecturers

Prerequisites

Basics knowledge about programming in any programming language, including issues such as: loops, conditional statements, data types, functions. Basic knowledge of mathematics on the following issues: approximation, matrix calculus and statistical analysis.

Course objective

The aim of the course is to broaden the knowledge of object-oriented programming techniques, in such a way that the student can using already developed Python packages and create object-oriented software characterized by high agility and efficiency.

Course-related learning outcomes

Knowledge:

The student has extended knowledge in the field of programming techniques and the use of modern IT tools for the analysis and synthesis of electric systems, hybrid and electric vehicles, including traction

Skills:

The student is able to use modern information and communication tools, advanced programming techniques and machine learning methods to collect, process and analyze data

Social competences:

The student understands that in the area of technology, knowledge and skills quickly devalue, which requires constant replenishment

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Project:

through tasks performed independently during project classes.

Programme content

Project:

Programming paradigms, code execution mechanisms, code interpretation vs. compilation, discussion of the basic structure of Python, including classes, objects, abstraction, inheritance and encapsulation. Discussion of selected design patterns and their exemplary implementation. UML class diagrams.

Course topics

The following topics will be discussed during the project classes:

- 1) design patterns: their division and application
- 2) implementation of selected design patterns in Python
- 3) unit tests: reason for use and the unittest package
- 3) development of stand-alone object-oriented software

Teaching methods

Project:

Presentations, quizzes and e-learning tasks, initiating discussions during project classes.

Bibliography

Basic:

1. Johansson R., Matematyczny Python. Obliczenia naukowe i analiza danych z użyciem NumPy, SciPy i Matplotlib, Helion 2021
2. Python documentation, <https://docs.python.org/3/>
3. Lott S. F., Phillips D., Programowanie zorientowane obiektowo w Pythonie. Tworzenie solidnych i łatwych w utrzymaniu aplikacji i bibliotek. Wydanie IV, Helion 2023

Additional:

1. Chollet F., Deep learning. Praca z językiem Python i biblioteką Keras, Helion 2019

Breakdown of average student's workload

	Hours	ECTS
Total workload	29	1,00
Classes requiring direct contact with the teacher	15	0,50
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	14	0,50