

SELECT / FROM / WHERE / ORDER BY / AS / ROWNUM

ZADANIE 1

WYŚWIETL WSZYSTKIE DANE ZNAJDUJĄCE SIĘ W TABELI REGIONS.

```
SELECT *  
FROM REGIONS;
```

ZADANIE 2

WYŚWIETL KOLUMNY FIRST_NAME, LAST_NAME, SALARY Z TABELI EMPLOYEES

```
SELECT FIRST_NAME, LAST_NAME, SALARY  
FROM employees;
```

ZADANIE 3

WYŚWIETL PONOWNIE KOLUMNY Z ZADANIA DRUGIEGO ZMIENIAJĄC ICH NAZWY W NAGŁÓWKU WYNIKOWYM NA POLSKIE TŁUMACZENIE

```
SELECT FIRST_NAME as imie, LAST_NAME as nazwisko, SALARY as pensja  
FROM employees;
```

ZADANIE 4

ZMODYFIKUJ POPRZEDNIE ZAPYTANIE TAK ABY WYŚWIETLAŁO TYLKO 10 PIERWSZYCH REKORDÓW

```
SELECT FIRST_NAME as imie, LAST_NAME as nazwisko, SALARY as pensja  
FROM employees  
WHERE ROWNUM<11;
```

ZADANIE 5

WYŚWIETL IMIĘ, NAZWISKO I DATĘ ZATRUDNIENIA (HIRE_DATE) WSZYSTKICH PRACOWNIKÓW, KTÓRZY ZARABIAJĄ WIĘCEJ NIŻ 10 000

```
SELECT first_name, last_name, hire_date  
FROM employees  
WHERE salary>10000;
```

ZADANIE 6

ROZBUDUJ POPRZEDNIE ZAPYTANIE TAK, ABY WYŚWIETLIĆ PRACOWNIKÓW ZARABIAJĄCYCH POWYŻEJ 10000 W KOLEJNOŚCI OD NAJDŁUŻEJ PRACUJĄCEGO (SORTOWANIE W KOLEJNOŚCI ROSNĄCEJ)

```
SELECT first_name, last_name, hire_date  
FROM employees  
WHERE salary>10000  
ORDER BY hire_date ASC;
```

DESCRIBE / LIKE / AND / OR / DISTINCT / GROUP BY / COUNT / MIN / MAX / AVG

ZADANIE 1

UŻYWAJĄC POLECENIA DESCRIBE WYPISZ PONIŻEJ TRZY DOWOLNE NAZWY KOLUMN Z WIDOKU ALL_TABLES.
ZAPISZ PONIŻEJ SPOSÓB UŻYCIA WW. POLECENIA.

```
DESCRIBE all_tables;  
OWNER TABLE_NAME TABLESPACE_NAME
```

ZADANIE 2

KORZYSTAJĄC Z WIDOKU ALL_TABLES WYPISZ PONIŻEJ TRZY DOWOLNE NAZWY ORAZ WŁAŚCICIELI TABELI,
DO KTÓRYCH MASZ DOSTĘP. ZAPISZ PONIŻEJ ZAPYTANIE, KTÓRE WYKONAŁEŚ ABY UZYSKAĆ WYNIK

```
SELECT *  
FROM all_tables;  
SYS DUAL SYSTEM OL$ HR REGIONS
```

ZADANIE 3

KORZYSTAJĄC Z WIDOKU ALL_TABLES WYŚWIETL WSZYSTKIE NAZWY TABEL, KTÓRYCH WŁAŚCICIELEM JEST UŻYTKOWNIK,
KTÓREGO NAZWA ROZPOCZYNA SIĘ OD 'PWR_'

```
SELECT *  
FROM all_tables  
WHERE OWNER LIKE 'PWR%';
```

ZADANIE 4

ZMODYFIKUJ POPRZEDNIE ZAPYTANIE TAK ABY WYŚWIETLAŁO WŁAŚCICIELA ORAZ NAZWĘ TABELI, DLA KTÓRYCH NAZWA
WŁAŚCICIELA ROZPOCZYNA SIĘ OD 'PWR_' LUB WŁAŚCICIELEM JEST 'XDB'

```
SELECT *  
FROM all_tables  
WHERE OWNER LIKE 'PWR%' OR OWNER like 'XDB';
```

