Examples - Simple SQLPlus & SQL

Bipin C. DESAI

Getting & Installing {Apache, Oracle, PHP} or, XAMPP

Consult:

http://www.oracle.com/technology/tech/php/htdocs/inst_php_apache_windows.html

For Oracle you need to register with OTN

http://www.apachefriends.org/en/xampp.html

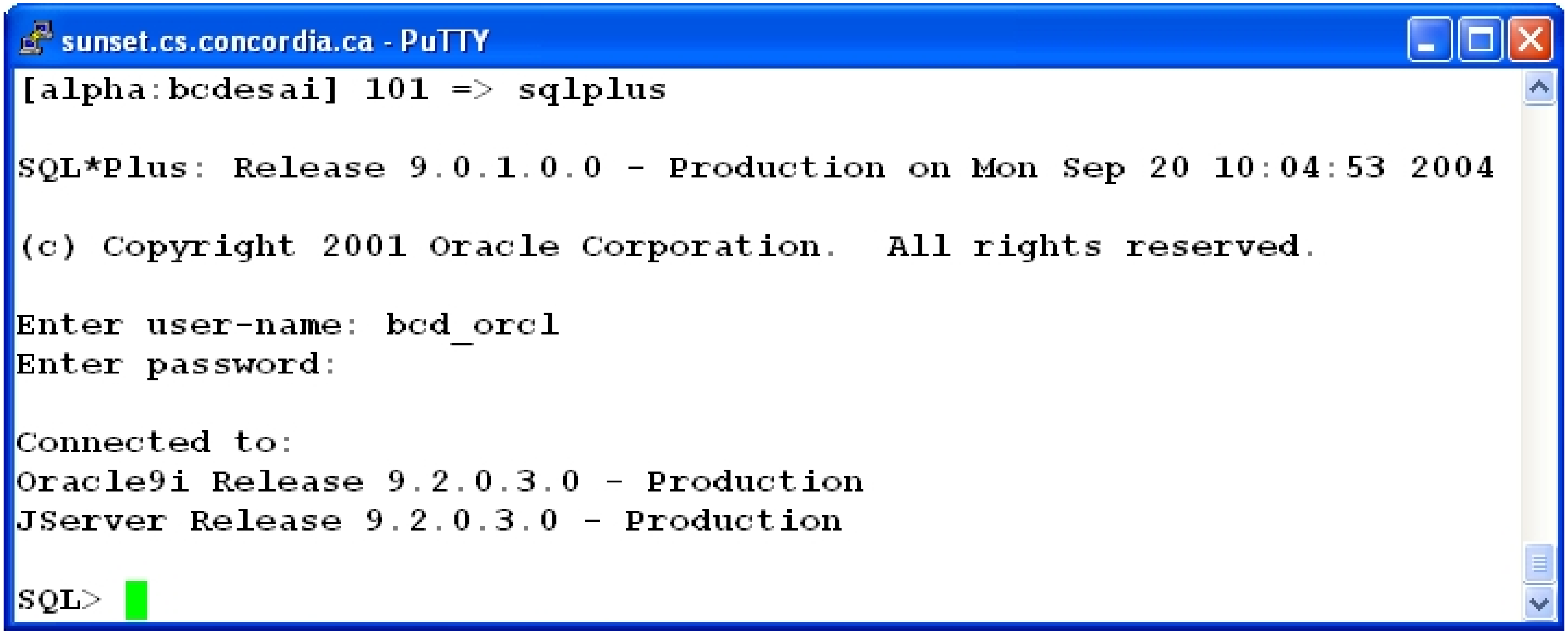
The projects are to be demonstrated on one of the systems in our labs.
So if you develop the projects on your own systems, make sure you could:

- Upload all the code to CrsMgr
- Have it run on one of our systems which has one of the above configurations
- It works as specified

These notes uses Oracle and/or MySQL

Connecting to SQLPlus

SQLPlus is a “user friendly interface” to ORACLE SQL to be used interactively.
You need Oracle USERID/PASSWORD and appropriate permission to a Oracle DB.
May connect remotely using a secure shell (e.g., Putty)



