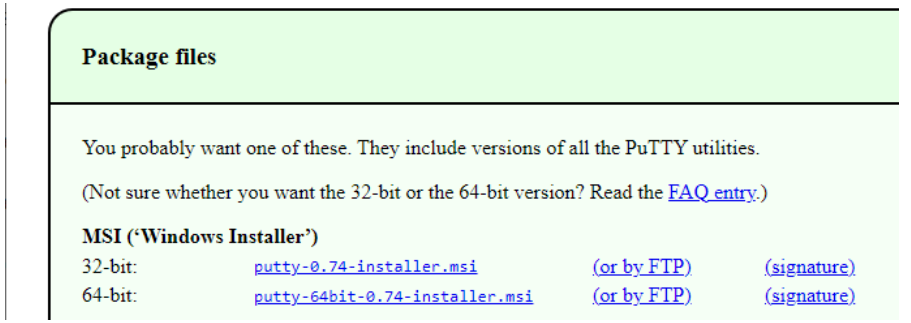
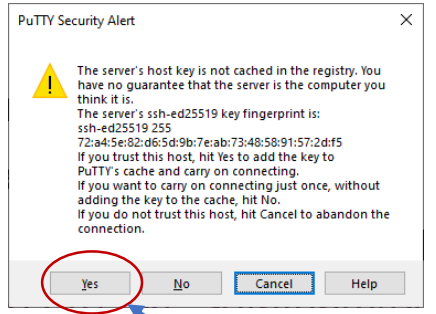
Tutorial 4: Basic SQL (Structured Query Language)

CS3402 Database Systems

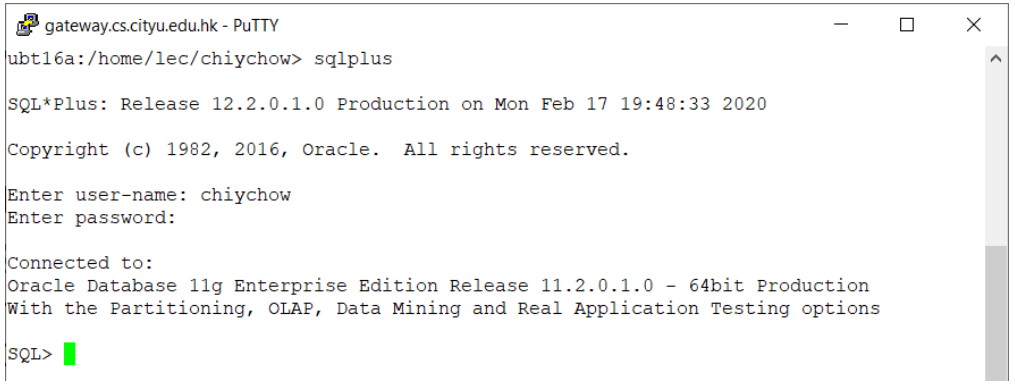
Connect to Oracle Database Server

- Step 1: Use the ssh client to connect to
  Host Name: gateway.cs.cityu.edu.hk
  Username: e-id
  Password: your CityU wifi password
  - For MacOS, you can use default Terminal app
  - For Windows, you can download and install PuTTY through https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html
- Step 2: Use the command “sqlplus” to connect to Oracle
  Username: e-id
  Password: student-id



