*--Zad 1*

**Pokaż atrybuty tabeli Studenci**

*desc studenci;*

*--Zad 2*

**Wyświetl wszystkie wiersze znajdujące się w tabeli Studenci**

*Select \* from Studenci;*

*--Zad 3.1*

**Wyświetl listę zawierającą nazwiska i imiona wszystkich studentów.**

*Select imiona, nazwisko from studenci;*

*--Zad 3.2*

**Wyświetl informacje dotyczącą kierunków (bez powtórzeń) na jakich studiują studenci, których dane znajdą się w tabeli**

*Select distinct kierunek from studenci;*

*--Zad 3.3*

**Wyświetl listę prezentującą nazwiska, imiona, tryb studiów, rok, grupę i specjalność studentów studiujących na kierunku informatyka. Dane uporządkuj wg. roku w formie niemalejącej (od 1) i nazwiska w formie „od Ż do A”**

*Select imiona, nazwisko, tryb, rok, gr\_dziekan, specjalnosc from studenci where kierunek like 'INF%' order by rok, nazwisko desc;*

*--Zad 3.4*

**Wyświetl dane studentów studiujących na 3-im roku i przynależących do 2-ej grupy dziekańskiej**

*Select \* from studenci where rok=3 and gr\_dziekan=2;*

*--Zad 3.5*

**Wyświetl dane studentów, którzy nie mają określonej specjalności.**

*Select \* from studenci where specjalnosc is null;*

*--Zad 3.6*

**Wyświetl nazwiska, imiona i roczniki studentów (rok) pierwszego stopnia studiów stacjonarnych z lat (rok): 2, 3, 4 (warunek zbuduj na kilka sposobów). Dane wyświetl uporządkowane w kolejności alfabetycznej wg nazwisk studentów**

*Select imiona, nazwisko, rok from studenci where stopien=1 and tryb like 'S%';*

*--Zad 3.7*

**Wyświetl imiona i nazwiska studentek.**

*Select imiona, nazwisko from studenci where imiona like '%a';*

*--Zad 3.8*

**Wyświetl imiona i nazwiska studentów (panów).**

*Select imiona, nazwisko from studenci where imiona not like '%a';*

*--Zad 3.9*

**Wyświetl dane wszystkich Adamów, Konradów i Magdalen**

*Select \* from studenci where imiona in('Adam', 'Konrad', 'Magdalena');*

*--Zad 3.10*

**Wyświetl dane wszystkich Kowalskich oraz Nowaków**

*Select \* from studenci where nazwisko like 'Kowalsk\_' or nazwisko like 'Nowak';*

*--Zad 3.11*

**Wyświetl alfabetycznie uporządkowaną listę imion studentów, których dwie pierwsze litery mieszczą się w przedziale alfabetycznym od DO do MI (np. Dorota, Helena, Jan, Marcin itd.;**

*Select imiona from studenci where upper(imiona) between 'DO' and 'MI';*

*--Zad 4*

**Pokaż atrybuty tabeli Pracownicy.**

*Desc Pracownicy;*

*--Zad 5.1*

**Wyświetl listę zawierającą nazwiska pracowników i ich płacę (kolumna placa).**

*Select nazwisko, placa from Pracownicy;*

*--Zad 5.2*

**Wyświetl listę zawierającą nazwiska pracowników wraz z wartościami ich tzw. „dniówek” obliczonych na podstawie wartości znajdujących się w ramach kolumny placa (Rys. 9). W ramach zadania należy przyjąć założenie, że w każdym miesiącu występuje 20 dni roboczych.**

*Select nazwisko, placa/20 as „dniowka” from Pracownicy;*

*--Zad 5.3*

**Wyświetl listę miesięcznych całkowitych pensji aktualnych pracowników (nazwisko, nr\_akt) pracujących w ramach umowy na czas nieokreślony (nieokreślona wartość kolumny data\_zwol); lista ma być uporządkowaną po kolumnie pensja w trybie niemalejącym (od najgorzej do najlepiej zarabiającego pracownika; patrz Rys. 10); uwaga w ramach pensji należy uwzględnić wartości z kolumn placa, dod\_funkcyjny, dod\_staz i koszt\_ubezpieczenia.**

*Select nazwisko, imiona, placa+placa\*NVL(dod\_staz,0)\*0.01+NVL(dod\_funkcyjny,0)-koszt\_ubezpieczenia as pensja from pracownicy where data\_zwol is null;*

*--Zad 5.4*

**Wyświetl listę rocznych pensji aktualnych pracowników posiadających umowy na czas nieokreślony, niepełniących funkcji prezesa oraz dyrektora. Lista ma być uporządkowana od najlepiej do najgorzej zarabiającego pracownika**

*Select nazwisko,nr\_akt, (placa+placa\*dod\_staz\*0.01-nvl(dod\_funkcyjny,0)-nvl(koszt\_ubezpieczenia,0))\*12 placa from pracownicy where data\_zwol is null and stanowisko not in('Dyrektor','Prezes') order by 3 desc;*

*--Zad 5.5*

**Wyświetl pod postacią jednej kolumny informującą w jakich działach i na jakich stanowiskach pracują aktualni pracownicy firmy zatrudnieni ma umowy na czas nieokreślony**

*Select nazwisko||' '|| 'aktualnie pracuje w dziale'||' '||id\_dzialu||'-tym na stanowisku '||stanowisko||' w firmie od '|| data\_zatr as "Aktualni pracownicy" from pracownicy where data\_zwol is null;*

*--Zad 6*

**Pokaż atrybuty tabeli Pojazdy.**

*Desc Pojazdy;*

*--Zad 7*

**Na podstawie danych znajdujących się w tabeli Pojazdy wyświetl listę zawierającą informacje o samochodach osobowych których (Rys. 13): - numer rejestracyjny rozpoczyna się na 'SC' a kończy cyfrą 5, - wartość pojemności silnika nie należy do przedziału od 1500 do 2000, - nazwa koloru jest ustalona w formie jednego wyrazu, w którym występuje co najmniej jedna litera 'z'.**

*Select \* from pojazdy where trim(nr\_rejestr) like 'SC%5' and pojemnosc not between 1500 and 2000 and kolor like '%z%' and kolor not like '% %';*

--Zad 1.1

**Wyświetl informacje o liczbie studentek studiów stacjonarnych pierwszego stopnia studiujących na 3-im roku informatyki.**

*Select count(\*) as " Liczba studentek 3r INF ss" from studenci where imiona like '%a' and tryb like 'STACJONARNY' and stopien=1 and rok=3 AND KIERUNEK= 'INFORMATYKA';*

--Zad 1.2

**Wyświetl informacje o liczbie Nowakowskich (nazwisko) wśród studentów (w zapytaniu wykorzystaj jedną z funkcji: lower lub upper).**

*select count(nazwisko) as "Liczba Nowakowskich" from Studenci where nazwisko like 'Nowakowsk%';*

--Zad 1.3

**Wyświetl informacje (w ramach jednego zapytania) na temat ogólnej liczby studentów mężczyzn, których imię rozpoczyna się na literę M oraz ile takich imion (różnych) znajduje się w tabeli.**

*Select count(imiona) as " liczba studentów na litere M", count(DISTINCT imiona) as "Liczba roznych imiona na M" from studenci where imiona like 'M%' and imiona not like '%a';*

--Zad 1.4

**Wyświetl uporządkowaną alfabetycznie listę zawierającą imiona i nazwiska doktorantów 4-tego roku (studia 3-ego stopnia) w postaci pojedynczych łańcuchów – patrz Rys. 1.4 (min. 2 sposoby w tym zastosuj funkcję concat).**

*Select concat(imiona,nazwisko) nazwa from studenci where stopien=3 and rok=4;*

--Zad 1.5

**Wyświetl listę, w ramach której jedna z kolumn będzie zwierała pierwsze 3 litery imion a druga kolumna ostatnie 3 litery nazwisk studentów nieposiadających zdefiniowanej specjalności.**

*select substr(imiona,1,3),substr(nazwisko,-3),imiona,nazwisko from studenci*

--Zad 1.6

**Wyświetl listę zawierającą nazwiska i imiona studentów wraz z ich inicjałami w formie 'A.A.', których łączna liczba liter w imieniu i nazwisku wynosi 9, 11 lub 13.**

*Select substr(imiona,1,1)||'.'||substr(nazwisko,1,1)||'.' as "Inicjaly", imiona,nazwisko from studenci where length(concat(nazwisko,imiona)) in(9,11,13);*

*--Zad 1.7*

**Wyświetl listę zawierającą kierunki studiów reprezentowane przez studentów w formie pierwsza litera każdego wyrazu będzie duża a pozostałe małe oraz tylko pierwsza litera będzie duża a pozostałe małe (patrz Rys. 1.7); zadanie wykonaj na 2 sposoby poprzez zastosowanie funkcji Initcap i Concat.**

*Select DISTINCT Initcap(kierunek) from studenci;*

--Zad 1.8

**Wyświetl listę, w ramach której pierwsza kolumna będzie zawierać nazwiska studentów bez pierwszych dwóch liter 'Ko' a druga kolumna imiona tych studentów bez dwóch ostatnich liter 'sz' (patrz Rys. 1.8). Oznacza to, że w wyniku należy uwzględnić tylko studentów, których nazwiska rozpoczynają się na 'Ko' a imiona kończą na 'sz' (bez zastosowania funkcji substr).**

*Select Substr(nazwisko,3) as "nazwisko bez sz", Substr(imiona,1,Length(imiona)-2) as*

*"Imie bez Ko" from studenci where nazwisko like 'Ko%' and imiona like '%sz';*

--Zad 1.9

**Wyświetl listę studentów 2-ego roku, których nazwiska składają się od 6 do 9 liter, z których przynajmniej jedna litera to a, wraz z informacją o liczbie liter w nazwisku oraz pozycji wystąpienia pierwszej litery A (patrz Rys. 1.9). Lista ma być uporządkowana nierosnąco wg liczby liter w nazwisku.**

*Select Instr(upper(nazwisko),'A'),nazwisko, length(nazwisko) from studenci where length(nazwisko) between 6 and 9 and Instr(upper(nazwisko),'A')>0*

--Zad 1.10

**Wyświetl listę (patrz Rys. 1.10) zawierającą zmodyfikowane nazwiska i imiona studentek, których nazwiska rozpoczynają się od liter Ba (bez zastosowania funkcji substr): - zamiast Ba na początku nazwiska wstaw formułę 'Start', - zamiast litery a na końcu imienia staw formułę 'End'.**

--Zad 1.11

**Wykorzystując funkcje Rpad() i Lpad() wyświetl nazwiska studentów poprzedzone 3 gwiazdkami (\*) oraz zakończone 4 plusikami (+)**

*Select Rpad(Lpad(nazwisko,length(nazwisko)+3,'\*'),length(nazwisko)+7,'+') from studenci*

