Ryan Li
Oracle - Programming Section 4

Database Programming with SQL 4-1: Case and Character Manipulation

DUAL	Dummy table used to view results from functions and calculations
Data Format	The arrangement of data for storage or display.
INTICAP	Converts alpha character values to uppercase for the first letter of each word, all other letters in lowercas
Character Manipulation Function	Functions that accept character data as input and can return both character and numeric values
TRIM	Removes all specified characters from either the beginning or the ending of a string
Operator	A symbol that represents a quantity or a relationship between quantities
Single row functions	Functions that operate on single rows only and return one result per row
UPPER	Converts alpha characters to uppercase
input	Raw data entered into the computer
CONCAT	Concatenates the first character value to the second character value; equivalent to concatenation operator (||)
Output	Date that is processed into information
LOWER	Converts alpha character values to lowercase
LPAD	Pads the left side of a character, resulting in a right-justified value
SUBSTR	Returns specific characters from character value starting at a specific character position and going specified character positions long
REPLACE	Replaces a sequence of characters in a

	string with another set of characters
INSTR	Returns the numeric position of a named string
LENGTH	Returns the number of characters in the expression
RPAD	Pads the right-hand side of a character, resulting in a left- justified value

Try It / Solve It

1.
Using the three separate words “Oracle,” “Internet,” and “Academy,” use one command to produce the following output:

The Best Class
Oracle Internet Academy

- **SELECT CONCAT(‘Oracle’, CONCAT(‘Internet’, ‘Academy’)) AS “The Best Class”
FROM dual;**

2. Use the string “Oracle Internet Academy” to produce the following output:

The Net
net

- **SELECT SUBSTR(‘Oracle Internet Academy’, INSTR(‘Oracle Internet Academy’, ‘net’), 3) AS “The Net”
FROM dual;**

3. What is the length of the string “Oracle Internet Academy”?

- **SELECT LENGTH(“Oracle Internet Academy”) AS “length”
FROM dual;**

4.What’s the position of “I” in “Oracle Internet Academy”?

- **SELCT INSTR(“Oracle Internet Academy”, “I”) AS “position”
FROM dual;**

5.Starting with the string “Oracle Internet Academy”, pad the string to create
****Oracle****Internet****Academy****

- **SELECT CONCAT(CONCAT(‘****’,”Oracle”),CONCAT(‘****’,CONCAT(‘Internet’,**
  CONCAT(‘**’, ‘Academy***’))))AS padded_string**
  FROM dual;

6. Starting with the string “Oracle Internet Academy”, pad the string to produce:
Oracle$$$Internet$$$Academy

- **SELECT REPLACE(‘Oracle Internet Academy’,”,$$$) AS padded_string**
  FROM dual;

7.Using the string ‘Oracle Internet Academy’, produce the output shown using the REPLACE function. List the order date and the order total from the Global Fast Foods F_ORDERS table. Name the order total as TOTAL, and fill in the empty spaces to the left of the order total with $.

- **SELECT REPLACE(‘Oracle Internet Academy’, ‘Internet’,**
  ‘2013-2014’) AS “The Best Class”
  FROM dual;

8. List the order date and the order total from the Global Fast Foods F_ORDERS table. Name the order total as TOTAL, and fill in the empty spaces to the left of the order total with $

- **SELECT order_date, CONCAT(‘$’, RPAD(TO_CHAR(order_total),10, ‘ ’)) AS TOTAL**
  FROM F_ORDERS;

9. Write a query that will output a column called “ADDRESS” which has the following information:
ZOE TWEE 1009 OLIVER AVENUE BOSTON, MA 12889. Use the Global Fast Foods F_CUSTOMERS table.

- **SELECT ‘ZOE TWEE 1009 OLIVER AVENUE BOSTON, MA 12889’ AS ADDRESS**
  FROM F_CUSTOMERS;

10. Write a query to return the first character of the first name concatenated to the last_name, the salary, and the department id for employees working in department 20.
Give the first expression an alias of Name. Use the EMPLOYEES table. Change the query to use a substitution variable instead of the hard coded value 20 for department id. Run the query for department 30 and 50 without changing the original where-clause in your statement.