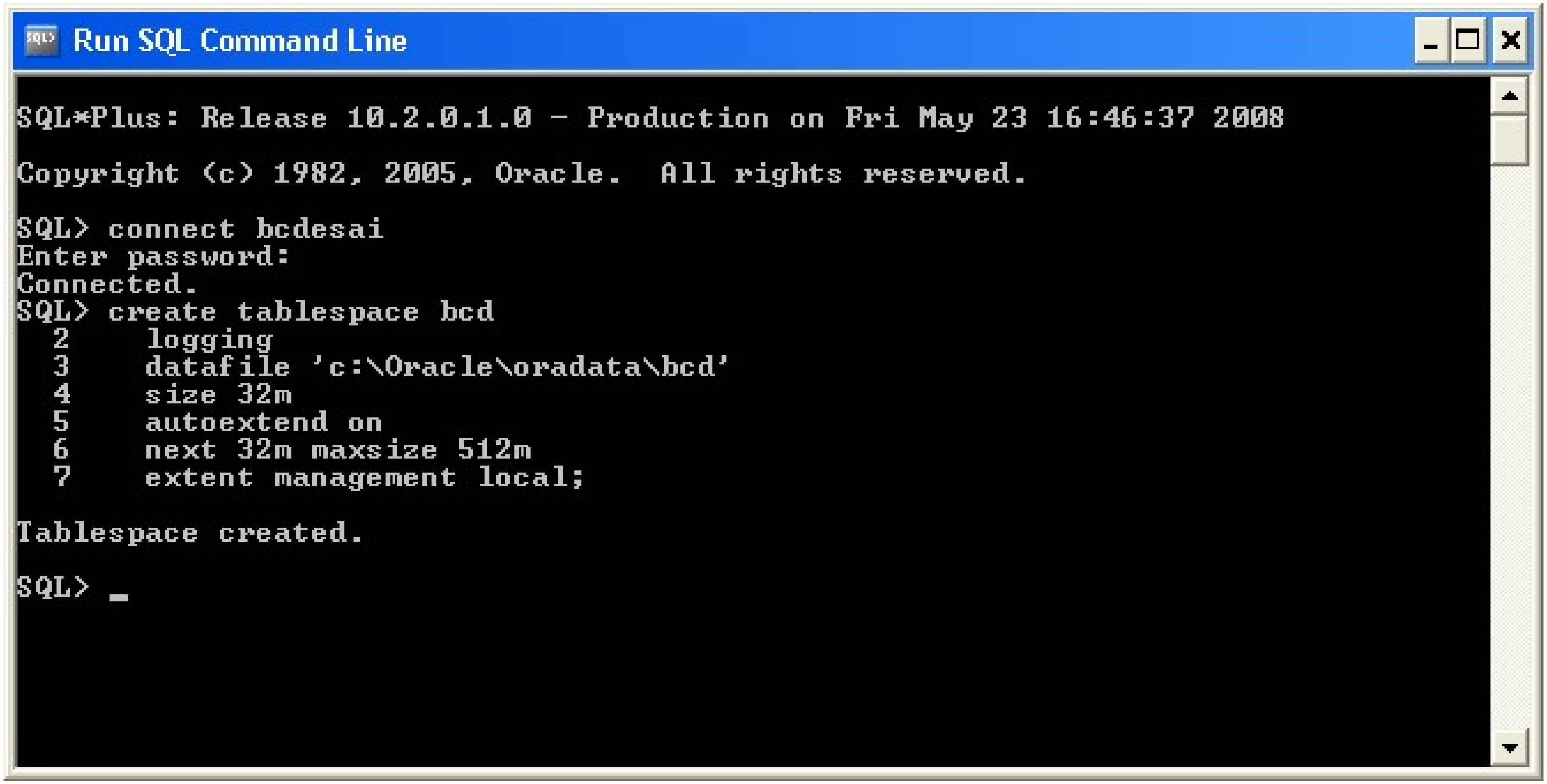
```
[alpha:bcdesai] 101 => sqlplus

SQL*Plus: Release 9.0.1.0.0 - Production on Mon Sep 20 10:04:53 2004

(c) Copyright 2001 Oracle Corporation.  All rights reserved.

Enter user-name: bcd_orcl
Enter password:

Connected to:
Oracle9i Release 9.2.0.3.0 - Production
JServer Release 9.2.0.3.0 - Production

SQL>
```

