DEPARTMENT OF COMPUTER SCIENCE

The University of Chittagong

Student Name:	Sumaiya Tabassum
Student ID:	23701025
Session:	2022-2023
Course Title:	Database Systems Lab
Course Code:	CSE 414
Task:	Assignment 02 - Oracle Database 11g: SQL Fundamentals I Chap 05 - 07 Exercise
Submitted to:	Rudra Pratap Deb Nath, Associate Professor, Department of Computer Science and Engineering
Submission Date:	May 30, 2025

Contents

Chapter 05

Problem 1

Q: Group functions work across many rows to produce one result per group.

A: True.

Problem 2

Q: Group functions include nulls in calculations.

A: False.

Problem 3

Q: The `WHERE` clause restricts rows before inclusion in a group calculation.

A: True.

Problem 4

Q: Find the highest, lowest, sum, and average salary of all employees. Label the columns as `Maximum`, `Minimum`, `Sum`, and `Average`, respectively. Round your results to the nearest whole number. Save your SQL statement as `lab_05_04.sql`. Run the query.

	Maximum	Minimum	Sum	Average
1	24000	2500	175500	8775

A:

`lab_05_04.sql`

```
SELECT ROUND(MAX(salary)) AS Maximum,
       ROUND(MIN(salary)) AS Minimum,
       ROUND(SUM(salary)) AS Sum,
       ROUND(AVG(salary)) AS Average
FROM hr.employees;
```

	MAXIMUM	MINIMUM	SUM	AVERAGE
1	24000	2100	691416	6462

Problem 5

Q: Modify the query in `lab_05_04.sql` to display the minimum, maximum, sum, and average salary for each job type. Resave `lab_05_04.sql` as `lab_05_05.sql`. Run the statement in `lab_05_05.sql`.

	JOB_ID	Maximum	Minimum	Sum	Average
1	IT_PROG	9000	4200	19200	6400
2	AC_MGR	12000	12000	12000	12000
3	AC_ACCOUNT	8300	8300	8300	8300
4	ST_MAN	5800	5800	5800	5800
5	AD_ASST	4400	4400	4400	4400
6	AD_VP	17000	17000	34000	17000
7	SA_MAN	10500	10500	10500	10500
8	MK_MAN	13000	13000	13000	13000
9	AD_PRES	24000	24000	24000	24000
10	SA_REP	11000	7000	26600	8867
11	MK_REP	6000	6000	6000	6000
12	ST_CLERK	3500	2500	11700	2925

A:

`lab_05_05.sql`

```
SELECT job_id, ROUND(MAX(salary)) AS Maximum,
            ROUND(MIN(salary)) AS Minimum,
            ROUND(SUM(salary)) AS Sum,
            ROUND(AVG(salary)) AS Average
FROM hr.employees
GROUP BY job_id;
```

	JOB_ID	MAXIMUM	MINIMUM	SUM	AVERAGE
1	AD_PRES	24000	24000	24000	24000
2	AD_VP	17000	17000	34000	17000
3	IT_PROG	9000	4200	28800	5760
4	FI_MGR	12008	12008	12008	12008
5	FI_ACCOUNT	9000	6900	39600	7920
6	PU_MAN	11000	11000	11000	11000
7	PU_CLERK	3100	2500	13900	2780

Problem 6

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

	JOB_ID	COUNT(*)
1	AC_ACCOUNT	1
2	AC_MGR	1
3	AD_ASST	1
4	AD_PRES	1
5	AD_VP	2
6	IT_PROG	3
7	MK_MAN	1
8	MK_REP	1
9	SA_MAN	1
10	SA_REP	3
11	ST_CLERK	4
12	ST_MAN	1

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_05_06.sql`. Run the query. Enter `IT_PROG` when prompted.

