

Oracle Academy

Database Programming with SQL

Instructor Resource Guide

Notes for PowerPoint Slides:

SECTION 2 LESSON 1 – Conversion Functions

Slide 1: Conversion Functions

Lesson Preparation

When there is extra time in class, have students work on the self-test software.

What to Watch For

This is a lesson that requires considerable student practice. Check to make sure that, when students get output returned from a query, it is the correct output.

Relate the idea of presenting information in a more-readable format to the process of preparing the data-modeling presentation for the clients. It was important to have the information in a format that was easily understood by everyone.

Connections

Ask students to find examples of formatted data on the Internet or in the newspaper. Sources include the financial page, statistical information, grade reports, etc. Ask them to write a SQL statement to convert default format to the source format. Have students present their findings to the class.

Slide 2: What Will I Learn?

No instructor notes for this slide

Slide 3: Why Learn It?

Ask students: How do you choose a book to read or buy? Does the cover help you make a decision? What if you open the book and there are no pictures and the type is very small. Are you still interested?

Why do CDs and video tapes have colorful, interesting labels? Have you ever bought a new candy bar because the wrapper attracted your attention? As someone once said, "looks are everything"!

Slide 4: Tell Me / Show Me – When a table is created for a database ...

Before beginning this lesson's discussion of data-type conversions and the Oracle Server's capability to implicitly do data-type conversions, explain to students the difference between "implicit" and "explicit." Use the example, "When she rolled her eyes and sighed, Sue implicitly told Bill she didn't want to go the dance with him. Bill didn't understand her actions until she explicitly said, 'Bill, I already told you, I have a date for the dance!'" The dictionary defines "implicit" as something that is "implied but not directly expressed" and explicit as "clearly formulated or defined."

Students may not be familiar with concept of data types. Explain that most programming languages require the programmer to declare the data type of every data object. For the data stored in a database, the SQL programmer defines a data type for every column in

the database. Explain that later in the course, they will learn more about SQL data types, but for now they will be using VARCHAR2, CHAR, NUMBER, and DATE.

Slide 5: Tell Me / Show Me – VARCHAR2: Used for character data...

In this lesson, students will explicitly define data-type conversions to ensure the reliability of SQL statements.

Slide 6: Tell Me / Show Me – The ORACLE Server can internally...

No instructor notes for this slide

Slide 7: Tell Me / Show Me – DATE CONVERSION TO CHARACTER DATA

No instructor notes for this slide

Slide 8: Tell Me / Show Me – DATE CONVERSION TO CHARACTER DATA

The tables show the different format models that ..

No instructor notes for this slide

Slide 9: Tell Me / Show Me – For example, the following query returns..

After introducing each of the four data-type conversions, explain the format model table and the use of "fm" and "spth." STOP and allow students time to practice. SELECT id, TO_CHAR(event_date, 'MONTH DD, YYYY') FROM d_events; returns:

```
100    MAY 14, 2004
105    APRIL 28, 2004
```

Slide 10: Tell Me / Show Me – DATE AND TIME FORMAT MODELS

After introducing each of the four data-type conversions, explain the format model table and the use of "fm" and "th." STOP and allow students time to practice.

Conversion answers: (The date will vary depending on the SYSDATE.)

SELECT TO_CHAR(SYSDATE, 'fmMonth ddth, YYYY') FROM DUAL;

returns: August 6th, 2004 **Sysdate returns the current date**

SELECT TO_CHAR(SYSDATE, 'Month dd, YYYY') FROM DUAL;

returns: August 06, 2004 **Sysdate returns the current date**

SELECT TO_CHAR(SYSDATE, 'fmMON dd, YYYY') FROM DUAL;

returns: AUG 6, 2004 **Sysdate returns the current date**

SELECT TO_CHAR(SYSDATE, 'fmMonth ddth, Day, YyYY') FROM DUAL;

returns: August 6th, Friday, Two Thousand Four **Sysdate returns the current date**

YYYY returns the year in number.