Download and install OracleWinXE

Start database (unless it has been installed as service which starts on boot)

From Start select RunSQL command line

Connect to oracle:

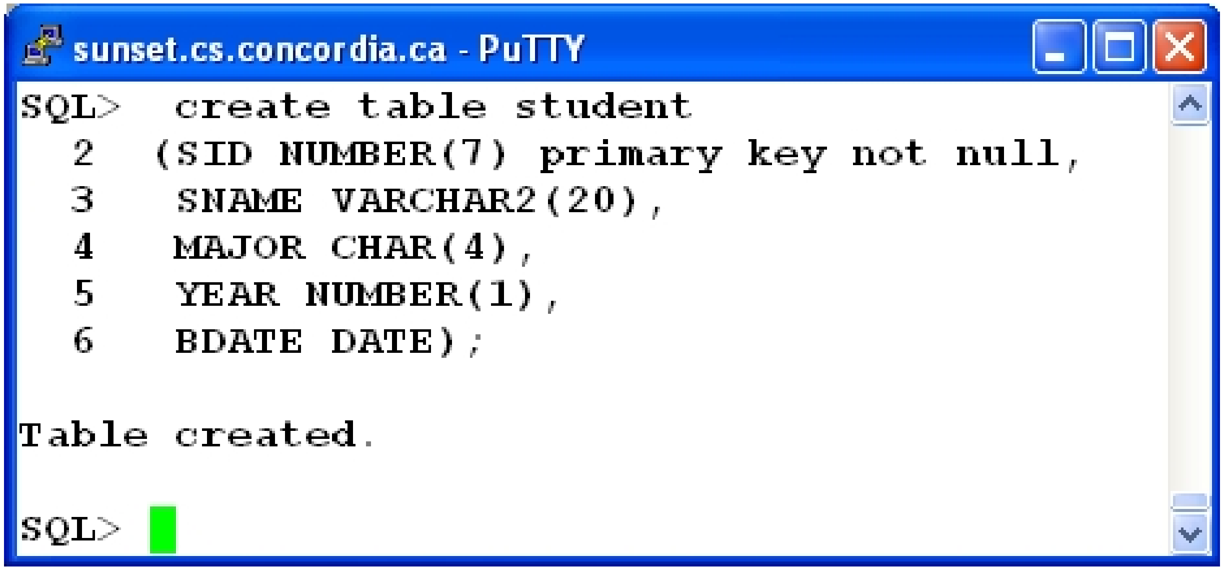


```
SQL*Plus: Release 10.2.0.1.0 - Production on Fri May 23 16:46:37 2008

Copyright (c) 1982, 2005, Oracle.  All rights reserved.

SQL> connect bcdesai
Enter password:
Connected.
SQL> create tablespace bcd
  2    logging
  3    datafile 'c:\Oracle\oradata\bcd'
  4    size 32m
  5    autoextend on
  6    next 32m maxsize 512m
  7    extent management local;

Tablespace created.

SQL> _
```

```
create table student
(SID NUMBER(7) primary key not null,
 SNAME VARCHAR2(20),
 MAJOR CHAR(4),
 YEAR NUMBER(1),
 BDATE DATE)
 tablespace bcd pctfree 2;
```

To execute a text file containing sql statements interactively from the sql prompt use @ followed by the full path to file

```
sql>@student.sql
```



```
SQL>  create table student
  2  (SID NUMBER(7) primary key not null,
  3   SNAME VARCHAR2(20),
  4   MAJOR CHAR(4),
  5   YEAR NUMBER(1),
  6   BDATE DATE);

Table created.

SQL>
```

