Jay Fuller

Final Project

--Drop Tables:

drop table order\_details;

drop table orders;

drop table inventory;

drop table product;

drop table warehouse;

drop table dependents;

drop table employee;

drop table department;

drop table benefits;

drop table supplier;

drop table shipping;

drop table customer;

--Create Tables:

CREATE TABLE CUSTOMER

(CUST# INT NOT NULL ,

C\_FNAME VARCHAR(20) NOT NULL ,

C\_LNAME VARCHAR(20) NOT NULL ,

STREET VARCHAR(30) ,

CITY VARCHAR(20) ,

STATE CHAR(2) ,

ZIP CHAR(5) ,

PRIMARY KEY (CUST#));

--Customer

Insert into customer values('123','John','Smith','2000 Main Street','Oxford','MS','38655');

Insert into customer values('456','Leslie','Jones','555 North Road','Oxford','MS','38655');

Insert into customer values('1','Boss','Hog','1 Big House Street','Jackson','MS','39216');

Insert into customer values('13','John','Currence','13 Big Bad Breakfast Avenue','Oxford','MS','38655');

Insert into customer values('24','Yaboi','Tierney','15 Hospital Street','Chicago','IL','60605');

Insert into customer values('574','Jimmbo','Jimmy','256 St. Andrews Drive','Jackson','MS','39211');

Insert into customer values('7','Ash','Ketchum','151 Catch Em All Lane','Memphis','TN','37501');

Insert into customer values('99','Mike','Wazowsky','300 Monster Street','Oxford','MS','38655');

Insert into customer values('675','Houston','Nutt','125 Football Street','Tupelo','MS','38801');

Insert into customer values('111','Danny','Devito','250 Congress Park Drive','Boca Raton','FL','33445');

CREATE TABLE SHIPPING

(SHIPPING# INT NOT NULL ,

S\_COMPANY VARCHAR(30) NOT NULL ,

S\_COST INT NOT NULL ,

S\_PHONE CHAR(10) ,

S\_METHOD VARCHAR(20) ,

PRIMARY KEY (SHIPPING#));

--Shipping

Insert into shipping values('1','CompanyA','100','5551111111','Ground');

Insert into shipping values('2','CompanyA','200','5552222222','Ground');

Insert into shipping values('3','CompanyA','300','5553333333','Air');

Insert into shipping values('4','CompanyB','300','5554444444','Air');

Insert into shipping values('5','CompanyB','100','5556666666','Ground');

Insert into shipping values('6','CompanyA','200','8889999999','Ground');

Insert into shipping values('7','CompanyC','500','8887777777','Air');

Insert into shipping values('8','CompanyC','100','8885555555','Train');

Insert into shipping values('9','CompanyA','200','4449999999','Train');

Insert into shipping values('10','CompanyB','600','4448888888','Air');

CREATE TABLE SUPPLIER

(SUPPLIER\_ID INT NOT NULL ,

SUPP\_NAME VARCHAR(20) NOT NULL ,

SUPP\_CITY VARCHAR(20) ,

SUPP\_STATE CHAR(2) ,

SUPP\_ZIP CHAR(5) ,

PRIMARY KEY (SUPPLIER\_ID));

--Supplier

Insert into supplier values('22','SupplierA','Jackson','MS','39211');

Insert into supplier values('24','SupplierB','Jackson','MS','39211');

Insert into supplier values('26','SupplierC','Madison','MS','39110');

Insert into supplier values('28','SupplierD','Madison','MS','39110');

Insert into supplier values('33','SupplierE','Memphis','TN','37501');

Insert into supplier values('35','SupplierF','Memphis','TN','37501');

Insert into supplier values('37','SupplierG','Nashville','TN','37011');

Insert into supplier values('39','SupplierH','Boca Raton','FL','33445');

Insert into supplier values('44','SupplierI','Boca Raton','FL','33445');

Insert into supplier values('46','SupplierJ','Delray Beach','FL','33484');

CREATE TABLE BENEFITS

(POLICY# INT NOT NULL ,

HEALTHCARE VARCHAR(40) NOT NULL ,

DENTAL VARCHAR(3) ,

EYES VARCHAR(3) ,

PRESCRIPTION VARCHAR(10) ,

FAMILY\_COVERAGE VARCHAR(10) ,

PRIMARY KEY (POLICY#));

--Benefits

Insert into benefits values('101','UnitedHealth','NO','NO','PARTIAL','SPOUSE');

Insert into benefits values('102','UnitedHealth','NO','YES','PARTIAL','ALL');

Insert into benefits values('103','UnitedHealth','YES','YES','FULL','ALL');

Insert into benefits values('201','Aetna','NO','NO','PARTIAL','SPOUSE');

Insert into benefits values('202','Aetna','NO','YES','PARTIAL','ALL');

Insert into benefits values('203','Aetna','YES','YES','FULL','ALL');