--Zad 2.1

**Wyświetl wszystkie pojazdy zarejestrowane w Częstochowie (o numerach rejestracyjnych rozpoczynających się od SC i posiadających cyfry na 3-ich pozycjach, np. SCZ to powiat częstochowski a nie miasto:)) o pojemnościach silników mieszczących się w przedziale od 1000 do 2000 cm 3 .**

*Select \* from pojazdy where nr\_rejestr between 'SC0%' and 'SC99999' and pojemnosc between 1000 and 2000*

--Zad 2.2

**Wyświetl pojazdy marki Ford w kolorze metalik (np. zielony metalik), których liczby utworzone przez dwie ostatnie cyfry numeru rejestracyjnego dzielą się przez liczbę 3 np. SC12345 → 45 / 3 =15 (patrz Rys. 2.2). Wszystkie numery rejestracyjne kończą się się 3 lub 4-ma cyframi (konwersja na liczbę → to\_number()).**

*select nr\_rejestr, substr(Nr\_Rejestr,6) as "Liczba" , Marka, Kolor from pojazdy where kolor like '%talik%' and marka like 'Ford' and Mod(to\_number(substr(Nr\_Rejestr,6)),3)=0*

--Zad 2.3

**Wyświetl wszystkie motocykle posiadające w numerach rejestracyjnych przynajmniej dwie cyfry 6 (np. SCZ1626), których kolory określają przynajmniej dwuczłonowe nazwy (np. zielony metalik) a pojemności silników nie mieszczą się w przedziale od 250 do 500**

*select \* from pojazdy where typ like 'motocykl' and NR\_REJESTR like '%6%6%' and kolor like '% %' and pojemnosc between 250 and 500;*

--Zad 2.4

**Wyświetl wszystkie pojazdy innego typu niż samochod ciezarowy mające pojemności silnika równe 1000, 2000 lub 3000 wraz z komentarzami w zakresie oceny tego parametru (Rys. 2.4): - maly pojazd wobec pojazdów o pojemności 1000, - sredni pojazd wobec pojazdów o pojemności 2000, - duzy pojazd wobec pojazdów o pojemności 3000.**

*Select marka,modell, typ, pojemnosc, decode(pojemnosc,1000,'maly pojazd',2000,'sredni pojazd',3000,'duzy pojazd')from pojazdy where typ not like 'samochod ciezarowy' AND pojemnosc in(1000,2000,3000);*

--Zad 2.5

**W oparciu o analizę numerów rejestracyjnych zidentyfikuj pochodzenie pojazdów marki Opel o pojemnościach silnia nie należących do przedziału [1600, 2200]. W procesie klasyfikacji pojazdów uwzględniaj dwie pierwsze litery numerów rejestracyjnych wg następujących zasad (patrz Rys. 2.5): – SC (równocześnie SCZ), SK – slaskie, – OP – opolskie, – DW – dolnoslaskie, – KR – malopolskie, – inne – niezidentyfikowane.**

*select nr\_rejestr, modell, pojemnosc, decode(substr(nr\_rejestr,1,2),'SK','Slaskie',*

*'OP','Opolskie',*

*'DW','Dolnoslaskie',*

*'KR','Malopolskie',*

*'niezidentyfikowane')*

*from pojazdy where marka like 'Opel' and pojemnosc not between 1600 and 2200*

--Zad 3.1

**W postaci jednego zdania (jak na Rys. 3.1) wyświetl informację od kiedy do kiedy odnotowywano zdarzenia połowów ryb, ile było wszystkich takich zdarzeń w tym ile było udanych połowów (określona wartość kolumny id\_gatunku) oraz wody ilu zarządców odwiedzili wędkarze (założenie: elementem identyfikującym zarządców wód jest pierwszy znak identyfikatora łowiska czyli id\_lowiska).**

*Select 'OD '||trunc(min(czas))||'do '|| trunc(max(czas))||' odnotowano'|| count(czas) ||' polowow w tym udanych '|| count(ID\_Gatunku)|| 'na wodach '||count(distinct substr(id\_lowiska,1,1))|| 'zarzadcow ' from rejestry;*

--Zad 3.2

**Wyświetl listę zawierającą wszystkie połowy dotyczące ryb o identyfikatorach (id\_gatunku) 1, 3, 9 lub 10 złowione na wodach zarządzanych przez PZW Częstochowa (id\_lowiska rozpoczynające się od litery C) o długościach od 40 do 60 cm., których waga została ustalona z precyzją nie większą niż 0.1**

*select \* from rejestry where*

*id\_gatunku like '1' or*

*id\_gatunku like '3' or*

*id\_gatunku like '9' or*

*id\_gatunku like '10' and*

*id\_lowiska like 'C%' and*

*Dlugosc between 40 and 60;*

--Zad 3.3

**Wyświetl statystyki połowów gatunku o identyfikatorze 1 w zakresie liczby złowionych sztuk, liczby łowców (różnych wędkarzy, którzy złowili ten gatunek), liczby łowisk (różnych łowisk, na których odnotowano połów tego gatunku) , łącznej wagi wszystkich złowionych ryb oraz ich średniej wagi (z dokładnością do 1 grama) i średniej długości**

*Select count(\*),count(distinct id\_wedkarza),count(distinct id\_lowiska), sum(waga),round(avg(waga),3),round(avg(dlugosc)) from rejestry where id\_gatunku=1*

--Zad 3.4

**Wyświetl podstawowe informacje związane z wpisami dotyczącymi zdarzeń złowienia ryb o następujących identyfikatorach 2 (lin), 4 (amur), 15 (ploc), 17 (okon) oraz dotyczącymi nieudanych połowów (nieokreślona wartość id\_gatunku → brak polowu). Informacje podaj w formie: dzien polowu, id\_gatunku oraz słowna nazwa gatunku**

*Select trunc(czas), id\_gatunku,decode(2,'lin',4,'amur',15,'ploc',17,'okon','brak polowu') from rejestry where id\_gatunku in(2,4,15,17) or id\_gatunku is null;*

--Zad 1

**W oparciu o dane zawarte w tabelach Pojazdy i Kierowcy wyświetl listę pojazdów wyprodukowanych w 2018 roku wraz z personaliami ich właścicieli**

*select nr\_rejestr, marka, modell, data\_prod, wlasciciel, nazwisko, imie, data\_urodzenia*

*from pojazdy join kierowcy on(wlasciciel=id\_kierowcy) where extract(year from data\_prod)=2018;*

--Zad 2

**W oparciu o dane zawarte w tabelach Pojazdy i Kierowcy wyświetl listę zawierającą dane pojazdów starszych od swoich właścicieli. Lista poza danymi pojazdów i ich właścicieli powinna zawierać informację o wielkości różnicy pomiędzy wiekiem pojazdu i wiekiem jego właściciela, podanej w dniach i pełnych latach. Dane uporządkuj wg różnicy wieku w trybie nierosnącym**

*Select nr\_rejestr, marka , modell, data\_prod ,wlasciciel, nazwisko, imie, data\_urodzenia,*

*data\_urodzenia-data\_prod,trunc(months\_between(data\_urodzenia,data\_prod)/12)*

*from pojazdy join kierowcy on (wlasciciel=id\_kierowcy) where*

*data\_prod<data\_urodzenia order by 10 desc;*

--Zad 3

**W oparciu o dane zawarte w tabelach Pracownicy i Stanowiska wyświetl listę pracowników, których płaca nie mieści się w widełkach wynagrodzeń przewidzianych dla stanowisk, na których aktualnie pracują. Zawartość listy zgodnie z Rys. 3 (bez analizy okresów zatrudnienia).**

*select nr\_akt, stanowisko, placa, placa\_min, placa\_max*

*from pracownicy join stanowiska using(stanowisko) where placa not between placa\_min and placa\_max;*

--Zad 4

**W oparciu o dane zawarte w tabelach Rejestry, Wedkarze, Gatunki i Lowiska wyświetl listę wszystkich udanych połowów ryb podając dane wędkarza, nazwy gatunków i łowisk. Dane wyświetl chronologicznie wg dat zdarzeń połowów**

*Select trunc(czas), nazwisko, imie, lowiska.nazwa, gatunki.nazwa, waga, dlugosc from rejestry*

*join wedkarze using(id\_wedkarza) left join gatunki using(id\_gatunku) join lowiska using(id\_lowiska)*

*order by czas;*

--Zad 5

**Zadanie stanowiące rozszerzenie zadania 4. W oparciu o dane zawarte w tabelach Rejestry, Wedkarze, Gatunki i Lowiska wyświetl informacje o wszystkich próbach połowów ryb na przestrzeni ostatniego roku. W przypadku braku połowu w ramach kolumny gatunek umieść frazę „ brak polowu” (patrz Rys. 5). Dane wyświetl chronologicznie wg dat zdarzeń połowów**

*select trunc(czas), nazwisko, imie, NVL(gatunki.nazwa, 'brak polowu'),*

*lowiska.nazwa, dlugosc, waga from rejestry join lowiska using(id\_lowiska) join wedkarze using (id\_wedkarza) left join gatunki using (id\_gatunku)*

*where czas between systimestamp-interval'1'year(1) and systimestamp*

*order by 1;*

--Zad 6

**W oparciu o dane zawarte w tabelach Licencje i Wedkarze wyświetl listę wędkarzy, którzy w aktualnym roku kalendarzowym opłacili już roczną składkę podstawową na rzecz dowolnego okręgu PZW**

--Zad 7

**W oparciu o dane zawarte w tabelach Licencje i Wedkarze wyświetl listę wszystkich wniesionych opłat okresowych (dziennych) w 2018 roku.**

*select id\_wedkarza, nazwisko, id\_okregu, id\_licencji, od\_dnia poczatek, do\_dnia koniec*

*from licencje join wedkarze using(id\_wedkarza)*

*where rodzaj like 'dodatkowa' and rok=2018*

*and not(od\_dnia like'01-01' and do\_dnia like '31-21');*

--Zad 8

**Zadanie stanowiące rozszerzenie zadania 7. W oparciu o dane zawarte w tabelach Licencje, Oplaty i Wedkarze wyświetl listę wszystkich wniesionych opłat okresowych (dziennych) w 2018 roku. W ramach prezentowanej informacji podaj liczbę dni obowiązywania danej licencji oraz jej koszt (patrz Rys. 8; tzw. nocka tj. połów rozpoczynający się pod wieczór a kończący się w kolejnym dniu to w sumie 2 dni).**

*select \* from licencje;*

*select id\_wedkarza,nazwisko,id\_okregu,od\_dnia,do\_dnia,*

*(to\_date(do\_dnia||'-'||rok,'dd-mm-yyyy')-to\_date(od\_dnia||'-'||rok,'dd-mm-yyyy')+1) as "Liczba dni",*

