

```
SELECT * FROM EMPLOYEES;
```

```
SELECT  
  
    first_name,  
  
    last_name,  
  
    salary,  
  
    RANK() OVER (ORDER BY salary DESC) AS ranking  
FROM EMPLOYEES;
```

```
SELECT  
  
    employee_id,  
  
    first_name,  
  
    last_name,  
  
    salary,  
  
    SUM(salary) OVER () AS suma_calkowita  
FROM EMPLOYEES;
```

```
SELECT table_name  
  
FROM all_tables  
  
WHERE owner = 'HR';
```

```
CREATE TABLE JOB_GRADES AS SELECT * FROM HR.JOB_GRADES;  
  
CREATE TABLE PRODUCTS AS SELECT * FROM HR.PRODUCTS;  
  
CREATE TABLE SALES AS SELECT * FROM HR.SALES;
```

```
SELECT * FROM JOB_GRADES;  
  
SELECT * FROM PRODUCTS;
```

```
SELECT * FROM SALES;
```

```
SELECT

    s.employee_id,

    e.last_name,

    p.product_name,

    ROUND(SUM(s.quantity * s.price), 2) AS wartosc_sprzedazy, RANK() OVER (ORDER BY SUM(s.quantity * s.price) DESC) AS ranking

FROM SALES s JOIN EMPLOYEES e ON s.employee_id = e.employee_id JOIN PRODUCTS p ON s.product_id = p.product_id

GROUP BY s.employee_id, e.last_name, p.product_name;
```

```
SELECT

    e.last_name,

    p.product_name,

    s.price,

    COUNT(*) OVER (PARTITION BY s.product_id, s.sale_date) AS liczba_transakcji,

    SUM(s.quantity * s.price) OVER (PARTITION BY s.product_id, s.sale_date) AS suma_zaplaty,

    LAG(s.price) OVER (PARTITION BY s.product_id ORDER BY s.sale_date, s.sale_id) AS poprzednia_cena,

    LEAD(s.price) OVER (PARTITION BY s.product_id ORDER BY s.sale_date, s.sale_id) AS nastepna_cena

FROM SALES s JOIN EMPLOYEES e ON s.employee_id = e.employee_id JOIN PRODUCTS p ON s.product_id = p.product_id;
```

```
SELECT

    p.product_name,

    s.price,

    TO_CHAR(s.sale_date, 'YYYY-MM') AS miesiac,

    SUM(s.quantity * s.price) OVER (PARTITION BY p.product_id, TO_CHAR(s.sale_date, 'YYYY-MM')) AS suma_miesieczna,

    SUM(s.quantity * s.price) OVER (PARTITION BY p.product_id, TO_CHAR(s.sale_date, 'YYYY-MM') ORDER BY s.sale_date, s.sale_id) AS suma_rosnaca

FROM SALES s JOIN PRODUCTS p ON s.product_id = p.product_id;
```

```
SELECT

    p.product_name,

    p.product_category,

    s2022.price AS cena_2022,

    s2023.price AS cena_2023,

    s2023.price - s2022.price AS roznica_cen,

    TO_CHAR(s2022.sale_date, 'MM-DD') AS dzien_miesiaca

FROM SALES s2022 JOIN SALES s2023

    ON s2022.product_id = s2023.product_id

    AND TO_CHAR(s2022.sale_date, 'MM-DD') = TO_CHAR(s2023.sale_date, 'MM-DD')

    AND EXTRACT(YEAR FROM s2022.sale_date) = 2022

    AND EXTRACT(YEAR FROM s2023.sale_date) = 2023

JOIN products p ON s2022.product_id = p.product_id

ORDER BY p.product_name, dzien_miesiaca;
```

```
SELECT

    p.product_category,

    p.product_name,

    s.price,

    MIN(s.price) OVER (PARTITION BY p.product_category) AS min_cena_kategorii,

    MAX(s.price) OVER (PARTITION BY p.product_category) AS max_cena_kategorii,

    MAX(s.price) OVER (PARTITION BY p.product_category) - MIN(s.price) OVER (PARTITION BY p.product_category) AS roznica

FROM SALES s JOIN PRODUCTS p ON s.product_id = p.product_id;
```

```
SELECT

    p.product_name,
```

```
s.sale_date,  
  
s.price,  
  
ROUND(AVG(s.price) OVER (PARTITION BY s.product_id ORDER BY s.sale_date ROWS BETWEEN 1 PRECEDING AND 1 FOLLOWING), 2) AS srednia_kroczaca  
FROM SALES s JOIN PRODUCTS p ON s.product_id = p.product_id  
  
ORDER BY p.product_name, s.sale_date;
```

```
SELECT  
  
p.product_name,  
  
p.product_category,  
  
s.price,  
  
RANK() OVER (PARTITION BY p.product_category ORDER BY s.price DESC) AS ranking,  
  
ROW_NUMBER() OVER (PARTITION BY p.product_category ORDER BY s.price DESC) AS numer_w_kategorii,  
  
DENSE_RANK() OVER (PARTITION BY p.product_category ORDER BY s.price DESC) AS ranking_gesty  
FROM SALES s JOIN PRODUCTS p ON s.product_id = p.product_id  
  
ORDER BY p.product_category, ranking;
```

```
SELECT  
  
e.last_name, p.product_name, s.sale_date, s.quantity * s.price AS wartosc_sprzedazy,  
  
SUM(s.quantity * s.price) OVER (PARTITION BY s.employee_id ORDER BY s.sale_date, s.sale_id) AS suma_rosnaca_pracownika,  
  
RANK() OVER (ORDER BY s.quantity * s.price DESC) AS globalny_ranking  
FROM SALES s JOIN EMPLOYEES e ON s.employee_id = e.employee_id JOIN products p ON s.product_id = p.product_id  
  
ORDER BY e.last_name, s.sale_date;
```

```
SELECT  
  
e.first_name,  
  
e.last_name,  
  
e.job_id
```

```
FROM EMPLOYEES e JOIN SALES s ON e.employee_id = s.employee_id
```

```
GROUP BY e.first_name, e.last_name, e.job_id;
```