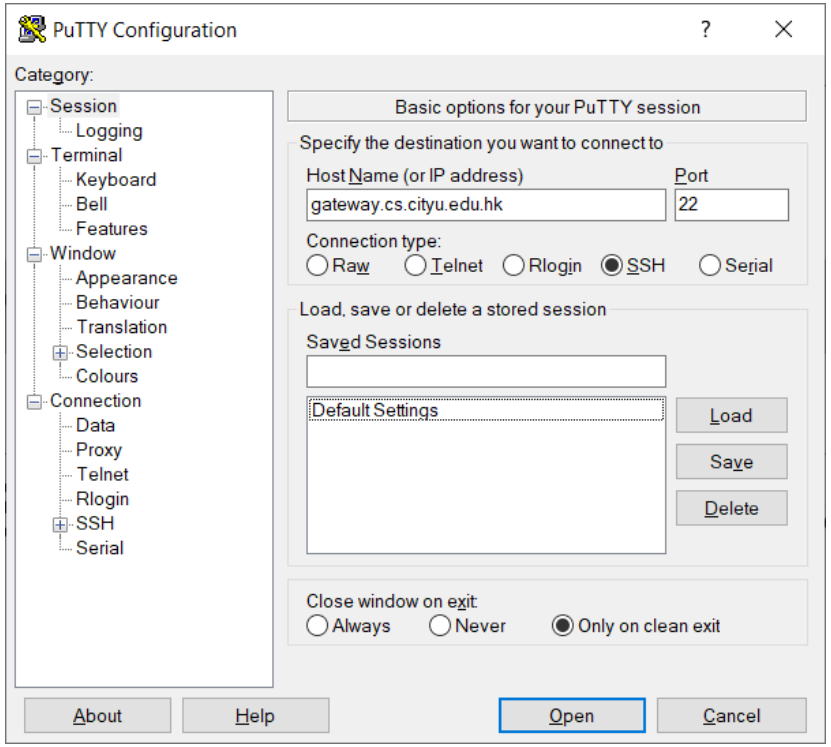


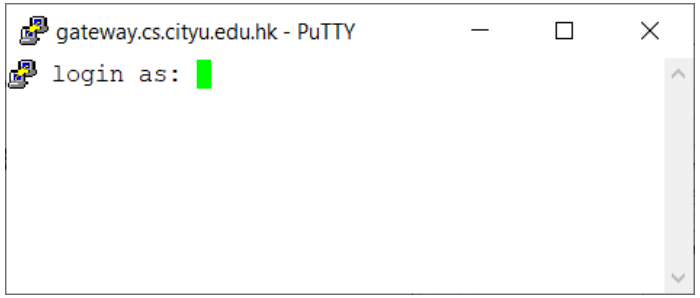
Select “Yes” when you see this popup.



PuTTY

- After you installed PuTTY
- Start PuTTY from the START menu
- Type gateway.cs.cityu.edu.hk in the Host Name
- Click Open button
- You can log into gateway.cs.cityu.edu.hk





Introduction

- In this lab, we will cover two of the most important components of a database: tables and queries. Each student needs to create the following tables and insert records into the corresponding tables.
- Information stored in these tables is needed for all the laboratory exercise in this course.

Dept

Deptno	Dname	Loc
10	Accounting	New York
20	Research	Dallas
30	Sales	Chicago
40	Operations	Boston

Salgrade

Grade	Lowsal	highsal
1	700	1200
2	1201	1400
3	1401	2000
4	2001	3000
5	3001	9999

Emp

Empno	Ename	Job	Mgr	Hiredate	Sal	Comm	Deptno
7369	Smith	Clerk	7902	17-Dec-80	800		20
7499	Allen	Salesman	7698	20-Feb-81	1600	300	30
7521	Ward	Salesman	7698	22-Feb-81	1250	500	30
7655	Jones	Manager	7839	2-Apr-81	2975		20
7654	Martin	Salesman	7698	28-Sep-81	1250	1400	30
7698	Blake	Manager	7839	1-May-91	2850		30
7782	Clark	Manager	7839	9-Jun-81	2450		10
7788	Scott	Analyst	7655	21-Mar-87	3000		20
7839	King	President		12-Nov-81	5000	0	10
7844	Turner	Salesman	7698	18-Sep-81	1500		30
7876	Adams	Clerk	7788	24-Apr-87	1100		20
7900	James	Clerk	7698	3-Dec-81	950		30
7902	Ford	Analyst	7655	3-Dec-81	3000		20
7934	Miller	Clerk	7782	3-Jan-81	1300		10

Create Tables

- Create table Dept:

```
CREATE TABLE Dept
(Deptno NUMBER (2) NOT NULL,
Dname VARCHAR (15),
Loc VARCHAR (15));
```

- Create table SALGRADE:

```
CREATE TABLE SALGRADE
(GRADE NUMBER,
LOSAL NUMBER,
HISAL NUMBER);
```

- Create table Emp:

```
CREATE TABLE EMP (
Empno NUMBER(4) NOT NULL,
Ename VARCHAR(15),
Job VARCHAR(15),
Mgr NUMBER(4),
Hiredate DATE,
Sal NUMBER(7, 2),
Comm NUMBER(7, 2),
Deptno NUMBER(2));
```

DESCRIBE and DROP Commands

- You can use the DESCRIBE command to obtain a description of a specific table, for example,

```
DESCRIBE Dept
```

- To destroy the table and its contents, use the DROP TABLE command, for example

```
DROP TABLE Dept;
```

INSERT STATEMENT

- After the table is created, you can use the INSERT statement to insert tuples into your table. For example,

```
INSERT INTO Dept VALUES
(10, 'Accounting', 'New York');
COMMIT;
```

- This command inserts the first tuple into Dept table.
- After each insert, update and delete operations that would modify the data in the database, you are suggested to do a "COMMIT" operation to ensure the changes take effect immediately.