Slide 11: Tell Me / Show Me – NUMBER CONVERSION TO CHARACTER DATA (VARCHAR2)

No instructor notes for this slide

Slide 12: Tell Me / Show Me – NUMBER CONVERSION TO CHARACTER DATA (VARCHAR2) The table illustrates some of the format ...

No instructor notes for this slide

Slide 13: Tell Me / Show Me – NUMBER CONVERSION TO CHARACTER DATA (VARCHAR2) Can you identify the format models ...

Conversion answers:

\$3000.00	TO_CHAR(salary, '\$9999.99')
4,500	TO_CHAR(salary, '9,999')
9,000.00	TO_CHAR(salary, '9,999.99')
0004422	TO_CHAR(salary, '0009999')

Slide 14: Tell Me / Show Me – CHARACTER CONVERSION TO NUMBER

Ask students to use the DUAL table to convert their age to a number.

Slide 15: Tell Me / Show Me – CHARACTER CONVERSION TO DATE

No instructor notes for this slide

Slide 16: Tell Me / Show Me – When making a character-to-date ...

The fx format may require more practice. Remind students to think of fx as "format exact" so the fxdate of the format model must exactly match the format of the date being converted.

Ask students to use the fx modifier to convert each of the following to the default date format.

June19 2004

```
SELECT TO_DATE('June19 2004', 'fxMonthDD RRRR')AS Convert
FROM DUAL;
```

July312004

```
SELECT TO_DATE('July312004', 'fxMonthDDRRRR')AS Convert
FROM DUAL;
```

Using the DUAL table, ask students to format their birthday in five different ways.

Remind students that they will first have to convert their "character" data to date data and then format it. If they were using a DATE data-type column from a table such as hire_date, the conversion TO_DATE before formatting would not be necessary.

```
SELECT TO_CHAR(TO_DATE('June 19, 1990', 'Month dd, YYYY'), 'MON DD
YYYY')AS Birthday
FROM DUAL;
```

Slide 17: Tell Me / Show Me – The fx modifier rules are:

No instructor notes for this slide

Slide 18: Try It/Solve It

June19 2004

```
SELECT TO_DATE('June19 2004', 'fxMonthDD RRRR')AS Convert
FROM DUAL;
```

July312004

```
SELECT TO_DATE('July312004', 'fxMonthDDRRRR')AS Convert
FROM DUAL;
```

June 19, 1990

```
SELECT TO_CHAR(TO_DATE('June 19, 1990','Month dd, YYYY'),'MON DD
YYYY')AS Birthday
FROM DUAL;
```

Slide 19: Tell Me / Show Me – THE RR DATE FORMAT AND THE YY FORMAT
No instructor notes for this slide**Slide 20: Tell Me / Show Me – A Few Simple Rules**
No instructor notes for this slide**Slide 21: Tell Me / Show Me – A Few Simple Rules (continued)**
No instructor notes for this slide**Slide 22: Tell Me / Show Me – A Few Simple Rules (continued) When I query my ...**

Use the following example to show the effects of the RRRR and YYYY formats:

```
SELECT last_name, TO_CHAR(hire_date, 'DD-Mon-YYYY')
FROM employees
```

```
WHERE hire_date < TO_DATE('01-Jan-90', 'DD-Mon-RR');
```

You'll get about three rows. Examine the WHERE clause. It asks to return only those hire dates less than (before) those of 01-Jan-90. The RR used in the format model here makes sure that the date is seen as 01-Jan-90. The three rows returned were those few people who were hired before 01-Jan -90.

Reformat the query:

```
SELECT last_name, TO_CHAR(hire_date, 'DD-Mon-YYYY')
FROM employees
```

```
WHERE hire_date < TO_DATE('01-Jan-90', 'DD-Mon-YY');
```

Now 20 rows were returned. Again, examine the format model in the WHERE clause.

Using the YY format causes the date to default to the current century, and since we are in the 2000s, 01-JAN-90 is seen as 01-JAN-2090. Everyone was hired before that date, right? Ask students to research the "Millennium Bug" or "Y2K" and the concern people had about how databases recorded their information.

