

**AGH**AGH UNIVERSITY OF SCIENCE  
AND TECHNOLOGY

Nazwa modułu zajęć:	Programowanie w języku Java				
Rok akademicki:	2019/2020	Kod:	ZIIE-1-412-n	Punkty ECTS:	3
Wydział:	Zarządzania				
Kierunek:	Informatyka i Ekonometria	Specjalność:	—		
Poziom studiów:	Studia I stopnia	Forma studiów:	Niestacjonarne		
Język wykładowy:	Polski	Profil:	Ogólnoakademicki (A)	Semestr:	4
Strona www:	—				
Prowadzący moduł:	dr inż. Duda Jerzy (jduda@zarz.agh.edu.pl)				

### Treści programowe zapewniające uzyskanie efektów uczenia się dla modułu zajęć

Moduł poświęcony jest podstawom programowania obiektowego w środowisku Java. Po jego ukończeniu studenci będą mogli tworzyć samodzielne aplikacje klienckie na komputery stacjonarne oraz potencjalnie aplikacje internetowe i dla systemu Android.

### Opis efektów uczenia się dla modułu zajęć

Kod MEU	Student, który zaliczył moduł zajęć zna i rozumie/potrafi/jest gotów do	Powiązania z KEU	Sposób weryfikacji i oceny efektów uczenia się osiągniętych przez studenta w ramach poszczególnych form zajęć i dla całego modułu zajęć
Wiedza: zna i rozumie			
M_W001	zna i rozumie ogólne zasady tworzenia aplikacji desktopowych (konsola, Swing) i sieciowych (JSP) w środowisku Java	IIE1A_W06	Wynik testu zaliczeniowego
M_W002	zna i rozumie podstawy programowania obiektowego w języku Java	IIE1A_W06	Wynik testu zaliczeniowego
Umiejętności: potrafi			
M_U001	potrafi, współdziałając w zespole, zaprojektować i utworzyć prosty graficzny interfejs użytkownika oparty o bibliotekę Swing lub Java FX dla aplikacji wspomagającej wybrany proces zarządzania	IIE1A_U09, IIE1A_U04, IIE1A_K02, IIE1A_U06	Wykonanie ćwiczeń laboratoryjnych, Projekt

M_U002	potrafi zaprojektować i utworzyć aplikację w języku Java bazującą na autonomicznych klasach	IIE1A_U05, IIE1A_U06	Projekt, Wykonanie ćwiczeń laboratoryjnych
--------	---	----------------------	--

### Liczba godzin zajęć w ramach poszczególnych form zajęć

Suma	Forma zajęć dydaktycznych										
	Wykład	Ćwiczenia audytoryjne	Ćwiczenia laboratoryjne	Ćwiczenia projektowe	Konwersatorium	Zajęcia seminaryjne	Zajęcia praktyczne	Zajęcia terenowe	Zajęcia warsztatowe	Prace kontrolne i przejściowe	Lektorat
16	8	0	0	8	0	0	0	0	0	0	0

### Matryca kierunkowych efektów uczenia się w odniesieniu do form zajęć i sposobu zaliczenia, które pozwalają na ich uzyskanie

Kod MEU	Student, który zaliczył moduł zajęć zna i rozumie/potrafi/jest gotów do	Forma zajęć dydaktycznych										
		Wykład	Ćwiczenia audytoryjne	Ćwiczenia laboratoryjne	Ćwiczenia projektowe	Konwersatorium	Zajęcia seminaryjne	Zajęcia praktyczne	Zajęcia terenowe	Zajęcia warsztatowe	Prace kontrolne i przejściowe	Lektorat
Wiedza: zna i rozumie												
M_W001	zna i rozumie ogólne zasady tworzenia aplikacji desktopowych (konsola, Swing) i sieciowych (JSP) w środowisku Java	+	-	-	-	-	-	-	-	-	-	-
M_W002	zna i rozumie podstawy programowania obiektowego w języku Java	+	-	-	-	-	-	-	-	-	-	-
Umiejętności: potrafi												
M_U001	potrafi, współdziałając w zespole, zaprojektować i utworzyć prosty graficzny interfejs użytkownika oparty o bibliotekę Swing lub Java FX dla aplikacji wspomagającej wybrany proces zarządzania	-	-	-	+	-	-	-	-	-	-	-
M_U002	potrafi zaprojektować i utworzyć aplikację w języku Java bazującą na autonomicznych klasach	-	-	-	+	-	-	-	-	-	-	-

