

Zadanie 1

1. Wyświetl **średnią pensję** wszystkich pracowników.
2. Wyświetl **średnią pensję oraz liczbę pracowników** dla każdego stanowiska (JOB_POSITION_ID).
3. Wyświetl **średni wiek** dla każdego departamentu (DEPARTMENT_ID), do którego należy pracownik.

Podpowiedź:

- Użyj funkcji agregujących (AVG, COUNT).
- W zadaniu 2 potrzebne będzie GROUP BY.
- W zadaniu 3 można wykorzystać funkcję YEAR() do obliczenia wieku na podstawie daty urodzenia.

```
mysql> SELECT AVG(pensja) FROM EMPLOYEE;
+-----+
| AVG(pensja) |
+-----+
| 5041.666667 |
+-----+
1 row in set (0.06 sec)
```

```
mysql> SELECT AVG(pensja), COUNT(EMPLOYEE_NAME) FROM EMPLOYEE GROUP BY JOB_POSITION_ID;
+-----+-----+
| AVG(pensja) | COUNT(EMPLOYEE_NAME) |
+-----+-----+
| 5580.000000 | 5 |
| 4925.000000 | 4 |
| 5933.333333 | 3 |
| 4766.666667 | 3 |
| 4900.000000 | 3 |
| 4683.333333 | 3 |
| 4616.666667 | 3 |
| 4766.666667 | 3 |
| 4883.333333 | 3 |
+-----+-----+
9 rows in set (0.00 sec)
```

```
mysql> SELECT AVG(YEAR(CURDATE()) - YEAR(BIRTH_DATE)) AS wiek, DEPARTMENT_ID FROM EMPLOYEE GROUP BY DEPARTMENT_ID;
+-----+-----+
| wiek | DEPARTMENT_ID |
+-----+-----+
| 40.2500 | 1 |
| 37.0000 | 2 |
| 41.6667 | 3 |
| 32.0000 | 4 |
| 33.6667 | 6 |
| 35.0000 | 7 |
| 33.6667 | 8 |
| 37.3333 | 5 |
| 37.6667 | 9 |
| 32.0000 | 10 |
+-----+-----+
10 rows in set (0.00 sec)
```

Zadanie 2

1. Dla każdego **departamentu (DEPARTMENT_ID)** wyświetl **łączną pensję** jego pracowników.
Dane posortuj **malejąco**.
2. Dla każdego **stanowiska pracy** pracownika wyświetl **średnią pensję**, jeśli liczba pracowników na danym stanowisku jest ≥ 3 oraz średnia jest > 3000 PLN.
3. Wyświetl, **ile jest różnych pensji** dla każdego departamentu, jeśli minimalna liczba pracowników departamentu jest > 4 .
Dane posortuj **rosnąco** według ilości różnych pensji.

```
mysql> SELECT SUM(pensja), DEPARTMENT_ID FROM EMPLOYEE GROUP BY DEPARTMENT_ID;
+-----+-----+
| SUM(pensja) | DEPARTMENT_ID |
+-----+-----+
| 24100.00 | 1 |
| 14900.00 | 2 |
| 18500.00 | 3 |
| 12400.00 | 4 |
| 14450.00 | 6 |
| 14250.00 | 7 |
| 13550.00 | 8 |
| 15100.00 | 5 |
| 15300.00 | 9 |
| 8700.00 | 10 |
+-----+-----+
10 rows in set (0.00 sec)
```

```
mysql> SELECT AVG(pensja), COUNT(*) FROM EMPLOYEE GROUP BY DEPARTMENT_ID HAVING COUNT(*) >= 3 AND AVG(pensja) > 3000;
+-----+-----+
| AVG(pensja) | COUNT(*) |
+-----+-----+
| 6025.000000 | 4 |
| 4966.666667 | 3 |
| 8166.666667 | 3 |
| 4133.333333 | 3 |
| 4816.666667 | 3 |
| 4750.000000 | 3 |
| 4516.666667 | 3 |
| 5033.333333 | 3 |
| 5100.000000 | 3 |
+-----+-----+
9 rows in set (0.00 sec)
```

```
mysql> SELECT COUNT(DISTINCT(pensja)) FROM EMPLOYEE GROUP BY DEPARTMENT_ID HAVING COUNT(*) > 4 ORDER BY COUNT(DISTINCT(pensja)) ASC;
Empty set (0.00 sec)
```

