

ZADANIA Z LISTY

1. Zbuduj tabelę:

(a) powiatów w województwie dolnośląskim;

```
select p.powiat from powiaty as p
join wojewodztwa as w on p.klwoj=w.klwoj
where wojewodztwo = 'Dolnośląskie'
```

albo

```
select from powiaty
where klwoj in(select klwoj from wojewodztwa where wojewodztwo =
'Dolnośląskie')
```

(b) łączną listą powiatów w województwach dolnośląskim i wielkopolskim;

```
select count(p.powiat) from powiaty as p
join wojewodztwa as w on p.klwoj=w.klwoj
where wojewodztwo = 'Dolnośląskie' or wojewodztwo = 'Wielkopolskie'
```

(c) gmin w województwie dolnośląskim;

```
select gmina from gminy as g
join powiaty as p on g.klpow=p.klpow
join wojewodztwa as w on p.klwoj=w.klwoj
where wojewodztwo = 'Dolnośląskie'
```

albo

```
select gmina from gminy where klpow in
(select klpow from powiaty where klwoj in
(select klwoj from wojewodztwa
where wojewodztwo = 'Dolnośląskie'))
```

(d) miejscowości na rozpoczynających się od zadanej litery z województwa dolnośląskiego;

```
select miejscowosc from miejscowosci where klgm in
(select klgm from gminy where klpow in
(select klpow from powiaty where klwoj in
(select klwoj from wojewodztwa
where wojewodztwo = 'Dolnośląskie' and miejscowosc like 'A%'))))
```

(e) leśnictw i leśniczówek w danym województwie.

```
select m.miejscowosc, t.typmsc, w.wojewodztwo from typymsc as t
join miejscowosci as m on t.kltm=m.kltm
join gminy as g on m.klgm=g.klgm
join powiaty as p on g.klpow=p.klpow
join wojewodztwa as w on p.klwoj=w.klwoj
where wojewodztwo = 'Dolnośląskie'
and typmsc like 'leśniczów.'
```

2. Przeglądając tabele Osoby i Studenci przedstaw listy:

(a) wszystkich dziewcząt;

```
select imie, nick
from osoby where plec = 'K'
```

(b) wszystkich osób o wzroście powyżej 1.65 m;

```
select imie, nick
from osoby where wzrost > 1.65
```

(c) wszystkich studentów z województwa dolnośląskiego;

```
select o.imie, o.nick from osoby as o
join miejscowosci as m on m.klmisc=o.klmisc
join gminy as g on g.klgm=m.klgm
join powiaty as p on p.klpow=g.klpow
join wojewodztwa as w on w.klwoj=p.klwoj
where wojewodztwo = 'Dolnośląskie'
```

(d) wszystkich studentów znających angielski;

```
select o.imie, o.nick, j.jezyk from osoby as o
join znajezyki as z on z.klos=o.klos
join jezyki as j on j.klj=z.klj
where jezyk = 'angielski'
```

(e) wszystkich studentów znających angielski i niemiecki;

```
select imie, nick from osoby
where klos in(select klos from znajezyki where klj in (select klj
from jezyki where jezyk='angielski'))
and klos in(select klos from znajezyki where klj in (select klj from
jezyki where jezyk='niemiecki'))
```

(f) wszystkich studentów znających francuski lub niemiecki;

```
select imie, nick from osoby
where klos in(select klos from znajezyki where klj in (select klj
from jezyki where jezyk='francuski'))
or klos in(select klos from znajezyki where klj in (select klj from
jezyki where jezyk='niemiecki'))
```

(g) wszystkich studentów znających biegle jakiś język;

```
select imie, nick from osoby
where klos in(select klos from znajezyki
where klpz in (select klpz from poziomyzn where poziom='biegły'))
```

albo

```
select distinct imie, nick from osoby as o
join znajezyki as z on z.klos=o.klos
join poziomyzn as pz on pz.klpz = z.klpz
where poziom = 'biegły'
```

(h) dziewczyn mających braci (zakładamy, że imię żeńskie kończy się na literę "a", zaś imię męskie nie kończy się na "a");

```
select imie,nick, plec from osoby
where klos in(select klos from rodzenstwo where imie not like '%a')
and plec = 'K'
```

(i) studentów nie mających rodzeństwa, a znających niemiecki;

```
select imie,nick from osoby
where klos not in(select klos from rodzenstwo)
and klos in(select klos from znajezyki where klj in(select klj from
jezyki where jezyk = 'niemiecki'))
```