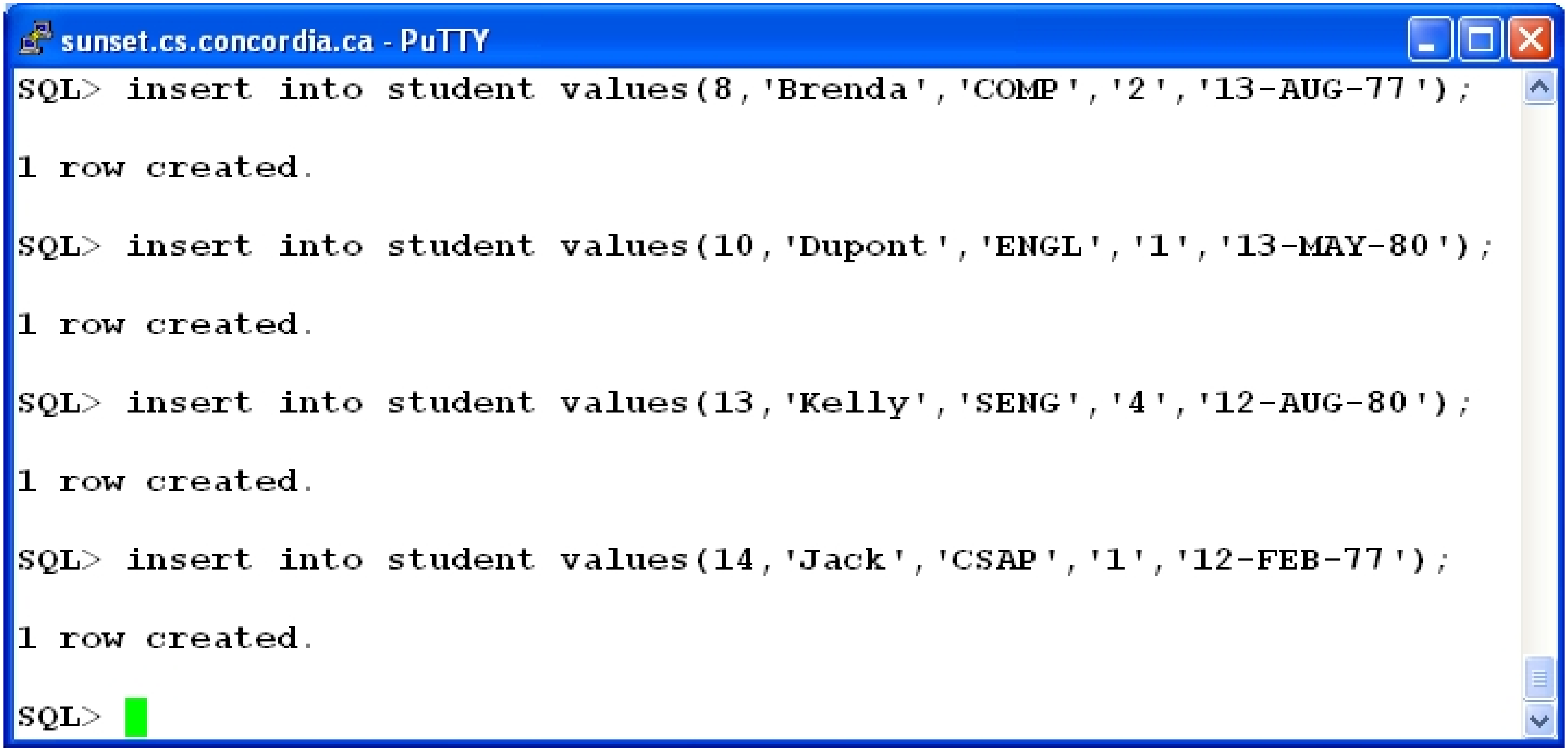
```
mysql> create table student
(SID DECIMAL(7) primary key not null,
SNAME VARCHAR (20),
MAJOR CHAR(4),
YEAR DEC(1),
BDATE DATE);
```

