

SYLABUS
DOTYCZY CYKLU KSZTAŁCENIA 2026/2027 - 2029/2030
Rok akademicki 2026/2027, 2027/2028

1. PODSTAWOWE INFORMACJE O PRZEDMIOCIE

Nazwa przedmiotu	<i>bazy danych</i>
Kod przedmiotu*	
Nazwa jednostki prowadzącej kierunek	<i>Instytut Informatyki, Wydział Nauk Ścisłych i Technicznych</i>
Nazwa jednostki realizującej przedmiot	<i>Instytut Informatyki, Wydział Nauk Ścisłych i Technicznych</i>
Kierunek studiów	<i>sztuczna inteligencja</i>
Poziom studiów	<i>studia I stopnia</i>
Profil	<i>ogólnoakademicki</i>
Forma studiów	<i>stacjonarne</i>
Rok i semestr/y studiów	<i>rok I, semestr 2; rok II, semestr 3</i>
Rodzaj przedmiotu	<i>przedmiot kierunkowy</i>
Język wykładowy	<i>polski</i>
Koordinator	<i>dr hab. Barbara Pękala, prof. UR</i>
Imię i nazwisko osoby prowadzącej / osób prowadzących	<i>dr hab. Barbara Pękala, prof. UR</i>

* - *opcjonalnie, zgodnie z ustaleniami w Jednostce*

1.1. Formy zajęć dydaktycznych, wymiar godzin i punktów ECTS

Semestr (nr)	Wykł.	Ćw.	Konw.	Lab.	Sem.	ZP	Prakt.	Inne (jakie?)	Liczba pkt. ECTS
2	20			30					4
3	30			30					5

1.2. Sposób realizacji zajęć

zajęcia w formie tradycyjnej

1.3 Forma zaliczenia przedmiotu (z toku)

wykład – zaliczenie bez oceny, laboratoria – zaliczenie z oceną

2. WYMAGANIA WSTĘPNE

Rozumienie reguł przetwarzania podstawowych oraz złożonych typów danych w dowolnym języku proceduralnym.
--

3. CELE, EFEKTY UCZENIA SIĘ, TREŚCI PROGRAMOWE I STOSOWANE METODY DYDAKTYCZNE

3.1 Cele przedmiotu

C ₁	Dostarczenie wiedzy i wykształcenie umiejętności projektowania i eksploatacji systemów relacyjnych baz danych z wykorzystaniem języków strukturalnego i proceduralnego.
C ₂	Dostarczenie wiedzy i wykształcenie umiejętności projektowania i eksploatacji nierelacyjnych baz danych.

3.2 Efekty uczenia się dla przedmiotu

EK (efekt uczenia się)	Treść uczenia się zdefiniowanego dla przedmiotu	Odniesienie do efektów kierunkowych ¹
EK_01	Student zna podstawowe metody projektowania relacyjnych baz danych, zna język SQL oraz wybrany język programowania proceduralnego baz danych, a także zna co najmniej jedną technologię dostępu do relacyjnych baz danych z poziomu języka programowania. Potrafi tłumaczyć język zapytań biznesowych na zadania eksploracji danych stosując podstawowe techniki, metody oraz narzędzia wykorzystywane w procesie rozwiązywania zadań informatycznych.	K_Wo4
EK_02	Ma pogłębioną wiedzę o alternatywnych architekturach i modelach baz danych i konfrontuje te modele wobec modelu relacyjnego. Dobiera model danych z uwzględnieniem postulatów ACID vs BASE.	K_Wo7
EK_03	Potrafi projektować relacyjne bazy danych, programować w języku SQL oraz tworzyć aplikacje w wybranym języku programowania, mające dostęp do bazy danych. Potrafi zrealizować projekt prostego przetwarzania danych wewnątrz bazy danych z wykorzystaniem odpowiednich procedur i narzędzi, wraz z implementacją w wybranym środowisku, w szczególności formułując algorytmy i je implementując z użyciem przynajmniej jednego z popularnych narzędzi, stosując odpowiednie typy danych i wybrane biblioteki oraz potrafi stosować podstawowe metody, techniki oraz dobierać narzędzia odpowiednie do rozwiązywania problemów w wybranych obszarach informatyki.	K_Uo9

3.3 Treści programowe

A. Problematyka wykładu

Część pierwsza (semestr 2)

