1. select employee_id, salary, department_id, dense_rank() over(partition by department_id order by salary desc) as salary_rank from employees;

2. select department_id, extract(year from hire_date) as hire_year, salary, hire_date, sum(salary) over(partition by department_id order by hire_date) as updated_salary from employees;

3. select e.*, sum(salary) over(order by department_id rows between unbounded preceding and current row) as tot from employees e;

4. select extract(year from hire_date) as hire_year, extract(month from hire_date) as hire_month, salary, sum(salary) over(partition by extract(year from hire_date), extract(month from hire_date) order by extract(year from hire_date), extract(month from hire_date)) as tot_sal from employees order by hire_date;

5. select extract(year from hire_date) as hire_year, extract(month from hire_date) as hire_month, extract(day from hire_date) as hire_day, sum(salary) over(partition by extract(year from hire_date), extract(month from hire_date) order by hire_date) as updated_salary from employees where hire_date = last_day(hire_date);

6. select hire_year, emp_cnt, lag(emp_cnt) over(order by hire_year) as prev_year_emp_cnt, nvl(emp_cnt - lag(emp_cnt) over(order by hire_year), 0) diff from (select extract(year from hire_date) as hire_year, count(*) as emp_cnt from employees group by extract(year from hire_date)) yearly_emp_cnt order by hire_year;

7. select employee_id, salary, department_id, rank() over(partition by department_id order by salary desc) as salary_rank from employees;

8. select ord_no, purch_amt, round(purch_amt / sum(purch_amt) over(), 4)*100 as perc, sum(purch_amt) over() from orders;

9. select first_value(employee_id) over(partition by department_id order by salary desc) highest, e.* from employees e;

10. select sum(pro_price) over (partition by pro_com order by pro_price rows between unbounded preceding and current row) sum_pr, it.* from item_mast it;

11. select last_value(hire_date) over(partition by department_id order by hire_date) first_accepted_emp, e.* from employees e;

12. select * from (select row_number() over(partition by location_id order by department_id) rownumm, d.* from departments d) where rownumm = 1;

13. select dense_rank() over(partition by customer_id order by ord_no), o.* from orders o;

14. select * from (select sum(salary) over(partition by job_title order by salary rows between unbounded preceding and current row) as cum_sal_jname, job_title, salary from employees join jobs using(job_id));

15. select max(purch_amt) over(partition by extract(month from ord_date) order by ord_date) as c, extract(month from ord_date) monthh, o.* from orders o;

16. select rank() over(partition by salesman_id order by purch_amt desc) rank_order, o.* from orders o;

17. select min(grade) over(partition by salesman_id), c.* from customer c;

18. select employee_id, job_id, hire_date, salary, lag(salary) over(partition by job_id order by hire_date) as prev_salary, salary - lag(salary) over(partition by job_id order by hire_date) as salary_difference from employees;

19. select * from (select e.*, max(salary) over(partition by department_id) maxx from employees e) where salary = maxx;

20. select job_id, max(salary) over(partition by job_id) as max_salary from employees;