ZADANIE 5

WYŚWIETL DEPARTMENT_ID Z TABELI JOB_HISTORY W KOLEJNOŚCI MALEJĄCEJ (10 WIERSZY), A NASTĘPNIE WYŚWIETL
TO POLE UNIKALNE (BEZ POWTÓRZEŃ – 6 WIERSZY)

```
SELECT department_id  
FROM job_history  
ORDER BY DEPARTMENT_ID DESC;
```

```
SELECT DISTINCT department_id  
FROM job_history  
ORDER BY DEPARTMENT_ID DESC;
```

ZADANIE 7

ZMODYFIKUJ POWYŻSZE ZAPYTANIE TAK ABY WYŚWIETLIĆ UNIKALNE DEPARTMENT_ID TYLKO DLA PRZYPADKÓW
GDY EMPLOYEE_ID JEST MNIEJSZE LUB RÓWNE 122 I JOB_ID ROZPOCZYNA SIĘ OD 'AC_' LUB 'ST_'

```
SELECT DISTINCT department_id, EMPLOYEE_ID, JOB_ID  
FROM job_history  
WHERE employee_id<=122 AND (job_id='AC_' OR job_id='ST_');
```

ZADANIE 8

KORZYSTAJĄC Z TABELI EMPLOYEES WYŚWIETL MANAGER_ID ORAZ LICZBĘ UNIKALNYCH EMPLOYEE_ID

```
SELECT COUNT(DISTINCT employee_id), manager_id  
FROM employees  
GROUP BY manager_id ;
```

ZADANIE 9

KORZYSTAJĄC Z TABELI EMPLOYEES WYŚWIETL MANAGER_ID ORAZ MINIMUM, MAXIMUM ORAZ ŚREDNIĄ ZAROBKÓW PODLEGLYCH MU PRACOWNIKÓW (CHECK: DLA MANAGER_ID = 100 PRAWIDŁOWYM WYNIKIEM JEST MIN = 5800, MAX = 17000 I ŚREDNIA = 11100)

```
SELECT MIN(SALARY), MAX(salary), AVG(salary), MANAGER_ID FROM employees  
GROUP BY manager_id;
```

ZAPYTANIE Z ZADANIA 9 WYŚWIETLA WARTOŚCI ŚREDNIEJ Z RÓŻNĄ DOKŁADNOŚCIĄ ORAZ W RÓŻNEJ NOTACJI (NP. 24000, 3237.5, 7.8E+03). W JAKI SPOSÓB SPOWODOWAĆ ABY WYŚWIETLIŁO WARTOŚCI Z DOKŁADNOŚCIĄ DO DWÓCH MIEJSC PO PRZECINKU DLA WSZYSTKICH REKORDÓW? (CZYLI 24000.00, 3237.50, 7766.67) NAPISZ ODPOWIEDNIE ZAPYTANIE.

```
SELECT MIN(SALARY), MAX(salary), ROUND(AVG(salary),2) MANAGER_ID  
FROM employees; --dokładność do dwóch miejsc po przecinku
```

```
SELECT MIN(SALARY), MAX(salary), TO_CHAR(AVG(salary), '99999999.99') MANAGER_ID  
FROM employees;
```

ALTER TABLE / ADD CONSTRAINT / PRIMARY KEY / (left/right/inner/full) JOIN / ON

ZADANIE 1

SPRAWDŹ W TABELI USER_INDEXES CZY W TWOJEJ BAZIE SĄ STWORZONE JAKIEŚ INDEXY UŻYTKOWNIKA. ZAPISZ PONIŻEJ ODPOWIEDŹ ORAZ SPOSÓB W JAKI NALEŻY TO SPRAWDZIĆ (ZAPYTANIE).

```
SELECT *
```

```
FROM user_indexes;
```

W owej bazie danych istnieją indexy użytkownika.

ZADANIE 2

A) UTWÓRZ KLUCZ GŁÓWNY (PRIMARY KEY) NA POLU COUNTRY_ID W TABELI COUNTRIES. NAZWIJ KLUCZ COUNTRY_ID_PK.

B) PONOWNIE SPRAWDŹ ZAWARTOŚĆ TABELI USER_INDEXES, TYM RAZEM WYŚWIETLAJĄC JEDYNNIE KOLUMNY TABLE_NAME, INDEX_NAME, UNIQUENESS. CO TERAZ ZWRACA TO ZAPYTANIE?

```
ALTER TABLE countries
```

```
ADD CONSTRAINT country_id_pk PRIMARY KEY(country_id);
```

SQL Error: ORA-02260: table can have only one primary key

```
SELECT table_name,index_name,uniqueness
```

```
FROM user_indexes;
```

Nic to nie zmieniło, bo ten klucz już istnieje.

