

Subject card

Subject name and code	Scripting languages, PG_00143818						
Field of study	Informatics						
Date of commencement of studies	October 2025		Academic year of realisation of subject		2025/2026		
Education level	Bachelor's studies		Subject group		Obligatory subject group in the field of study Subject group related to scientific research in the field of study		
Mode of study	part-time studies		Mode of delivery		at the university		
Year of study	1		Language of instruction		Polish		
Semester of study	2		ECTS credits		8.0		
Learning profile	academic		Assessment form		exam		
Conducting unit							
Name and surname of lecturer (lecturers)	Subject supervisor		dr Tomasz Borzyszkowski				
	Teachers						
Lesson types	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	30.0	0.0	30.0	0.0	0.0	60
	E-learning hours included: 0.0						
Learning activity and number of study hours	Learning activity	Participation in didactic classes included in study plan		Participation in consultation hours		Self-study	SUM
	Number of study hours	60		0.0		140.0	200
Subject objectives	To teach students the basic programming constructs available in scripting languages using the Python language as an example.						
Learning outcomes	Course outcome		Subject outcome		Method of verification		
	[INFL3_U06] is able to design, create, run and test programs using dedicated tools and adequate templates		The student understands the principles of scripting programming languages and knows how the basic programming constructs work in these languages		[SU8] observation of student's independent or team work		
	[INFL3_U09] is able to assess the suitability of programming paradigms and tools for solving problems of various types		The student is able to use selected libraries of the Python language to solve simple programming problems		[SU1] oral statement/conversation/discussion [SU8] observation of student's independent or team work		
	[INFL3_W05] has general knowledge of various programming paradigms and programming languages; has detailed knowledge of object-oriented design and programming methods and patterns		The student understands the principles of scripting programming languages and knows how the basic programming constructs work in these languages		[SW4] test/exam - oral or written [SW2] presentation/project/paper/report		

Subject contents	1. Basic concepts: compilers and interpreters, basic data types, dynamic data types. 2. Functions: function construction in Python, documenting functions, filtering and lambda-functions. 3. Modules and packages: the concept of namespaces, definition of a module and how to import it, creating packages. 4. Classes and objects: class definition and instance creation, inheritance and attributes, special methods. 5. Files and exceptions: basic file operations, the pickle module, the try statement and its sections, creating and raising exceptions. 6. Regular expressions: basic constructions, compilation of expressions, groups and subgroups. 7. HTML language processing: sgmlib.py library, locals() and globals() functions, Dialectizer example. 8. The unittest module: testing the correctness, errors and soundness of a programme, unit tests vs. program patching.		
Prerequisites and co-requisites	None		
Assessment methods and criteria	Subject passing criteria	Passing threshold	Percentage of the final grade
	Activity in class	0.0%	10.0%
	project presentation, observation of student work	51.0%	50.0%
	Exam	51.0%	40.0%
Recommended reading	Basic literature	1. Guido van Rossum, Python Tutorial, http://docs.python.org/tut/. 2. Mark Pilgrim, Dive into Python. http://diveintopython.org/. 3. Bruce Eckel, Thinking in Python, http://www.mindview.net/Books/TIPython. 4. Python's official documentation, http://docs.python.org/.	
	Supplementary literature	None	
	eResources addresses		
Example issues/ example questions/ tasks being completed			
Work placement	Not applicable		

Document generated electronically. Does not require a seal or signature.