

Object-oriented design and programming - course description

General information	
Course name	Object-oriented design and programming
Course ID	11.3-WE-BEP-PPO
Faculty	Faculty of Computer Science, Electrical Engineering and Automatics
Field of study	E-business
Education profile	practical
Level of studies	First-cycle studies leading to Engineer's degree
Beginning semester	winter term 2016/2017

Course information	
Semester	1
ECTS credits to win	5
Course type	obligatory
Teaching language	polish
Author of syllabus	- dr inż. Mariusz Jacyno - dr inż. Andrzej Marciniak

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	18	1,2	Credit with grade
Laboratory	30	2	18	1,2	Credit with grade

Aim of the course

Prezentowanie podstawowych pojęć związanych z programowaniem i projektowaniem obiektowym oraz ich realizacją w środowisku Java. Zapoznanie z nowoczesnymi środowiskami deweloperskimi wspierającymi wytwarzanie oprogramowania na platformie Java.

Prerequisites

Brak

Scope

Podstawy kompilacji i uruchamiania programów na platformie Java. Omówienie środowiska Java Development Kit oraz zintegrowanych środowisk deweloperskich IDE dla platformy Java.

Podstawy programowania imperatywnego i strukturalnego w języku Java. Typy danych, zmienne proste i referencyjne, literały, operatory, tablice, instrukcje sterujące, zasięg widoczności zmiennych, funkcje, właściwości zmiennych.

Podstawy programowania obiektowego w języku Java. Klasy i instancje, typy wyliczeniowe, pakiety, właściwości klas i metod. Zasady konstrukcji obiektów i poznanie mechanizmu czyszczenia pamięci (kolektora śmieci).

Mechanizmy i właściwości programowania obiektowego. Dziedziczenie, polimorfizm i enkapsulacja. Projektowanie złożonych typów obiektowych przy użyciu kompozycji i dziedziczenia.

Zaawansowane techniki obiektowe. Tworzenie interfejsów programistycznych z wykorzystaniem klas abstrakcyjnych i interfejsów. Rozszerzanie interfejsów. Klasy wewnętrzne i klasy statycznie zagnieżdżone.

Obsługa narzędzi deweloperskich na platformie Java. Tworzenia dokumentacji API w środowisku Javy. Archiwizacja programów i bibliotek Javy. Mechanizmy wdrażania i automatycznej instalacji programów. Obsługa debugera. Podstawy tworzenia programów odpornych na błędy. Metody walidacji danych, obsługa sytuacji wyjątkowych.

Wybrane zagadnienia implementacyjne języka Java. Klasy użytkowe, klasy strumieniowe do obsługi systemu wejścia i wyjścia, przechowywanie obiektów w kolekcjach.

Programowanie wielowątkowe w języku Java. Tworzenie i synchronizacja działania wątków. Mechanizm synchronizacji wewnątrzprocesowej Hoare'a, wyłączenie wątków, segmentacja czasu, priorytety.

Podstawy tworzenia aplikacji sieciowych w języku Java. Programowanie gniazd sieciowych, architektura klient-serwer, implementacja serwerów, identyfikacja zasobów sieciowych.

Teaching methods

Wykład - wykład konwencjonalny z wykorzystaniem wideoprojektora.

Laboratorium - zajęcia praktyczne w laboratorium komputerowym.

Learning outcomes and methods of their verification

Outcome description	Outcome symbols	Methods of verification	The class form
---------------------	-----------------	-------------------------	----------------

Outcome description	Outcome symbols	Methods of verification	The class form
Zna zasady projektowania i programowania obiektowego	• K_W03 • K_W10	• a quiz	• Lecture
Ma wiedzę na temat procesu wytwarzania oprogramowania w oparciu o technologie Java,	• K_W11	• a quiz	• Lecture
Zna definicje podstawowych paradygmatów programowania	• K_W09	• a quiz	• Lecture
Potrafi skompilować i uruchomić samodzielnie napisaną aplikację w języku Java	• K_U14	• a preparation of a project	• Laboratory
Potrafi wytworzyć i wdrożyć aplikację internetową, oraz wytworzyć niezbędną dokumentację(API, wdrożeniową)	• K_U11	• a preparation of a project	• Laboratory

Assignment conditions

Wykład - sprawdzian w formie pisemnej i/lub ustnej, realizowany na koniec semestru.

Laboratorium – ocena końcowa stanowi sumę ważoną ocen uzyskanych za realizację poszczególnych ćwiczeń laboratoryjnych oraz sprawdzianów kontrolnych weryfikujących przygotowanie merytoryczne do ćwiczeń.

Ocena końcowa = 50 % oceny zaliczenia z formy zajęć wykład + 50 % oceny zaliczenia z formy zajęć laboratorium.

Recommended reading

1. Eckel, B., Thinking in Java, Wydanie IV, Warszawa, Helion, 2006.
2. Horstmann, C.S., Cornell, G., Java. Techniki zaawansowane, Wydanie dziewiąte, Gliwice, Helion, 2013.
3. Horstmann, C.S., Cornell, G.: Java., Podstawy, Wydanie dziewiąte, Gliwice, Helion, 2013.

Further reading

1. Lis, M., Praktyczny kurs Java, Wydanie II, Gliwice, Helion, 2004.
2. Naughton, P., Podręcznik języka programowania Java, Poznań, Nakom, 1999.

Notes

Modified by dr hab. inż. Marcin Mrugalski, prof. UZ (last modification: 03-10-2016 00:03)

Generated automatically from SyllabUZ computer system