PL/SQL PROJECT FOR MIS 409

By: Carter Butz

carterbutz@gmail.com

Carter Butz

carterbutz@gmail.com

Dr. Sumali Conlon

MIS 409 Project

7 April 2019

1) Use your own database to create PL/SQL blocks that use all of the techniques covered during the semester.

1) SQL commands for create tables and insert data

2) Each PL/SQL block should include:

1) English query

2) PL/SQL

3) Output (also how to run your PL/SQL blocks)

This database is created from scratch by me for the purpose of executing queries we have learned over the semester.

TABLES: Customer, Employee, Product, Supplier, Orders, OrderDetails, Shipping, Departments, Payroll, Materials, Warehouse, Warehouse Details

--Drop Tables

```
DROP TABLE Customer CASCADE CONSTRAINTS;

DROP TABLE Employee CASCADE CONSTRAINTS;

DROP TABLE Product CASCADE CONSTRAINTS;

DROP TABLE Supplier CASCADE CONSTRAINTS;

DROP TABLE Orders CASCADE CONSTRAINTS;

DROP TABLE OrderDetails CASCADE CONSTRAINTS;

DROP TABLE Shipping CASCADE CONSTRAINTS;

DROP TABLE Departments CASCADE CONSTRAINTS;

DROP TABLE Payroll CASCADE CONSTRAINTS;

DROP TABLE Materials CASCADE CONSTRAINTS;

DROP TABLE Warehouse CASCADE CONSTRAINTS;

DROP TABLE WarehouseDetails2 CASCADE CONSTRAINTS;
```

```
--Create table customer

CREATE TABLE Customer

(CustID CHAR(5) NOT NULL,

CustFname VARCHAR(15),

CustLname VARCHAR(15),

CustCompany VARCHAR(25),

CustPhone NUMBER(10,0),

CustEmail VARCHAR (30),

CustCity VARCHAR(20),

CustState Char(2),

Primary Key(CustID));

Insert into Customer values ('11111','Bill','Gates','Company One','8041112222','bgates@gmail.com','Los Angeles','CA');

Insert into Customer values ('22222','Steve','Jobs','Company Two','8042223333','sjobs@gmail.com','San Francisco','CA');

Insert into Customer values ('33333','Jeff','Bezos','Company Three','8043334444','jbezos@gmail.com','Seattle','WA');

Insert into Customer values ('44444','Mark','Zuckerberg','Company Four','8044445555','mzuckerberg@gmail.com','Dallas','TX');

Insert into Customer values ('55555','Elon','Musk','Company Five','8045556666','emusk@gmail.com','New York','NY');

Commit;


--create table warehouse

CREATE TABLE Warehouse

(WarehouseID CHAR(5) NOT NULL,

WarehouseCity VARCHAR(20),

WarehouseState Char(10),

WarehouseColor VARCHAR (20),
```

```
PRIMARY KEY (WarehouseID));
Insert into Warehouse values ('10000','Dallas','TX','Blue');
Insert into Warehouse values ('20000','Seattle','WA','Green');
Insert into Warehouse values ('30000','Richmond','VA','Grey');
Insert into Warehouse values ('40000','New York','NY','Black');
Insert into Warehouse values ('50000','Memphis','TN','White');
Commit;

--create table supplier
CREATE TABLE Supplier
(SupID CHAR(5) NOT NULL,
SupCompany VARCHAR(30),
SupAddress VARCHAR(40),
SupCity VARCHAR (20),
SupPhone NUMBER(10, 0),
Primary Key (SupID));
Insert into Supplier values ('99999','Supplier One','123 East Avenue','Houston','1234445555');
Insert into Supplier values ('88888','Supplier Two','123 City Boulevard','Nashville','1231112222');
Insert into Supplier values ('77777','Supplier Three','456 University','Miami','1232223333');
Insert into Supplier values ('66666','Supplier Four','234 Shadow Lane','Washington D.C.','1233334444');
Insert into Supplier values ('55556','Supplier Five','777 Second Street','Memphis','1235556666');
Commit;

--create table shipping
CREATE TABLE Shipping
(ShippingID CHAR(5) NOT NULL,
ShippingCompany VARCHAR(30),
ShippingCost VARCHAR(10),
ShippingDate DATE,
```

```
Primary Key (ShippingID));
Insert into Shipping Values ('00011','Shipping One','50.00',To_DATE('01/01/2017','mm/dd/yyyy'));
Insert into Shipping Values ('00022','Shipping Two','10.00',To_DATE('02/01/2017','mm/dd/yyyy'));
Insert into Shipping Values ('00033','Shipping Three','5.00',To_DATE('05/02/2017','mm/dd/yyyy'));
Insert into Shipping Values ('00044','Shipping Four','0.00',To_DATE('10/15/2017','mm/dd/yyyy'));
Insert into Shipping Values ('00055','Shipping Five','25.00',To_DATE('12/25/2017','mm/dd/yyyy'));
Commit;

--create table departments
CREATE TABLE Departments
(DeptID CHAR(5) NOT NULL,
DeptName VARCHAR(25),
DeptPhone NUMBER(10, 0),
DeptCity VARCHAR(20),
DeptState Char(2),
DeptMgr NUMBER(*, 0),
Primary Key (DeptID));
Insert into Departments values ('11000','Department One','7771112222','Seattle','WA','12234');
Insert into Departments values ('12000','Department Two','7772223333','Nashville','TN','12344');
Insert into Departments values ('13000','Department Three','7773334444','Richmond','VA','11234');
Insert into Departments values ('14000','Department Four','7774445555','San Francisco','CA','12234');
Insert into Departments values ('15000','Department Five','7775556666','New York','NY','12345');
Insert into Departments values ('16000','Department Six','7776667777','Oxford','MS','12234');
Commit;

--create table employee
CREATE TABLE Employee
(EmpID CHAR(5) NOT NULL,
EmpFname VARCHAR(15),
```

```
EmpLname VARCHAR(15),

EmpPhone NUMBER(10, 0),

EmpSSN NUMBER(9, 0),

EmpCity VARCHAR(20),

EmpState Char(10),

EmpSalary CHAR(10),

Deptno CHAR(5),

Primary Key (EmpID),

FOREIGN Key (Deptno) REFERENCES Departments(DeptID));

Insert into Employee values
('12345','Larry','Page','6621112222','123456789','Nashville','TN','30000','15000');

Insert into Employee values
('11234','Tim','Cook','6622223333','234567890','Richmond','VA','75000','12000');

Insert into Employee values
('12234','Satya','Nadella','6623334444','345678901','Seattle','WA','90000','11000');

Insert into Employee values ('12334','Travis','Kalanick','6624445555','456789012','San
Francisco','CA','45000','16000');

Insert into Employee values ('12344','Jack','Dorsey','6625556666','567890123','New
York','NY','50000','13000');

Commit;

--create table orderdetails

CREATE TABLE OrderDetails

(OrderID CHAR(5) NOT NULL,

ProdID CHAR(5) NOT NULL,

QtyOrderPerProduct NUMBER(*, 0),

SubTotalPerProduct NUMBER(*, 2),

ShippingID CHAR(5),

WarehouseID CHAR(5),

PRIMARY KEY (OrderID, ProdID),

FOREIGN KEY (ShippingID) REFERENCES Shipping(ShippingID),
```

```
FOREIGN KEY (WarehouseID) REFERENCES Warehouse(WarehouseID));

insert into OrderDetails values ('00002','67890','1000','50.00','00022','20000');
insert into OrderDetails values ('00004','23456','5000','150.00','00044','30000');
insert into OrderDetails values ('00001','45678','50','20.00','00011','40000');
insert into OrderDetails values ('00005','56789','10','250.00','00055','10000');
insert into OrderDetails values ('00003','34567','250','9.99','00033','50000');
commit;

--create table product
CREATE TABLE Product
(ProdID CHAR(5) NOT NULL,
ProdName VARCHAR(25),
ProdPrice NUMBER(*, 2),
ProdColor VARCHAR (10),
SupID CHAR(5),
Primary Key (ProdID),
Foreign Key (SupID) REFERENCES Supplier(SupID));
Insert into Product values ('23456','Product One','150.00','Green','88888');
Insert into Product values ('34567','Product Two','9.99','Blue','66666');
Insert into Product values ('45678','Product Three','20.00','Red','77777');
Insert into Product values ('56789','Product Four','250.00','Black','99999');
Insert into Product values ('67890','Product Five','50.00','White','55556');
Commit;

--create table orders
CREATE TABLE Orders
(OrderID CHAR(5) NOT NULL,
OrderDate DATE,
```

```
OrderTotalAmt NUMBER(*, 0),

CreditCard# VARCHAR(25),

CustID CHAR(5),

Primary Key (OrderID),

Foreign Key (CustID) REFERENCES Customer(CustID));

Insert into Orders values
('00001',TO_DATE('01/02/2017','mm/dd/yyyy'),'2500','1111222233334444','33333');

Insert into Orders values
('00002',TO_DATE('02/03/2017','mm/dd/yyyy'),'250000','2222333344445555','44444');

Insert into Orders values
('00003',TO_DATE('05/06/2017','mm/dd/yyyy'),'62500','3333444455556666','22222');

Insert into Orders values
('00004',TO_DATE('10/22/2017','mm/dd/yyyy'),'50000','4444555566667777','55555');

Insert into Orders values
('00005',TO_DATE('12/30/2017','mm/dd/yyyy'),'1500','5555666677778888','11111');

Commit;


--create table payroll

CREATE TABLE Payroll

(EmpID CHAR(5) NOT NULL,

DeptID CHAR(5)  NOT NULL,

EmpHours VARCHAR(10),

EmpPay VARCHAR(10),

essn NUMBER(9,0),

Primary Key (EmpID, DeptID));

Insert into Payroll values ('12345','12000','20.00','40.00','123456789');

Insert into Payroll values ('11234','13000','40.00','70.00','234567890');

Insert into Payroll values ('12334','14000','60.00','20.00','345678901');

Insert into Payroll values ('12344','15000','15.00','35.00','456789012');

Insert into Payroll values ('12234','11000','35.00','60.00','567890123');
```

```
Commit;

--create table materials
CREATE TABLE Materials
(MatID CHAR(5) NOT NULL,
MatName VARCHAR(25),
MatSupply NUMBER(*,0),
SupID CHAR(5),
Primary Key (MatID),
Foreign Key (SupID) REFERENCES Supplier(SupID));
Insert into Materials values ('99990','Material One','100','88888');
Insert into Materials values ('88880','Material Two','250','66666');
Insert into Materials values ('77770','Material Three','50','99999');
Insert into Materials values ('66660','Material Four','500','77777');
Commit;

--create table warehousedetails
CREATE TABLE WarehouseDetails2
(WarehouseID CHAR(5) NOT NULL,
ProdID CHAR(5),
ProdQty CHAR(10),
PRIMARY KEY (WarehouseID, ProdID));
Insert into WarehouseDetails2 values ('10000','34567','5000');
Insert into WarehouseDetails2 values ('20000','67890','300');
Insert into WarehouseDetails2 values ('30000','23456','1000');
Insert into WarehouseDetails2 values ('40000','45678','10000');
Insert into WarehouseDetails2 values ('50000','56789','100');
Commit;
```

