SELECT 2.01

Nazwy wszystkich wykładów, liczby godzin oraz identyfikatory, nazwiska i imiona prowadzących. Dane posortowane rosnąco według nazw wykładów a następnie nazwisk i imion prowadzących.

**16 rekordów, pierwszy rekord: Ekonomia, prowadzący Jacek Malinowski**

SELECT nazwa\_wykladu, liczba\_godzin, id\_wykladowcy, nazwisko, imie

FROM wyklady INNER JOIN pracownicy ON id\_wykladowcy=id\_pracownika

ORDER BY nazwa\_wykladu, nazwisko, imie

SELECT 2.02

Nazwiska i imiona studentów zapisanych na wykład ze Statystyki, posortowane rosnąco według nazwiska i imienia.

**Dwóch studentów: Maciej Iwanicki i Aneta Janowska.**

SELECT nazwisko, imie

FROM (studenci s INNER JOIN studenci\_wyklady sw ON s.id\_studenta=sw.id\_studenta)

INNER JOIN wyklady w ON w.id\_wykladu=sw.id\_wykladu

WHERE nazwa\_wykladu='Statystyka'

ORDER BY nazwisko, imie

SELECT 2.03

Nazwiska, imiona i identyfikatory studentów, którzy przystąpili do egzaminu co najmniej raz oraz daty egzaminów. Jeśli student danego dnia przystąpił do wielu egzaminów, jego dane mają się pojawić tylko raz. Dane posortowane rosnąco względem dat.

**9 rekordów.   
Ostatnie trzy: student Wojciech Jarkowski z datami 23-09-2017, 11-10-2017   
oraz 13-10-2017**SELECT DISTINCT nazwisko, imie, s.id\_studenta, data\_egzaminu

FROM studenci s INNER JOIN oceny\_studentow o ON s.id\_studenta=o.id\_studenta

ORDER BY data\_egzaminu

SELECT 2.04

Nazwiska, imiona i stopnie/tytuły naukowe pracowników Katedry Informatyki. Dane posortowane rosnąco według nazwisk i imion.

**2 rekordy. Pierwszy: Agnieszka Kowalczyk, doktor**

SELECT nazwisko, imie, stopien\_tytul

FROM pracownicy INNER JOIN wykladowcy

ON pracownicy.id\_pracownika=wykladowcy.id\_wykladowcy

WHERE katedra='Katedra Informatyki'

ORDER BY nazwisko, imie

SELECT 2.05

Nazwiska i imiona wszystkich pracowników, a dla tych, którzy są pracownikami naukowymi także nazwy katedr. Dane posortowane rosnąco według nazwisk oraz malejąco według imion.

**10 rekordów. 4 rekordy bez wpisanej katedry.   
Ostatni rekord: Iwona Wilga bez wpisanej katedry**

SELECT nazwisko, imie, katedra

FROM pracownicy LEFT JOIN wykladowcy

ON pracownicy.id\_pracownika=wykladowcy.id\_wykladowcy

ORDER BY nazwisko, imie DESC

SELECT 2.06

Identyfikatory, nazwiska, imiona i stopnie/tytuły naukowe wykładowców, którzy nie prowadzą żadnego wykładu. Dane posortowane według stopni naukowych.

**2 rekordy. Wykładowcy o identyfikatorach 9 i 8 (w takiej kolejności)**SELECT wykladowcy.id\_wykladowcy, nazwisko, imie, stopien\_tytul

FROM (pracownicy INNER JOIN wykladowcy ON

pracownicy.id\_pracownika=wykladowcy.id\_wykladowcy)

LEFT JOIN wyklady w ON w.id\_wykladowcy=wykladowcy.id\_wykladowcy

WHERE id\_wykladu IS NULL

ORDER BY stopien\_tytul

SELECT 2.07

Imiona i nazwiska wszystkich studentów, nazwy wykładów, na które są zapisani, nazwiska prowadzących te wykłady (pole ma mieć nazwę NazwiskoWykladowcy) oraz nazwy katedr w których każdy z wykładowców pracuje. Dane posortowane malejąco według nazw wykładów a następnie rosnąco według nazwisk wykładowców.

