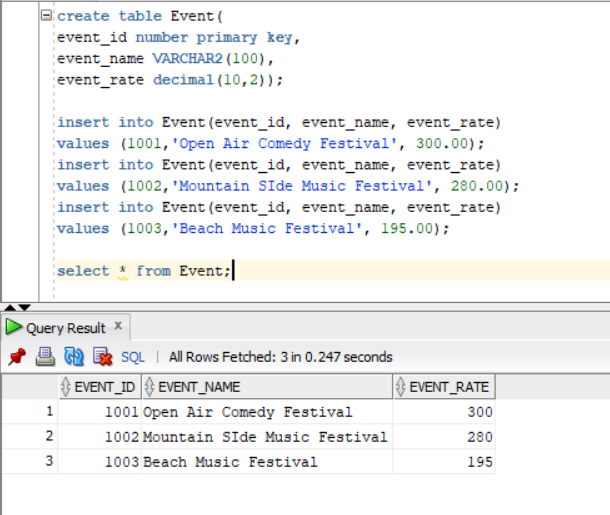
ADDB7311

ICE Task 3 - PL/SQL Pg 101

ST10305921 – Zoe Heyneke

Create tables:

* Event Table:

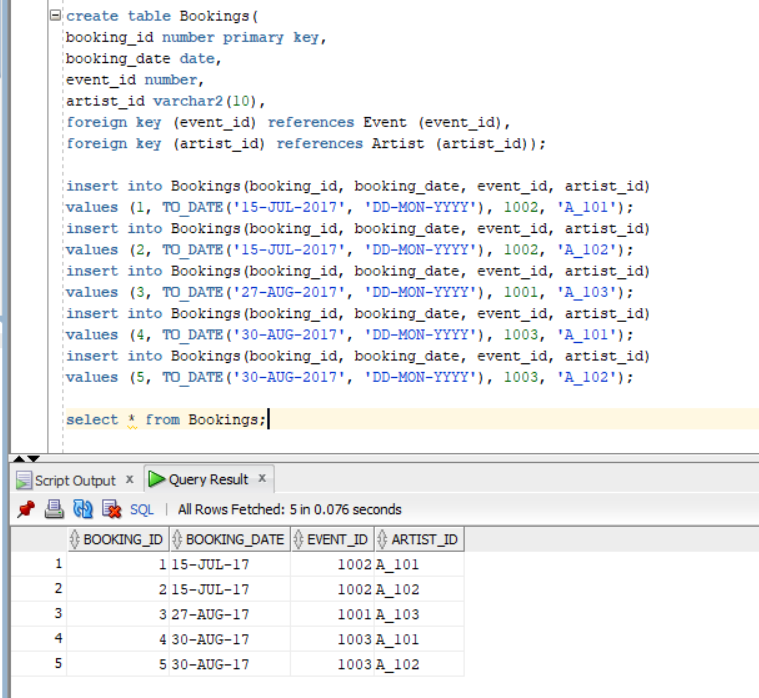


* Artists Table:

A screenshot of a computer

Description automatically generated

* Bookings Table:



create table Event(

event\_id number primary key,

event\_name VARCHAR2(100),

event\_rate decimal(10,2));

insert into Event(event\_id, event\_name, event\_rate)

values (1001,'Open Air Comedy Festival', 300.00);

insert into Event(event\_id, event\_name, event\_rate)

values (1002,'Mountain SIde Music Festival', 280.00);

insert into Event(event\_id, event\_name, event\_rate)

values (1003,'Beach Music Festival', 195.00);

select \* from Event;

create table Artist(

artist\_id varchar2(10) primary key,

artist\_name varchar2(100),

artist\_email varchar2(100));

insert into Artist (artist\_id, artist\_name, artist\_email)

values ('A\_101', 'Max Trillion', 'maxt@isat.com');

insert into Artist (artist\_id, artist\_name, artist\_email)

values ('A\_102', 'Music Mayhem', 'mayhem@ymail.com');

insert into Artist (artist\_id, artist\_name, artist\_email)

values ('A\_103', 'LOL Man', 'lol@isat.com');

select \* from Artist;

create table Bookings(

booking\_id number primary key,

booking\_date date,

event\_id number,

artist\_id varchar2(10),

foreign key (event\_id) references Event (event\_id),

foreign key (artist\_id) references Artist (artist\_id));

insert into Bookings(booking\_id, booking\_date, event\_id, artist\_id)

values (1, TO\_DATE('15-JUL-2017', 'DD-MON-YYYY'), 1002, 'A\_101');

insert into Bookings(booking\_id, booking\_date, event\_id, artist\_id)

values (2, TO\_DATE('15-JUL-2017', 'DD-MON-YYYY'), 1002, 'A\_102');

insert into Bookings(booking\_id, booking\_date, event\_id, artist\_id)

values (3, TO\_DATE('27-AUG-2017', 'DD-MON-YYYY'), 1001, 'A\_103');

insert into Bookings(booking\_id, booking\_date, event\_id, artist\_id)