--CARTER BUTZ PROJECT ENGLISH QUERIES / PL/SQL / OUTPUT

--Number 1 (225 lines) Goes through 6 departments and gives a raise of 10%, 5%, 15%, 20%, 7.5%, and 3.5% respectively. This is what a salary increase would mean for each department and how it affects the overall cost using cursors. This information is then reset at the end of the query.

```
SET SERVEROUTPUT ON;

DECLARE

Cursor cur_emp_department IS

Select EmpFname, EmpLname, EmpSalary, DeptName

From employee e, departments d

WHERE deptno = DeptID

AND e.deptno = '11000';


Cursor cur IS

Select EmpFname, EmpLname, EmpSalary, DeptName

From employee e, departments d

WHERE deptno = DeptID

AND e.deptno = '12000';


Cursor cur_emp IS

Select EmpFname, EmpLname, EmpSalary, DeptName

From employee e, departments d

WHERE deptno = DeptID

AND e.deptno = '15000';


Cursor cur_emp_three IS

Select EmpFname, EmpLname, EmpSalary, DeptName

From employee e, departments d

WHERE deptno = DeptID

AND e.deptno = '13000';
```

```
Cursor cur_emp_four IS
Select EmpFname, EmpLname, EmpSalary, DeptName
From employee e, departments d
WHERE deptno = DeptID
AND e.deptno = '14000';

Cursor cur_emp_six IS
Select EmpFname, EmpLname, EmpSalary, DeptName
From employee e, departments d
WHERE deptno = DeptID
AND e.deptno = '16000';

TYPE type_emp_dept IS RECORD (
firstname employee.EmpFname%type,
lastname employee.EmpLname%type,
empsalary employee.EmpSalary%type,
departmentname departments.deptname%type);

rec_emp_dept  type_emp_dept;
totalsalary NUMBER := 0;
totaloriginalsalary NUMBER := 0;
grandtotaloriginal NUMBER := 0;
grandtotalnew NUMBER := 0;
math NUMBER := 0;
BEGIN

Open cur_emp_department;
Rec_emp_dept.departmentname := 'Department One';
```

```
dbms_output.put_line('Department One***** salary increased by 10% ***** ');
 dbms_output.put_line(' ');
 dbms_output.put_line('     original salary          new Salary ');
 dbms_output.put_line(' ');

 LOOP

 FETCH cur_emp_department INTO rec_emp_dept;
 EXIT WHEN cur_emp_department%notfound;
totaloriginalsalary := totaloriginalsalary + to_number(rec_emp_dept.empsalary);
 dbms_output.put_line('  ' || trim(rec_emp_dept.firstname) || '  ' || rec_emp_dept.lastname || ' '
 || TO_CHAR(rec_emp_dept.empsalary,'$999,999.99') || ' ' || TO_CHAR(rec_emp_dept.empsalary *
1.10, '$999,999.00') );
totalsalary := totalsalary + (to_number(rec_emp_dept.EmpSalary) * 1.10);
END LOOP;
Grandtotaloriginal := grandtotaloriginal + totaloriginalsalary;
Grandtotalnew := grandtotalnew + totaloriginalsalary;
Dbms_output.put_line('');
dbms_output.put_line(' Total Original Salary for Department One ==== ' ||
TO_CHAR(totaloriginalsalary,'$999,999.00') );
 dbms_output.put_line(' Total New Salary for Department One==== ' ||
TO_CHAR(totalsalary,'$999,999.00'));
 totalsalary := 0;
 totaloriginalsalary := 0;
dbms_output.put_line('================================================================');

CLOSE cur_emp_department;
OPEN cur;
 rec_emp_dept.departmentName := 'Department Two';
 dbms_output.put_line('Department Two**** salary increased by 5%**** ');
```

```
 dbms_output.put_line(' ');
 dbms_output.put_line('     original salary         new Salary ');
 dbms_output.put_line(' ');
 LOOP
 FETCH cur INTO rec_emp_dept;
EXIT WHEN cur%notfound;
totaloriginalsalary := totaloriginalsalary + to_number(rec_emp_dept.empsalary);
 dbms_output.put_line(' ' || trim(rec_emp_dept.firstname) || ' ' || rec_emp_dept.lastname || ' '
 || TO_CHAR(rec_emp_dept.empsalary,'$999,999.99') || ' ' || TO_CHAR(
 rec_emp_dept.empsalary * 1.05,
 '$999,999.00' ) );
totalsalary := totalsalary + ( to_number(rec_emp_dept.empsalary) * 1.05 );
 END LOOP;

grandtotaloriginal := grandtotaloriginal + totaloriginalsalary;
 grandtotalnew := grandtotalnew + totalsalary;
dbms_output.put_line('');
 dbms_output.put_line(' Total Original Salary for Department Two==== '
 || TO_CHAR(totaloriginalsalary,'$999,999.00') );
 dbms_output.put_line(' Total New Salary for Department Two==== '
 || TO_CHAR(totalsalary,'$999,999.00') );
 totalsalary := 0;
 totaloriginalsalary := 0;
dbms_output.put_line('================================================================');
CLOSE cur;

 OPEN cur_emp;
 rec_emp_dept.departmentname := 'Department Five';
 dbms_output.put_line('Department Five***** salary increased by 15%*****');
```

```
 dbms_output.put_line(' ');

 dbms_output.put_line('     original salary      new Salary ');

 dbms_output.put_line(' ');

 LOOP

 FETCH cur_emp INTO rec_emp_dept;

 EXIT WHEN cur_emp%notfound;

 totaloriginalsalary := totaloriginalsalary + to_number(rec_emp_dept.empsalary);

 dbms_output.put_line(' '|| trim(rec_emp_dept.firstname) || ' ' || rec_emp_dept.lastname || ' ' ||
TO_CHAR(rec_emp_dept.empsalary,'$999,999.99') || ' ' || TO_CHAR( rec_emp_dept.empsalary * 1.15,
'$999,999.00' ) );

totalsalary := totalsalary + (to_number(rec_emp_dept.empsalary) * 1.15);

END LOOP;


grandtotaloriginal := grandtotaloriginal + totaloriginalsalary;

 grandtotalnew := grandtotalnew + totalsalary;

 dbms_output.put_line('');

 dbms_output.put_line(' Total Original Salary for Department Five==== ' ||
TO_CHAR(totaloriginalsalary,'$999,999.00') );

 dbms_output.put_line(' Total New Salary for Department Five==== ' ||
TO_CHAR(totalsalary,'$999,999.00'));

dbms_output.put_line('');


dbms_output.put_line('=================================================================
');

CLOSE cur_emp;


OPEN cur_emp_three;

 rec_emp_dept.departmentName := 'Department Three';

 dbms_output.put_line('Department Three**** salary increased by 20%**** ');

 dbms_output.put_line(' ');

 dbms_output.put_line('     original salary        new Salary ');
```

```
 dbms_output.put_line(' ');
 LOOP
 FETCH cur_emp_three INTO rec_emp_dept;
EXIT WHEN cur_emp_three%notfound;
totaloriginalsalary := totaloriginalsalary + to_number(rec_emp_dept.empsalary);
 dbms_output.put_line(' ' || trim(rec_emp_dept.firstname) || ' ' || rec_emp_dept.lastname || ' '
 || TO_CHAR(rec_emp_dept.empsalary,'$999,999.99') || ' ' || TO_CHAR(
 rec_emp_dept.empsalary * 1.20,
 '$999,999.00' ) );
totalsalary := totalsalary + ( to_number(rec_emp_dept.empsalary) * 1.20);
 END LOOP;

grandtotaloriginal := grandtotaloriginal + totaloriginalsalary;
 grandtotalnew := grandtotalnew + totalsalary;
dbms_output.put_line('');
 dbms_output.put_line(' Total Original Salary for Department Three==== '
 || TO_CHAR(totaloriginalsalary,'$999,999.00') );
 dbms_output.put_line(' Total New Salary for Department Three==== '
 || TO_CHAR(totalsalary,'$999,999.00') );
 totalsalary := 0;
 totaloriginalsalary := 0;
dbms_output.put_line('================================================================');
CLOSE cur_emp_three;

OPEN cur_emp_four;
 rec_emp_dept.departmentName := 'Department Four';
 dbms_output.put_line('Department Four**** salary increased by 7.5%**** ');
 dbms_output.put_line(' ');
 dbms_output.put_line('    original salary        new Salary ');
```

```
 dbms_output.put_line(' ');
 LOOP
 FETCH cur_emp_four INTO rec_emp_dept;
EXIT WHEN cur_emp_four%notfound;
totaloriginalsalary := totaloriginalsalary + to_number(rec_emp_dept.empsalary);
 dbms_output.put_line(' ' || trim(rec_emp_dept.firstname) || ' ' || rec_emp_dept.lastname || ' '
 || TO_CHAR(rec_emp_dept.empsalary,'$999,999.99') || ' ' || TO_CHAR(
 rec_emp_dept.empsalary * 1.075,
 '$999,999.00' ) );
totalsalary := totalsalary + ( to_number(rec_emp_dept.empsalary) * 1.075);
 END LOOP;

grandtotaloriginal := grandtotaloriginal + totaloriginalsalary;
 grandtotalnew := grandtotalnew + totalsalary;
dbms_output.put_line('');
 dbms_output.put_line(' Total Original Salary for Department Four==== '
 || TO_CHAR(totaloriginalsalary,'$999,999.00') );
 dbms_output.put_line(' Total New Salary for Department Four==== '
 || TO_CHAR(totalsalary,'$999,999.00') );
 totalsalary := 0;
 totaloriginalsalary := 0;
dbms_output.put_line('================================================================');
CLOSE cur_emp_four;

OPEN cur_emp_six;
 rec_emp_dept.departmentName := 'Department Six';
 dbms_output.put_line('Department Six**** salary increased by 3.5%**** ');
 dbms_output.put_line(' ');
 dbms_output.put_line('    original salary        new Salary ');
```

```
 dbms_output.put_line(' ');
 LOOP
 FETCH cur_emp_six INTO rec_emp_dept;
EXIT WHEN cur_emp_six%notfound;
totaloriginalsalary := totaloriginalsalary + to_number(rec_emp_dept.empsalary);
 dbms_output.put_line(' ' || trim(rec_emp_dept.firstname) || ' ' || rec_emp_dept.lastname || ' '
 || TO_CHAR(rec_emp_dept.empsalary,'$999,999.99') || ' ' || TO_CHAR(
 rec_emp_dept.empsalary * 1.035,
 '$999,999.00' ) );
totalsalary := totalsalary + ( to_number(rec_emp_dept.empsalary) * 1.035 );
 END LOOP;

grandtotaloriginal := grandtotaloriginal + totaloriginalsalary;
 grandtotalnew := grandtotalnew + totalsalary;
dbms_output.put_line('');
 dbms_output.put_line(' Total Original Salary for Department Six==== '
 || TO_CHAR(totaloriginalsalary,'$999,999.00') );
 dbms_output.put_line(' Total New Salary for Department Six==== '
 || TO_CHAR(totalsalary,'$999,999.00') );
 totalsalary := 0;
 totaloriginalsalary := 0;
dbms_output.put_line('================================================================');
CLOSE cur_emp_six;

 dbms_output.put_line(' Grand Total Original Salaries for the departments ==== '||
TO_CHAR(grandtotaloriginal,'$999,999,999.00') );
 dbms_output.put_line(' Grand Total of salaries after raises ==== '||
TO_CHAR(grandtotalnew,'$999,999,999.00') );
 totalsalary := 0;
```

```
 totaloriginalsalary := 0;
 dbms_output.put_line('');
math:=grandtotalnew-grandtotaloriginal;
 dbms_output.put_line('The difference is: '||TO_CHAR(math,'$999,999,999.00'));
end;
```

