

SQL1 - Practice

Practice 1: Overview

This practice covers the following topics:

- **Selecting all data from different tables**
- **Describing the structure of tables**
- **Performing arithmetic calculations and specifying column names**
- **Using *iSQL*Plus***

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Practice 1: Overview

This is the first of many practices in this course. The solutions (if you require them) can be found in Appendix A. Practices are intended to cover all topics that are presented in the corresponding lesson.

Note the following location for the lab files:

E:\labs\SQL1\labs

If you are asked to save any lab files, save them at this location.

To start *iSQL*Plus*, start your browser. You need to enter a URL to access *iSQL*Plus*. The URL requires the host name, which your instructor will provide. Enter the following command, replacing the host name with the value that your instructor provides:

`http://<HOSTNAME:5561>/isqlplus`

In any practice, there may be exercises that are prefaced with the phrases “If you have time” or “If you want an extra challenge.” Work on these exercises only if you have completed all other exercises in the allocated time and would like a further challenge to your skills.

Perform the practices slowly and precisely. You can experiment with saving and running command files. If you have any questions at any time, ask your instructor.

Practice 1

Part 1

Test your knowledge:

1. Initiate an iSQL*Plus session using the user ID and password that are provided by the instructor.
2. iSQL*Plus commands access the database.

True/False

3. The following SELECT statement executes successfully:

```
SELECT last_name, job_id, salary AS Sal
FROM   employees;
```

True/False

4. The following SELECT statement executes successfully:

```
SELECT *
FROM   job_grades;
```

True/False

5. There are four coding errors in the following statement. Can you identify them?

```
SELECT      employee_id, last_name
sal x 12    ANNUAL SALARY
FROM        employees;
```

Part 2

Note the following location for the lab files:

E:\labs\SQL1\labs

If you are asked to save any lab files, save them at this location.

*To start iSQL*Plus, start your browser. You need to enter a URL to access iSQL*Plus. The URL requires the host name, which your instructor will provide. Enter the following command, replacing the host name with the value that your instructor provides:*

http://<HOSTNAME:5561>/isqlplus

You have been hired as a SQL programmer for Acme Corporation. Your first task is to create some reports based on data from the Human Resources tables.

6. Your first task is to determine the structure of the DEPARTMENTS table and its contents.

Name	Null?	Type
DEPARTMENT_ID	NOT NULL	NUMBER(4)
DEPARTMENT_NAME	NOT NULL	VARCHAR2(30)
MANAGER_ID		NUMBER(6)
LOCATION_ID		NUMBER(4)

Practice 1 (continued)

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
50	Shipping	124	1500
60	IT	103	1400
80	Sales	149	2500
90	Executive	100	1700
110	Accounting	205	1700
190	Contracting		1700

8 rows selected.

7. You need to determine the structure of the EMPLOYEES table.

Name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER(6)
FIRST_NAME		VARCHAR2(20)
LAST_NAME	NOT NULL	VARCHAR2(25)
EMAIL	NOT NULL	VARCHAR2(25)
PHONE_NUMBER		VARCHAR2(20)
HIRE_DATE	NOT NULL	DATE
JOB_ID	NOT NULL	VARCHAR2(10)
SALARY		NUMBER(8,2)
COMMISSION_PCT		NUMBER(2,2)
MANAGER_ID		NUMBER(6)
DEPARTMENT_ID		NUMBER(4)

The HR department wants a query to display the last name, job code, hire date, and employee number for each employee, with employee number appearing first. Provide an alias STARTDATE for the HIRE_DATE column. Save your SQL statement to a file named lab_01_07.sql so that you can disperse this file to the HR department.

Practice 1 (continued)

- Test your query in the `lab_01_07.sql` file to ensure that it runs correctly.

EMPLOYEE_ID	LAST_NAME	JOB_ID	STARTDATE
100	King	AD_PRES	17-JUN-87
101	Kochhar	AD_VP	21-SEP-89
102	De Haan	AD_VP	13-JAN-93
103	Hunold	IT_PROG	03-JAN-90
104	Ernst	IT_PROG	21-MAY-91
107	Lorentz	IT_PROG	07-FEB-99
124	Mourgos	ST_MAN	16-NOV-99
141	Rajs	ST_CLERK	17-OCT-95
142	Davies	ST_CLERK	29-JAN-97
143	Matos	ST_CLERK	15-MAR-98
144	Vargas	ST_CLERK	09-JUL-98
149	Zlotkey	SA_MAN	29-JAN-00
174	Abel	SA_REP	11-MAY-96
176	Taylor	SA_REP	24-MAR-98
...			
206	Gietz	AC_ACCOUNT	07-JUN-94

20 rows selected.

- The HR department needs a query to display all unique job codes from the `EMPLOYEES` table.

JOB_ID
AC_ACCOUNT
AC_MGR
AD_ASST
AD_PRES
AD_VP
IT_PROG
MK_MAN
MK_REP
SA_MAN
SA_REP
ST_CLERK
ST_MAN

12 rows selected.

Practice 1 (continued)

Part 3

If you have time, complete the following exercises:

- The HR department wants more descriptive column headings for its report on employees. Copy the statement from lab_01_07.sql to the *iSQL*Plus* text box. Name the column headings Emp #, Employee, Job, and Hire Date, respectively. Then run your query again.

Emp #	Employee	Job	Hire Date
100	King	AD_PRES	17-JUN-87
101	Kochhar	AD_VP	21-SEP-89
102	De Haan	AD_VP	13-JAN-93
103	Hunold	IT_PROG	03-JAN-90
104	Ernst	IT_PROG	21-MAY-91
107	Lorentz	IT_PROG	07-FEB-99
124	Mourgos	ST_MAN	16-NOV-99
141	Rajs	ST_CLERK	17-OCT-95
142	Davies	ST_CLERK	29-JAN-97
143	Matos	ST_CLERK	15-MAR-98
144	Vargas	ST_CLERK	09-JUL-98
...			
206	Gietz	AC_ACCOUNT	07-JUN-94

20 rows selected.