*(to\_date(do\_dnia||'-'||rok,'dd-mm-yyyy')-to\_date(od\_dnia||'-'||rok,'dd-mm-yyyy')+1)\*dzienna\_oplata as"oplata"*

*from licencje join oplaty using(rok,id\_okregu) join wedkarze using(id\_wedkarza) where trim(rodzaj) like'dodatkowa'*

*and not(od\_dnia like '01-01' and do\_dnia like'31-12');*

--Zad 9

**W oparciu o dane zawarte w tabeli Pracownicy wyświetl listę porównującą parami place pracowników działu 20-ego z płacami pracowników działu 30-ego (bez analizy okresu zatrudnienia; patrz Rys. 9).**

*select*

*p1.id\_dzialu,p1.nr\_akt, p1.nazwisko, p1.placa,p2.id\_dzialu,p2.nr\_akt, p2.nazwisko, p2.placa*

*from pracownicy p1 cross join pracownicy p2 where*

*p1.id\_dzialu=20 and p2.id\_dzialu=30;*

*--zad10*

*select ls.nr\_akt,ls.nazwisko,ls.imiona,ls.przelozony,rs.imiona||' '||rs.nazwisko*

*from pracownicy ls left join pracownicy rs on(ls.przelozony=rs.nr\_akt);*

--Zad 10

**W oparciu o dane zawarte w tabeli Pracownicy wyświetl listę pracowników i ich przełożonych (Rys. 10). W przypadku braku przełożonego w ramach kolumny przełożony zastosuj frazę „ Brak” (bez analizy okresu zatrudnienia).**

*select p1.nr\_akt, p1.nazwisko,p1.imiona,p2.nr\_akt,p2.nazwisko || ' ' || p2.imiona from pracownicy p1 left join pracownicy p2 on(p1.przelozony=p2.nr\_akt);*

--Zad 11

**W oparciu o dane zawarte w tabeli Studenci wyświetl informację ilu studentów kierunku informatyka studiuje na danym roku studiów (patrz Rys. 11).**

*Select rok,count(\*) from studenci where lower(kierunek) like 'informatyka' group by rok;*

--Zad 12

**W oparciu o dane zawarte w tabeli Studenci wyświetl informację ilu studentów studiuje dany kierunek w danym trybie studiów. W zestawieniu uwzględnij tylko podzbiory składające się z przynajmniej 100-u studentów (patrz Rys. 12).**

*select tryb,kierunek,count(kierunek) from studenci group by kierunek, tryb having count(kierunek)>=100;*

--Zad 13

**W oparciu o dane zawarte w tabeli Studenci wyświetl informację ile studentek (pań) studiuje w ramach danego stopnia, roku i grupy dziekańskiej na kierunku matematyka. Informację uzupełnij o średnią liczoną z średnich ocen studentek w danej grupie (patrz Rys. 13).**

*select rok,stopien,gr\_dziekan,count(\*)liczba\_studentek,round(avg(srednia),2)*

*from studenci where imiona like '%a' and kierunek like 'MATEMATYKA'*

*group by rok,stopien,gr\_dziekan having count(\*)>=10 and avg(srednia)>=4;*

--Zad 14

**W oparciu o dane zawarte w tabeli Studenci wyświetl informacje o datach urodzeń najstarszego studenta i najmłodszego studenta studiujących na danym kierunku danego roku studiów pierwszego stopnia w trybie stacjonarnym. Listę uporządkuj w trybie nierosnącym wg liczby pełnych miesięcy jakie dzielą daty urodzeń tych studentów pod warunkiem, że różnica ta wynosi przynajmniej 100 miesięcy (patrz Rys. 14).**

*Select kierunek, rok, min(data\_urodzenia) najstarszy, max(data\_urodzenia) najmlodsdzy,trunc(months\_between(max(data\_urodzenia),min(data\_urodzenia))) from studenci where stopien=1 and tryb='STACJONARNY' group by rok,kierunek having trunc(months\_between(max(data\_urodzenia),min(data\_urodzenia)))>=150;*

--Zad 15

**W oparciu o dane zawarte w tabeli Rejestry wyświetl informacje o liczbie wszystkich prób połowów oraz o liczbie udanych połowów zrealizowanych w danym dniu tygodnia (słownie) danego roku kalendarzowego w parzysty dzień miesiąca (2,4, …, 28, ew. 30). Dane wyświetl uporządkowane w trybie niemalejącym kolejno wg liczby wszystkich prób połowów oraz liczby udanych połowów (patrz Rys. 15).**

*select extract(year from czas), to\_char(czas,'day'), count(\*), count(id\_gatunku) from rejestry where mod(extract(day from czas),2)=0 group by extract(year from czas), to\_char(czas,'day') order by 3 desc, 4 desc;*

--Zad 16

**W oparciu o dane zawarte w tabelach Pojazdy i Kierowcy wyświetl informację o kierowcach, którzy posiadają do 5 do 15 pojazdów ciężarowych. Dane uzupełnij o informację w zakresie liczby marek posiadanych przez danego właściciela a wyniki wyświetl uporządkowane w trybie nierosnącym kolejno wg liczby posiadanych pojazdów oraz liczby marek (patrz Rys. 16).**

*select wlasciciel, nazwisko, imie, count(nr\_rejestr), count(distinct marka)*

*from pojazdy join kierowcy on(wlasciciel=id\_kierowcy)*

*where typ like 'samochod ciezarowy' group by wlasciciel, nazwisko, imie*

*having count(nr\_rejestr) between 5 and 15 order by 4 desc;*

--Zad 17

**W oparciu o dane zawarte w tabelach Pracownicy i Dzialy wyświetl informację o średnich wartościach płac (kolumna placa) pracowników aktualnie pracujących w poszczególnych działach firmy z uwzględnieniem grupy pracowników nieposiadających przynależności do żadnego działu. Dane wyświetl uporządkowane w trybie nierosnącym wg średniej płacy (patrz Rys. 17) .**

--Zad 18

**W oparciu o dane zawarte w tabelach Pracownicy i Dzialy wyświetl informacje o kosztach jakie ponosi firma w poszczególnych miastach (z adresu działu) z tytułu wypłacania miesięcznej pensji swoim aktualnym pracownikom (płaca+dodatki). Informację uzupełnij o dane w zakresie liczby wszystkich pracowników pracujących w danym mieście oraz ilu z nich posiada ubezpieczenie. W zestawieniu uwzględnij pracowników „nieposiadających działu” a dane wyświetl w trybie nierosnącym wg kosztów pracowniczych (patrz Rys. 18).**

--Zad 19

**W oparciu o dane zawarte w tabelach Pracownicy i Stanowiska wyświetl informacje w zakresie ilu aktualnie pracowników pracuje na danym stanowisku (z uwzględnieniem nieobstawionych stanowisk), ile wynosi średnia, najniższa i najwyższa płaca pracownika pracującego na danym stanowisku. Dane wyświetl uporządkowane w trybie niemalejącym kolejno wg liczy pracowników oraz średniej płacy (patrz Rys. 19).**

*Select st.stanowisko, count(\*),*

*avg(placa+placa\*dod\_staz\*0.01-nvl(pr.dod\_funkcyjny,0)-nvl(koszt\_ubezpieczenia,0)) srednia\_pensja,*

*min(placa+placa\*dod\_staz\*0.01-nvl(pr.dod\_funkcyjny,0)-nvl(koszt\_ubezpieczenia,0)) min\_pensja,*

*max(placa+placa\*dod\_staz\*0.01-nvl(pr.dod\_funkcyjny,0)-nvl(koszt\_ubezpieczenia,0)) max\_pensja*

*from stanowiska st left join pracownicy pr on(st.stanowisko=pr.stanowisko)where data\_zwol is null or data\_zwol>=sysdate group by st.stanowisko;*

--Zad 20

**W oparciu o dane zawarte w tabelach Rejestry i Gatunki wyświetl statystyki połowów poszczególnych gatunków (włącznie z tym, które nigdy nie zostały nigdy złowione) w zakresie liczby złowionych sztuk, łącznej i średniej ich wagi (z dokładnością do 1 grama) oraz średniej długości (z dokładnością do 0.1 cm.). W zestawieniu uwzględnij również połowy nieudane wstawiając w kolumny id\_gatunku i nazwa kolejno frazy brak i brak połowu (patrz Rys. 20). Dane wyświetl uporządkowane w trybie nierosnącym wg liczby złowionych ryb oraz wg nazwy gatunku (alfabetycznie).**

*select decode(id\_gatunku, null, 'brak', id\_gatunku), nvl(nazwa, 'brak polowu'), count(\*) sztuk,*

*round(sum(nvl(waga,0)),3), round(avg(waga),3), round(avg(dlugosc),3)*

*from rejestry left join gatunki using (id\_gatunku) group by decode(id\_gatunku, null, 'brak', id\_gatunku), nvl(nazwa, 'brak polowu');*

--Zad 21

**W oparciu o dane zawarte w tabelach Rejestry i Lowiska wyświetl informacje o łowikach, na których złowiono przynajmniej 5 ryb i odnotowano przynajmniej 2 nieudane połowy w okresie trwającym 2 lata 21 dni 21 godzin 21 minut i 21 sekund poczynając od godz. 15:15 w dniu 11.03.2016. Informację uzupełnij o liczbę wędkarzy jacy przynajmniej raz odwiedzili dane łowisko w rozpatrywanym okresie (patrz Rys. 21).**

*select id\_lowiska, nazwa, count(id\_gatunku), count(distinct id\_wedkarza)*

*from rejestry join lowiska using(id\_lowiska) where czas between timestamp '2016-03-11 15:15:00'*

*and timestamp '2016-03-11 15:15:00'+interval '1' year+interval '21 21:21:21' day(2) to second*

*group by id\_lowiska, nazwa having count(id\_gatunku)>=5 and count(czas)-count(id\_gatunku)>=2;*

--Zad 22

**W oparciu o dane zawarte w tabelach Okregi, Oplaty i Licencje wyświetl informacje ile licencji przez ilu wędkarzy (różnych) zostało wykupionych w danym roku u danego zarządcy wód. W zestawieniu uwzględnij wszystkie oferty z poszczególnych lat (dane w tabeli Oplaty) włącznie z tymi, z których nie skorzystał żaden wędkarz. Dane wyświetl uporządkowane kolejno wg roku i identyfikatora okręgu (patrz Rys. 22).**

*Select \* from licencje;*

*Select \* from oplaty;*

*Select rok, id\_okregu,count(id\_licencji), count(distinct id\_wedkarza) from licencje right join oplaty using(rok,id\_okregu) join okregi using(id\_okregu) group by rok,id\_okregu;*

