



Faculty of Science
Department of Computer Science

| | |
|-----------------------|-------------------|
| Course Title: | Database System I |
| Course Number: | CPS 510 |
| Semester/Year | Semester 5 |

| | |
|-------------------|----------------------------|
| Instructor | Soheila Bashardoust Tajali |
| Section | 8 |

| | |
|-------------------------------|---|
| Assignment/Lab Number: | Assignment 4 |
| Assignment/Lab Title: | Demo of Designing Views/Simple Quarries |

| LAST Name | FIRST Name | Student # | Signature* |
|------------------|-------------------|------------------|-------------------|
| Hoang | Justin | 500884336 | JH |
| Syed | Omar | 500809837 | OS |
| George | Allan | 500962264 | AG |

*By signing above you attest that you have contributed to this written lab report and confirm that all work you have contributed to this lab report is your own work. Any suspicion of copying or plagiarism in this work will result in an investigation of Academic Misconduct and may result in a "0" on the work, an "F" in the course, or possibly more severe penalties, as well as a Disciplinary Notice on your academic record under the Student Code of Academic Conduct, which can be found online at: <http://www.ryerson.ca/senate/current/pol60.pdf>

COMPANY

```
1. SELECT DISTINCT PHONENUMBER
   FROM company
   ORDER BY PHONENUMBER ASC;
```

(Used by user for when they need to contact the Company for Job Application. The phone number is sorted in Ascending order)

The screenshot displays the Oracle SQL Developer interface. The 'Query Builder' window shows the following SQL query:

```
SELECT DISTINCT PHONENUMBER
FROM company
ORDER BY PHONENUMBER ASC;
```

The 'Script Output' window shows an error report:

```
Error report -
SQL Error: ORA-00936: missing expression
00936. 00000 - "missing expression"
*Cause:
*Action:
```

The 'Columns' window shows the schema for the 'COMPANY' table:

| COLUMN_NAME | DATA_TYPE | NULLABLE | DATA_DEFAULT | COLUMN_ID | COMMENTS |
|---------------|--------------------|----------|--------------|-----------|----------|
| 1 USERID | NUMBER (4, 0) | No | (null) | 1 (null) | |
| 2 EMAIL | VARCHAR2 (50 BYTE) | No | (null) | 2 (null) | |
| 3 COMPANYNAME | VARCHAR2 (30 BYTE) | No | (null) | 3 (null) | |
| 4 PHONENUMBER | VARCHAR2 (10 BYTE) | Yes | (null) | 4 (null) | |

The 'Database Connection' window shows the connection to '11g Ryerson DB'.

END_USER

```
1. SELECT ROLEID, count(distinct USERNAME)
   FROM end_user
   Group by ROLEID
```

(Used to determine the number of Jobseekers and Job Posters in the END_USER table. They will be grouped according to their role id)

The screenshot shows the SQL Developer interface with the 'END_USER' table selected. The table has columns: USERID, ROLEID, USERNAME, and EMAIL. The data is as follows:

| USERID | ROLEID | USERNAME | EMAIL |
|--------|---------|---------------|-----------------------|
| 1 | 1000 EE | PetePark | PeterParker@gmail.com |
| 2 | 1004 ER | ElonMusk | careers@tesla.com |
| 3 | 1001 EE | TheStar | TheStar@gmail.com |
| 4 | 1003 EE | HomerJSimp | HomerJSimp@gmail.com |
| 5 | 1005 ER | A.Abhari | sbtajali@ryerson.ca |
| 6 | 1006 ER | M.Myers | M.Myers@gmail.com |
| 7 | 1007 ER | L.Lang | L.Lang@gmail.com |
| 8 | 1002 EE | ScooberDoober | ScooberDoo@gmail.com |

The 'Query Builder' window shows the following SQL query:

```
SELECT ROLEID, count(distinct USERNAME)
FROM end_user
Group by ROLEID
```

The 'Script Output' window shows the results of the query:

```
PROVINCE
-----
Albert
Ontario
Quebec

RO COUNT (DISTINCTUSERNAME)
--
ER 4
EE 4
```

JOB

```
1. SELECT distinct Province
FROM job
ORDER BY province ASC;
```

(Used to determine all Provinces with Job listings available for Search. The search will be ordered by all provinces in ascending order)

The screenshot shows the SQL Developer interface with the 'JOB' table selected. The table has columns: JOBID, USERID, JOBTITLE, JOBFIELD, JOBTTYPE, SALARY, INFORM..., PROVINCE, and CO. The data is as follows:

| JOBID | USERID | JOBTITLE | JOBFIELD | JOBTTYPE | SALARY | INFORM... | PROVINCE | CO |
|-------|--------|----------------|------------|-----------|--------|------------|----------|------|
| 1 | 10000 | 1004Mechani... | Mechani... | Full-Time | 96000 | The Rol... | Ontario | Cana |
| 2 | 10001 | 1005Compute... | Education | Full-Time | 51000 | Job Sum... | Ontario | Cana |
| 3 | 10003 | 1006Sales A... | Retail | Part-Time | 22000 | The Sal... | Albert | Cana |
| 4 | 10004 | 1007Warehou... | Courier | Part-Time | 30000 | Job Sum... | Quebec | Cana |