Insert into benefits values('301','Blue Cross/Blue Shield','NO','NO','PARTIAL','SPOUSE');

Insert into benefits values('302','Blue Cross/Blue Shield','NO','YES','PARTIAL','ALL');

Insert into benefits values('303','Blue Cross/Blue Shield','YES','NO','FULL','ALL');

Insert into benefits values('304','Blue Cross/Blue Shield','YES','YES','FULL','ALL');

CREATE TABLE DEPARTMENT

(DEPT\_ID INT NOT NULL ,

DEPT\_NAME VARCHAR(30) NOT NULL ,

DEPT\_REGION VARCHAR(20) ,

MANAGER\_ID INT NOT NULL ,

MGR\_START\_DATE VARCHAR(10) ,

PRIMARY KEY (DEPT\_ID));

--Department

Insert into department values('99','Management','South','1234','2004-05-22');

Insert into department values('98','Administration','South','2345','2004-05-23');

Insert into department values('97','Research','South','3456','2006-09-19');

Insert into department values('96','Purchasing','East','4567','2004-12-03');

Insert into department values('95','Marketing','North','5678','2005-10-14');

Insert into department values('94','Human Resource','South','6789','2008-04-14');

Insert into department values('93','Accounting','South','7890','2004-06-26');

Insert into department values('92','Finance','North','8901','2004-08-22');

Insert into department values('91','Sales','East','9012','2006-01-29');

Insert into department values('90','Information Technology','South','4321','2009-10-31');

CREATE TABLE EMPLOYEE

(EMP\_ID INT NOT NULL ,

EMP\_FNAME VARCHAR(20) NOT NULL ,

EMP\_LNAME VARCHAR(20) NOT NULL ,

SALARY INT ,

EMP\_CITY VARCHAR(20) ,

EMP\_STATE CHAR(2) ,

EMP\_DEPT\_NUM INT NOT NULL ,

POLICY# INT NOT NULL ,

PRIMARY KEY (EMP\_ID) ,

FOREIGN KEY (EMP\_DEPT\_NUM) REFERENCES DEPARTMENT(DEPT\_ID));

--Employee

Insert into employee values('1234','Aristides','Aquino','60000','Jackson','MS','99','102');

Insert into employee values('2345','Billy','Hamilton','65000','Jackson','MS','99','103');

Insert into employee values('3456','Curtis','Granderson','63000','Jackson','MS','98','102');

Insert into employee values('4567','Dee','Gordon','55000','Oxford','MS','97','102');

Insert into employee values('5678','Eugenio','Suarez','70000','Oxford','MS','93','302');

Insert into employee values('6789','Freddie','Freeman','55000','Oxford','MS','90','302');

Insert into employee values('7891','Giancarlo','Stanton','68000','Tupelo','MS','91','303');

Insert into employee values('8901','Trea','Turner','75000','Boca Raton','FL','96','304');

Insert into employee values('9012','Byron','Buxton','45000','Boca Raton','FL','91','201');

Insert into employee values('4321','Joey','Freshwater','99000','Boca Raton','FL','99','203');

CREATE TABLE DEPENDENTS

(EMP\_ID INT NOT NULL REFERENCES EMPLOYEE(EMP\_ID) ,

DEP\_NAME VARCHAR(20) NOT NULL ,

SEX CHAR(1) ,

DOB VARCHAR(10) ,

RELATIONSHIP VARCHAR(10) NOT NULL ,

PRIMARY KEY (EMP\_ID, DEP\_NAME));

--Dependents

Insert into dependents values('1234','Allen','M','1989-10-31','Son');

Insert into dependents values('1234','Scott','M','1992-09-26','Son');

Insert into dependents values('1234','Ann','F','1965-04-13','Spouse');

Insert into dependents values('3456','Charles','M','1970-01-21','Spouse');

Insert into dependents values('5678','Ellen','F','1995-06-24','Daughter');

Insert into dependents values('5678','Eileen','F','1960-12-31','Spouse');

Insert into dependents values('6789','Frances','F','1992-03-09','Daughter');

Insert into dependents values('4321','Jimmy','M','1991-10-15','Son');

Insert into dependents values('4321','John','M','1994-12-26','Daughter');

Insert into dependents values('4321','Janice','F','1963-02-14','Spouse');

CREATE TABLE WAREHOUSE

(WAREHOUSE# INT NOT NULL ,

W\_SIZE VARCHAR(20) NOT NULL ,

W\_CITY VARCHAR(20) ,

W\_STATE CHAR(2) ,

W\_ZIP CHAR(5) ,

PRIMARY KEY (WAREHOUSE#));

--Warehouse

Insert into warehouse values('111','10000 sqft','Oxford','MS','38655');

Insert into warehouse values('222','20000 sqft','Oxford','MS','38655');

Insert into warehouse values('333','30000 sqft','Oxford','MS','38655');