- **SELECT SUBSTR (first_name, 1, 1) || last_name AS Name, salary, department_id,**
  FROM EMPLOYEES;

11. Using a substitution variable for the department name, write a query listing department id, department name, and location id for departments located in the_department_of_your_choice.

Use the DEPARTMENTS table. Note: All substitution variables in OAE are treated as character strings, so no quotes (‘ ‘) are needed.

- **SELECT department_id, department_name, location_id**

```
FROM DEPARTMENTS
WHERE department_name = :dept_name;
```

12. Write a query that returns all the employee data depending on the month of their hire date. Use the EMPLOYEES table. The statement should return the month part of the hiredate which is then compared to an abbreviated month (JAN, FEB, MAR) passed into the query via a substitution variable.

-
  ```
  SELECT *
  FROM EMPLOYEES
  WHERE TO_CHAR(hire_date, ‘MON’) = UPPER(:month_abbrev);
  ```

Database Programming with SQL 4-2: Number Functions

TRUNC	Used to terminate the column, expression, or value to a specified number of decimal places
Number functions	These functions accept numeric input and return values
MOD	Returns the remainder of a division
ROUND	Rounds the column, expression, or value to a set number of decimal places

1. Display Oracle database employee last_name and salary for employee_ids between 100 and 102. Include a third column that divides each salary by 1.55 and rounds the result to two decimal places

   -
     ```
     SELECT last_name, salary, ROUND(salary/1.55) as adjusted_salary
     FROM EMPLOYEES
     WHERE employee_id BETWEEN 100 and 102;
     ```

2. Display employee last_name and salary for those employees who work in department 80. Give each of them a raise of 5.333% and truncate the result to two decimal places.

   -
     ```
     SELECT last_name, TRUNC (salary * 1.05333, 2) AS new_salary
     FROM EMPLOYEES
     WHERE department_id = 80
     ```

3. Use a MOD number function to determine whether 38873 is an even number or an odd number.

   -
     ```
     SELECT
             CASE
                     WHEN MOD(38873, 2) = 0 THEN ‘Even’
                     ELSE ‘Odd’
             END AS Number_Type;
     FROM dual;
     ```

4. Use the DUAL table to process the following numbers:
    a. 845.553 - round to one decimal place
    b. 30695.348 - round to two decimal places
    c. 30695.348 - round to -2 decimal places
    d. 2.3454 - truncate the 454 from the decimal place
    - **SELECT**
      ROUND(845.553, 1) AS rounded_one
      ROUND(30695.348, 2) AS rounded_two
      ROUND(30695.348, -2) AS rounded_three
      TRUNC(2.3454, 3) AS truncated_value
      FROM dual;

5. Divide each employee's salary by 3. Display only those employees' last names and salaries who earn a salary that is a multiple of 3.
    - **SELECT last_name, salary**
      FROM EMPLOYEES
      WHERE MOD(salary, 3) = 0;

6. Divide 34 by 8. Show only the remainder of the division. Name the output as EXAMPLE.
    - **SELECT MODE(34,8) AS EXAMPLE**
      FROM dual;
7. How would you like your paycheck – rounded or truncated? What if your paycheck was calculated to be $565.784 for the week, but you noticed that it was issued for $565.78. The loss of .004 cent would probably make very little difference to you. However, what if this was done to one thousand people, one hundred thousand people, or one million people! Would it make a difference then? How much of a difference?
    a. **In this example, rounding or truncating wouldn't affect the paycheck of $565.784 for an individual because both methods would result in $565.78. The third decimal is too small to change the second decimal, so the amount remains the same whether it's rounded down or truncated. However, when applied to larger groups, the cumulative effect becomes significant. If 1,000 people lose $0.004 each, the total shortfall would be $4. For 100,000 people, it would amount to $400, and for a million people, the difference would be $4,000. So, while it may seem insignificant for one person, the financial impact grows when scaled to larger numbers.**

Database Programming with SQL 4-3: Date Functions

SYSDATE	A function that returns the current date and time of the database
ADD_MONTHS	Add calendar months to date
LAST_DAY	Last day of the month
NEXT_DAY	Next day of the date specified

MONTHS_BETWEEN	Number of months between due dates

1. For DJs on Demand, display the number of months between the event_date of the Vigil wedding and today’s date. Round to the nearest month.

   -
     ```
     SELECT ROUND(MONTHS_BETWEEN(SYSDATE), event_date)) AS
     Months_Between
     FROM dj_list
     WHERE event_name = ‘Vigil wedding)
     ```