(j) język z listy języków nie znany żadnemu ze studentów;

```
select jezyk from jezyki
where klj not in(select klj from znajezyki)
```

(k) osoby urodzone w listopadzie;

```
select * from osoby where extract (month from dataur)=11
```

(l) osoby urodzone w niedzielę;

```
select * from osoby where extract (dow from dataur)=1
```

(m) osoby urodzone 13-tego dnia miesiąca.

```
select * from osoby where extract (day from dataur)=13
```

3. Korzystając z funkcji zliczających podaj:

(a) liczbę powiatów w danym województwie;

```
select wojewodztwo, count(p.klpow) from powiaty as p
join wojewodztwa as w on p.klwoj=w.klwoj
group by wojewodztwo
```

(b) województwo w którym jest najwięcej miejscowości o nazwie "Nowy Dwór";

```
select wojewodztwo, count(m.klmsc) from miejscowosci as m
join gminy as g on m.klgm=g.klgm
join powiaty as p on p.klpow=g.klpow
join wojewodztwa as w on w.klwoj=p.klwoj
where miejscowosc = 'Nowy Dwór'
group by wojewodztwo
order by count(m.klmsc) desc
```

(c) liczbę miejscowości o nazwie "Nowa Wieś" oraz "Nowa Wieś Wielka";

```
select miejscowosc, count(klmsc) as ilosc from miejscowosci
where miejscowosc = 'Nowa Wieś' or miejscowosc = 'Nowa Wieś Wielka'
group by miejscowosc
```

(d) województwo o największej liczbie miast;

```
select wojewodztwo, count(m.klmsc) from miejscowosci as m
join gminy as g on g.klgm=m.klgm
join powiaty as p on p.klpow=g.klpow
join wojewodztwa as w on p.klwoj=w.klwoj
join typymisc as t on t.kltm=m.kltm
where t.typmisc='miasto'
group by wojewodztwo
order by count desc limit 1
```

(e) imię i nick najwyższego studenta.

```
select imie, nick from osoby
order by wzrost desc limit 1
```

4. Poprzez łączenie zbuduj tabele o polach (załóż porządek wydruku):

(a) kierunek studiów, rok studiów, grupa;

```
select k.kierunek, gr.rok, gr.grupa from grupy as gr
join kierunki as k on k.klkier=gr.klkier
order by kierunek
```

(b) nazwa województwa, nazwa powiatu;

```
select wojewodztwo, powiat from wojewodztwa as w
join powiaty as p on p.klwoj=w.klwoj
order by wojewodztwo, powiat
```

(c) nazwa powiatu, miasta w tym powiecie;

```
select powiat, miejscowosc from powiaty as p
join gminy as g on g.klpow=p.klpow
join miejscowosci as m on g.klgm = m.klgm
order by powiat,miejscowość
```

(d) dla wybranego województwa - gminy i miejscowości w gminie;

```
select gmina, miejscowosc from gminy as g
join miejscowosci as m on g.klgm=m.klgm
where g.klgm in (select klgm from gminy where klpow in
(select klpow from powiaty where klwoj in
(select klwoj from wojewodztwa where wojewodztwo='Małopolskie'))))
order by gmina
```

(e) imię i nick studenta, imię rodzeństwa;

```
select o.imie, o.nick, r.imie from osoby as o
join rodzenstwo as r on r.klos=o.klos
```

(f) imię i nick studenta, imię brata;

```
select o.imie, o.nick, r.imie from osoby as o
join rodzenstwo as r on r.klos=o.klos
where r.imie not like '%a'
```

(g) imię, nick studenta, nazwa powiatu zamieszkania;

```
select o.imie, o.nick, p.powiat from osoby as o
join miejscowosci as m on o.klmsc=m.klmsc
join gminy as g on g.klgm=m.klgm
join powiaty as p on g.klpow=p.klpow
order by o.imie, o.nick
```

(h) imię, nick studenta, nazwa województwa zamieszkania;

```
select o.imie, o.nick, w.wojewodztwo from osoby as o
join miejscowosci as m on o.klmsc=m.klmsc
join gminy as g on g.klgm=m.klgm
join powiaty as p on g.klpow=p.klpow
join wojewodztwa as w on w.klwoj=p.klwoj
order by o.imie, o.nick
```