OUTPUT:

```
Department One***** salary increased by 10% *****

    original salary         new Salary

 Satya  Nadella   $90,000.00   $99,000.00

 Total Original Salary for Department One ====   $90,000.00
 Total New Salary for Department One====   $99,000.00
================================================================
Department Two**** salary increased by 5%****

    original salary        new Salary

 Tim Cook   $75,000.00   $78,750.00

 Total Original Salary for Department Two====   $75,000.00
 Total New Salary for Department Two====   $78,750.00
================================================================
Department Five***** salary increased by 15%*****

    original salary      new Salary

 Larry Page   $30,000.00   $34,500.00
```

Total Original Salary for Department Five==== $30,000.00

Total New Salary for Department Five==== $34,500.00

==========

Department Three**** salary increased by 20%****

original salary new Salary

Jack Dorsey $50,000.00 $60,000.00

Total Original Salary for Department Three==== $80,000.00

Total New Salary for Department Three==== $94,500.00

==========

Department Four**** salary increased by 7.5%****

original salary new Salary

Total Original Salary for Department Four==== $.00

Total New Salary for Department Four==== $.00

==========

Department Six**** salary increased by 3.5%****

original salary new Salary

Travis Kalanick $45,000.00 $46,575.00

Total Original Salary for Department Six==== $45,000.00

Total New Salary for Department Six==== $46,575.00

==========

Grand Total Original Salaries for the departments ==== $320,000.00

Grand Total of salaries after raises ==== $344,325.00

The difference is: $24,325.00

END OUTPUT:

--Number 2 (139 lines) Displays information ordered anything from us and displays their customer id, name, company, phone, email, the total amount spent by customer, and the number of orders placed by customer. Then customers who spent less than $50000 are displayed to remind them to come back. This program utilizes procedures.

```
CREATE OR REPLACE PROCEDURE p_order
(p_order IN orders.custid%type,
p_num OUT number,
p_total out number)

IS
BEGIN
select count(orderid), sum(ordertotalamt)
into p_num, p_total
FROM orders
where custid = p_order;
end;
/

SET SERVEROUTPUT ON
DECLARE
c_id customer.custid%type;
c_fname customer.custfname%type;
```

```
c_lname customer.custlname%type;
c_company customer.custcompany%type;
c_phone customer.custphone%type;
c_email customer.custemail%type;
total number;
totalorders number;

BEGIN
select custid, custfname, custlname, custcompany, custphone, custemail
into c_id, c_fname, c_lname, c_company, c_phone, c_email
from Customer
where custid = '11111';
p_order(c_id,totalorders, total);
dbms_output.put_line('Customer ID: '||c_id);
dbms_output.put_line('Customer Name: '||trim(c_fname)||''||trim(c_lname));
dbms_output.put_line('Customer Company: '||c_company);
dbms_output.put_line('Customer Phone: '||c_phone);
dbms_output.put_line('Customer Email: '||c_email);
dbms_output.put_line('Amount Customer Spent:$'||total||'.00');
dbms_output.put_line('Number of Orders by Customer: '||totalorders);
dbms_output.put_line('');
dbms_output.put_line('==================================');
dbms_output.put_line('');

select custid, custfname, custlname, custcompany, custphone, custemail
into c_id, c_fname, c_lname, c_company, c_phone, c_email
from Customer
where custid = '22222';
p_order(c_id,totalorders, total);
```

```
dbms_output.put_line('Customer ID: '||c_id);
dbms_output.put_line('Customer Name: '||trim(c_fname)||''||trim(c_lname));
dbms_output.put_line('Customer Company: '||c_company);
dbms_output.put_line('Customer Phone: '||c_phone);
dbms_output.put_line('Customer Email: '||c_email);
dbms_output.put_line('Amount Customer Spent:$'||total||'.00');
dbms_output.put_line('Number of Orders by Customer: '||totalorders);
dbms_output.put_line('');
dbms_output.put_line('==================================');
dbms_output.put_line('');

select custid, custfname, custlname, custcompany, custphone, custemail
into c_id, c_fname, c_lname, c_company, c_phone, c_email
from Customer
where custid = '33333';
p_order(c_id,totalorders, total);
dbms_output.put_line('Customer ID: '||c_id);
dbms_output.put_line('Customer Name: '||trim(c_fname)||''||trim(c_lname));
dbms_output.put_line('Customer Company: '||c_company);
dbms_output.put_line('Customer Phone: '||c_phone);
dbms_output.put_line('Customer Email: '||c_email);
dbms_output.put_line('Amount Customer Spent:$'||total||'.00');
dbms_output.put_line('Number of Orders by Customer: '||totalorders);
dbms_output.put_line('');
dbms_output.put_line('==================================');
dbms_output.put_line('');

select custid, custfname, custlname, custcompany, custphone, custemail
into c_id, c_fname, c_lname, c_company, c_phone, c_email
```

```
from Customer
where custid = '44444';
p_order(c_id,totalorders, total);
dbms_output.put_line('Customer ID: '||c_id);
dbms_output.put_line('Customer Name: '||trim(c_fname)||''||trim(c_lname));
dbms_output.put_line('Customer Company: '||c_company);
dbms_output.put_line('Customer Phone: '||c_phone);
dbms_output.put_line('Customer Email: '||c_email);
dbms_output.put_line('Amount Customer Spent:$'||total||'.00');
dbms_output.put_line('Number of Orders by Customer: '||totalorders);
dbms_output.put_line('');
dbms_output.put_line('==================================');
dbms_output.put_line('');

select custid, custfname, custlname, custcompany, custphone, custemail
into c_id, c_fname, c_lname, c_company, c_phone, c_email
from Customer
where custid = '55555';
p_order(c_id,totalorders, total);
dbms_output.put_line('Customer ID: '||c_id);
dbms_output.put_line('Customer Name: '||trim(c_fname)||''||trim(c_lname));
dbms_output.put_line('Customer Company: '||c_company);
dbms_output.put_line('Customer Phone: '||c_phone);
dbms_output.put_line('Customer Email: '||c_email);
dbms_output.put_line('Amount Customer Spent:$'||total||'.00');
dbms_output.put_line('Number of Orders by Customer: '||totalorders);
dbms_output.put_line('');
dbms_output.put_line('==================================');
```

```
dbms_output.put_line('Customers who spent less than $50000 are receiving this to remind you to come back and shop.');

dbms_output.put_line('');

dbms_output.put_line('==================================');


select custid, custfname, custlname, custcompany, custphone, custemail

into c_id, c_fname, c_lname, c_company, c_phone, c_email

from Customer

where custid = '11111';

p_order(c_id,totalorders, total);

dbms_output.put_line('Customer ID: '||c_id);

dbms_output.put_line('Customer Name: '||trim(c_fname)||''||trim(c_lname));

dbms_output.put_line('Customer Company: '||c_company);

dbms_output.put_line('Customer Phone: '||c_phone);

dbms_output.put_line('Customer Email: '||c_email);

dbms_output.put_line('Amount Customer Spent:$'||total||'.00');

dbms_output.put_line('Number of Orders by Customer: '||totalorders);

dbms_output.put_line('');

dbms_output.put_line('==================================');


select custid, custfname, custlname, custcompany, custphone, custemail

into c_id, c_fname, c_lname, c_company, c_phone, c_email

from Customer

where custid = '33333';

p_order(c_id,totalorders, total);

dbms_output.put_line('Customer ID: '||c_id);

dbms_output.put_line('Customer Name: '||trim(c_fname)||''||trim(c_lname));

dbms_output.put_line('Customer Company: '||c_company);

dbms_output.put_line('Customer Phone: '||c_phone);
```

```
dbms_output.put_line('Customer Email: '||c_email);
dbms_output.put_line('Amount Customer Spent:$'||total||'.00');
dbms_output.put_line('Number of Orders by Customer: '||totalorders);
dbms_output.put_line('');
dbms_output.put_line('==================================');
end;
```

