

Scripting languages - course description

General information	
Course name	Scripting languages
Course ID	11.3-WE-INFP-ScripLang
Faculty	Faculty of Computer Science, Electrical Engineering and Automatics
Field of study	Computer Science
Education profile	academic
Level of studies	First-cycle Erasmus programme
Beginning semester	winter term 2021/2022

Course information	
Semester	4
ECTS credits to win	6
Course type	obligatory
Teaching language	polish
Author of syllabus	<ul style="list-style-type: none">dr inż. Piotr Witczak

Classes forms					
The class form	Hours per semester (full-time)	Hours per week (full-time)	Hours per semester (part-time)	Hours per week (part-time)	Form of assignment
Lecture	30	2	-	-	Exam
Laboratory	30	2	-	-	Credit with grade

Aim of the course

- To familiarize students with scripting languages (in particular, Python)
- Developing skills in the practical application of scripting languages
- Developing skills in using ready-made libraries and frameworks

Prerequisites

- The basics of programming
- Object-oriented programming

Scope

- the scripting and interpreted languages paradigm
- differences between interpreted and compiled languages (on the example of Python and C++)
- the use of scripting languages to create websites
- using scripting languages in creating modern software
- use of scripting languages for intelligent calculations

Teaching methods

- lecture: conventional lecture
- laboratory: lab exercises, group work, programming in pairs,

Learning outcomes and methods of their verification

Outcome description	Outcome symbols	Methods of verification	The class form
Potrąfi zastosować techniki skryptowe do realizacji zadania będącego częścią większego projektu lub systemu informatycznego		<ul style="list-style-type: none">• an observation and evaluation of the student's practical skills	<ul style="list-style-type: none">• Laboratory
Ma wiedzę na temat trendów rozwojowych dyscypliny oraz potrzeby usprawniania codziennych czynności informatycznych poprzez stosowanie języków skryptowych		<ul style="list-style-type: none">• an exam - oral, descriptive, test and other	<ul style="list-style-type: none">• Lecture
Potrąfi zastosować języki skryptowe do rozwiązania problemu naukowego oraz inżynierskiego		<ul style="list-style-type: none">• an observation and evaluation of the student's practical skills	<ul style="list-style-type: none">• Laboratory
Rozumie paradygmat języków skryptowych oraz interpretowanych oraz różnice w stosunku do języków kompilowanych		<ul style="list-style-type: none">• an exam - oral, descriptive, test and other	<ul style="list-style-type: none">• Lecture
Rozumie potrzebę stosowania języków skryptowych		<ul style="list-style-type: none">• a discussion	<ul style="list-style-type: none">• Lecture

Outcome description	Outcome symbols	Methods of verification	The class form
Dokonyuje wyboru właściwych narzędzi oraz rozwiązań skryptowych w zależności od wymagań projektowych		<ul style="list-style-type: none"> an observation and evaluation of the student's practical skills 	<ul style="list-style-type: none"> Laboratory

Assignment conditions

Lecture - the condition of passing is obtaining a positive grade from the exam

Laboratory - the condition of passing is obtaining positive grades from all laboratory exercises, planned to be implemented under the laboratory program

Components of the final grade = lecture: 50% + laboratory: 50%

Recommended reading

1. Learning Python: Powerful Object-Oriented Programming, Mark Lutz, "O'Reilly Media, Inc.", 2013, 9781449355692
2. Python Crash Course: A Hands-On, Project-Based Introduction to Programming, Eric Matthes, No Starch Press, 2015, ISBN: 9781593276034
3. Deep Learning with TensorFlow: Explore neural networks and build intelligent systems with Python, 2nd Edition, Giancarlo Zaccone, Md. Rezaul Karim, Packt Publishing Ltd, 2018, ISBN: 9781788831833
4. Beginning Django: Web Application Development and Deployment with Python, Daniel Rubio, Apress, 2017 ISBN: 978148422787
5. Practical Python Design Patterns: Pythonic Solutions to Common Problems, Wessel Badenhorst, Apress, 2017, ISBN: 9781484226803

Further reading

1. Python Cookbook, Alex Martelli, Anna Ravenscroft, David Ascher, "O'Reilly Media, Inc.", 2005, ISBN: 9780596554743
2. Practical Django 2 and Channels 2: Building Projects and Applications with Real-Time Capabilities, Federico Marani, Apress, 2018, ISBN: 9781484240991
3. Pro Python 3: Features and Tools for Professional Development, J. Burton Browning, Marty Alchin, Apress, 2019, ISBN: 9781484243855
4. Fluent Python: Clear, Concise, and Effective Programming, Luciano Ramalho, "O'Reilly Media, Inc.", 2015, ISBN: 9781491946251
5. Python Continuous Integration and Delivery: A Concise Guide with Examples, Moritz Lenz, Apress, 2018, 9781484242810
6. Clean Architecture: A Craftsman's Guide to Software Structure and Design, *Martin, Robert C*, Prentice Hall, 2018, 9780134494166
7. Clean Python: Elegant Coding in Python, Sunil Kapil, Apress, 2019, ISBN: 9781484248782
8. Effective Python: 59 Specific Ways to Write Better Python, Brett Slatkin, Addison-Wesley Professional, 2015, ISBN: 9780134034409
9. Scripting Languages: Automating the Web, Rohit Khare, O'Reilly, 1997, ISBN: 9781565922655
10. Software Architecture: A Comprehensive Framework and Guide for Practitioners, Oliver Vogel, Ingo Arnold, Arif Chughtai, Timo Kehrer, Springer Science & Business Media, 2011, ISBN: 9783642197369

Notes

Modified by dr inż. Piotr Witczak (last modification: 02-09-2021 08:48)

Generated automatically from SyllabUZ computer system