(i) imię, nick studenta, znany mu język;

```
select o.imie, o.nick, j.jezyk from osoby as o
join znajezyki as zn on o.klos=zn.klos
join jezyki as j on zn.klj=j.klj
order by imie,nick,jezyk
```

(j) imię, nick studenta, znany mu język przynajmniej na poziomie dobrym.

```
select o.imie, o.nick, j.jezyk, poziom from osoby as o
join znajezyki as zn on o.klos=zn.klos
join jezyki as j on zn.klj=j.klj
join poziomozn as pz on pz.klpz=zn.klpz
where pz.ranga>=3
order by imie,nick,jezyk
```

5. Korzystając z funkcji zliczających podaj (dla studentów Bioinformatyki zapisanych w bazie):

(a) liczbę studentów (ogółem);

```
select count(*) as "liczba studentow" from studenci
```

(b) liczbę osób (pań) o wzroście powyżej średniego (średniego pań);

```
select count(klos) from osoby
where wzrost > (select avg(wzrost) from osoby where plec = 'K')
and plec = 'K'
```

(c) średni poziom znajomości języka angielskiego;

```
select round(avg(ranga),2) as "średni poziom" from znajezyki as zn
join poziomozn as pz on zn.klpz=pz.klpz
join jezyki as j on zn.klj=j.klj
where jezyk='angielski'
```

(d) liczbę osób znających dany język;

```
select j.jezyk, count(zn.klos) from znajezyki as zn
join jezyki as j on j.klj = zn.klj
group by j.jezyk
order by count(zn.klos) desc
```

(e) liczbę powiatów (gmin) w danym województwie zamieszkałych przez studentów Bioinfo;

```
select wojewodztwo, count(p.klpow) from osoby as o
join miejscowosci as m on m.klmsc=o.klmsc
join gminy as g on g.klgm=m.klgm
join powiaty as p on p.klpow=g.klpow
join wojewodztwa as w on w.klwoj=p.klwoj
group by wojewodztwo
order by count desc
```

(f) średni poziom zaawansowania języków ogółem;

```
select poziom from poziomozn
where ranga = (select round(avg(ranga),0) as "średni poziom" from
znajezyki as zn
join poziomozn as pz on zn.klpz=pz.klpz)
```

(g) liczbę języków, które studenci Bioinfo zadeklarowali jako znane;

```
select count(jezyk) from jezyki
where klj in(select klj from znajezyki where klos in(select klos
from osoby))
```

(h) liczbę panów;

```
select count(klos) from osoby
where plec = 'M'
```

(i) średni wzrost panów;

```
select round(avg(wzrost),2) from osoby
where plec = 'M'
```

(j) imię i nick panów (pań) których wzrost jest bliski odpowiedniej średniej;

```
select imie, nick from osoby
where abs(wzrost - (select avg(wzrost) from osoby where plec = 'M'))
< 0.02
and plec = 'M'
```

(k) imię i nick najwyższych i najniższych studentów (pań);

```
select imie, nick from osoby
where wzrost = (select max(wzrost) from osoby where plec='K')
and plec = 'K'
```

```
select imie, nick from osoby
where wzrost = (select min(wzrost) from osoby where plec='K')
and plec = 'K'
```

(l) imię i nick pań wyższych niż średni wzrost panów.

```
select imie, nick from osoby
where wzrost > (select avg(wzrost) from osoby where plec='M')
and plec = 'K'
```

6. Grupując, dla studentów Bioinformatyki zapisanych w bazie, podaj:

(a) język i liczbę osób go znających;

```
select jezyk, count(klos) as lj from jezyki as j
join znajezyki as zn on j.klj=zn.klj
group by jezyk
order by lj desc
```

(b) województwa i liczbę studentów w nich mieszkających;

```
select wojewodztwo, count(klos) from osoby as o
join miejscowosci as m on m.klmsc=o.klmsc
join gminy as g on g.klgm=m.klgm
join powiaty as p on p.klpow=g.klpow
join wojewodztwa as w on w.klwoj=p.klwoj
group by wojewodztwo
order by count desc
```

(c) powiaty i liczbę studentów w nich mieszkających;

```
select powiat, count(klos) from osoby as o
join miejscowosci as m on m.klmsc=o.klmsc
join gminy as g on g.klgm=m.klgm
join powiaty as p on p.klpow=g.klpow
group by powiat
order by count desc
```