Expect students to be a bit confused learning the RR and YY formats. Emphasize that the problem exists when character data such as 01-Jan-90 is being formatted into a default date format. Instead of using RR in the format model, YY is used. Because we are in the 21st century, the YY is interpreted in the current century as 2090. Provide additional practice by choosing a random date in either the 20th or 21st century, and ask students to provide the correct result.

Slide 20: Tell Me /Show Me - Terminology

CHAR-Used for text and character data of fixed length, including numbers, dashes, and special characters.

Fm-Used to remove padded blanks or to suppress leading zeros

Conversion function-Functions that convert a value from one data type to another.

NUMBER-Used to store variable-length numeric data.

VARCHAR2-Used for character data of variable length, including numbers, special characters, and dashes.

DATE-Used for date and time values.

TO_CHAR-Converts dates or numbers to character strings with optional formatting

RR date format-Century value depends on the specified year and the last two digits of the current year

TO_NUMBER-Converts a character string containing digits to a number with optional formatting

DD date format-Numeric day of the month

TO_DATE-Converts a character string representing a date to a date value with optional formatting

Slide 20: Summary – In this lesson you have learned to:

No instructor notes for this slide

Slide 21: Summary - Practice Guide

No instructor notes for this slide

SECTION 2 LESSON 2 – NULL Functions

Slide 1: NULL Functions

What to Watch For

Students may be suffering from "information overload" at this point. Keep encouraging practice and use the self-test software as a break and reinforcement. Ask students to generate query examples and challenge the class.

Connections

Many of the functions students learned in this lesson, they do every day.

Ask: Have you ever gone to a store to buy a something you were looking for only to find out that the store didn't have it? Have you been in the lunch line only to find out that your favorite item was sold out? Did you choose something else? If you've ever had this experience, you actually were doing a NVL2 or COALESCE function. If the first choice was null, choose the second or the third.

A lot of the power of SQL is to emulate decisions that we need to make every day. We need the date format changed to comply with how another country writes it. We need to issue payroll checks that have currency signs instead of just numbers.

Ask students to draw parallels between the functions and syntax they are learning and why the function was developed in the first place. After all, if no one ever needed to give a column another name in a report, would aliases ever have been developed?

Slide 2: What Will I Learn?

No instructor notes for this slide

Slide 3: Why Learn It?

No instructor notes for this slide

Slide 4: Tell Me / Show Me – Up to now, you have applied ...

No instructor notes for this slide

Slide 5: Tell Me / Show Me – SELECT

Students should be familiar with nested functions taught in mathematics. In the example, ask students to list the order of processing.

Step 1: The hire date is going to have six months added to it.

Step 2: The first Friday following the future day will be identified.

Step 3: The default date format will be formatted to read and display the Friday in a format similar to: Friday, December 18TH, 1987, and will appear in the output under the column name "Next Evaluation."

Slide 6: Tell Me / Show Me – At the beginning of the course..

No instructor notes for this slide

Slide 7: Tell Me / Show Me – NVL FUNCTION

The NVL function converts null valude to a date, a character...

No instructor notes for this slide

Slide 8: Tell Me / Show Me – NVL FUNCTION

The data types of the null value...

No instructor notes for this slide

Slide 9: Tell Me / Show Me – NVL functions can be used to ...

No instructor notes for this slide

Slide 10: Tell Me / Show Me – In the example, the auth_expense_amt...

No instructor notes for this slide

Slide 11: Tell Me / Show Me – NVL2 FUNCTION

Students have not formally studied data types. Tell them that LONG is variable-length character data up to 2 gigabytes in size.

Slide 12: Tell Me / Show Me – NVL2 FUNCTION (continued)

The NVL2 sample code illustrates a SQL statement returning an employee's salary and income. For employees who work on commission, the income is equivalent to the salary plus the salary multiplied by the commission percentage. For those who do not work on commission, the income is equivalent to the salary.

```
SELECT last_name, salary, NVL2(commission_pct, salary + (salary * commission_pct),  
salary) income  
FROM employees;
```

Slide 13: Tell Me / Show Me – NULLIF FUNCTION

In the example NULLIF compares the length of a partners first name with the length of the same partners last name, and if they are the same length then returns NULL and if they are different, then it returns the value of the first expression, in this case, first name.

