**Report Midlands Theatre (MT) Company**

Submitted by: Ahsan Mushtaq

P2688331

**Task 1: Dimension selection and fact identification**

Dimensions:

dTheatre for Theatre

dProduction for Production

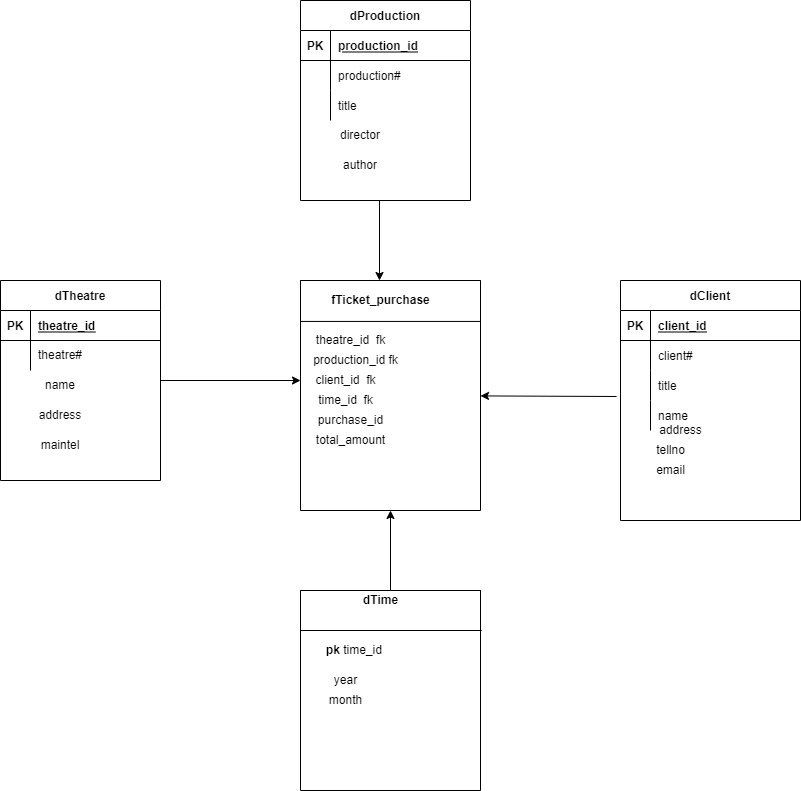
dClient for Client

dTime for time

Fact:

fTicket\_purchase for the fact

**Task 2: Design the star schema for the data mart, and identify the corresponding PKs and FKs.**



**Task 3: According to given data and requirements, determine the relevant attributes and suitable granularity in your data mart.**

For star schema, please see the figure. Theatre#, Production# and Client# are kept as nature keys for dimensions. Theatre\_id, Production\_id, Client\_id and time\_id are surrogate keys for dimensions. For dTheatre, the PK is Theatre\_id. For dProduction , the PK is Production\_id. For dClient, the PK is Client\_id and dtime PK is time\_id. For fTicket\_purchase, the PK is a composite key consisting of Theatre\_id, Production\_id, Client\_id and time\_id. Theatre\_id is a FK referring to dTheatre, Production\_id is a FK referring to dProduction, Client\_id is a FK referring to dClient and time\_id is FK referring to dtime.

The relevant attributes are shown in the star schema.

**Granularity**

**The granularity here in each row of ftp are yearly total sale for each theatre and Productions with highest sale and client who visited theatre for more than 4 months**.

**Task 4: Map your star schema to logical relations.**

Logical relations:

dTheatre(**theatre\_id**, theatre#, name, address, maintel);

dProduction(**production\_id**, production#, title, director, author);

dClient( **client\_id**, client#, title, name, tellno, email);

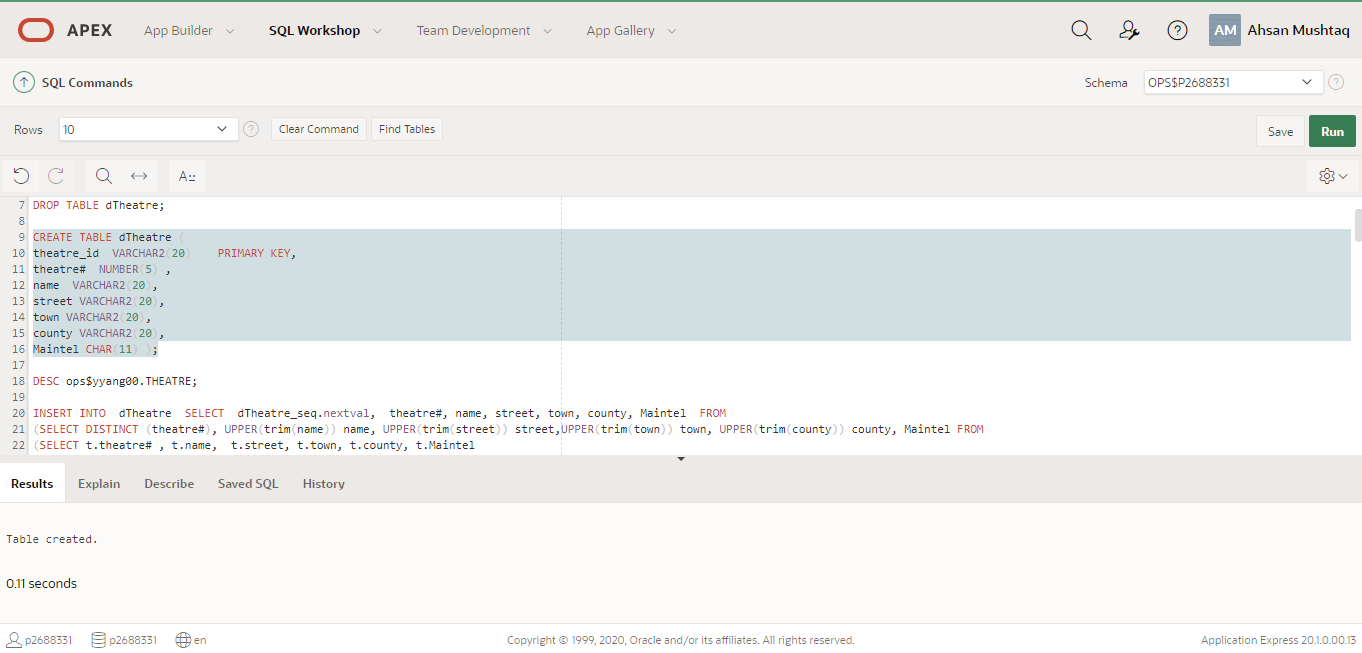
dTime(**time\_id**, year, month);