(d) język i średnią rangę jego znajomości;

```
select jezyk, round(avg(ranga),2) from znajezyki as zn
join poziomyzn as pz on zn.klpz=pz.klpz
join jezyki as j on zn.klj=j.klj
group by jezyk
order by round desc
```

(e) kierunek i liczbę studentów na nim - dotyczy wszystkich studentów;

```
select kierunek, count(gr.nralbumu) from grupyst as gr
join grupy as g on gr.klgr=g.klgr
join kierunki as k on g.klkier=k.klkier
group by kierunek
```

(f) kierunek studiów, rok studiów, numer grupy - dotyczy wszystkich studentów;

```
select kierunek, rok, grupa from grupy as g join kierunki as k on
k.klkier=g.klkier
order by kierunek, rok, grupa
```

(g) język i liczbę osób go znających w randze wyższej niż 2;

```
select j.jezyk, count(o.klos) from osoby as o
join znajezyki as zn on zn.klos=o.klos
join jezyki as j on j.klj=zn.klj
join poziomyzn as pz on zn.klpz=pz.klpz
where pz.ranga > 2
group by j.jezyk
order by count desc
```

(h) liczbę panów i liczbę pań (w jednej tabeli);

```
select plec, count(klos) as ilość_osób from osoby
group by plec
```

(i) liczbę osób znających 1 język.

```
create view osobyjezyki1 as
select o.imie, o.klos, count(j.jezyk) from osoby as o
join znajezyki as zn on zn.klos=o.klos
join jezyki as j on j.klj=zn.klj
group by o.imie, o.klos
having count(j.jezyk)=1
order by o.imie, o.klos
```

```
select count(count) from osobyjezyki1
drop view osobyjezyki1
```

(j) grupy i liczbę studentów (studentek) w grupie;

```
select g.grupa, count(gr.nralbumu) from grupyst as gr
join grupy as g on g.klgr=gr.klgr
group by g.grupa
order by g.grupa
```

(k) w jednej tabeli: numer grupy, liczbę studentów oraz liczbę studentek w grupie.

```
select g.grupa, o.plec, count(o.klos) from osoby as o
join studenci as s on s.klos=o.klos
join grupyst as gr on gr.nralbumu = s.nralbumu
join grupy as g on g.klgr=gr.klgr
group by o.plec, g.grupa
order by g.grupa
```

7. Grupując, zbuduj tabelę:

(a) liczbę gmin w każdym z powiatów Twojego województwa;

```
select p.powiat, count(g.klgm) as liczba_gmin from gminy as g
join powiaty as p on g.klpow=p.klpow
join wojewodztwa as w on w.klwoj=p.klwoj
where wojewodztwo = 'Dolnośląskie'
group by p.powiat
order by p.powiat
```


(b) liczbę miast w każdym z województw;

```
select w.województwo, count(m.klmsc) from miejscowosci as m
join gminy as g on g.klgm=m.klgm
join powiaty as p on p.klpow=g.klpow
join wojewodztwa as w on w.klwoj=p.klwoj
group by w.województwo
order by w.województwo
```

(c) liczbą miast na prawach powiatu w każdym z województw;

```
select w.województwo, count(p.powiat) from powiaty as p
join wojewodztwa as w on w.klwoj=p.klwoj
where p.powiat like 'M.%'
group by w.województwo
order by w.województwo
```

(d) jednocześnie liczbę miast oraz liczbą miast na prawach powiatu w każdym z województw;

```
create view miasta as
select w.województwo, count(p.powiat) as miasta_na_prawach_powiatu
from powiaty as p
join wojewodztwa as w on w.klwoj=p.klwoj
where p.powiat like 'M.%'
group by w.województwo
order by w.województwo
```

```
select w.województwo, count(m.klmsc) as miasta_w_województwie,
miasta_na_prawach_powiatu from miasta as ms
join wojewodztwa as w on ms.województwo=w.województwo
join powiaty as p on p.klwoj=w.klwoj
join gminy as g on g.klpow=p.klpow
join miejscowosci as m on m.klgm=g.klgm
join typymisc as t on t.kltm=m.kltm
where t.typmisc='miasto'
group by w.województwo, miasta_na_prawach_powiatu
order by wojewodztwo
```

```
drop view miasta
```