--Zad 23

**Na podstawie danych zwartych w tabelach Pojazdy i Kierowcy wyświetl listę kierowców, którzy nie posiadają żadnego pojazdu lub posiadają dokładnie 3 pojazdy 3-ech różnych marek. Dane wyświetl uporządkowane kolejno wg liczby posiadanych pojazdów (w trybie niemalejącym), nazwisk i imion (w trybie alfabetycznym; patrz Rys. 23)**

*Select \* from kierowcy;*

*Select \* from pojazdy;*

*Select id\_kierowcy,nazwisko,imie, count(\*), count(distinct marka) from kierowcy left join pojazdy on(id\_kierowcy=wlasciciel)*

*group by id\_kierowcy,nazwisko, imie having count(distinct nr\_rejestr)=0 or count(distinct marka)=3 and count(nr\_rejestr)=3;*

--Zad 24

**Na podstawie danych zwartych w tabelach Rejestry i Gatunki wyświetl informację o tym ile procent rekordowej wagi odnotowanej w ramach danego gatunku (kolumna rekord\_waga w tabeli Gatunki) stanowiła masa najcięższej złowionej ryby tego gatunku, której połów odnotowano w tabeli Rejestry. W wyniku wyświetl tylko gatunki, w ramach których odnotowano połów ryby o masie wynoszącej przynajmniej 25% masy rekordu (podając jaki procent rekordowej wagi stanowi waga złowionej ryby) oraz gatunki, których nie złowiono nawet jednej sztuki (ze stosowanym komentarzem, patrz Rys. 24). Dane uporządkuj według kolumny procentu rekordu**

--Zad 1

**W oparciu o dane zgromadzone w tabeli Studenci wyświetl dane związane z najstarszym *studentem (ew. studentami) studiującym na kierunku informatyka (Rys. 1).***

Select nr\_indeksu,nazwisko,imiona,data\_urodzenia,adres from studenci

where kierunek like 'INFORMATYKA' and data\_urodzenia=(Select min(data\_urodzenia) from studenci where kierunek like 'INFORMATYKA');

--Zad 2

**Na podstawie danych zwartych w tabelach Rejestry, Wedkarze, Gatunki i Lowiska wyświetl informacje (patrz Rys. 2) dotyczące połowu najdłuższej ryby danego gatunku w danym roku. W prezentowanej informacji należy uwzględnić datę połowu, nazwisko łowcy (wędkarza) i miejsce połowu (nazwa łowiska). Dane uporządkuj według roku a w dalszej kolejności według nazwy gatunku.**

*select \* from rejestry join wedkarze using (id\_wedkarza) join lowiska using(id\_lowiska) join gatunki using(id\_gatunku)*

*where id\_gatunku is not null and (extract(year from czas), id\_gatunku, dlugosc) in*

*(select extract(year from czas) ,id\_gatunku, max(dlugosc) from rejestry*

*where id\_gatunku is not null*

*group by extract(year from czas), id\_gatunku)*

--Zad 3

**W oparciu o dane zgromadzone w tabelach Rejestry, Gatunki, Lowiska i Wedkarze wyświetl informację (jak na Rys. 3) dotyczące połowu najcięższej ryby mającego miejsce w majową sobotę lub niedzielę (dowolnego roku).**

*Select waga,gatunki.nazwa as "gatunek", lowiska.nazwa as "lowisko",nazwisko,imie,trunc(czas)*

*as "dzien" from rejestry join wedkarze using(id\_wedkarza)join lowiska using(id\_lowiska)join*

*gatunki using(id\_gatunku) where trim(to\_char(czas,'day'))in('sobota','niedziela')*

*and trim(to\_char(czas,'month'))like 'maj' and waga=(Select max(waga) from rejestry where trim(to\_char(czas,'day'))in('sobota','niedziela')*

*and trim(to\_char(czas,'month'))like 'maj');*

--Zad 4

**W oparciu o dane zgromadzone w tabeli Studenci wyświetl najgorszych studentów (wg średniej oceny) studiujących w ramach danego trybu i kierunku studiów. W zestawieniu uwzględnij tylko studentów drugiego roku studiów pierwszego stopnia. Dane wyświetl uporządkowane w trybie niemalejącym wg średniej oceny (Rys. 4).**

*Select \* from studenci where stopien=1 and rok =2 and (kierunek,tryb,srednia)in(Select kierunek,tryb,max(srednia) from studenci where stopien=1 and rok=2 group by kierunek,tryb);*

*Select kierunek,tryb,min(srednia) from studenci where stopien=1 and rok=2 group by kierunek,tryb;*

--Zad 5

**W oparciu o dane zgromadzone w tabeli Studenci wyświetl najlepsze studentki (panie) i najlepszych studentów (panów) studiujących na danym kierunku studiów (o ile to możliwe to przynajmniej po 1-ym studencie i 1-ej studentce na każdym kierunku). Dane wyświetl uporządkowane alfabetycznie wg nazwy kierunku, średniej (nierosnąco) i trybu studiów (alfabetycznie; patrz Rys. 5).**

*Select kierunek, srednia, nazwisko, imiona, nr\_indeksu, stopien, rok, tryb from studenci where srednia is not null order by srednia desc group by kierunek;*

--Zad 6

**W oparciu o dane zgromadzone w tabelach Pojazdy i Kierowcy wyświetl dane kierowców, którzy posiadają największą łączną liczbę samochodów osobowych i motocykli. Dane wyświetl uporządkowane wg identyfikatorów kierowców (patrz Rys. 6)**

**Prezentowaną w ramach zadania nr 6 listę rozszerz o informację z zakresu ile samochodów osobowych i ile motocykli posiada dany kierowca (patrz Rys. 7).**

*Select id\_kierowcy, nazwisko, imie, adres, count(nr\_rejestr) from pojazdy join kierowcy on(wlasciciel=id\_kierowcy) where typ in('samochod osobowy','motocykl') group by id\_kierowcy,imie,nazwisko,adres having count(nr\_rejestr) =(Select max(count(nr\_rejestr)) from pojazdy where typ in('samochod osobowy','motocykl') group by wlasciciel);*

*Select max(count(nr\_rejestr)) from pojazdy where typ in('samochod osobowy','motocykl') group by wlasciciel;*

--Zad 8

**W oparciu o dane zgromadzone w tabelach Rejestry, Gatunki, Wedkarze i Lowiska wyświetl listę zawierającą szczegóły dotyczące ostatnich odnotowanych połowów ryb danego gatunku z uwzględnieniem pozycji brak połowu (ostatni nieudany połów; patrz Rys. 8).**

*Select id\_gatunku,czas,gatunki.nazwa,nazwisko,lowiska.nazwa,systimestamp-czas from rejestry left join gatunki using(id\_gatunku) join lowiska using(id\_lowiska)join wedkarze using(id\_wedkarza) where (nvl(id\_gatunku,0),czas) in(Select nvl(id\_gatunku,0), max(czas) from rejestry group by id\_gatunku);*

*Select id\_gatunku, max(czas) from rejestry group by id\_gatunku;*

--Zad 9

**Zmodyfikuj rozwiązanie zadania 8 tak aby lista zawierała wszystkie gatunki jakie znajdują się w tabeli Gatunki (również te bez żadnego zdarzenia połowu) oraz pozycję brak połowu (patrz Rys. 9).**

--Zad 10

**W oparciu o dane zgromadzone w tabelach Pracownicy i Dzialy wyświetl listę prezentującą najdłużej i najkrócej pracujących pracowników w ramach danego działu wśród aktualnych pracowników. W zestawieniu należy uwzględnić tylko aktualnych pracowników firmy. Dane wyświetl uporządkowane wg identyfikatorów działów i dat zatrudnienia (patrz Rys. 10).**

*Select pr.id\_dzialu,nazwa, nazwisko, data\_zatr,data\_zwol from pracownicy pr join dzialy dz on(pr.id\_dzialu=dz.id\_dzialu) where*

*(data\_zwol is null or data\_zwol>= sysdate) and*

*data\_zatr=(Select min(data\_zatr)from pracownicy where (data\_zwol is null or data\_zwol >=sysdate) and id\_dzialu=pr.id\_dzialu)*

*or data\_zatr=(Select max(data\_zatr)from pracownicy where data\_zwol is null or data\_zwol >=sysdate)*

*;*

*Select id\_dzialu,min(data\_zatr)from pracownicy where data\_zwol is null or data\_zwol >=sysdate group by id\_dzialu;*

--Zad 11

**W oparciu o dane zgromadzone w tabeli Pojazdy wyświetl listę zawierającą informację o najbardziej popularnych modelach (najliczniejszych) pojazdów w ramach poszczególnych typów pojazdów (Rys. 11). Zaproponuj rozwiązanie niezawierające klauzuli Having.**

select\* from

(

select typ, max (liczba)maxliczba from

(select typ, modell, count(nr\_rejestr)liczba from pojazdy group by typ, marka, modell)

group by typ)t1

join( select typ, modell, count(nr\_rejestr)liczba from pojazdy group by typ, marka, modell)t2

on (t1.typ=t2.typ and liczba=maxliczba);

--Zad 12

**Na podstawie danych zawartych w tabeli Studenci wyświetl nazwę ulicy/alei (łącznie bez rozróżniania) jaka najczęściej widnieje w adresach studentów (kolumna adres). Założenia: nazwa ulicy/alei zaczyna się od 5-ego znaku w adresie, nazwy miast są maksymalnie dwu wyrazowe gdzie wyrazy w nazwach dwuczłonowych oddzielone są spacją lub minusem (-) oraz numery domów mogą przyjmować jedną z form X, X-Y, X/Y, X-Y/Z (X, Y, Z – wartości od 1 do max. 999).**

--Zad 13

**W oparciu o dane zgromadzone w tabelach Rejestry, Gatunki, Lowiska i Wedkarze wyświetl zdarzenia połowów ryb, w ramach których złowiona ryba była dłuższa od średniej długości ryb tego gatunku w danym roku połowu (np. długość danego sandacza złowionego w 2018 roku należy porównać ze średnią obliczoną z długości wszystkich sandaczy złowionych w 2018 r.). Dane wyświetl uporządkowane kolejno wg id\_gatunku, roku połowu (chronologicznie) i długości ryb (nierosnąco; patrz Rys. 13).**

*select to\_char(czas,'YYYY-MM-DD HH24:MI')kiedy, gatunek, re.id\_gatunku, re.dlugosc, round(SREDNIA,2)SREDNIADL*

*from rejestry re join(*

*select extract(year from czas)ROK, id\_gatunku, ry.nazwa gatunek, avg(dlugosc)SREDNIA*

*from rejestry join gatunki ry using(id\_gatunku) where id\_gatunku is not null*

*group by extract(year from czas), id\_gatunku, nazwa) dd on (extract(year from re.czas)=dd.ROK and re.id\_gatunku= dd.id\_gatunku)*

