East West University
Department of Computer Science and Engineering

CSE 301: LAB 05 [Assessed Lab]
Course Instructor: Dr. Mohammad Rezwanul Huq

Writing SQL Statements involving Outer Joins, Creating Views and Granting and Revoking Authorization

Lab Objective

Familiarize students with Outer joins, Views and Authorization in Oracle.

Lab Outcome

After completing this lab successfully, students will be able to:

1. **Construct** SQL Statements involving outer joins.
2. **Understand, create and use** views in a SQL Statement.
3. **Understand and execute** authorization statement.

Psychomotor Learning Levels

This lab involves activities that encompass the following learning levels in psychomotor domain.

Level	**Category**	**Meaning**	**Keywords**
P1	Imitation	Copy action of another; observe and replicate.	Relate, Repeat, Choose, Copy, Follow, Show, Identify, Isolate.
P2	Manipulation	Reproduce activity from instruction or memory	Copy, response, trace, Show, Start, Perform, Execute, Recreate.

Instructions

- ➢ Download and save banking script from https://goo.gl/q78ANw
- ➢ Execute SQLDeveloper tool and follow the instructor during the class.

Lab Activities (Introducing built-in functions in Oracle)

Activity 1: Outer Joins

➔ Find both account holder and non-account holder customers.

```
select * from customer natural left outer join depositor;
```

Activity 2: Views

➔ Create a view named CustomerAtStamford that contains customer name and street and those who live in Stamford.

```
create view CustomerAtStamford as
     select customer_name, customer_street
     from customer
     where customer_city = 'Stamford';
```

➔ Check the user-defined views by executing the following SQL statement.

```
Select * from user_views;
```

➔ Show the content of CustomerAtStamford view.

```
Select * from CustomerAtStamford;
```

➔ Create another view named CustomerAtStamford_Putnam that contains name of customers who live in 'Putnam' street based on CustomerAtStamford view.

```
Create view CustomerAtStamford_Putnam as
     Select customer_name
     From CustomerAtStamford
     Where customer_street = 'Putnam';
```

➔ Check the content of the CustomerAtStamford_Putnam view.

<u>Activity 3: Authorization</u>

➔ Create a user '`alice`'; password is `123456`;

➔ Grant the privileges as shown in the following SQL statements.

```
grant CREATE SESSION to alice;
grant UNLIMITED TABLESPACE to alice;
grant CREATE TABLE, CREATE VIEW, CREATE SEQUENCE to alice;
```

➔ Grant the privilege to alice so that she can access (only select) CustomerAtStamford view created by spring18.

```
Grant select, insert on CustomerAtStamford to alice;
```

➔ Now, connect as alice to the database and execute the following SQL statement.

```
Select * from spring18.CustomerAtStamford;
```

➔ Insert a tuple into the view spring18.CustomerAtStamford.

```
Insert into spring18.CustomerAtStamford values ('Peter',
'Bricklane');
```

➔ **Is this view updateable?**

➔ Check the granted authorizations to/from alice by executing the following SQL statement.

```
Select * from user_tab_privs;
```

➔ Revoke insert privilege from alice. (executed as spring18)

```
Revoke insert on CustomerAtStamford from alice;
```

➔ **Check the granted authorizations to/from alice again. Can you see any change?**

<u>Activity 4: Remote Connection using SQLDeveloper</u>

➔ Form a group of two students;
➔ Exchange the IP address of your machines.
➔ Follow the class discussion.

Exercise will be given during the class.