Zadanie 3

1. Skopiuj tabele DEPARTMENT i JOB_POSITION z bazy company_2025 do swojej bazy danych.
2. Wyświetl dla każdej **nazwy departamentu** liczbę jego pracowników.
3. Wyświetl dla każdej **nazwy departamentu** imiona oraz nazwiska pracowników, którzy w nim pracują (łącznie).

```
mysql> CREATE TABLE DEPARTMENT AS SELECT * FROM company_2025.DEPARTMENT;
Query OK, 10 rows affected (0.23 sec)
Records: 10  Duplicates: 0  Warnings: 0
```

```
mysql> CREATE TABLE JOB_POSITION AS SELECT * FROM company_2025.JOB_POSITION;
Query OK, 10 rows affected (0.22 sec)
Records: 10  Duplicates: 0  Warnings: 0
```

```
mysql> SELECT DEPARTMENT.DEPARTMENT_NAME, COUNT(EMPLOYEE.EMPLOYEE_NAME) FROM DEPARTMENT LEFT JOIN EMPLOYEE ON DEPARTMENT.DEPARTMENT_ID = EMPLOYEE.DEPARTMENT_ID GROUP BY DEPARTMENT.DEPARTMENT_NAME;
+-----+-----+
| DEPARTMENT_NAME | COUNT(EMPLOYEE.EMPLOYEE_NAME) |
+-----+-----+
| Sprzedaz      |          4 |
| Magazyn       |          3 |
| Administracja |          3 |
| Ksiegowosc    |          3 |
| HR             |          3 |
| IT             |          3 |
| Logistyka     |          3 |
| Marketing     |          3 |
| Obsluga klienta |          3 |
| Produkcja     |          2 |
+-----+-----+
10 rows in set (0.00 sec)
```

```
mysql> SELECT d.DEPARTMENT_NAME, COUNT(e.EMPLOYEE_NAME) FROM DEPARTMENT d LEFT JOIN EMPLOYEE e ON d.DEPARTMENT_ID = e.DEPARTMENT_ID GROUP BY d.DEPARTMENT_NAME;
+-----+-----+
| DEPARTMENT_NAME | COUNT(e.EMPLOYEE_NAME) |
+-----+-----+
| Sprzedaz      |          4 |
| Magazyn       |          3 |
| Administracja |          3 |
| Ksiegowosc    |          3 |
| HR             |          3 |
| IT             |          3 |
| Logistyka     |          3 |
| Marketing     |          3 |
| Obsluga klienta |          3 |
| Produkcja     |          2 |
+-----+-----+
```

```
mysql> SELECT d.DEPARTMENT_NAME, GROUP_CONCAT(CONCAT(e.FIRST_NAME, ' ', e.LAST_NAME) SEPARATOR ', ') AS pracownik FROM DEPARTMENT d LEFT JOIN EMPLOYEE e ON d.DEPARTMENT_ID = e.DEPARTMENT_ID GROUP BY d.DEPARTMENT_NAME;
+-----+-----+
| DEPARTMENT_NAME | pracownik           |
+-----+-----+
| Administracja  | Wojciech Krol, Barbara Kowalczyk, Marek Nowicki
| HR              | Rafal Sikora, Tomasz Zielinski, Grzegorz Kubik
| IT              | Ewa Mazur, Joanna Wojcik, Monika Sikora
| Ksiegowosc     | Agnieszka Lis, Magdalena Sadowska, Agnieszka Lewandowska
| Logistyka      | Mariusz Zielony, Pawel Gajda, Jakub Wozniak
| Magazyn        | Katarzyna Lis, Janusz Nowak, Dorota Michalak
| Marketing      | Ewelina Wrona, Sylwia Baran, Aneta Kowalska
| Obsluga klienta | Marcin Krupa, Artur Bednarz, Olga Czajka
| Produkcja      | Patrycja Sowa, Elzbieta Krawczyk
| Sprzedaz       | Tomasz Kowal, Pawel Kaczor, Michal Zajac, Krzysztof Cybulski
+-----+-----+
10 rows in set (0.01 sec)
```

