LAB 3 ARSHIA A KALHORANI,

CODE Q1:

SET SERVEROUTPUT ON

SET VERIFY OFF

DECLARE

TYPE course\_table\_type IS TABLE OF

COURSE.DESCRIPTION%TYPE INDEX BY PLS\_INTEGER;

course\_table course\_table\_type;

v\_count NUMBER(5) := 1;

CURSOR c\_course\_cursor IS Select DESCRIPTION from course where PREREQUISITE IS NULL ORDER BY DESCRIPTION;

BEGIN

OPEN c\_course\_cursor;

LOOP

FETCH c\_course\_cursor INTO course\_table(v\_count);

EXIT WHEN c\_course\_cursor%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE('Course Description : ' || v\_count || ':' || course\_table(v\_count));

v\_count := v\_count + 1;

END LOOP;

v\_count := v\_count - 1;

DBMS\_OUTPUT.PUT\_LINE('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

DBMS\_OUTPUT.PUT\_LINE('Total # of Courses' || ' without the Prerequisite is: ' || v\_count);

CLOSE c\_course\_cursor;

END;

/

OUTPUTQ1:

Course Description : 1:DP Overview

Course Description : 2:Intro to Computers

Course Description : 3:Java for C/C++ Programmers

Course Description : 4:Operating Systems

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Total # of Courses without the Prerequisite is: 4

PL/SQL procedure successfully completed.

CODEQ2:

SET LINESIZE 130;

SET PAGESIZE 130;

SET SERVEROUTPUT ON

SET VERIFY OFF

DECLARE

TYPE course\_table\_type IS TABLE OF

COURSE%ROWTYPE;

course\_table course\_table\_type := course\_table\_type();

v\_count NUMBER(5) := 1;

CURSOR c\_course\_cursor IS Select DESCRIPTION from course where PREREQUISITE IS NULL ORDER BY DESCRIPTION;

BEGIN

FOR i IN c\_course\_cursor

LOOP

course\_table.EXTEND;

DBMS\_OUTPUT.PUT\_LINE('Course Description : ' || v\_count || ': ' || i.description);

v\_count := v\_count + 1;

END LOOP;

v\_count := v\_count - 1;

DBMS\_OUTPUT.PUT\_LINE('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

DBMS\_OUTPUT.PUT\_LINE('Total # of Courses' || ' without the Prerequisite is: ' || v\_count);

END;

/

OUTPUTQ2:

Course Description : 1: DP Overview

Course Description : 2: Intro to Computers

Course Description : 3: Java for C/C++ Programmers

Course Description : 4: Operating Systems

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Total # of Courses without the Prerequisite is: 4

PL/SQL procedure successfully completed.

CODEQ3:

SET LINESIZE 130;

SET PAGESIZE 130;

SET SERVEROUTPUT ON

SET VERIFY OFF

ACCEPT zip PROMPT 'Enter first 3 digit of zip code: ';

DECLARE

type t\_rec is RECORD

(

v\_zip NUMBER,

v\_num\_stud NUMBER

);

v\_total NUMBER := 0;

v\_total\_students NUMBER;

t\_course t\_rec;

CURSOR c\_zip IS Select ZIP, COUNT(STUDENT\_ID)

FROM (SELECT DISTINCT s.zip, s.student\_id from Student s

LEFT JOIN enrollment e

ON s.student\_id = e.student\_id

WHERE s.zip LIKE '&zip' || '%')

GROUP BY zip

ORDER BY zip;

BEGIN

SELECT COUNT(\*) INTO v\_total\_students FROM student where zip LIKE '&zip'|| '%';

IF(v\_total\_students > 0) THEN

OPEN c\_zip;

LOOP

FETCH c\_zip INTO t\_course;

EXIT WHEN c\_zip%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE('Zip code: ' || t\_course.v\_zip || ' has exactly ' || t\_course.v\_num\_stud || ' students enrolled.');

v\_total := v\_total + 1;

END LOOP;

DBMS\_OUTPUT.PUT\_LINE('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

DBMS\_OUTPUT.PUT\_LINE('Total # of zip codes under ' || '&zip' || ' is ' || v\_total);

DBMS\_OUTPUT.PUT\_LINE('Total # of Students under zip code ' || '&zip' || ' is ' || v\_total\_students);

CLOSE c\_zip;

ELSE

DBMS\_OUTPUT.PUT\_LINE('No students found under this zip code');

END IF;

END;

OUTPUTQ3:

Enter first 3 digit of zip code: 073

Zip code: 7302 has exactly 1 students enrolled.

Zip code: 7304 has exactly 2 students enrolled.

Zip code: 7306 has exactly 4 students enrolled.

Zip code: 7307 has exactly 3 students enrolled.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Total # of zip codes under 073 is 4

Total # of Students under zip code 073 is 10

PL/SQL procedure successfully completed.

Enter first 3 digit of zip code: 013

No students found under this zip code

PL/SQL procedure successfully completed.

CODEQ4:

SET LINESIZE 130;

SET PAGESIZE 130;

SET SERVEROUTPUT ON

SET VERIFY OFF

ACCEPT zip PROMPT 'Enter first 3 digit of zip code: ';

DECLARE

CURSOR c\_zip IS Select ZIP, COUNT(STUDENT\_ID) AS stud\_total

FROM (SELECT DISTINCT s.zip, s.student\_id from Student s

LEFT JOIN enrollment e

ON s.student\_id = e.student\_id

WHERE s.zip LIKE '&zip' || '%')

GROUP BY zip

ORDER BY zip;

c\_stud c\_zip%ROWTYPE;

TYPE zip\_table\_type IS TABLE OF

c\_stud%TYPE INDEX BY PLS\_INTEGER;

zip\_table zip\_table\_type;

v\_total NUMBER := 0;

v\_total\_students NUMBER;

BEGIN

SELECT COUNT(\*) INTO v\_total\_students FROM student where zip LIKE '&zip'|| '%';

IF(v\_total\_students > 0) THEN

FOR i in c\_zip

LOOP

DBMS\_OUTPUT.PUT\_LINE('Zip code: ' || i.zip || ' has exactly ' || i.stud\_total || ' students enrolled.');

v\_total := v\_total + 1;

END LOOP;

DBMS\_OUTPUT.PUT\_LINE('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*');

DBMS\_OUTPUT.PUT\_LINE('Total # of zip codes under ' || '&zip' || ' is ' || v\_total);

DBMS\_OUTPUT.PUT\_LINE('Total # of Students under zip code ' || '&zip' || ' is ' || v\_total\_students);

ELSE

DBMS\_OUTPUT.PUT\_LINE('No students found under this zip code');

END IF;

END;

/

OUTPUTQ4:

Enter first 3 digit of zip code: 073

Zip code: 07302 has exactly 1 students enrolled.

Zip code: 07304 has exactly 2 students enrolled.

Zip code: 07306 has exactly 4 students enrolled.

Zip code: 07307 has exactly 3 students enrolled.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Total # of zip codes under 073 is 4

Total # of Students under zip code 073 is 10

PL/SQL procedure successfully completed.