2. Display the days between the start of last summer’s school vacation break and the day school started this year. Assume 30.5 days per month. Name the output “Days

   -
     ```
     SELECT (TO_DATE('2024-09-01', 'YYYY-MM-DD') -
     TO_DATE('2024-06-15', 'YYYY-MM-DD')) AS Days
     FROM dual;
     ```

3. Display the days between January 1 and December 31

   -
     ```
     SELECT (TO_DATE('2024-12-31', 'YYYY-MM-DD') -
     TO_DATE('2024-01-01', 'YYYY-MM-DD')) AS Days
     FROM dual;
     ```

4. Using one statement, round today's date to the nearest month and nearest year, and truncate it to the nearest month and nearest year. Use an alias for each column.

   -
     ```
     SELECT ROUND(SYSDATE, ‘MM’) AS Rounded_Month,
     TRUNC(SYSDATE, 'MM') AS Truncated_Month,
     ROUND(SYSDATE, 'YYYY') AS Rounded_Year,
     TRUNC(SYSDATE, 'YYYY') AS Truncated_Year
     FROM dual;
     ```

5. What is the last day of the month for June 2005? Use an alias for the output

   -
     ```
     SELECT LAST_DAY(TO_DATE('2005-06-01', 'YYYY-MM-DD')) AS
     Last_Day_June_2005
     FROM dual;
     ```
6. Display the number of years between the Global Fast Foods employee Bob Miller’s birthday and today. Round to the nearest year

   -
     ```
     SELECT ROUND(MONTHS_BETWEEN(SYSDATE, TO_DATE(('1985-05-10',
     'YYYY-MM-DD')) / 12) AS Years_Between
     FROM dual;
     ```
7. Your next appointment with the dentist is six months from today. On what day will you go to the dentist? Name the output, “Appointment.”

   -
     ```
     SELECT ADD_MONTHS(SYSDATE, 6) AS Appointment
     FROM dual;
     ```
8. The teacher said you have until the last day of this month to turn in your research paper. What day will this be? Name the output, “Deadline.

   -
     ```
     SELECT LAST_DAY(SYSDATE) AS Deadline
     FROM dual;
     ```
9. How many months between your birthday this year and January 1 next year?

   -
     ```
     SELECT MONTHS_BETWEEN(TO_DATE('2025-01-01',
     'YYYY-MM-DD'), TO_DATE('2024-11-20', 'YYYY-MM-DD')) AS
     Months_Between
     FROM dual;
     ```

10. What’s the date of the next Friday after your birthday this year? Name the output, “first Friday”.
    - **SELECT NEXT_DAY(TO_DATE('2024-11-20', 'YYYY-MM-DD'), 'FRIDAY') AS First_Friday**
      FROM dual;
11. Name a date function that will return a number
    - **MONTHS_BETWEEN**
12. Name a date function that will return a date
    - **LAST_DAY**
      NEXT_DAY
13. Give one example of why it is important for businesses to be able to manipulate date Data?
    - **It is important for businesses to manipulate date data bcause it can assist with business operations when it comes to scheduling or reviewing incoming business needs.**