Zadanie 4

1. Wyświetl **nazwiska pracowników**, którzy urodzili się w **latach 90-tych XX wieku**, oraz **nazwy stanowisk**, na których są zatrudnieni (użyj NATURAL JOIN, jeśli to możliwe).
2. Wyświetl imiona i nazwiska **5 najmłodszych pracowników**, którzy pracują jako „**Magazynier**”.
3. Wypisz obok siebie **skrócone nazwy klientów**, których numer CLIENT_ID różni się o **5** (np. Kowalski - AutoMax, Nowak - Kaczmarek).

```
mysql> SELECT e.LAST_NAME, jp.POSITION_NAME FROM EMPLOYEE e JOIN JOB_POSITION jp ON e.JOB_POSITION_ID = jp.JOB_POSITION_ID WHERE YEAR(e.BIRTH_DATE) BETWEEN 1990 AND 1999;
Empty set, 2 warnings (0.00 sec)
```

```
mysql> SELECT e.FIRST_NAME, e.LAST_NAME FROM EMPLOYEE e JOIN JOB_POSITION jp ON e.JOB_POSITION_ID = jp.JOB_POSITION_ID WHERE jp.POSITION_NAME = 'Magazynier' ORDER BY e.BIRTH_DATE DESC LIMIT 5;
+-----+-----+
| FIRST_NAME | LAST_NAME |
+-----+-----+
| Patrycja   | Sowa      |
| Katarzyna  | Lis       |
| Janusz    | Nowak     |
| Krzysztof | Cybulska  |
+-----+-----+
```

Zadanie 5

1. Dla każdego **stanowiska pracy** wyświetl średnie zarobki pracowników, jeśli pracownik **nie pracuje w departamencie „IT” ani „HR”** i liczba pracowników jest **powyżej 3**.
2. Dla każdego **stanowiska pracownika** wyświetl:
 - o imię, nazwisko,
 - o datę urodzenia **najmłodszej i najstarszej** osoby,
 - o oraz **departament**, w którym pracują.

```
mysql> SELECT jp.POSITION_NAME, AVG(e.pensja), COUNT(*) FROM EMPLOYEE e JOIN DEPARTMENT d ON e.DEPARTMENT_ID = d.DEPARTMENT_ID JOIN JOB_POSITION jp ON e.JOB_POSITION_ID = jp.JOB_POSITION_ID WHERE d.DEPARTMENT_NAME NOT IN ('IT', 'HR') GROUP BY jp.POSITION_NAME HAVING COUNT(*) > 3;
+-----+-----+-----+
| POSITION_NAME | AVG(e.pensja) | COUNT(*) |
+-----+-----+-----+
| Kierownik sprzedazy | 5580.000000 | 5 |
| Magazynier | 4925.000000 | 4 |
+-----+-----+-----+
```

```
mysql> SELECT e.FIRST_NAME, e.LAST_NAME, jp.POSITION_NAME FROM EMPLOYEE e JOIN JOB_POSITION jp ON e.JOB_POSITION_ID = jp.JOB_POSITION_ID ORDER BY e.BIRTH_DATE DESC LIMIT 1;
+-----+-----+-----+
| FIRST_NAME | LAST_NAME | POSITION_NAME |
+-----+-----+-----+
| Agnieszka | Lewandowska | Ksiegowy      |
+-----+-----+-----+
```

```
mysql> SELECT e.FIRST_NAME, e.LAST_NAME, jp.POSITION_NAME FROM EMPLOYEE e JOIN JOB_POSITION jp ON e.JOB_POSITION_ID = jp.JOB_POSITION_ID ORDER BY e.BIRTH_DATE ASC LIMIT 1;
+-----+-----+-----+
| FIRST_NAME | LAST_NAME | POSITION_NAME |
+-----+-----+-----+
| Wojciech  | Krol      | Ksiegowy      |
+-----+-----+-----+
```

```

mysql> SELECT e.FIRST_NAME, e.LAST_NAME, jp.POSITION_NAME FROM EMPLOYEE e JOIN JOB_POSITION jp ON e.JOB_POSITION_ID =
jp.JOB_POSITION_ID;
+-----+-----+-----+
| FIRST_NAME | LAST_NAME | POSITION_NAME |
+-----+-----+-----+
| Tomasz | Kowal | Kierownik sprzedazy |
| Katarzyna | Lis | Magazynier |
| Marek | Nowicki | Administrator biura |
| Agnieszka | Lewandowska | Ksiegowy |
| Pawel | Kaczor | Kierownik sprzedazy |
| Ewa | Mazur | Programista |
| Mariusz | Zielony | Operator logistyki |
| Aneta | Kowalska | Specjalista ds. marketingu |
| Rafal | Sikora | Specjalista ds. HR |
| Olga | Czajka | Pracownik obslugi klienta |
| Janusz | Nowak | Magazynier |
| Barbara | Kowalczyk | Administrator biura |
| Michal | Zajac | Kierownik sprzedazy |
| Magdalena | Sadowska | Ksiegowy |
| Grzegorz | Kubiak | Specjalista ds. HR |
| Joanna | Wojcik | Programista |
| Pawel | Gajda | Operator logistyki |
| Sylwia | Baran | Specjalista ds. marketingu |
| Artur | Bednarz | Pracownik obslugi klienta |
| Elzbieta | Krawczyk | Kierownik sprzedazy |
| Krzysztof | Cybulski | Magazynier |
| Dorota | Michalak | Administrator biura |
| Wojciech | Krol | Ksiegowy |
| Agnieszka | Lis | Specjalista ds. HR |
| Tomasz | Zielinski | Programista |
| Monika | Sikora | Operator logistyki |
| Jakub | Wozniak | Specjalista ds. marketingu |
| Ewelina | Wrona | Pracownik obslugi klienta |
| Marcin | Krupa | Kierownik sprzedazy |
| Patrycja | Sowa | Magazynier |
+-----+-----+-----+