To execute a text file containing sql statements interactively from the sql prompt use @ followed by the full path to file

```
sql>@student.sql
```

```
mysql> desc student;
+-------+--------------+------+-----+---------+-------+
| Field | Type         | Null | Key | Default | Extra |
+-------+--------------+------+-----+---------+-------+
| SID   | decimal(7,0) | NO   | PRI | NULL    |       |
| SNAME | varchar(20)  | YES  |     | NULL    |       |
| MAJOR | char(4)      | YES  |     | NULL    |       |
| YEAR  | decimal(1,0) | YES  |     | NULL    |       |
| BDATE | date         | YES  |     | NULL    |       |
+-------+--------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

Inserting Data in a table – table must exist!



```
SQL> insert into student values(8,'Brenda','COMP','2','13-AUG-77');

1 row created.

SQL> insert into student values(10,'Dupont','ENGL','1','13-MAY-80');

1 row created.

SQL> insert into student values(13,'Kelly','SENG','4','12-AUG-80');

1 row created.

SQL> insert into student values(14,'Jack','CSAP','1','12-FEB-77');

1 row created.

SQL>
```

Find all students

SQL> select * from student;

```
       SID SNAME                MAJO       YEAR BDATE
---------- -------------------- ---- ---------- ---------
         8 Brenda               COMP          2 13-AUG-77
        10 Dupont               ENGL          1 13-MAY-80
        13 Kelly                SENG          4 12-AUG-80
        14 Jack                 CSAP          1 12-FEB-77
```

```
SQL>column major format a5
```

format-stud.sql

```
SQL>column sid format 9,9
SQL>column sname format a12
SQL>column major format a5
SQL>column year format 999
SQL>column bdate format a12
```

```
 SID SNAME        MAJOR YEAR BDATE
---- ---------- ----- ---- ---------
   8 Brenda        COMP     2 13-AUG-77
 1,0 Dupont        ENGL     1 13-MAY-80
 1,3 Kelly         SENG     4 12-AUG-80
 1,4 Jack          CSAP     1 12-FEB-77
```

```
select s.sname
from student s
where to_date(s.bdate) like '%13%';
```

```
SNAME
------------
Brenda
Dupont
```

```
select s.sname
from student s
where s.bdate like '%13%';
```

SQL script: date.sql

Find students born in August

```
select s.sname
from student s
where to_date(s.bdate) like '%AUG%';
```

```
SNAME
------------
Brenda
Kelly
```

```
select s.sname
from student s
where s.bdate like '%AUG%';
```

SQL script: month.sql

Find student born in 1977

```
select s.sname
from student s
where to_date(s.bdate) like '%77%';
```

```
SNAME
------------
Brenda
Jack
```

```
select s.sname
from student s
where s.bdate like '%77%';
```

SQL script: year.sql

```
create table dept
(DEPT CHAR(20) not null,
 CODE CHAR(4) primary key not null);

insert into dept values('Computer Science', 'COMP');
insert into dept values('Decision Science', 'DISC');

create table deptmajor
(CODE CHAR(4),
 MAJOR CHAR(20),
primary key (CODE, MAJOR))

insert into deptmajor values('COMP', 'COTH');
insert into deptmajor values('COMP', 'SENG');
insert into deptmajor values('COMP', 'CSAP');
insert into deptmajor values('DISC', 'OPRS');
```

```
create table course
(CNAME CHAR(20),
 CNUMBER CHAR(8) primary key NOT NULL,
 CREDITS NUMBER(2),
 ODEPT CHAR(4),
 foreign key (ODEPT) references dept(code)
 on delete cascade)

insert into course values('C++','COMP248',3,'COMP');
insert into course values('DATA STRUCTURES ','COMP352',3,
 'COMP');
insert into course values('OPERATING SYSTEMS','COMP346',4
,'COMP');
insert into course values('DATABASE','COMP353',4,'COMP');
insert into course values('Operation Research','DISC253',4,'DISC');
```

