## SZKOŁA WYŻSZA im. PAWŁA WŁODKOWICA

**w PŁOCKU**

# WYDZIAŁ INFORMATYKI

Przygotowali:

Kamil Zima

Mariusz Zając

Piotr Nowakowski

**System Geografia**

Bazy Danych

(dr inż. Maciej Dorobek)  
Semestr IV

Płock 2017

|  |  |
| --- | --- |
| **Imię** | Kamil |
| **Nazwisko** | Zima |
| **Numer albumu** |  |
| **Imię** | Mariusz |
| **Nazwisko** | Zając |
| **Numer Albumu** |  |
| **Imię** | Piotr |
| **Nazwisko** | Nowakowski |
| **Numer Albumu** | 52923 |
| **Rok akademicki** | 2016/2017 |
| **Semestr** | 4 |
| **Temat projektu** | System Geografia |

Wstęp

System geografia jest systemem analitycznym

Obiekty:

* kontynenty
* kontynenty\_panstwa
* miara
* miasta
* panstwa
* panstwo\_panstwo
* pasmo\_gorskie
* pasmo\_gorskie\_panstwa
* pustynie
* pustynie\_kont\_panst
* szczyty
* zbiornik\_typ
* zbiorniki
* zbiorniki\_miary
* zbiorniki\_panstwa

­­­

**Przykładowe zapytania zwracające:**

Zapytania zostały wygenerowane przy użyciu MySQL Workbench 6.3 CE

1. Wszystkie państwa na danym kontynencie

SELECT p.nazwa, j.nazwa as kontynent

FROM panstwa AS p LEFT JOIN

(

SELECT k.id AS kontynentId, k.nazwa, kp.id\_kontynent, kp.id\_panstwo

FROM kontynenty as k LEFT JOIN kontynenty\_panstwa AS kp ON k.id = kp.id\_kontynent

) AS j

ON j.id\_panstwo = p.id

1. Wymień wszystkich sąsiadów danego kraju

SELECT p.nazwa,

CASE

WHEN p.id = q.id\_panstwo1 THEN (SELECT p1.nazwa FROM panstwa AS p1 WHERE p1.id = q.id\_panstwo2)

ELSE (SELECT p1.nazwa FROM panstwa AS p1 WHERE p1.id = q.id\_panstwo1) END

AS graniczyZ

FROM panstwa as p LEFT JOIN (SELECT pp.id\_panstwo1, pp.id\_panstwo2

FROM panstwo\_panstwo AS pp) AS q ON p.id = q.id\_panstwo1 OR p.id = q.id\_panstwo2

WHERE p.id = 1

1. Wymień wszystkie kraje które mają minimum 2 sąsiadów

SELECT p.nazwa, COUNT(

CASE

WHEN p.id = q.id\_panstwo1 THEN (SELECT 1 FROM panstwa AS p1 WHERE p1.id = q.id\_panstwo2)

ELSE (SELECT 1 FROM panstwa AS p1 WHERE p1.id = q.id\_panstwo1) END)

AS liczbaSasiadow

FROM panstwa as p LEFT JOIN (SELECT pp.id\_panstwo1, pp.id\_panstwo2

FROM panstwo\_panstwo AS pp) AS q ON p.id = q.id\_panstwo1 OR p.id = q.id\_panstwo2

GROUP BY p.nazwa

HAVING COUNT(

CASE

WHEN p.id = q.id\_panstwo1 THEN (SELECT 1 FROM panstwa AS p1 WHERE p1.id = q.id\_panstwo2)

ELSE (SELECT 1 FROM panstwa AS p1 WHERE p1.id = q.id\_panstwo1) END) >=2

1. Nazwa i liczba mieszkańców miast powyżej 1 mln mieszkańców

SELECT nazwa, ludnosc FROM miasta

WHERE ludnosc > 1000000

1. Liczba ludności w miastach europy które sa stolicami

SELECT sum(ludnosc) as liczbaLudnosci

FROM miasta

WHERE id\_panstwo IN (SELECT id\_panstwo FROM kontynenty\_panstwa WHERE id\_kontynent = 1)

AND czystolica = 1

1. Kraje przez które przepływa najdłuższa rzeka

SELECT zp.\*, (

SELECT p.nazwa FROM panstwa AS p WHERE p.id = zp.id\_panstwo

) AS nazwaPanstwa

FROM zbiorniki\_panstwa AS zp LEFT JOIN

(

SELECT z.id, z.nazwa FROM zbiorniki AS z INNER JOIN zbiorniki\_miary AS m ON z.id = m.id\_zbiornik

WHERE z.typ = 'rzeka' and m.id\_miara = 1

) AS q

ON zp.id\_zbiornik = q.id

1. Największy zbiornik na danym kontynencie

SELECT k.nazwa AS kontynent, q.nazwa AS zbiornik, q.wartosc AS powierzchnia

FROM kontynenty AS k LEFT JOIN

(

SELECT z.id, z.nazwa, m.wartosc, zp.id\_panstwo, kp.id\_kontynent

FROM zbiorniki AS z left JOIN zbiorniki\_miary AS m

ON z.id = m.id\_zbiornik

LEFT JOIN zbiorniki\_panstwa AS zp ON z.id = zp.id\_panstwo

LEFT JOIN kontynenty\_panstwa AS kp ON zp.id\_panstwo = kp.id\_panstwo

WHERE m.id\_miara = 3

ORDER BY m.wartosc DESC

LIMIT 1

) AS q

ON k.id = q.id\_kontynent

1. Średnia głębokość jezior danego kontynentu
2. Ilość jezior w krajach przez które przepływa najdłuższa rzeka
3. Kraje w porządku malejącym wg najwyższego szczytu