

## TRIM LTRIM RTRIM Functions

```

SELECT TRIM ('      My Name is Adam      ') trimmed_text from dual;
SELECT TRIM (' ' FROM '      My Name is Adam      ') trimmed_text from dual;
SELECT TRIM (BOTH ' ' FROM '      My Name is Adam      ') trimmed_text from dual;
SELECT TRIM (LEADING ' ' FROM '      My Name is Adam      ') trimmed_text from dual;
SELECT TRIM (TRAILING ' ' FROM '      My Name is Adam      ') trimmed_text from dual;
SELECT TRIM (TRAILING 'm' FROM '      my Name is Adam      ') trimmed_text from dual;
SELECT TRIM (TRAILING 'm' FROM 'my Name is Adam') trimmed_text from dual;
SELECT TRIM (TRAILING 'm' FROM 'my Name is Adammmmm') trimmed_text from dual;
SELECT TRIM (LEADING 'm' FROM 'my Name is Adam') trimmed_text from dual;
SELECT TRIM (BOTH 'm' FROM 'my Name is Adam') trimmed_text from dual;
SELECT TRIM ('m' FROM 'my Name is Adam') trimmed_text from dual;
SELECT TRIM ('m' FROM 'my Name is Ada') trimmed_text from dual;
SELECT TRIM (TRAILING 'm' FROM 'my Name is Ada') trimmed_text from dual;
SELECT TRIM (TRAILING 'my' FROM 'my Name is Ada') trimmed_text from dual;

SELECT RTRIM ('  my Name is Adam  ') r_trimmed_text from dual;
SELECT LTRIM ('  my Name is Adam  ') l_trimmed_text from dual;
SELECT LTRIM ('my Name is Adam', 'my') l_trimmed_text from dual;
SELECT RTRIM ('my Name is Adam', 'my') r_trimmed_text from dual;
SELECT RTRIM ('my Name is Adammmmm', 'my') r_trimmed_text from dual;
SELECT LTRIM ('www.yourwebsite.com', 'w.') l_trimmed_text from dual;
SELECT RTRIM(LTRIM('www.yourwebsitename.com', 'w.'), '.com') trimmed_text from dual;
SELECT ltrim('1237982434www.yourwebsitename.com', '0123456789') trimmed_text from dual;

```

Character+Functions+--+Part+3+(TRIM,+LTRIM,+RTRIM+Functions)(Code+Samples).sql

## REPLACE LPAD RPAD Functions

```
SELECT first_name, REPLACE(first_name,'a') rpl FROM employees;  
SELECT first_name, REPLACE(first_name,'a','-') rpl FROM employees;  
SELECT first_name, REPLACE(first_name,'le','-') rpl FROM employees;  
SELECT first_name, REPLACE(first_name,'und','-') rpl FROM employees;  
SELECT first_name, lpad(first_name,10,'*') pad FROM employees;  
SELECT first_name, rpad(first_name,10,'*') pad FROM employees;  
SELECT first_name, rpad(first_name,6,'*') pad FROM employees;  
SELECT first_name, lpad(first_name,6,'*') pad FROM employees;  
SELECT first_name, lpad('My name is ',20,'-') pad FROM employees;  
SELECT first_name, lpad('My name is '||last_name ,20,'-') pad FROM employees;
```



Character+Functions+--+Part+4+(REPLACE,+LPAD,+RPAD+Functions)(Code+Samples).sql

# TO\_CHAR



TO\_CHAR+TO\_DATE+TO\_NUMBER+Functions+(Part+1)+(Code+Samples).sql

```
SELECT first_name, hire_date FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'YYYY') "Formatted Date" FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'YY') "Formatted Date" FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'RR') "Formatted Date" FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'YEAR') "Formatted Date" FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'MM') "Formatted Date" FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'MM-YYYY') "Formatted Date" FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'MON-YYYY') "Formatted Date" FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'MON-yyyy') "Formatted Date" FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'mon-yyyy') "Formatted Date" FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'Mon-yyyy') "Formatted Date" FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'MONTH-yyyy') "Formatted Date" FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'Month-yyyy') "Formatted Date" FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'DD-Month-yyyy') "Formatted Date" FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'DY-Month-yyyy') "Formatted Date" FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'Dy-Month-yyyy') "Formatted Date" FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'Day-Month-yyyy') "Formatted Date" FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'Dy-Month-yyyy HH12') "Formatted Date" FROM employees;
SELECT first_name, hire_date, to_char(hire_date, 'Dy-Month-yyyy HH24') "Formatted Date" FROM employees;
```