Insert into warehouse values('444','40000 sqft','Jackson','MS','39216');

Insert into warehouse values('555','50000 sqft','Jackson','MS','39216');

Insert into warehouse values('666','60000 sqft','Jackson','MS','39216');

Insert into warehouse values('777','50000 sqft','Tupelo','MS','38801');

Insert into warehouse values('888','40000 sqft','Tupelo','MS','38801');

Insert into warehouse values('999','30000 sqft','Boca Raton','FL','33445');

Insert into warehouse values('1000','20000 sqft','Boca Raton','FL','33445');

CREATE TABLE PRODUCT

(PRODUCT# INT NOT NULL ,

P\_NAME VARCHAR(20) NOT NULL ,

P\_COLOR VARCHAR(20) ,

P\_WEIGHT INT ,

SUPPLIER\_ID INT NOT NULL REFERENCES SUPPLIER(SUPPLIER\_ID),

PRIMARY KEY (PRODUCT#));

--Product

Insert into product values('1111','ProductX','Red','10','22');

Insert into product values('2222','ProductY','Blue','10','22');

Insert into product values('3333','ProductZ','Gray','40','44');

Insert into product values('4444','ProductA','Orange','30','33');

Insert into product values('5555','ProductB','Yellow','40','33');

Insert into product values('6666','ProductC','Green','80','33');

Insert into product values('7777','ProductJ','Purple','10','28');

Insert into product values('8888','ProductK','White','80','28');

Insert into product values('9999','ProductL','Black','45','24');

Insert into product values('10000','ProductF','Powder Blue','15','46');

CREATE TABLE INVENTORY

(WAREHOUSE# INT NOT NULL REFERENCES WAREHOUSE(WAREHOUSE#) ,

PRODUCT# INT NOT NULL REFERENCES PRODUCT(PRODUCT#) ,

QTY\_IN\_STOCK INT NOT NULL ,

QTY\_AVAILABLE INT ,

QTY\_ORDERED INT ,

PRIMARY KEY (WAREHOUSE#, PRODUCT#));

--Inventory

Insert into inventory values('111','1111','100','80','100');

Insert into inventory values('111','2222','200','200','40');

Insert into inventory values('111','3333','300','150','300');

Insert into inventory values('333','4444','150','100','25');

Insert into inventory values('444','5555','200','200','0');

Insert into inventory values('444','6666','75','75','225');

Insert into inventory values('555','5555','100','50','100');

Insert into inventory values('555','7777','700','400','500');

Insert into inventory values('888','9999','250','200','135');

Insert into inventory values('888','10000','150','55','123');

CREATE TABLE ORDERS

(ORDER# INT NOT NULL ,

O\_DATE VARCHAR(20) ,

CUST# INT NOT NULL REFERENCES CUSTOMER(CUST#) ,

EMP\_ID INT NOT NULL REFERENCES EMPLOYEE(EMP\_ID) ,

SHIPPING# INT NOT NULL REFERENCES SHIPPING(SHIPPING#) ,

PRIMARY KEY (ORDER#));

--Orders

Insert into orders values('1','9/2019','123','1234','1');

Insert into orders values('2','9/2019','123','1234','2');

Insert into orders values('3','8/2019','123','1234','6');

Insert into orders values('4','10/2019','13','4567','4');

Insert into orders values('5','9/2019','13','4567','10');

Insert into orders values('6','9/2019','456','8901','7');

Insert into orders values('7','10/2019','456','8901','7');

Insert into orders values('8','10/2019','99','4321','3');

Insert into orders values('9','9/2019','99','2345','6');

Insert into orders values('10','10/2019','13','7890','10');

CREATE TABLE ORDER\_DETAILS

(ORDER# INT NOT NULL REFERENCES ORDERS(ORDER#) ,

PRODUCT# INT NOT NULL REFERENCES PRODUCT(PRODUCT#) ,

UNIT\_PRICE INT ,

QTY\_P\_PART INT ,

TOTAL\_P\_PART INT ,

PRIMARY KEY (ORDER#, PRODUCT#));

--Order Details

Insert into order\_details values('1','1111','50','20','1000');

Insert into order\_details values('2','1111','20','20','400');

Insert into order\_details values('3','3333','25','10','250');

Insert into order\_details values('4','6666','70','35','2450');

Insert into order\_details values('5','7777','15','40','600');

Insert into order\_details values('6','3333','40','10','400');

Insert into order\_details values('7','2222','85','15','1275');

Insert into order\_details values('8','4444','15','30','450');

Insert into order\_details values('9','9999','50','50','2500');

Insert into order\_details values('10','1111','33','20','660');

--1 inserts a new employee into the database from user input

--returns new employee info and their healthcare provider

CREATE OR REPLACE FUNCTION find\_fname (p\_emp\_id employee.emp\_id%type)