**18 rekordów,   
np. na wykład ze Statystyki zapisanych jest dwóch studentów a z Matematyki jeden**

SELECT s.nazwisko, s.imie, nazwa\_wykladu,

p.nazwisko AS NazwiskoWykladowcy, katedra

FROM studenci s LEFT JOIN

(((studenci\_wyklady sw INNER JOIN wyklady w ON sw.id\_wykladu=w.id\_wykladu)

INNER JOIN wykladowcy wy ON wy.id\_wykladowcy=w.id\_wykladowcy)

INNER JOIN pracownicy p ON wy.id\_wykladowcy=p.id\_pracownika)

ON s.id\_studenta=sw.id\_studenta

ORDER BY nazwa\_wykladu DESC, NazwiskoWykladowcy

SELECT 2.08

Zapytanie zwracające jedną liczbę: kwotę, jaką uczelnia musi przygotować na wypłaty z tytułu prowadzonych wykładów. Kwotę za jeden wykład należy wyliczyć jako iloczyn stawki godzinowej danego pracownika oraz liczby godzin przeznaczonych na prowadzone przez niego wykłady.

**Wynikiem jest kwota 22 200**

SELECT SUM(stawka\*liczba\_godzin)

FROM (stopnie\_tytuly st INNER JOIN wykladowcy w ON st.stopien\_tytul=w.stopien\_tytul)

INNER JOIN wyklady wk ON wk.id\_wykladowcy=w.id\_wykladowcy

SELECT 2.09

Identyfikatory i nazwy wykładów na które nie został zapisany żaden student. Dane posortowane malejąco według nazw wykładów.

**7 rekordów,   
trzy razy pojawia się wykład z Informatyki, raz z Informatyki dla zaawansowanych**

SELECT id\_wykladu, nazwa\_wykladu

FROM wyklady

WHERE id\_wykladu NOT IN

(SELECT id\_wykladu FROM studenci\_wyklady)

ORDER BY nazwa\_wykladu DESC

SELECT 2.10

Nazwy wykładów, których nazwa zaczyna się od słowa Historia oraz liczbę godzin przewidzianych na te wykłady, nazwiska prowadzących i nazwy katedr, w których pracują. Dane posortowane malejąco według nazwisk.

**Trzy rekordy. Wszystkie wykłady prowadzi Jaskowski**

SELECT nazwa\_wykladu, liczba\_godzin, nazwisko, katedra

FROM (pracownicy p INNER JOIN wykladowcy w ON

p.id\_pracownika=w.id\_wykladowcy)

INNER JOIN wyklady wk ON wk.id\_wykladowcy=w.id\_wykladowcy

WHERE nazwa\_wykladu LIKE 'Historia%'

ORDER BY nazwisko DESC

SELECT 2.11

Nazwiska, imiona i PESELe wykładowców, którzy prowadzą wykłady wraz z nazwami tych wykładów   
oraz   
nazwiska, imiona i numery grup wszystkich studentów wraz z nazwami wykładów na które się zapisali. Trzecia kolumna ma mieć nazwę pesel\_grupa. Dane posortowane rosnąco według kolumny pesel\_grupa a następnie według nazwisk i imion.

**30 rekordów. W pierwszych ośmiu w polu pesel\_grupa jest NULL**

SELECT nazwisko, imie, pesel AS pesel\_grupa, nazwa\_wykladu

FROM pracownicy INNER JOIN wyklady ON id\_pracownika=id\_wykladowcy

UNION

SELECT nazwisko, imie, nr\_grupy, nazwa\_wykladu

FROM studenci s LEFT JOIN (wyklady w INNER JOIN studenci\_wyklady sw ON w.id\_wykladu=sw.id\_wykladu) ON s.id\_studenta=sw.id\_studenta

ORDER BY pesel\_grupa, nazwisko, imie

SELECT 2.12

Identyfikatory, nazwiska i imiona pracowników, którzy nie są wykładowcami. Dane posortowane malejąco według nazwisk i rosnąco według imion.