- The HR department has requested a report of all employees and their job IDs. Display the last name concatenated with the job ID (separated by a comma and space) and name the column Employee and Title.

Employee and Title
King, AD_PRES
Kochhar, AD_VP
De Haan, AD_VP
Hunold, IT_PROG
Ernst, IT_PROG
Lorentz, IT_PROG
Mourgos, ST_MAN
Rajs, ST_CLERK
Davies, ST_CLERK
...
Gietz, AC_ACCOUNT

20 rows selected.

Practice 1 (continued)

If you want an extra challenge, complete the following exercise:

12. To familiarize yourself with the data in the EMPLOYEES table, create a query to display all the data from that table. Separate each column output by a comma. Name the column title THE_OUTPUT.

THE_OUTPUT													
100	Steven	King	SKING	515.123.4567	AD_PRES		17-JUN-87	24000		90			
101	Neena	Kochhar	NKOCHHAR	515.123.4568	AD_VP	100	21-SEP-89	17000		90			
102	Lex	De Haan	LDEHAAN	515.123.4569	AD_VP	100	13-JAN-93	17000		90			
103	Alexander	Hunold	AHUNOLD	590.423.4567	IT_PROG	102	03-JAN-90	9000		60			
104	Bruce	Ernst	BERNST	590.423.4568	IT_PROG	103	21-MAY-91	6000		60			
107	Diana	Lorentz	DLORENTZ	590.423.5567	IT_PROG	103	07-FEB-99	4200		60			
124	Kevin	Mourgos	KMOURGOS	650.123.5234	ST_MAN	100	16-NOV-99	5800		50			
141	Trenna	Rajs	TRAJS	650.121.8009	ST_CLERK	124	17-OCT-95	3500		50			
142	Curtis	Davies	CDAVIES	650.121.2994	ST_CLERK	124	29-JAN-97	3100		50			
143	Randall	Matos	RMATOS	650.121.2874	ST_CLERK	124	15-MAR-98	2600		50			
144	Peter	Vargas	PVARGAS	650.121.2004	ST_CLERK	124	09-JUL-98	2500		50			
■ ■ ■													
206	William	Gietz	WGIETZ	515.123.8181	AC_ACCOUNT	205	07-JUN-94	8300		110			

20 rows selected.

Practice 2: Overview

This practice covers the following topics:

- **Selecting data and changing the order of the rows that are displayed**
- **Restricting rows by using the `WHERE` clause**
- **Sorting rows by using the `ORDER BY` clause**
- **Using substitution variables to add flexibility to your `SQL SELECT` statements**

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Practice 2: Overview

In this practice, you build more reports, including statements that use the `WHERE` clause and the `ORDER BY` clause. You make the `SQL` statements more reusable and generic by including ampersand substitution.

Practice 2

The HR department needs your assistance with creating some queries.

1. Due to budget issues, the HR department needs a report that displays the last name and salary of employees who earn more than \$12,000. Place your SQL statement in a text file named `lab_02_01.sql`. Run your query.

LAST_NAME	SALARY
King	24000
Kochhar	17000
De Haan	17000
Hartstein	13000

2. Create a report that displays the last name and department number for employee number 176.

LAST_NAME	DEPARTMENT_ID
Taylor	80

3. The HR departments needs to find high-salary and low-salary employees. Modify `lab_02_01.sql` to display the last name and salary for any employee whose salary is not in the range of \$5,000 to \$12,000. Place your SQL statement in a text file named `lab_02_03.sql`.

LAST_NAME	SALARY
King	24000
Kochhar	17000
De Haan	17000
Lorentz	4200
Rajs	3500
Davies	3100
Matos	2600
Vargas	2500
Whalen	4400
Hartstein	13000

10 rows selected.

Practice 2 (continued)

4. Create a report to display the last name, job ID, and start date for the employees with the last names of Matos and Taylor. Order the query in ascending order by start date.

LAST_NAME	JOB_ID	HIRE_DATE
Matos	ST_CLERK	15-MAR-98
Taylor	SA_REP	24-MAR-98

5. Display the last name and department number of all employees in departments 20 or 50 in ascending alphabetical order by name.

LAST_NAME	DEPARTMENT_ID
Davies	50
Fay	20
Hartstein	20
Matos	50
Mourgos	50
Rajs	50
Vargas	50

7 rows selected.

6. Modify lab_02_03.sql to display the last name and salary of employees who earn between \$5,000 and \$12,000 and are in department 20 or 50. Label the columns Employee and Monthly Salary, respectively. Resave lab_02_03.sql as lab_02_06.sql. Run the statement in lab_02_06.sql.

Employee	Monthly Salary
Fay	6000
Mourgos	5800

Practice 2 (continued)

7. The HR department needs a report that displays the last name and hire date for all employees who were hired in 1994.

LAST_NAME	HIRE_DATE
Higgins	07-JUN-94
Gietz	07-JUN-94

8. Create a report to display the last name and job title of all employees who do not have a manager.

LAST_NAME	JOB_ID
King	AD_PRES

9. Create a report to display the last name, salary, and commission of all employees who earn commissions. Sort data in descending order of salary and commissions.

LAST_NAME	SALARY	COMMISSION_PCT
Abel	11000	.3
Zlotkey	10500	.2
Taylor	8600	.2
Grant	7000	.15

10. Members of the HR department want to have more flexibility with the queries that you are writing. They would like a report that displays the last name and salary of employees who earn more than an amount that the user specifies after a prompt. (You can use the query that you created in practice exercise 1 and modify it.) Save this query to a file named `lab_02_10.sql`. If you enter 12000 when prompted, the report displays the following results:

LAST_NAME	SALARY
King	24000
Kochhar	17000
De Haan	17000
Hartstein	13000

Practice 2 (continued)

11. The HR department wants to run reports based on a manager. Create a query that prompts the user for a manager ID and generates the employee ID, last name, salary, and department for that manager's employees. The HR department wants the ability to sort the report on a selected column. You can test the data with the following values:

manager ID = 103, sorted by employee last name:

EMPLOYEE_ID	LAST_NAME	SALARY	DEPARTMENT_ID
104	Ernst	6000	60
107	Lorentz	4200	60

manager ID = 201, sorted by salary:

EMPLOYEE_ID	LAST_NAME	SALARY	DEPARTMENT_ID
202	Fay	6000	20

manager ID = 124, sorted by employee ID:

EMPLOYEE_ID	LAST_NAME	SALARY	DEPARTMENT_ID
141	Rajs	3500	50
142	Davies	3100	50
143	Matos	2600	50
144	Vargas	2500	50

Practice 2 (continued)

If you have time, complete the following exercises:

12. Display all employee last names in which the third letter of the name is *a*.

LAST_NAME
Grant
Whalen

13. Display the last name of all employees who have both an *a* and an *e* in their last name.

LAST_NAME
Davies
De Haan
Hartstein
Whalen

If you want an extra challenge, complete the following exercises:

14. Display the last name, job, and salary for all employees whose job is sales representative or stock clerk and whose salary is not equal to \$2,500, \$3,500, or \$7,000.

LAST_NAME	JOB_ID	SALARY
Abel	SA_REP	11000
Taylor	SA_REP	8600
Davies	ST_CLERK	3100
Matos	ST_CLERK	2600

15. Modify `lab_02_06.sql` to display the last name, salary, and commission for all employees whose commission amount is 20%. Resave `lab_02_06.sql` as `lab_02_15.sql`. Rerun the statement in `lab_02_15.sql`.

Employee	Monthly Salary	COMMISSION_PCT
Zlotkey	10500	.2
Taylor	8600	.2

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**

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Practice 3: Overview of Part 1

Part 1 of this lesson's practice provides a variety of exercises using different functions that are available for character, number, and date data types.

For Part 1, complete questions 1–6 at the end of this lesson.

Practice 3

Part 1

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

Date
31-DEC-03

2. The HR department needs a report to display the employee number, last name, salary, and salary increased by 15.5% (expressed as a whole number) for each employee. Label the column `New Salary`. Place your SQL statement in a text file named `lab_03_02.sql`.
3. Run your query in the file `lab_03_02.sql`.

EMPLOYEE_ID	LAST_NAME	SALARY	New Salary
100	King	24000	27720
101	Kochhar	17000	19635
...
202	Fay	6000	6930
205	Higgins	12000	13860
206	Gietz	8300	9587

20 rows selected.

4. Modify your query `lab_03_02.sql` to add a column that subtracts the old salary from the new salary. Label the column `Increase`. Save the contents of the file as `lab_03_04.sql`. Run the revised query.

EMPLOYEE_ID	LAST_NAME	SALARY	New Salary	Increase
100	King	24000	27720	3720
101	Kochhar	17000	19635	2635
102	De Haan	17000	19635	2635
...
202	Fay	6000	6930	930
205	Higgins	12000	13860	1860
206	Gietz	8300	9587	1287

20 rows selected.

Practice 3 (continued)

5. 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.

Name	Length
Abel	4
Matos	5
Mourgos	7

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*.

Name	Length
Hartstein	9
Higgins	7
Hunold	6

Oracle Internal & OAI Use Only

Practice 3 (continued)

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.

Note: Your results will differ.

LAST_NAME	MONTHS_WORKED
Zlotkey	47
Mourgos	50
Grant	55
Lorentz	59
Vargas	66
Taylor	69
Matos	70
Fay	76
Davies	83
Abel	92
Hartstein	94
Rajs	98
Higgins	115
Gietz	115
De Haan	132
Ernst	151
Hunold	168
Kochhar	171
Whalen	195
King	198

20 rows selected.

Practice 3: Overview of Part 2

This practice covers the following topics:

- **Creating queries that require the use of numeric, character, and date functions**
- **Using concatenation with functions**
- **Writing case-insensitive queries to test the usefulness of character functions**
- **Performing calculations of years and months of service for an employee**
- **Determining the review date for an employee**

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Practice 3: Overview of Part 2

Part 2 of this lesson's practice provides a variety of exercises using different functions that are available for character, number, and date data types. For Part 2, complete exercises 7–14. Remember that for nested functions, the results are evaluated from the innermost function to the outermost function.

Practice 3 (continued)

Part 2

7. Create a report that produces the following for each employee:
<employee last name> earns <salary> monthly but wants <3 times salary>. Label the column Dream Salaries.

Dream Salaries	
King	King earns \$24,000.00 monthly but wants \$72,000.00.
Kochhar	Kochhar earns \$17,000.00 monthly but wants \$51,000.00.
De Haan	De Haan earns \$17,000.00 monthly but wants \$51,000.00.
...	
Hartstein	Hartstein earns \$13,000.00 monthly but wants \$39,000.00.
Fay	Fay earns \$6,000.00 monthly but wants \$18,000.00.
Higgins	Higgins earns \$12,000.00 monthly but wants \$36,000.00.
Gietz	Gietz earns \$8,300.00 monthly but wants \$24,900.00.

20 rows selected.

If you have time, complete the following exercises:

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.

LAST_NAME	SALARY
King	\$\$\$\$\$\$\$\$\$24000
Kochhar	\$\$\$\$\$\$\$\$\$17000
De Haan	\$\$\$\$\$\$\$\$\$17000
Hunold	\$\$\$\$\$\$\$\$\$9000
...	
Fay	\$\$\$\$\$\$\$\$\$6000
Higgins	\$\$\$\$\$\$\$\$\$12000
Gietz	\$\$\$\$\$\$\$\$\$8300

20 rows selected.

Practice 3 (continued)

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."

LAST_NAME	HIRE_DATE	REVIEW
King	17-JUN-87	Monday, the Twenty-First of December, 1987
Kochhar	21-SEP-89	Monday, the Twenty-Sixth of March, 1990
De Haan	13-JAN-93	Monday, the Nineteenth of July, 1993
Hunold	03-JAN-90	Monday, the Ninth of July, 1990
Ernst	21-MAY-91	Monday, the Twenty-Fifth of November, 1991
Lorentz	07-FEB-99	Monday, the Ninth of August, 1999
...		
Higgins	07-JUN-94	Monday, the Twelfth of December, 1994
Gietz	07-JUN-94	Monday, the Twelfth of December, 1994

20 rows selected.

