****

**College of Engineering and Information Technology**

Database Administration – INT321

Lab # 4 Model Answer

**Employee**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Empno** | **Ename** | **Job** | **MGR** | **Hiredate** | **Sal** | **Comm** | **DeptNo** |
| 7839 | King | President |  | 17-Nov-81 | 5000 |  | 10 |
| 7698 | Blake | Manager | 7839 | 01-May-81 | 2850 |  | 30 |
| 7782 | Clark | Manager | 7839 | 09-Jun-81 | 2450 |  | 10 |
| 7566 | Johns | Manager | 7839 | 02-Apr-81 | 2950 |  | 20 |
| 7654 | Martin | Salesman | 7698 | 28-Sep-81 | 1250 | 1400 | 30 |
| 7499 | Allen | Salesman | 7698 | 20-Feb-81 | 1600 | 300 | 30 |
| 7844 | Turner | Salesman | 7698 | 08-Sep-81 | 1500 | 0 | 30 |
| 7900 | James | Clerk | 7698 | 03-Dec-81 | 950 |  | 30 |
| 7521 | Ward | Salesman | 7698 | 22-Feb-81 | 1250 | 500 | 30 |
| 7902 | Ford | Analyst | 7566 | 03-Dec-81 | 3000 |  | 20 |
| 7369 | Smith | Clerk | 7902 | 17-Dec-80 | 800 |  | 20 |
| 7788 | Scott | Analyst | 7566 | 09-Dec-82 | 3000 |  | 20 |
| 7876 | Adams | Clerk | 7788 | 12-Jan-83 | 1100 |  | 20 |
| 7934 | Miller | Clerk | 7782 | 23-Jan-82 | 1300 |  | 10 |

**Department**

|  |  |  |
| --- | --- | --- |
| **DeptNo** | **Dname** | **Location** |
| 10 | Accounting | Ajman |
| 20 | Research | Sharajah |
| 30 | Sales | Abu Dhabi |
| 40 | Operations | Dubai |

**Department\_Audit**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DeptNo** | **Dname** | **Old\_Location** | **New\_Location** | **Change\_Date** |
| 10 | Accounting | Ajman | Hata | 12-Dec-2018 |
| 20 | Research | Sharajah | Zaid | 10-Jan-2017 |

Consider the following table two tables named bookshelf and bookshelf\_audit

Bookshelf

|  |  |  |
| --- | --- | --- |
| Column Name | Data type |  |
| title | varchar2(100) |  |
| publisher | varchar2(20) |  |
| category\_name | varchar2(20) |  |
| rating | number(2) |  |
|  |  |  |
|  |  |  |

Bookshelf\_Audit

|  |  |  |
| --- | --- | --- |
| Column Name | Data type |  |
| title | varchar2(100) |  |
| publisher | varchar2(20) |  |
| category\_name | varchar2(20) |  |
| old\_rating | number(2) |  |
| new\_rating | number(2) |  |
| audit\_date | date |  |

1. Write a PL/SQL statements that create a trigger named bookshelf\_bef\_upd\_row. The trigger fires when there is an update on the table Bookshelf. Specifically the trigger executes if the value of the new value of the rating is less than the old value of the rating. When the trigger fires, a new row should be inserted into the Bookshelf Audit keeping the old rating, the new rating, and the date of update.

ANS:

Create or replace trigger bookshelf\_bef\_upd\_row

before update on bookshelf

For each row

when (new.rating < old.rating)

Begin

Insert into bookshelf\_audit (title, publisher, category\_name, old\_rating,

new\_rating, audit\_date) values

(:old.title, :old.publisher, :old.category\_name, :old.rating,

:new.rating, sysdate);

end;

2. Write a PL/SQL statements that create a trigger named bookshelf\_bef\_upd\_ins\_row. The trigger when it fires it will perform one of the following tasks: If the triggering event is inserting, the triggers inserts a row in the bookshelf\_audit table. The new row should use the values of the columns of bookshelf along with the current data. When the triggering event is update on the column rating of the table Bookshelf, the trigger should insert a row in the bookshelf\_audit using all column values of bookshelf a long with old rating value and the current date.

ANS:

Create or replace trigger bookshelf\_bef\_upd\_ins\_row

before insert or update of rating on bookshelf

For each row

Begin

if inserting then

Insert into bookshelf\_audit (title, publisher, category\_name,

new\_rating, audit\_date) values

(:new.title, :new.publisher, :new.category\_name,

:new.rating, sysdate);

else

Insert into bookshelf\_audit (title, publisher, category\_name,

old\_rating, new\_rating, audit\_date) values

(:old.title, :old.publisher, :old.category\_name, :old.rating,

, :new.rating, sysdate);

end if;

end bookshelf\_bef\_upd\_ins\_row;

3. Write a PL/SQL statements that create a trigger named department\_bef\_upd\_row. The trigger fires when there is an update on Location column of the department table. Specifically the trigger should perform the following task if it fires. The triggers update the salaries of all employees affected by change of department location by adding 1000 to their salaries. The trigger should also insert a record in the department\_audit table. For the **Change\_Date** using the current date

ANS:

Create or replace trigger department\_bef\_upd\_row

before update of location on Department

For each row

Begin

update employee set sal = sal +1000 where employee.deptNo = :old.deptNO;

insert into Department\_Audit (DeptNo, Dname, Old\_Location, New\_Location, Change\_Date) Values (:old.DeptNo, :old.Dname, :old.Location, :new.Location, sysdate);

end department\_bef\_upd\_row;

/