

EXERCISE 6 SINGLE ROW FUNCTIONS

NAME: VIDHYASHREE J

REG NO: 241801310

1. Write a query to display the current date. Label the column Date.

The screenshot shows the Oracle Database Express Edition interface. At the top, it says "ORACLE Database Express Edition". Below that, it says "User: SYSTEM". The breadcrumb navigation shows "Home > SQL > SQL Commands". There is a toolbar with "Autocommit" checked and a "Display" dropdown set to "10". The SQL editor contains the query: `SELECT SYSDATE AS "Date"`
`FROM DUAL;`. Below the editor, there are tabs for "Results", "Explain", "Describe", "Saved SQL", and "History". The "Results" tab is active, showing a table with one column "Date" and one row with the value "20-AUG-25". At the bottom, it says "1 rows returned in 0.00 seconds" and there is a "CSV Export" link.

Date
20-AUG-25

1 rows returned in 0.00 seconds [CSV Export](#)

2. The HR department needs a report to display the employee number, last name, salary, and increased by 15.5% (expressed as a whole number) for each employee. Label the column New Salary.

User: SYSTEM

Home > SQL > SQL Commands

☒ Autocommit Display 10

```
SELECT employee_id,  
       last_name,  
       salary,  
       ROUND(salary * 1.155) AS "New Salary"  
FROM employees;
```

Results Explain Describe Saved SQL History

EMPLOYEE_ID	LAST_NAME	SALARY	New Salary
1001	Smith	60000	69300
1002	Doe	55000	63525
1003	Johnson	52000	60060
1004	Brown	70000	80850

4 rows returned in 0.00 seconds

[CSV Export](#)

3. Modify your query lab_03_02.sql to add a column that subtracts the old salary from the new salary. Label the column Increase.

User: SYSTEM

Home > SQL > SQL Commands

☒ Autocommit Display 10 ▼

```
SELECT employee_id,
       last_name,
       salary,
       ROUND(salary * 1.155) AS "New Salary",
       ROUND(salary * 1.155) - salary AS "Increase"
FROM employees;
```

Results Explain Describe Saved SQL History

EMPLOYEE_ID	LAST_NAME	SALARY	New Salary	Increase
1001	Smith	60000	69300	9300
1002	Doe	55000	63525	8525
1003	Johnson	52000	60060	8060
1004	Brown	70000	80850	10850

4 rows returned in 0.00 seconds

[CSV Export](#)

4. Write a query that displays the last name (with the first letter uppercase and all other letters lowercase) and the length of the last name for all employees whose name starts with the letters J, A, or M. Give each column an appropriate label. Sort the results by the employees' last names.

ORACLE® Database Express Edition

User: SYSTEM

Home > SQL > **SQL Commands**

☒ Autocommit Display **10** ▼

```
SELECT INITCAP(last_name) AS "Last Name",  
       LENGTH(last_name) AS "Name Length"  
FROM employees  
WHERE UPPER(SUBSTR(last_name, 1, 1)) IN ('J', 'A', 'M')  
ORDER BY last_name;
```

Results Explain Describe Saved SQL History

Last Name	Name Length
Johnson	7

1 rows returned in 0.01 seconds

[CSV Export](#)

5. Rewrite the query so that the user is prompted to enter a letter that starts the last name. For example, if the user enters H when prompted for a letter, then the output should show all employees whose last name starts with the letter H.

User: SYSTEM

Home > SQL > SQL Commands

☒ Autocommit Display 10 ▼

```
SELECT INITCAP(last_name) AS "Last Name",  
       LENGTH(last_name) AS "Name Length"  
FROM employees  
WHERE UPPER(SUBSTR(last_name, 1, 1)) = UPPER('&enter_letter')  
ORDER BY last_name;
```

[Results](#) [Explain](#) [Describe](#) [Saved SQL](#) [History](#)

no data found

6. The HR department wants to find the length of employment for each employee. For each employee, display the last name and calculate the number of months between today and the date on which the employee was hired. Label the column MONTHS_WORKED. Order your results by the number of months employed. Round the number of months up to the closest whole number