Slide 14: Tell Me / Show Me – COALESCE FUNCTION

No instructor notes for this slide

Slide 15: Tell Me / Show Me – COALESCE FUNCTION (continued)

No instructor notes for this slide

Slide 16: Tell Me /Show Me – Terminology

NVL-Converts nulls to an actual value

COALESCE>Returns the first non-null expression in the list

NVL2-Examines the first expression; if the first expression is not null, it returns the second expression; if the first expression is null, it returns the third expression

NULLIF-Compares two expressions; if they are equal, the function returns null; if they are not equal, the function returns the first expression

Slide 17: Summary – In this lesson you have learned to:

No instructor notes for this slide

Slide 18: Summary - Practice Guide
No instructor notes for this slide

SECTION 2 LESSON 3 – Conditional Expressions

Slide 1: Conditional Expressions

What to Watch For

The CASE and DECODE functions need practice. Remind students to think of these functions as IF-THEN-ELSE decision operations. Read the syntax to them in this way.

Connections

Other programming languages use similar IF-ELSE or IF-THEN-ELSE processes. Understanding their function and use will make learning other programming languages easier.

Slide 2: What Will I Learn?

No instructor notes for this slide

Slide 3: Why Learn It?

No instructor notes for this slide

Slide 4: Why Learn It? (continued)

No instructor notes for this slide

Slide 5: Tell Me / Show Me – CONDITIONAL EXPRESSIONS

No instructor notes for this slide

Slide 6: Tell Me / Show Me – The syntax for CASE is:

No instructor notes for this slide

Slide 7: Tell Me / Show Me – DECODE Expression

Students may have difficulty distinguishing between CASE and DECODE. The CASE expression is a more flexible version of the DECODE function. The CASE expression complies with ANSI SQL; DECODE is specific to Oracle syntax.

Slide 8: Tell Me / Show Me – DECODE Expression (continued)

No instructor notes for this slide

Slide 9: Tell Me /Show Me – Terminology

DECODE-Compares an expression to each of the search values

Conditional expression-An if-then-else expression whose value depends on the truth-value of a Boolean expression.

CASE-Implements conditional processing within a SQL statement; it meets the ANSI standard

Slide 10: Summary – In this lesson you have learned to:

No instructor notes for this slide

Slide 11: Summary - Practice Guide
No instructor notes for this slide

Notes For Practice Activities:

Conversion Functions S02 L01

Vocabulary

<u>CHAR</u>	Used for text and character data of fixed length, including numbers, dashes, and special characters.
<u>fm</u>	Used to remove padded blanks or to suppress leading zeros
<u>Conversion function</u>	Functions that convert a value from one data type to another.
<u>NUMBER</u>	Used to store variable-length numeric data.
<u>VARCHAR2</u>	Used for character data of variable length, including numbers, special characters, and dashes.
<u>DATE</u>	Used for date and time values.
<u>TO_CHAR</u>	Converts dates or numbers to character strings with optional formatting
<u>RR date format</u>	Century value depends on the specified year and the last two digits of the current year
<u>TO_NUMBER</u>	Converts a character string containing digits to a number with optional formatting
<u>DD date format</u>	Numeric day of the month
<u>TO_DATE</u>	Converts a character string representing a date to a date value with optional formatting

Try It / Solve It

1. SELECT last_name, TO_CHAR(birthdate, 'Month fmDD, RRRR')AS "Birthday"
FROM f_staffs;
2. SELECT TO_DATE('January 3, 2004', 'Month dd, YYYY')as "Date"
FROM DUAL;

3. SELECT 'The promotion began on the ' || TO_CHAR(start_date, 'ddspth "of" Month YYYY')as "Date"
FROM f_promotional_menus
WHERE code = 110;