	JOB_ID	COUNT(*)
1	IT_PROG	3

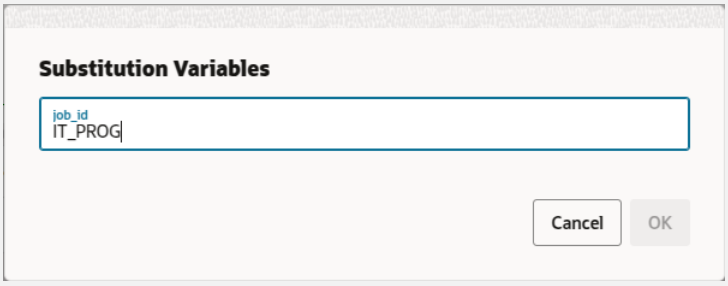
A:

```
SELECT job_id, COUNT(*)
FROM hr.employees
GROUP BY job_id;
```

	JOB_ID	COUNT(*)
1	AC_ACCOUNT	1
2	AC_MGR	1
3	AD_ASST	1
4	AD_PRES	1
5	AD_VP	2
6	FI_ACCOUNT	5
7	FI_MGR	1

lab_05_06.sql

```
SELECT job_id, COUNT(*) AS employee_count
FROM hr.employees
WHERE job_id = '&job_id'
GROUP BY job_id;
```



	JOB_ID	EMPLOYEE_COUNT
1	IT_PROG	5

Problem 7

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

	Number of Managers
1	8

A:

```
SELECT COUNT(DISTINCT manager_id) AS "Number of Managers"
FROM hr.employees;
```

	NUMBER OF MANAGERS
1	18

Problem 8

Q: Find the difference between the highest and lowest salaries. Label the column `DIFFERENCE`.

	DIFFERENCE
1	21500

A:

```
SELECT (MAX(salary) - MIN(salary)) AS DIFFERENCE
FROM hr.employees;
```

	DIFFERENCE
1	21900

Problem 9

Q: 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)
1	102	9000
2	205	8300
3	149	7000

A:

```
SELECT manager_id, MIN(salary)
FROM hr.employees
WHERE manager_id IS NOT NULL
GROUP BY manager_id
HAVING MIN(salary) > 6000
ORDER BY MIN(salary) DESC;
```

	MANAGER_ID	MIN(SALARY)
1	102	9000
2	205	8300
3	145	7000
4	146	7000
5	108	6900
6	147	6200
7	149	6200

Problem 10

Q: 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
1	20	1	2	2	3

A: [Since records from 1995, 1996, 1997, 1998 does not exist anymore, we will use records from 2015, 2016, 2017, 2018]

```
SELECT
  COUNT(*) AS total_employees,
  SUM(CASE WHEN TO_CHAR(hire_date, 'YYYY') = '2015' THEN 1 ELSE 0 END)
      AS "2015",
  SUM(CASE WHEN TO_CHAR(hire_date, 'YYYY') = '2016' THEN 1 ELSE 0 END)
      AS "2016",
  SUM(CASE WHEN TO_CHAR(hire_date, 'YYYY') = '2017' THEN 1 ELSE 0 END)
      AS "2017",
  SUM(CASE WHEN TO_CHAR(hire_date, 'YYYY') = '2018' THEN 1 ELSE 0 END)
      AS "2018"
FROM hr.employees;
```

	TOTAL_EMPLOYEES	2015	2016	2017	2018
1	107	29	24	19	11

Problem 11

Q: 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
1	IT_PROG	(null)	(null)	(null)	(null)	19200
2	AC_MGR	(null)	(null)	(null)	(null)	12000
3	AC_ACCOUNT	(null)	(null)	(null)	(null)	8300
4	ST_MAN	(null)	5800	(null)	(null)	5800
5	AD_ASST	(null)	(null)	(null)	(null)	4400
6	AD_VP	(null)	(null)	(null)	34000	34000
7	SA_MAN	(null)	(null)	10500	(null)	10500
8	MK_MAN	13000	(null)	(null)	(null)	13000
9	AD_PRES	(null)	(null)	(null)	24000	24000
10	SA_REP	(null)	(null)	19600	(null)	26600
11	MK_REP	6000	(null)	(null)	(null)	6000
12	ST_CLERK	(null)	11700	(null)	(null)	11700

