Database Programming with SQL
5-1: Conversion Functions
Practice Activities

Objectives

- Provide an example of an explicit data-type conversion and an implicit data-type conversion
- Explain why it is important, from a business perspective, for a language to have built-in data-conversion capabilities
- Construct a SQL query that correctly applies TO_CHAR, TO_NUMBER, and TO_DATE single-row functions to produce a desired result
- Apply the appropriate date and/or character format model to produce a desired output
- Explain and apply the use of YY and RR to return the correct year as stored in the database

Vocabulary

Identify the vocabulary word for each definition below.

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 Functions	Functions that convert a value from one datatype 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	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	Numeric day of the month
TO_DATE	Converts a character string representing a date to a date value with optional formatting

Try It / Solve It

In each of the following exercises, feel free to use labels for the converted column to make the output more readable.

1. List the last names and birthdays of Global Fast Food Employees. Convert the birth dates to character data in the Month DD, YYYY format. Suppress any leading zeros.

```
SELECT last_name as "Last Name",
       TO_CHAR(birthdate, 'fmMonth DD, YYYY') AS "Birthday"
FROM f_staffs;
```

2. Convert January 3, 04, to the default date format 03-Jan-2004.

```
SELECT TO_CHAR(TO_DATE('January 3, 04', 'Month DD, RR'), 'DD-Mon-YYYY') AS "Formatted Date"
FROM dual;
```

3. Format a query from the Global Fast Foods f_promotional_menus table to print out the start_date of promotional code 110 as: The promotion began on the tenth of February 2004.

```
SELECT 'The promotion began on the ' ||
       LOWER(TO_CHAR(start_date, 'fmDDthsp')) || ' of ' ||
       TO_CHAR(start_date, 'Month YYYY') AS "Promotion Start"
FROM f_promotional_menus
WHERE code = 110;
```

4. Convert today's date to a format such as: "Today is the Twentieth of March, Two Thousand Four"

```
SELECT 'Today is the ' ||
       INITCAP(TO_CHAR(SYSDATE, 'fmDDthsp')) || ' of ' ||
       TO_CHAR(SYSDATE, 'Month, Year') AS "Today"
FROM dual;
```

5. List the ID, name, and salary for all Global Fast Foods employees. Display salary with a $ sign and two decimal places.

```
SELECT employee_id,
       (first_name || ' ' || last_name) name,
       TO_CHAR(salary, '$99999.99') salary
FROM employees;
```

6. Ellen Abel is an employee who has received a $2,000 raise. Display her first name and last name, her current salary, and her new salary. Display both salaries with a $ and two decimal places. Label her new salary column AS New Salary.

```
SELECT first_name as "First Name", last_name as "Last Name",
      TO_CHAR(salary, '$99999.99') AS "Current Salary",
      TO_CHAR(salary + 2000, '$99999.99') AS "New Salary"
FROM employees
WHERE first_name = 'Ellen' AND last_name = 'Abel';
```

7. On what day of the week and date did Global Fast Foods' promotional code 110 Valentine's Special begin?

```
SELECT initcap(TO_CHAR(start_date, 'fmDay, fmDDth Month YYYY')) AS "Promo Start"
FROM f_promotional_menus
WHERE code = 110;
```

8. Create one query that will convert 25-Dec-2004 into each of the following (you will have to convert 25-Dec-2004 to a date and then to character data):

December 25th, 2004
DECEMBER 25TH, 2004
25th december, 2004

```
SELECT initcap(TO_CHAR(TO_DATE('25-Dec-2004', 'DD-Mon-YYYY'), 'Month DDth, YYYY')) AS "Format1",
       TO_CHAR(TO_DATE('25-Dec-2004', 'DD-Mon-YYYY'), 'fmMONTH DDth, YYYY') AS "Format2",
       lower(TO_CHAR(TO_DATE('25-Dec-2004', 'DD-Mon-YYYY'), 'fmDDth Month, YYYY')) AS "Format3"
FROM dual;
```

9. Create a query that will format the DJs on Demand d_packages columns, low-range and high-range package costs, in the format $2500.00.

```
SELECT TO_CHAR(low_range, '$99999.99') AS "Low Range",
       TO_CHAR(high_range, '$99999.99') AS "High Range"
FROM d_packages;
```

10. Convert JUNE192004 to a date using the fx format model.

```
SELECT TO_DATE('JUNE192004', 'fxMonthDDYYYY') AS "Formatted Date"
FROM dual;
```

Implicit:

```
SELECT employee_id, salary + '500'
FROM employees;
```

Explicit:

```
SELECT employee_id, salary +
TO_NUMBER('500')
FROM employees;
```

11. What is the distinction between implicit and explicit datatype conversion? Give an example of each.

```
Implicit conversion happens automatically, like when a VARCHAR2 is implicitly converted to a NUMBER.
Explicit conversion requires functions like TO_CHAR, TO_NUMBER, and TO_DATE.
```

12. Why is it important from a business perspective to have datatype conversions?

```
Datatype conversions allow flexibility in how data is presented.
```

Database Programming with SQL
5-2: NULL Functions
Practice Activities

Objectives

- Demonstrate and explain the evaluation of a nested function
- List at least four general functions that work with any data type and relate to handling null values
- Explain the use of the COALESCE and the NVL functions
- Explain the use of general functions to deal with null values in data
- Construct and execute a SQL query that correctly applies NVL, NVL2, NULLIF, and COALESCE single-row functions

Vocabulary

Identify the vocabulary word for each definition below.

