

# Sheet No : 5

## Assignment No : 8

Write a pl/sql block for the following:

1. Insert data into a table containing two attributes namely radius & circumference of circles. You may get different values of radius either from keyboard or you may generate different values.

Creation of table:

```
create table circle (  
    radius number,  
    circumference number  
);
```

Pl/sql block to insert values into table circle:

```
declare  
    z number;  
begin  
    for i in 1..50 loop  
        z := 100*dbms_random.value();  
        insert into circle values(z,2*3.14*z);  
    end loop;  
end;  
/  
select * from circle  
/
```

Result:

RADIUS	CIRCUMFERENCE
84.92526547786031	533.3306672009627
6.178498280948241	38.80096920435495
28.76250875148666	180.6285549593362
28.053322248377246	176.1748637198091
88.37016026985276	554.9646064946754
80.96225446314334	508.4429580285402
61.88736256412177	388.6526369026847
86.93689527423699	545.9637023222083
46.64389495425946	292.9236603127494
49.75648220948631	312.470708275574
10.25186506810677	64.38171262771053
19.545668166409524	122.74679608505181
9.771098186154267	61.362496609048804
14.032215540479712	88.12231359421259
32.78998624275296	205.9211136044886
8.008736916865145	50.29486783791311
26.393835762720087	165.75328858988215
73.70583747611154	462.87265934998044

2. **Update the balance of each customer from a cust\_acct table showing withdrawal of Rs.1000/- as service charge provided that the customer balance shows at least Rs.1000/-.**

**Creation of table:**

```
create table cust_acct (
  cust_id number primary key,
  balance number
);
```

**Snapshot of the table:**

CUST_ID	BALANCE
93877	733.63
23366	1307.2
10348	4948.58
66826	459.14
51615	2969.2
60873	2542.32
76649	4376
72114	4786.04
29833	2905.87
33613	2782.9
41617	747.45
58444	934.9
30138	1127.51
58067	2646.64
55774	3443.73

**PL/sql block for updation:**

```
declare
begin
    update cust_acct
    set balance = balance - 1000
    where balance >= 1000;
    commit;
    dbms_output.put_line('balances updated successfully.');
```

exception

```
    when others then
        dbms_output.put_line('an error occurred while updating balances.');
```

```
end;
/
```

**Table after updation:**

CUST_ID	BALANCE
93877	733.63
23366	307.2
10348	3948.58
66826	459.14
51615	1969.2
60873	1542.32
76649	3376
72114	3786.04
29833	1905.87
33613	1782.9
41617	747.45
58444	934.9
30138	127.51
58067	1646.64
55774	2443.73

**3. Update the salary of each employee from EMP table by 15% using cursor.**

**Creation of table EMP:**

```
create table emp (  
    emp_id number primary key,  
    salary number  
);
```

**Snapshot of the table:**

EMP_ID	SALARY
9458	74918.91
1254	71931.86
2406	64507.57
3918	30022.05
6397	68882.66
7290	47920.93
9349	63927.31
5354	32032.94
8942	49375.83
6545	65496.67
4643	39507.21
2851	32599.6
2786	37826.34
8230	70569.31
5452	58119.86

**PL/sql block for updation:**

```

declare
    cursor emp_cursor is
        select emp_id, salary
        from emp;
    v_emp_id emp.emp_id%type;
    v_salary emp.salary%type;
begin
    for emp_rec in emp_cursor loop
        v_emp_id := emp_rec.emp_id;
        v_salary := emp_rec.salary * 1.15;
        update emp
        set salary = v_salary
        where emp_id = v_emp_id;
    end loop;

```

```

commit;
dbms_output.put_line('salaries updated successfully.');
```

exception

```

when others then
    dbms_output.put_line('an error occurred while updating salaries.');
```

end;

/

**Table after updation:**

EMP_ID	SALARY
9458	86156.7465
1254	82721.639
2406	74183.7055
3918	34525.3575
6397	79215.059
7290	55109.0695
9349	73516.4065
5354	36837.881
8942	56782.2045
6545	75321.1705
4643	45433.2915
2851	37489.54
2786	43500.291
8230	81154.7065
5452	66837.839

- Update the balance in the ITEM\_MSTR table each time a transaction takes place in the ITEM\_TR table. If this item id is already present in the ITEM\_MSTR table an update is performed to decrease the balance by the quantity specified in the ITEM\_TR table. If the item id is not present in the ITEM\_MSTR table, the tuple is to be inserted.**

**Creation of table ITEM\_MSTR:**

```
create table item_mstr (
  item_id number primary key,
  balance number
);
```

**Snapshot of table ITEM\_MSTR:**

ITEM_ID	BALANCE
3568	7244.35
9868	9587.62
3310	8612.64
5977	3468.11
6670	829.53
9109	8132.7
4662	2543.46
1336	3785.48
1798	2019.01
4559	3785.38