## Nakład pracy studenta (bilans punktów ECTS)

Forma aktywności studenta	Obciążenie studenta
Udział w zajęciach dydaktycznych/praktyka	16 godz
Przygotowanie do zajęć	20 godz
przygotowanie projektu, prezentacji, pracy pisemnej, sprawozdania	18 godz
Samodzielne studiowanie tematyki zajęć	20 godz
Egzamin lub kolokwium zaliczeniowe	1 godz
Sumaryczne obciążenie pracą studenta	75 godz
Punkty ECTS za moduł	3 ECTS

## Pozostałe informacje

### Szczegółowe treści kształcenia w ramach poszczególnych form zajęć (szczegółowy program wykładów i pozostałych zajęć)

#### Wykład

##### Wprowadzenie do języka Java i maszyny wirtualnej

Rozwój języka Java. Platformy Java. Środowiska programistyczne. Maszyna wirtualna Java (JVM). Dokumentacja języka Java.

##### Podstawy programowania obiektowego w języku Java

Tworzenie klas w języku Java. Pakiety i ich wykorzystanie. Podstawowe klasy z pakietów java.lang i java.util. Obsługa wyjątków.

##### Zaawansowane techniki programowania obiektowego

Dziedziczenie i interfejsy. Kolekcje i kolekcje generyczne. Klasy anonimowe.

##### Tworzenie GUI w środowisku Java. Obsługa baz danych (JDBC)

Biblioteka AWT. Biblioteka Swing. Podstawowe komponenty GUI. Interfejs Matisse w środowisku NetBeans. Definiowanie połączenia z bazą danych JDBC. Standardowe bazy danych. Obsługa baz danych w środowisku NetBeans

#### Ćwiczenia projektowe

##### Tworzenie i uruchamianie aplikacji konsolowej w języku Java

Środowisko JDK. Kompilator javac. Uruchamianie aplikacji w maszynie wirtualnej. Wykorzystanie środowiska programistycznego NetBeans.

##### Tworzenie aplikacji obiektowej w języku Java

Tworzenie klas i obiektów. Wykorzystanie pakietów java.lang i java.util. Obsługa wyjątków w bloku try..catch.

##### Wykorzystanie klas kolekcji w języku Java

Wykorzystanie tablic. Tablice a kolekcje. Kolekcje generyczne. Pętla foreach.

##### Tworzenie GUI w środowisku NetBeans

Projektowanie GUI z wykorzystaniem interfejsu Matisse. Podstawowe kontrolki. Obsługa zdarzeń.

### Wykonanie projektu

Przygotowanie projektu w grupie 3-4 osobowej, składającego się z kilku klas, wykorzystującego agregację. Dane powinny być zapisywane i pobierane z pliku lub bazy SQL. Na wyższą ocenę konieczne jest zaprojektowanie GUI do obsługi aplikacji.

### **Metody i techniki kształcenia:**

Wykład: Treści prezentowane na wykładzie są przekazywane w formie prezentacji multimedialnej w połączeniu z klasycznym wykładem tablicowym wzbogaconymi o pokazy odnoszące się do prezentowanych zagadnień.

Ćwiczenia projektowe: Studenci wykonują zadany projekt samodzielnie, bez większej ingerencji prowadzącego. Ma to wykształcić poczucie odpowiedzialności za pracę w grupie oraz odpowiedzialności za podejmowane decyzje.

