Database systems Lab

Course Code	: 30102422
Credit Hours	: 1
Prerequisite	: 30102421

Instructor Information

Name	:Dr. Jihad Nader				
Office No.	B17 F4 R8				
Tel (Ext)	N/A				
E-mail	jihadnader@bau.edu.jo				
Office Hours	Sun, Thurs ,Tues (9-10) mon, wed (10-11)				
Class Times	Building	Day	Start Time	End Time	Room No.
	-	Monday	14	17	مختبر المعالجات
	-	Wednesday	14	17	مختبر المعالجات

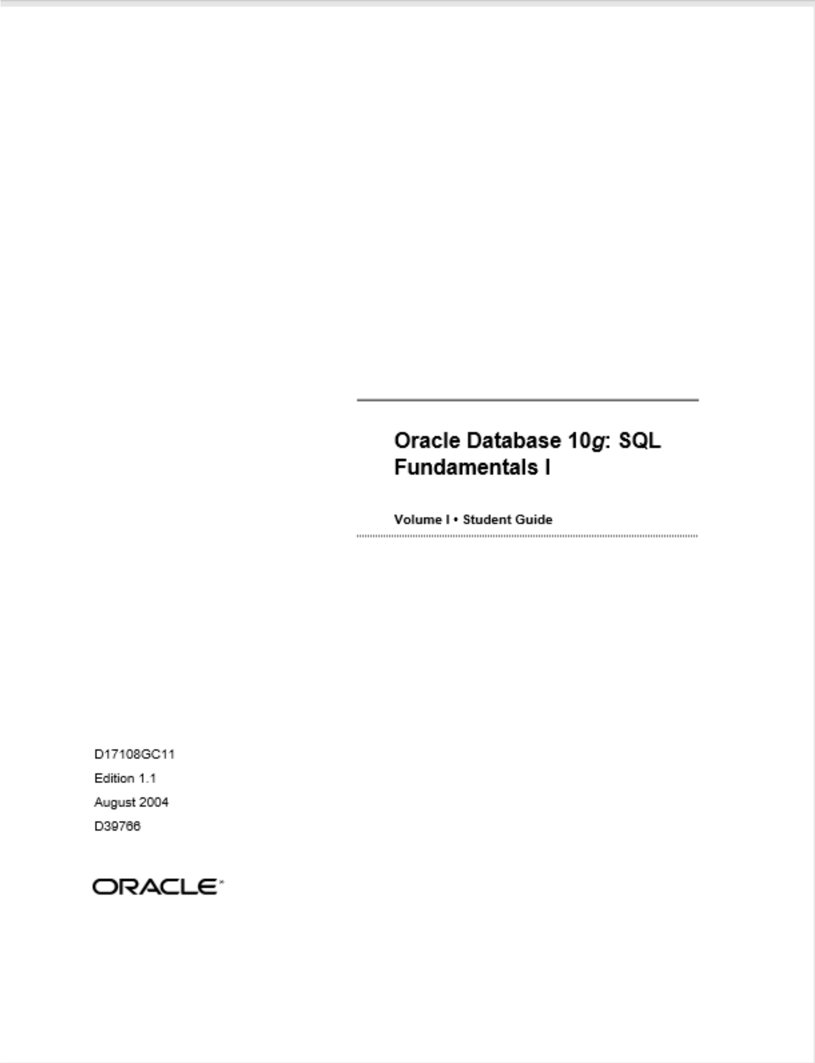
Course Description:

This Lab. practices the concepts introduced in the Database systems course using Oracle Database. The students are expected to implement a database project for some problem.

Course Title: Database Systems Lab Credit Hour(1)

[Pre-req. Course Code(30102421)]

Textbook: *Oracle Database 10g: SQL Fundamentals I, Volume I • Student Guide*



Oracle Database 10*g*: SQL Fundamentals I

Volume I • Student Guide

D17108GC11
Edition 1.1
August 2004
D39766

ORACLE

COURSE OBJECTIVES:

Upon completion of this course, students will have gained knowledge of the DBMS (Oracle) concepts and the ability to:

- Understand the concepts of relational databases and the Oracle Database 10g database technology.
- Use the powerful SQL programming language and its features.
- Identify features of Relational Database Management System (RDBMS).
- Categorize the main database objects
- Understand how constraints are created at the time of table creation
- Describe each data manipulation language (DML) statement
- List the capabilities of SQL SELECT statements
- Write SELECT statements to access data from more than one table using equijoins and non-equijoins
- Employ SQL functions to generate and retrieve customized data
- Identify when a subquery can help solve a question
- Write subqueries when a query is based on unknown values
- Use a set operator to combine multiple queries into a single query

COURSE SYLLABUS

Week	Course Topic	Notes
Week 1	**Creating and Managing Tables:** - Database Objects - Naming Conventions - The Create Table Statement - Creating a Table by Using a Subquery - Querying the Data Dictionary - The Alter Table Statement - Truncating a Table - Adding Comments to a Table	
Week 2	Including Constraints - Defining Constraints o The Not Null Constraint o The Unique Constraint o The Primary Key Constraint o The Foreign Key Constraint o The Check Constraint - Adding a Constraint - Dropping a Constraint - Enabling and Disabling Constraints - Viewing Constraints	
Week 3	Manipulating Data - Data Manipulating Language. - The Insert Statement - Copying Rows from another Table - The Update Statement - The Delete Statement - Database Transactions - Commit and Rollback Statements	
Week 4	Writing Basic SQL Statements - Selecting Specific Columns - Arithmetic Expressions - Concatenation Operator - Using Column Aliases - Eliminating Duplicate Rows	