```
create table crs_section
(SECID NUMBER(6) primary key NOT NULL,
 COURSE_NUM CHAR(8),
 SECTION CHAR(2),
 SEMESTER CHAR(4),
 YEAR CHAR(4),
 SCHEDULE CHAR(10),
 ROOM CHAR(7));

insert into crs_section values
(85,'COMP352','A','FALL', '1998','TH16001715','H123');
insert into crs_section values
(90,'COMP353','B','FALL','1999','MW08451000','H631');
insert into crs_section values
(95,'DISC253','B','FALL','1999','MW10151130','H631');
```

```
create table prereq
(COURSE_Number CHAR(8),
 PREREQ CHAR(8),   primary key (course_number, prereq));
insert into prereq values('COMP353','COMP352');

insert into prereq values('COMP353','COMP346');
insert into prereq values('COMP352','COMP248');

create table enrollment
(STUDENT_NUMBER NUMBER(3) not null,
 SECTION_ID NUMBER(6) not null,  GRADE CHAR(1),
 primary key(student_number, section_id));
insert into enrollment values(8,85,null);
insert into enrollment values(10,90,null);
insert into enrollment values(8,90,null);
insert into enrollment values(14,90,null);
insert into enrollment values(14,95,null);
```

Find details of studs. taking a course offered by the “DISC” dept.

```
select s.SID, s.SNAME, s.MAJOR, s.YEAR, s.BDATE
from student s, dept d, course c, crs_section r, enrolment e
where  c.ODEPT=d.CODE and
         r.COURSE_NUM=c.CNUMBER and
         r.SECID=e.SECTION_ID and
         e.STUDENT_NUMBER = s.SID and
         d.CODE= 'DISC';
```

```
SID SNAME           MAJOR YEAR BDATE
---- ------------ ----- ---- ------------
 1,4 Jack            CSAP     1 12-FEB-77
```

SQL script: ex-select3.sql

Find student who are registered in a course offered by their majoring dept.

```
select * from student
where student.sid in
(select s.sid from student s, dept d, course c, crs_section r, enrollment e
where
c.ODEPT=d.CODE and           -- c Offering Dept same as the d dept
s.MAJOR=c.ODEPT and          -- s major Dept same as the c.ODEPT
r.COURSE_NUM=c.CNUMBER and   -- the section is for the course c
r.SECID=e.SECTION_ID and     -- r course section same as e section
e.STUDENT_NUMBER = s.SID);
```

```
 SID SNAME        MAJOR     YEAR  BDATE
---- ------------ ---------- ------- -----------
   8 Brenda       COMP         2    13-AUG-80
```

Find students who are currently registered.

sql > @ex-select1.sql

```
select * from student
where student.sid in
      (select s.sid
      from student s, dept d, course c, crs_section r, enrolment e
      where  c.ODEPT=d.CODE and
      r.COURSE_NUM=c.CNUMBER and
      r.SECID=e.SECTION_ID and
      e.STUDENT_NUMBER = s.SID);
```

```
SID SNAME        MAJOR YEAR BDATE
---- ------------ ----- ---- ------------
   8 Brenda       COMP     2 13-AUG-80
 1,0 Dupont       ENGL     1 13-MAY-80
 1,4 Jack         CSAP     1 12-FEB-77
```

```
select s.SID, s.SNAME, s.MAJOR, s.YEAR, s.BDATE
from student s, dept d, course c, crs_section  r, enrolment e
where  c.ODEPT=d.CODE and
       r.COURSE_NUM=c.CNUMBER and
       r.SECID=e.SECTION_ID and
       e.STUDENT_NUMBER = s.SID and
       d.CODE= 'COMP';
```

