

## KARTA KURSU

Nazwa	<b>Programowanie aplikacji internetowych (Java)</b>
Nazwa w j. ang.	Web application programming (Java)

Koordynator	dr Wojciech Gwizdała	Zespół dydaktyczny
Punktacja ECTS*	st. stacjonarne: 5 st. niestacjonarne: 5	Zespół dydaktyczny: dr Wojciech Gwizdała mgr Patryk Mazurek

### Opis kursu (cele kształcenia)

Celem kursu jest nauczanie studentów podstaw programowania oraz tworzenia aplikacji internetowych w języku Java. W ramach kursu definiowana jest składnia języka oraz omawiane są podstawowe zagadnienia związane z programowaniem obiektowym w języku Java, a także zagadnienia dotyczące tworzenia aplikacji internetowych. Kurs prowadzony jest w języku polskim.

### Warunki wstępne

Wiedza	Podstawy działania oraz implementacji algorytmów, kompilatory, składnia i semantyka języka programowania, pojęcia oraz idee programowania obiektowego
Umiejętności	Programowanie w językach C i C++ z zastosowaniem prostych i złożonych struktur danych, projektowanie, pisanie i testowanie programów zorientowanych obiektowo z użyciem odpowiednich edytorów i kompilatorów
Kursy	Programowanie, Algorytmy i struktury danych, Języki i narzędzia programowania obiektowego

### Efekty uczenia się

	Efekt uczenia się dla kursu	Odniesienie do efektów dla specjalności (określonych w karcie programu studiów dla specjalności)
Wiedza	Po zakończeniu kursu	-
	W01: w zaawansowanym stopniu fakty i teorie stanowiące wiedzę w zakresie zabezpieczania architektury systemów komputerowych, systemów operacyjnych i urządzeń sieciowych.	K_W02
	W02: elementarne algorytmy, języki i techniki programowania oraz zasady projektowania systemów baz danych w kontekście wymagań bezpieczeństwa	K_W03
	W03: zasady działania aplikacji i usług elektronicznych w Internecie i w sieciach lokalnych ze szczególnym uwzględnieniem aspektów bezpieczeństwa	K_W06

Umiejętności	Efekt uczenia się dla kursu	Odniesienie do efektów dla specjalności (określonych w karcie programu studiów dla specjalności)
	Po zakończeniu kursu	
	U01: korzystać z nowoczesnych narzędzi IT w zakresie planowania, budowania i eksploatacji sieci komputerowych o lokalnym i rozszerzonym zasięgu w oparciu o zasady bezpieczeństwa funkcjonowania tych struktur	K_U01
	U02: używać dedykowanych środowisk programistycznych wraz z wybranymi bibliotekami w celu efektywnego i bezpiecznego tworzenia aplikacji desktopowych, mobilnych czy internetowych	K_U03
	U03: konstruować algorytmy i pisać pojedyncze aplikacje oraz większe projekty programistyczne, w oparciu o języki programowania niskiego i wysokiego poziomu z uwzględnieniem zasad bezpieczeństwa	K_U04
	U04: pracować indywidualnie lub w zespole (m.in. opracować dokumentację, przedstawić prezentację i prowadzić dyskusję na temat zadania, projektu lub zagadnień w szczególności zw. z cyberbezpieczeństwem, również w jęz. obcym) oraz planować pracę, a także komunikować się przy użyciu technik właściwych dla branży IT	K_U06

Kompetencje społeczne	Efekt uczenia się dla kursu	Odniesienie do efektów dla specjalności (określonych w karcie programu studiów dla specjalności)
	Po zakończeniu kursu	
	K01: inicjowania działań na rzecz współdziałania z innymi osobami w ramach prac zespołowych i podejmowania różnych wiodących ról w interdyscyplinarnych zespołach zajmujących się analizowaniem cyberbezpieczeństwa oraz myślenia i działania w sposób przedsiębiorczy	K_K01
	K02: krytycznej oceny poziomu swojej wiedzy oraz ciągłego dokształcania się i konsultacji z innymi ekspertami z branży IT w szczególności związanej z cyberbezpieczeństwem, a także planowania własnego rozwoju zawodowego	K_K02

### Studia stacjonarne

Organizacja											
Forma zajęć	Wykład (W)	Ćwiczenia w grupach									
		A		K		L		S		P	E
Liczba godzin	20					30					

## Studia niestacjonarne

Organizacja												
Forma zajęć	Wykład (W)	Ćwiczenia w grupach										
		A		K		L		S		P		E
Liczba godzin	15				20							

### Opis metod prowadzenia zajęć

Kurs składa się z laboratoriów praktycznych oraz wykładów, w trakcie których studenci zapoznają się z podstawami programowania w języku Java oraz zasadami tworzenia aplikacji internetowych. Uzupełnieniem zajęć są zadania problemowe udostępniane na platformie e-learningowej do samodzielnego rozwiązania.

