1)

```
SET SERVEROUTPUT ON
DECLARE
        v_a VARCHAR2(30); /*Introduction to Oracle PL/SQL -> has length 29 */
        v_b NUMBER(8,2);  /* total number of digits = 8 and rounding occurs at the second */
        v_c CONSTANT VARCHAR2(4) := '603D'; /*->auto_init to 603D*/
        v_d BOOLEAN;
         v_e DATE := SYSDATE + 7; /*->auto_init = one week from today*/
BEGIN
DBMS_OUTPUT.PUT_LINE('The constant is: '||v_c);
DBMS_OUTPUT.PUT_LINE('The constant date is: '||v_e);
/* v_b is an EMPTY numeric variable */
END;
```

2)

```
SET SERVEROUTPUT ON
DECLARE
        v_a VARCHAR2(30) := 'Data Bases 2'; /*Introduction to Oracle PL/SQL -> has length 29 */
        v_b NUMBER(8,2);  /* total number of digits = 8 and rounding occurs at the second */
        v_c CONSTANT VARCHAR2(4) := '603D'; /*->auto_init to 603D*/
        v_d BOOLEAN;
        v_e DATE := SYSDATE + 7; /*->auto_init = one week from today*/
BEGIN
IF v_a LIKE 'Introduction to Underwater Basketweaving' THEN
DBMS_OUTPUT.PUT_LINE('Was a match: '||v_a);
ELSE IF v_c LIKE '603D' THEN
DBMS_OUTPUT.PUT_LINE('The course name is: '||v_a||' and the room number is: '||v_c);
ELSE
DBMS_OUTPUT.PUT_LINE('Unidentified course and room number!');
END IF;
END IF;

EXCEPTION
   WHEN OTHERS THEN
      DBMS_OUTPUT.PUT_LINE('An error occured!');
END;
```

3)

```
CREATE TABLE Ex3(
ID NUMBER(10,0),
NAME VARCHAR2(20)
);

CREATE SEQUENCE Ex3_SEQ INCREMENT BY 5;
```

4)

Error report -
ORA-01422: exact fetch returns more than requested number of rows
ORA-06512: at line 5
01422. 00000 - "exact fetch returns more than requested number of rows"
*Cause: The number specified in exact fetch is less than the rows returned.
*Action: Rewrite the query or change number of rows requested

I added : EXCEPTION WHEN OTHERS THEN, because of the error above, since the result for “most classes” can generate more than 1 student entry!

a) & b)

```
SET SERVEROUTPUT ON
DECLARE
   v_name STUDENT.last_name%TYPE;
   v_id STUDENT.student_id%TYPE;

BEGIN
SELECT st.last_name INTO v_name FROM STUDENT st, ENROLLMENT et
WHERE st.student_id = et.student_id
HAVING COUNT(*) = (SELECT MAX(COUNT(*)) FROM STUDENT stt, ENROLLMENT ett WHERE
stt.student_id = ett.student_id GROUP BY ett.student_id)
GROUP BY st.last_name;
EXCEPTION
   WHEN OTHERS THEN
      v_name := 'More than one entry!';

INSERT INTO Ex3(id, name) VALUES (Ex3_SEQ.NEXTVAL, v_name);
DBMS_OUTPUT.PUT_LINE('Student name and id: '||v_name);
END;
SAVEPOINT A;
```

c)

```
SET SERVEROUTPUT ON
DECLARE
   v_name STUDENT.last_name%TYPE;
   v_id STUDENT.student_id%TYPE;
BEGIN
SELECT st.last_name INTO v_name FROM STUDENT st, ENROLLMENT et
WHERE st.student_id = et.student_id
HAVING COUNT(*) = (SELECT MIN(COUNT(*)) FROM STUDENT stt, ENROLLMENT ett WHERE
stt.student_id = ett.student_id GROUP BY ett.student_id)
GROUP BY st.last_name;
```

```
EXCEPTION
   WHEN OTHERS THEN
      v_name := 'More than one entry!';

INSERT INTO Ex3(id, name) VALUES (Ex3_SEQ.NEXTVAL, v_name);
DBMS_OUTPUT.PUT_LINE('Student name and id: '||v_name);
END;
SAVEPOINT B;
```

d)

```
SET SERVEROUTPUT ON
DECLARE
   v_name STUDENT.last_name%TYPE;
   v_id STUDENT.student_id%TYPE;
BEGIN
SELECT it.last_name INTO v_name FROM INSTRUCTOR it, SECTION st
WHERE it.instructor_id = st.section_id
HAVING COUNT(*) = (SELECT MAX(COUNT(*)) FROM INSTRUCTOR itt, SECTION stt WHERE stt.section_id
= itt.instructor_id GROUP BY itt.instructor_id)
GROUP BY it.last_name;
EXCEPTION
   WHEN OTHERS THEN
      v_name := 'More than one entry!';

INSERT INTO Ex3(id, name) VALUES (Ex3_SEQ.NEXTVAL, v_name);
DBMS_OUTPUT.PUT_LINE('Student name and id: '||v_name);
END;
SAVEPOINT C;
```

e)

```
BEGIN
SELECT instructor_id INTO v_id FROM INSTRUCTOR WHERE last_name = v_name;
EXCEPTION
   WHEN OTHER THEN
      v_id:=0;
END;
```

f)

```
ROLLBACK [TO_B] to SAVEPOINT B;
```

g)

```
SET SERVEROUTPUT ON
DECLARE
   v_name STUDENT.last_name%TYPE;
```

```
   v_id STUDENT.student_id%TYPE;
BEGIN
SELECT it.last_name INTO v_name FROM INSTRUCTOR it, SECTION st
WHERE it.instructor_id= st.instructor_id
HAVING COUNT(*) = (SELECT MIN(COUNT(*)) FROM INSTRUCTOR itt, SECTION stt WHERE
stt.instructor_id = itt.instructor_id GROUP BY itt.instructor_id)
GROUP BY it.last_name;
EXCEPTION
   WHEN OTHERS THEN
      v_name := 'More than one entry!';

INSERT INTO Ex3(id, name) VALUES (v_id, v_name);
DBMS_OUTPUT.PUT_LINE('Instructor name and id: '||v_name);
END;
```

h)

```
SET SERVEROUTPUT ON
DECLARE
   v_name STUDENT.last_name%TYPE;
   v_id STUDENT.student_id%TYPE;
BEGIN
SELECT it.last_name INTO v_name FROM INSTRUCTOR it, SECTION st
WHERE it.instructor_id= st.instructor_id
HAVING COUNT(*) = (SELECT MIN(COUNT(*)) FROM INSTRUCTOR itt, SECTION stt WHERE
stt.instructor_id = itt.instructor_id GROUP BY itt.instructor_id)
GROUP BY it.last_name;
EXCEPTION
   WHEN OTHERS THEN
      v_name := 'More than one entry!';

INSERT INTO Ex3(id, name) VALUES (Ex3_SEQ.NEXTVAL, v_name);
DBMS_OUTPUT.PUT_LINE('Instructor name and id: '||v_name);
END;
```