The 'Query Builder' window shows the following SQL query:

```
SELECT distinct Province
FROM job
```

The 'Script Output' window shows the results of the query:

```
PHONENUMBE
-----
4373177623

PROVINCE
-----
Albert
Quebec
Ontario
```

JOBS APPLIED

```
1. SELECT distinct seekerid, count(jobid)
   FROM jobs_applied
   group by seekerid;
```

(Used to see how many jobs each jobseeker has applied to)

The screenshot shows a database management tool interface. On the left, the 'JOBS_APPLIED' table is displayed with columns: SEEKERID, JOBID, and COMPANYID. The table contains 5 rows of data:

| | SEEKERID | JOBID | COMPANYID |
|---|----------|-------|-----------|
| 1 | 1000 | 10000 | 1004 |
| 2 | 1002 | 10001 | 1005 |
| 3 | 1001 | 10003 | 1006 |
| 4 | 1003 | 10004 | 1007 |
| 5 | 1000 | 10003 | 1006 |

On the right, the 'Query Builder' tab shows the following SQL query:

```
SELECT distinct seekerid, count(jobid)
FROM jobs_applied
group by seekerid;
```

Below the query, the 'Script Output' tab shows the execution results. It indicates 'Task completed in 0.029 seconds' and displays two tables. The first table shows the count of jobs applied for each seeker:

| SEEKERID | COUNTI (JOBID) |
|----------|----------------|
| 1000 | 2 |
| 1001 | 1 |
| 1002 | 1 |
| 1003 | 1 |

JOBSEEKER

```
1. SELECT DISTINCT EMAIL, FIRSTNAME, LASTNAME
   FROM JOBSEEKER
   ORDER BY LASTNAME ASC;
```

(Used to display information about the Jobseeker such as email, first name, last name in ascending order.)

The screenshot shows the Oracle SQL Developer interface. In the Worksheet, a query is entered: `SELECT DISTINCT EMAIL, FIRSTNAME, LASTNAME FROM JOBSSEEKER ORDER BY LASTNAME ASC;`. The Script Output pane shows the results of this query, displaying columns EMAIL, FIRSTNAME, and LASTNAME with their respective values. The right pane shows the structure of the JOBSSEEKER table.

| COLUMN_NAME | DATA_TYPE | NULLABLE | DATA_DEFAULT | COLUMN_ID | COMMENTS |
|-------------|--------------------|----------|--------------|-----------|----------|
| 1 USERID | NUMBER | No | (null) | 1 | (null) |
| 2 EMAIL | VARCHAR2 (50 BYTE) | No | (null) | 2 | (null) |
| 3 FIRSTNAME | VARCHAR2 (30 BYTE) | No | (null) | 3 | (null) |
| 4 LASTNAME | VARCHAR2 (30 BYTE) | No | (null) | 4 | (null) |

| EMAIL | FIRSTNAME | LASTNAME |
|-----------------------|-----------|----------|
| ScoobertDoo@gmail.com | Scoobert | Doo |
| careers@tesla.com | Peter | Parker |
| HomerJSimp@gmail.com | Homer | Simpson |
| thestar@gmail.com | Patrick | Star |

LOG

```
1. SELECT DISTINCT Username,password
FROM log
ORDER BY USERNAME ASC;
```

(Used for Login, when a user attempts to match Username and Password.)

The screenshot shows the Oracle SQL Developer interface. In the Worksheet, a query is entered: `SELECT DISTINCT Username,password FROM log ORDER BY USERNAME ASC;`. The Script Output pane shows the results of this query, displaying columns USERNAME and PASSWORD with their respective values. The right pane shows the structure of the JOBSSEEKER table.

| COLUMN_NAME | DATA_TYPE | NULLABLE | DATA_DEFAULT | COLUMN_ID | COMMENTS |
|-------------|--------------------|----------|--------------|-----------|----------|
| 1 USERID | NUMBER | No | (null) | 1 | (null) |
| 2 EMAIL | VARCHAR2 (50 BYTE) | No | (null) | 2 | (null) |
| 3 FIRSTNAME | VARCHAR2 (30 BYTE) | No | (null) | 3 | (null) |
| 4 LASTNAME | VARCHAR2 (30 BYTE) | No | (null) | 4 | (null) |

| USERNAME | PASSWORD |
|---------------|---------------|
| A.Abhari | CPSS10 |
| ElonMuskkrat | Tesla#1 |
| HomerJSimp | uhhhhh123 |
| L.Lang | IluvKats77 |
| M.Myers | Halloween21 |
| PetePark | wallclimber99 |
| ScooberDoober | ScoobySnax88 |
| TheStar | password123 |

8 rows selected.