*where re.dlugosc>=dd.srednia*

*order by 1 desc,4 desc;*

--Zad 14

**W oparciu o dane zgromadzone w tabelach Rejestry i Lowiska wyświetl listę najlepszych łowisk w ramach poszczególnych okręgów PZW pod względem łącznej wagi wszystkich złowionych na nich ryb. Dane wyświetl uporządkowane w trybie nierosnącym wg łącznej wagi złowionych ryb (patrz Rys. 14). Zaproponuj rozwiązanie niezawierające klauzuli Having.**

*Select \* from (*

*Select id\_okregu, max(laczna) maxwaga from (Select id\_okregu,id\_lowiska,nazwa,sum(waga) laczna from rejestry join lowiska using(id\_lowiska) where id\_okregu like 'PZW%' group by id\_okregu,id\_lowiska,nazwa) group by id\_okregu*

*) tabela join (Select id\_okregu,id\_lowiska,nazwa,sum(waga) laczna from rejestry join lowiska using(id\_lowiska) where id\_okregu like 'PZW%' group by id\_okregu,id\_lowiska,nazwa) p2 on(tabela.id\_okregu=p2.id\_okregu) where maxwaga=laczna;*

--Zad 15

**W oparciu o dane zgromadzone w tabeli Studenci utwórz zestawienie zawierające informację ilu studentów studiuje w ramach danego (patrz Rys. 15):**

**• trybu, stopnia, kierunku, roku,**

**• trybu, stopnia, kierunku**

**• trybu, stopnia,**

**• trybu.**

*Select tryb,stopien,kierunek, rok, count(nr\_indeksu) from studenci group by rollup(tryb,stopien,kierunek,rok);*

--Zad 16

**Składnię zapytania z zadania nr 15 rozszerz o dodatkową kolumnę o nazwie komentarz, w ramach której pojawią się następujące informacje (patrz Rys. 16): 1A / 1.5A**

**• Na danym roku w TSK – w przypadku informacji dotyczącej liczby studentów studiujących na danym roku w ramach danego trybu, stopnia i kierunku studiów,**

**• Na danym kierunku w TS – w przypadku informacji dotyczącej liczby studentów studiujących na danym kierunku w ramach danego tryby i stopnia studiów,**

**• W ramach danego stopnia w T – w przypadku informacji dotyczącej liczby studentów studiujących na danym stopniu w ramach danego trybu studiów,**

**• W danym trybie – w przypadku informacji dotyczącej liczby studentów studiujących w ramach danego trybu studiów,**

**• Ogolem studiuje – w przypadku informacji dotyczącej ogólnej liczby studentów.**

--Zad 17

**W oparciu o dane zgromadzone w tabelach Pojazdy i Kierowcy utwórz zestawienie zawierające informację o liczbie pojazdów przynależących do danego kierowcy w ramach:**

**• danego typu i marki,**

**• danego typu,**

**• danej marki, oraz dodatkowo zawierające następujące informacje o:**

**• liczbie pojazdów danego kierowcy,**

**• liczbie pojazdów danego typu,**

**• liczbie pojazdów danej marki,**

**• liczbie wszystkich pojazdów. W zestawieniu uwzględnij tylko podzbiory składające się z minimum 3 pojazdów (patrz Rys. 17).**

*Select decode(grouping\_id(typ,marka,id\_kierowcy),7,'podsumowania',6,'tyb-marka','inne') komentarz,*

*nazwisko,id\_kierowcy,typ,marka,count(\*) from pojazdy join kierowcy on(id\_kierowcy=wlasciciel) group by grouping sets((typ,marka),(typ),(marka),(id\_kierowcy,nazwisko),());*

--Zad 18

**Na podstawie danych zawartych w tabeli Rejestry utwórz zestawienie zawierające informacje na temat jakie gatunki i w jakiej liczbie były łowione na poszczególnych łowiskach. W zestawieniu uwzględnij podsumowania połowów w ramach poszczególnych łowisk oraz poszczególnych gatunków ryb. W składni zapytania zastosuj Grouping Sets, Decode i Grouping. W miejsca puste wstaw odpowiednie komentarze (Razem, Wszystkie lowiska itp.) . Zakres danych zgodnie z zawartością zaprezentowaną na Rysunku 18. Założenie: Nazwy łowisk mogą powtarzać się.**

--Zad 1

**Na podstawie danych zawartych w tabelach Rejestry, Wedkarze i Gatunki wyświetl informacje o wędkarzach (jak na Rys. 1), którzy dokonali połowu przynajmniej jednego sandacza w okresie od początku czerwca do końca września (zastosuj operator ANY).**

*select \* from wedkarze*

*where id\_wedkarza = any (*

*select id\_wedkarza from rejestry join gatunki using(id\_gatunku)*

*where nazwa like 'SANDACZ');*

--Zad 2

**Kontynuacja zadania nr 1. W wyniku uwzględnij tych wędkarzy, których wszystkie złowione w rozważanym okresie sandacze miały przynajmniej 60 cm. długości (Rys. 2) (zastosuj jeden z operatorów).**

*select \* from wedkarze where id\_wedkarza = any(*

*select id\_wedkarza from Rejestry join gatunki using (id\_gatunku)*

*where nazwa like upper ('sandacz') and extract(month from czas) between 6 and 9*

*minus*

*select id\_wedkarza from Rejestry join gatunki using (id\_gatunku)*

*where nazwa like upper ('sandacz') and extract(month from czas) between 6 and 9*

*and dlugosc<60);*

--Zad 3

**Na podstawie danych zawartych w tabelach Stanowiska i Pracownicy wyświetl listę stanowisk (jak na Rys. 3), w ramach których wszyscy aktualni pracownicy firmy otrzymują płacę (placa) zgodną z przyjętymi widełkami płacowymi ustalonymi dla tych stanowisk pracy (placa\_min, placa\_max) (zastosuj jeden z operatorów).**

*select stanowisko, placa\_min, placa\_max, staz\_min,dod\_funkcyjny*

*from stanowiska s\_zew where placa\_min<=all*

*(select placa from pracownicy where stanowisko=s\_zew.stanowisko*

*and(data\_zwol is not null or data\_zwol>=sysdate))and*

*placa\_max>=all*

*(select placa from pracownicy where stanowisko=s\_zew.stanowisko*

*and (data\_zwol is not null or data\_zwol >=sysdate));*

--Zad 4

**Na podstawie danych zawartych w tabelach Gatunki i Rejestry wyświetl gatunki ryb, w ramach połowu których przynajmniej raz złamano przepisy z zakresu wymiaru ochronnego (rejestry.dlugosc>= gatunki.wymiar). W wyniku uwzględnij tylko gatunki, które posiadają zdefiniowany wymiar ochronny (zastosuj jeden z operatorów).**

--Zad 4/5

**Kontynuacja zadania nr 4. Na podstawie danych zawartych w tabelach Gatunki, Rejestry i Wedakrze wyświetl listę wędkarzy oraz gatunki ryb, które zostały przez nich zabrane z łowisk pomimo naruszenia przepisów ochronnych z zakresu wymiaru ochronnego (patrz Rys. 5).**

*select \* from gatunki GA cross join wedkarze WE where exists(*

*select id\_gatunku from gatunki join rejestry using (id\_gatunku) where dlugosc<wymiar*

*and id\_gatunku=GA.id\_gatunku and id\_wedkarza=WE.id\_wedkarza);*

--Zad 6

**Na podstawie danych zawartych w tabelach Lowiska, Gatunki i Rejestry wyświetl informacje (jak na Rys. 6) o tym jakie gatunki ryb były złowione (przynajmniej 1 sztuka) na wodach zarządzanych przez dany okręg PZW (zastosuj EXISTS).**

*select distinct id\_okregu,ga.nazwa from lowiska lo cross join gatunki ga*

*where id\_okregu like 'PZW%'*

*and not Exists(select \* from lowiska join rejestry using (id\_lowiska)*

*join gatunki using (id\_gatunku)*

*where id\_okregu like 'PZW%' and gatunki.nazwa=ga.nazwa and id\_okregu=lo.id\_okregu);*

--Zad 7

**Dokonaj modyfikacji zadania nr 6 tak aby prezentowana lista (Rys. 7) zawierała informacje o gatunkach ryb złowionych na wodach danego zarządcy przez przynajmniej 3-ech różnych wędkarzy w ramach przynajmniej jednego (dowolnego) roku kalendarzowego.**

*select distinct id\_okregu, GA.nazwa from lowiska LO cross join gatunki GA*

*where id\_okregu like 'PZW%'*

*and Exists*

*(*

*select gatunki.nazwa, id\_okregu, extract(year from czas),*

*count(distinct id\_wedkarza) from lowiska join rejestry using (id\_lowiska)*

*join gatunki using(id\_gatunku)*

*where id\_okregu like 'PZW%' and gatunki.nazwa = GA.nazwa and id\_okregu = LO.id\_okregu*

*group by gatunki.nazwa, id\_okregu, extract(year from czas) having count(\*) >= 3*

*);*

--Zad 8

**Na podstawie danych zgromadzonych w tabelach Gatunki, Okregi, Rejestry i Lowiska wyświetl gatunki ryb, które na wodach danego okręgu PZW nigdy nie zostały złowione (nawet 1 raz; patrz Rys. 8).**

*Select \* from gatunki ga cross join okregi ok where id\_okregu like 'PZW%' and not exists(*

*Select \* from rejestry re join gatunki gat on(re.id\_gatunku=gat.id\_gatunku) join lowiska lo using(id\_lowiska) where*

*re.id\_gatunku=ga.id\_gatunku and id\_okregu=ok.id\_okregu*

*);*

--Zad 9

**Na podstawie danych przechowywanych w tabelach Wedkarze i Rejestry wyświetl listę wędkarzy (Rys. 9), którzy na przestrzeni wszystkich lat, w jakich odnotowywano połowy ryb, w każdym roku złowili przynajmniej jedną rybę.**

--Zad 10

**Na podstawie danych zawartych w tabelach Rejestry i Wedkarze wyświetl listę wędkarzy, którzy na każdym odwiedzonym łowisku przynajmniej raz nie złowili ryby (Rys. 10).**

*select \* from wedkarze t1 join (select id\_wedkarza from rejestry join wedkarze*

*using (id\_wedkarza) group by id\_wedkarza*

*having count (id\_gatunku) =0) t2 on t1.id\_wedkarza=t2.id\_wedkarza;*

--Zad 11

**Na podstawie danych zawartych w tabelach Rejestry, Lowiska i Gatunki wyświetl listę łowisk (patrz Rys. 11), w których każdego roku (w którym odnotowywano połowy na łowisku) złowiono ryby o łącznej wadze minimum 1 kg., w tym przynajmniej raz (min. w jedym roku) powyżej 10 kg. oraz na których w całym badanym okresie odnotowano połów przynajmniej jednego szczupaka. W składni rozwiązania zastosuj operatory ANY, ALL i EXISTS.**

