**TOPIC 03: Single-row Functions**

**Exercises**

**\*\*\* This exercise is performed on HR Schema (HR database) and on DUAL table\*\*\***

**\*\* This exercise may include some of the topics examined previously\*\***

1. Write a query in SQL to display the current system date (table: dual).

**SELECT SYSDATE FROM dual;**

1. Write a query in SQL to extract day, month and year from sysdate (table: dual).

**SELECT EXTRACT (DAY FROM SYSDATE) AS Day, EXTRACT (MONTH FROM SYSDATE) AS Month, EXTRACT (YEAR FROM SYSDATE) AS Year FROM dual;**

1. Write a query in SQL to show previous and next month using ADD\_MONTHS function (table:dual).

**SELECT EXTRACT (MONTH FROM ADD\_MONTHS(SYSDATE,-1)) AS "Last Month",EXTRACT (MONTH FROM ADD\_MONTHS(SYSDATE,1)) AS "Next Month"**

**FROM dual;**

1. Write a query in SQL to show last day of the current month, as well as the first day of the following month using function LAST\_DAY (table: dual).

**SELECT**

**LAST\_DAY(SYSDATE) "Last Day",**

**LAST\_DAY(SYSDATE) +1 "First Day"**

**FROM DUAL;**

1. Write a query in SQL to show number of days before the end of the month using function LAST\_DAY (table: dual).

**SELECT SYSDATE,**

**LAST\_DAY(SYSDATE) "Last",**

**LAST\_DAY(SYSDATE) - SYSDATE "Days Left"**

**FROM DUAL;**

1. Write a query in SQL to calculate MONTHS\_BETWEEN 31 JAN 2017 and 30 JAN 2018 (table: dual).

**SELECT MONTHS\_BETWEEN**

**(TO\_DATE('01-30-2020','MM-DD-YYYY'),**

**TO\_DATE('01-30-2010','MM-DD-YYYY') ) "Months"**

**FROM DUAL;**

1. Write a query in SQL to show the number of months the employee is working for a company (table: employees).

**SELECT ROUND(MONTHS\_BETWEEN (SYSDATE,hire\_date)) FROM employees;**

1. Write a query in SQL to show the next Monday (table: dual).

**SELECT NEXT\_DAY (SYSDATE, 'MONDAY') FROM dual;**

1. Write a query in SQL to show date 25-DEC-2017, but rounded to the years (table: dual).

**SELECT EXTRACT (YEAR FROM ROUND(TO\_DATE('25-Dec-2017', 'DD-Mon-YYYY'),'YEAR')) AS "Rounded year" FROM dual;**

1. Write a query in SQL to truncate date 25-DEC-2017 to year component (table: dual).

**SELECT EXTRACT (YEAR FROM TRUNC (TO\_DATE('25-Dec-2017', 'DD-Mon-YYYY'), 'YEAR')) AS "Truncated year" FROM dual;**

1. You are required to retrieve a list of FIRST\_NAME and LAST\_NAME values and formal names for employees where the combined length of FIRST\_NAME and LAST\_NAME exceeds 15 characters. The formal name is constructed by concatenating the first character in the FIRST\_NAME field with a space and the first 14 characters of the LAST\_NAME field to return a string that is 16 characters long (table: employees).

**SELECT first\_name, last\_name, SUBSTR(first\_name,1,1) || ' ' || SUBSTR(last\_name,1,14) AS "formal name"**

**FROM employees**

**WHERE LENGTH(first\_name || last\_name) > 15;**

1. You are required to obtain a list of EMPLOYEE\_ID, LAST\_NAME, and HIRE\_DATE values for the employees who have worked more than 100 months between the date they were hired and 01-JAN-2012 (table: employees).

**SELECT employee\_id, last\_name, hire\_date**

**FROM employees**

**WHERE MONTHS\_BETWEEN(TO\_DATE('01-Jan-2012','DD-Mon-YYYY'),hire\_date) > 100;**

1. You wish to retrieve the duration of employment in days for each employee (table: employees).

**SELECT ROUND (SYSDATE - hire\_date) AS "Days of employment"**

**FROM employees;**

1. You are tasked with identifying the date the end-of-year staff bonus will be paid. Bonuses are usually paid on the last Friday in December (table:dual).

**SELECT NEXT\_DAY (ADD\_MONTHS (TRUNC (SYSDATE, 'YEAR'), 12) - 8, 'FRIDAY')**

**FROM dual;**

1. Employees working in the IT department have moved to new offices and, although the last four digits of their phone numbers are the same, the set of the three digits 423 is changed to 623. A typical phone number of an IT staff member is 590.423.4567. You are required to provide a list of employees’ names with their old and new phone numbers.

**SELECT**

**first\_name || ' ' || last\_name AS "Full Name",**

**phone\_number AS "Old Phone Number",**

**REPLACE (phone\_number, '423', '623') AS "New Phone Number"**

**FROM**

**employees**

**WHERE**

**UPPER(job\_id) LIKE 'IT%';**