RESUME

```
1. SELECT DISTINCT job_location,count (USERID)
FROM resume
GROUP BY JOB_LOCATION;
```

(Used to determine the number of jobs in a particular location.)

The screenshot shows the Oracle SQL Developer interface. The 'Worksheet' tab contains the following SQL query:

```
SELECT DISTINCT job_location, count(USERID)
FROM resume
GROUP BY JOB_LOCATION;
```

The 'Script Output' tab shows the results of the query, which are displayed in two tables:

| JOB_LOCATION | COUNT (JOB_TITLES) |
|---------------|--------------------|
| Springfield | 1 |
| New York City | 1 |
| Toronto | 2 |

| JOB_LOCATION | COUNT (USERID) |
|---------------|----------------|
| Springfield | 1 |
| New York City | 1 |
| Toronto | 2 |

The 'Columns' tab on the right shows the schema of the 'RESUME' table:

| COLUMN_NAME | DATA_TYPE | NULLABLE | DATA_DEFAULT | COLUMN_ID | COMMENTS |
|----------------|--------------------|----------|--------------|-----------|----------|
| 1 USERID | NUMBER (4, 0) | No | (null) | 1 (null) | |
| 2 RESUMEID | NUMBER (4, 0) | No | (null) | 2 (null) | |
| 3 FIRSTNAME | VARCHAR2 (30 BYTE) | Yes | (null) | 3 (null) | |
| 4 LASTNAME | VARCHAR2 (30 BYTE) | Yes | (null) | 4 (null) | |
| 5 SCHOOL | VARCHAR2 (30 BYTE) | No | (null) | 5 (null) | |
| 6 JOB_TITLES | VARCHAR2 (30 BYTE) | No | (null) | 6 (null) | |
| 7 JOB_LOCATION | VARCHAR2 (100 B... | No | (null) | 7 (null) | |
| 8 SKILLS | VARCHAR2 (4000 ... | Yes | (null) | 8 (null) | |

RESUME_SAVED

```
1. SELECT companyid, count(distinct seekerid)
FROM resumes_saved
group by companyid;
```

(Used to determine the number of Job Seekers a Company is looking at)

The screenshot shows the Oracle SQL Developer interface. The 'Worksheet' tab contains the following SQL query:

```
SELECT companyid, count(distinct seekerid)
FROM resumes_saved
group by companyid;
```

The 'Script Output' tab shows the results of the query, which are displayed in two tables:

| COMPANYID | COUNT (DISTINCT SEEKERID) |
|-----------|---------------------------|
| 1006 | 1 |
| 1007 | 1 |
| 1004 | 2 |

| COMPANYID | COUNT (SEEKERID) |
|-----------|------------------|
| 1006 | 1 |
| 1007 | 1 |
| 1004 | 2 |

ROLE

```
1. SELECT DISTINCT Roleid, per_job, per_resume  
FROM Role  
ORDER BY ROLEID ASC;
```

(Used to display the possible roles available)

The screenshot displays the Oracle SQL Developer interface. The main window shows a query execution result for the query: `SELECT DISTINCT Roleid, per_job, per_resume FROM Role ORDER BY ROLEID ASC;`. The result is displayed in a table with columns: `COMPANYID`, `COUNT (DISTINCT SEEKERID)`, and a header row `RO PER_JOB PER_RESUME`. The data rows are:

| COMPANYID | COUNT (DISTINCT SEEKERID) |
|-----------|---------------------------|
| 1006 | 1 |
| 1007 | 1 |
| 1004 | 2 |

The right-hand pane shows the 'RESUMES_SAVED' table structure with columns: `COLUMN_NAME`, `DATA_TYPE`, `NULLABLE`, `DATA_DEFAULT`, `COLUMN_ID`, and `COMMENTS`. The data rows are:

| COLUMN_NAME | DATA_TYPE | NULLABLE | DATA_DEFAULT | COLUMN_ID | COMMENTS |
|-------------|--------------|----------|--------------|-----------|----------|
| 1 SEEKERID | NUMBER (4,0) | No | (null) | 1 | (null) |
| 2 RESUMEID | NUMBER (4,0) | No | (null) | 2 | (null) |
| 3 COMPANYID | NUMBER (4,0) | No | (null) | 3 | (null) |

The bottom right pane shows the 'Database Connection' section with 'Recent' and 'Databases Detected' tabs. The 'Recent' tab shows '11g Ryerson DB'. The 'Getting Started' section includes links for 'Get a Database', 'Information', 'Tutorials', 'Demos', and 'Training'.