<u>Extension Excercises</u>

1. Using DUAL, write a statement that will convert 86.678 to 86.68.
    - **SELECT ROUND(86.678, 2) AS Rounded_Value FROM dual;**
2. Write a statement that will display the DJs on Demand CD titles for cd_numbers 90 and 91 in uppercase in a column headed “DJs on Demand Collections.”
    - **SELECT UPPER(cd_title) AS "DJs on Demand Collections"**
      FROM djs_on_demand
      WHERE cd_number IN (90, 91);
3. Write a statement that will create computer usernames for the DJs on Demand partners. The usernames will be the lowercase letters of the last name + the uppercase first letter in the first name. Title the column “User Passwords.” For example, Mary Smythers would be smythersM.
    - **SELECT LOWER(last_name) || UPPER(SUBSTR(first_name, 1, 1)) AS "User Passwords"**
      FROM dj_list;
4. Write a statement that will convert “It’s a small world” to “HELLO WORLD.”
    - **SELECT REPLACE (‘it's a small world”, “hello world”) AS modified_string**
      FROM dual

5. Write a statement that will remove the “fiddle” from “fiddledeedee” and the “dum” from “fiddledeedum.” Display the result “fiddledeedeedee” in a column with the heading “Nonsense.”
    - **SELECT REPLACE(REPLACE('fiddledeedum', 'dum', 'dee'), 'fiddle', '') AS Nonsense FROM dual;**
6. Replace every “i” in Mississippi with “$.”
    - **SELECT REPLACE('Mississippi', 'i', '$') AS Modified_String FROM dual;**
7. Using DUAL, convert 5332.342 to 5300.
    - **SELECT TRUNC(5332.342, -2) AS Rounded_Value FROM dual;**
8. Using DUAL, convert 3.14159 to 3.14.
    - **SELECT ROUND(3.14159, 2) AS Rounded_Value FROM dual;**
9. Using DUAL, convert 73.892 to 73.8.
    - **SELECT ROUND(73.892, 1) AS Rounded_Value FROM dual;**
10. What is the next Friday six months from now? Label the column “Future.”
    - **SELECT NEXT_DAY(ADD_MONTHS(SYSDATE, 6), 'FRIDAY') AS**

```
Future
FROM dual;
```

11. What is the date 10 years from now? Label the column “Future.”

    -
      ```
      SELECT SYSDATE + INTERVAL '10' YEAR AS Future FROM dual;
      ```
12. Leap years occur every four years. Remember, 2004 was a leap year. Now create a function that will show the date of the next leap year as 29-Feb-2008. Label the column “Future.”

    -
      ```
      CREATE OR REPLACE FUNCTION next_leap_year (start_year IN
      NUMBER) RETURN DATE AS future_leap_year NUMBER; BEGIN -- Find
      the next leap year future_leap_year := start_year + (4 - MOD(start_year, 4));
      -- Ensure it is actually a leap year by checking divisibility rules IF
      (MOD(future_leap_year, 100) = 0 AND MOD(future_leap_year, 400) != 0)
      THEN future_leap_year := future_leap_year + 4; END IF; -- Return
      February 29th of the next leap year RETURN TO_DATE('29-FEB-' ||
      future_leap_year, 'DD-MON-YYYY'); END;

      SELECT next_leap_year(2004) AS Future FROM dual;
      ```

13. Write a statement that will find any of the DJs on Demand CD themes that have an “ie” in their names.

    -
      ```
      SELECT DJ_names
      FROM DJ_list
      WHERE DJ_names LIKE '%ie%';
      ```
14. Write a statement that will return only the DJs on Demand CDs with years greater than 2000 but less than 2003. Display both the title and year.

    -
      ```
      SELECT DJ_name, CD_years
      FROM DJ_list
      WHERE CD_years > 2000 AND CD_years < 2003;
      ```
15. Write a statement that will return the Oracle database employee’s employee ID and his starting hire dates between January 1, 1997, and today. Display the result ordered from most recently hired to the oldest.

    -
      ```
      SELECT employee_id, hire_date
      FROM employees
      WHERE MONTHS_BETWEEN TO_DATE('01-JAN-1997',
      'DD-MON-YYYY') AND SYSDATE
      ORDER BY hire_date DESC;
      ```