

3

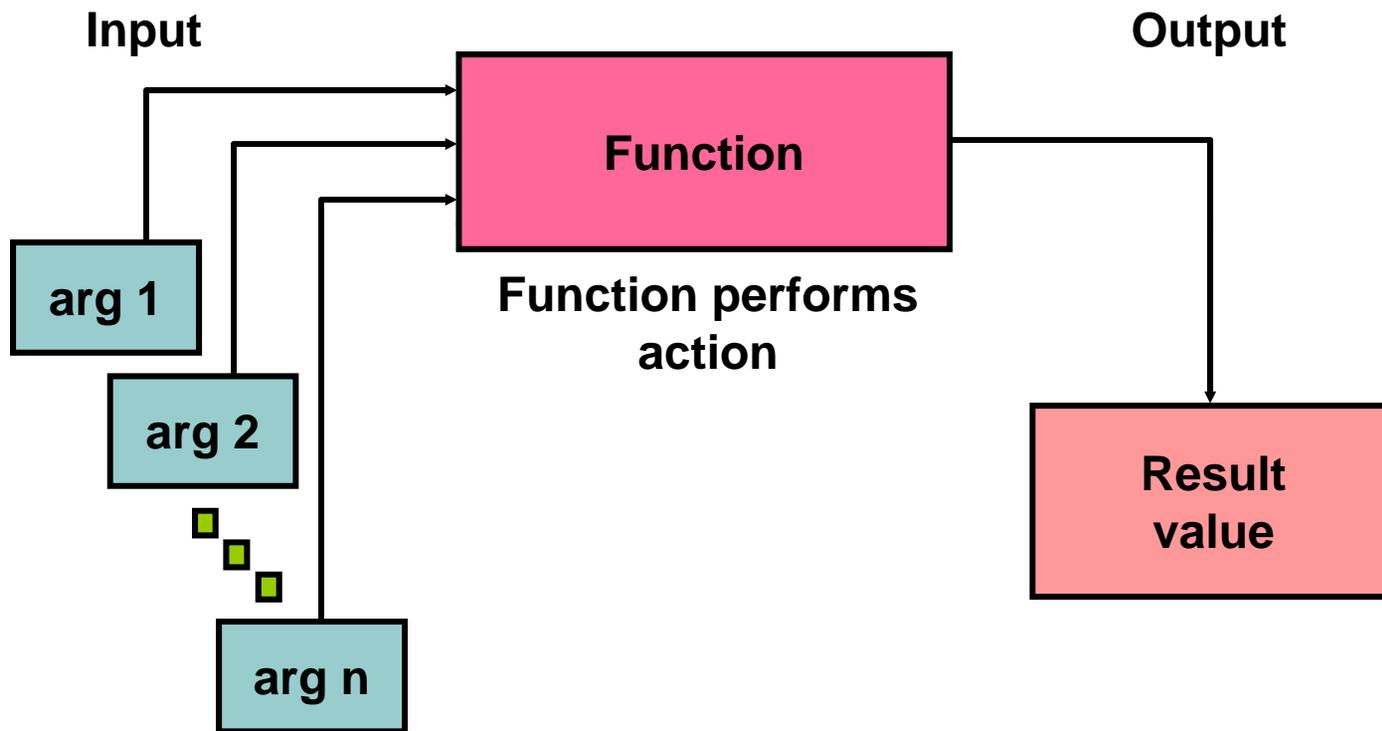
Using Single-Row Functions to Customize Output