(e) liczbę wsi w powiatach danego województwa;

```
select p.powiat, count(m.klmsc) from miejscowosci as m
join gminy as g on g.klgm=m.klgm
join powiaty as p on p.klpow=g.klpow
join wojewodztwa as w on w.klwoj=p.klwoj
join typymisc as t on t.kltm=m.kltm
where typmisc = 'wieś'
and wojewodztwo = 'Dolnośląskie'
group by p.powiat
order by p.powiat
```

(f) powiatową średnią liczby wsi w województwach;

```
create view oky as
select w.wojewodztwo, p.powiat, count(m.klmsc) from miejscowosci as m
join gminy as g on g.klgm=m.klgm
join powiaty as p on p.klpow=g.klpow
join wojewodztwa as w on w.klwoj=p.klwoj
join typymisc as t on t.kltm=m.kltm
where typymisc = 'wieś'
group by w.wojewodztwo, p.powiat
order by w.wojewodztwo, p.powiat

select wojewodztwo, round(avg(count),2) from oky
group by wojewodztwo

drop view oky
```

(g) o polach wojewodztwo, nazwa powiatu o największej liczbie miast w województwie;

```
select w.wojewodztwo, p.powiat, count(m.klmsc) as
ilosc_miast_w_powiecie from miejscowosci as m
join typymisc as t on t.kltm=m.kltm
join gminy as g on g.klgm=m.klgm
join powiaty as p on p.klpow=g.klpow
join wojewodztwa as w on w.klwoj=p.klwoj
where typymisc = 'miasto'
group by w.wojewodztwo, p.powiat
order by w.wojewodztwo, count(m.klmsc) desc
```

(h) kierunek, rok studiów, grupę i liczbę osób w grupie.

```
select k.kierunek, g.rok, g.grupa, count(gs.nralbumu) from grupyst
as gs
join grupy as g on g.klgr=gs.klgr
join kierunki as k on k.klkier=g.klkier
group by k.kierunek, g.rok, g.grupa
order by k.kierunek, g.rok, g.grupa
```

8. Zbuduj tabele:

(a) powiatów w których jest jedno miasto;

```
create view powiaty_z_miastem as
select p.powiat, count(klmsc) from miejscowosci as m
join gminy as g on g.klgm=m.klgm
join powiaty as p on p.klpow=g.klpow
join typymisc as t on t.kltm=m.kltm
where typymisc = 'miasto'
group by p.powiat
order by count

select powiat from powiaty_z_miastem
where count = 1
```

```
drop view powiaty_z_miastem
```

(b) studentów znających dokładnie 2 języki;

```
create view zna_dwa_jezyki as
select o.imie, o.nick, count(j.klj) from osoby as o
join znajezyki as zn on o.klos=zn.klos
join jezyki as j on j.klj=zn.klj
group by o.imie, o.nick
```

```
select imie, nick from zna_dwa_jezyki
where count = 2
order by imie, nick
```

```
drop view zna_dwa_jezyki
```

albo

```
select o.imie, o.nick, count(j.klj) from osoby as o
join znajezyki as zn on o.klos=zn.klos
join jezyki as j on j.klj=zn.klj
group by o.imie, o.nick
having count(*)=2
```

(c) o polach: województwo, liczba powiatów, liczba gmin;

```
select w.wojewodztwo, count(distinct gmina) as "Ilosc gmin",
count(distinct powiat) as "Ilosc powiatow" from wojewodztwa as w
join powiaty as p on w.klwoj = p.klwoj
join gminy as g on p.klpow = g.klpow
group by w.wojewodztwo
order by w.wojewodztwo
```

(d) o polach: województwo, liczba miast, liczba wsi, liczba leśnictw i leśniczówek.

```
select wojewodztwo, count((select typmsc where typmsc='miasto')) as
"liczba miast",
count((select typmsc where typmsc='wieś')) as "liczba wsi",
count((select typmsc where typmsc='leśnictwo')) as "liczba
leśnictw",
count((select typmsc where typmsc='leśniczów.'))as "liczba
leśniczowek" from wojewodztwa as w
join powiaty as p on w.klwoj=p.klwoj
join gminy as g on p.klpow=g.klpow
join miejscowosci as m on g.klgm=m.klgm
join typymisc as t on m.kltm=t.kltm
group by wojewodztwo
order by wojewodztwo
```