COURSE SYLLABUS

Week	Course Topic	Notes
Week 5	**Restricting and Sorting Data** - **Where Clause** - **Comparison Operators** - **Special Operators** - **Logical Operator (And, Or, Not)** - **Order By Clause**	
Week 6	Displaying Data from Multiple Tables - Cartesian Product. - Types of Joins - Table Aliases.	
Week 7	Single-Row Functions - Character Functions. - Number Functions - Date Functions ▪	
Week 8	Midterm Exam	Midterm Exam
Week 9	Project Proposal	
Week 10	Single-Row Functions - Conversion Functions - General Functions	

COURSE SYLLABUS

Week	Course Topic	Notes
Week 11	**Aggregating Data using Group Functions** - **Types of Group Functions (AVG, SUM, MAX, MIN, COUNT).** - **Creating Groups of data: Group By Clause.** - **Excluding Group Results: Having Clause.** - **Nested Group Functions**	
Week 12	Subqueries - Types of Subqueries ▪ Single-Row Subqueries ▪ Multiple-Row Subqueries	
Week 13	Multiple-Column Subqueries - Column Comparisons - Null Values in a subquery - Using a subquery in the From Clause	
Week 14	Using the Set Operators - Union / Union All - Intersect - Minus	
Week 15	Project Discussion	
Week 16	Final Exam	Final Exam

Week 7

Chapter 6:

Single Row Functions

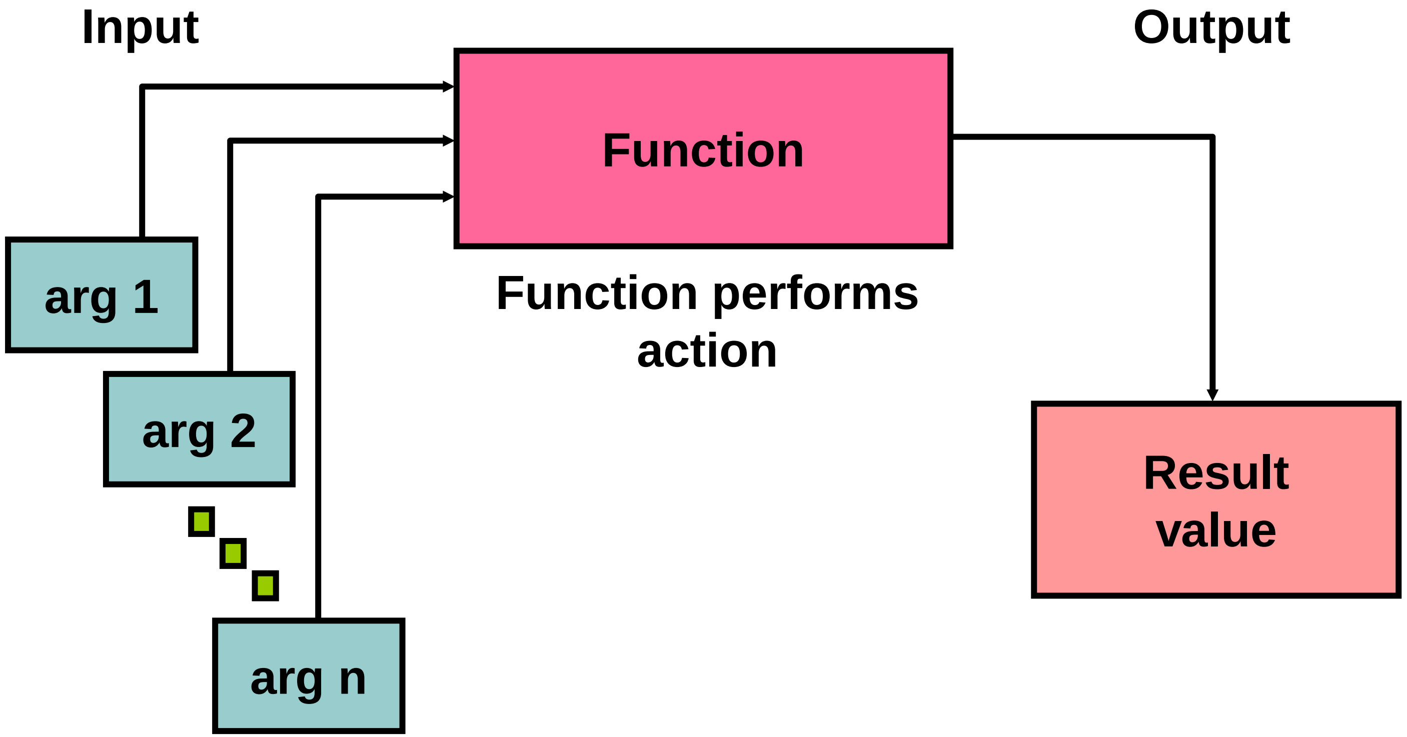
Using Single-Row Functions to Customize Output