ZADANIE 3

WYŚWIETL COUNTRY_NAME, REGION_NAME Z POŁĄCZONYCH TABEL COUNTRIES ORAZ REGIONS. TABELĘ POŁĄCZ WYKORZYSTUJĄC POLE REGION_ID.

```
SELECT countries.country_name, regions.region_name
```

```
FROM countries
```

```
INNER JOIN regions ON regions.region_id=countries.region_id;
```

ZADANIE 4

W TRZECZ ZAPYTANIACH WYKONAJ TRZY ZŁĄCZENIA (WEWNĘTRZNE, LEWE I PRAWY) TABEL DEPARTMENTS ORAZ EMPLOYEES (DO ZŁĄCZENIA WYKORZYSTAJ TYLKO POLE DEPARTMENT_ID). DLA KAŻDEGO ZŁĄCZENIA WYŚWIETL POLE DEPARTMENT_ID Z OBU TABEL RÓWNOCZEŚNIE. WYJAŚNIJ RÓŻNICE MIĘDZY OTRZYMANYMI WYNIKAMI – CZEMU RÓŻNE ZŁĄCZENIA DAJĄ RÓŻNE WYNIKI?

```
SELECT departments.department_id, employees.department_id
```

```
FROM departments
```

```
INNER JOIN employees ON departments.DEPARTMENT_ID=employees.DEPARTMENT_ID;
```

```
SELECT departments.department_id, employees.department_id
```

```
FROM departments
```

```
LEFT JOIN employees ON departments.DEPARTMENT_ID=employees.DEPARTMENT_ID;
```

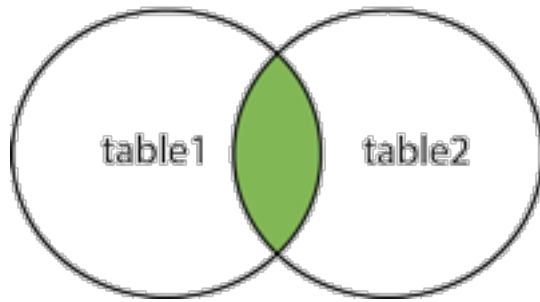
```
SELECT departments.department_id, employees.department_id
```

```
FROM departments
```

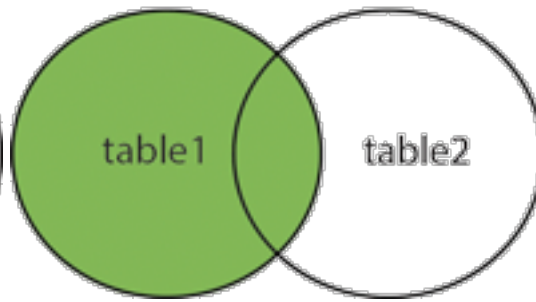
```
RIGHT JOIN employees ON departments.DEPARTMENT_ID=employees.DEPARTMENT_ID;
```

To wynika z poniższych diagramów:

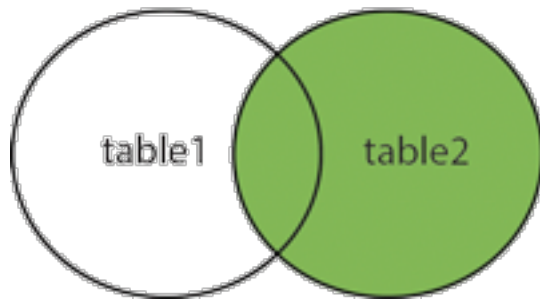
INNER JOIN



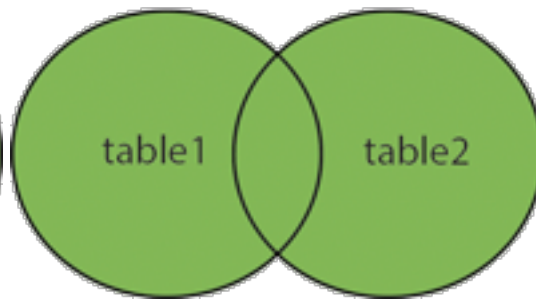
LEFT JOIN



RIGHT JOIN



FULL OUTER JOIN



ZADANIE 5

ZIDENTYFIKUJ NA JAKIE POLE POWINNO ZAŁOŻYĆ SIĘ KLUCZ GŁÓWNY W TABELI EMPLOYEES.