Wprowadzenie do systemów baz danych. Relacyjny model danych: struktury danych, ograniczenia wynikające z integralności danych, operacje.
Modelowanie pojęciowe: model związków-encji. Transformacja z modelu pojęciowego do relacyjnego. Normalizacja i denormalizacja schematu relacyjnej bazy danych.
Język opisu danych, omówienie składni języka DDL. Tworzenie, modyfikacja i destrukcja poszczególnych obiektów bazy danych. Słownik bazy danych.
Język manipulowania danymi, polecenia i składnia języka DML. Pojęcie transakcji. Zarządzanie transakcjami. Integralność danych, zarządzanie więzami integralności. Indeksy i optymalizacja bazy danych.
Język SQL, projekcja, selekcja, grupowanie, sortowanie, połączenie, suma, iloczyn, różnica, podzapytania, zapytania skorelowane.
Wybrane funkcje języka SQL: numeryczne, znakowe, daty, konwersji, warunkowe.
Zagadnienia bezpieczeństwa informacji w systemach zarządzania bazami danych. Zarządzanie prawami dostępu do danych, poziomy uprawnień.
Część druga (semestr 3)
Proceduralny język bazy danych PL/SQL: definiowanie danych, instrukcje sterujące, obsługa kursorów, obsługa wyjątków.
Funkcje i procedury składowane w bazie danych. Sekwencje.
Aktywne bazy danych: procedury wyzwalane (ang. triggers).
Podstawy teoretyczne projektowania interfejsu użytkownika. Architektura interfejsu użytkownika.
Odwzorowanie struktur danych w graficznym interfejsie użytkownika. Rodzaje elementów graficznych.
Właściwości elementów graficznych i sposoby ich definiowania. Omówienie funkcjonalności aplikacji.
Omówienie dwu i trójwarstwowej architektury aplikacji, z przykładami implementacji z użyciem współczesnych języków programowania (np.: C#, Java, PHP).
Wprowadzenie do systemów nierelacyjnych baz danych przy uwzględnieniu rysu historycznego rozwoju baz danych, przeglądu kluczowych własności relacyjnych baz danych, z wyszczególnieniem tych ograniczeń, które determinują potrzebę opracowania modeli nierelacyjnych. Przegląd najpopularniejszych modeli nierelacyjnych.
Wprowadzenie do modelowania struktur danych opartych o relacje vs dokumenty vs key/value. Omówienie funkcjonalności baz danych w modelu „Klucz-wartość”.
Omówienie kolumnowego modelu danych, zalety i wady, kompresja danych, optymalizacja w analitycznym przetwarzaniu danych.
Wprowadzenie i omówienie dokumentowych baz danych, na przykładzie środowisk MongoDB lub Lotus Notes. Przetwarzanie danych w środowisku Tiny MongoDB Browser Shell.
Omówienie prostych zapytań wybierających, zapytań agregujących oraz metody MapReduce w środowisku MongoDB.

B. Problematyka laboratoriów

Część 1 (semestr 2)
Modelowanie pojęciowe: model związków-encji. Transformacja z modelu pojęciowego do relacyjnego. Normalizacja i denormalizacja schematu relacyjnej bazy danych.
Język opisu danych, omówienie składni języka DDL. Tworzenie, modyfikacja i destrukcja poszczególnych obiektów bazy danych. Język manipulowania danymi, polecenia i składnia języka DML. Pojęcie transakcji. Zarządzanie transakcjami. Integralność danych, zarządzanie więzami integralności. Indeksy i optymalizacja bazy danych.
Język SQL, projekcja, selekcja, grupowanie, sortowanie, połączenie, suma, iloczyn, różnica, podzapytania, zapytania skorelowane.
Wybrane funkcje języka SQL: numeryczne, znakowe, daty, konwersji, warunkowe.
Część 2 (semestr 3)
Proceduralny język bazy danych PL/SQL: definiowanie danych, instrukcje sterujące, obsługa kursorów, obsługa wyjątków.
Funkcje i procedury składowane w bazie danych. Sekwencje.
Aktywne bazy danych: procedury wyzwalane (ang. triggers).
<ul style="list-style-type: none"> • Projekt aplikacji dwu- lub trójwarstwowej (warstwa bazy danych, model związków encji, implementacja w środowisku relacyjnym, procedury składowane realizujące scenariusze biznesowe; warstwa serwera aplikacji i GUI w dowolnym języku programowania) • Projekt skryptu wykonywalnego, usprawniającego pracę administratora baz danych, w zakresie kontroli stanu serwera bazodanowego oraz automatycznej eksploracji danych procesowych zaprojektowanego scenariusza biznesowego (skrypt wpisany w harmonogram shell/crone uruchamia procedury składowane w bazie danych a rezultat biznesowy raportuje do pliku/wysyła mailem).
Opracowanie dokumentacji projektu (opis kontekstu biznesowego, model repozytorium danych, kod wykonywalny, instrukcja obsługi). Dokonanie transformacji całości/części modelu danych procesowych/lub wynikowych do postaci nierelacyjnej, wraz z uzasadnieniem/opisaniem podjętych działań w dokumentacji projektu.