# Oracle Conditional+Expressions CASE Expressions

```
SELECT first_name, last_name, job_id, salary, CASE job_id
      WHEN 'ST_CLERK' THEN salary * 1.2
      WHEN 'SA_REP'   THEN salary * 1.3
      WHEN 'IT_PROG'  THEN salary * 1.4
      ELSE 0
    END "UPDATED SALARY"
FROM employees;
```

```
SELECT first_name, last_name, job_id, salary,
      CASE job_id
      WHEN 'ST_CLERK' THEN salary * 1.2
      WHEN 'SA_REP'   THEN salary * 1.3
      WHEN 'IT_PROG'  THEN salary * 1.4
      ELSE salary
    END "UPDATED SALARY"
FROM employees;
```



Oracle+Conditional+Expressions--+CASE+Expressions+(Code+Samples).sql

```
SELECT first_name, last_name, job_id, salary,
      CASE
      WHEN job_id = 'ST_CLERK' THEN salary*1.2
      WHEN job_id = 'SA_REP'   THEN salary*1.3
      WHEN job_id = 'IT_PROG'  THEN salary*1.4
      ELSE salary
    END "UPDATED SALARY"
FROM employees;
```

```
SELECT first_name, last_name, job_id, salary,
      CASE
      WHEN job_id = 'ST_CLERK' THEN salary*1.2
      WHEN job_id = 'SA_REP'   THEN salary*1.3
      WHEN job_id = 'IT_PROG'  THEN salary*1.4
      WHEN last_name = 'King'  THEN 2*salary
      ELSE salary END "UPDATED SALARY"
FROM employees;
```

```
SELECT first_name, last_name, job_id, salary,
      CASE
      WHEN job_id = 'AD_PRES'   THEN salary*1.2
      WHEN job_id = 'SA_REP'   THEN salary*1.3
      WHEN job_id = 'IT_PROG'  THEN salary*1.4
      WHEN last_name = 'King'  THEN 2*salary
      ELSE salary
    END "UPDATED SALARY"
FROM employees;
```

```
SELECT first_name, last_name, job_id, salary
FROM employees
WHERE (CASE
      WHEN job_id = 'IT_PROG' AND salary > 5000 THEN 1
      WHEN job_id = 'SA_MAN' AND salary > 10000 THEN 1
      ELSE 0
    END) = 1;
```

## Oracle Conditional Expressions DECODE Function

```
SELECT DECODE (1, 1, 'One', 2, 'Two') result FROM dual;
```

```
SELECT DECODE (25, 1, 'One', 2, 'Two', 3, 'Three', 'Not Found') result FROM dual;
```

```
SELECT first_name, last_name, job_id, salary,  
       DECODE(job_id, 'ST_CLERK', salary*1.20,  
               'SA_REP'   , salary*1.30,  
               'IT_PROG'  , salary*1.50 ) as updated_salary  
FROM EMPLOYEES;
```

```
SELECT first_name, last_name, job_id, salary,  
       DECODE(job_id, 'ST_CLERK', salary*1.20,  
               'SA_REP'   , salary*1.30,  
               'IT_PROG'  , salary*1.50,  
               salary) as updated_salary  
FROM EMPLOYEES;
```



Oracle+Conditional+Expressions+--DECODE+Function+(Code+Samples).sql



## AVG Function

```
SELECT avg(salary), avg(all salary), avg(distinct salary) FROM employees;
```

```
SELECT avg(salary), avg(all salary), avg(distinct salary)  
FROM employees WHERE job_id = 'IT_PROG';
```

```
SELECT avg(salary), avg(all salary), avg(distinct salary), salary  
FROM employees WHERE job_id = 'IT_PROG';
```

```
SELECT avg(commission_pct) FROM employees;
```



AVG+Function+(Code+Samples).sql

```
SELECT avg(commission_pct), avg(nvl(commission_pct,0)) FROM employees;
```

## COUNT Function

```
SELECT count(*),  
        count(commission_pct),  
        count(distinct commission_pct),  
        count(distinct nvl(commission_pct,0))  
FROM employees;
```