RETURN employee.emp\_fname%type

IS

ret\_name employee.emp\_fname%type;

BEGIN

select emp\_fname into ret\_name

from employee

where p\_emp\_id = emp\_id;

return ret\_name;

END;

/

CREATE OR REPLACE FUNCTION find\_lname (p\_emp\_id employee.emp\_id%type)

RETURN employee.emp\_lname%type

IS

ret\_name employee.emp\_lname%type;

BEGIN

select emp\_lname into ret\_name

from employee

where p\_emp\_id = emp\_id;

return ret\_name;

END;

/

CREATE OR REPLACE FUNCTION find\_sal (p\_emp\_id employee.emp\_id%type)

RETURN employee.salary%type

IS

ret\_sal employee.salary%type;

BEGIN

select salary into ret\_sal

from employee

where p\_emp\_id = emp\_id;

return ret\_sal;

END;

/

CREATE OR REPLACE FUNCTION find\_city (p\_emp\_id employee.emp\_id%type)

RETURN employee.emp\_city%type

IS

ret\_city employee.emp\_city%type;

BEGIN

select emp\_city into ret\_city

from employee

where p\_emp\_id = emp\_id;

return ret\_city;

END;

/

CREATE OR REPLACE FUNCTION find\_state (p\_emp\_id employee.emp\_id%type)

RETURN employee.emp\_state%type

IS

ret\_state employee.emp\_state%type;

BEGIN

select emp\_state into ret\_state

from employee

where p\_emp\_id = emp\_id;

return ret\_state;

END;

/

CREATE OR REPLACE FUNCTION find\_benefit (p\_emp\_id employee.emp\_id%type)

RETURN benefits.healthcare%type

IS

ret\_healthcare benefits.healthcare%type;

BEGIN

select b.healthcare into ret\_healthcare

from benefits b, employee e

where p\_emp\_id = e.emp\_id

and b.policy# = e.policy#;

return ret\_healthcare;

END;

/

SET SERVEROUTPUT ON;

DECLARE

eempid employee.emp\_id%type;

finame employee.emp\_fname%type;

laname employee.emp\_lname%type;

empid employee.emp\_id%type;

empsal employee.salary%type;

empcity employee.emp\_city%type;

empstate employee.emp\_state%type;

fname employee.emp\_fname%type;

lname employee.emp\_lname%type;

health benefits.healthcare%type;

city employee.emp\_city%type;

state employee.emp\_state%type;

sal employee.salary%type;

BEGIN

--user inputs

finame:='&FirstName';

laname:='&LastName';

empid:='&EmployeeID';

empsal:='&Salary';

empcity:='&City';

empstate:='&State';

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

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

insert into employee values(empid, finame, laname, empsal, empcity, empstate, '93', '303');

DBMS\_OUTPUT.PUT\_LINE('Here is the new employee you have entered.');

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

fname:=find\_fname(empid);

lname:=find\_lname(empid);

sal:=find\_sal(empid);

city:=find\_city(empid);

state:=find\_state(empid);

health:=find\_benefit(empid);

DBMS\_OUTPUT.PUT\_LINE('Employee Name: '||trim(fname)||' '||trim(lname));

DBMS\_OUTPUT.PUT\_LINE('Location: '||city||' '||state);

DBMS\_OUTPUT.PUT\_LINE('Salary: $'||sal);

DBMS\_OUTPUT.PUT\_LINE('Healthcare Provider: '||health);

END;

--Delete from employee where emp\_id = [whatever id user entered]

--2 discount amount calculations; enter A, B, or C for code

--enter 1-10 for order number

SET SERVEROUTPUT ON

DECLARE

discount\_code CHAR(1);

total\_amt order\_details.total\_p\_part%type;

disc\_amt NUMBER(5,2) :=0;

order\_num order\_details.order#%type;

BEGIN

order\_num :='&OrderNumber';

discount\_code :='&DiscountCode';

select total\_p\_part into total\_amt

from order\_details

where order# = order\_num;

IF

discount\_code = 'A'

THEN

disc\_amt := total\_amt \* .05;

ELSIF discount\_code = 'B' THEN

disc\_amt := total\_amt \* .15;

ELSIF discount\_code = 'C' THEN

disc\_amt := total\_amt \* .15;

END IF;

DBMS\_OUTPUT.PUT\_LINE('Total Order Cost: '||total\_amt);

DBMS\_OUTPUT.PUT\_LINE('Discount Code: '||discount\_code);

DBMS\_OUTPUT.PUT\_LINE('Discount Amount: '||disc\_amt);

END;

--3 Procedure for entering new department info

--Prompt user for all department info

CREATE OR REPLACE PROCEDURE Insert\_Department

(p\_dept\_id IN department.dept\_id%type,

p\_dept\_name IN department.dept\_name%type,

p\_dept\_region IN department.dept\_region%type,

p\_manager\_id IN department.manager\_id%type,

p\_mgr\_start\_date IN department.mgr\_start\_date%type)