4. SELECT 'Today is the ' ||TO_CHAR(SYSDATE, 'Ddspth "of" Month, Yyyysp')
FROM DUAL;

5. SELECT id, first_name, TO_CHAR(salary, '\$999.99')
FROM f_staffs;

6. SELECT first_name, last_name, TO_CHAR(salary, '\$99,999.99'), TO_CHAR(salary +
2000, '\$99,999.99') AS "New Salary"
FROM f_staffs;

7. SELECT TO_CHAR(start_date, 'Day Month ddth') AS "Valentine's"
FROM f_promotional_menus
WHERE code = 110;

8. SELECT TO_CHAR(TO_DATE('25-DEC-04','dd-MON-yy'),'Month ddth,
YYYY')Convert,TO_CHAR(TO_DATE('25-DEC-04','dd-MON-yy'),'MONTH DDth,
YYYY')Convert,TO_CHAR(TO_DATE('25-DEC-04','dd-MON-yy'),'month ddth,
YYYY')AS Convert
FROM DUAL;

9. SELECT TO_CHAR(low_range, '\$99999.99')AS LOW, TO_CHAR(high_range,
'\$99999.99')AS HIGH
FROM d_packages;

10. SELECT TO_DATE('JUNE192004','fxMONTHDDYYYY') AS BIRTHDAY
FROM DUAL;

A common error when formatting dates is to omit the single quotation mark. The error message is missing a right parenthesis.

11. Implicit data type conversion occurs when the Oracle server internally converts data types eg Varchar2 to number. Explicit data conversion is when the a specific function is used to convert one data type to another eg TO_DATE
(actual examples may vary)

12. Data type conversions offer the facility to display numbers as local currency, format dates in a variety of formats, display time to the second, and keep track of what century a date refers to.

NULL Functions S02 L02

Vocabulary

<u>NVL</u>	Converts nulls to an actual value
<u>COALESCE</u>	Returns the first non-null expression in the list
<u>NVL2</u>	Examines the first expression; if the first expression is not null, it returns the second expression; if the first expression is null, it returns the third expression
<u>NULLIF</u>	Compares two expressions; if they are equal, the function returns null; if they are not equal, the function returns the first expression

Try It / Solve It

Encourage the students to use aliases to make the output more readable.

1. SELECT name, NVL2(end_date,'end in two weeks', SYSDATE)AS Promotion
FROM f_promotional_menus;
2. SELECT last_name,NVL(overtime_rate,0)AS "Overtime Status"
FROM f_staffs;
3. SELECT last_name, TO_CHAR(NVL(overtime_rate,5.00), '\$9999.99')AS "Overtime Rate"
FROM f_staffs;
4. SELECT last_name, NVL(manager_id, 9999)
FROM f_staffs;
5. c
- 6.

LAST_NAME
DOE
MILLER
TUTTLE

7. Part 1: SELECT first_name||' '||last_name "Name", to_char(hire_date,'Month')
"Anniversary Month"
FROM employees;

Part 2: SELECT first_name||' '||last_name "Name",
NULLIF(to_char(hire_date,'Month'),'September') "Anniversary Month"
FROM employees;

8. SELECT first_name, NVL(specialty,'No Specialty')AS Specialty
FROM d_partners;

Extension:

Show/discuss Demo: The Basics of Using Functions

Conditional Expressions S02 L03

Vocabulary

<u>DECODE</u>	Compares an expression to each of the search values
<u>Conditional expression</u>	An if-then-else expression whose value depends on the truth-value of a Boolean expression.
<u>CASE</u>	Implements conditional processing within a SQL statement; it meets the ANSI standard

Try It / Solve It

1. SELECT id, title, duration,
DECODE(duration, '2 min', 'shortest', '10 min', 'longest')
AS "Play Times"
FROM d_songs;

2. SELECT last_name, department_id, salary,
CASE department_id WHEN 10 THEN salary*1.25
WHEN 90 THEN salary*1.50
WHEN 130 THEN salary*1.75
ELSE salary END As "No Change"
FROM employees;

3. SELECT first_name, last_name, manager_id, commission_pct,
COALESCE(manager_id, commission_pct, 99999) AS "Review"
FROM employees
WHERE department_id IN(80,90);