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.

LAST_NAME	HIRE_DATE	DAY
Grant	24-MAY-99	MONDAY
Ernst	21-MAY-91	TUESDAY
Mourgos	16-NOV-99	TUESDAY
Taylor	24-MAR-98	TUESDAY
...		
Lorentz	07-FEB-99	SUNDAY
Fay	17-AUG-97	SUNDAY
Matos	15-MAR-98	SUNDAY

20 rows selected.

Practice 3 (continued)

If you want an extra challenge, complete the following exercises:

11. Create a query that displays the employees' last names and commission amounts. If an employee does not earn commission, show "No Commission." Label the column COMM.

LAST_NAME	COMM
King	No Commission
Kochhar	No Commission
...	
Zlotkey	.2
Abel	.3
Taylor	.2
Grant	.15
Whalen	No Commission
Hartstein	No Commission
Fay	No Commission
Higgins	No Commission
Gietz	No Commission

20 rows selected.

12. Create a query that displays the first eight characters of the employees' last names and indicates the amounts of their salaries with asterisks. Each asterisk signifies a thousand dollars. Sort the data in descending order of salary. Label the column EMPLOYEES_AND_THEIR_SALARIES.

EMPLOYEES_AND_THEIR_SALARIES
King *****
Kochhar *****
De Haan *****
Hartstei *****
Higgins *****
...
Matos **
Vargas **

20 rows selected.

Practice 3 (continued)

13. Using the DECODE function, write a query that displays the grade of all employees based on the value of the column JOB_ID, using the following data:

<i>Job</i>	<i>Grade</i>
AD_PRES	A
ST_MAN	B
IT_PROG	C
SA_REP	D
ST_CLERK	E
None of the above	0

JOB_ID	GRA
AC_ACCOUNT	0
AC_MGR	0
AD_ASST	0
AD_PRES	A
AD_VP	0
AD_VP	0
IT_PROG	C
IT_PROG	C
IT_PROG	C
MK_MAN	0
MK_REP	0
SA_MAN	0
SA_REP	D
SA_REP	D
SA_REP	D
ST_CLERK	E
ST_CLERK	E
ST_CLERK	E
ST_CLERK	E
ST_MAN	B

20 rows selected.

14. Rewrite the statement in the preceding exercise using the CASE syntax.

Practice 4: Overview

This practice covers the following topics:

- **Writing queries that use the group functions**
- **Grouping by rows to achieve more than one result**
- **Restricting groups by using the `HAVING` clause**

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Practice 4: Overview

At the end of this practice, you should be familiar with using group functions and selecting groups of data.

Practice 4

Determine the validity of the following three statements. Circle either True or False.

1. Group functions work across many rows to produce one result per group.
True/False
2. Group functions include nulls in calculations.
True/False
3. The WHERE clause restricts rows prior to inclusion in a group calculation.
True/False

The HR department needs the following reports:

4. Find the highest, lowest, sum, and average salary of all employees. Label the columns Maximum, Minimum, Sum, and Average, respectively. Round your results to the nearest whole number. Place your SQL statement in a text file named lab_04_04.sql.

Maximum	Minimum	Sum	Average
24000	2500	175500	8775

5. Modify the query in lab_04_04.sql to display the minimum, maximum, sum, and average salary for each job type. Resave lab_04_04.sql as lab_04_05.sql. Run the statement in lab_04_05.sql.

JOB_ID	Maximum	Minimum	Sum	Average
AC_ACCOUNT	8300	8300	8300	8300
AC_MGR	12000	12000	12000	12000
AD_ASST	4400	4400	4400	4400
AD PRES	24000	24000	24000	24000
AD_VP	17000	17000	34000	17000
IT_PROG	9000	4200	19200	6400
MK_MAN	13000	13000	13000	13000
MK_REP	6000	6000	6000	6000
SA_MAN	10500	10500	10500	10500
SA_REP	11000	7000	26600	8867
ST_CLERK	3500	2500	11700	2925
ST_MAN	5800	5800	5800	5800

12 rows selected.

Practice 4 (continued)

6. Write a query to display the number of people with the same job.

JOB_ID	COUNT(*)
AC_ACCOUNT	1
AC_MGR	1
AD_ASST	1
AD PRES	1
AD_VP	2
IT_PROG	3
MK_MAN	1
MK_REP	1
SA_MAN	1
SA_REP	3
ST_CLERK	4
ST_MAN	1

12 rows selected.

Generalize the query so that the user in the HR department is prompted for a job title. Save the script to a file named lab_04_06.sql.

7. Determine the number of managers without listing them. Label the column Number of Managers. *Hint: Use the MANAGER_ID column to determine the number of managers.*

Number of Managers
8

8. Find the difference between the highest and lowest salaries. Label the column DIFFERENCE.

DIFFERENCE
21500

If you have time, complete the following exercises:

9. Create a report to display the manager number and the salary of the lowest-paid employee for that manager. Exclude anyone whose manager is not known. Exclude any groups where the minimum salary is \$6,000 or less. Sort the output in descending order of salary.

MANAGER_ID	MIN(SALARY)
102	9000
205	8300
149	7000

Practice 4 (continued)

If you want an extra challenge, complete the following exercises:

10. Create a query to display the total number of employees and, of that total, the number of employees hired in 1995, 1996, 1997, and 1998. Create appropriate column headings.

TOTAL	1995	1996	1997	1998
20	1	2	2	3

11. Create a matrix query to display the job, the salary for that job based on department number, and the total salary for that job, for departments 20, 50, 80, and 90, giving each column an appropriate heading.