A:

```
SELECT
   job_id,
   SUM(DECODE(department_id, 20, salary, NULL)) AS "Dept 20",
   SUM(DECODE(department_id, 50, salary, NULL)) AS "Dept 50",
   SUM(DECODE(department_id, 80, salary, NULL)) AS "Dept 80",
   SUM(DECODE(department_id, 90, salary, NULL)) AS "Dept 90",
   SUM(salary) AS "Total"
FROM hr.employees
GROUP BY job_id;
```

	JOB_ID	DEPT 20	DEPT 50	DEPT 80	DEPT 90	TOTAL
1	AD_PRES	*(null)*	*(null)*	*(null)*	24000	24000
2	AD_VP	*(null)*	*(null)*	*(null)*	34000	34000
3	IT_PROG	*(null)*	*(null)*	*(null)*	*(null)*	28800
4	FI_MGR	*(null)*	*(null)*	*(null)*	*(null)*	12008
5	FI_ACCOUNT	*(null)*	*(null)*	*(null)*	*(null)*	39600
6	PU_MAN	*(null)*	*(null)*	*(null)*	*(null)*	11000
7	PU_CLERK	*(null)*	*(null)*	*(null)*	*(null)*	13900
8	ST_MAN	*(null)*	36400	*(null)*	*(null)*	36400
9	ST_CLERK	*(null)*	55700	*(null)*	*(null)*	55700
10	SA_MAN	*(null)*	*(null)*	61000	*(null)*	61000
11	SA_REP	*(null)*	*(null)*	243500	*(null)*	250500
12	SH_CLERK	*(null)*	64300	*(null)*	*(null)*	64300

Chapter 06

Problem 1

Q: 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
1	1400	2014 Jabberwocky Rd	Southlake	Texas	United States of America
2	1500	2011 Interiors Blvd	South San Francisco	California	United States of America
3	1700	2004 Charade Rd	Seattle	Washington	United States of America
4	1800	460 Bloor St. W.	Toronto	Ontario	Canada
5	2500	Magdalen Centre, The ...	Oxford	Oxford	United Kingdom

A:

```
SELECT location_id, street_address, city, state_province, country_name
FROM hr.locations
NATURAL JOIN hr.countries;
```

	LOCATION_ID	STREET_ADDRESS	CITY	STATE_PROVINCE	COUNTRY_NAME
1	1000	1297 Via Cola di Rie	Roma	*(null)*	Italy
2	1100	93091 Calle della Testa	Venice	*(null)*	Italy
3	1200	2017 Shinjuku-ku	Tokyo	Tokyo Prefecture	Japan
4	1300	9450 Kamiya-cho	Hiroshima	*(null)*	Japan
5	1400	2014 Jabberwocky Rd	Southlake	Texas	United States of America
6	1500	2011 Interiors Blvd	South San Francisco	California	United States of America
7	1600	2007 Zagora St	South Brunswick	New Jersey	United States of America
8	1700	2004 Charade Rd	Seattle	Washington	United States of America

Problem 2

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

	LAST_NAME	DEPARTMENT_ID	DEPARTMENT_NAME
1	Whalen	10	Administration
2	Hartstein	20	Marketing
3	Fay	20	Marketing
4	Davies	50	Shipping
5	Vargas	50	Shipping
6	Rajs	50	Shipping
7	Mourgos	50	Shipping
8	Matos	50	Shipping
9	Hunold	60	IT
10	Ernst	60	IT

...