Objectives

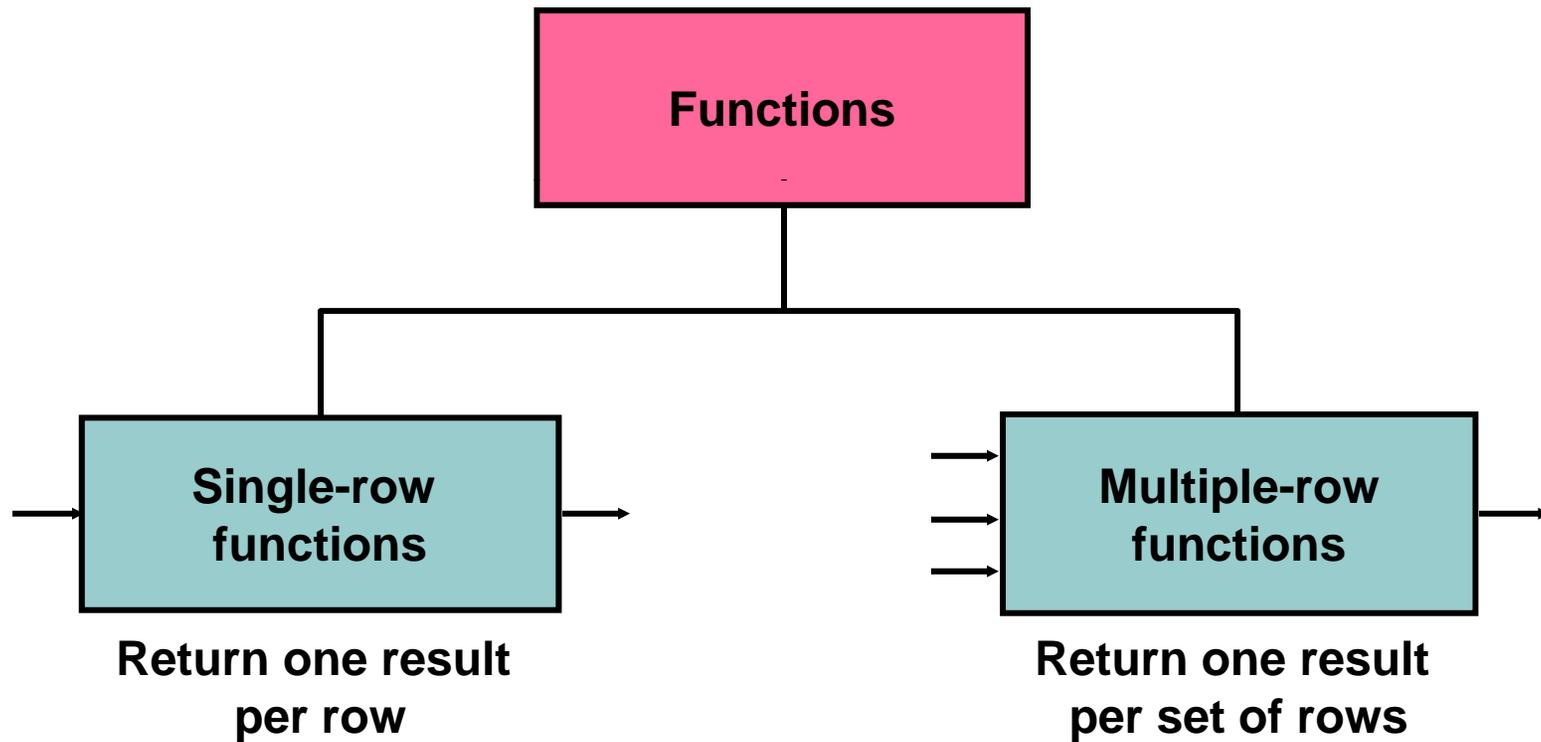
After completing this lesson, you should be able to do the following:

- **Describe various types of functions that are available in SQL**
- **Use character, number, and date functions in `SELECT` statements**
- **Describe the use of conversion functions**