--OUTPUT:

Procedure P_ORDER compiled

Customer ID: 11111

Customer Name: BillGates

Customer Company: Company One

Customer Phone: 8041112222

Customer Email: bgates@gmail.com

Amount Customer Spent:$1500.00

Number of Orders by Customer: 1

==========

Customer ID: 22222

Customer Name: SteveJobs

Customer Company: Company Two

Customer Phone: 8042223333

Customer Email: sjobs@gmail.com

Amount Customer Spent:$62500.00

Number of Orders by Customer: 1

==========

Customer ID: 33333

Customer Name: JeffBezos

Customer Company: Company Three

Customer Phone: 8043334444

Customer Email: jbezos@gmail.com

Amount Customer Spent:$2500.00

Number of Orders by Customer: 1

==========

Customer ID: 44444

Customer Name: MarkZuckerberg

Customer Company: Company Four

Customer Phone: 8044445555

Customer Email: mzuckerberg@gmail.com

Amount Customer Spent:$250000.00

Number of Orders by Customer: 1

==========

Customer ID: 55555

Customer Name: ElonMusk

Customer Company: Company Five

Customer Phone: 8045556666

Customer Email: emusk@gmail.com

Amount Customer Spent:$50000.00

Number of Orders by Customer: 1

==========

Customers who spent less than $50000 are receiving this to remind you to come back and shop.

==========

Customer ID: 11111

Customer Name: BillGates

Customer Company: Company One

Customer Phone: 8041112222

Customer Email: bgates@gmail.com

Amount Customer Spent:$1500.00

Number of Orders by Customer: 1

==========

Customer ID: 33333

Customer Name: JeffBezos

Customer Company: Company Three

Customer Phone: 8043334444

Customer Email: jbezos@gmail.com

Amount Customer Spent:$2500.00

Number of Orders by Customer: 1

==========

--END OUTPUT

--Number 3 (117 lines) The user can create their own employee using this program utilizing functions and exceptions. The user is prompted to enter ssn, first name, last name, location, phone number, and salary to make the employee part of the database. Although, if a user enters a social security number already in use, then an error will occur.

```
CREATE OR REPLACE FUNCTION find_fname (p_essn employee.empssn%type)
```

```
return employee.empfname%type IS
ret_name employee.empfname%type;
BEGIN
select empfname into ret_name
from employee
where p_essn = empssn;
return ret_name;
end;
/

CREATE OR REPLACE FUNCTION find_lname (p_essn employee.empssn%type)
return employee.emplname%type IS
ret_name employee.emplname%type;
BEGIN
select emplname into ret_name
from employee
where p_essn = empssn;
return ret_name;
end;
/

CREATE OR REPLACE FUNCTION find_phone (p_essn employee.empssn%type)
return employee.empphone%type IS
ret_phone employee.empphone%type;
BEGIN
select empphone into ret_phone
from employee
where p_essn = empssn;
return ret_phone;
```

```
end;
/

CREATE OR REPLACE FUNCTION find_city (p_essn employee.empssn%type)
return employee.empcity%type IS
ret_city employee.empcity%type;
BEGIN
select empcity into ret_city
from employee
where p_essn = empssn;
return ret_city;
end;
/

CREATE OR REPLACE FUNCTION find_state (p_essn employee.empssn%type)
return employee.empstate%type IS
ret_state employee.emplname%type;
BEGIN
select empstate into ret_state
from employee
where p_essn = empssn;
return ret_state;
end;
/

CREATE OR REPLACE FUNCTION find_salary (p_essn employee.empssn%type)
return employee.empsalary%type IS
ret_salary employee.empsalary%type;
BEGIN
```

```
select empsalary into ret_salary
from employee
where p_essn = empssn;
return ret_salary;
end;
/
SET SERVEROUTPUT ON;
DECLARE

ex_ssn_pk EXCEPTION;
PRAGMA EXCEPTION_INIT (ex_ssn_pk,-00001);

xssn employee.empssn%type;
xfirstname employee.empfname%type;
xlastname employee.emplname%type;
xphonenumber employee.empphone%type;
xcity employee.empcity%type;
xstate employee.empstate%type;
xsalary employee.empsalary%type;
essn employee.empssn%type;
fname employee.empfname%type;
lname employee.emplname%type;
phonenum employee.empphone%type;
city employee.empcity%type;
estate employee.empstate%type;
salary employee.empsalary%type;
BEGIN
xssn := '&SSN';
xfirstname := '&FirstName';
```

```
xlastname := '&LastName';
xphonenumber :='&PhoneNumber';
xcity :='&City';
xstate :='&State';
xsalary :='&Salary';
dbms_output.put_line('');
dbms_output.put_line('');

INSERT into employee
values('05322', xfirstname, xlastname, xphonenumber, xssn, xcity, xstate, xsalary, '11000');
dbms_output.put_line('Name, PhoneNumber, City, State, and Salary for your employee.');
dbms_output.put_line('');

fname := find_fname(xssn);
lname := find_lname(xssn);
phonenum := find_phone(xssn);
city := find_city(xssn);
estate := find_state(xssn);
salary := find_salary(xssn);

dbms_output.put_line('Employee Name: ' ||trim(fname)||''||trim(lname));
dbms_output.put_line('Employee Location: ' ||trim(city)||' '||trim(estate));
dbms_output.put_line('Employee Phone: ' ||trim(phonenum));
dbms_output.put_line('Employee Salary: ' ||TO_CHAR(salary,'$999,999.00'));

EXCEPTION
WHEN ex_ssn_pk THEN
dbms_output.put_line('You have entered a duplicate SSN. Retry.');
end;
```

OUTPUT:

Name, PhoneNumber, City, State, and Salary for your employee.

