CSC 352/452: DATABASE PROGRAMMING
ASSIGNMENT # 3 (100 Points) Due Date 8/6 11:59pm
Database Interactions

1. **(50 Points)** Create part of mail-order database described on page 9 of your book:

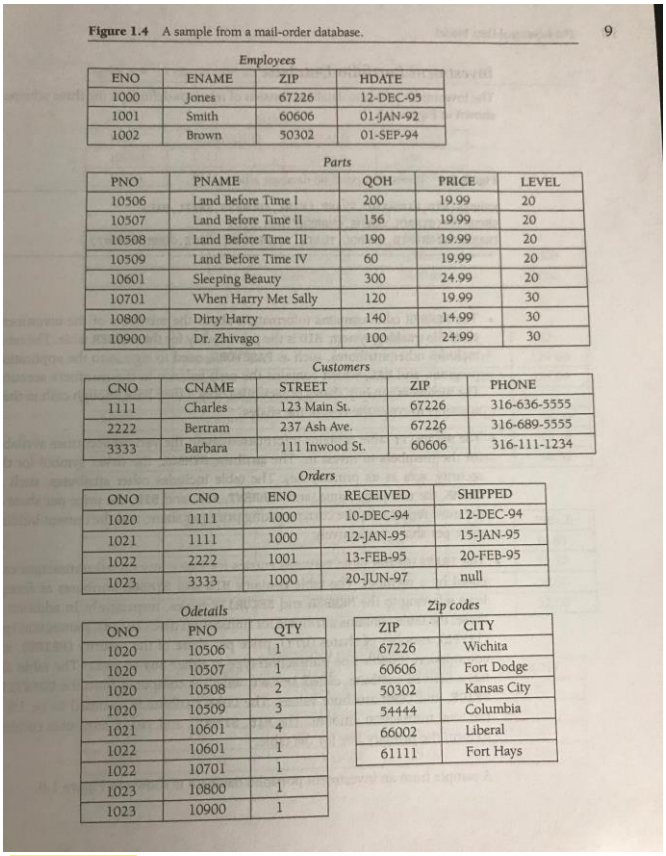

Figure 1.4 A sample from a mail-order database. 9

Employees

ENO	ENAME	ZIP	HDATE
1000	Jones	67226	12-DEC-95
1001	Smith	60606	01-JAN-92
1002	Brown	50302	01-SEP-94

Parts

PNO	PNAME	QOH	PRICE	LEVEL
10506	Land Before Time I	200	19.99	20
10507	Land Before Time II	156	19.99	20
10508	Land Before Time III	190	19.99	20
10509	Land Before Time IV	60	19.99	20
10601	Sleeping Beauty	300	24.99	20
10701	When Harry Met Sally	120	19.99	30
10800	Dirty Harry	140	14.99	30
10900	Dr. Zhivago	100	24.99	30

Customers

CNO	CNAME	STREET	ZIP	PHONE
1111	Charles	123 Main St.	67226	316-636-5555
2222	Bertram	237 Ash Ave.	67226	316-689-5555
3333	Barbara	111 Inwood St.	60606	316-111-1234

Orders

ONO	CNO	ENO	RECEIVED	SHIPPED
1020	1111	1000	10-DEC-94	12-DEC-94
1021	1111	1000	12-JAN-95	15-JAN-95
1022	2222	1001	13-FEB-95	20-FEB-95
1023	3333	1000	20-JUN-97	null

Odetails

ONO	PNO	QTY
1020	10506	1
1020	10507	1
1020	10508	2
1020	10509	3
1021	10601	4
1022	10601	1
1022	10701	1
1023	10800	1
1023	10900	1

Zip codes

ZIP	CITY
67226	Wichita
60606	Fort Dodge
50302	Kansas City
54444	Columbia
66002	Liberal
61111	Fort Hays

a. Write a script named ***prog3a.sql*** for creating the ZIPCODES, CUSTOMERS and EMPLOYEES tables.

b. Write a script named ***prog3b.sql*** for population the ZIPCODES, CUSTOMERS and EMPLOYEES tables.

c. Write a PL/SQL subprogram that takes the old and new values of the zip code and performs an update of zip code values in the mail-order database described on page 9 of the textbook. Note that zip code values appear in three different tables: ZIPCODES, CUSTOMERS and EMPLOYEES. Note also the foreign key referential integrities on zip code. You can assume that a given zip code is always mapped to exactly one city. However, your program should handle spcial situations such as when the zip code to be changed does not exist in the database. In such cases, an appropriate message should be printed. Further, your program should always maintain the integrity of the underlying database.