*select \* from lowiska LO where 1<=all(*

*select sum(waga) from rejestry where id\_lowiska=LO.id\_lowiska*

*group by extract(year from czas)) and exists(*

*select sum(waga) from rejestry where id\_lowiska=LO.id\_lowiska*

*group by extract(year from czas)*

*having sum(waga)>10)*

*and exists(select \* from rejestry join gatunki using (id\_gatunku)*

*where id\_lowiska=LO.id\_lowiska and nazwa like upper('szczupak'));*

--Zad 12

**Na podstawie danych zawartych w tabelach wchodzących w skład bazy Wędkarstwo wyświetl wędkarzy (Rys. 12), którzy nigdy nie złamali przepisów w zakresie: a) wymiaru ochronnego (rejestry.dlugosc>=gatunki.wymiar), b) okresu ochronnego (niektóre gatunki posiadają okres ochronny, w trakcie którego nie można ich łowić; kolumny w tabeli ryba: DPO, MPO, DKO i MKO; uwaga – dla pewnych gatunków okres ochronny kończy się w innym roku kalendarzowym niż rozpoczyna się np. sum 1.11 – 30.06).**

--Zad 13

**Na podstawie danych zgromadzonych w tabeli Studenci wyświetl w ramach jednego wiersza dane personalne (imię i nazwisko) studentek studiujących w ramach danej grupy dziekańskiej (z uwzględnieniem podziału na stopień i rok) informatykę w trybie stacjonarnym. Dane personalne studentek mają być oddzielone od siebie przecinkiem (patrz Rys. 13).**

--Zad 14

**Na podstawie danych zawartych w tabelach Rejestry, Gatunki i Wedkarze wyświetl informacje (jak na Rys. 14) dotyczącą wędkarzy, którzy złowili poszczególne egzemplarze ryb danego gatunek. Uzupełnieniem prezentowanych danych jest liczba złowionych ryb w ramach danego gatunku (kolumna sztuk).**

*Select count(\*),listagg(nazwisko,',')within group(order by nazwisko)*

*from rejestry join gatunki using(id\_gatunku) join wedkarze using(id\_wedkarza) group by nazwa;*

--Zad 15

**Zmodyfikuj rozwiązanie zadania nr 14, tak aby poza nazwiskami wędkarzy pojawiały się pierwsze litery ich imion oraz w nawiasach liczby określające ile sztuk danego gatunku złowił dany wędkarz (patrz Rys. 15). Pamiętaj, że nazwiska wędkarzy mogą powtarzać się.**

*Select count(\*),listagg(nazwisko,',')within group(order by nazwisko)*

*from rejestry join gatunki using(id\_gatunku) join wedkarze using(id\_wedkarza) group by nazwa;*

*Select nazwisko,sum(liczba),sum(waga),listagg(nazwisko||' '||litera||'('||liczba||'szt'||waga||'kg)',',')*

*within group(order by liczba desc,waga desc,nazwisko) as komentarz from(*

*Select nazwa,id\_wedkarza,nazwisko,substr(imie,1,1)||'.' litera,count(\*) liczba,sum(waga) waga*

*from rejestry join gatunki using(id\_gatunku) join wedkarze using(id\_wedkarza) group by nazwa,id\_wedkarza,nazwisko);*

--Zad 16

Na podstawie danych zgromadzonych w tabelach Rejestry, Lowiska i Wedkarze wyświetl zestawienie **(jak na Rys. 16), w ramach którego zostaną wyszczególnione poszczególne łowiska zarządzane przez danego zarządcę z informacjami o największych łącznych wagach ryb złowionych na nich przez jednego wędkarza (w formie największa łączna waga nazwisko wędkarza). W zestawieniu uwzględnij tzw. dzikie łowiska stosując frazę Brak w ramach kolumny miejsce id\_okregu.**

--Zad 1

**W oparciu o dane zawarte w tabeli Rejestry wykorzystując w składni zapytania operator UNION (bez CASE) wyświetl listę złowionych ryb o identyfikatorach równych 10 (id\_gatunku=10) podając daty poszczególnych połowów z dokładnością co do minuty oraz długości poszczególnych ryb wraz z komentarzem informującym czy długość była poniżej, równa czy powyżej średniej długości dla tego gatunku przyjętej na poziomie 58 cm. (patrz Rys. 1).**

*Select to\_char(czas,'yyy-mm-dd hh24:mi'),dlugosc,'ponizej sredniej'*

*from rejestry where id\_gatunku=10 and dlugosc<58 UNION*

*Select to\_char(czas,'yyy-mm-dd hh24:mi'),dlugosc,'rowna sredniej'*

*from rejestry where id\_gatunku=10 and dlugosc=58 UNION*

*Select to\_char(czas,'yyy-mm-dd hh24:mi'),dlugosc,'powyzej sredniej'*

*from rejestry where id\_gatunku=10 and dlugosc>58;*

--Zad 2

**W oparciu o dane zawarte w tabelach Rejestry i Gatunki wykorzystując w składni zapytania wyrażenie CASE (i nie korzystając z operatorów typu UNION) wyświetl listę złowionych sandaczy (nazwa gatunku ryby) podając daty poszczególnych połowów z dokładnością co do minuty oraz długości poszczególnych ryb wraz z komentarzem informującym czy długość była poniżej, równa czy powyżej rzeczywistej średniej długości złowionych ryb tego gatunku (patrz Rys. 2). Zaproponuj dwa rozwiązania:**

**• rozwiązanie z jednym podzapytaniem zwracającym wartość średniej długości (w ramach złączenia),**

**• rozwiązanie z większą liczbą podzapytań (w ramach klauzuli SELECT).**

*select to\_char(czas,'YYYY-MM-DD HH24:MI'),dlugosc,*

*case*

*when dlugosc > (select avg(nvl(dlugosc,0)) from rejestry join gatunki using(id\_gatunku) where nazwa='SANDACZ')then 'powyzej'*

*when dlugosc < (select avg(nvl(dlugosc,0)) from rejestry join gatunki using(id\_gatunku) where nazwa='SANDACZ')then 'ponizej'*

*else 'rowna' end komentarz*

*from rejestry join gatunki using(id\_gatunku) where nazwa='SANDACZ';*

*select avg(nvl(dlugosc,0)) from rejestry join gatunki using(id\_gatunku) where nazwa='SANDACZ';*

--Zad 3

**Wykorzystując w składni zapytania operator MINUS na podstawie danych zawartych w tabelach Gatunki i Rejestry podaj gatunki ryb, których połów nie został odnotowany (w tabeli Rejestry) więcej niż 5 razy (patrz Rys. 3 A). Następnie zmodyfikuj składnię zapytania tak aby w wyniku prezentowana była również informacja o liczbie złowionych ryb z danego gatunku (patrz Rys. 3 B)**

*select id\_gatunku,nazwa from gatunki ga left join(select id\_gatunku from rejestry) re*

*on(ga.id\_gatunku=re,id\_gatunku) group by ga.id\_gatunku,ga,nazwa*

*Minus*

*select id\_gatunku,nazwa from gatunki ga join(select id\_gatunku from rejestry) re*

*on(ga.id\_gatunku=re,id\_gatunku)group by ga.id\_gatunku,ga.nazwa having count(\*) >5*

--Zad 4

**Wykorzystując w składni zapytania operator INTERSECT na podstawie danych zawartych w tabelach Kierowcy i Pojazdy wyświetl listę kierowców, którzy posiadają dokładnie 1 motocykl, przynajmniej 1 samochód osobowy i przynajmniej 2 samochody ciężarowe (Rys. 4A). Następnie dokonaj modyfikacji składni zapytania tak aby w wyniku pojawiła się dodatkowa kolumna informująca o łącznej liczbie pojazdów posiadanych przez danego kierowcę (Rys. 4B).**

***select id\_kierowcy,nazwisko,imie,count(\*) as "Liczba pojazdow" from pojazdy p join***

***select id\_kierowcy,nazwisko,imie from kierowcy join pojazdy on(id\_kierowcy=wlasciciel)***

***where lower(typ) like'motocykl' group by id\_kierowcy,naziwsko, imie having count(\*)=1***

***intersect***

***select id\_kierowcy,nazwisko,imie from kierowcy join pojazdy on(id\_kierowcy=wlasciciel)***

***where lower(typ) like'samochod osobowy' group by id\_kierowcy,naziwsko, imie having count(\*)>=1***

***intersect***

***(select id\_kierowcy,nazwisko,imie from kierowcy join pojazdy on(id\_kierowcy=wlasciciel)***

***where lower(typ) like'samochod ciezarowy' group by id\_kierowcy,nazwisko, imie having count(\*)>=2 )k***

***on(k.id\_kierowcy=p.wlasciciel)group by id\_kierowcy,nazwisko,imie;***

--Zad 5

**Na podstawie danych zgromadzonych w tabelach Rejestry, Gatunki, Lowiska i Wedkarze wyświetl listę gatunków ryb (jak na Rys. 5 A), których połów odnotowano na łowiskach o nazwach Poraj i Pilica (przynajmniej po jednej rybie danego gatunku na każdym z nich) i które nigdy nie złowił wędkarz/e o nazwisku Andrysiak (na dowolnym łowisku). Następnie zmodyfikuj składnię zapytania tak aby w wyniku prezentowana była również informacja o liczbie złowionych ryb z danego gatunku (patrz Rys. 5 B).**

*Select t.id\_gatunku,t.nazwa,count(re.id\_gatunku) from Rejestry re join*

*(*

*Select id\_gatunku,gatunki.nazwa from rejestry join gatunki using(id\_gatunku) join lowiska using(id\_lowiska)*

*where lowiska.nazwa like 'Poraj' INTERSECT*

*Select id\_gatunku,gatunki.nazwa from rejestry join gatunki using(id\_gatunku) join lowiska using(id\_lowiska)*

*where lowiska.nazwa like 'Pilica' MINUS*

*Select id\_gatunku,gatunki.nazwa nazwa from rejestry join gatunki using(id\_gatunku) join wedkarze using(id\_wedkarza)*

*where nazwisko like 'Andrysiak')t on(re.id\_gatunku=t.id\_gatunku) join lowiska lo on(re.id\_lowiska = lo.id\_lowiska)*

*where lo.nazwa in('Poraj','Pilica') group by t.id\_gatunku,t.nazwa;*

--Zad 6

**Na podstawie danych zawartych w tabelach Licencje, Wedkarze, Rejestry i Lowiska wyświetl listę wędkarzy, którzy w danym roku kalendarzowym (z wyłączeniem aktualnego) pomimo opłacenia rocznej składki wędkarskiej (dowolnego rodzaju) na rzecz danego okręgu PZW (id\_okregu – nazwy rozpoczynają się od frazy PZW) nie odnotowali przynajmniej jednego połowu ryb (pobytu) na wodach zarządzanych przez ten okręg PZW (patrz Rys. 6 A). Następnie zmodyfikuj składnię zapytania tak aby zaprezentować wędkarzy, którzy wykorzystali opłaconą licencję, podając jednocześnie liczbę połowów (patrz Rys. 6 B).**