Job	Dept 20	Dept 50	Dept 80	Dept 90	Total
AC_ACCOUNT					8300
AC_MGR					12000
AD_ASST					4400
AD_PRES				24000	24000
AD_VP				34000	34000
IT_PROG					19200
MK_MAN	13000				13000
MK_REP	6000				6000
SA_MAN			10500		10500
SA_REP			19600		26600
ST_CLERK		11700			11700
ST_MAN		5800			5800

12 rows selected.

Practice 5: Overview

This practice covers the following topics:

- **Joining tables using an equijoin**
- **Performing outer and self-joins**
- **Adding conditions**

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Practice 5: Overview

This practice is intended to give you practical experience in extracting data from more than one table using SQL:1999-compliant joins.

Practice 5

1. Write a query for the HR department to produce the addresses of all the departments. Use the `LOCATIONS` and `COUNTRIES` tables. Show the location ID, street address, city, state or province, and country in the output. Use a `NATURAL JOIN` to produce the results.

LOCATION_ID	STREET_ADDRESS	CITY	STATE_PROVINCE	COUNTRY_NAME
1400	2014 Jabberwocky Rd	Southlake	Texas	United States of America
1500	2011 Interiors Blvd	South San Francisco	California	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1800	460 Bloor St. W.	Toronto	Ontario	Canada
2500	Magdalen Centre, The Oxford Science Park	Oxford	Oxford	United Kingdom

2. The HR department needs a report of all employees. Write a query to display the last name, department number, and department name for all employees.

LAST_NAME	DEPARTMENT_ID	DEPARTMENT_NAME
Whalen	10	Administration
Hartstein	20	Marketing
Fay	20	Marketing
Mourgos	50	Shipping
Rajs	50	Shipping
Davies	50	Shipping
Vargas	50	Shipping
■ ■ ■		
De Haan	90	Executive
Higgins	110	Accounting
Gietz	110	Accounting

19 rows selected.

Practice 5 (continued)

- The HR department needs a report of employees in Toronto. Display the last name, job, department number, and department name for all employees who work in Toronto.

LAST_NAME	JOB_ID	DEPARTMENT_ID	DEPARTMENT_NAME
Hartstein	MK_MAN	20	Marketing
Fay	MK_REP	20	Marketing

- Create a report to display employees' last name and employee number along with their manager's last name and manager number. Label the columns Employee, Emp#, Manager, and Mgr#, respectively. Place your SQL statement in a text file named lab_05_04.sql.

Employee	EMP#	Manager	Mgr#
Kochhar	101	King	100
De Haan	102	King	100
Mourgos	124	King	100
Zlotkey	149	King	100
Hartstein	201	King	100
Whalen	200	Kochhar	101
Higgins	205	Kochhar	101
Hunold	103	De Haan	102
Ernst	104	Hunold	103
Lorentz	107	Hunold	103
Rajs	141	Mourgos	124
Davies	142	Mourgos	124
Matos	143	Mourgos	124
Vargas	144	Mourgos	124
Employee	EMP#	Manager	Mgr#
Abel	174	Zlotkey	149
Taylor	176	Zlotkey	149
Grant	178	Zlotkey	149
Fay	202	Hartstein	201
Gietz	206	Higgins	205

19 rows selected.

Practice 5 (continued)

5. Modify `lab_05_04.sql` to display all employees including King, who has no manager. Order the results by the employee number. Place your SQL statement in a text file named `lab_05_05.sql`. Run the query in `lab_05_05.sql`.

Employee	EMP#	Manager	Mgr#
King	100		
Kochhar	101	King	100
De Haan	102	King	100
Hunold	103	De Haan	102
Ernst	104	Hunold	103
Lorentz	107	Hunold	103
Mourgos	124	King	100

■ ■ ■

20 rows selected.

6. Create a report for the HR department that displays employee last names, department numbers, and all the employees who work in the same department as a given employee. Give each column an appropriate label. Save the script to a file named `lab_05_06.sql`.

DEPARTMENT	EMPLOYEE	COLLEAGUE
20	Fay	Hartstein
20	Hartstein	Fay
50	Davies	Matos
50	Davies	Mourgos
50	Davies	Rajs
50	Davies	Vargas
50	Matos	Davies
50	Matos	Mourgos
50	Matos	Rajs
50	Matos	Vargas
50	Mourgos	Davies
50	Mourgos	Matos
50	Mourgos	Rajs
50	Mourgos	Vargas

■ ■ ■

42 rows selected.

Practice 5 (continued)

7. The HR department needs a report on job grades and salaries. To familiarize yourself with the `JOB_GRADES` table, first show the structure of the `JOB_GRADES` table. Then create a query that displays the name, job, department name, salary, and grade for all employees.

Name	Null?	Type
GRADE_LEVEL		VARCHAR2(3)
LOWEST_SAL		NUMBER
HIGHEST_SAL		NUMBER

LAST_NAME	JOB_ID	DEPARTMENT_NAME	SALARY	GRA
Matos	ST_CLERK	Shipping	2600	A
Vargas	ST_CLERK	Shipping	2500	A
Lorentz	IT_PROG	IT	4200	B
Mourgos	ST_MAN	Shipping	5800	B
Rajs	ST_CLERK	Shipping	3500	B
Davies	ST_CLERK	Shipping	3100	B
Whalen	AD_ASST	Administration	4400	B

■ ■ ■

19 rows selected.

If you want an extra challenge, complete the following exercises:

8. The HR department wants to determine the names of all employees who were hired after Davies. Create a query to display the name and hire date of any employee hired after employee Davies.

LAST_NAME	HIRE_DATE
Lorentz	07-FEB-99
Mourgos	16-NOV-99
Matos	15-MAR-98
Vargas	09-JUL-98
Zlotkey	29-JAN-00
Taylor	24-MAR-98
Grant	24-MAY-99
Fay	17-AUG-97

8 rows selected.

Practice 5 (continued)

9. The HR department needs to find the names and hire dates for all employees who were hired before their managers, along with their managers' names and hire dates. Save the script to a file named `lab5_09.sql`.