COUNT+Function+(Code+Samples).sql

## MAX Function

```
SELECT max(salary), max(hire_date), max(first_name) FROM employees;
```

```
SELECT * FROM employees ORDER BY first_name;
```



MAX+Function+(Code+Samples).sql



## MIN Function

```
SELECT * FROM employees;  
SELECT min(salary), min(commission_pct), min(nvl(commission_pct,0)),  
       min(hire_date), min(first_name)  
FROM employees
```



MIN+Function+(Code+Samples).sql

## SUM Function

```
SELECT sum(salary), sum(ALL salary), sum(DISTINCT salary), sum(hire_date) FROM employees;
```

```
SELECT sum(salary), sum(ALL salary), sum(DISTINCT salary) FROM employees;
```



SUM+Function+(Code+Samples).sql

# GROUP BY Clause

```
SELECT avg(salary) FROM employees;
```

```
SELECT avg(salary) FROM employees WHERE job_id = 'IT_PROG';
```

```
SELECT avg(salary) FROM employees WHERE job_id = 'IT_PROG' or job_id = 'SA_REP';
```

```
SELECT job_id, avg(salary) FROM employees  
GROUP BY job_id;
```

```
SELECT job_id, avg(salary) FROM employees  
GROUP BY job_id  
ORDER BY avg(salary);
```



GROUP+BY+Clause+(Part+1)+(Code+Samples).sql

```
SELECT job_id, avg(salary) FROM employees  
GROUP BY job_id  
ORDER BY avg(salary) DESC;
```

```
SELECT job_id, department_id, avg(salary) FROM employees  
GROUP BY job_id, department_id;
```

```
SELECT job_id, department_id, avg(salary), count(*) FROM employees  
GROUP BY job_id, department_id  
ORDER BY count(*) DESC;
```

```
SELECT job_id, department_id, manager_id, avg(salary), count(*) FROM employees  
GROUP BY job_id, department_id, manager_id  
ORDER BY count(*) DESC;
```

```
SELECT job_id, department_id, avg(salary), count(*) FROM employees  
GROUP BY department_id, job_id, manager_id;
```

## GROUP BY Clause

```
SELECT job_id, department_id, avg(salary) FROM employees  
GROUP BY job_id;
```

```
SELECT job_id, department_id, avg(salary) FROM employees  
GROUP BY job_id, department_id;
```

```
SELECT job_id, avg(salary) FROM employees  
GROUP BY job_id;
```

```
SELECT avg(salary) FROM employees  
GROUP BY job_id;
```



GROUP+BY+Clause+(Part+2)+(Code+Samples).sql

```
SELECT job_id, avg(salary) FROM employees  
GROUP BY job_id, department_id;
```

```
SELECT job_id, sum(salary), max(hire_date), count(*) FROM employees  
GROUP BY job_id, department_id;
```

```
SELECT job_id, sum(salary), max(hire_date), count(*) FROM employees  
GROUP BY job_id;
```

```
SELECT job_id, sum(salary), max(hire_date), count(*) FROM employees  
WHERE job_id IN ('IT_PROG', 'ST_MAN', 'AC_ACCOUNT')  
GROUP BY job_id;
```

# HAVING Clause

```
SELECT job_id, avg(salary) FROM employees
GROUP BY job_id;

SELECT job_id, avg(salary) FROM employees
WHERE avg(salary) > 10000
GROUP BY job_id;

SELECT job_id, avg(salary) FROM employees
GROUP BY job_id
HAVING avg(salary) > 10000;

SELECT job_id, avg(salary) FROM employees
HAVING avg(salary) > 10000
GROUP BY job_id;

SELECT job_id, avg(salary) FROM employees
WHERE hire_date > '28-MAY-05'
GROUP BY job_id
HAVING avg(salary) > 10000;

SELECT job_id, avg(salary) FROM employees
WHERE manager_id = 101
GROUP BY job_id
HAVING avg(salary) > 10000;

SELECT job_id, avg(salary) FROM employees
WHERE salary > 5000
GROUP BY job_id
--HAVING avg(salary) > 10000;
/

SELECT job_id, avg(salary) FROM employees
--WHERE salary > 10000
GROUP BY job_id
HAVING avg(salary) > 5000;
```



HAVING+Clause+(Code+Samples).sql