```
SQL> @ex-select2.sql
 SID SNAME        MAJOR YEAR BDATE
---- ------------ ----- ---- ------------
   8 Brenda       COMP     2 13-AUG-80
 1,0 Dupont       ENGL     1 13-MAY-80
   8 Brenda       COMP     2 13-AUG-80
 1,4 Jack         CSAP     1 12-FEB-77
```

The DUAL table in Oracle

SQL> describe dual;

```
 Name            Null?    Type
 --------------- -------- --------------
 DUMMY                     VARCHAR2(1)
```

Contains one row and one column. Can be used to put results

SQL> select power(2,10) from dual;

```
POWER(2,10)
-----------
       1024
```

select sysdate from dual;

SQL> select to_date(sysdate) from dual;

```
TO_DATE(S
---------
29-SEP-02
```

```
SQL> select add_months(sysdate,2) from dual;
ADD_MONTH
---------
29-NOV-02
```

Lets make Brenda younger

```
SQL> update student
set bdate=(select add_months(bdate,36)from dual)
where sid=8
```

```
update student
set bdate= add_months(bdate,36)
where sid=8
```

```
SQL> select * from student where sid=8;

 SID SNAME        MAJOR YEAR BDATE
---- ------------ ----- ---- ------------
   8 Brenda       COMP     2 13-AUG-80
                                13-AUG-77
```

Editing SQL Buffer

Command	abbrev.	Operation on crnt. line/all lines
append txt	a text	adds text at the end of a line
change /old/new/	c /old/new/	change old to new in a line
change /txt	c /txt	delete text from a line
clear buffer	cl buff	delete all lines in the buffer
delete	del	delete the current line
delete n	**del** n	delete line n
delete last	**del** last	delete the last line of the buffer
delete n,m	**del** n,m	delete lines n - m from buffer
ed	**ed**	edit the buffer or a file
get file		load file into buffer
input	i	add one or more lines
input txt	i txt	add text as a line
host		exit temp to OS, exit back to SQLPlus
list	l	list all lines of buffer
list n	l n (n)	list line n and make it current
list *	l *	list current. line

Editing SQL Buffer

Command	abbrev.	Operation on crnt. line/all lines
list last	l last	list last line
list m n	l m n	list lines m – n
save file	sav file	save buffer to file
run	**/**	execute the commands in buffer

Other useful commands:

alter user *userid* **identified by** *newpasswor***d**

spool nameoffile

Comments

/* for multilane comments */

rem for a single line comment

-- comments that can start anywhere in a line up to the eol

```
create table student  -- we will create a table for students
(SID NUMBER(7) primary key not null, --not null is redundant
 SNAME VARCHAR2(20),  --varchar2 is a variable length string
 /*
  We will now define
  the student’s major and year
 */
MAJOR CHAR(4),
YEAR NUMBER(1),
rem BDATE is his/her birth date
rem It can be used to compute the age which is not stored.
BDATE DATE)
```

The editor used for the ed command is the default editor set using

```
setenv EDITOR {emas| vi | gedit | xemacs | ndedit}  for tcsh/csh
export EDITOR={ emas| vi | gedit | xemacs | ndedit} for bash
```

Alternatively, you can set up your editor using the define command:

```
SQL> define _editor=emacs
SQL> define _USER=scott

SQL> define _PW=tiger

SQL> define

DEFINE _CONNECT_IDENTIFIER = "cind" (CHAR)

DEFINE _SQLPLUS_RELEASE = "902000100" (CHAR)

DEFINE _EDITOR        = "emacs" (CHAR)

DEFINE _O_VERSION      = "Oracle9i Enterprise Edition Release
9.2.0.1.0 - Production

With the Partitioning, OLAP and Oracle Data Mining options

JServer Release 9.2.0.1.0 - Production" (CHAR)

DEFINE _O_RELEASE      = "902000100" (CHAR)

DEFINE _RC            = "0" (CHAR)

DEFINE _USER          = "scott" (CHAR)

DEFINE _PW            = "tiger" (CHAR)
```

Show user defined varaibles