CSE 304

Online on PL/SQL

Set A

Problem 1: (Marks -3)

Write a procedure that will show the department names which have department_id greater than or equal 150.

Problem 2: (Marks-7)

First create table employee_db using the following DDL command.

Create table employee_db(

employee_id number(6),

first_name varchar2(50),

last_name varchar2(50),

department_id number(4)

);

Now carefully read the following description.

You can insert a row of the employee_db table where you must have to give the last_name. But any of the employee_id, first_name or department_id can be given or not. If the any of these three (employee_id, first_name and department_id) column values is not given, i.e., any of these is NULL then you have to search the "employees" table of the HR schema and fetch the corresponding entry based on only the 'last_name' [These should be case insensitive, that means, last_name 'Austin' should match with 'aUSTIN']. If you don't find any corresponding entry for this last_name or find multiple entries for this last_name then you have nothing to do and leave NULL values as these are.

For example, consider the following insert statement

insert into employee_db(employee_id, last_name) values (170, 'Austin');

Here, only the employee_id and last_name are given. That means, both the first_name and the department_id contain NULL values. So, search the “employees” table of the HR schema where the last_name is Austin and fetch corresponding first_name and department_id which are David and 60 respectively. So finally insert into the employee_db table with the corresponding full entry (170,’David’,’Austin’,60).

Instruction:

1. **USE TRIGGER and FUNCTIONS.**
2. **You cannot write any “select” operation in the trigger!**

More inputs and corresponding outputs:

```
insert into employee_db(last_name) values ('Abel');
insert into employee_db(last_name) values ('aUSTIN');
insert into employee_db(last_name) values ('Cambrault');
insert into employee_db(last_name) values ('Ben');
insert into employee_db(first_name,last_name) values ('Abel','Abel');
insert into employee_db(employee_id, last_name) values (170, 'Austin');
```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	DEPARTMENT_ID
174	Ellen	Abel	80
105	David	aUSTIN	60
(Null)	(Null)	Cambrault	(Null)
(Null)	(Null)	Ben	(Null)
174	Abel	Abel	80
170	David	Austin	60