*select t1.rok, t1.id\_wedkarza, t1.nazwisko, t1.id\_okregu, count(\*) from (*

*select rok, id\_wedkarza,nazwisko, id\_okregu from licencje join wedkarze using(id\_wedkarza) where od\_dnia like '01-01' and do\_dnia like '31-12' and id\_okregu like 'PZW%' and rok<> extract(YEAR from sysdate)*

*intersect*

*select extract(YEAR from czas), id\_wedkarza,nazwisko,id\_okregu from rejestry join lowiska using (id\_lowiska) join wedkarze using(id\_wedkarza) ) t1 join rejestry re on(t1.id\_wedkarza = re.id\_wedkarza and t1.rok=extract(year from re.czas)) join lowiska lo on(re.id\_lowiska = lo.id\_lowiska) where t1.id\_okregu = lo.id\_okregu*

*group by (t1.rok, t1.id\_wedkarza, t1.nazwisko, t1.id\_okregu);*

--Zad 7

**Na podstawie danych zawartych w tabelach Rejestry, Lowiska i Wedkarze wyświetl listę wędkarzy (patrz Rys. 7), którzy przynajmniej 5 razy łowili (w całym analizowanym okresie) na wodach zarządzanych przez PZW Czestochowa i przynajmniej 3 na wodach niezarządzanych przez PZW (łowiska prywatne oraz niezagospodarowane tzw. dzikie) oraz w 2018 roku odnotowali tylko nieudane połowy (wszystkie połowy bez złowionej ryby).**

*select ID\_WEDKARZA, nazwisko, imie from Rejestry join wedkarze using(ID\_wedkarza) join lowiska using (ID\_lowiska)*

*where id\_okregu like 'PZW Czestochowa' group by ID\_WEDKARZA, nazwisko, imie having count(\*)>=5*

*intersect*

*select ID\_WEDKARZA, nazwisko, imie from Rejestry join wedkarze using(ID\_wedkarza) join lowiska using (ID\_lowiska)*

*where id\_okregu not like 'PZW%' or Id\_okregu is null group by ID\_WEDKARZA, nazwisko, imie having count(\*)>=3*

*intersect*

*select ID\_WEDKARZA, nazwisko, imie from Rejestry join wedkarze using(ID\_wedkarza) join lowiska using (ID\_lowiska)*

*where extract(year from czas)=2018*

*minus*

*select ID\_WEDKARZA, nazwisko, imie from Rejestry join wedkarze using(ID\_wedkarza) join lowiska using (ID\_lowiska)*

*where extract(year from czas)=2018 and id\_gatunku is not null;*

--Zad 8

**Na podstawie danych zawartych w tabeli Rejestry wyświetl informacje o największych łącznych wagach ryb złowionych przez jednego wędkarza w danym roku kalendarzowym oraz ile w analogicznych okresach wynosiły średnie roczne połowy (w kg) przypadające na jednego wędkarza (patrz Rys. 8).**

--Zad 9

**Na podstawie danych zawartych w tabelach Rejestry i Wedkarze wyświetl informacje o największych i najmniejszych łącznych wagach ryb złowionych przez jednego wędkarza w danym roku kalendarzowym. Wyświetl nazwiska i imiona poszczególnych wędkarzy (patrz Rys. 9). W ramach analizy danych należy uwzględniać wędkarzy, którzy w danym roku dokonali przynajmniej jednej próby połowu.**

--Zad 10

**Na podstawie danych zawartych w tabelach Licencje i Oplaty wyświetl informacje o łącznych kwotach jakie poszczególne okręgi PZW (tylko PZW) uzyskały w danych latach z tytułu opłat wniesionych przez wędkarzy za podstawowe i dodatkowe licencje roczne oraz informacje o łącznych kwotach zebranych przez poszczególnych zarządców wód (wszystkich) w danych latach z tytułu opłat okresowych. W ramach informacji dodatkowych podaj liczbę licencji (rocznych / liczbę dni, na które wykupiono licencje okresowe) oraz stosowny komentarz do prezentowanych informacji (patrz Rys. 10).**

*Select \* from Licencje;*

*Select \* from Oplaty;*

*Select rok,id\_okregu,sum(roczna\_oplata\_pod),count(id\_licencji),*

*'roczna podstawowa' from*

*Licencje join Oplaty using(rok,id\_okregu) where rodzaj like 'podstawowa' group by rok,id\_okregu*

*UNION*

*Select rok,id\_okregu,sum(roczna\_oplata\_dod),count(id\_licencji),*

*'dzienna dodatkowa' from*

*Licencje join Oplaty using(rok,id\_okregu) where rodzaj like 'dodatkowa' and(od\_dnia like '01-01' and do\_dnia like '31-12')group by rok,id\_okregu*

*UNION*

*Select rok,id\_okregu,*

*sum((to\_date(do\_dnia || '-' || rok, 'dd-mm-yyyy')-to\_date(od\_dnia || '-' || rok, 'dd-mm-yyyy')+1)\*dzienna\_oplata) koszt,*

*sum((to\_date(do\_dnia || '-' || rok, 'dd-mm-yyyy')-to\_date(od\_dnia || '-' || rok, 'dd-mm-yyyy')+1)) liczba\_licencji*

*,'roczna dodatkowa' from*

*Licencje join Oplaty using(rok,id\_okregu) where rodzaj like 'dodatkowa' and not(od\_dnia like '01-01' and do\_dnia like '31-12')group by rok,id\_okregu order by 1,2;*

--Zad 11

**Na podstawie danych przechowywanych w tabelach Pojazdy wyświetl informację w jakim dniu tygodnia (słownie) zostało wyprodukowanych najmniej i najwięcej pojazdów jednej marki w ramach danego typu pojazdu (patrz Rys. 11).**

*select typ||'najwiecej jednej marki wyprodukowano w '||marka,to\_char(data\_prod,'day')||' ('||marka||' sztuk '||count(nr\_rejestr)|| ')' informacja from pojazdy group by typ,marka,to\_char(data\_prod,'day') having (typ,count(nr\_rejestr))in*

*(*

*select typ,max(liczba) from(*

*select typ,count(nr\_rejestr) liczba from pojazdy group by typ,marka,to\_char(data\_prod,'day')) group by typ) or*

*(typ,count(nr\_rejestr))in*

*(*

*select typ,min(liczba) from(*

*select typ,count(nr\_rejestr) liczba from pojazdy group by typ,marka,to\_char(data\_prod,'day')) group by typ);*

--Zad 12

**Na podstawie danych zawartych w tabelach Rejestry, Lowiska i Wedkarze wyświetl informacje dotyczącą poszczególnych wędkarzy w zakresie odbytych przez nich połowów na wodach gospodarowanych kolejno przez: PZW Czestochowa, PZW Katowice, PZW Opole oraz łowiskach niezarządzanych przez okręgi PZW (prywatne + tzw. „dzikie łowiska”). W zestawieniu uwzględnij wszystkich wędkarzy (nawet tych bez żadnego połowu). Forma prezentacji danych jak na poniższym Rysunku 12.**

*select id\_wedkarza , nazwisko ,*

*case*

*(select count(\*) from rejestry join lowiska using (id\_lowiska) where id\_okregu like'PZW Czestochowa' and*

*id\_wedkarza = wr.id\_wedkarza)*

*when 0 then 'nie'*

*else 'Tak' end czestochowa,*

*(select count(\*) from rejestry join lowiska using (id\_lowiska) where id\_okregu like'PZW Czestochowa' and*

*id\_wedkarza = wr.id\_wedkarza) czestochowa\_liczba,*

*case*

*(select count(\*) from rejestry join lowiska using (id\_lowiska) where id\_okregu like'PZW Opole' and*

*id\_wedkarza = wr.id\_wedkarza)*

*when 0 then 'nie'*

*else 'Tak' end Opole,*

*(select count(\*) from rejestry join lowiska using (id\_lowiska) where id\_okregu like'PZW Opole' and*

*id\_wedkarza = wr.id\_wedkarza) opole\_liczba,*

*case*

*(select count(\*) from rejestry join lowiska using (id\_lowiska) where (id\_okregu not like'PZW %'*

*and id\_okregu is null )and id\_wedkarza = wr.id\_wedkarza)*

*when 0 then 'nie'*

*else 'Tak' end pozostale,*

*(select count(\*) from rejestry join lowiska using (id\_lowiska) where (id\_okregu not like'PZW %'*

*and id\_okregu is null ) and*

*id\_wedkarza = wr.id\_wedkarza) pozostala\_liczba*

*from wedkarze wr;*

*--Zad 13*

**Na podstawie danych zawartych w tabelach Gatunki i Rejestry wyświetl informacje „na wodach” zarządzanych przez ile okręgów PZW odnotowano w tabeli Rejestry połów danego gatunku ryby. Opcje do rozpatrzenia: 1 okregu, 2 okregow, 3 okregow, wielu okregow (powyżej 3) i niezlowiona (patrz Rys. 13).**

*Select ga.id\_gatunku,ga.nazwa,*

*case count(distinct id\_okregu)*

*when 0 then 'niezlowiona'*

*when 1 then '1 okregu'*

*when 2 then '2 okregi'*

*when 3 then '3 okregi'*

*else 'wiecej okregow'*

*end*

*from Rejestry re right join gatunki ga on(re.id\_gatunku=ga.id\_gatunku) left join lowiska lo on(re.id\_lowiska=lo.id\_lowiska)*

*group by ga.id\_gatunku,ga.nazwa;*

*--Zad 14*

**Na podstawie danych zawartych w tabeli Gatunki wyświetl porównanie wymiarów ochronnych poszczególnych gatunków (każdy z każdym; tylko gatunki posiadające okres ochronny) pod warunkiem, że różnica pomiędzy wymiarami jest nie większa niż 10 cm. Możliwe frazy w ramach kolumny komentarz to mniejszy od, rowny i większy od (patrz Rys. 14).**

*Select g1.nazwa,g1.wymiar,*

*case*

*when g1.wymiar>g2.wymiar then 'wiekszy od'*

*when g1.wymiar<g2.wymiar then 'mniejszy'*

*else 'rowny'*

*end komentarz,*

*g2.nazwa,g2.wymiar,abs(g1.wymiar-g2.wymiar) from gatunki g1*

*cross join gatunki g2 where g1.dpo is not null and g2.dpo is not null and abs(g1.wymiar-g2.wymiar)<=10 and g1.nazwa!=g2.nazwa;*

*--Zad 15*

**Na podstawie danych zawartych w tabelach Pracownicy i Dzialy dokonaj porównania płac (kolumna placa) wśród aktualnych pracowników poszczególnych działów (patrz Rys. 15). Możliwe komentarze w ramach kolumny informacja to: 1.5A**

• Zarabia najwiecej w swoim dziale,

• Zarabia najmniej w swoim dziale,

• Zarabia przynajmniej tyle co X oraz mniej od Y pozostalych pracownikow swojego dzialu, gdzie X i Y to wartości obliczane indywidualne dla każdego pracownika.