EGZAMINY

Grupa 1

Zadanie 1. Zbuduj listę imię,nick,kierunek,rok,grupa,powiat studentów bioinformatyki

```
select o.imie,o.nick,k.kierunek,g.rok,g.grupa,p.powiat from osoby as  
o  
join studenci as s on o.klos = s.klos  
join grupyst as gs on gs.nralbumu = s.nralbumu  
join grupy as g on g.klgr = gs.klgr  
join kierunki as k on k.klkier = g.klkier  
join miejscowosci as m on o.klmsc = m.klmsc  
join gminy as gm on gm.klgm = m.klgm  
join powiaty as p on p.klpow = gm.klpow  
where kierunek= 'Bioinformatyka'  
order by o.imie,o.nick
```

Zadanie 2. Liczba wsi w danym województwie

```
select w.wojewodztwo,count(t.typmsc) as "Liczba wsi w danym  
wojewodztwie"  
from wojewodztwa as w join powiaty as p on w.klwoj = p.klwoj  
join gminy as g on g.klpow = p.klpow  
join miejscowosci as m on m.klgm = g.klgm  
join typymsc as t on t.kltm = m.kltm  
where t.typmsc = 'wieś'  
group by w.wojewodztwo  
order by w.wojewodztwo
```

Zadanie 3. Powiat z największą liczbą miast

```
select p.powiat,count(t.typmsc) as "Liczba miast w powiecie" from  
powiaty as p  
join gminy as g on p.klpow = g.klpow  
join miejscowosci as m on g.klgm = m.klgm  
join typymsc as t on t.kltm = m.kltm  
where t.typmsc = 'miasto'  
group by p.powiat  
order by count(t.typmsc) desc limit 1
```

Zadanie 4. Osoby nieznające języka angielskiego

```
select o.imie,o.nick,j.jezyk from jezyki as j  
join znajezyki as z on j.klj = z.klj  
join osoby as o on o.klos = z.klos  
where j.jezyk <> 'angielski' and o.klos not in (select klos from  
znajezyki where klj in (select klj from jezyki  
where jezyk = 'angielski'))
```

Grupa 2

Zadanie 1. Lista imię, nick, kierunek, rok, grupa, poziom języka i język studentów bioinformatyki

```
select o.imie, o.nick, k.kierunek, g.rok, g.grupa, pz.poziom,  
j.jezyk from studenci as s  
join osoby as o on s.klos = o.klos  
join znajezyki as zn on o.klos = zn.klos  
join jezyki as j on zn.klj = j.klj  
join poziomozn as pz on zn.klpz = pz.klpz  
join grupyst as gr on s.nralbumu = gr.nralbumu  
join grupy as g on gr.klgr = g.klgr  
join kierunki as k on g.klkier = k.klkier  
where kierunek = 'Bioinformatyka'  
order by kierunek
```

Zadanie 2. Liczbę studentów bioinformatyki zamieszkałych w powiatach

```
select p.powiat, count(o.klos) from osoby as o  
join studenci as s on s.klos=o.klos  
join grupyst as gr on gr.nralbumu=s.nralbumu  
join grupy as g on g.klgr=gr.klgr  
join kierunki as k on k.klkier=g.klkier  
join miejscowosci as m on m.klmsc=o.klmsc  
join gminy as gm on gm.klgm=m.klgm  
join powiaty as p on p.klpow=gm.klpow  
where k.kierunek = 'Bioinformatyka'  
group by p.powiat  
order by count desc
```

Zadanie 3. Liczbę studentów mieszkającą w konkretnych typach miejscowości

```
select typmsc, count(o.klos) from osoby as o  
join studenci as s on s.klos=o.klos  
join grupyst as gr on gr.nralbumu=s.nralbumu  
join grupy as g on g.klgr=gr.klgr  
join kierunki as k on k.klkier=g.klkier  
join miejscowosci as m on m.klmsc=o.klmsc  
join typymsc as t on t.kltm=m.kltm  
where k.kierunek = 'Bioinformatyka'  
group by typmsc  
order by count desc
```

Zadanie 4. Ilość rodzeństwa osób w bazie

```
select o.imie, o.nick, count(r.klrodz) from osoby as o  
join rodzenstwo as r on r.klos=o.klos  
group by o.imie, o.nick
```

order by count desc

Grupa 3.