3.4 Metody dydaktyczne

Wykład: wykład problemowy, wykład z prezentacją multimedialną

Laboratorium: praca indywidualna przy komputerze, metoda projektów (projekt praktyczny), praca w grupach (rozwiązywanie zadań, dyskusja).

4. METODY I KRYTERIA OCENY

4.1 Sposoby weryfikacji efektów uczenia się

Symbol efektu	Metody oceny efektów uczenia się	Forma zajęć dydaktycznych
---------------	----------------------------------	---------------------------

	(np.: kolokwium, egzamin ustny, egzamin pisemny, projekt, sprawozdanie, obserwacja w trakcie zajęć)	(w, ćw, ...)
CZĘŚĆ 1 (SEMESTR 2)		
EK_01, EK_03	weryfikacja wiedzy w postaci kolokwium/testu zaliczeniowego, obserwacja w trakcie zajęć	w, lab
CZĘŚĆ 2 (SEMESTR 3)		
EK_01, EK_03	weryfikacja wiedzy w postaci kolokwium/testu zaliczeniowego, obserwacja w trakcie zajęć	w, lab
EK_03	projekt	lab

4.2 Warunki zaliczenia przedmiotu (kryteria oceniania)

W SEMESTRZE DRUGIM

Zaliczenie wykładu:

Następuje na podstawie pozytywnie ocenionej wiedzy z zakresu zagadnień z pierwszej części wykładu. Weryfikacja wiedzy może odbywać się w postaci np. kolokwium pisemnego, kolokwium testowego lub w innej formie - w zakresie osiągnięcia efektu EK_01. Negatywna oceny wiedzy studenta skutkuje brakiem zaliczenia wykładu.

Zaliczenie laboratorium:

Następuje na podstawie zaliczenia dwóch kolokwiów z:

1) Projektowania i normalizacji baz danych, podczas którego studenci rozwiązują zadania dotyczące tworzenia diagramu ERD na podstawie wymagań podanych w opisie wybranego wycinka rzeczywistości, na którym umieszczają odpowiedniki encji, atrybutów i związków pomiędzy encjami, następnie definiują istniejące (lub proponują nowe) reguły integralności. W dalszej części kolokwium na podstawie przedstawionych tabel/baz danych studenci muszą dokonać określenia obecnej postaci bazy danych lub oceny czy spełniona jest konkretna zadana postać oraz dokonać normalizacji.

Za kolokwium przyznawana jest ocena, w zależności proporcjonalnie od uzyskanej liczby punktów za całość. Ocena *dostateczny* wymaga uzyskania przynajmniej 50% punktów.

2) Języka SQL, w którym studenci na podstawie zadanych pytań w mowie potocznej potrafią napisać: zapytania SQL oparte o 1...n tabel źródłowych, a także zapytania zawierające złączenia tabel, grupujące wiersze, zawierające formuły wbudowane w system bazy danych w zakresie przetwarzania tekstu, konwersji formatu daty i wartości numerycznych, zapytania zagnieżdżone np. podzapytania skorelowane.

Oprócz zapytań SQL studenci formułują także polecenia DDL w zakresie implementacji modelu ERD w bazie danych oraz w zakresie rekonfiguracji struktury baz danych, a także tworzące widoki (perspektywy).

Za kolokwium przyznawana jest ocena, w zależności proporcjonalnie od uzyskanej liczby punktów za całość. Ocena *dostateczny* wymaga uzyskania przynajmniej 50% punktów.

Efekt EK_03 jest uznany za zaliczony, gdy student z obu kolokwiów otrzyma ocenę przynajmniej „dostateczny”.

W SEMESTRZE TRZECIM

Zaliczenie wykładu następuje w oparciu o zaliczenie efektów EK_01 – EK_03 w ramach zaliczonego projektu i zaliczonego testu lub innej formy weryfikacji wiedzy i umiejętności.