IS

BEGIN

insert into department values(p\_dept\_id, p\_dept\_name, p\_dept\_region, p\_manager\_id, p\_mgr\_start\_date);

commit;

END;

/

SET SERVEROUTPUT ON

DECLARE

deptid department.dept\_id%type;

deptname department.dept\_name%type;

deptregion department.dept\_region%type;

deptmgrid department.manager\_id%type;

deptmgrstart department.mgr\_start\_date%type;

BEGIN

deptid :='&DepartmentID';

deptname :='&DepartmentName';

deptregion :='&DepartmentRegion';

deptmgrid :='&ManagerID';

deptmgrstart :='&ManagerStartDate';

Insert\_Department(deptid, deptname, deptregion, deptmgrid, deptmgrstart);

END;

--4 Find number of orders an employee is associated with

--Enter 1234, 2345, 3456, 4567, or 8901

CREATE OR REPLACE FUNCTION f\_emp\_orders (f\_emp employee.emp\_id%type)

RETURN orders.order#%type

IS

ret\_num\_orders orders.order#%type;

BEGIN

select count(order#) into ret\_num\_orders from orders o, employee e

where e.emp\_id = o.emp\_id

and e.emp\_id = f\_emp;

return ret\_num\_orders;

END;

/

SET SERVEROUTPUT ON

DECLARE

emp\_num employee.emp\_id%type :='&Employee\_ID';

num\_orders orders.order#%type;

BEGIN

num\_orders := f\_emp\_orders(emp\_num);

DBMS\_OUTPUT.PUT\_LINE('Employee Number '||emp\_num||' works on '||num\_orders);

END;

--5 incomplete

CREATE OR REPLACE PROCEDURE p\_product

(p\_product IN product.product#%type,

p\_id OUT product.product#%type,

p\_name OUT product.p\_name%type,

p\_color OUT product.p\_color%type,

p\_weight OUT product.p\_weight%type,

p\_supp OUT product.supplier\_id%type)

IS

BEGIN

select product#, p\_name, p\_color, p\_weight, supplier\_id

into p\_id, p\_name, p\_color, p\_weight, p\_supp

from product

where product# = p\_product;

END;

/

CREATE OR REPLACE PROCEDURE p\_supplier

(p\_supplier IN supplier.supplier\_id%type,

p\_sname OUT supplier.supp\_name%type,

p\_scity OUT supplier.supp\_city%type,

p\_sstate OUT supplier.supp\_state%type)

IS

BEGIN

select supp\_name, supp\_city, supp\_state

into p\_sname, p\_scity, p\_sstate

from supplier

where supplier\_id = p\_supplier;

END;

/

SET SERVEROUTPUT ON

DECLARE

pid\_fk EXCEPTION;

PRAGMA Exception\_Init(pid\_fk,+0100);

pid product.product#%type;

p\_id product.product#%type;

p\_name product.p\_name%type;

p\_color product.p\_color%type;

p\_weight product.p\_weight%type;

p\_supp product.supplier\_id%type;

p\_sname supplier.supp\_name%type;

p\_scity supplier.supp\_city%type;

p\_sstate supplier.supp\_state%type;

bl number;

tot number;

tot2 number;

tot3 number;

dif number;

BEGIN

pid:='&product\_id';

p\_product(pid, p\_id,p\_name,p\_color,p\_weight,p\_supp);

p\_supplier(p\_supp,p\_sname, p\_scity, p\_sstate);

DBMS\_OUTPUT.PUT\_LINE('Product ID; '||p\_id);

DBMS\_OUTPUT.PUT\_LINE('Name: '||p\_name);

DBMS\_OUTPUT.PUT\_LINE('Color: '||p\_color);

DBMS\_OUTPUT.PUT\_LINE('Weight: '||p\_weight);

DBMS\_OUTPUT.PUT\_LINE('Supplier ID: '||p\_supp);

DBMS\_OUTPUT.PUT\_LINE('Supplier Name: '||p\_sname);

DBMS\_OUTPUT.PUT\_LINE('Supplier City: '||p\_scity);

DBMS\_OUTPUT.PUT\_LINE('Supplier State: '||p\_sstate);

END;

--6 big block

--using multiple cursors to go through database to categorize employees by department

--also lists their healthcare provider

--one cursor with no data to show what happens when no employee there

SET SERVEROUTPUT ON;

DECLARE

CURSOR cur1 IS --cursor for department 99

select

e.emp\_fname, e.emp\_lname, b.healthcare, dept\_name

from

employee e, department, benefits b

where

e.emp\_dept\_num = dept\_id

and

e.policy# = b.policy#

and

e.emp\_dept\_num = '99';