Zadanie 1. Lista grup i średnia liczba znanych języków przez wszystkich studentów danej grupy

```
create view logi as
select grupa, count(jezyk), o.nick from grupy as g
join grupyst as gs on g.klgr=gs.klgr join studenci as s on
gs.nralbumu=s.nralbumu
join osoby as o on s.klos=o.klos join znajezyki as zn on
o.klos=zn.klos
join jezyki as j on j.klj=zn.klj join poziomozn as pz on
zn.klpz=pz.klpz group by grupa, jezyk, o.nick
```

```
create view logil as
select nick, count(count), grupa from logi group by grupa, nick
```

```
select grupa, round(avg(count), 0) as "Średnia znanych jezykow" from
logil
group by grupa order by grupa asc
```

Zadanie 2. Studenta mającego największą ilość języków

```
select o.imie, o.nick, count(j.klj) from osoby as o
join znajezyki as zn on zn.klos=o.klos
join jezyki as j on j.klj=zn.klj
group by o.imie, o.nick
order by count desc limit 1
```

Zadanie 3. Liczbę pań o wzroście większym niż średni wzrost pań

```
select count(klos) from osoby
where plec = 'K' and wzrost > (select avg(wzrost) from osoby where
plec= 'K')
```

Zadanie 4. Liczbę studentów i studentek w grupach ćwiczeniowych

```
select grupa, plec, count(o.klos) as "liczba studentow" from osoby
as o
join studenci as s on o.klos=s.klos
join grupyst as grs on s.nralbumu=grs.nralbumu
join grupy as g on grs.klgr=g.klgr
group by grupa, plec
order by grupa asc
```

Grupa 4.

Zadanie 1. Lista studentów znających tylko 1 język

```
select o.imie, o.nick, count(j.klj) from osoby as o
join znajezyki as zn on zn.klos=o.klos
join jezyki as j on j.klj=zn.klj
group by o.imie, o.nick
having count(*)=1
order by o.imie,o.nick
```

Zadanie 2. Województwo o największej ilości gmin wiejskich (gmina bez żadnego miasta)

```
create view gminywiej as
select w.wojewodztwo, g.gmina, count(m.miejscowosc) from typmsc as
t
join miejscowosci as m on m.kltm=t.kltm
join gminy as g on g.klgm=m.klgm
join powiaty as p on p.klpow=g.klpow
join wojewodztwa as w on w.klwoj=p.klwoj
where typmsc <> 'miasto' and typmsc <> 'cz.miasta'
group by w.wojewodztwo, g.gmina

select wojewodztwo, count(gmina) as "liczba gmin wiejskich w
wojewodztwie" from gminywiej
group by wojewodztwo
order by "liczba gmin wiejskich w wojewodztwie" desc limit 1

drop view gminywiej
```

Zadanie 3. Liczba studentów znająca język angielski i rosyjski na poziomie co najmniej dobrym

```
select imie, nick from osoby
where klos in(select klos from znajezyki where klj in (select klj
from jezyki where jezyk='rosyjski'))
and klpz in(select klpz from poziomyzn where ranga>2))
and klos in(select klos from znajezyki where klj in (select klj from
jezyki where jezyk='angielski'))
and klpz in(select klpz from poziomyzn where ranga>2))
```

Zadanie 4. Liczbę osób z tego samego roku i miesiąca

```
select imie, nick, dataur from osoby where extract (year from
dataur)=2000 and extract (month from dataur)=6
```

Grupa 5.

Zadanie 1. Liczba osób zamieszkałych w tych samych miejscowościach

```
select m.miejscowosc, count(o.klos) from osoby as o
join miejscowosci as m on m.klmsc=o.klmsc
group by m.miejscowosc
order by count desc
```

Zadanie 2. Lista poziomów znajomości języka angielskiego oraz liczbę studentów znających język na tym poziomie

```
select pz.poziom, count(o.klos) from osoby as o
join znajemyki as zn on zn.klos=o.klos
join jezyki as j on j.klj=zn.klj
join poziomyzn as pz on pz.klpz=zn.klpz
where j.jezyk='angielski'
group by pz.poziom
order by pz.poziom
```

Zadanie 3. Liczbę studentów wyższych niż najwyższa studentka

```
select count(klos) from osoby
where wzrost > (select max(wzrost) from osoby where plec ='K')
and plec = 'M'
```

Zadanie 4. Lista powiatów z dokładnie dwoma miastami

```
select p.powiat, count(m.klmsc) as "ilosc miast w powiecie" from
miejscowosci as m
join typymsc as t on t.kltm=m.kltm
join gminy as g on g.klgm=m.klgm
join powiaty as p on g.klpow=p.klpow
where typymsc = 'miasto'
group by p.powiat
having count(*)=2
order by p.powiat
```