SQL Functions



Two Types of SQL Functions



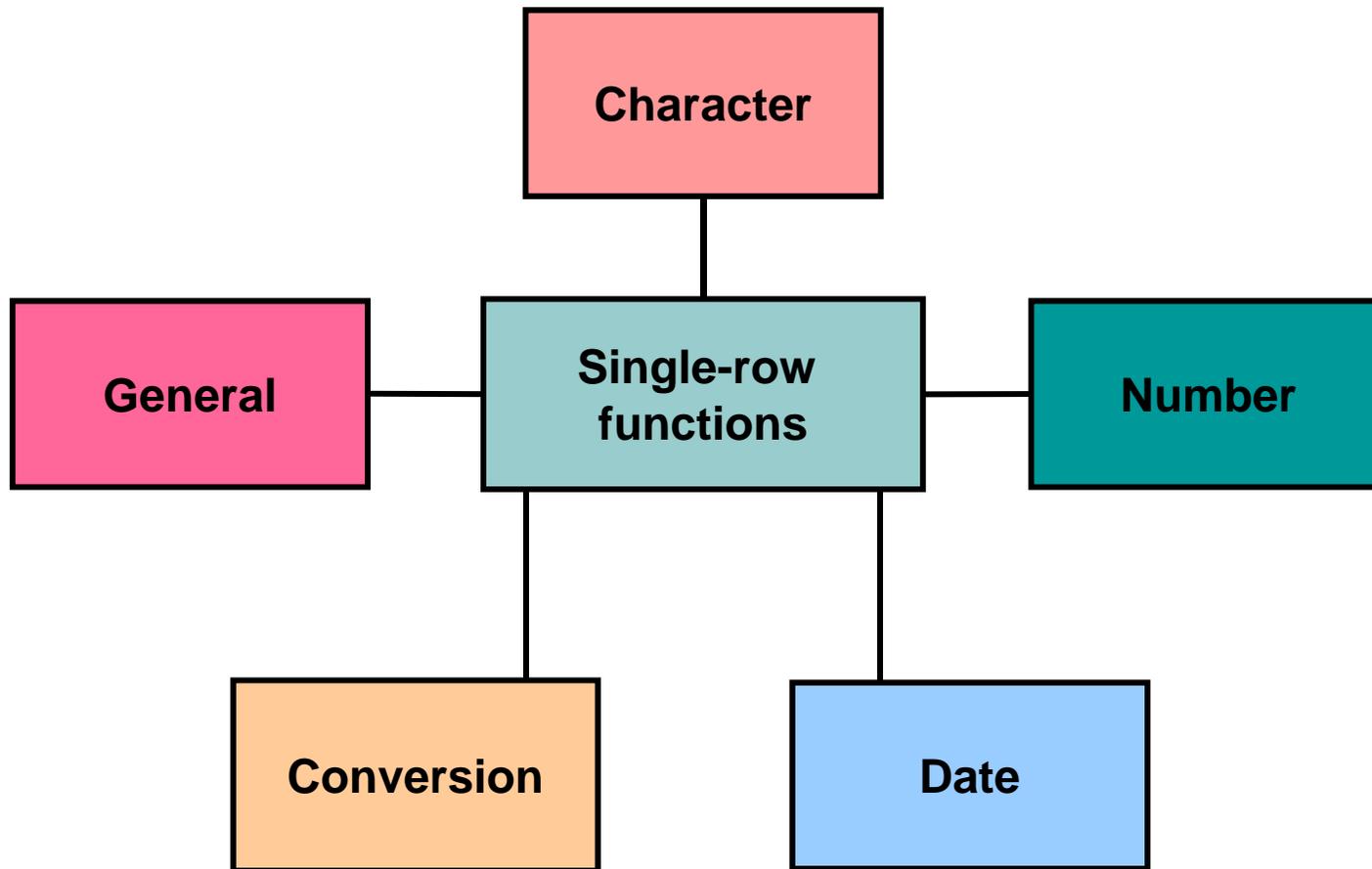
Single-Row Functions

Single-row functions:

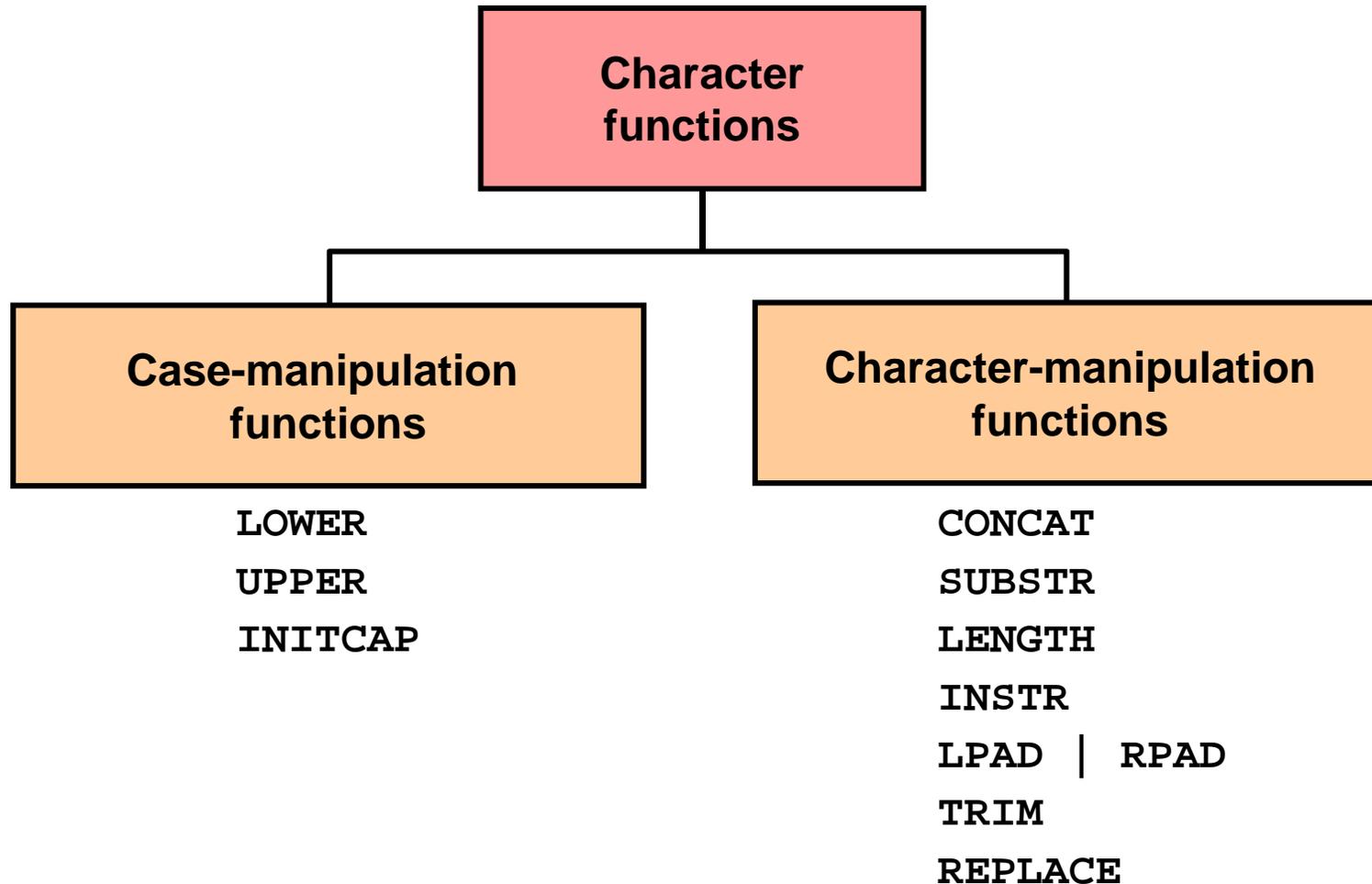
- Manipulate data items
- Accept arguments and return one value
- Act on each row that is returned
- Return one result per row
- May modify the data type
- Can be nested
- Accept arguments that can be a column or an expression

```
function_name [(arg1, arg2,...)]
```

