

Database and Information Systems, Test Variant: PL/SQL

Duration of test: 75min

Maximal number of points/minimal number of points: 15/8

Only an result satisfying the whole task can be evaluated by non-zero points.

Tasks

1. Add attribute `price` into the table `Product` and set price to each product. Create procedure `PriceFix` with the output parameter `p_count` that checks the correct price of each purchase (the price of purchase is correct if it is equal to the price of product * number of bought pieces). If the price is not correct, the procedure updates it. The output parameter `p_count` returns the number of purchases with the uncorrect price.

```
alter table Product add price integer;

update Product set price = 1000;

create or replace procedure PriceFix (p_count out integer)
as
  CURSOR c is select nid, pid, purchasecount, price from purchase;
  pprice product.price%type;
begin
  for r in c loop
    select price into pprice from product where pid=r.pid;
    if r.purchasecount*pprice != r.price then
      update purchase set price = r.purchasecount*pprice where nid=r.nid;
      p_count := p_count + 1;
    end if;
  end loop;
end;
```

2. Create procedure InsertComplaint with the parameters p_nID, p_length and p_price that inserts the new complaint into the database. As a first step the procedure checks, if the purchase with p_nID exists. If it is not, the procedure prints out the warning and ends. If the purchase exists, the procedure find out, if the purchase has been complaint yet. If it is not, the order of the complaint is set to value 1, otherwise it is set to the number of purchase complaints + 1. The complaint can be inserted only in the case, that purchase has maximally 3 complaints yet. Otherwise the complaint is not inserted and the purchase is deleted together with all its complaints.

```
create or replace procedure InsertComplaint
(p_nID Complaint.nid%type,
p_duration Complaint.duration%type,
p_price Complaint.price%type)
as
v_count int;
begin
    select count(*) into v_count from Purchase where nID=p_nID;
    if (v_count = 1) then
        select count(*) into v_count from Complaint where nID=p_nID;
        if (v_count = 3) then
            delete from Complaint where nID=p_nID;
            delete from Purchase where nID=p_nID;
        else
            insert into Complaint values (p_nID, v_count + 1, p_duration, p_price);
            dbms_output.put_line('Complaint has been saved.');
```

```
end if;
    else
        dbms_output.put_line('The purchase with the nID = ' || p_nID || '
        does not exist.');
```