Objectives

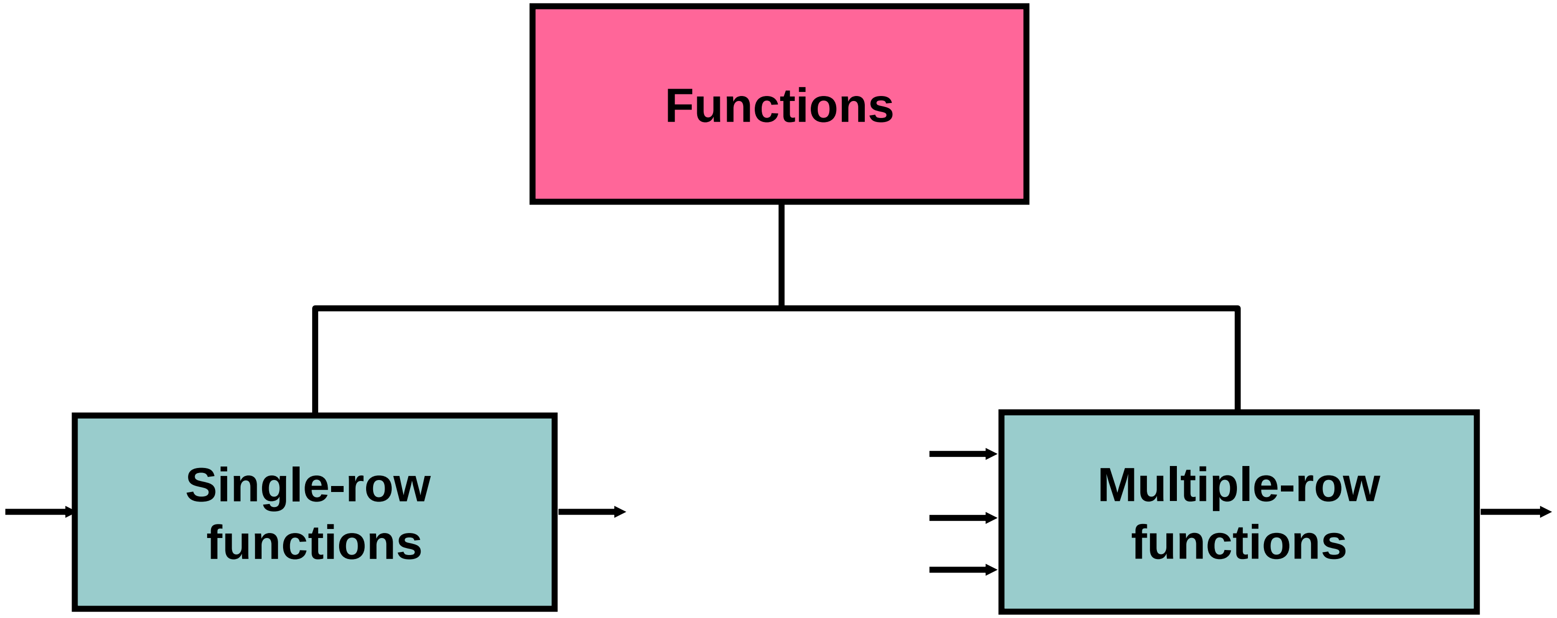
After completing this lesson, you should be able to do the following:

- 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



Functions

Single-row functions

Multiple-row functions

Return one result per row

Return one result per set of rows

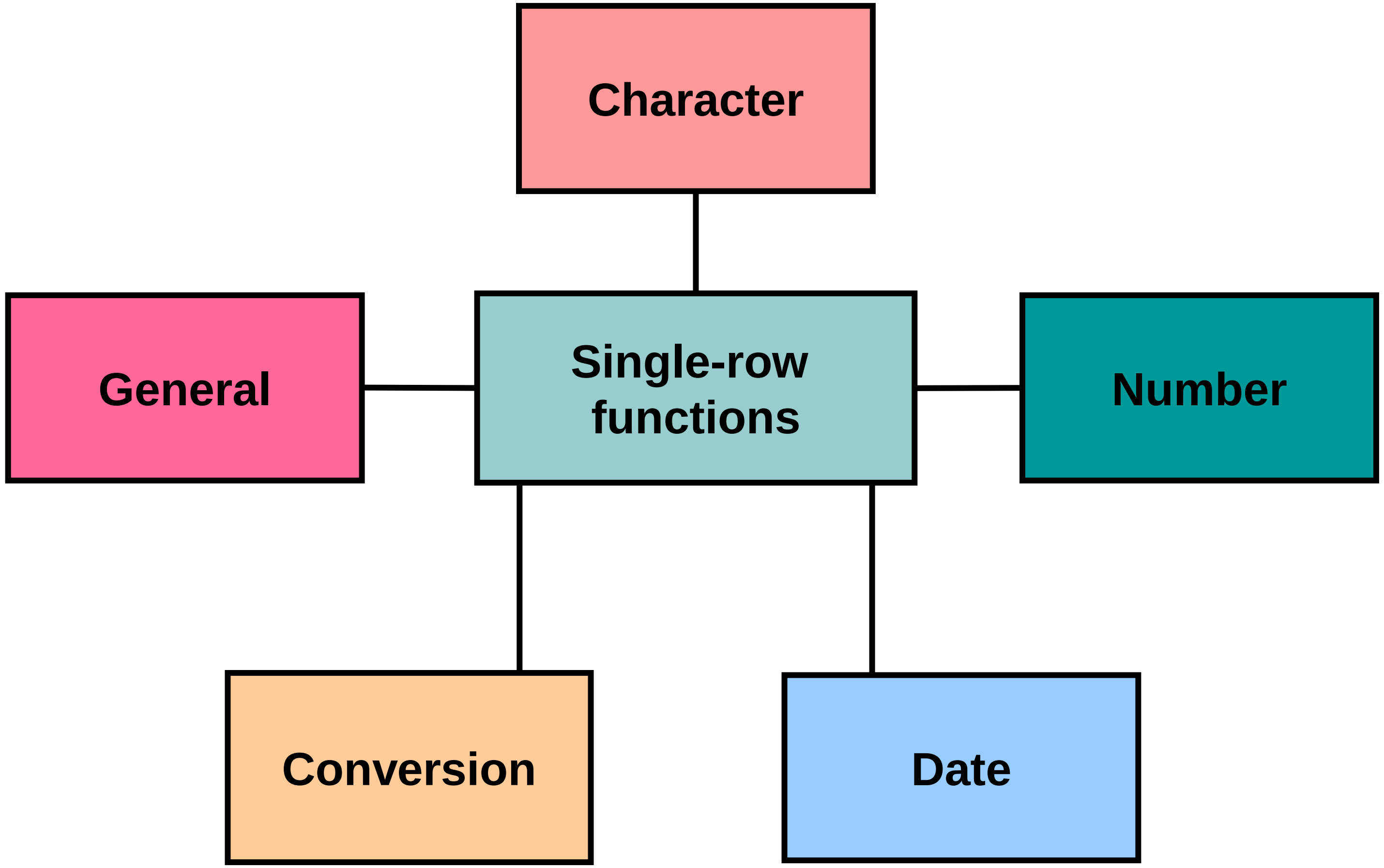
Single-Row Functions

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

General

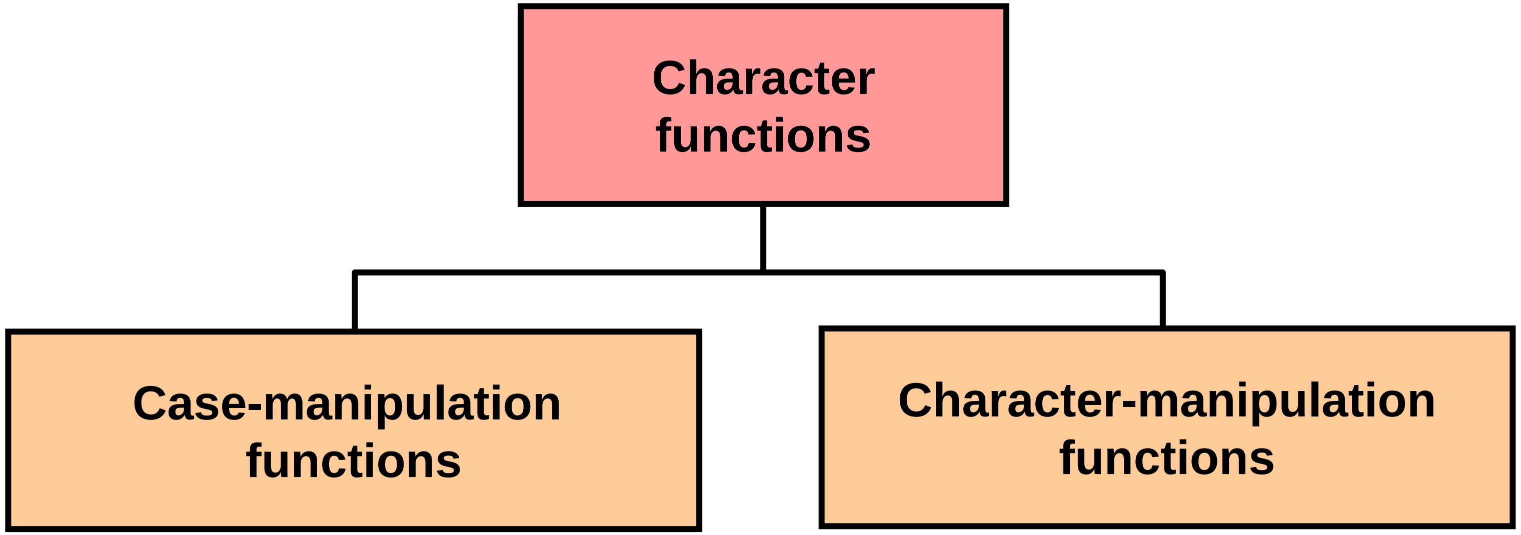
Single-row functions

Number

Conversion

Date

Character Functions



Character functions

Case-manipulation functions

Character-manipulation functions

```
LOWER
UPPER
INITCAP
```

```
CONCAT
SUBSTR
LENGTH
INSTR
LPAD | RPAD
REPLACE
```

Case-Manipulation Functions

These functions convert case for character strings:

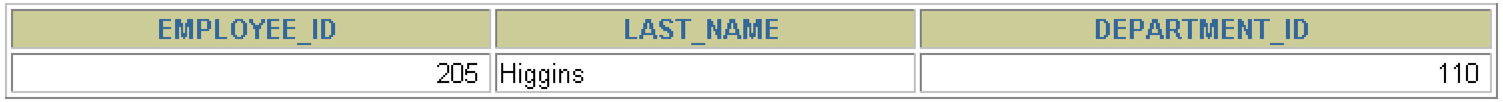
Function	Result
`LOWER('SQL Course')`	`sql course`
`UPPER('SQL Course')`	`SQL COURSE`
`INITCAP('SQL Course')`	`Sql Course`