LAST_NAME	HIRE_DATE	LAST_NAME	HIRE_DATE
Whalen	17-SEP-87	Kochhar	21-SEP-89
Hunold	03-JAN-90	De Haan	13-JAN-93
Rajs	17-OCT-95	Mourgos	16-NOV-99
Davies	29-JAN-97	Mourgos	16-NOV-99
Matos	15-MAR-98	Mourgos	16-NOV-99
Vargas	09-JUL-98	Mourgos	16-NOV-99
Abel	11-MAY-96	Zlotkey	29-JAN-00
Taylor	24-MAR-98	Zlotkey	29-JAN-00
Grant	24-MAY-99	Zlotkey	29-JAN-00

9 rows selected.

Oracle Internal & OAI Use Only

Practice 6: Overview

This practice covers the following topics:

- **Creating subqueries to query values based on unknown criteria**
- **Using subqueries to find out which values exist in one set of data and not in another**

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Practice 6: Overview

In this practice, you write complex queries using nested SELECT statements.

Paper-Based Questions

You may want to create the inner query first for these questions. Make sure that it runs and produces the data that you anticipate before you code the outer query.

Practice 6

1. The HR department needs a query that prompts the user for an employee last name. The query then displays the last name and hire date of any employee in the same department as the employee whose name they supply (excluding that employee). For example, if the user enters Zlotkey, find all employees who work with Zlotkey (excluding Zlotkey).

LAST_NAME	HIRE_DATE
Abel	11-MAY-96
Taylor	24-MAR-98

2. Create a report that displays the employee number, last name, and salary of all employees who earn more than the average salary. Sort the results in order of ascending salary.

EMPLOYEE_ID	LAST_NAME	SALARY
103	Hunold	9000
149	Zlotkey	10500
174	Abel	11000
205	Higgins	12000
201	Hartstein	13000
101	Kochhar	17000
102	De Haan	17000
100	King	24000

8 rows selected.

3. Write a query that displays the employee number and last name of all employees who work in a department with any employee whose last name contains a *u*. Place your SQL statement in a text file named lab_06_03.sql. Run your query.

EMPLOYEE_ID	LAST_NAME
124	Mourgos
141	Rajs
142	Davies
143	Matos
144	Vargas
103	Hunold
104	Ernst
107	Lorentz

8 rows selected.

Practice 6 (continued)

- The HR department needs a report that displays the last name, department number, and job ID of all employees whose department location ID is 1700.

LAST_NAME	DEPARTMENT_ID	JOB_ID
Whalen	10	AD_ASST
King	90	AD_PRES
Kochhar	90	AD_VP
De Haan	90	AD_VP
Higgins	110	AC_MGR
Gietz	110	AC_ACCOUNT

6 rows selected.

Modify the query so that the user is prompted for a location ID. Save this to a file named `lab_06_04.sql`.

- Create a report for HR that displays the last name and salary of every employee who reports to King.

LAST_NAME	SALARY
Kochhar	17000
De Haan	17000
Mourgos	5800
Zlotkey	10500
Hartstein	13000

- Create a report for HR that displays the department number, last name, and job ID for every employee in the Executive department.

DEPARTMENT_ID	LAST_NAME	JOB_ID
90	King	AD_PRES
90	Kochhar	AD_VP
90	De Haan	AD_VP

If you have time, complete the following exercise:

- Modify the query in `lab_06_03.sql` to display the employee number, last name, and salary of all employees who earn more than the average salary and who work in a department with any employee whose last name contains a *u*. Resave `lab_06_03.sql` as `lab_06_07.sql`. Run the statement in `lab_06_07.sql`.

EMPLOYEE_ID	LAST_NAME	SALARY
103	Hunold	9000

Practice 7: Overview

In this practice, you use the set operators to create reports:

- Using the **UNION** operator
- Using the **INTERSECTION** operator
- Using the **MINUS** operator

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Practice 7: Overview

In this practice, you write queries using the set operators.

Practice 7

1. The HR department needs a list of department IDs for departments that do not contain the job ID ST_CLERK. Use set operators to create this report.

DEPARTMENT_ID	
	10
	20
	60
	80
	90
	110
	190

7 rows selected.

2. The HR department needs a list of countries that have no departments located in them. Display the country ID and the name of the countries. Use set operators to create this report.

CO	COUNTRY_NAME
DE	Germany

3. Produce a list of jobs for departments 10, 50, and 20, in that order. Display job ID and department ID using set operators.

JOB_ID	DEPARTMENT_ID
AD_ASST	10
ST_CLERK	50
ST_MAN	50
MK_MAN	20
MK_REP	20

4. Create a report that lists the employee IDs and job IDs of those employees who currently have a job title that is the same as their job title when they were initially hired by the company (that is, they changed jobs but have now gone back to doing their original job).

EMPLOYEE_ID	JOB_ID
176	SA_REP
200	AD_ASST

Practice 7 (continued)

5. The HR department needs a report with the following specifications:

- Last name and department ID of all the employees from the EMPLOYEES table, regardless of whether or not they belong to a department
- Department ID and department name of all the departments from the DEPARTMENTS table, regardless of whether or not they have employees working in them

Write a compound query to accomplish this.

LAST_NAME	DEPARTMENT_ID	TO_CHAR(NULL)
Abel	80	
Davies	50	
De Haan	90	
Ernst	60	
Fay	20	
Gietz	110	
Grant		
Hartstein	20	
Higgins	110	
Hunold	60	
King	90	
Kochhar	90	
Lorentz	60	
Matos	50	
LAST_NAME	DEPARTMENT_ID	TO_CHAR(NULL)
Mourgos	50	
Rajs	50	
Taylor	80	
Vargas	50	
Whalen	10	
Zlotkey	80	
	10	Administration
	20	Marketing
	50	Shipping
	60	IT
	80	Sales
	90	Executive
	110	Accounting
	190	Contracting

28 rows selected.

Practice 8: Overview

This practice covers the following topics:

- **Inserting rows into the tables**
- **Updating and deleting rows in the table**
- **Controlling transactions**

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Practice 8: Overview

In this practice, you add rows to the MY_EMPLOYEE table, update and delete data from the table, and control your transactions.

