

Scripting languages - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Scripting languages
Kod przedmiotu	11.3-WE-INF-D-ScripLang-Er
Wydział	Wydział Informatyki, Elektrotechniki i Automatyki
Kierunek	Informatyka
Profil	ogólnoakademicki
Rodzaj studiów	Program Erasmus drugiego stopnia
Semestr rozpoczęcia	semestr zimowy 2020/2021

Informacje o przedmiocie	
Semestr	2
Liczba punktów ECTS do zdobycia	5
Typ przedmiotu	obieralny
Język nauczania	angielski
Sylabus opracował	• dr inż. Grzegorz Bazydło • dr hab. inż. Remigiusz Wiśniewski, prof. UZ • dr inż. Iwona Grobelna

Formy zajęć					
Forma zajęć	Liczba godzin w semestrze (stacjonarne)	Liczba godzin w tygodniu (stacjonarne)	Liczba godzin w semestrze (niestacjonarne)	Liczba godzin w tygodniu (niestacjonarne)	Forma zaliczenia
Wykład	15	1	-	-	Zaliczenie na ocenę
Laboratorium	30	2	-	-	Zaliczenie na ocenę
Projekt	15	1	-	-	Zaliczenie na ocenę

Cel przedmiotu

- Familiarize students with scripting languages for their practical use.
- Shaping basic skills for improving daily computer tasks (at home, work) using scripting languages.

Wymagania wstępne

Basic knowledge of operating systems (Windows, Linux), computer networks and websites design.

Zakres tematyczny

- Virtualization, virtual machines, run the system environment from another operating system.
- Tips and tricks of process management and automation tasks in Linux and Windows systems, practical use of the command line and scripting languages (e.g., Bash, CMD) in order to improve the efficiency of the usage of an operation system. Lifehacking.
- Improve performance of the use of MS Office software, introduction to the VBA language. Markup languages (XML, HTML, XHTML). Websites design using Cascading Style Sheets (CSS), design of the document structure, using block elements, inserting embedded hyperlinks, tables and forms, validation of the HTML source.
- Using scripting languages in design of websites and web services. Using PHP language in applications embedded on the server, data structures, fundamentals of the web applications design using frameworks, data transfer using *GET* and *POST* method, session control using cookies. Animating websites using JavaScript language, data operations, using Document Object Model (DOM), dynamic modification of Cascading Style Sheets (CSS).
- Introduction to *Python* language: areas of application (from command line to web services), using data structures and files, implementation of the scripts using command prompt (data security, streamline daily tasks using scripts), using Python language (e.g., *Django* framework).
- Security of web applications based on scripting languages (e.g., *PHP*). Threats from inappropriate use of cookies, data transfer using *POST* or *GET* methods. Protection against injection of malicious code (e.g., *Code Injection*, *Cross-Site Scripting*, *SQL-Injection* attacks).

Metody kształcenia

Lecture: conventional lecture, discussion.

Laboratory: laboratory exercises, work in groups.

Project: project method, discussion.

Efekty uczenia się i metody weryfikacji osiągnięcia efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Is able to select the right tools and scripting solutions depending on the design requirements.		• projekt	• Projekt

Opis efektu	Symbol e efektów	Metody weryfikacji	Forma zajęć
Is able to apply scripting languages to solve scientific and engineering problems.		- bieżąca kontrola na zajęciach - projekt - sprawdzian	- Laboratorium - Projekt
Knows current trends and the need to streamline daily computer tasks through using scripting languages.		- dyskusja - kolokwium	Wykład
Understands the need of using scripting languages.		- bieżąca kontrola na zajęciach - sprawdzian	Laboratorium
Is able to apply scripting techniques to perform tasks that are a part of a larger project or IT system.		- bieżąca kontrola na zajęciach - projekt	- Laboratorium - Projekt

Warunki zaliczenia

Lecture – the passing condition is to obtain a positive mark from the final test.

Laboratory – the passing condition is to obtain positive marks from all laboratory exercises to be planned during the semester.

Project – the passing condition is to obtain a positive mark from all projects conducted during the semester.

Final mark components: lecture 30% + laboratory 40% + project 30%.

Literatura podstawowa

1. C. Albing, JP Vossen, C. Newham, bash Cookbook: Solutions and Examples for bash Users, O'Reilly Media, 2007.
2. Chromatic, D. Conway, C. Poe, Perl Hacks: Tips & Tools for Programming, Debugging, and Surviving, O'Reilly Media, 2006.
3. S. Stefanov, JavaScript for PHP Developers, O'Reilly Media, 2013.
4. P. Barry, Head First Python: A Brain-Friendly Guide, 2nd Edition, O'Reilly Media, 2016.
5. E. Freeman, E. Freeman, Head First HTML and CSS: A Learner's Guide to Creating Standards-Based Web Pages, O'Reilly Media, 2012.
6. A. Pash, G. Trapani, Lifehacker: The Guide to Working Smarter, Faster, and Better, Third Edition, Wiley, 2011.
7. A. Pash, Gina Trapani, Lifehacker: The Guide to Working Smarter, Faster, and Better, part 2, Wiley, 2011.

Literatura uzupełniająca

1. M. Lutz, Learning Python, 5th Edition, O'Reilly Media, 2013.
2. J. Forcier, P. Bissex, W. Chun, Python Web Development with Django (Developer's Library), Addison-Wesley Professional, 2008.
3. M. Dawson, Python Programming for the Absolute Beginner, 3rd Edition, Course Technology, 2010.
4. E. T. Freeman, E. Robson, Head First HTML5 Programming: Building Web Apps with JavaScript, O'Reilly Media, 2011.
5. L. Welling, L. Thomson, PHP i MySQL. PHP and MySQL Web Development (5th Edition), Addison-Wesley Professional, 2016.
6. S. Armand, Extending Symfony 2 Web Application Framework, Packt Publishing, 2014.

Uwagi

Zmodyfikowane przez dr inż. Grzegorz Bazydło (ostatnia modyfikacja: 27-04-2020 20:58)

Wygenerowano automatycznie z systemu SylabUZ