**4 rekordy. Pierwszy Iwona Wilga, id=10**

SELECT id\_pracownika, nazwisko, imie

FROM pracownicy WHERE id\_pracownika NOT IN

(SELECT id\_wykladowcy FROM wykladowcy)

ORDER BY nazwisko DESC, imie

SELECT 2.13

Identyfikatory studentów, którzy przystąpili do egzaminu zarówno 2017-04-12 jak i 2017‑10‑11. Dane posortowane malejąco według identyfikatorów. Użyj składni podzapytania.

**Student o identyfikatorze 6**

SELECT id\_studenta

FROM oceny\_studentow os

WHERE os.data\_egzaminu='20170412' AND id\_studenta IN

(SELECT id\_studenta

FROM oceny\_studentow

WHERE oceny\_studentow.data\_egzaminu='20171011')

ORDER BY id\_studenta DESC

SELECT 2.14

Identyfikatory i nazwiska studentów, którzy nie otrzymali dotychczas oceny z wykładu, na który się zapisali wraz z nazwą tego wykładu (uwaga: każdy student ma się pojawić tyle razy z ilu wykładów nie otrzymał oceny). Dane posortowane rosnąco według identyfikatorów i nazwisk studentów.

**8 rekordów**

SELECT s.id\_studenta,nazwisko,nazwa\_wykladu

FROM ((studenci s INNER JOIN studenci\_wyklady sw ON s.id\_studenta=sw.id\_studenta)   
LEFT JOIN oceny\_studentow os ON sw.id\_studenta=os.id\_studenta AND sw.id\_wykladu=os.id\_wykladu)

INNER JOIN wyklady w ON w.id\_wykladu=sw.id\_wykladu

WHERE os.id\_studenta IS NULL

ORDER BY id\_studenta,nazwisko

SELECT 2.15

Imiona, nazwiska oraz identyfikatory studentów, którzy zapisali się zarówno na wykład o identyfikatorze 4 jak i 11. Użyj składni podzapytania.

**Jest tylko jeden taki student: Wojciech Jarkowski o identyfikatorze 6**

SELECT imie, nazwisko, id\_studenta

FROM studenci  
WHERE id\_studenta IN

(SELECT id\_studenta

FROM studenci\_wyklady

WHERE id\_wykladu=11 AND id\_studenta IN

(SELECT id\_studenta  
FROM studenci\_wyklady  
WHERE id\_wykladu=4))

SELECT 2.16

Identyfikatory i nazwy wykładów, na które został zapisany co najmniej jeden student i z których nie została dotychczas przyznana żadna ocena.  
**Są cztery takie wykłady o identyfikatorach 5, 6, 11 i 12**

SELECT DISTINCT w.id\_wykladu, nazwa\_wykladu

FROM wyklady w INNER JOIN studenci\_wyklady s ON w.id\_wykladu=s.id\_wykladu

LEFT JOIN oceny\_studentow o ON s.id\_wykladu=o.id\_wykladu

WHERE o.id\_studenta IS NULL

lub

SELECT DISTINCT w.id\_wykladu, nazwa\_wykladu

FROM wyklady w INNER JOIN studenci\_wyklady s ON w.id\_wykladu=s.id\_wykladu

WHERE w.id\_wykladu NOT IN

(SELECT id\_wykladu FROM oceny\_studentow)

lub

SELECT id\_wykladu, nazwa\_wykladu

FROM wyklady

WHERE id\_wykladu IN

(SELECT id\_wykladu

FROM studenci\_wyklady

WHERE id\_wykladu NOT IN

(SELECT id\_wykladu

FROM oceny\_studentow))