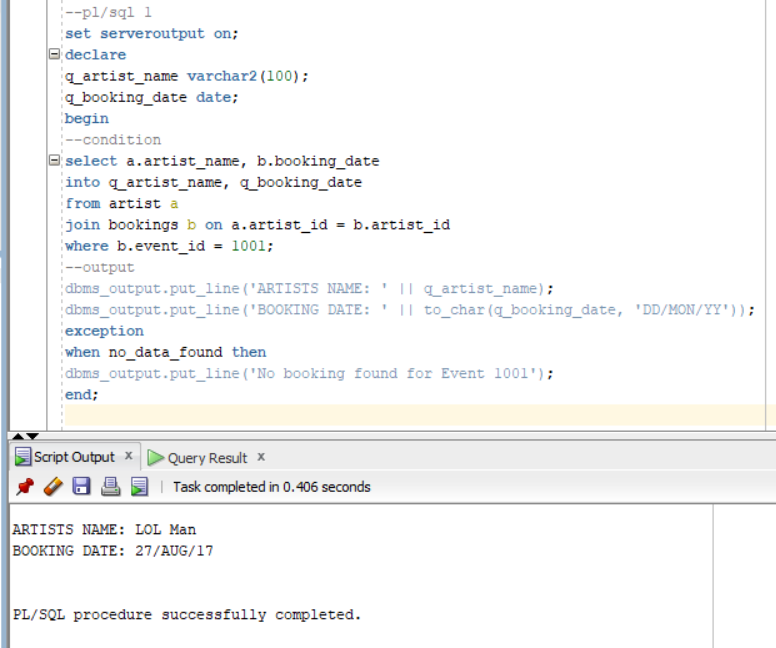
values (4, TO\_DATE('30-AUG-2017', 'DD-MON-YYYY'), 1003, 'A\_101');

insert into Bookings(booking\_id, booking\_date, event\_id, artist\_id)

values (5, TO\_DATE('30-AUG-2017', 'DD-MON-YYYY'), 1003, 'A\_102');

select \* from Bookings;

* Display the artist name and the booking date for Event ID equal to 1001



--pl/sql 1

set serveroutput on;

declare

q\_artist\_name varchar2(100);

q\_booking\_date date;

begin

--condition

select a.artist\_name, b.booking\_date

into q\_artist\_name, q\_booking\_date

from artist a

join bookings b on a.artist\_id = b.artist\_id

where b.event\_id = 1001;

--output

dbms\_output.put\_line('ARTISTS NAME: ' || q\_artist\_name);

dbms\_output.put\_line('BOOKING DATE: ' || to\_char(q\_booking\_date, 'DD/MON/YY'));

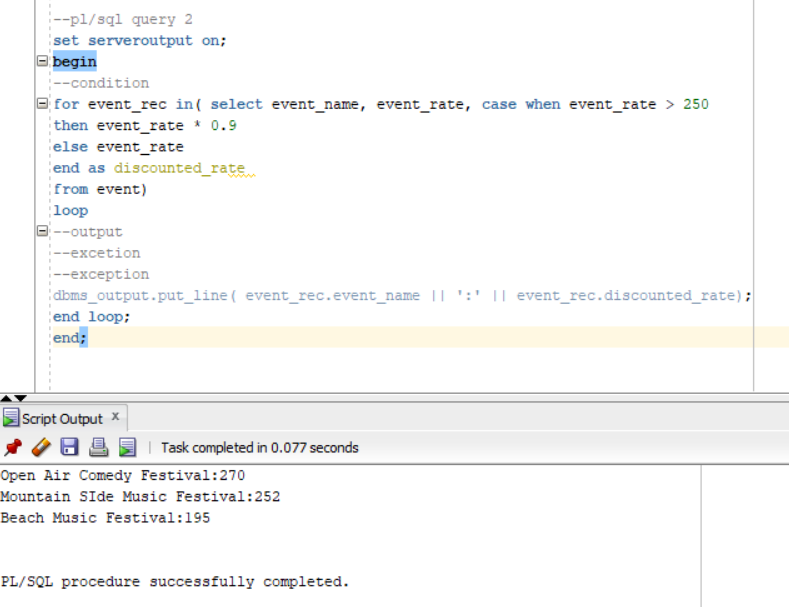
exception

when no\_data\_found then

dbms\_output.put\_line('No booking found for Event 1001');

end;

* 10% discount on events that cost over R250



--pl/sql query 2

set serveroutput on;

begin

--condition

for event\_rec in( select event\_name, event\_rate, case when event\_rate > 250

then event\_rate \* 0.9

else event\_rate

end as discounted\_rate

from event)

loop

--output

--excetion

--exception

dbms\_output.put\_line( event\_rec.event\_name || ':' || event\_rec.discounted\_rate);

end loop;

end;