Practice 8

The HR department wants you to create SQL statements to insert, update, and delete employee data. As a prototype, you use the MY_EMPLOYEE table, prior to giving the statements to the HR department.

Insert data into the MY_EMPLOYEE table.

1. Run the statement in the lab_08_01.sql script to build the MY_EMPLOYEE table to be used for the lab.
2. Describe the structure of the MY_EMPLOYEE table to identify the column names.

Name	Null?	Type
ID	NOT NULL	NUMBER(4)
LAST_NAME		VARCHAR2(25)
FIRST_NAME		VARCHAR2(25)
USERID		VARCHAR2(8)
SALARY		NUMBER(9,2)

3. Create an INSERT statement to add *the first row* of data to the MY_EMPLOYEE table from the following sample data. Do not list the columns in the INSERT clause. *Do not enter all rows yet.*

ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	Patel	Ralph	rpatel	895
2	Dancs	Betty	bdancs	860
3	Biri	Ben	bbiri	1100
4	Newman	Chad	cnewman	750
5	Ropeburn	Audrey	aropebur	1550

4. Populate the MY_EMPLOYEE table with the second row of sample data from the preceding list. This time, list the columns explicitly in the INSERT clause.
5. Confirm your addition to the table.

ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	Patel	Ralph	rpatel	895
2	Dancs	Betty	bdancs	860

Practice 8 (continued)

- Write an insert statement in a dynamic reusable script file named `loademp.sql` to load rows into the `MY_EMPLOYEE` table. Concatenate the first letter of the first name and the first seven characters of the last name to produce the user ID. Save this script to a file named `lab_08_06.sql`.
- Populate the table with the next two rows of sample data by running the insert statement in the script that you created.
- Confirm your additions to the table.

ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	Patel	Ralph	rpatel	895
2	Dancs	Betty	bdancs	860
3	Biri	Ben	bbiri	1100
4	Newman	Chad	cnewman	750

- Make the data additions permanent.

Update and delete data in the `MY_EMPLOYEE` table.

- Change the last name of employee 3 to Drexler.
- Change the salary to \$1,000 for all employees who have a salary less than \$900.
- Verify your changes to the table.

ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	Patel	Ralph	rpatel	1000
2	Dancs	Betty	bdancs	1000
3	Drexler	Ben	bbiri	1100
4	Newman	Chad	cnewman	1000

- Delete Betty Dancs from the `MY_EMPLOYEE` table.
- Confirm your changes to the table.

ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	Patel	Ralph	rpatel	1000
3	Drexler	Ben	bbiri	1100
4	Newman	Chad	cnewman	1000

Practice 8 (continued)

15. Commit all pending changes.

Control data transaction to the MY_EMPLOYEE table.

16. Populate the table with the last row of sample data by using the statements in the script that you created in step 6. Run the statements in the script.
17. Confirm your addition to the table.

ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	Patel	Ralph	rpatel	1000
3	Drexler	Ben	bbiri	1100
4	Newman	Chad	cnewman	1000
5	Ropeburn	Audrey	aropebur	1550

18. Mark an intermediate point in the processing of the transaction.
19. Empty the entire table.
20. Confirm that the table is empty.
21. Discard the most recent DELETE operation without discarding the earlier INSERT operation.
22. Confirm that the new row is still intact.

ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	Patel	Ralph	rpatel	1000
3	Drexler	Ben	bbiri	1100
4	Newman	Chad	cnewman	1000
5	Ropeburn	Audrey	aropebur	1550

23. Make the data addition permanent.

Practice 9: Overview

This practice covers the following topics:

- Creating new tables
- Creating a new table by using the `CREATE TABLE AS` syntax
- Verifying that tables exist
- Dropping tables

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Practice 9: Overview

Create new tables by using the `CREATE TABLE` statement. Confirm that the new table was added to the database. Create the syntax in the command file, and then execute the command file to create the table.

Practice 9

1. Create the DEPT table based on the following table instance chart. Place the syntax in a script called lab_09_01.sql, then execute the statement in the script to create the table. Confirm that the table is created.

Column Name	ID	NAME
Key Type	Primary key	
Nulls/Unique		
FK Table		
FK Column		
Data type	NUMBER	VARCHAR2
Length	7	25

Name	Null?	Type
ID		NUMBER(7)
NAME		VARCHAR2(25)

2. Populate the DEPT table with data from the DEPARTMENTS table. Include only columns that you need.
3. Create the EMP table based on the following table instance chart. Place the syntax in a script called lab_09_03.sql, and then execute the statement in the script to create the table. Confirm that the table is created.

Column Name	ID	LAST_NAME	FIRST_NAME	DEPT_ID
Key Type				
Nulls/Unique				
FK Table				DEPT
FK Column				ID
Data type	NUMBER	VARCHAR2	VARCHAR2	NUMBER
Length	7	25	25	7

Name	Null?	Type
ID		NUMBER(7)
LAST_NAME		VARCHAR2(25)
FIRST_NAME		VARCHAR2(25)
DEPT_ID		NUMBER(7)

Practice 9 (continued)

4. Create the EMPLOYEES2 table based on the structure of the EMPLOYEES table. Include only the EMPLOYEE_ID, FIRST_NAME, LAST_NAME, SALARY, and DEPARTMENT_ID columns. Name the columns in your new table ID, FIRST_NAME, LAST_NAME, SALARY, and DEPT_ID, respectively.
5. Drop the EMP table.

Oracle Internal & OAI Use Only

Practice 10: Overview of Part 2

This practice covers the following topics:

- **Creating sequences**
- **Using sequences**
- **Creating nonunique indexes**
- **Creating synonyms**

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Practice 10: Overview of Part 2

Part 2 of this lesson's practice provides you with a variety of exercises in creating and using a sequence, an index, and a synonym.

Complete questions 7–10 at the end of this lesson.

Practice 10

Part 1

1. The staff in the HR department wants to hide some of the data in the EMPLOYEES table. They want a view called EMPLOYEES_VU based on the employee numbers, employee names, and department numbers from the EMPLOYEES table. They want the heading for the employee name to be EMPLOYEE.
2. Confirm that the view works. Display the contents of the EMPLOYEES_VU view.