Klucz ten jest założony na employee_id, ponieważ służy on odróżnianiu pracowników.

PODZAPYTANIA SQL / ZŁOŻONE ZŁĄCZENIA

ZADANIE 1

NAZW DZIAŁÓW (DEPARTMENT_NAME), W KTÓRYCH PRACUJE CO NAJMNIEJ DWÓCH PRACOWNIKÓW, KTÓRZY ZARABIAJĄ PONAD 80% MAKSYMALNEJ PŁACY USTALONEJ NA SWOIM STANOWISKU (MAX_SALARY), WRAZ Z ILOŚCIAMI PRACOWNIKÓW SPEŁNIAJĄCYMI TO KRYTERIUM.

```
SELECT department_name, COUNT(employee_id) ilosc_pracownikow
FROM DEPARTMENTS
JOIN EMPLOYEES ON DEPARTMENTS.DEPARTMENT_ID = EMPLOYEES.DEPARTMENT_ID JOIN JOBS
ON EMPLOYEES.JOB_ID = JOBS.JOB_ID
WHERE (100*salary/max_salary) > 80
GROUP BY department_name
HAVING COUNT(employee_id) >= 2;
```

ZADANIE 2

WYŚWIEL NAZWISKA PRACOWNIKÓW O NUMERZE WIĘKSZYM, NIŻ PRACOWNIK O NAZWISKU (LAST_NAME) CHEN

```
SELECT last_name, employee_id
FROM employees
WHERE employee_id >
(
    SELECT employee_id FROM employees
    WHERE last_name='Chen'
);
```

ZADANIE 3

WYŚWIEL NAZWY DZIAŁÓW (DEPARTMENT_NAME), ICH NUMERY (DEPARTMENT_ID) ORAZ ILOŚCI PRACOWNIKÓW, KTÓRZY W NICH PRACUJĄ.

```
SELECT departments.department_name, departments.department_id, COUNT(*) ilosc_pracownikow
FROM DEPARTMENTS
FULL JOIN employees
ON DEPARTMENTS.DEPARTMENT_ID=employees.department_id
GROUP BY DEPARTMENTS.DEPARTMENT_NAME, departments.department_id;
```

ZADANIE 4

WYŚWIEL NAZWISKA PRACOWNIKÓW (LAST_NAME) O NUMERACH (EMPLOYEE_ID) WIĘKSZYCH, NIŻ CO NAJMNIEJ JEDEN (ANY) Z PRACOWNIKÓW DZIAŁU (DEPARTMENT_NAME) IT.

```
SELECT last_name, employee_id
FROM employees
WHERE employee_id > ANY
(
    SELECT department_id
    FROM departments
    WHERE department_name='IT'
);
```

ZADANIE 5

WYŚWIEL NAZWISKA PRACOWNIKÓW (LAST_NAME), KTÓRYCH WYPŁATA (SALARY) JEST TAKA SAMA, JAK KTÓREGOKOLWIEK PRACOWNIKA DZIAŁU (DEPARTMENT_NAME) IT

```
SELECT last_name, salary FROM employees WHERE salary = ANY
(
    SELECT department_id
    FROM departments
    WHERE department_name='IT'
);
```

PODZAPYTANIA SQL

NAPISZ ZAPYTANIE SQL DO BAZY DANYCH, KTÓRE SPOWODUJE WYŚWIETLENIE:

ZADANIE 1

NAZWISK (LAST_NAME) ORAZ WYPŁAT (SALARY) PRACOWNIKÓW, KTÓRZY ZARABIAJĄ PONAD 90% WYPŁATY PRACOWNIKA O NAZWISKU CHEN (TABELA EMPLOYEES)

```
SELECT last_name
FROM employees WHERE salary>(90/100)*
(
  SELECT salary
  FROM employees
  WHERE last_name='Chen'
);
```

ZADANIE 2

NAZWISK (LAST_NAME), NAZW DZIAŁÓW (DEPARTMENT_NAME) ORAZ WYPŁAT (SALARY) PRACOWNIKÓW, KTÓRZY ZARABIAJĄ WIĘCEJ, NIŻ KAŻDY PRACOWNIK DZIAŁU 'MARKETING' (TABELA EMPLOYEES ORAZ DEPARTMENTS)

```
SELECT last_name, department_name, salary
FROM employees
JOIN departments ON employees.department_id=departments.department_id WHERE salary>ALL
(
  SELECT salary
  FROM employees
  JOIN departments ON employees.department_id=departments.department_id
  WHERE department_name='Marketing'
);
```