--Zad 1

**Wyświetl o jaką kwotę (wartość bezwzględna) odbiegają zarobki (pełna pensja) aktualnych pracowników (tabela Pracownicy) od średniej krajowej wynoszącej 4521.45 PLN. Uzyskane wartości zaokrąglij kolejno do 1-ego miejsca po przecinku stosując funkcje Round oraz do pełnych 100-ek stosując funkcję Trunc. Wyniki przedstaw od najlepiej do najgorzej zarabiających podając m. in. ich nazwisko, zajmowane stanowisko oraz pensję**

--Zad 2

**Wyświetl wartości płac (kolumna placa) poszczególnych pracowników pomniejszone o 18% podatek (płaca netto). W osobnej kolumnie wyświetl płacę netto zaokrągloną do setek (np. 2343.12 → 2300) a w ostatniej różnice w poszczególnych płacach wynikające z rodzaju zastosowanych funkcji zaokrąglających wartości płac „do setek” (funkcje Round i Trunc).**

*select nazwisko,stanowisko,(placa\*0.82)as PlacaNetto,trunc(placa\*0.82,-2) as PlacaZaokraglona,round((placa\*0.82)-trunc(placa\*0.82,-2),-2) as Roznica from pracownicy;*

--Zad 3

**Napisz zapytanie obliczające wartość następującego wyrażenia √(12.342+ 3 √77) oraz wartości jakie uzyska się po zastosowaniu wobec tego wyrażenia funkcji Round, Trunc, Ceil oraz Floor.**

*SELECT sqrt(power(12.34,2)+power(77,-3)),round(sqrt(power(12.34,2)+power(77,-3))), Trunc(sqrt(power(12.34,2)+power(77,-3))),ceil(sqrt(power(12.34,2)+power(77,-3))), floor(sqrt(power(12.34,2)+power(77,-3))) from dual*

--Zad 4

**Wyświetl kolejno przedział 101 lat i 11 miesięcy oraz przedział 25 dni 3 godziny 5 minut i 36.6 sekund oraz znacznik czasowy obejmujący łącznie oba te przedziały**

--Zad 5

**Wyświetl informacje o tym jaka data będzie za 321 dni i jaka data była 321 dni temu oraz w odrębnych kolumnach ile pełnych miesięcy i ile pełnych tygodni dzieli obie daty.**

*SELECT sysdate+321,sysdate-321, floor(months\_between(sysdate+321,sysdate-321)), round((sysdate+321)-(sysdate-321)/7) from dual*

--Zad 6

**Wyświetl datę i czas jaki był 117 dni, 8 godzin i 9 minut temu oraz jaki będzie za 19 lat i 10 miesięcy**

--Zad 7

**Wyświetl ile czasu upłynęło od 2015.02.15 22:04:19 do 2018.12.11 19:07. Wynik podaj w dwóch formach/kolumnach**

*select timestamp '2018-12-11 19:07:00' - timestamp '2015-02-15 22:04:19', (timestamp '2018-12-11 19:07:00' - timestamp '2015-02-15 22:04:19') year to month from dual*

--Zad 8

**W odrębnych kolumnach kolejno wyświetl aktualny rok, miesiąc, dzień oraz aktualną godzinę, minutę i sekundę**

select extract(year from sysdate),extract(month from sysdate), extract(day from sysdate)

,extract(hour from systimestamp), extract(minute from systimestamp), extract(second from systimestamp)from dual;

--Zad 9

**W oparciu o dane zawarte w tabeli Studenci wyświetl studentów, którzy dzisiaj obchodzą urodziny.**

--Zad 10

**Napisz zapytanie zwracające informację jaka data będzie za 42 miesiące oraz który to będzie tydzień w roku.**

Select add\_months(sysdate,42), to\_char(add\_months(sysdate,42),'ww'),

to\_char(add\_months(sysdate,42),'day')from dual;

--Zad 11

**Wyświetl datę ostatniego dnia aktualnego miesiąca oraz informację jaki to będzie dzień tygodnia.**

*SELECT Last\_day(sysdate), to\_char(Last\_day(sysdate), 'day') from dual*

--Zad 12

**Zaokrąglij aktualną datę kolejno do lat (funkcją Trunc), miesięcy (funkcją Round) i wieków.**

--Zad 13

**Wyświetl nazwiska i imiona studentów, których imię rozpoczyna się na 'M' uporządkowanych od najstarszego do najmłodszego z informacją ile aktualnie mają lat**

*Select nazwisko, imiona, trunc(months\_between(sysdate,data\_urodzenia)/12)as "wiek" from studenci where imiona like 'M%' order by 3 desc;*

--Zad 14

**Dokonaj konwersji aktualnego znacznika czasowego do daty (zastosuj funkcję Cast;)**

--Zad 15

**W ramach zapytania dokonaj połączenia dwóch łańcuchów '987' i '654' oraz wyświetl wynik różnicy otrzymanej liczby i wartości 123456 (zastosuj konwersję typów;**

--Zad 16

**Wyświetl aktualny wiek cyframi rzymskimi.**

*SELECT to\_char(to\_char(Last\_day(sysdate), 'CC'), 'RN') from dual*

--Zad 17

**Wyświetl aktualną datę w postaci część, jest dzisiaj poniedziałek, 14 styczeń 2019 roku.**

--Zad 18

**Wyświetl wiek, kwartał, miesiąc, tydzień miesiąca oraz dzień tygodnia daty podanej w formie '12- 09-1683'**

--Zad 19

**Wyświetl dane studentek (patrz Rys. 13; źródło danych tabela Studenci), które urodziły się w wakacyjny weekend (w lipcową lub sierpniową sobotę lub niedzielę) pod warunkiem, że dzień miesiąca nie jest dzielnikiem liczby 5 (np. 6.07.1999 ale 10.08.1999 już nie;**

*Select nazwisko, imiona, data\_urodzenia, to\_char(data\_urodzenia, 'day')*

*from studenci where imiona like '%a'*

*and trim(to\_char(data\_urodzenia,'day'))in('sobota','niedziela')*

*and trim(to\_char(data\_urodzenia,'month'))in('sierpien','lipiec')*

*and mod(extract(day from data\_urodzenia),5)!=0;*

--Zad 20

**Z tabeli Pracownicy wyświetl aktualnie pracujących pracowników, porządkując ich według stażu pracy podanego w pełnych latach i miesiącach pracy oraz kolejności alfabetycznej nazwisk**

--Zad 21

**Wyświetl listę wynagrodzeń pracowników za miesiąc styczeń 2010, którzy byli pracownikami przez cały ten okres**

Select nazwisko,stanowisko,data\_zatr,data\_zwol,placa+dod\_staz\*0.01+nvl(dod\_funkcyjny,0)-nvl(koszt\_ubezpieczenia,0)

from pracownicy where data\_zatr<=to\_date('2010/01/01','yyyy/mm/dd') and (data\_zwol>=to\_date('2010/01/31','yyyy/mm/dd')

or data\_zwol is null)order by 5 desc;

--Zad 22

**W oparciu o dane zawarte w tabeli Studenci wyświetl listę studentów, którzy urodzili się w pierwszą sobotę danego miesiąca w okresie od 13.02.1995 do 28.11.1998. W rozwiązaniach poniższych zadań 23-26 należy wykorzystać wyrażenia regularne.**

*Select \* from studenci where data\_urodzenia between to\_date('12-02-1995','DD-MM-YYYY') and to\_date('28-11-1998','DD-MM-YYYY')*

--Zad 23

**Wyświetl personalia studentów (kolumny imiona i nazwisko z tabeli Studenci), których imię składa się dokładnie z 5 liter a nazwisko rozpoczyna od frazy Ko a kończy frazą ski**

Select imiona, nazwisko from studenci where regexp\_count(imiona,'[[:alpha:]]')=5 and

regexp\_like(nazwisko,'^(Ko)[[:alpha:]]{0,}(ski)$');

--Zad 24

**W oparciu o dane zgromadzone w tabeli Studenci za pośrednictwem wyrażeń regularnych dokonaj prezentacji zawartości kolumny adres w „zaszyfrowanej” formie, w której to każdy wyraz składający się z przynajmniej 3 liter będzie zastąpiony frazą Alpha a każda cyfra/liczba będzie zastąpiona frazą Digit**

*Select Adres, regexp\_replace(regexp\_replace(adres,'[[:alpha:]]{1,}','alpha'),'[[:digit:]]{3,}','Digit') from studenci*

--Zad 25

**W oparciu o dane zgromadzone w tabeli Studenci wyświetl adresy (kolumna adres) wskazujące na aleje (al. na początku adresu), których nazwa rozpoczyna się od litery B, P, T lub W, a numer domu/budynku to dokładnie 68**

*Select adres from studenci where regexp\_like(adres,'^(al.) (B|P|T|W).\* 68');*

--Zad 26

**W oparciu o dane zgromadzone w tabeli Studenci wyświetl adresy (kolumna adres), w ramach których zarówno nazwy ulic lub alei jak i nazwa miejscowości składają się dokładnie z dwóch przynajmniej 5-cio literowych wyrazów**

--Zad 27

**W oparciu o dane zgromadzone w tabeli Pojazdy wyświetl raport dotyczący pojazdów marki Ford modelu Focus w zakresie posiadanych przez nich kolorów. Raport ma zawierać kolumnę licznik zawierającą numer porządkowy nieuporządkowanej listy oraz kolumny Pozycja w rankingu i Identyfikator koloru utworzone w wyniku zastosowania funkcji szeregujących**

*Select kolor, rownum, rank()over(order by kolor), dense\_rank()over(order by kolor) from pojazdy*

--Zad 28

**W oparciu o dane zgromadzone w tabeli Rejestry wyświetl raport prezentujący wagi poszczególnych złowionych ryb danego gatunku w danym roku kalendarzowym na tle największej wagi (Min waga gatunku) i najmniejszej wagi (Max waga gatunku) odnotowanych wśród złowionych ryb danego gatunku na przestrzeni wszystkich lat (Rys. 21; wykorzystaj funkcje analityczne).**

*Select extract(year from czas), id\_gatunku, waga,*

*min(waga) keep(dense\_rank first order by waga)over(partition by id\_gatunku) "min",*

*max(waga) keep(dense\_rank last order by waga)over(partition by id\_gatunku) "max"*

*from rejestry where id\_gatunku is not null order by 1;*