```

```

mysql> SELECT e.FIRST_NAME, e.LAST_NAME, jp.POSITION_NAME, d.DEPARTMENT_NAME FROM EMPLOYEE e JOIN JOB_POSITION jp ON
e.JOB_POSITION_ID = jp.JOB_POSITION_ID JOIN DEPARTMENT d ON e.DEPARTMENT_ID = d.DEPARTMENT_ID;
+-----+-----+-----+-----+
| FIRST_NAME | LAST_NAME | POSITION_NAME | DEPARTMENT_NAME |
+-----+-----+-----+-----+
| Krzysztof | Cybulski | Magazynier | Sprzedaz |
| Michal | Zajac | Kierownik sprzedazy | Sprzedaz |
| Pawel | Kaczor | Kierownik sprzedazy | Sprzedaz |
| Tomasz | Kowal | Kierownik sprzedazy | Sprzedaz |
| Dorota | Michalak | Administrator biura | Magazyn |
| Janusz | Nowak | Magazynier | Magazyn |
| Katarzyna | Lis | Magazynier | Magazyn |
| Wojciech | Krol | Ksiegowy | Administracja |
| Barbara | Kowalczyk | Administrator biura | Administracja |
| Marek | Nowicki | Administrator biura | Administracja |
| Agnieszka | Lis | Specjalista ds. HR | Ksiegowosc |
| Magdalena | Sadowska | Ksiegowy | Ksiegowosc |
| Agnieszka | Lewandowska | Ksiegowy | Ksiegowosc |
| Tomasz | Zielinski | Programista | HR |
| Grzegorz | Kubiak | Specjalista ds. HR | HR |
| Rafal | Sikora | Specjalista ds. HR | HR |
| Monika | Sikora | Operator logistyki | IT |
| Joanna | Wojcik | Programista | IT |
| Ewa | Mazur | Programista | IT |
| Jakub | Wozniak | Specjalista ds. marketingu | Logistika |
| Pawel | Gajda | Operator logistyki | Logistika |
| Mariusz | Zielony | Operator logistyki | Logistika |
| Ewelina | Wrona | Pracownik obslugi klienta | Marketing |
| Sylwia | Baran | Specjalista ds. marketingu | Marketing |
| Aneta | Kowalska | Specjalista ds. marketingu | Marketing |
| Marcin | Krupa | Kierownik sprzedazy | Obsluga klienta |
| Artur | Bednarz | Pracownik obslugi klienta | Obsluga klienta |
| Olga | Czajka | Pracownik obslugi klienta | Obsluga klienta |
| Patrycja | Sowa | Magazynier | Produkcja |
| Elzbieta | Krawczyk | Kierownik sprzedazy | Produkcja |
+-----+-----+-----+-----+

```