   You should wrap your subprogram within an anonymous PL/SQL block.
   Save your program in the script file ***prog3c.sql***

2. (50 Points) Create the following database tables through a PL/SQL subprogram:

```
CREATE TABLE dept
( DEPTNO        NUMBER(3) PRIMARY KEY,
  DNAME         VARCHAR2(16),
  LOC           VARCHAR2(16));

CREATE TABLE emp
( EMPNO         NUMBER(4) PRIMARY KEY,
  ENAME         VARCHAR2(16),
  JOB           VARCHAR2(16),
  MGR           NUMBER(4),
  HIREDATE              DATE,
  SAL           NUMBER(7, 2),
  COMM          NUMBER(7, 2),
  DEPTNO  NUMBER(3) NOT NULL REFERENCES DEPT(DEPTNO));
```

After the tables have been created, in the same PL/SQL block, insert the following data into the tables. When inserting records for each table, if an error occurred, your PL/SQL block should commit the records that have been inserted before the one that caused the error and ignore the rest of the records starting from the one that caused the error condition.

```
Dept = {(10, 'ACCOUNTING', 'NEW YORK'),
        (20, 'RESEARCH',   'DALLAS'),
        (30, 'SALES',      'CHICAGO'),
        (40, 'OPERATIONS', 'WASHINGTON (D.C.)'),
        (50, 'MARKETING', 'BOSTON')}

emp = { (7839, 'KING',   'PRESIDENT',  NULL, '17-NOV-81', 5000, NULL, 10),
        (7698, 'BLAKE',  'MANAGER',  7839, '01-MAY-81', 2850, NULL, 30),
        (7782, 'CLARK',  'MANAGER',  7839, '09-JUN-81', 2450, NULL, 10),
        (7566, 'JONES',  'MANAGER',  7839, '02-APR-81', 2975, NULL, 20),
        (7654, 'MARTIN', 'SALESMAN', 7698, '28-SEP-81', 1250, 1400, 30),
        (7499, 'ALLEN',  'SALESMAN', 7698, '20-FEB-81', 1600,  300, 30),
        (7844, 'TURNER', 'SALESMAN', 7698, '08-SEP-81', 1500, NULL, 30),
        (7900, 'JAMES',  'CLERK',    7698, '03-DEC-81',  950, NULL, 30),
        (7521, 'WARD',   'SALESMAN', 7698, '22-FEB-81', 1250,  500, 30),
        (7902, 'FORD',   'ANALYST',  7566, '03-DEC-81', 3000, NULL, 20),
        (7369, 'SMITH',  'CLERK',    7902, '17-DEC-81',  800, NULL, 20),
        (7788, 'SCOTT',  'ANALYST',  7566, '09-DEC-82', 4000, NULL, 20),
        (7876, 'ADAMS',  'CLERK',    7788, '12-JAN-83', 1100, NULL, 20),
        (7934, 'MILLER', 'CLERK',    7782, '22-JAN-82', 1300, NULL, 10),
        (7698, 'BLAKE',  'MANAGER',  7839, '01-MAY-81', 2850, NULL, 30),
        (7935, 'JONES', 'ACCOUNT',    7782, '22-JAN-82', 1700, NULL, 10)}
```

Save your program in the script file *prog3d.sql*

Note: There are two parts to this assignment; each part may require you to submit a file. So please create a folder for this assignment and submit an electronic copy of your solution files of every question/part, all in one folder zipped and named "LastName HW3" and must be submitted to your D2L/Assignment 3 Submission page. I will give you one submission locations on the course web site.

Again: For example, for assignment #3, you need to create a folder named your **LastName HW3** under your c: home directory and save your script files **prog3a.sql, prog3b.sql, prog3c.sql** and **prog3d.sql** under this folder. Then zip the folder and then submit the zipped file to your D2L/Assignment 3 Submission link.

SUBMIT YOUR HW3 FOLDER AS ZIP FILE TO YOUR D2L ASSIGNMENT 3 SUBMISSION LINK FOR GRADING. Make sure only one copy is submitted.