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';
```



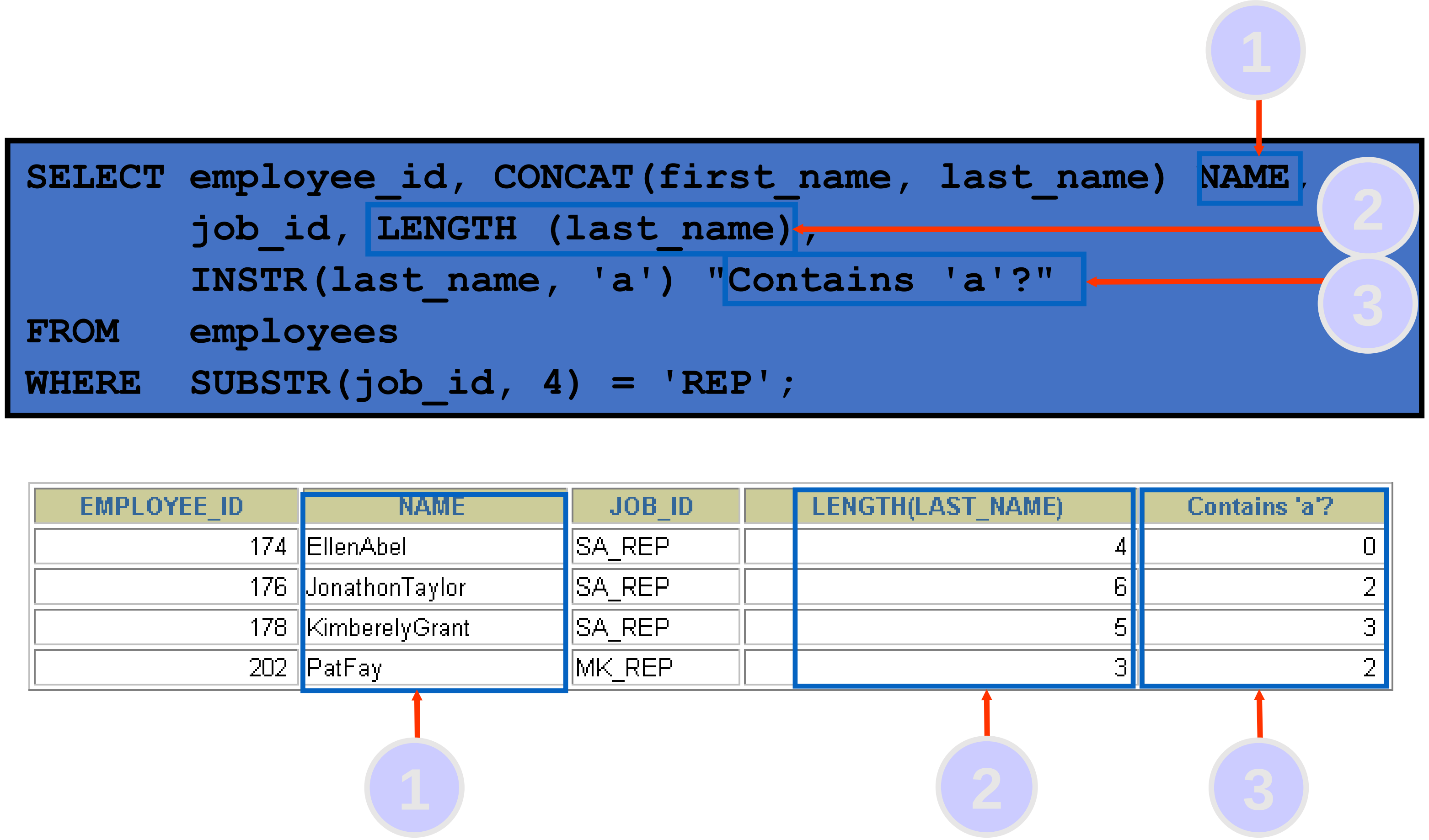
EMPLOYEE_ID	LAST_NAME	DEPARTMENT_ID
205	Higgins	110

Character-Manipulation Functions

These functions manipulate character strings:

Function	Result
CONCAT('Hello', 'World')	HelloWorld
SUBSTR('HelloWorld',1,5)	Hello
LENGTH('HelloWorld')	10
INSTR('HelloWorld', 'W')	6
LPAD(salary,10,'*')	*****24000
RPAD(salary, 10, '*')	24000*****
REPLACE ('JACK and JUE','J','BL')	BLACK and BLUE

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';
```