Employee Name: CarterButz

Employee Location: Oxford MS

Employee Phone: 1234567890

Employee Salary: $70,000.00

--Number 4 (127 lines) This program counts the number of employees working for each department and then displays their names and the number of employees working there. This program is using packages for the information.

```
CREATE OR REPLACE PACKAGE pg_number
IS
PROCEDURE p_number(p_number IN employee.deptno%type, p_num OUT NUMBER);
end pg_number;
/

CREATE OR REPLACE PACKAGE body pg_number AS
PROCEDURE p_number
(p_number IN employee.deptno%type,
p_num OUT NUMBER) IS
BEGIN
select count(deptno)
into p_num
from employee
where deptno = p_number;
end p_number;
end;
/
```

```
SET SERVEROUTPUT ON
DECLARE
CURSOR d1 IS
select empfname, emplname
from employee
where deptno = '11000';
CURSOR d2 IS
select empfname, emplname
from employee
where deptno = '12000';
CURSOR d3 IS
select empfname, emplname
from employee
where deptno = '13000';
CURSOR d4 IS
select empfname, emplname
from employee
where deptno = '14000';
CURSOR d5 IS
select empfname, emplname
from employee
where deptno = '15000';
CURSOR d6 IS
select empfname, emplname
from employee
where deptno = '16000';

TYPE type_emp_dep IS RECORD
(firstname employee.empfname%type,
```

```
lastname employee.emplname%type);

rec_emp_dept type_emp_dep;

tot number;


BEGIN


OPEN d1;

dbms_output.put_line('List of employees who work for Department One in Seattle, WA.');

LOOP

FETCH d1 INTO rec_emp_dept;

EXIT WHEN d1%notfound;

dbms_output.put_line('Employee Name:
'||trim(rec_emp_dept.firstname)||''||trim(rec_emp_dept.lastname));

END LOOP;

pg_number.p_number('11000',tot);

dbms_output.put_line('');

dbms_output.put_line('Total employees working for department one: '||tot);

tot := 0;

CLOSE d1;

dbms_output.put_line('');

dbms_output.put_line('========================================');


OPEN d2;

dbms_output.put_line('List of employees who work for Department Two in Seattle, WA.');

LOOP

FETCH d2 INTO rec_emp_dept;

EXIT WHEN d2%notfound;

dbms_output.put_line('Employee Name:
'||trim(rec_emp_dept.firstname)||''||trim(rec_emp_dept.lastname));
```

```
END LOOP;
pg_number.p_number('12000',tot);
dbms_output.put_line('');
dbms_output.put_line('Total employees working for department two: '||tot);
tot := 0;
CLOSE d2;
dbms_output.put_line('');
dbms_output.put_line('========================================');

OPEN d3;
dbms_output.put_line('List of employees who work for Department Three in Seattle, WA.');
LOOP
FETCH d3 INTO rec_emp_dept;
EXIT WHEN d3%notfound;
dbms_output.put_line('Employee Name:
'||trim(rec_emp_dept.firstname)||''||trim(rec_emp_dept.lastname));
END LOOP;
pg_number.p_number('13000',tot);
dbms_output.put_line('');
dbms_output.put_line('Total employees working for department three: '||tot);
tot := 0;
CLOSE d3;
dbms_output.put_line('');
dbms_output.put_line('========================================');
OPEN d4;
dbms_output.put_line('List of employees who work for Department Four in Seattle, WA.');
LOOP
FETCH d4 INTO rec_emp_dept;
EXIT WHEN d4%notfound;
```

```
dbms_output.put_line('Employee Name:
'||trim(rec_emp_dept.firstname)||''||trim(rec_emp_dept.lastname));

END LOOP;

pg_number.p_number('14000',tot);

dbms_output.put_line('');

dbms_output.put_line('Total employees working for department four: '||tot);

tot := 0;

CLOSE d4;

dbms_output.put_line('');

dbms_output.put_line('========================================');


OPEN d5;

dbms_output.put_line('List of employees who work for Department Five in Seattle, WA.');

LOOP

FETCH d5 INTO rec_emp_dept;

EXIT WHEN d5%notfound;

dbms_output.put_line('Employee Name:
'||trim(rec_emp_dept.firstname)||''||trim(rec_emp_dept.lastname));

END LOOP;

pg_number.p_number('11000',tot);

dbms_output.put_line('');

dbms_output.put_line('Total employees working for department five: '||tot);

tot := 0;

CLOSE d5;

dbms_output.put_line('');

end;
```

OUTPUT:

```
Package PG_NUMBER compiled
```

Package Body PG_NUMBER compiled

List of employees who work for Department One in Seattle, WA.

Employee Name: SatyaNadella

Employee Name: CarterButz

Total employees working for department one: 2

==========

List of employees who work for Department Two in Seattle, WA.

Employee Name: TimCook

Total employees working for department two: 1

==========

List of employees who work for Department Three in Seattle, WA.

Employee Name: JackDorsey

Total employees working for department three: 1

==========

List of employees who work for Department Four in Seattle, WA.

Total employees working for department four: 0

==========

List of employees who work for Department Five in Seattle, WA.

Employee Name: LarryPage

Total employees working for department five: 2

--Number 5 (129 lines) This program finds the salary, ssn, department number, city, and state using first and last name given by the user. The user also updates the salary but is reset at the end to keep the database integrity. If a user enters a name not in the database, an exception will occur.

```
CREATE OR REPLACE PACKAGE pg_empdep

IS

function find_sal(p_fname employee.empfname%type,p_lname employee.emplname%type) return
employee.empsalary%type;

function find_ssn(p_fname employee.empfname%type,p_lname employee.emplname%type) return
employee.empssn%type;

function find_dept(p_fname employee.empfname%type,p_lname employee.emplname%type) return
employee.deptno%type;

function find_city(p_fname employee.empfname%type,p_lname employee.emplname%type) return
employee.empcity%type;

function find_state(p_fname employee.empfname%type,p_lname employee.emplname%type) return
employee.empstate%type;

end pg_empdep;

/


CREATE OR REPLACE PACKAGE body pg_empdep

AS

function find_sal

(p_fname employee.empfname%type,

p_lname employee.emplname%type)

return employee.empsalary%type IS

disal employee.empsalary%type;

BEGIN

select empsalary
```

```
into disal
from employee
where empfname = p_fname
and emplname = p_lname;
return disal;
end find_sal;

function find_ssn
(p_fname employee.empfname%type,
p_lname employee.emplname%type)
return employee.empssn%type IS
dssn employee.empssn%type;
BEGIN
select empssn
into dssn
from employee
where empfname = p_fname
and emplname = p_lname;
return dssn;
end find_ssn;

function find_dept
(p_fname employee.empfname%type,
p_lname employee.emplname%type)
return employee.deptno%type IS
ddept employee.deptno%type;
BEGIN
select deptno
into ddept
```

```
from employee
where empfname = p_fname
and emplname = p_lname;
return ddept;
end find_dept;

function find_city
(p_fname employee.empfname%type,
p_lname employee.emplname%type)
return employee.empcity%type IS
dcity employee.empcity%type;
BEGIN
select empcity
into dcity
from employee
where empfname = p_fname
and emplname = p_lname;
return dcity;
end find_city;

function find_state
(p_fname employee.empfname%type,
p_lname employee.emplname%type)
return employee.empstate%type IS
dstate employee.empstate%type;
BEGIN
select empstate
into dstate
from employee
```

```
where empfname = p_fname
and emplname = p_lname;
return dstate;
end find_state;
end;

SET SERVEROUTPUT ON;
DECLARE
employee_name_fk EXCEPTION; PRAGMA EXCEPTION_INIT(employee_name_fk,+0100);
aa employee.empfname%type;
bb employee.emplname%type;
cc employee.empsalary%type;
dd employee.empssn%type;
ee employee.deptno%type;
ff employee.empcity%type;
gg employee.empstate%type;
c2 employee.empsalary%type;
c3 employee.empsalary%type;
st employee.empstate%type;
BEGIN
aa:='&Enter_First_Name';
bb:='&Enter_Last_Name';
cc:=pg_empdep.find_sal(aa,bb);
st:=pg_empdep.find_state(aa,bb);
dd:=pg_empdep.find_ssn(aa,bb);
ee:=pg_empdep.find_dept(aa,bb);
ff:=pg_empdep.find_city(aa,bb);
gg:=pg_empdep.find_state(aa,bb);
dbms_output.put_line('Employees Name: '||trim(aa)||' '||trim(bb));
```

```
dbms_output.put_line('Employees Salary: '||to_CHAR(cc,'$999,999.00'));
dbms_output.put_line('Employees SSN: '||trim(dd));
dbms_output.put_line('Employees Department: '||trim(ee));
dbms_output.put_line('Employees City: '||trim(ff));
dbms_output.put_line('Employees State: '||trim(gg));

c2:='&Enter_New_Salary';
UPDATE employee
SET empsalary = c2
WHERE empstate = st;

c3:=pg_empdep.find_sal(aa,bb);
dbms_output.put_line('');
dbms_output.put_line('New Salary for employee: '||to_CHAR(c3,'$999,999.00'));

UPDATE employee
SET empsalary = cc
WHERE empstate = st;
EXCEPTION when employee_name_fk then
dbms_output.put_line('');
dbms_output.put_line('');
dbms_output.put_line('Name is not found.');
end;
```

INPUT:

Tim[enter]Cook[enter]105000[enter]

END INPUT

OUTPUT:

Employees Name: Tim Cook

Employees Salary: $75,000.00

Employees SSN: 234567890

Employees Department: 12000

Employees City: Richmond

Employees State: VA

New Salary for employee: $105,000.00

--Number 6 (109 lines) This program goes through the database and retrieves employees based on their departments. It then displays their name and department manager number.

```
SET SERVEROUTPUT ON
DECLARE
CURSOR cur IS
select empfname, emplname, deptname, deptmgr
from employee, departments
where deptno = deptid
and empid = deptmgr
and deptno = '11000';
CURSOR cur2 IS
select empfname, emplname, deptname, deptmgr
from employee, departments
where deptno = deptid
and empid = deptmgr
and deptno = '12000';
CURSOR cur3 IS
select empfname, emplname, deptname, deptmgr
from employee, departments
where deptno = deptid
and empid = deptmgr
and deptno = '13000';
CURSOR cur4 IS
```

```
select empfname, emplname, deptname, deptmgr
from employee, departments
where deptno = deptid
and empid = deptmgr
and deptno = '15000';
CURSOR cur5 IS
select empfname, emplname, deptname, deptmgr
from employee, departments
where deptno = deptid
and empid = deptmgr
and deptno = '16000';

TYPE type_emp_dep IS RECORD
(firstname employee.empfname%type,
lastname employee.emplname%type,
dname departments.deptname%type,
dmanager departments.deptmgr%type);
rec_emp_dept    type_emp_dep;
BEGIN
OPEN cur;
rec_emp_dept.dname := '11000';
dbms_output.put_line('Department One');
dbms_output.put_line('');
LOOP
FETCH cur INTO rec_emp_dept;
EXIT WHEN cur%notfound;
dbms_output.put_line('First Name: '||rec_emp_dept.firstname);
dbms_output.put_line('Last Name: '||rec_emp_dept.lastname);
dbms_output.put_line('Department Manager: '||rec_emp_dept.dmanager);
```

```
END LOOP;
dbms_output.put_line('');
close cur;

OPEN cur2;
rec_emp_dept.dname := '12000';
dbms_output.put_line('Department Two');
dbms_output.put_line('');
LOOP
FETCH cur2 INTO rec_emp_dept;
EXIT WHEN cur2%notfound;
dbms_output.put_line('First Name: '||rec_emp_dept.firstname);
dbms_output.put_line('Last Name: '||rec_emp_dept.lastname);
dbms_output.put_line('Department Manager: '||rec_emp_dept.dmanager);
END LOOP;
dbms_output.put_line('');
close cur2;

OPEN cur3;
rec_emp_dept.dname := '13000';
dbms_output.put_line('Department Three');
dbms_output.put_line('');
LOOP
FETCH cur3 INTO rec_emp_dept;
EXIT WHEN cur3%notfound;
dbms_output.put_line('First Name: '||rec_emp_dept.firstname);
dbms_output.put_line('Last Name: '||rec_emp_dept.lastname);
dbms_output.put_line('Department Manager: '||rec_emp_dept.dmanager);
END LOOP;
```

```
dbms_output.put_line('');
close cur3;

OPEN cur4;
rec_emp_dept.dname := '15000';
dbms_output.put_line('Department Five');
dbms_output.put_line('');
LOOP
FETCH cur4 INTO rec_emp_dept;
EXIT WHEN cur4%notfound;
dbms_output.put_line('First Name: '||rec_emp_dept.firstname);
dbms_output.put_line('Last Name: '||rec_emp_dept.lastname);
dbms_output.put_line('Department Manager: '||rec_emp_dept.dmanager);
END LOOP;
dbms_output.put_line('');
close cur4;

OPEN cur5;
rec_emp_dept.dname := '16000';
dbms_output.put_line('Department Six');
dbms_output.put_line('');
LOOP
FETCH cur5 INTO rec_emp_dept;
EXIT WHEN cur5%notfound;
dbms_output.put_line('First Name: '||rec_emp_dept.firstname);
dbms_output.put_line('Last Name: '||rec_emp_dept.lastname);
dbms_output.put_line('Department Manager: '||rec_emp_dept.dmanager);
END LOOP;
dbms_output.put_line('');
```

```
close cur5;

end;
```

OUTPUT:

```
Department One


First Name: Satya

Last Name: Nadella

Department Manager: 12234


Department Two



Department Three



Department Five


First Name: Larry

Last Name: Page

Department Manager: 12345


Department Six
```

--Number 7 (90 lines) This program goes through the database and retrieves material name, material id, supplier company, and supplier city through cursors executed by the main program. It displays the information properly and formatted.

```
SET SERVEROUTPUT ON
```

```
DECLARE
CURSOR cur IS
select matid, matname, matsupply, supcompany, supcity
from materials, supplier
where materials.supid = supplier.supid
and materials.matid = '99990';
CURSOR cur2 IS
select matid, matname, matsupply, supcompany, supcity
from materials, supplier
where materials.supid = supplier.supid
and materials.matid = '88880';
CURSOR cur3 IS
select matid, matname, matsupply, supcompany, supcity
from materials, supplier
where materials.supid = supplier.supid
and materials.matid = '77770';
CURSOR cur4 IS
select matid, matname, matsupply, supcompany, supcity
from materials, supplier
where materials.supid = supplier.supid
and materials.matid = '66660';

TYPE type_mat_sup IS RECORD
(material materials.matid%type,
materialname materials.matname%type,
materialsupplier materials.matsupply%type,
supcomp supplier.supcompany%type,
suppliercity supplier.supcity%type);
rec_mat_sup    type_mat_sup;
```

```
BEGIN
OPEN cur;
rec_mat_sup.materialname := '99990';
dbms_output.put_line('Material One');
dbms_output.put_line('');
LOOP
FETCH cur INTO rec_mat_sup;
EXIT WHEN cur%notfound;
dbms_output.put_line('MaterialID: '||rec_mat_sup.material);
dbms_output.put_line('Material Supplier: '||rec_mat_sup.materialsupplier);
dbms_output.put_line('Supplier Company: '||rec_mat_sup.supcomp);
dbms_output.put_line('Supplier City: '||rec_mat_sup.suppliercity);
END LOOP;
dbms_output.put_line('');
close cur;

OPEN cur2;
rec_mat_sup.materialname := '88880';
dbms_output.put_line('Material Two');
dbms_output.put_line('');
LOOP
FETCH cur2 INTO rec_mat_sup;
EXIT WHEN cur2%notfound;
dbms_output.put_line('MaterialID: '||rec_mat_sup.material);
dbms_output.put_line('Material Supplier: '||rec_mat_sup.materialsupplier);
dbms_output.put_line('Supplier Company: '||rec_mat_sup.supcomp);
dbms_output.put_line('Supplier City: '||rec_mat_sup.suppliercity);
END LOOP;
dbms_output.put_line('');
```

```
close cur2;

OPEN cur3;
rec_mat_sup.materialname := '77770';
dbms_output.put_line('Material Three');
dbms_output.put_line('');
LOOP
FETCH cur3 INTO rec_mat_sup;
EXIT WHEN cur3%notfound;
dbms_output.put_line('MaterialID: '||rec_mat_sup.material);
dbms_output.put_line('Material Supplier: '||rec_mat_sup.materialsupplier);
dbms_output.put_line('Supplier Company: '||rec_mat_sup.supcomp);
dbms_output.put_line('Supplier City: '||rec_mat_sup.suppliercity);
END LOOP;
dbms_output.put_line('');
close cur3;

OPEN cur4;
rec_mat_sup.materialname := '66660';
dbms_output.put_line('Material Four');
dbms_output.put_line('');
LOOP
FETCH cur4 INTO rec_mat_sup;
EXIT WHEN cur4%notfound;
dbms_output.put_line('MaterialID: '||rec_mat_sup.material);
dbms_output.put_line('Material Supplier: '||rec_mat_sup.materialsupplier);
dbms_output.put_line('Supplier Company: '||rec_mat_sup.supcomp);
dbms_output.put_line('Supplier City: '||rec_mat_sup.suppliercity);
END LOOP;
```

```
dbms_output.put_line('');
close cur4;
end;
```

OUTPUT:

Material One

MaterialID: 99990

Material Supplier: 100

Supplier Company: Supplier Two

Supplier City: Nashville

Material Two

MaterialID: 88880

Material Supplier: 250

Supplier Company: Supplier Four

Supplier City: Washington D.C.

Material Three

MaterialID: 77770

Material Supplier: 50

Supplier Company: Supplier One

Supplier City: Houston

Material Four

MaterialID: 66660

Material Supplier: 500

Supplier Company: Supplier Three

Supplier City: Miami

--Number 8 (22 lines) Create a PL/SQL block that will accept the user's input for a beginning and ending date. The PL/SQL block should then return all the orders that have been placed during that period and the amount.

RUN THE ALTER SESSION ALONE FIRST

```
ALTER session set NLS_DATE_FORMAT='DD/MM/YYYY'
SET SERVEROUTPUT ON;
DECLARE
beginningdate orders.orderdate%type :='&Enter_Beginning_Date';
endingdate orders.orderdate%type :='&Enter_Ending_Date';
CURSOR cur_order
IS
Select orders.orderid, orders.orderdate, orders.ordertotalamt
from orders
where orders.orderdate >=beginningdate
and orders.orderdate <=endingdate;
totalorder NUMBER := 0;
BEGIN
dbms_output.put_line('');
dbms_output.put_line('-------Order Summary------');
dbms_output.put_line('Orders placed from '||' '||beginningdate||' to'||' '||endingdate);
For rec_order IN cur_order
LOOP
dbms_output.put_line('Orderid: '||rec_order.orderid||' '||'Ordertotal: '||rec_order.ordertotalamt);
totalorder := totalorder + rec_order.ordertotalamt;
END LOOP;
```

```
dbms_output.put_line('Total Cost: '||TO_CHAR(totalorder));
end;
```

INPUT:

01/02/2017[enter]31/12/2017[enter]

END INPUT

OUTPUT:

-------Order Summary------

Orders placed from 01/02/2017 to 31/12/2017

Orderid: 00002 Ordertotal: 250000

Orderid: 00003 Ordertotal: 62500

Orderid: 00004 Ordertotal: 50000

Orderid: 00005 Ordertotal: 1500

Total Cost: 364000

--number 9(105 lines) user inputs a product id and it retrieves the sales information associated with the product. Also, an error will occur if the user inputs an invalid product id.

```
Create table orderdetails2 as select * from orderdetails;
Create table product3 as select * from product;
--drop package pg_misc; --drops the package uncomment after running
CREATE OR REPLACE PACKAGE pg_misc
IS
function find_price(p_essn product.prodid%type) return number;
function find_name(p_essn product.prodid%type) return product.prodname%type;
function find_color(p_essn product.prodid%type) return product.prodcolor%type;
procedure sumqty(p_essn in orderdetails.prodid%type, p_numor out number);
procedure p_og(p_og out number);
procedure p_ot(p_ot in orderdetails.prodid%type,p_og out number);
end pg_misc;
/
```

```
CREATE OR REPLACE PACKAGE body pg_misc AS
FUNCTION find_price(p_essn product.prodid%type)
RETURN number IS
ret_price number;
BEGIN
select prodprice into ret_price
from product
where p_essn = prodid;
return ret_price;
end find_price;

FUNCTION find_name(p_essn product.prodid%type)
RETURN product.prodname%type IS
ret_price product.prodname%type;
BEGIN
select prodname into ret_price
from product
where p_essn = prodid;
return ret_price;
end find_name;

 FUNCTION find_color(p_essn product.prodid%type)
RETURN product.prodcolor%type IS
ret_price product.prodcolor%type;
BEGIN
select prodcolor into ret_price
from product
where p_essn = prodid;
return ret_price;
```

```
end find_color;

procedure sumqty(p_essn orderdetails.prodid%type,p_numor out number) IS
BEGIN
select sum(qtyorderperproduct)
into p_numor
from orderdetails
where prodid = p_essn;
end sumqty;

procedure p_og(p_og out number) IS
BEGIN
select sum(ordertotalamt)
into p_og
from orders o
where orderid = p_og;
end p_og;

procedure p_ot
(p_ot IN orderdetails.prodid%type,
p_og out number)
IS
BEGIN
select count(orderid)
into p_og
from orderdetails od
where od.prodid = p_ot;
end p_ot;
end;
```

```
/
set serveroutput on
declare
product_prodid_fk EXCEPTION;
PRAGMA Exception_Init(product_prodid_fk,+0100);
pno product.prodid%type;
price product.prodprice%type;
name product.prodname%type;
color product.prodcolor%type;
total number;
qty number;
bl number;
totorders number;
begin
pno:='&productid';
name:=pg_misc.find_name(pno);
color:=pg_misc.find_color(pno);
price:=pg_misc.find_price(pno);
pg_misc.sumqty(pno,qty);
total:=qty*price; --gets the grand total for
bl:=price*0.6;
pg_misc.p_ot(pno,totorders);
dbms_output.put_line(' ');
dbms_output.put_line(' ');
dbms_output.put_line(' ');
dbms_output.put_line(' ');

dbms_output.put_line('The product the user selected was '||pno||' which is named '||trim(name)||'
and is color '||color||', which we buy at '||to_char(bl,'$999,999.00')||' from our supplier, and we sell
```

```
it at '||trim(to_char(price,'$999,999.00'))||'.');
dbms_output.put_line('We have sold '||qty||' of that product at the price of
'||trim(to_char(price,'$999,999.00'))||' in '||totorders||' orders.');
Exception when product_prodid_fk then dbms_output.put_line(''); dbms_output.put_line('');
dbms_output.put_line('entered product not found in orderdetails table');
end;
```

INPUT: 34567

END INPUT

OUTPUT:

```
The product the user selected was 34567 which is named Product Two
and is color Blue, which we buy at       $5.99 from our supplier, and we sell
it at $9.99.
We have sold 250 of that product at the price of
$9.99 in 1 orders.
```

END OUTPUT

--number 10(75 lines) this query gets the product information when prompted to enter a productid and displays all product information and supplier information that pertains to the product.

```
CREATE OR REPLACE PROCEDURE p_product
 (p_product IN product.prodid%type,
 p_id out product.prodid%type,
 p_name out product.prodname%type,
 p_price out product.prodprice%type,
 p_color out product.prodcolor%type,
 p_sup out product.supid%type
 )
 IS
BEGIN
```

```
select prodid, prodname, prodprice, prodcolor, supid
into p_id, p_name, p_price, p_color, p_sup
from product
where prodid = p_product;
end;
/
 CREATE OR REPLACE PROCEDURE p_si
 (p_si IN supplier.supid%type,
 p_supc out supplier.supcompany%type,
 p_supadd out supplier.supaddress%type,
 p_city out supplier.supcity%type,
 p_phone out supplier.supphone%type)
 IS
BEGIN
select supcompany, supaddress, supcity, supphone
into p_supc, p_supadd, p_city, p_phone
from supplier
where supid = p_si;
end;
/

set serveroutput on
declare
pid_fk EXCEPTION;
PRAGMA Exception_Init(pid_fk,+0100);
pid product.prodid%type;
p_id product.prodid%type;
p_name product.prodname%type;
p_price product.prodprice%type;
```

```
p_color product.prodcolor%type;
p_sup product.supid%type;
p_supc supplier.supcompany%type;
p_supadd supplier.supcompany%type;
p_city supplier.supcity%type;
p_phone supplier.supphone%type;
bl number;
tot number;
tot2 number;
tot3 number;
dif number;
BEGIN
pid:='&product_id';
p_product(pid,p_id, p_name,p_price,p_color, p_sup);
p_si(p_sup, p_supc, p_supadd, p_city, p_phone);
bl:=(p_price*0.6);
dif:=p_price-bl;
dbms_output.put_line('Product ID: '||p_id);
dbms_output.put_line('Product Name: '||p_name);
dbms_output.put_line('Product Price: '||p_price);
dbms_output.put_line('Product Color: '||p_color);
dbms_output.put_line('Product Supplier ID: '||p_sup);
dbms_output.put_line('Supplier ID: '||p_sup);
dbms_output.put_line('Supplier Company: '||p_supc);
dbms_output.put_line('Supplier Address: '||p_supadd);
dbms_output.put_line('Supplier City: '||p_city);
dbms_output.put_line('Supplier Phone: '||p_phone);
dbms_output.put_line('Current profit we gain per unit: '||to_char(dif,'$999,999.00'));
dbms_output.put_line('Current price we buy the product for, per unit: '||to_char(bl,'$999,999.00'));
```

```
dbms_output.put_line('Current profit we gain per unit: '||to_char(dif,'$999,999.00'));
dbms_output.put_line(' ');
dbms_output.put_line(' ');
Exception when pid_fk then
dbms_output.put_line('');
dbms_output.put_line('');
dbms_output.put_line('entered product id not found in product table table.');
end;
```

INPUT: 34567, or 45678, or 56789

OUTPUT:

Product ID: 34567

Product Name: Product Two

Product Price: 9.99

Product Color: Blue

Product Supplier ID: 66666

Supplier ID: 66666

Supplier Company: Supplier Four

Supplier Address: 234 Shadow Lane

Supplier City: Washington D.C.