Zaliczenie laboratorium następuje na podstawie zaliczenia efektów EK_02 i EK_03 poprzez realizację projektu bazy danych wraz z oprogramowaną logiką dla rozwiązywanego zagadnienia biznesowego w warstwie bazy danych. Wymagania na poszczególną ocenę:

dostateczny:

W ramach laboratorium studenci zostają podzieleni na zespoły. Studenci w ramach zespołów przy wykorzystaniu mechanizmów pracy zespołowej mają zaprojektować bazę danych w wybranym temacie, następnie zaimplementować ją w wybranym środowisku bazodanowym wspierającym proceduralny język bazy danych np.: PL/SQL. Baza danych powinna opisywać wybrany temat w postaci tabel połączonych między sobą różnymi typami powiązań. W zdefiniowanej bazie danych przy wykorzystaniu języka proceduralnego mają zostać zaimplementowane podstawowe operacje procesowe danego zagadnienia biznesowego w procedurach zeskładowanych. Wymagane jest uruchomienie procedur z linii komend SQL w trybie interaktywnym.

dobry:

Student spełnia kryterium oceny dostateczny, a ponadto powinien w ramach realizowanego projektu zaimplementować interakcję z wbudowanymi procedurami (funkcjami) z poziomu zewnętrznego środowiska aplikacyjnego (gui) lub harmonogramu zadań środowiska systemu operacyjnego (shell).

bardzo dobry:

Student spełnia kryterium oceny dobry, a ponadto powinien w ramach realizowanego projektu opracować alternatywny model nierelacyjny bazy danych wraz z zaprezentowaniem interakcji z tym modelem w odpowiednim środowisku.

Efekt EK_03 jest uznany za zaliczony, gdy projekt wykonany przez studenta spełni wszystkie wymagania określone przynajmniej na ocenę „dostateczny”.

Efekt EK_02 jest uznawany za zaliczony, wyłącznie w przypadku, gdy dany student uczestniczy w realizacji projektu, w którym opracowano praktyczne rozwiązanie modelu nierelacyjnego oraz zagadnienie to opisano w dokumentacji projektu.

Zaliczenie następuje poprzez uzyskanie oceny pozytywnej (50% i wyżej) z testu lub innej formy weryfikacji wiedzy uzyskanej na wykładzie i projektu, który weryfikuje wiedzę z zakresu zagadnień z pierwszej i drugiej części przedmiotu.

5. CAŁKOWITY NAKŁAD PRACY STUDENTA POTRZEBNY DO OSIĄGNIĘCIA ZAŁOŻONYCH EFEKTÓW W GODZINACH ORAZ PUNKTACH ECTS

Forma aktywności	Średnia liczba godzin na zrealizowanie aktywności
Godziny kontaktowe wynikające z harmonogramu studiów	110
Inne z udziałem nauczyciela akademickiego (udział w konsultacjach, egzaminie)	5
Godziny niekontaktowe – praca własna studenta (przygotowanie do zajęć, egzaminu, napisanie referatu itp.)	110
SUMA GODZIN	225
SUMARYCZNA LICZBA PUNKTÓW ECTS	9

* Należy uwzględnić, że 1 pkt ECTS odpowiada 25-30 godzin całkowitego nakładu pracy studenta.

6. PRAKTYKI ZAWODOWE W RAMACH PRZEDMIOTU

wymiar godzinowy	–
zasady i formy odbywania praktyk	–

7. LITERATURA

LITERATURA PODSTAWOWA:

1. Jeffrey D. Ullman: Podstawowy wykład z systemów baz danych, Warszawa; Wydawnictwa Naukowo-Techniczne, 2001.
2. Krystyna Czapla: Bazy danych: podstawy projektowania i języka SQL, Gliwice; Helion, 2015.
3. Barbara Pękala: Bazy danych: teoria i praktyka, Wydawnictwo Uniwersytetu Rzeszowskiego, 2015.
4. Jeff Dickey: Nowoczesne aplikacje internetowe: MongoDB, Express, AngularJS, Node.js; poznaj nowe podejście do aplikacji internetowych [tł. Robert Górczyński], Gliwice, Helion, 2016.
5. Hernandez, Michael J.: Projektowanie baz danych dla każdego: przewodnik krok po kroku, Helion, 2022.

LITERATURA UZUPEŁNIAJĄCA:

1. Andrzej Barczak, Jacek Florek, Tadeusz Sydoruk: Bazy danych. Akademia Podlaska. Wydział Nauk Ścisłych. Instytut Informatyki. Siedlce; Wydaw. AP, 2006.
2. Adam Pelikant: Bazy danych: pierwsze starcie. Gliwice; Wydaw. Helion, 2009.
3. Nathan Marz, James Warren: Big data: najlepsze praktyki budowy skalowalnych systemów obsługi danych w czasie rzeczywistym [tł.: Lech Lachowski], Gliwice, Helion, 2016.

4. Anthony Molinaro, Robert De Graaf : SQL : zapytania i techniki dla bazodanowców:
receptury, Helion, 2021.