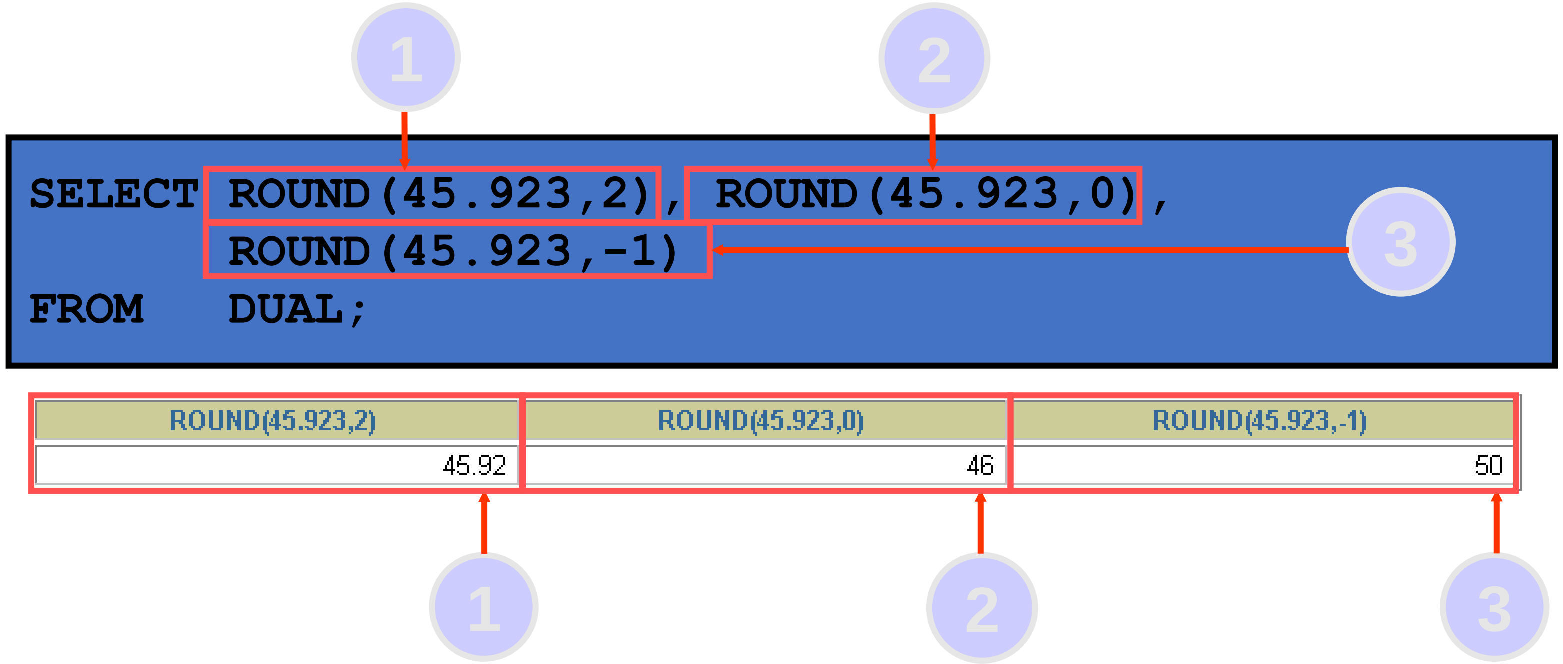
EMPLOYEE_ID	NAME	JOB_ID	LENGTH(LAST_NAME)	Contains 'a'?
174	EllenAbel	SA_REP	4	0
176	JonathonTaylor	SA_REP	6	2
178	KimberelyGrant	SA_REP	5	3
202	PatFay	MK_REP	3	2

Number Functions

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

Function	Result
`ROUND(45.926, 2)`	**`45.93`**
`TRUNC(45.926, 2)`	**`45.92`**
`MOD(1600, 300)`	**`100`**

Using the ROUND Function

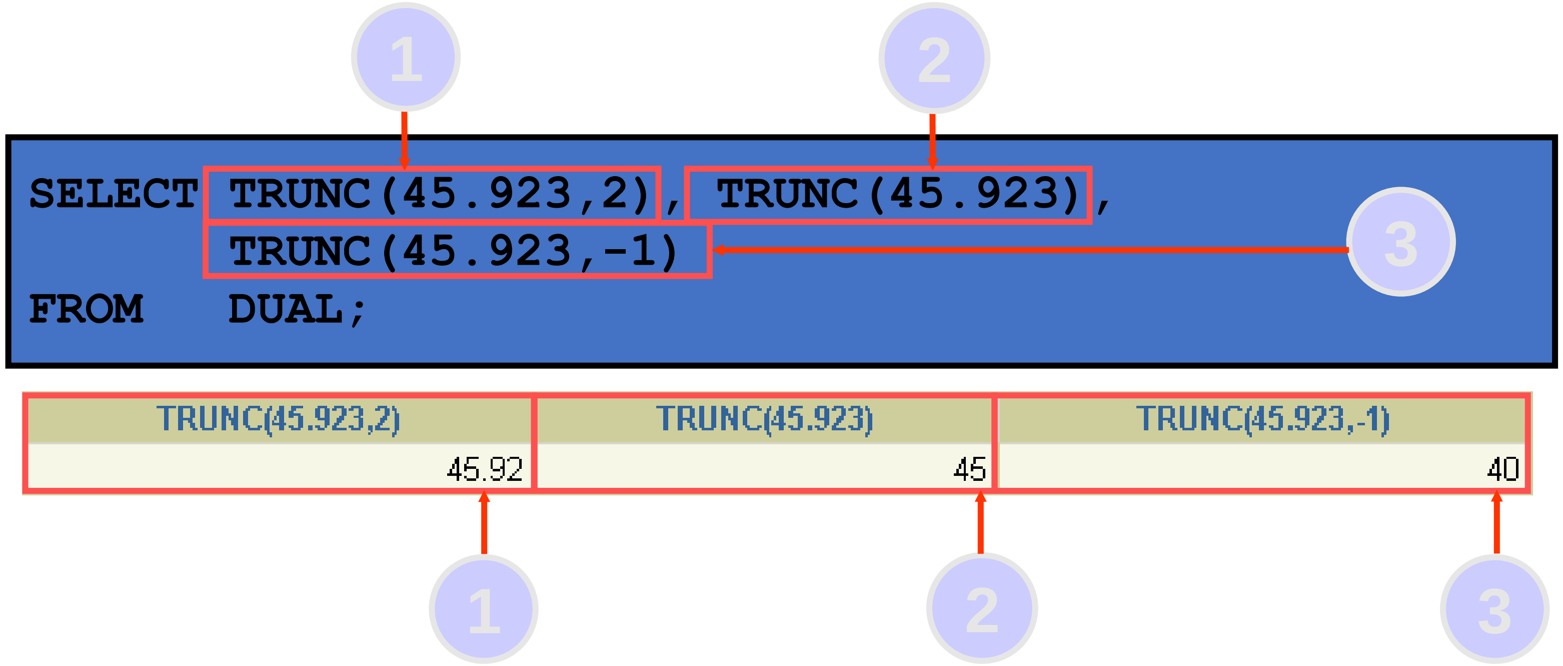


```
SELECT ROUND(45.923,2), ROUND(45.923,0),
       ROUND(45.923,-1)
FROM   DUAL;
```