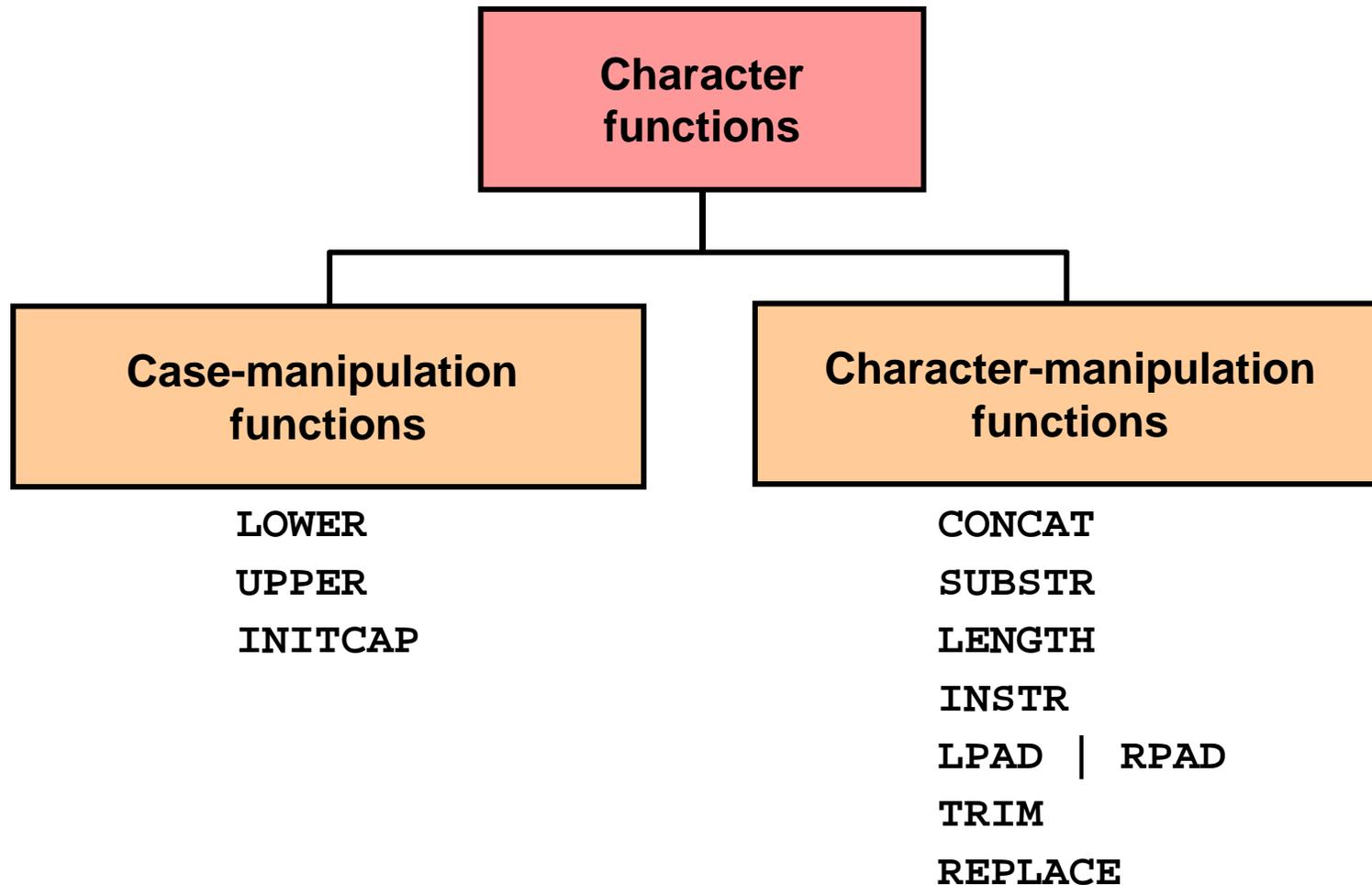
Single-Row Functions



Character Functions



Character Functions



Case-Manipulation Functions

These functions convert case for character strings:

Function	Result
<code>LOWER('SQL Course')</code>	sql course
<code>UPPER('SQL Course')</code>	SQL COURSE
<code>INITCAP('SQL Course')</code>	Sql Course

Using Case-Manipulation Functions

Display the employee number, name, and department number for employee Higgins:

```
SELECT employee_id, last_name, department_id
FROM employees
WHERE last_name = 'higgins';
no rows selected
```

```
SELECT employee_id, last_name, department_id
FROM employees
WHERE LOWER(last_name) = 'higgins';
```