ZADANIE 3

NAZW DZIAŁÓW (DEPARTMENT_NAME) I ILOŚCI PRACOWNIKÓW, W KTÓRYCH PRACUJE WIĘCEJ PRACOWNIKÓW, NIŻ W DZIALE 'IT' (TABELA EMPLOYEES ORAZ DEPARTMENTS)

```
SELECT department_name, COUNT(employee_id) ilosc_pracownikow
FROM departments
JOIN employees ON departments.department_id=employees.department_id
GROUP BY department_name
HAVING COUNT(employee_id) >
(
  SELECT COUNT(employee_id)
  FROM departments
  JOIN employees ON departments.department_id=employees.department_id
  WHERE department_name='IT'
  GROUP BY department_name
);
```

ZADANIE 4

NUMERÓW PRACOWNIKÓW (EMPLOYEE_ID), MANAGERA (MANAGER_ID) ORAZ NUMERU DZIAŁU (DEPARTMENT_ID), KTÓRZY MAJĄ TEGO SAMEGO MANAGERA LUB PRACUJĄ W TYM SAMYM DEPARTAMENCIE CO PRACOWNIK O NUMERZE 107 (TABELA EMPLOYEES)

```
SELECT employees.employee_id, departments.manager_id, departments.department_id FROM departments
JOIN employees ON departments.department_id=employees.department_id
WHERE departments.manager_id =
    (
        SELECT departments.manager_id
        FROM departments
        JOIN employees ON departments.department_id=employees.department_id
        WHERE employee_id='107'
    )
OR departments.department_name =
    (
        SELECT departments.department_name
        FROM departments
        JOIN employees ON departments.department_id=employees.department_id
        WHERE employee_id='107'
    );
```

ZADANIE 5

UŻYWAJĄC SŁOWA KLUCZOWEGO „EXISTS” ORAZ PODZAPYTANIA, SPOWODUJ WYŚWIETLENIE WSZYSTKICH NAZW DEPARTAMENTÓW, KTÓRYCH ID ISTNIEJE W TABELI EMPLOYEES.

(TIP: JEŻELI NAZWIEMY TABELĘ Z „NADZAPYTANIA” ALIASEM TO MOŻNA SIĘ ODWOŁAĆ DO JEJ POLA W PODZAPYTANIU POPRZEC ALIAS.NAZWA_POLA, TIP2: SCHEMAT ZAPYTANIA POWINIEN BYĆ NASTĘPUJĄCY – „WYBIERZ COŚ Z TABELKI1 GDZIE ISTNIEJE (WYBIERZ COŚ Z TABELKI2 GDZIE TABELKA2.KLUCZ_GŁÓWNY = TABELKA1.KLUCZ_GŁÓWNY)”)

```
SELECT department_name
FROM departments
WHERE EXISTS
    (
        SELECT department_id
        FROM employees
        WHERE departments.department_id=employees.department_id
    );
```

ZADANIE 6

SPOWODUJ WYŚWIETLENIE NUMERÓW PRACOWNIKÓW, IMIENIA, NAZWISKA ORAZ POLA, W KTÓRYM ZNAJDZIE SIĘ WARUNEK – JEŻELI ID DEPARTAMENTU ZNAJDUJE SIĘ W LOKALIZACJI O ID=1800 TO WYŚWIETL ‘TORONTO’, W PRZECIWNYM RAZIE WYŚWIETL ‘INNE’. DO ROZWIĄZANIA TEGO ZADANIA POTRZEBNE SĄ TABELY EMPLOYEES ORAZ DEPARTMENTS. DODATKOWO NALEŻY SKORZYSTAĆ Z PODZAPYTANIA ORAZ FUNKCJI „CASE WHEN WARUNEK THEN ... ELSE ... END”

```
SELECT employee_id as numer_pracownika, first_name as imie, last_name as nazwisko,
    CASE location_id
    WHEN 1800
    THEN 'Toronto'
    ELSE 'inne'
    END
    as lokalizacja
FROM employees
JOIN departments ON employees.department_id=departments.department_id;
```