ROUND(45.923,2)	ROUND(45.923,0)	ROUND(45.923,-1)
45.92	46	50

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

Using the TRUNC Function

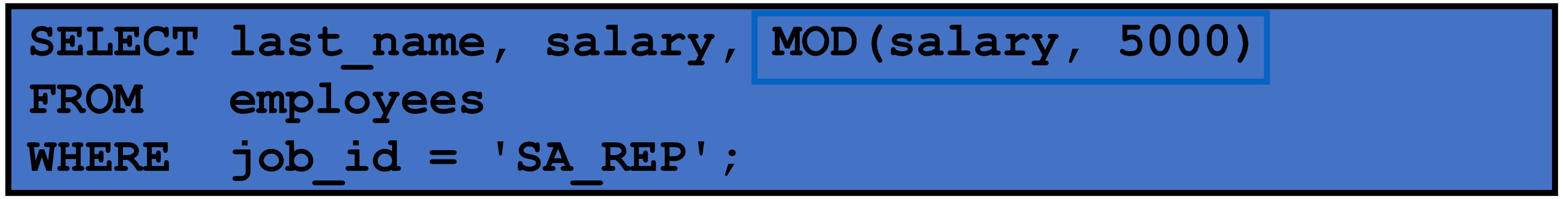


```
SELECT TRUNC(45.923,2), TRUNC(45.923),
       TRUNC(45.923,-1)
FROM   DUAL;
```

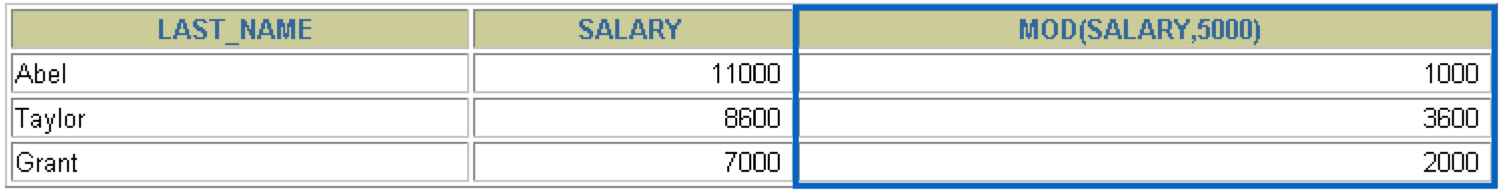
```
TRUNC(45.923,2)   TRUNC(45.923)   TRUNC(45.923,-1)
          45.92              45                 40
```

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';
```

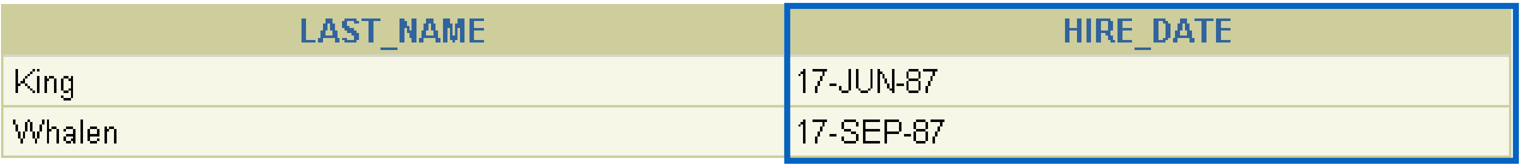


LAST_NAME	SALARY	MOD(SALARY,5000)
Abel	11000	1000
Taylor	8600	3600
Grant	7000	2000

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:

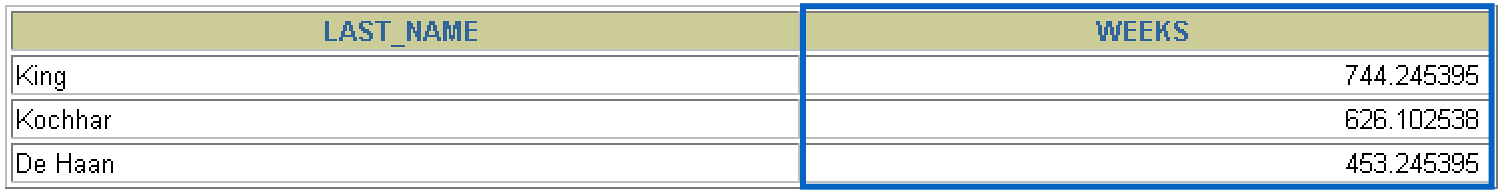
- Date
- Time

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.

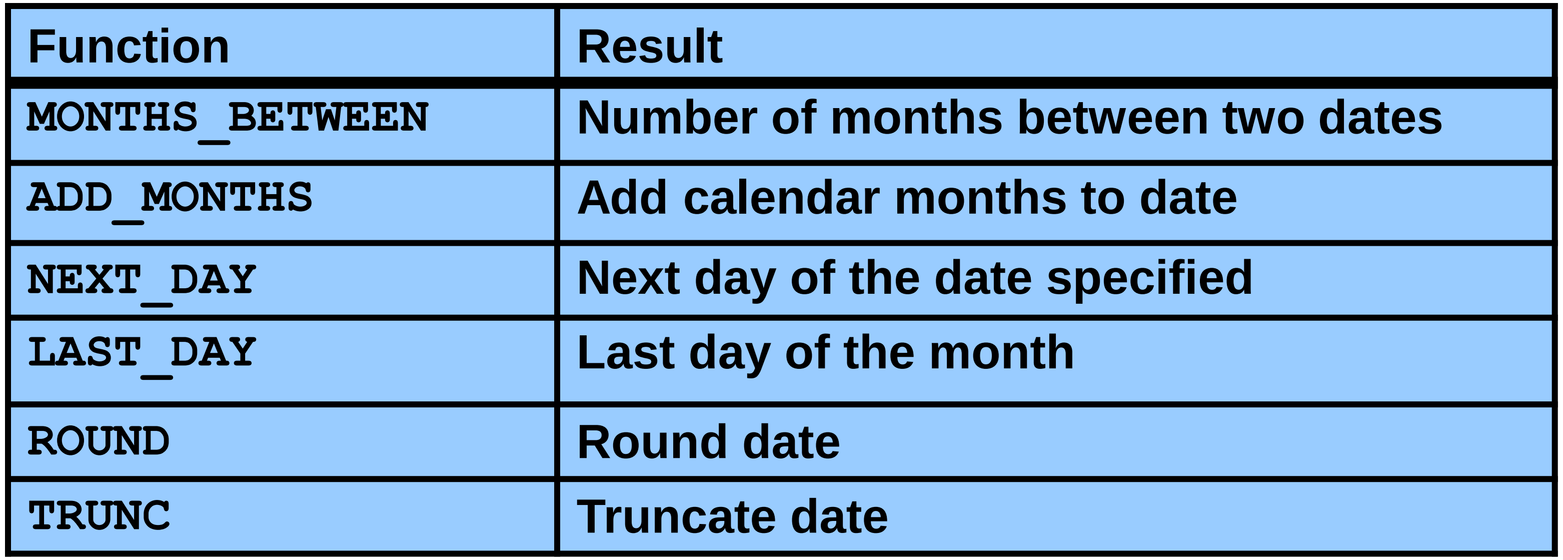
Using Arithmetic Operators with Dates