**Creation of table ITEM\_TR:**

```
create table item_tr (
  trans_id number primary key,
  item_id number,
  quantity number
);
```

**Snapshot of table ITEM\_TR:**

TRANS_ID	ITEM_ID	QUANTITY
110001	3568	20
110010	6670	10
110011	4662	12
110100	1798	15
110101	4559	8

### **Pl/sql block to update tables:**

```
declare
  v_item_id number;
  v_quantity number;
  v_balance number;
begin
  for trans_rec in (select item_id, quantity from item_tr) loop
    v_item_id := trans_rec.item_id;
    v_quantity := trans_rec.quantity;

    -- check if the item_id exists in item_mstr.
    select balance into v_balance
    from item_mstr
    where item_id = v_item_id;

    if v_balance is not null then
      -- item exists, update the balance.
      update item_mstr
      set balance = balance - v_quantity
      where item_id = v_item_id;
    else
      -- item doesn't exist, insert a new record.
      insert into item_mstr (item_id, balance) values (v_item_id, -v_quantity);
    end if;
  end loop;
  commit;
  dbms_output.put_line('balances updated in item_mstr table.');
```

exception

```
  when others then
    dbms_output.put_line('an error occurred while updating balances.');
```

end;

/

### **Table after updation:**



ITEM_ID	BALANCE
3568	7224.35
9868	9587.62
3310	8612.64
5977	3468.11
6670	819.53
9109	8132.7
4662	2531.46
1336	3785.48
1798	2004.01
4559	3777.38

5. **Write a PROCEDURE for raising salary of some employee by some amount. The PROCEDURE to be written may carry two parameters emp\_id and amt to be raised. Include two exceptions which will be raised when either emp\_id is not present or salary is NULL.**

**Creation of table:**

```
create table employees (
  employee_id number,
  employee_name varchar2(50),
  salary number
);
```

**Snapshot of the table:**

EMPLOYEE_ID	EMPLOYEE_NAME	SALARY
11001	Anish Banerjee	50000
11010	Devika Simlai	56000
11011	Sudip Dutta	33500
11100	Ashish Reddy	15000
11101	Vivek Kumar	(null)
11110	Prashanjit Basu	10000
11111	Ananya Sutradhar	20000
(null)	Tanis Ahamed	40000

**Procedure to raise salary:**

```
CREATE OR REPLACE PROCEDURE raise_salary(p_emp_id IN NUMBER, p_amt IN
NUMBER) IS
v_salary NUMBER;
BEGIN
SELECT salary INTO v_salary
FROM employees
WHERE employee_id = p_emp_id;
IF v_salary IS NULL THEN
RAISE_APPLICATION_ERROR(-20001, 'Employee ID not found.');
```

END IF;

```
IF v_salary + p_amt IS NULL THEN
RAISE_APPLICATION_ERROR(-20002, 'Cannot raise salary. Current salary is NULL.');
```

END IF;

```
UPDATE employees
SET salary = salary + p_amt
WHERE employee_id = p_emp_id;
COMMIT;
```

DBMS\_OUTPUT.PUT\_LINE('Salary raised successfully.');

```
EXCEPTION
WHEN NO_DATA_FOUND THEN
RAISE_APPLICATION_ERROR(-20001, 'Employee ID not found.');
```

WHEN OTHERS THEN

```
RAISE_APPLICATION_ERROR(-20000, 'An error occurred: ' || SQLCODE || ' - ' ||
SQLERRM);
```

END raise\_salary;

/

**Calling the procedure - I:**

```
begin
raise_salary(11001, 5000);
end;
```

/

```
select* from employees
```

/

**Update in the table:**

EMPLOYEE_ID	EMPLOYEE_NAME	SALARY
11001	Anish Banerjee	55000
11010	Devika Simlai	56000
11011	Sudip Dutta	33500

### **Calling the procedure - II:**

```
begin  
  raise_salary(11010, 4000);  
end;  
/
```

```
select * from employees  
/
```

### **Update in the table:**

EMPLOYEE_ID	EMPLOYEE_NAME	SALARY
11001	Anish Banerjee	55000
11010	Devika Simlai	60000
11011	Sudip Dutta	33500

### **Calling the procedure where salary is null:**

```
begin  
  raise_salary(11101, 5000);  
end;  
/
```

```
select * from employees  
/
```

### **Update in the table:**

✘ ORA-20000: An error occurred: -20001 - ORA-20001: Employee ID not found. ORA-06512: at "USER\_4\_BC570E.RAISE\_SALARY", line 23 ORA-06512: at line 4

### **Calling the procedure where employee\_id is null:**

```
begin  
  raise_salary(11000, 5000);  
end;  
/
```

```
select * from employees  
/
```

### **Update in the table:**

✘ ORA-20001: Employee ID not found. ORA-06512: at "USER\_4\_BC570E.RAISE\_SALARY", line 21 ORA-06512: at line 4