SELECT STATEMENT (1/6)

- List all attributes of all departments in table Dept:

```
SELECT *
FROM Dept;
```

- Retrieve Dname of all departments in table Dept:

```
SELECT Dname
FROM Dept;
```

```
    DEPTNO DNAME          LOC
---------- -------------- -------------
        10 Accounting     New York
        20 Research       Dallas
        30 Sales          Chicago
        40 Operations     Boston

SQL>
```

```
DNAME
--------------
Accounting
Research
Sales
Operations

SQL>
```

SELECT STATEMENT (2/6)

- Retrieve all the jobs in the Emp table without eliminating duplicates

```
SELECT Job
FROM Emp;
```

- Retrieve all the DISTINCT jobs in the Emp table

```
SELECT DISTINCT Job
FROM Emp;
```

```
JOB
---------------
Clerk
Salesman
Salesman
Manager
Salesman
Manager
Manager
Analyst
President
Salesman
Clerk

JOB
---------------
Clerk
Analyst
Clerk

14 rows selected.

SQL>
```

```
JOB
---------------
Manager
Analyst
Clerk
President
Salesman

SQL>
```

SELECT STATEMENT (3/6)

- Retrieve employees in department number 30:

```
SELECT *
FROM Emp
WHERE Deptno=30;
```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7499	Allen	Salesman	7698	20-FEB-81	1600	300	30
7521	Ward	Salesman	7698	22-FEB-81	1250	500	30
7654	Martin	Salesman	7698	28-SEP-81	1250	1400	30
7698	Blake	Manager	7839	01-MAY-91	2850		30
7844	Turner	Salesman	7698	18-SEP-81	1500		30
7900	James	Clerk	7698	03-DEC-81	950		30

6 rows selected.

- Tips: Use the SET LINESIZE command to control the width of the data displayed, e.g.,

```
SET LINESIZE 256
```

SELECT STATEMENT (4/6)

- A WHERE clause search condition can use any of the following comparison operators:
  - = Equal to
  - <> Not equal to
  - > Greater than
  - > = Greater than or equal to
  - < Less than
  - <= Less than or equal to
- For example, find the employees whose commission is greater than his salary:

```
SELECT Ename, Sal, Comm
FROM Emp
WHERE Comm > Sal;
```

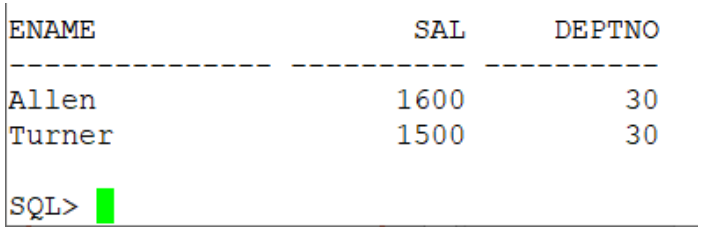
```
ENAME                 SAL       COMM
--------------- ---------- ----------
Martin                1250       1400

SQL>
```

SELECT STATEMENT (5/6)

- Find all the salesmen in department 30 who have a salary greater than or equal to $1,500.

```
SELECT Ename, Sal, Deptno
FROM Emp
WHERE Job='Salesman' AND Deptno=30
AND Sal>=1500;
```

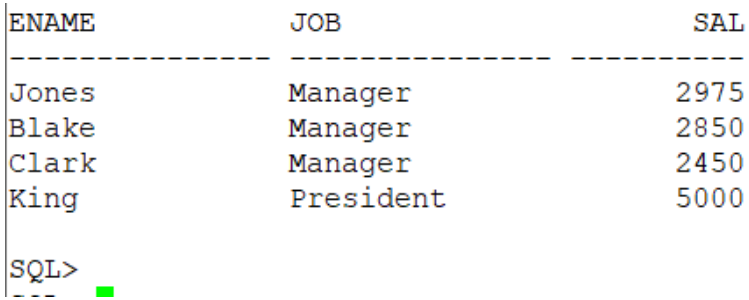


```
ENAME                SAL     DEPTNO
-------------- ---------- ----------
Allen                1600         30
Turner               1500         30

SQL>
```