18	Higgins	110	Accounting
19	Gietz	110	Accounting

A:

```
SELECT e.last_name,
     e.department_id,
     d.department_name
FROM hr.employees e
JOIN hr.departments d
ON (e.department_id = d.department_id);
```

	LAST_NAME	DEPARTMENT_ID	DEPARTMENT_NAME
1	Whalen	10	Administration
2	Martinez	20	Marketing
3	Davis	20	Marketing
4	Baida	30	Purchasing
5	Tobias	30	Purchasing
6	Li	30	Purchasing
7	Khoo	30	Purchasing

Problem 3

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

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

A:

`lab2_3.sql`

```
SELECT last_name, job_id, department_id, department_name
FROM hr.employees
JOIN hr.departments USING (department_id)
JOIN hr.locations USING (location_id)
WHERE city = 'Toronto';
```

	LAST_NAME	JOB_ID	DEPARTMENT_ID	DEPARTMENT_NAME
1	Martinez	MK_MAN	20	Marketing
2	Davis	MK_REP	20	Marketing

Problem 4

Q: 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. Save your SQL statement as `lab_06_04.sql`. Run the query.

	Employee	EMP#	Manager	Mgr#
1	Kochhar	101	King	100
2	De Haan	102	King	100
3	Hunold	103	De Haan	102
4	Ernst	104	Hunold	103
5	Lorentz	107	Hunold	103
6	Mourgos	124	King	100
7	Rajs	141	Mourgos	124
8	Davies	142	Mourgos	124
9	Matos	143	Mourgos	124
10	Vargas	144	Mourgos	124

▪ ▪ ▪

15	Whalen	200	Kochhar	101
16	Hartstein	201	King	100
17	Fay	202	Hartstein	201
18	Higgins	205	Kochhar	101
19	Gietz	206	Higgins	205

A:

lab_06_04.sql

```
SELECT emp.last_name AS Employee,
     emp.employee_id AS EMP#,
     mgr.last_name AS Manager,
     mgr.employee_id AS Mgr#
FROM hr.employees emp
JOIN hr.employees mgr
ON (emp.manager_id = mgr.employee_id);
```

	EMPLOYEE	EMP#	MANAGER	MGR#
1	Ozer	168	Cambrault	148
2	Bloom	169	Cambrault	148
3	Fox	170	Cambrault	148
4	Smith	171	Cambrault	148
5	Bates	172	Cambrault	148
6	Kumar	173	Cambrault	148
7	Vishney	162	Errazuriz	147
8	Greene	163	Errazuriz	147

Problem 5

Q: Modify `lab_06_04.sql` to display all employees including King, who has no manager. Order the results by the employee number. Save your SQL statement as `lab_06_05.sql`. Run the query in `lab_06_05.sql`.

	Employee	EMP#	Manager	Mgr#
1	King	100	(null)	(null)
2	Kochhar	101	King	100
3	De Haan	102	King	100
4	Hunold	103	De Haan	102
5	Ernst	104	Hunold	103
6	Lorentz	107	Hunold	103
7	Mourgos	124	King	100
8	Rajs	141	Mourgos	124
9	Davies	142	Mourgos	124
10	Matos	143	Mourgos	124

▪ ▪ ▪

18	Fay	202	Hartstein	201
19	Higgins	205	Kochhar	101
20	Gietz	206	Higgins	205

A:

lab_06_05.sql

```
SELECT emp.last_name AS Employee,
     emp.employee_id AS EMP#,
     mgr.last_name AS Manager,
     mgr.employee_id AS Mgr#

FROM hr.employees emp
LEFT OUTER JOIN hr.employees mgr
ON (emp.manager_id = mgr.employee_id)
ORDER BY emp.employee_id;
```

	EMPLOYEE	EMP#	MANAGER	MGR#
1	King	100	*(null)*	*(null)*
2	Yang	101	King	100
3	Garcia	102	King	100
4	James	103	Garcia	102
5	Miller	104	James	103
6	Williams	105	James	103
7	Jackson	106	James	103
8	Nguyen	107	James	103
9	Gruenberg	108	Yang	101
10	Faviet	109	Gruenberg	108

Problem 6

Q: 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_06_06.sql.