EMPLOYEE_ID	LAST_NAME	DEPARTMENT_ID
205	Higgins	110

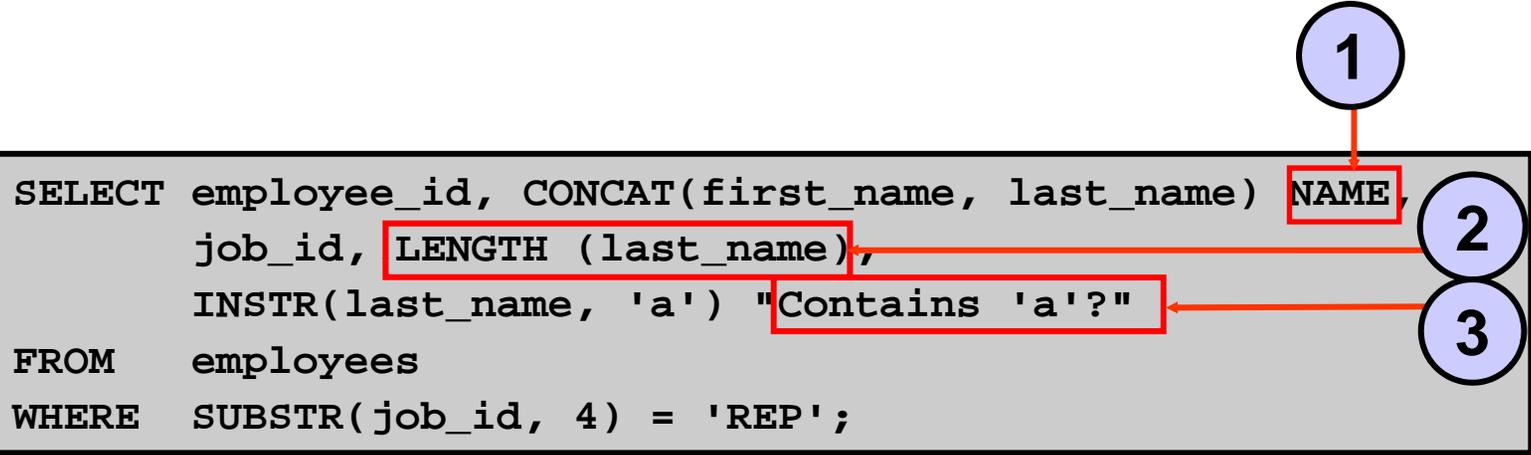
Character-Manipulation Functions

These functions manipulate character strings:

Function	Result
<code>CONCAT('Hello', 'World')</code>	HelloWorld
<code>SUBSTR('HelloWorld',1,5)</code>	Hello
<code>LENGTH('HelloWorld')</code>	10
<code>INSTR('HelloWorld', 'W')</code>	6
<code>LPAD(salary,10,'*')</code>	*****24000
<code>RPAD(salary, 10, '*')</code>	24000*****
<code>REPLACE('JACK and JUE', 'J', 'BL')</code>	BLACK and BLUE
<code>TRIM('H' FROM 'HelloWorld')</code>	elloWorld

Using the Character-Manipulation Functions

```
SELECT employee_id, CONCAT(first_name, last_name) NAME,  
       job_id, LENGTH(last_name),  
       INSTR(last_name, 'a') "Contains 'a'?"  
FROM employees  
WHERE SUBSTR(job_id, 4) = 'REP';
```



EMPLOYEE_ID	NAME	JOB_ID	LENGTH(LAST_NAME)	Contains 'a'?
174	EllenAbel	SA_REP	4	0
176	JonathonTaylor	SA_REP	6	2
178	KimberelyGrant	SA_REP	5	3
202	PatFay	MK_REP	3	2