### **Warunki i sposób zaliczenia poszczególnych form zajęć, w tym zasady zaliczeń poprawkowych, a także warunki dopuszczenia do egzaminu:**

1. Wykonanie zadań prezentowanych na ćwiczeniach i przesłanie ich na platformę UPEL.
2. Zaliczenie testu.
3. Wykonanie projektu.

Test poprawkowy zostanie zorganizowany w dodatkowych godzinach, nie wcześniej niż tydzień po pierwszym terminie.

### **Zasady udziału w poszczególnych zajęciach, ze wskazaniem, czy obecność studenta na zajęciach jest obowiązkowa:**

Wykład:

- Obecność obowiązkowa: Nie
- Zasady udziału w zajęciach: Studenci uczestniczą w zajęciach poznając kolejne treści nauczania zgodnie z sylabusem przedmiotu. Studenci winni na bieżąco zadawać pytania i wyjaśniać wątpliwości. Rejestracja audiowizualna wykładu wymaga zgody prowadzącego.

Ćwiczenia projektowe:

- Obecność obowiązkowa: Tak
- Zasady udziału w zajęciach: Studenci wykonują prace praktyczne mające na celu uzyskanie kompetencji zakładanych przez sylabus. Ocenie podlega sposób wykonania projektu oraz efekt końcowy.

### **Sposób obliczania oceny końcowej**

cena końcowa wystawiana jest na podstawie wyniku testu zaliczeniowego (waga 0,5), oddania ćwiczeń laboratoryjnych (waga 0,1) oraz oceny z aplikacji przygotowanej w zespole (projekt, waga 0,4).

### **Sposób i tryb wyrównywania zaległości powstałych wskutek nieobecności studenta na zajęciach:**

Zaległe ćwiczenia student wykonuje samodzielnie i przedstawia wyniki prowadzącemu.

### **Wymagania wstępne i dodatkowe, z uwzględnieniem sekwencyjności modułów**

Wymagane ukończenie modułu Programowanie komputerów.

### **Zalecana literatura i pomoce naukowe**

Literatura:

1. Bruce Eckel: Thinking in Java, Helion 2006
2. Cay Horstmann, Gary Cornell: Java. Podstawy, Helion 2008
3. Marcin Lis: Praktyczny kurs Java, Helion 2007
4. Marcin Lis: Java. Ćwiczenia praktyczne, Helion, 2006
5. Patrick Niemeyer, Jonathan Knudsen: Java. Wprowadzenie, Helion, 2003

6. The Java Tutorials, <http://java.sun.com/docs/books/tutorial/>

Oprogramowanie:

Java SDK wersja 5.0 lub wyższa

NetBeans Java SE wersja 6.0 lub wyższa

### **Publikacje naukowe osób prowadzących zajęcia związane z tematyką modułu**

Semantic search extension based on Polish WordNet relations in business document exploration / Piotr POTIOPA, Michał KARWATOWSKI, Jerzy DUDA, Paweł Sasor, Maciej WIELGOSZ, Bartłomiej Muzykiewicz // W: IML 2017 [Dokument elektroniczny] : international conference on Internet of things and Machine Learning : October 17-18, 2017, Liverpool, United Kingdom (implementacja w Java)

Multi-agent approach to production scheduling in foundries / J. DUDA, A. STAWOWY // Archives of Foundry Engineering / Polish Academy of Sciences. Commission of Foundry Engineering ; ISSN 1897-3310 (wykorzystane środowisko JADE)

### **Informacje dodatkowe**

Wykłady nie są obowiązkowe. Nieobecność na ćwiczeniach projektowych jest dopuszczalna pod warunkiem, że student we własnym zakresie nadrobi zaległości – wykona zadane na platformie UPEL ćwiczenia i prześle rozwiązania. Nieobecność na teście wymaga zaliczenia go na konsultacjach, po wcześniejszym uzgodnieniu terminu.