

REM: Assignment 6 – PL/SQL STORED PROCEDURES & STORED FUNCTIONS

REM: LAKSHMI PRIYA B

REM: 185001083

REM: *****

SET SERVEROUTPUT ON;

SET ECHO ON;

DROP TABLE customer;

DROP TABLE pizza;

DROP TABLE orders;

DROP TABLE order_list;

```
CREATE TABLE customer(  
    cust_id varchar2(5),  
    cust_name varchar2(10),  
    address varchar2(30),  
    phone number(10),  
    CONSTRAINT cust_pk PRIMARY KEY(cust_id)  
);
```

```
CREATE TABLE pizza(  
    pizza_id varchar2(5),  
    pizza_type varchar2(10),  
    unit_price number(5),  
    CONSTRAINT pizza_pk PRIMARY KEY(pizza_id)  
);
```

```
CREATE TABLE orders(  
    order_no varchar2(5),  
    cust_id varchar2(5),  
    order_date date,  
    delv_date date,  
    CONSTRAINT order_pk PRIMARY KEY(order_no)  
);
```

```
CREATE TABLE order_list(  
    order_no varchar2(5),  
    pizza_id varchar2(5),  
    qty number(5),  
    CONSTRAINT list_pk PRIMARY KEY(order_no, pizza_id)  
);
```

@d:/Pizza_DB.sql

```
ALTER TABLE orders  
ADD (total_amt NUMBER, discount NUMBER, bill_amt NUMBER);
```

```

SELECT * FROM customer;
SELECT * FROM pizza;
SELECT * FROM orders;
SELECT * FROM order_list;

```

REM: 1. Write a stored function to display the total number of pizza's
 REM: ordered by the given order number.

```

CREATE OR REPLACE FUNCTION numPizza
(order_num IN order_list.order_no%type)
RETURN NUMBER
IS
totNum NUMBER;

BEGIN
SELECT sum(qty) INTO totNum FROM order_list
WHERE order_no=order_num
GROUP BY order_no;
RETURN totNum;
EXCEPTION
  WHEN no_data_found THEN
    dbms_output.put_line('Invalid order number!!');
    RETURN -1;
  WHEN others THEN
    dbms_output.put_line('ERROR');
    RETURN -1;
END;
/

DECLARE
order_no order_list.order_no%type;
totNum NUMBER;

BEGIN
order_no:='&order_no';
totNum:=numPizza(order_no);
IF totNum!=-1 THEN
  dbms_output.put('*****');
  dbms_output.put_line('*****');
  dbms_output.put_line('Total number of pizzas order by '||order_no||
    ' is '||totNum);
  dbms_output.put('*****');
  dbms_output.put_line('*****');
END IF;
END;
/

```

REM: 2. Write a PL / SQL block to calculate the total amount, discount
 REM: and billable amount (Amount to Be Paid) as given below:

REM: For Total Amount > 2000 and
 REM: Total Amount < 5000: Discount = 5%
 REM: For Total Amount > 5000 and Total Amount < 10000: Discount = 10%
 REM: For Total Amount > 10000: Discount = 20%
 REM: Calculate the Billable Amount (After the Discount) and Update the
 REM: same orders table.
 REM: Bill Amount = Total - Discount.

```

CREATE OR REPLACE PROCEDURE calcAmt
IS
  cur sys_refcursor;
  orderno orders.order_no%type;
  disc orders.discount%type;
  totamt orders.total_amt%type;

BEGIN
  UPDATE orders o
  SET total_amt=( SELECT SUM(qty*unit_price)
                  FROM pizza p, order_list ol
                  WHERE o.order_no=ol.order_no AND ol.pizza_id=p.pizza_id);

  OPEN cur FOR SELECT order_no, total_amt FROM orders;

  LOOP
    FETCH cur INTO orderno, totamt;
    EXIT WHEN cur%notfound;

    IF totamt > 2000 AND totamt <=5000 THEN
      disc:=0.05;
    ELSIF totamt > 5000 AND totamt <=10000 THEN
      disc:=0.1;
    ELSIF totamt > 10000 THEN
      disc:=0.2;
    ELSE
      disc:=0;
    END IF;

    UPDATE orders
      SET total_amt=totamt, discount=disc, bill_amt=totamt-disc*totamt
      WHERE order_no=orderno;
  END LOOP;
  CLOSE cur;
END;
/

BEGIN
  calcAmt();
END;
/

SELECT * FROM orders;
  