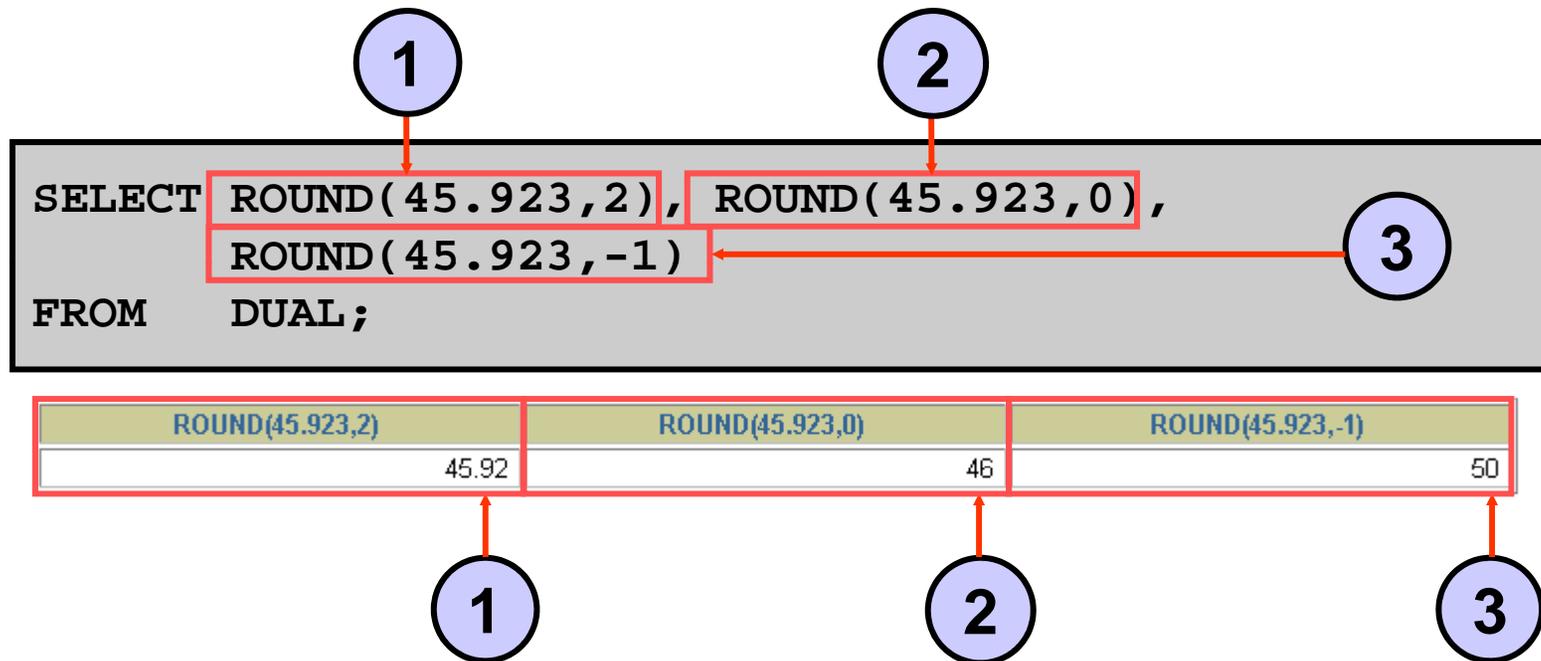


Number Functions

- **ROUND:** Rounds value to specified decimal
- **TRUNC:** Truncates value to specified decimal
- **MOD:** Returns remainder of division

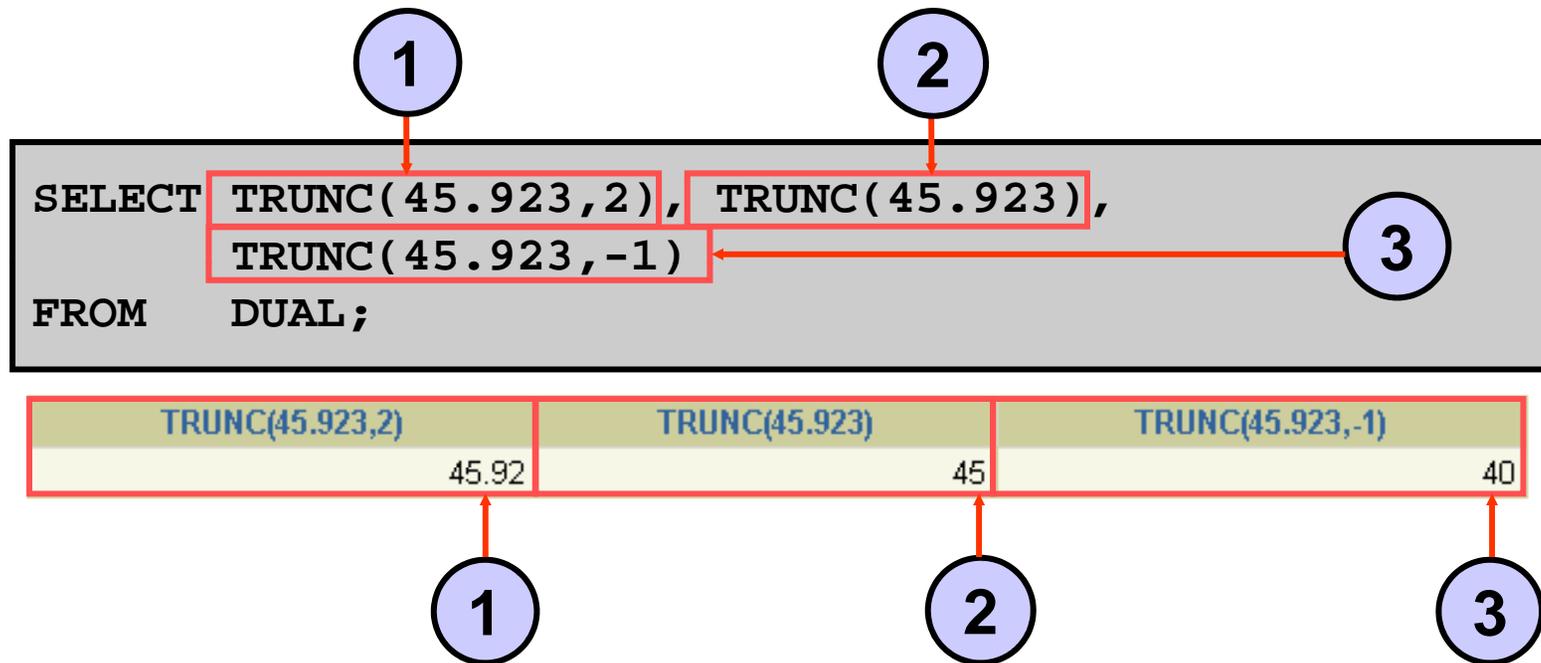
Function	Result
<code>ROUND(45.926, 2)</code>	45.93
<code>TRUNC(45.926, 2)</code>	45.92
<code>MOD(1600, 300)</code>	100