CURSOR cur2 IS --cursor for department 98

select

e.emp\_fname, e.emp\_lname, b.healthcare, dept\_name

from

employee e, department, benefits b

where

e.emp\_dept\_num = dept\_id

and

e.policy# = b.policy#

and

e.emp\_dept\_num = '98';

CURSOR cur3 IS --cursor for department 91

select

e.emp\_fname, e.emp\_lname, b.healthcare, dept\_name

from

employee e, department, benefits b

where

e.emp\_dept\_num = dept\_id

and

e.policy# = b.policy#

and

e.emp\_dept\_num = '91';

CURSOR cur4 IS --cursor for department 90

select

e.emp\_fname, e.emp\_lname, b.healthcare, dept\_name

from

employee e, department, benefits b

where

e.emp\_dept\_num = dept\_id

and

e.policy# = b.policy#

and

e.emp\_dept\_num = '90';

CURSOR cur5 IS --cursor for department 94

select

e.emp\_fname, e.emp\_lname, b.healthcare, dept\_name

from

employee e, department, benefits b

where

e.emp\_dept\_num = dept\_id

and

e.policy# = b.policy#

and

e.emp\_dept\_num = '94';

TYPE type\_emp\_dept IS RECORD(

fn employee.emp\_fname%type,

ln employee.emp\_lname%type,

bn benefits.healthcare%type,

dp department.dept\_name%type

);

rec\_emp\_dept type\_emp\_dept;

BEGIN

OPEN cur1;

rec\_emp\_dept.dp:='99';

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

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

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

LOOP

FETCH cur1 INTO rec\_emp\_dept;

EXIT WHEN cur1%notfound;

DBMS\_OUTPUT.PUT\_LINE('First Name: '||rec\_emp\_dept.fn);

DBMS\_OUTPUT.PUT\_LINE('Last Name: '||rec\_emp\_dept.ln);

DBMS\_OUTPUT.PUT\_LINE('Healthcare: '||rec\_emp\_dept.bn);

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

END LOOP;

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

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

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

CLOSE cur1;

OPEN cur2;

rec\_emp\_dept.dp:='98';

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

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

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

LOOP

FETCH cur2 INTO rec\_emp\_dept;

EXIT WHEN cur2%notfound;

DBMS\_OUTPUT.PUT\_LINE('First Name: '||rec\_emp\_dept.fn);

DBMS\_OUTPUT.PUT\_LINE('Last Name: '||rec\_emp\_dept.ln);

DBMS\_OUTPUT.PUT\_LINE('Healthcare: '||rec\_emp\_dept.bn);

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

END LOOP;

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

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

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

CLOSE cur2;

OPEN cur3;

rec\_emp\_dept.dp:='91';

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

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

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

LOOP

FETCH cur3 INTO rec\_emp\_dept;

EXIT when cur3%notfound;

DBMS\_OUTPUT.PUT\_LINE('First Name: '||rec\_emp\_dept.fn);

DBMS\_OUTPUT.PUT\_LINE('Last Name: '||rec\_emp\_dept.ln);

DBMS\_OUTPUT.PUT\_LINE('Healthcare: '||rec\_emp\_dept.bn);

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

END LOOP;

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

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

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

CLOSE cur3;

OPEN cur4;

rec\_emp\_dept.dp:='90';

DBMS\_OUTPUT.PUT\_LINE('Information Technology ');

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

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

LOOP

FETCH cur4 INTO rec\_emp\_dept;

EXIT WHEN cur4%notfound;

DBMS\_OUTPUT.PUT\_LINE('First Name: '||rec\_emp\_dept.fn);

DBMS\_OUTPUT.PUT\_LINE('Last Name: '||rec\_emp\_dept.ln);

DBMS\_OUTPUT.PUT\_LINE('Healthcare: '||rec\_emp\_dept.bn);

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

END LOOP;

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

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

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

CLOSE cur4;

OPEN cur5;

rec\_emp\_dept.dp:='94';

DBMS\_OUTPUT.PUT\_LINE('Human Resources ');

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

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

LOOP

FETCH cur5 INTO rec\_emp\_dept;

EXIT WHEN cur5%notfound;

DBMS\_OUTPUT.PUT\_LINE('First Name: '||rec\_emp\_dept.fn);

DBMS\_OUTPUT.PUT\_LINE('Last Name: '||rec\_emp\_dept.ln);

DBMS\_OUTPUT.PUT\_LINE('Healthcare: '||rec\_emp\_dept.bn);

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

END LOOP;

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

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

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

CLOSE cur5;

END;

--7 counts the number of employees in a city

--displays employee name and number in that city

CREATE OR REPLACE PACKAGE pg\_count

IS

PROCEDURE p\_count

(p\_count IN employee.emp\_city%type,

p\_p OUT number);

END pg\_count;

/

CREATE OR REPLACE PACKAGE body pg\_count AS

PROCEDURE p\_count

(p\_count IN employee.emp\_city%type,

p\_p OUT number)