	DEPARTMENT	EMPLOYEE	COLLEAGUE
1	20	Fay	Hartstein
2	20	Hartstein	Fay
3	50	Davies	Matos
4	50	Davies	Mourgos
5	50	Davies	Rajs
6	50	Davies	Vargas
7	50	Matos	Davies
8	50	Matos	Mourgos
9	50	Matos	Rajs
10	50	Matos	Vargas
...			
42	110	Higgins	Gietz

A:

lab_06_06.sql

```
SELECT emp.department_id AS DEPARTMENT,
     emp.last_name AS EMPLOYEE,
     col.last_name AS COLLEAGUE
FROM hr.employees emp
JOIN hr.employees col
ON (emp.department_id = col.department_id)
ORDER BY emp.last_name;
```

	DEPARTMENT	EMPLOYEE	COLLEAGUE
1	80	Abel	McEwen
2	80	Abel	Olsen
3	80	Abel	Ozer
4	80	Abel	Partners
5	80	Abel	Sewall
6	80	Abel	Singh
7	80	Abel	Smith

86	50	Atkinson	Venzl
87	50	Atkinson	Vollman
88	50	Atkinson	Walsh
89	50	Atkinson	Weiss

Problem 7

Q: 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.

```
DESC JOB_GRADES
Name                          Null     Type
----------------------------- -------- --------------
GRADE_LEVEL                            VARCHAR2(3)
LOWEST_SAL                             NUMBER
HIGHEST_SAL                            NUMBER

3 rows selected
```

	LAST_NAME	JOB_ID	DEPARTMENT_NAME	SALARY	GRADE_LEVEL
1	Vargas	ST_CLERK	Shipping	2500	A
2	Matos	ST_CLERK	Shipping	2600	A
3	Davies	ST_CLERK	Shipping	3100	B
4	Rajs	ST_CLERK	Shipping	3500	B
5	Lorentz	IT_PROG	IT	4200	B
6	Whalen	AD_ASST	Administration	4400	B
7	Mourgos	ST_MAN	Shipping	5800	B
8	Ernst	IT_PROG	IT	6000	C
9	Fay	MK_REP	Marketing	6000	C
10	Gietz	AC_ACCOUNT	Accounting	8300	C

▪ ▪ ▪

18	De Haan	AD_VP	Executive	17000	E
19	King	AD_PRES	Executive	24000	E

A:

```
There's no job_grade table in the oracle livesql
```

Problem 8

Q: The HR department wants to determine the names of all the 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
1	Lorentz	07-FEB-99
2	Mourgos	16-NOV-99
3	Matos	15-MAR-98
4	Vargas	09-JUL-98
5	Zlotkey	29-JAN-00
6	Taylor	24-MAR-98
7	Grant	24-MAY-99
8	Fay	17-AUG-97

A:

```
SELECT last_name, hire_date
from hr.employees
WHERE hire_date > (SELECT hire_date
                   FROM hr.employees
                   WHERE last_name = 'Davies'
                   );
```

	LAST_NAME	HIRE_DATE
1	Yang	9/21/2015, 12:00:00 AM
2	James	1/3/2016, 12:00:00 AM
3	Miller	5/21/2017, 12:00:00 AM
4	Williams	6/25/2015, 12:00:00 AM
5	Jackson	2/5/2016, 12:00:00 AM
6	Nguyen	2/7/2017, 12:00:00 AM
7	Chen	9/28/2015, 12:00:00 AM

Problem 9

Q: The HR department needs to find the names and hire dates of all the employees who were hired before their managers, along with their managers' names and hire dates. Save the script to a file named `lab_06_09.sql`.