ORACLE® Database Express Edition

User: SYSTEM

Home > SQL > **SQL Commands**

☒ Autocommit Display 10 ▼

```
SELECT last name,  
       CEIL(MONTHS_BETWEEN(SYSDATE, hire_date)) AS "MONTHS_WORKED"  
FROM employees  
ORDER BY "MONTHS_WORKED";
```

Results Explain Describe Saved SQL History

LAST_NAME	MONTHS_WORKED
Brown	92
Johnson	153
Doe	183
Smith	246

4 rows returned in 0.00 seconds

[CSV Export](#)

7. Create a report that produces the following for each employee: earns monthly but wants . Label the column Dream Salaries.

User: SYSTEM

Home > SQL > SQL Commands

☒ Autocommit Display 10 ▼

```
SELECT last_name || ' earns monthly but wants ' || TO_CHAR(salary * 3) AS "Dream Salaries"
FROM employees;
```

Results Explain Describe Saved SQL History

Dream Salaries
Smith earns monthly but wants 180000
Doe earns monthly but wants 165000
Johnson earns monthly but wants 156000
Brown earns monthly but wants 210000

4 rows returned in 0.00 seconds

[CSV Export](#)

8. Create a query to display the last name and salary for all employees. Format the salary to be 15 characters long, left-padded with the \$ symbol. Label the column SALARY.

User: SYSTEM

Home > SQL > SQL Commands

☒ Autocommit Display 10

```
SELECT last_name,  
       LPAD(TO_CHAR(salary, '$999,999,999.00'), 15, '$') AS SALARY  
FROM employees;
```

Results Explain Describe Saved SQL History

LAST_NAME	SALARY
Smith	\$60,000.0
Doe	\$55,000.0
Johnson	\$52,000.0
Brown	\$70,000.0

4 rows returned in 0.00 seconds

[CSV Export](#)

9. Display each employee's last name, hire date, and salary review date, which is the first Monday after six months of service. Label the column REVIEW. Format the dates to appear in the format similar to "Monday, the Thirty-First of July, 2000."

User: SYSTEM

Home > SQL > SQL Commands

☒ Autocommit Display 10 ▼

```
SELECT last_name,  
       TO_CHAR(hire_date, 'fmDay, "the" DDth "of" Month, YYYY') AS hire_date,  
       TO_CHAR(  
         NEXT_DAY(ADD_MONTHS(hire_date, 6) - 1, 'MONDAY'),  
         'fmDay, "the" DDth "of" Month, YYYY'  
       ) AS REVIEW  
FROM employees;
```

Results Explain Describe Saved SQL History

LAST_NAME	HIRE_DATE	REVIEW
Smith	Tuesday, the 15TH of March, 2005	Monday, the 19TH of September, 2005
Doe	Tuesday, the 1ST of June, 2010	Monday, the 6TH of December, 2010
Johnson	Friday, the 23RD of November, 2012	Monday, the 27TH of May, 2013
Brown	Wednesday, the 10TH of January, 2018	Monday, the 16TH of July, 2018

4 rows returned in 0.02 seconds

[CSV Export](#)

10. Display the last name, hire date, and day of the week on which the employee started. Label the column DAY. Order the results by the day of the week, starting with Monday.

User: SYSTEM

Home > SQL > SQL Commands

☒ Autocommit Display 10 ▼

```
SELECT last_name,  
       hire_date,  
       TO_CHAR(hire_date, 'Day') AS DAY  
FROM employees  
ORDER BY TO_CHAR(hire_date, 'D');
```

Results Explain Describe Saved SQL History

LAST_NAME	HIRE_DATE	DAY
Smith	15-MAR-05	Tuesday
Doe	01-JUN-10	Tuesday
Brown	10-JAN-18	Wednesday
Johnson	23-NOV-12	Friday

4 rows returned in 0.00 seconds

[CSV Export](#)