IS

BEGIN

select count(emp\_city)

into p\_p

from employee

where emp\_city = p\_count;

END p\_count;

END; -- package end

/

SET SERVEROUTPUT ON

DECLARE

CURSOR j IS -- cursor for jackson

select emp\_fname, emp\_lname

from employee

where emp\_city ='Jackson';

CURSOR o IS -- cursor for oxford

select emp\_fname, emp\_lname

from employee

where emp\_city ='Oxford';

CURSOR t IS -- cursor for tupelo

select emp\_fname, emp\_lname

from employee

where emp\_city ='Tupelo';

CURSOR b IS -- cursor for boca

select emp\_fname, emp\_lname

from employee

where emp\_city ='Boca Raton';

TYPE type\_emp\_city IS RECORD(

firstname employee.emp\_fname%type,

lastname employee.emp\_lname%type);

rec\_emp\_city type\_emp\_city;

ct number;

BEGIN

OPEN j;

DBMS\_OUTPUT.PUT\_LINE('List of employees located in Jackson ');

LOOP

FETCH j into rec\_emp\_city;

EXIT WHEN j%notfound;

DBMS\_OUTPUT.PUT\_LINE('Employee Name: '||trim(rec\_emp\_city.firstname)||' '||trim(rec\_emp\_city.lastname));

END LOOP;

pg\_count.p\_count('Jackson',ct);

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

DBMS\_OUTPUT.PUT\_LINE('Total employees in Jackson: '||ct);

ct:=0;

CLOSE j;

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

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

OPEN o;

DBMS\_OUTPUT.PUT\_LINE('List of employees located in Oxford ');

LOOP

FETCH o into rec\_emp\_city;

EXIT WHEN o%notfound;

DBMS\_OUTPUT.PUT\_LINE('Employee Name: '||trim(rec\_emp\_city.firstname)||' '||trim(rec\_emp\_city.lastname));

END LOOP;

pg\_count.p\_count('Oxford',ct);

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

DBMS\_OUTPUT.PUT\_LINE('Total employees in Oxford: '||ct);

ct:=0;

CLOSE o;

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

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

OPEN t;

DBMS\_OUTPUT.PUT\_LINE('List of employees located in Tupelo ');

LOOP

FETCH t into rec\_emp\_city;

EXIT WHEN t%notfound;

DBMS\_OUTPUT.PUT\_LINE('Employee Name: '||trim(rec\_emp\_city.firstname)||' '||trim(rec\_emp\_city.lastname));

END LOOP;

pg\_count.p\_count('Tupelo',ct);

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

DBMS\_OUTPUT.PUT\_LINE('Total employees in Tupelo: '||ct);

ct:=0;

CLOSE t;

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

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

OPEN b;

DBMS\_OUTPUT.PUT\_LINE('List of employees located in Boca Raton ');

LOOP

FETCH b into rec\_emp\_city;

EXIT WHEN b%notfound;

DBMS\_OUTPUT.PUT\_LINE('Employee Name: '||trim(rec\_emp\_city.firstname)||' '||trim(rec\_emp\_city.lastname));

END LOOP;

pg\_count.p\_count('Boca Raton',ct);

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

DBMS\_OUTPUT.PUT\_LINE('Total employees in Boca Raton: '||ct);

ct:=0;

CLOSE b;

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

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

END;

--8 Changes an employees city, state, and deptnum and updates their info in the employee table

--error if incorrect emp\_id entered

--enter any valid employee id (1234,2345,3456, etc.)

--drop table employee3;

CREATE TABLE employee3 as select \* from employee; --create employee table to edit

/

SET SERVEROUTPUT ON;

DECLARE

new\_city\_pk EXCEPTION;

PRAGMA EXCEPTION\_INIT(new\_city\_pk, -02291);

empid employee3.emp\_id%type;

efn employee3.emp\_fname%type;

eln employee3.emp\_lname%type;

sal employee3.salary%type;

cit employee3.emp\_city%type;

estate employee3.emp\_state%type;

edeptn employee3.emp\_dept\_num%type;

pol employee3.policy#%type;

prevcity employee3.emp\_city%type;

prevstate employee3.emp\_state%type;

prevdept employee3.emp\_dept\_num%type;

BEGIN

empid:='&Employee\_ID';

insert into department

values ('13','New York Office','North','4321','10-30-2000');

select e.emp\_id, e.emp\_fname, e.emp\_lname, e.salary, e.emp\_city, e.emp\_state, e.emp\_dept\_num, e.policy#

into empid, efn, eln, sal, prevcity, prevstate, prevdept, pol

from employee3 e

where e.emp\_id = empid;

update employee3

set emp\_city = 'New York City', emp\_state = 'NY', emp\_dept\_num = '13'

where emp\_id = empid;

select e.emp\_id, e.emp\_lname, e.emp\_fname, e.emp\_city, e.emp\_state, e.emp\_dept\_num

into empid, eln, efn, cit, estate, edeptn

from employee3 e

where e.emp\_id = empid;