```
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
MONTHS_BETWEEN ('01-SEP-95','11-JAN-94')	19.6774194
ADD_MONTHS ('11-JAN-94',6)	'11-JUL-94'
NEXT_DAY ('01-SEP-95','FRIDAY')	'08-SEP-95'
LAST_DAY ('01-FEB-95')	'28-FEB-95'

Using Date Functions

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

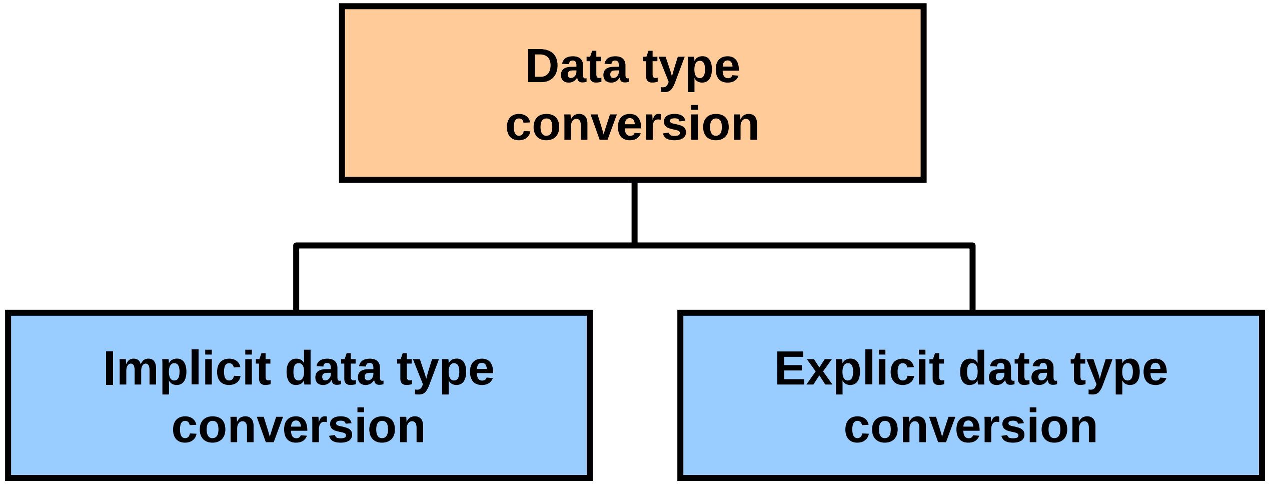
Function	Result
`ROUND(SYSDATE,'MONTH')`	`01-AUG-03`
`ROUND(SYSDATE ,'YEAR')`	`01-JAN-04`
`TRUNC(SYSDATE ,'MONTH')`	`01-JUL-03`
`TRUNC(SYSDATE ,'YEAR')`	`01-JAN-03`

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

Conversion Functions



Data type conversion

Implicit data type conversion

Explicit data type conversion

Implicit Data Type Conversion

For assignments, the Oracle server can automatically convert the following:

From	To
`VARCHAR2 or CHAR`	`NUMBER`
`VARCHAR2 or CHAR`	`DATE`
`NUMBER`	`VARCHAR2`
`DATE`	`VARCHAR2`

Implicit Data Type Conversion

For expression evaluation, the Oracle Server can automatically convert the following:

From	To
`VARCHAR2 or CHAR`	`NUMBER`
`VARCHAR2 or CHAR`	`DATE`