	LAST_NAME	HIRE_DATE	LAST_NAME_1	HIRE_DATE_1
1	Whalen	17-SEP-87	Kochhar	21-SEP-89
2	Hunold	03-JAN-90	De Haan	13-JAN-93
3	Vargas	09-JUL-98	Mourgos	16-NOV-99
4	Matos	15-MAR-98	Mourgos	16-NOV-99
5	Davies	29-JAN-97	Mourgos	16-NOV-99
6	Rajs	17-OCT-95	Mourgos	16-NOV-99
7	Grant	24-MAY-99	Zlotkey	29-JAN-00
8	Taylor	24-MAR-98	Zlotkey	29-JAN-00
9	Abel	11-MAY-96	Zlotkey	29-JAN-00

A:

lab_06_09.sql

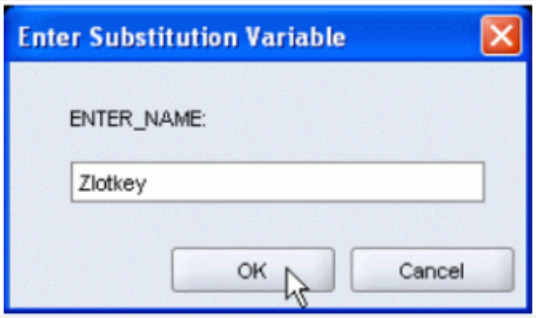
```
SELECT emp.last_name,
     emp.hire_date,
     mgr.last_name AS LAST_NAME_1,
     mgr.hire_date AS HIRE_DATE_1
FROM hr.employees emp
JOIN hr.employees mgr
ON (emp.manager_id = mgr.employee_id)
WHERE emp.hire_date < mgr.hire_date;
```

	LAST_NAME	HIRE_DATE	LAST_NAME_1	HIRE_DATE_1
1	Garcia	1/13/2011, 12:00:00 AM	King	6/17/2013, 12:00:00 AM
2	Li	12/7/2012, 12:00:00 AM	King	6/17/2013, 12:00:00 AM
3	Kaufling	5/1/2013, 12:00:00 AM	King	6/17/2013, 12:00:00 AM
4	Gruenberg	8/17/2012, 12:00:00 AM	Yang	9/21/2015, 12:00:00 AM
5	Whalen	9/17/2013, 12:00:00 AM	Yang	9/21/2015, 12:00:00 AM
6	Jacobs	6/7/2012, 12:00:00 AM	Yang	9/21/2015, 12:00:00 AM
7	Brown	6/7/2012, 12:00:00 AM	Yang	9/21/2015, 12:00:00 AM

Chapter 07

Problem 1

Q: 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
1	Abel	11-MAY-96
2	Taylor	24-MAR-98

A:

```
SELECT last_name, hire_date
FROM hr.employees
WHERE department_id = (
   SELECT department_id
   FROM hr.employees
   WHERE last_name = INITCAP('&last_name')
)
AND last_name <> INICAP('&last_name');
```

	LAST_NAME	HIRE_DATE
1	Singh	10/1/2014, 12:00:00 AM
2	Partners	1/5/2015, 12:00:00 AM
3	Errazuriz	3/10/2015, 12:00:00 AM
4	Cambrault	10/15/2017, 12:00:00 AM
5	Tucker	1/30/2015, 12:00:00 AM
6	Bernstein	3/24/2015, 12:00:00 AM
7	Hall	8/20/2015, 12:00:00 AM

Problem 2

Q: 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
1	103	Hunold	9000
2	149	Zlotkey	10500
3	174	Abel	11000
4	205	Higgins	12000
5	201	Hartstein	13000
6	101	Kochhar	17000
7	102	De Haan	17000
8	100	King	24000

A:

```
SELECT employee_id, last_name, salary
FROM hr.employees
WHERE salary > (SELECT AVG(salary)
            FROM hr.employees
            )
ORDER BY salary;
```

	EMPLOYEE_ID	LAST_NAME	SALARY
1	203	Jacobs	6500
2	123	Vollman	6500
3	165	Lee	6800
4	113	Popp	6900
5	155	Tuvault	7000
6	161	Sewall	7000
7	178	Grant	7000

Problem 3

Q: 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 the letter "u". Save your SQL statement as `lab_07_03.sql`. Run your query.