EMPLOYEE_ID	EMPLOYEE	DEPARTMENT_ID
100	King	90
101	Kochhar	90
102	De Haan	90
103	Hunold	60
104	Ernst	60
107	Lorentz	60
...		
206	Gietz	110

20 rows selected.

3. Using your EMPLOYEES_VU view, write a query for the HR department to display all employee names and department numbers.

EMPLOYEE	DEPARTMENT_ID
King	90
Kochhar	90
...	
Gietz	110

20 rows selected.

Practice 10

- Department 50 needs access to its employee data. Create a view named DEPT50 that contains the employee numbers, employee last names, and department numbers for all employees in department 50. You have been asked to label the view columns EMPNO, EMPLOYEE, and DEPTNO. For security purposes, do not allow an employee to be reassigned to another department through the view.
- Display the structure and contents of the DEPT50 view.

Name	Null?	Type
EMPNO	NOT NULL	NUMBER(6)
EMPLOYEE	NOT NULL	VARCHAR2(25)
DEPTNO		NUMBER(4)

EMPNO	EMPLOYEE	DEPTNO
124	Mourgos	50
141	Rajs	50
142	Davies	50
143	Matos	50
144	Vargas	50

- Test your view. Attempt to reassign Matos to department 80.

Oracle Internal & OAI Use Only

Practice 10

Part 2

7. You need a sequence that can be used with the primary key column of the DEPT table. The sequence should start at 200 and have a maximum value of 1,000. Have your sequence increment by 10. Name the sequence DEPT_ID_SEQ.
8. To test your sequence, write a script to insert two rows in the DEPT table. Name your script lab_10_08.sql. Be sure to use the sequence that you created for the ID column. Add two departments: Education and Administration. Confirm your additions. Run the commands in your script.
9. Create a nonunique index on the NAME column in the DEPT table.
10. Create a synonym for your EMPLOYEES table. Call it EMP.

Oracle Internal & OAI Use Only

Practice 11: Overview

This practice covers the following topics:

- **Querying the dictionary views for table and column information**
- **Querying the dictionary views for constraint information**
- **Querying the dictionary views for view information**
- **Querying the dictionary views for sequence information**
- **Querying the dictionary views for synonym information**
- **Adding a comment to a table and querying the dictionary views for comment information**

ORACLE

Copyright © 2004, Oracle. All rights reserved.

Practice 11: Overview

In this practice, you query the dictionary views to find information about objects in your schema.

Practice 11

1. For a specified table, create a script that reports the column names, data types, and data types' lengths, as well as whether nulls are allowed. Prompt the user to enter the table name. Give appropriate aliases to the columns DATA_PRECISION and DATA_SCALE. Save this script in a file named lab_11_01.sql.

For example, if the user enters DEPARTMENTS, the following output results:

COLUMN_NAME	DATA_TYPE	DATA_LENGTH	PRECISION	SCALE	NUL
DEPARTMENT_ID	NUMBER	22	4	0	N
DEPARTMENT_NAME	VARCHAR2	30			N
MANAGER_ID	NUMBER	22	6	0	Y
LOCATION_ID	NUMBER	22	4	0	Y

2. Create a script that reports the column name, constraint name, constraint type, search condition, and status for a specified table. You must join the USER_CONSTRAINTS and USER_CONS_COLUMNS tables to obtain all of this information. Prompt the user to enter the table name. Save the script in a file named lab_11_02.sql.

For example, if the user enters DEPARTMENTS, the following output results:

COLUMN_NAME	CONSTRAINT_NAME	CON	SEARCH_CONDITION	STATUS
DEPARTMENT_NAME	DEPT_NAME_NN	C	"DEPARTMENT_NAME" IS NOT NULL	ENABLED
DEPARTMENT_ID	DEPT_ID_PK	P		ENABLED
LOCATION_ID	DEPT_LOC_FK	R		ENABLED
MANAGER_ID	DEPT_MGR_FK	R		ENABLED

3. Add a comment to the DEPARTMENTS table. Then query the USER_TAB_COMMENTS view to verify that the comment is present.

COMMENTS
Company department information including name, code, and location.

4. Find the names of all synonyms that are in your schema.

SYNONYM_NAME	TABLE_OWNER	TABLE_NAME	DB_LINK
EMP	ORA1	EMPLOYEES	

Practice 11

- You need to determine the names and definitions of all of the views in your schema. Create a report that retrieves view information: the view name and text from the USER_VIEWS data dictionary view.

Note: Another view already exists. The EMP_DETAILS_VIEW was created as part of your schema. Also, if you completed practice 10, you will see the DEPT50 view.

Note: To see more contents of a LONG column, use the *iSQL*Plus* command SET LONG *n*, where *n* is the value of the number of characters of the LONG column that you want to see.

VIEW_NAME	TEXT
EMPLOYEES_VU	SELECT employee_id, last_name employee, department_id FROM employees
EMP_DETAILS_VIEW	SELECT e.employee_id, e.job_id, e.manager_id, e.department_id, d.location_id, l.country_id, e.first_name, e.last_name, e.salary, e.commission_pct, d.department_name, j.job_title, l.city, l.state_province, c.country_name, r.region_name FROM employees e, departments d, jobs j, locations l, countries c, regions r WHERE e.department_id = d.department_id AND d.location_id = l.location_id AND l.country_id = c.country_id AND c.region_id = r.region_id AND j.job_id = e.job_id WITH READ ONLY

- Find the names of your sequences. Write a query in a script to display the following information about your sequences: sequence name, maximum value, increment size, and last number. Name the script lab_11_06.sql. Run the statement in your script.

SEQUENCE_NAME	MAX_VALUE	INCREMENT_BY	LAST_NUMBER
DEPARTMENTS_SEQ	9990	10	280
DEPT_ID_SEQ	1000	10	200
EMPLOYEES_SEQ	1.0000E+27	1	207
LOCATIONS_SEQ	9900	100	3300