```

REM: 3. For the given order number, write a PL / SQL block to the print
REM: order shown below: Hint: Use the PL / SQL blocks created in 1 and 2.

```
CREATE OR REPLACE PROCEDURE printOrder
(order_num IN order_list.order_no%type)
IS
cur sys_refcursor;
totQty NUMBER;
cname customer.cust_name%type;
phoneNum customer.phone%type;
orderDate DATE;
sno NUMBER;
ptype pizza.pizza_type%type;
qty order_list.qty%type;
price pizza.unit_price%type;
totamt orders.total_amt%type;
disc orders.discount%type;
billamt orderS.bill_amt%type;

BEGIN
totQty:=numPizza(order_num);
IF totQty!=-1 THEN
    calcAmt();

    SELECT cust_name, order_date, phone
    INTO cname, orderDate, phoneNum
    FROM customer, orders
    WHERE customer.cust_id=orders.cust_id AND order_no=order_num;

    dbms_output.put('*****');
    dbms_output.put_line('*****');
    dbms_output.put_line('Order Number: '||rpad(order_num, 15)||
                          'Customer Name: '||lpad(cname, 10));
    dbms_output.put_line('Order Date : '||rpad(orderDate, 15)||
                          'Phone      : '||lpad(phoneNum, 10));
    dbms_output.put('*****');
    dbms_output.put_line('*****');
    OPEN cur FOR SELECT pizza_type, qty, unit_price
                  FROM order_list ol, orders o, pizza p
                  WHERE ol.order_no=order_num AND
                        ol.order_no=o.order_no AND
                        ol.pizza_id=p.pizza_id;
    SELECT total_amt, discount, bill_amt
    INTO totamt, disc, billamt
    FROM orders
    WHERE order_no=order_num;

    sno:=1;
    dbms_output.put_line(lpad('SNo',5)||lpad('Pizza Type',20)||
                        lpad('Qty',5)||lpad('Price',8)||lpad('Amount',10));
```

```

LOOP
    FETCH cur INTO ptype, qty, price;
    EXIT WHEN cur%notfound;
    dbms_output.put_line(lpad(sno,5) || lpad(ptype,20) || lpad(qty,5) ||
                        lpad(price,8) || lpad(qty*price,10));
    sno:=sno+1;
END LOOP;
CLOSE cur;
dbms_output.put('-----');
dbms_output.put_line('-----');
dbms_output.put_line(rpad('      Total = ',21) ||
                    lpad('      ',10) || lpad(totqty,5) ||
                    lpad(' ',8) || lpad(totamt,10));
dbms_output.put('-----');
dbms_output.put_line('-----');
dbms_output.put_line(rpad('Total Amount',20) || 'Rs. ' || totamt);
dbms_output.put_line('Discount(' || TO_CHAR(disc*100) || rpad('%',10) ||
                    'Rs. ' || TO_CHAR(disc*totamt));
dbms_output.put_line('-----');
dbms_output.put_line(rpad('Amount to be paid',20) || 'Rs. ' || billamt);
dbms_output.put_line('-----');
dbms_output.put_line('Great Offers! Discount upto 25% on DIWALI...');
dbms_output.put('*****');
dbms_output.put_line('*****');

END IF;
END;
/

DECLARE
ordernum order_list.order_no%type;

BEGIN
ordernum:='&order_no';
printOrder(ordernum);
END;
/

REM: *****
REM: LAKSHMI PRIYA B
REM: 185001083
REM: *****

```