

TRIGGERS

I. insert_customer_trigger

II. CREATE OR REPLACE TRIGGER insert_customer_trigger
AFTER insert ON customer FOR EACH ROW

```
begin
    insert into tx values(
        tra_seq.nextval,
        'customer is inserted',
        :new.cus_id,
        0,
        null,
        null,
        0,
        sysdate );
end;
```

III. update_customer_trigger

CREATE OR REPLACE TRIGGER update_customer_trigger
AFTER update or delete ON custome FOR EACH ROW

```
begin
    insert into tx values(
        tra_seq.nextval,
        'customer is updated/deleted',
        :old.cus_id,
        0,
        null,
        null,
        0,
        sysdate);
end;
```

IV. insert_rent_trigger

CREATE OR REPLACE TRIGGER insert_rent_trigger
AFTER insert or update or delete ON rent FOR EACH ROW

```
begin
    insert into tx values(
        tra_seq.nextval,
        'issue/return/fine',
        :new.cus_id,
        :old.item_id,
        sysdate,
        sysdate,0);
end;
```

V. insert_employee_Trigger

```
CREATE OR REPLACE TRIGGER insert_employee_trigger
AFTER insert ON employee FOR EACH ROW
begin
    insert into tx values(
        tra_seq.nextval,
        'employee is inserted',
        :new.emp_id,
        null,
        null,
        null,
        null,
        sysdate);
end;
```

VI. update_employee_trigger

```
CREATE OR REPLACE TRIGGER update_employee_trigger
AFTER update or delete ON employee FOR EACH ROW
begin
    insert into tx values(
        tra_seq.nextval,
        'employee is updated/deleted',
        :old.emp_id,
        null,
        null,
        null,
        null,
        sysdate);
end;
```

VII. Book_Insert_Trigger

```
create or replace trigger book_insert_trigger
after insert on book for each row
begin
    insert into rec_audit values(
        rec_seq.nextval,
        :new.book_id,
        'Book is Added',
        sysdate);
end;
```

VIII. Book_Update_Trigger

```
create or replace trigger book_update_trigger after update or delete on book for each row
begin
```

```
insert into rec_audit values(
    rec_seq.nextval,
    :old.book_id,
    'Book is updates/deleted',
    sysdate);
end;
```

IX. Trigger Video_Insert_Trigger

```
create or replace trigger video_insert_trigger after insert on video for each row
begin
    insert into rec_audit values(
        rec_seq.nextval,
        :new.video_id,
        'Video is Added',
        sysdate);
end;
```

X. Video_Update_Trigger

```
create or replace trigger Video_update_trigger after update or delete on video for each row
begin
    insert into rec_audit values(
        rec_seq.nextval,
        :old.video_id,
        'Video is updates/deleted',
        sysdate);
end;
```

XI. fine_trigger

```
create or replace trigger fine_trigger before insert or update or delete on tx for each row
declare
    today date:=sysdate;
    fine_date date;
begin
    select return_date into fine_date from tx where cus_id_emp_id=:new.cus_id_emp_id;
    if today > fine_date then
        update tx set fine = fine+5 where cus_id_emp_id = :new.cus_id_emp_id;
    end if;
end;
```