### Formy sprawdzania efektów uczenia się

	E – learning	Gry dydaktyczne	Ćwiczenia w szkole	Zajęcia terenowe	Praca laboratoryjna	Projekt indywidualny	Projekt grupowy	Udział w dyskusji	Referat	Praca pisemna (esej)	Egzamin ustny	Egzamin pisemny	Inne
W01					X			X				X	
W02					X			X				X	
W03					X			X				X	
U01					X	X						X	
U02					X	X						X	
U03					X	X						X	
U04					X	X						X	
K01					X							X	
K02					X							X	

Kryteria oceny	Skala ocen na podstawie uzyskanych punktów z egzaminu: 92% - 100%    bardzo dobry (5,0) 84% - 91,99%    dobry plus (4,5) 76% - 83,99%    dobry (4,0) 68% - 75,99%    dostateczny plus (3,5) 60% - 67,99%    dostateczny (3,0) 0% - 59,99%    niedostateczny (2,0)
----------------	---

Uwagi	Student może przystąpić do egzaminu po uzyskaniu zaliczenia z laboratorium.
-------	---

### Treści merytoryczne (wykaz tematów)

1. Podstawy programowania w języku Java, składnia, kompilacja, uruchamianie programów.
2. Pojęcie klasy, hermetyzacja, dziedziczenie, polimorfizm.
3. Wykorzystanie konstruktorów domyślnych oraz tworzenie konstruktorów z wykorzystaniem mechanizmu przeciążania.
4. Dostęp: publiczny, chroniony, prywatny i pakietowy do pól i metod.
5. Metody i pola statyczne.
6. Obsługa sytuacji wyjątkowych i zarządzanie zasobami, bloki "try..catch", "try...finally" i „try z zasobami”.
7. Typy generyczne i kolekcje.
8. Serwer aplikacji (kontener): instalacja, zarządzanie, wdrażanie aplikacji.
9. Integracja kontenera Java Enterprise Edition z wybranym środowiskiem programistycznym.
10. Serwlety (Java Servlet): obiekty żądania i odpowiedzi, obsługa sesji, ciasteczka.
11. Baza danych: połączenie (JDBC) i obsługa zapytań z poziomu serwletów oraz stron JSP.
12. JSF (JavaServer Faces): nawigacja, konwertery, walidatory.

13. JPA (Java Persistence API): encje i zapis do bazy danych.
14. Wprowadzenie do Spring i Spring Boot, konfiguracja projektu.
15. Tworzenie REST API, komponenty Spring.
16. Warstwa danych Spring Data JPA.

#### Wykaz literatury podstawowej

1. Cay S. Horstmann, Gary Cornell, Java Podstawy, wyd. 11, Helion 2019
2. Herbert Schildt, Java. Kompendium programisty, Helion 2005
3. Krzysztof Barteczko, Java Programowanie praktyczne od podstaw, wydawnictwo naukowe PWN SA, 2014
4. Wojciech Romowicz, Java Server Pages : oraz inne komponenty JavaPlatform, Helion 2001
5. Marty Hall, Larry Brown, Java Servlet i JavaServer Pages. Tom 1. Wydanie II, Helion, 2005
6. David Geary, Cay S. Horstmann, JavaServer Faces. Wydanie II, Helion, 2008
7. Krzysztof Rychlicki-Kicior, Java EE 6 : programowanie aplikacji WWW, Helion, 2010

#### Wykaz literatury uzupełniającej

1. Eckel Bruce, Thinking in Java, Helion 2006
2. John Ferguson Smart, Java: praktyczne narzędzia, Helion, 2009
3. <http://www.oracle.com/technetwork/java/javaee/javaserverfaces-139869.html>
4. Cay S. Horstmann, Gary Cornell, Java Techniki Zaawansowane, wyd. 9. Helion 2014
5. Robert C. Martin, Czysty kod : podręcznik dobrego programisty, Helion 2014
6. Walls Craig, Spring w akcji, Helion 2019

#### Bilans godzinowy zgodny z CNPS (Całkowity Nakład Pracy Studenta) studia stacjonarne

Ilość godzin w kontakcie z prowadzącymi	Wykład	20
	Konwersatorium (ćwiczenia, laboratorium itd.)	30
	Pozostałe godziny kontaktu studenta z prowadzącym	5
Ilość godzin pracy studenta bez kontaktu z prowadzącymi	Lektura w ramach przygotowania do zajęć	15
	Przygotowanie krótkiej pracy pisemnej lub referatu po zapoznaniu się z niezbędną literaturą przedmiotu	
	Przygotowanie projektu lub prezentacji na podany temat (praca w grupie)	15
	Przygotowanie do egzaminu	10
Ogółem bilans czasu pracy		95
Ilość punktów ECTS w zależności od przyjętego przelicznika		5

#### Bilans godzinowy zgodny z CNPS (Całkowity Nakład Pracy Studenta) studia niestacjonarne

Ilość godzin w kontakcie z prowadzącymi	Wykład	15
	Konwersatorium (ćwiczenia, laboratorium itd.)	20
	Pozostałe godziny kontaktu studenta z prowadzącym	5
Ilość godzin pracy studenta bez kontaktu z prowadzącymi	Lektura w ramach przygotowania do zajęć	25
	Przygotowanie krótkiej pracy pisemnej lub referatu po zapoznaniu się z niezbędną literaturą przedmiotu	
	Przygotowanie projektu lub prezentacji na podany temat (praca w grupie)	20
	Przygotowanie do egzaminu	10
Ogółem bilans czasu pracy		95
Ilość punktów ECTS w zależności od przyjętego przelicznika		5