- Find all the employees whose job is either manager or president.

```
SELECT Ename, Job, Sal
FROM Emp
WHERE Job='Manager' OR
Job='President';
```



```
ENAME          JOB                SAL
-------------- -------------- ----------
Jones          Manager              2975
Blake          Manager              2850
Clark          Manager              2450
King           President            5000

SQL>
```

SELECT STATEMENT (6/6)

- Find everyone who is neither a manger nor a clerk, but is in department 10.

```
SELECT *
FROM Emp
WHERE NOT (Job='Manager' OR Job='Clerk') AND
Deptno=10;
```

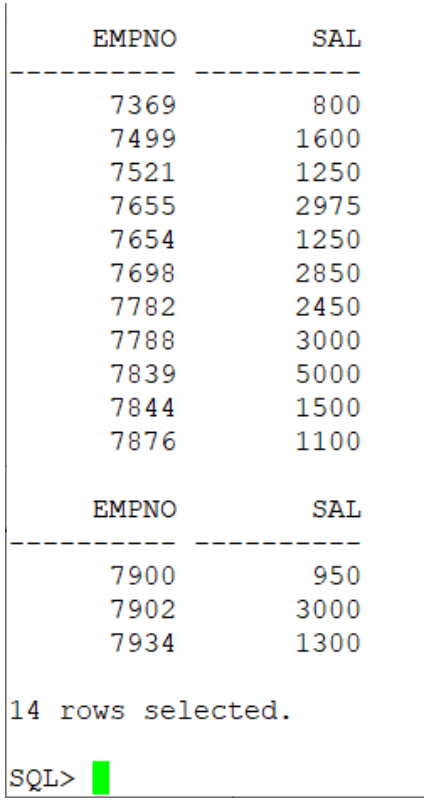
```
     EMPNO ENAME          JOB                MGR HIREDATE         SAL       COMM     DEPTNO
---------- -------------- -------------- ---------- --------- ---------- ---------- ----------
      7839 King           President                12-NOV-81       5000          0         10

SQL>
```

UPDATE STATEMENT

- Increase the salary of all employees by 10% in the Emp table.

```
UPDATE Emp
SET Sal = Sal *1.1;
Commit;
```



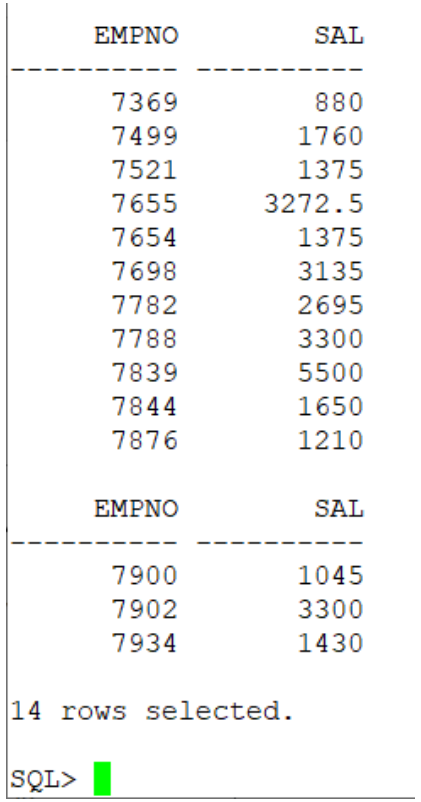
EMPNO	SAL
7369	800
7499	1600
7521	1250
7655	2975
7654	1250
7698	2850
7782	2450
7788	3000
7839	5000
7844	1500
7876	1100

EMPNO	SAL
7900	950
7902	3000
7934	1300

14 rows selected.

SQL>

Before the UPDATE statement



EMPNO	SAL
7369	880
7499	1760
7521	1375
7655	3272.5
7654	1375
7698	3135
7782	2695
7788	3300
7839	5500
7844	1650
7876	1210

EMPNO	SAL
7900	1045
7902	3300
7934	1430

14 rows selected.

SQL>

After the UPDATE statement

DELETE Statement

- Delete the tuples in table Emp where the employee's Job is Clerk.

```
DELETE FROM Emp
WHERE Job='Clerk';
Commit;
```

```
     EMPNO ENAME          JOB
---------- -------------- --------------
      7499 Allen          Salesman
      7521 Ward           Salesman
      7655 Jones          Manager
      7654 Martin         Salesman
      7698 Blake          Manager
      7782 Clark          Manager
      7788 Scott          Analyst
      7839 King           President
      7844 Turner         Salesman
      7902 Ford           Analyst

10 rows selected.

SQL>
```

After the DELETE statement