Using the ROUND Function



DUAL is a dummy table that you can use to view results from functions and calculations.

Using the TRUNC Function



Using the MOD Function

For all employees with job title of Sales Representative, calculate the remainder of the salary after it is divided by 5,000.

```
SELECT last_name, salary, MOD(salary, 5000)
FROM employees
WHERE job_id = 'SA_REP';
```

LAST_NAME	SALARY	MOD(SALARY,5000)
Abel	11000	1000
Taylor	8600	3600
Grant	7000	2000

Working with Dates

- The Oracle database stores dates in an internal numeric format: century, year, month, day, hours, minutes, and seconds.
- The default date display format is DD-MON-RR.
 - Enables you to store 21st-century dates in the 20th century by specifying only the last two digits of the year
 - Enables you to store 20th-century dates in the 21st century in the same way

```
SELECT last_name, hire_date
FROM employees
WHERE hire_date < '01-FEB-88';
```

LAST_NAME	HIRE_DATE
King	17-JUN-87
Whalen	17-SEP-87

Working with Dates

SYSDATE is a function that returns:

- **Date**
- **Time**

Arithmetic with Dates

- **Add or subtract a number to or from a date for a resultant date value.**
- **Subtract two dates to find the number of days between those dates.**
- **Add hours to a date by dividing the number of hours by 24.**

Using Arithmetic Operators with Dates

```
SELECT last_name, (SYSDATE-hire_date)/7 AS WEEKS  
FROM employees  
WHERE department_id = 90;
```

LAST_NAME	WEEKS
King	744.245395
Kochhar	626.102538
De Haan	453.245395

Date Functions

Function	Result
MONTHS_BETWEEN	Number of months between two dates
ADD_MONTHS	Add calendar months to date
NEXT_DAY	Next day of the date specified
LAST_DAY	Last day of the month
ROUND	Round date
TRUNC	Truncate date

Using Date Functions

Function	Result
<code>MONTHS_BETWEEN</code> <code>('01-SEP-95' , '11-JAN-94')</code>	19.6774194
<code>ADD_MONTHS</code> ('11-JAN-94' , 6)	'11-JUL-94'
<code>NEXT_DAY</code> ('01-SEP-95' , 'FRIDAY')	'08-SEP-95'
<code>LAST_DAY</code> ('01-FEB-95')	'28-FEB-95'

Using Date Functions

Assume `SYSDATE = '25-JUL-03'`:

Function	Result
<code>ROUND (SYSDATE , 'MONTH')</code>	01-AUG-03
<code>ROUND (SYSDATE , 'YEAR')</code>	01-JAN-04
<code>TRUNC (SYSDATE , 'MONTH')</code>	01-JUL-03
<code>TRUNC (SYSDATE , 'YEAR')</code>	01-JAN-03

Practice 3: Overview of Part 1

This practice covers the following topics:

- **Writing a query that displays the current date**
- **Creating queries that require the use of numeric, character, and date functions**
- **Performing calculations of years and months of service for an employee**