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

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

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

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

DBMS\_OUTPUT.PUT\_LINE('Last Name: '||trim(eln));

DBMS\_OUTPUT.PUT\_LINE('First Name: '||trim(efn));

DBMS\_OUTPUT.PUT\_LINE('Previous City: '||prevcity);

DBMS\_OUTPUT.PUT\_LINE('Previous State: '||prevstate);

DBMS\_OUTPUT.PUT\_LINE('Salary: '||sal);

DBMS\_OUTPUT.PUT\_LINE('Previous Department Number: '||prevdept);

DBMS\_OUTPUT.PUT\_LINE('Policy #: '||pol);

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

DBMS\_OUTPUT.PUT\_LINE('New City: '||cit);

DBMS\_OUTPUT.PUT\_LINE('New State: '||estate);

DBMS\_OUTPUT.PUT\_LINE('New Department Number: '||edeptn);

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

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

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

EXCEPTION

WHEN new\_city\_pk THEN

DBMS\_OUTPUT.PUT\_LINE('That Employee ID was not found.');

DBMS\_OUTPUT.PUT\_LINE('Please try a different Employee ID.');

END;

--9 Uses a trigger to count the number of employees each time one is deleted

--uses a cursor to go through the table to fetch all employee names

--drop table employee4;

--drop table test\_emp4;

create table employee4 as select \* from employee;

/

create table test\_emp4 --create table for trigger

(user\_name varchar2(20),

date\_time timestamp(6),

num\_records number);

/

CREATE OR REPLACE TRIGGER test\_deemp4 --delete trigger

after delete on employee4

declare

num\_records Number:=0;

BEGIN

delete from test\_emp4;

select count(\*) into num\_records from employee4;

insert into test\_emp4 values(user, sysdate, num\_records);

END;

/

SET SERVEROUTPUT ON

DECLARE

CURSOR cur IS

select emp\_fname, emp\_lname

from employee4;

validemp\_fk EXCEPTION;

PRAGMA Exception\_Init(validemp\_fk,+0100);

TYPE emp4 IS RECORD(

fname employee4.emp\_fname%type,

lname employee4.emp\_lname%type

);

rec emp4;

eid employee4.emp\_id%type;

nr number;

pr number;

BEGIN

eid:='&enter\_employeeid';

select count(emp\_id)

into nr

from employee4;

open cur;

loop

FETCH cur INTO rec;

EXIT WHEN cur%notfound;

DBMS\_OUTPUT.PUT\_LINE('Employee Name: '||trim(rec.fname)||' '||trim(rec.lname));

END LOOP;

DBMS\_OUTPUT.PUT\_LINE('The current total number of employees is: '||nr);

CLOSE cur;

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

nr:=0;

DBMS\_OUTPUT.PUT\_LINE('If the employee id is in the employee4 table it will show more info. If not there is an exception.');

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

select count(emp\_id)

into pr

from employee4

where emp\_id=eid;

if(pr=1)then

DBMS\_OUTPUT.PUT\_LINE('Now the employee will be deleted.');

delete from employee4

where emp\_id = eid;

select num\_records

into nr

from test\_emp4

where date\_time=sysdate;

OPEN cur;

LOOP

FETCH cur INTO rec;

EXIT WHEN cur%notfound;

DBMS\_OUTPUT.PUT\_LINE('Employee Name: '||trim(rec.fname)||' '||trim(rec.lname));

END LOOP;

DBMS\_OUTPUT.PUT\_LINE('The current number of employees is: '||nr);

CLOSE cur;

ELSE

RAISE validemp\_fk;

END IF;

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

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

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

DBMS\_OUTPUT.PUT\_LINE('The employee id is not in the employee4 table. Please try again.');

END;

Output:

(Input 1234 to test; enter 1234 again to test exception)

Employee Name: Aristides Aquino

Employee Name: Billy Hamilton

Employee Name: Curtis Granderson

Employee Name: Dee Gordon

Employee Name: Eugenio Suarez

Employee Name: Freddie Freeman

Employee Name: Giancarlo Stanton

Employee Name: Trea Turner

Employee Name: Byron Buxton

Employee Name: Joey Freshwater

The current total number of employees is: 10

If the employee id is in the employee4 table it will show more info. If not there is an exception.

Now the employee will be deleted.

Employee Name: Billy Hamilton

Employee Name: Curtis Granderson

Employee Name: Dee Gordon

Employee Name: Eugenio Suarez

Employee Name: Freddie Freeman

Employee Name: Giancarlo Stanton

Employee Name: Trea Turner

Employee Name: Byron Buxton

Employee Name: Joey Freshwater

The current number of employees is: 9

The employee id is not in the employee4 table. Please try again.