Supplier Phone: 1233334444

Current profit we gain per unit: $4.00

Current price we buy the product for, per unit: $5.99

Current profit we gain per unit: $4.00

--number 11(224 lines) Goes through 6 departments and gives a raise of 1%, 50%, 3%, 13%, 8%, and 3.5% respectively. This is what a salary increase would mean for each department and how it affects the overall cost using cursors. This information is then reset at the end of the query.

```
SET SERVEROUTPUT ON;
DECLARE
Cursor cur_emp_department IS
Select EmpFname, EmpLname, EmpSalary, DeptName
From employee e, departments d
WHERE deptno = DeptID
AND e.deptno = '11000';

Cursor cur IS
Select EmpFname, EmpLname, EmpSalary, DeptName
From employee e, departments d
WHERE deptno = DeptID
AND e.deptno = '12000';

Cursor cur_emp IS
Select EmpFname, EmpLname, EmpSalary, DeptName
From employee e, departments d
WHERE deptno = DeptID
AND e.deptno = '15000';

Cursor cur_emp_three IS
Select EmpFname, EmpLname, EmpSalary, DeptName
From employee e, departments d
WHERE deptno = DeptID
AND e.deptno = '13000';

Cursor cur_emp_four IS
Select EmpFname, EmpLname, EmpSalary, DeptName
From employee e, departments d
```

```
WHERE deptno = DeptID
AND e.deptno = '14000';

Cursor cur_emp_six IS
Select EmpFname, EmpLname, EmpSalary, DeptName
From employee e, departments d
WHERE deptno = DeptID
AND e.deptno = '16000';

TYPE type_emp_dept IS RECORD (
firstname employee.EmpFname%type,
lastname employee.EmpLname%type,
empsalary employee.EmpSalary%type,
departmentname departments.deptname%type);

rec_emp_dept              type_emp_dept;
totalsalary NUMBER := 0;
totaloriginalsalary NUMBER := 0;
grandtotaloriginal NUMBER := 0;
grandtotalnew NUMBER := 0;
math NUMBER := 0;
BEGIN

Open cur_emp_department;
Rec_emp_dept.departmentname := 'Department One';
dbms_output.put_line('Department One***** salary increased by 1% ***** ');
 dbms_output.put_line(' ');
 dbms_output.put_line('    original salary         new Salary ');
 dbms_output.put_line(' ');
```

```
 LOOP


 FETCH cur_emp_department INTO rec_emp_dept;

 EXIT WHEN cur_emp_department%notfound;

totaloriginalsalary := totaloriginalsalary + to_number(rec_emp_dept.empsalary);

 dbms_output.put_line('  ' || trim(rec_emp_dept.firstname) || '  ' || rec_emp_dept.lastname || ' '

 || TO_CHAR(rec_emp_dept.empsalary,'$999,999.99') || ' ' || TO_CHAR(rec_emp_dept.empsalary *
1.01, '$999,999.00') );

totalsalary := totalsalary + (to_number(rec_emp_dept.EmpSalary) * 1.01);

END LOOP;

Grandtotaloriginal := grandtotaloriginal + totaloriginalsalary;

Grandtotalnew := grandtotalnew + totaloriginalsalary;

Dbms_output.put_line('');

dbms_output.put_line(' Total Original Salary for Department One ==== ' ||
TO_CHAR(totaloriginalsalary,'$999,999.00') );

 dbms_output.put_line(' Total New Salary for Department One==== ' ||
TO_CHAR(totalsalary,'$999,999.00'));

 totalsalary := 0;

 totaloriginalsalary := 0;

dbms_output.put_line('================================================================');


CLOSE cur_emp_department;

OPEN cur;

 rec_emp_dept.departmentName := 'Department Two';

 dbms_output.put_line('Department Two**** salary increased by 50%**** ');

 dbms_output.put_line(' ');

 dbms_output.put_line('     original salary         new Salary ');

 dbms_output.put_line(' ');

 LOOP
```

```
 FETCH cur INTO rec_emp_dept;
EXIT WHEN cur%notfound;
totaloriginalsalary := totaloriginalsalary + to_number(rec_emp_dept.empsalary);
 dbms_output.put_line(' ' || trim(rec_emp_dept.firstname) || ' ' || rec_emp_dept.lastname || ' '
 || TO_CHAR(rec_emp_dept.empsalary,'$999,999.99') || ' ' || TO_CHAR(
 rec_emp_dept.empsalary * 1.50,
 '$999,999.00' ) );
totalsalary := totalsalary + ( to_number(rec_emp_dept.empsalary) * 1.5 );
 END LOOP;

grandtotaloriginal := grandtotaloriginal + totaloriginalsalary;
 grandtotalnew := grandtotalnew + totalsalary;
dbms_output.put_line('');
 dbms_output.put_line(' Total Original Salary for Department Two==== '
 || TO_CHAR(totaloriginalsalary,'$999,999.00') );
 dbms_output.put_line(' Total New Salary for Department Two==== '
 || TO_CHAR(totalsalary,'$999,999.00') );
 totalsalary := 0;
 totaloriginalsalary := 0;
dbms_output.put_line('================================================================');
CLOSE cur;

 OPEN cur_emp;
 rec_emp_dept.departmentname := 'Department Five';
 dbms_output.put_line('Department Five***** salary increased by 3%*****');
 dbms_output.put_line(' ');
 dbms_output.put_line('    original salary      new Salary ');
 dbms_output.put_line(' ');
 LOOP
```

```
 FETCH cur_emp INTO rec_emp_dept;

 EXIT WHEN cur_emp%notfound;

 totaloriginalsalary := totaloriginalsalary + to_number(rec_emp_dept.empsalary);

 dbms_output.put_line(' '|| trim(rec_emp_dept.firstname) || ' ' || rec_emp_dept.lastname || ' ' ||
TO_CHAR(rec_emp_dept.empsalary,'$999,999.99') || ' ' || TO_CHAR( rec_emp_dept.empsalary * 1.03,
'$999,999.00' ) );

totalsalary := totalsalary + (to_number(rec_emp_dept.empsalary) * 1.03);

END LOOP;


grandtotaloriginal := grandtotaloriginal + totaloriginalsalary;

 grandtotalnew := grandtotalnew + totalsalary;

 dbms_output.put_line('');

 dbms_output.put_line(' Total Original Salary for Department Five==== ' ||
TO_CHAR(totaloriginalsalary,'$999,999.00') );

 dbms_output.put_line(' Total New Salary for Department Five==== ' ||
TO_CHAR(totalsalary,'$999,999.00'));

dbms_output.put_line('');


dbms_output.put_line('=================================================================
');

CLOSE cur_emp;


OPEN cur_emp_three;

 rec_emp_dept.departmentName := 'Department Three';

 dbms_output.put_line('Department Three**** salary increased by 13%**** ');

 dbms_output.put_line(' ');

 dbms_output.put_line('     original salary         new Salary ');

 dbms_output.put_line(' ');

 LOOP

 FETCH cur_emp_three INTO rec_emp_dept;

EXIT WHEN cur_emp_three%notfound;
```

```
totaloriginalsalary := totaloriginalsalary + to_number(rec_emp_dept.empsalary);
 dbms_output.put_line(' ' || trim(rec_emp_dept.firstname) || ' ' || rec_emp_dept.lastname || ' '
 || TO_CHAR(rec_emp_dept.empsalary,'$999,999.99') || ' ' || TO_CHAR(
 rec_emp_dept.empsalary * 1.13,
 '$999,999.00' ) );
totalsalary := totalsalary + ( to_number(rec_emp_dept.empsalary) * 1.13 );
 END LOOP;

grandtotaloriginal := grandtotaloriginal + totaloriginalsalary;
 grandtotalnew := grandtotalnew + totalsalary;
dbms_output.put_line('');
 dbms_output.put_line(' Total Original Salary for Department Three==== '
 || TO_CHAR(totaloriginalsalary,'$999,999.00') );
 dbms_output.put_line(' Total New Salary for Department Three==== '
 || TO_CHAR(totalsalary,'$999,999.00') );
 totalsalary := 0;
 totaloriginalsalary := 0;
dbms_output.put_line('================================================================');
CLOSE cur_emp_three;

OPEN cur_emp_four;
 rec_emp_dept.departmentName := 'Department Four';
 dbms_output.put_line('Department Four**** salary increased by 1.08%**** ');
 dbms_output.put_line(' ');
 dbms_output.put_line('     original salary         new Salary ');
 dbms_output.put_line(' ');
 LOOP
 FETCH cur_emp_four INTO rec_emp_dept;
EXIT WHEN cur_emp_four%notfound;
```

```
totaloriginalsalary := totaloriginalsalary + to_number(rec_emp_dept.empsalary);
 dbms_output.put_line(' ' || trim(rec_emp_dept.firstname) || ' ' || rec_emp_dept.lastname || ' '
 || TO_CHAR(rec_emp_dept.empsalary,'$999,999.99') || ' ' || TO_CHAR(
 rec_emp_dept.empsalary * 1.08,
 '$999,999.00' ) );
totalsalary := totalsalary + ( to_number(rec_emp_dept.empsalary) * 1.08);
 END LOOP;

grandtotaloriginal := grandtotaloriginal + totaloriginalsalary;
 grandtotalnew := grandtotalnew + totalsalary;
dbms_output.put_line('');
 dbms_output.put_line(' Total Original Salary for Department Four==== '
 || TO_CHAR(totaloriginalsalary,'$999,999.00') );
 dbms_output.put_line(' Total New Salary for Department Four==== '
 || TO_CHAR(totalsalary,'$999,999.00') );
 totalsalary := 0;
 totaloriginalsalary := 0;
dbms_output.put_line('================================================================');
CLOSE cur_emp_four;

OPEN cur_emp_six;
 rec_emp_dept.departmentName := 'Department Six';
 dbms_output.put_line('Department Six**** salary increased by 3.5%**** ');
 dbms_output.put_line(' ');
 dbms_output.put_line('     original salary         new Salary ');
 dbms_output.put_line(' ');
 LOOP
 FETCH cur_emp_six INTO rec_emp_dept;
EXIT WHEN cur_emp_six%notfound;
```

```
totaloriginalsalary := totaloriginalsalary + to_number(rec_emp_dept.empsalary);
 dbms_output.put_line(' ' || trim(rec_emp_dept.firstname) || ' ' || rec_emp_dept.lastname || ' '
 || TO_CHAR(rec_emp_dept.empsalary,'$999,999.99') || ' ' || TO_CHAR(
 rec_emp_dept.empsalary * 1.035,
 '$999,999.00' ) );
totalsalary := totalsalary + ( to_number(rec_emp_dept.empsalary) * 1.035 );
 END LOOP;

grandtotaloriginal := grandtotaloriginal + totaloriginalsalary;
 grandtotalnew := grandtotalnew + totalsalary;
dbms_output.put_line('');
 dbms_output.put_line(' Total Original Salary for Department Six==== '
 || TO_CHAR(totaloriginalsalary,'$999,999.00') );
 dbms_output.put_line(' Total New Salary for Department Six==== '
 || TO_CHAR(totalsalary,'$999,999.00') );
 totalsalary := 0;
 totaloriginalsalary := 0;
dbms_output.put_line('================================================================');
CLOSE cur_emp_six;

 dbms_output.put_line(' Grand Total Original Salaries for the departments ==== '||
TO_CHAR(grandtotaloriginal,'$999,999,999.00') );
 dbms_output.put_line(' Grand Total of salaries after raises ==== '||
TO_CHAR(grandtotalnew,'$999,999,999.00') );
 totalsalary := 0;
 totaloriginalsalary := 0;
 dbms_output.put_line('');
math:=grandtotalnew-grandtotaloriginal;
 dbms_output.put_line('The difference is: '||TO_CHAR(math,'$999,999,999.00'));
```

```
end;
```

OUTPUT:

```
Department One***** salary increased by 1% *****

    original salary         new Salary

 Satya  Nadella   $90,000.00   $90,900.00

 Total Original Salary for Department One ====   $90,000.00
 Total New Salary for Department One====   $90,900.00
==========================================================
Department Two**** salary increased by 50%****

    original salary        new Salary

 Tim Cook   $75,000.00  $112,500.00

 Total Original Salary for Department Two====   $75,000.00
 Total New Salary for Department Two====  $112,500.00
==========================================================
Department Five***** salary increased by 3%*****

    original salary      new Salary

 Larry Page   $30,000.00   $30,900.00

 Total Original Salary for Department Five====   $30,000.00
 Total New Salary for Department Five====   $30,900.00
```

==========

Department Three**** salary increased by 13%****

original salary new Salary

Jack Dorsey $50,000.00 $56,500.00

Total Original Salary for Department Three==== $80,000.00

Total New Salary for Department Three==== $87,400.00

==========

Department Four**** salary increased by 1.08%****

original salary new Salary

Total Original Salary for Department Four==== $.00

Total New Salary for Department Four==== $.00

==========

Department Six**** salary increased by 3.5%****

original salary new Salary

Travis Kalanick $45,000.00 $46,575.00

Total Original Salary for Department Six==== $45,000.00

Total New Salary for Department Six==== $46,575.00

==========

Grand Total Original Salaries for the departments ==== $320,000.00

Grand Total of salaries after raises ==== $367,375.00

The difference is: $47,375.00