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

Try It / Solve It

Use aliases to make the output more readable.

1. Create a report that shows the Global Fast Foods promotional name, start date, and end date from the f_promotional_menus table. If there is an end date, temporarily replace it with “end in two weeks.” If there is no end date, replace it with today’s date.

```
SELECT name, start_date,
       NVL2(end_date, 'End in two weeks', TO_CHAR(SYSDATE, 'DD-Mon-YYYY')) AS end_date
FROM f_promotional_menus;
```

2. Not all Global Fast Foods staff members receive overtime pay. Instead of displaying a null value for these employees, replace null with zero. Include the employee’s last name and overtime rate in the output. Label the overtime rate as “Overtime Status”.

```
SELECT last_name,
       NVL(overtime_rate, 0) AS "Overtime Status"
FROM f_staffs;
```

3. The manager of Global Fast Foods has decided to give all staff who currently do not earn overtime an overtime rate of $5.00. Construct a query that displays the last names and the overtime rate for each staff member, substituting $5.00 for each null overtime value.

```
SELECT last_name AS "Last Name",
       NVL(overtime_rate, 5.00) AS "Overtime Rate"
FROM f_staffs;
```

4. Not all Global Fast Foods staff members have a manager. Create a query that displays the employee last name and 9999 in the manager ID column for these employees.

```
SELECT last_name,
       NVL(manager_id, 9999) AS manager_id
FROM f_staffs;
```

5. Which statement(s) below will return null if the value of v_sal is 50?
   a. SELECT nvl(v_sal, 50) FROM emp;
   b. SELECT nvl2(v_sal, 50) FROM emp;
   c. SELECT nullif(v_sal, 50) FROM emp;
   d. SELECT coalesce (v_sal, Null, 50) FROM emp;

6. What does this query on the Global Fast Foods table return?

   SELECT COALESCE(last_name, to_char(manager_id)) as NAME
   FROM f_staffs;

```
This query returns:

If last_name is not null, it returns the last_name.
If last_name is null, it converts the manager_id to a character string and returns that.
```

7. a. Create a report listing the first and last names and month of hire for all employees in the EMPLOYEES table (use TO_CHAR to convert hire_date to display the month).

```
SELECT first_name as "First Name", last_name as "Last Name",
       TO_CHAR(hire_date, 'Month') AS "Month of Hire"
FROM employees;
```

   b. Modify the report to display null if the month of hire is September. Use the NULLIF function.

```
SELECT first_name as "First Name", last_name as "Last Name",
       NULLIF(TO_CHAR(hire_date, 'Month'), 'September') AS "Month of Hire"
FROM employees;
```

8. For all null values in the specialty column in the DJs on Demand d_partners table, substitute “No Specialty.” Show the first name and specialty from the d_partners table.

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

Database Programming with SQL
5-3: Conditional Expressions
Practice Activities

Objectives

- Compare and contrast the DECODE and CASE functions
- Construct and execute a SQL query that correctly uses the DECODE and CASE functions
- Construct and execute two methods for implementing IF-THEN-ELSE conditional logic

Vocabulary

Identify the vocabulary word for each definition below.

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

Try It / Solve It

1. From the DJs on Demand d_songs table, create a query that replaces the 2-minute songs with “shortest” and the 10-minute songs with “longest”. Label the output column “Play Times”.

```
SELECT title,
DECODE(TO_NUMBER(REGEXP_SUBSTR(duration, '^\d+')), 2, 'Shortest', 10, 'Longest', duration) AS "Play Times"
FROM d_songs;
```

2. Use the Oracle database employees table and CASE expression to decode the department id. Display the department id, last name, salary, and a column called “New Salary” whose value is based on the following conditions:

If the department id is 10 then 1.25 * salary If the department id is 90 then 1.5 * salary If the department id is 130 then 1.75 * salary Otherwise, display the old salary.

```
SELECT department_id, last_name, salary,
       CASE
         WHEN department_id = 10 THEN salary * 1.25
         WHEN department_id = 90 THEN salary * 1.5
         WHEN department_id = 130 THEN salary * 1.75
         ELSE salary
       END as "NEW_SALARY"
FROM employees;
```

3. Display the first name, last name, manager ID, and commission percentage of all employees in departments 80 and 90. In a 5th column called “Review”, again display the manager ID. If they don’t have a manager, display the commission percentage. If they don’t have a commission, display 99999.

```
SELECT first_name, last_name, manager_id, commission_pct,
       CASE
         WHEN manager_id IS NULL THEN NVL(commission_pct, 99999)
         ELSE manager_id
       END AS "REVIEW"
FROM employees
WHERE department_id IN (80, 90);
```