	EMPLOYEE_ID	LAST_NAME
1	124	Mourgos
2	141	Rajs
3	142	Davies
4	143	Matos
5	144	Vargas
6	103	Hunold
7	104	Ernst
8	107	Lorentz

A:

`lab_07_03.sql`

```
SELECT employee_id, last_name
FROM hr.employees
WHERE department_id IN (SELECT department_id
                        FROM hr.employees
                        WHERE last_name LIKE '%u%'
                        );
```

	EMPLOYEE_ID	LAST_NAME
1	120	Weiss
2	121	Fripp
3	122	Kaufling
4	123	Vollman
5	124	Mourgos
6	125	Nayer
7	126	Mikkilineni

Problem 4

Q: 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.
Modify the query so that the user is prompted for a location ID. Save this to a file named `lab_07_04.sql`.

	LAST_NAME	DEPARTMENT_ID	JOB_ID
1	Whalen	10	AD_ASST
2	King	90	AD_PRES
3	Kochhar	90	AD_VP
4	De Haan	90	AD_VP
5	Higgins	110	AC_MGR
6	Gietz	110	AC_ACCOUNT

A:

`lab_07_04.sql`

```
SELECT e.last_name, e.department_id, e.job_id
FROM hr.employees e
JOIN hr.departments d
  ON (e.department_id = d.department_id)
WHERE d.location_id = 1700;
```

	LAST_NAME	DEPARTMENT_ID	JOB_ID
1	King	90	AD_PRES
2	Yang	90	AD_VP
3	Garcia	90	AD_VP
4	Gruenberg	100	FI_MGR
5	Faviet	100	FI_ACCOUNT
6	Chen	100	FI_ACCOUNT
7	Sciarra	100	FI_ACCOUNT

Problem 5

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

	LAST_NAME	SALARY
1	Kochhar	17000
2	De Haan	17000
3	Mourgos	5800
4	Zlotkey	10500
5	Hartstein	13000

A:

```
SELECT last_name, salary
FROM hr.employees
WHERE manager_id = (SELECT employee_id
                    FROM hr.employees
                    WHERE last_name = 'King'
                    AND job_id = 'AD_PRES'
                    );
```

	LAST_NAME	SALARY
1	Yang	17000
2	Garcia	17000
3	Li	11000
4	Weiss	8000
5	Fripp	8200
6	Kaufling	7900
7	Vollman	6500

Problem 6

Q: 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
1	90	King	AD_PRES
2	90	Kochhar	AD_VP
3	90	De Haan	AD_VP

A:

```
SELECT department_id, last_name, job_id
FROM hr.employees
WHERE department_id = (SELECT department_id
                  FROM hr.departments
                  WHERE department_name = 'Executive'
                  );
```

	DEPARTMENT_ID	LAST_NAME	JOB_ID
1	90	King	AD_PRES
2	90	Yang	AD_VP
3	90	Garcia	AD_VP

Problem 7

Q: Modify the query in `lab_07_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_07_03.sql` as `lab_07_07.sql`. Run the statement in `lab_07_07.sql`.

	EMPLOYEE_ID	LAST_NAME	SALARY
1	103	Hunold	9000

A:

`lab_07_07.sql`

```
SELECT employee_id, last_name, salary
from hr.employees
WHERE department_id IN (SELECT department_id
                  FROM hr.employees
                  WHERE last_name LIKE '%u%'
                  )
AND salary > (SELECT AVG(salary)
           FROM hr.employees
           );
```

	EMPLOYEE_ID	LAST_NAME	SALARY
1	120	Weiss	8000
2	121	Fripp	8200
3	122	Kaufling	7900
4	123	Vollman	6500
5	145	Singh	14000
6	146	Partners	13500
7	147	Errazuriz	12000