

Schaltjahr

```
SELECT
  to_char(last_day(add_months(trunc(sysdate,'y'),1)), 'DD') azt
FROM
  dual
```

Anzahl Wochentage

```
WITH x as (
  SELECT level lvl
         FROM dual
         CONNECT BY level <= (
           add_months(trunc(sysdate,'y'),12) - trunc(sysdate,'y')
         )
)
SELECT
  to_char(trunc(sysdate, 'y') + lvl -1, 'DAY') wochentag,
  count(*) anzahl
FROM
  x
GROUP BY to_char(trunc(sysdate, 'y') + lvl -1, 'DAY')
```

Differenz Nachfolger

```
WITH x
  as (
    SELECT level q,
           add_months(trunc(sysdate,'y'),(rownum-1) * 3) q_start
    FROM dual
    CONNECT BY level <= 4
  )
SELECT nvl((next_q - q_start), 0) diff FROM (
  SELECT q_start,
         lead(q_start,1) over (order by q_start) next_q
  FROM x)
```

Nachfolger die 2.

```
SELECT ename,
       hiredate,
       next_hd,
       next_hd - hiredate dif
FROM (
  SELECT deptno, ename, hiredate,
         lead(hiredate, cnt-rn+1)
           over (order by hiredate) next_hd
  FROM (
    SELECT deptno, ename, hiredate,
           count(*) over (partition by hiredate) cnt,
           row_number()
             over (partition by hiredate order by empno) rn
    FROM emp
    WHERE deptno = 10
  )
)
```

CSV

```

SELECT deptno,
       ltrim(SYS_CONNECT_BY_PATH(ename, ','), ',') emps
FROM (
       SELECT deptno,
              ename,
              row_number() over
                (partition by deptno order by empno) rn,
              count(*) over
                (partition by deptno) cnt
       FROM
         emp
       )
WHERE level = cnt
       start with rn = 1
       connect by prior deptno = deptno and prior rn = rn - 1

```

IN-List

```

SELECT
  empno, ename, sal, deptno
FROM
  emp
WHERE
  empno in (
    SELECT to_number(
      rtrim(
        substr(emps,
              instr(emps, ',', 1, iter.pos) + 1,
              instr(emps, ',', 1, iter.pos + 1) -
              instr(emps, ',', 1, iter.pos)), ',')
      ) emps
    FROM (
      SELECT ', ' || '7654,7698,7782,7788' || ', ' emps
      FROM dual) csv,
      (SELECT
        rownum pos
      FROM
        emp) iter
    WHERE
      iter.pos <= ((length(csv.emps) -
                    length(replace(csv.emps, ',')))-1)
  )

```

Zahl

```

SELECT to_number (
CASE
  when
    replace(translate(mixed, '0123456789', '9999999999'), '9', '')
    is not null
  then
    replace(translate(mixed, replace(
      translate(mixed, '0123456789', '9999999999'), '9'),
      rpad('#', length(mixed), '#')), '#', '')
  else
    mixed
END

```

```

    ) mixed
FROM
  v
WHERE
  instr(translate(mixed, '0123456789', '9999999999'), '9') > 0

```

Zahl die 2.

```

SELECT * from
  (SELECT
    REGEXP_REPLACE(mixed, '(\D)', '') mixed from v)
WHERE
  mixed is not null

```

Model

```

SELECT * FROM
  (SELECT country.*, 0 growth FROM country)
model
  dimension by (country_id,year)
  measures(population, growth)
  rules upsert all (
    population['INDIA',2022] = population['INDIA',2012]* 2,
    population['INDIA',2032] = population['INDIA',2012]* 6,
    population['USA', ANY] = population['INDIA',cv()] / 2,
    growth[ANY,ANY] = population[cv(),cv()]
      / population[cv(),cv() - 10]
  ) ORDER BY country_id, year

```

connect by

```

WITH
  x as (
    SELECT level id
      FROM
        dual
      connect by level <= 10
  ),
  y as (
    SELECT level id
      FROM
        dual
      connect by level <= 10
  )
SELECT * from x,y where x.id = y.id

```

model

```

WITH
  x as (
    SELECT array id
      FROM dual
  )
model
  dimension by (0 idx)
  measures (1 array)
  rules iterate (10) (
    array[iteration_number] = iteration_number + 1
  )

```

```

)
)
SELECT * FROM x

```

Quartale

```

WITH
  x as (
    SELECT level q,
           add_months(trunc(sysdate,'y'),(rownum-1) * 3) q_start,
           add_months(trunc(sysdate,'y'), rownum * 3) -1 q_end
    FROM dual
    connect by level <= 4
  )
SELECT * FROM x

```

Summe

```

SELECT ename, sal,
       sum(sal) over (order by sal, empno) as running_total
FROM
  emp
ORDER BY
  sal

```

Insert All

```

BEGIN
  execute immediate('truncate table dept_east');
  execute immediate('truncate table dept_mid');
  execute immediate('truncate table dept_west');
  INSERT ALL
    when loc in ('NEW YORK', 'BOSTON') then
      into dept_east(deptno, dname, loc)
      values(deptno, dname, loc)
    when loc = 'CHICAGO' then
      into dept_mid(deptno, dname, loc)
      values(deptno, dname, loc)
    else
      into dept_west(deptno, dname, loc)
      values(deptno, dname, loc)
  SELECT
    deptno, dname, loc
  FROM
    dept;
END;

```

Duplikate

```

DELETE FROM
  dupes
WHERE id not in (SELECT min(id) from dupes group by name)

```

Pivot

```

SELECT

```

```

    sum(case when deptno=10 then 1 else 0 end) as deptno_10,
    sum(case when deptno=20 then 1 else 0 end) as deptno_20,
    sum(case when deptno=30 then 1 else 0 end) as deptno_30
FROM
    emp

```

TOP-N

```

SELECT JOB, ENAME, SAL, SAL_RANK FROM
    (SELECT job, ename, sal,
        rank() over(
            partition by job
            order by sal desc) sal_rank
    FROM emp where SAL is not null)
WHERE
    sal_rank <=2
ORDER BY
    job, sal_rank

```

Nth-Value

```

SELECT
    DISTINCT JOB, sal_2nd_top
FROM
    (SELECT
        job, ename, sal,
        nth_value (sal, 2) over (
            partition by job
            order by sal desc
        ) sal_2nd_top
    FROM
        emp)
WHERE sal = sal_2nd_top

```

ISLEAF

```

SELECT
    lpad('<>', 15 * (level-1), '<>') || e.last_name Name,
    connect_by_isleaf
FROM
    hr.employees e
    start with e.manager_id is null
    connect by prior e.employee_id = e.manager_id
ORDER siblings BY e.last_name

```

ROLLUP

```

SELECT case
    when (total <> 'TOTAL') and (dep_total <> 'SUM JOB') then
        total
    else ' ' end c1,
    case
    when (total = 'TOTAL') and (dep_total = 'SUM JOB') then
        'TOTAL'
    when (total <> 'TOTAL') and (dep_total = 'SUM JOB') then
        'SUM JOB'
    else ' ' end c2,

```

```
    case when deptno is null then
      , ,
    else to_char(deptno,'9999999') end deptno,
    sal sal_
FROM (
SELECT
  decode(grouping(job),1, 'TOTAL', job) total,
  decode(grouping(deptno),1, 'SUM JOB', deptno) dep_total,
  deptno,
  sum(sal) sal
FROM
  emp
WHERE
  sal is not null
GROUP BY
  rollup(job, deptno))
```