fTicket\_Purchase (***theatre\_id, production\_id, time\_id*, client\_id** , totalamount)

theatre\_id is a FK referring to dTheatre, production\_id is a FK referring to dProduction, client\_id is a FK referring to dClient and time\_id is FK referring to dTime.

**Task 5: Create the corresponding tables in Oracle using SQL**.

**Dimension of Theatre**



CREATE SEQUENCE dTheatre\_seq

START WITH 1

INCREMENT BY 1

NOCACHE

NOCYCLE;

DROP TABLE dTheatre;

CREATE TABLE dTheatre (

theatre\_id VARCHAR2(20) PRIMARY KEY,

theatre# NUMBER(5) ,

name VARCHAR2(20),

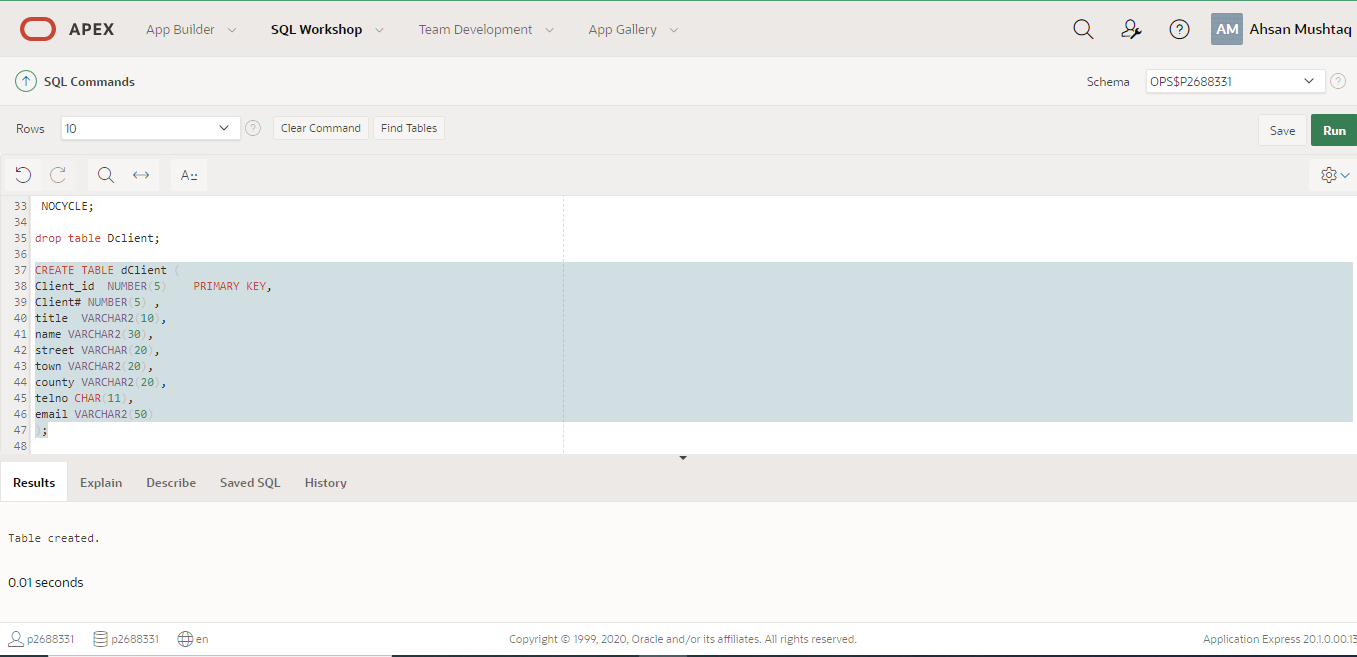
street VARCHAR2(20),

town VARCHAR2(20),

county VARCHAR2(20),

Maintel CHAR(11) );

**Dimension table for Client**



CREATE SEQUENCE dClient\_seq

START WITH 1

INCREMENT BY 1

NOCACHE

NOCYCLE;

CREATE TABLE dClient (

Client\_id NUMBER(5) PRIMARY KEY,

Client# NUMBER(5) ,

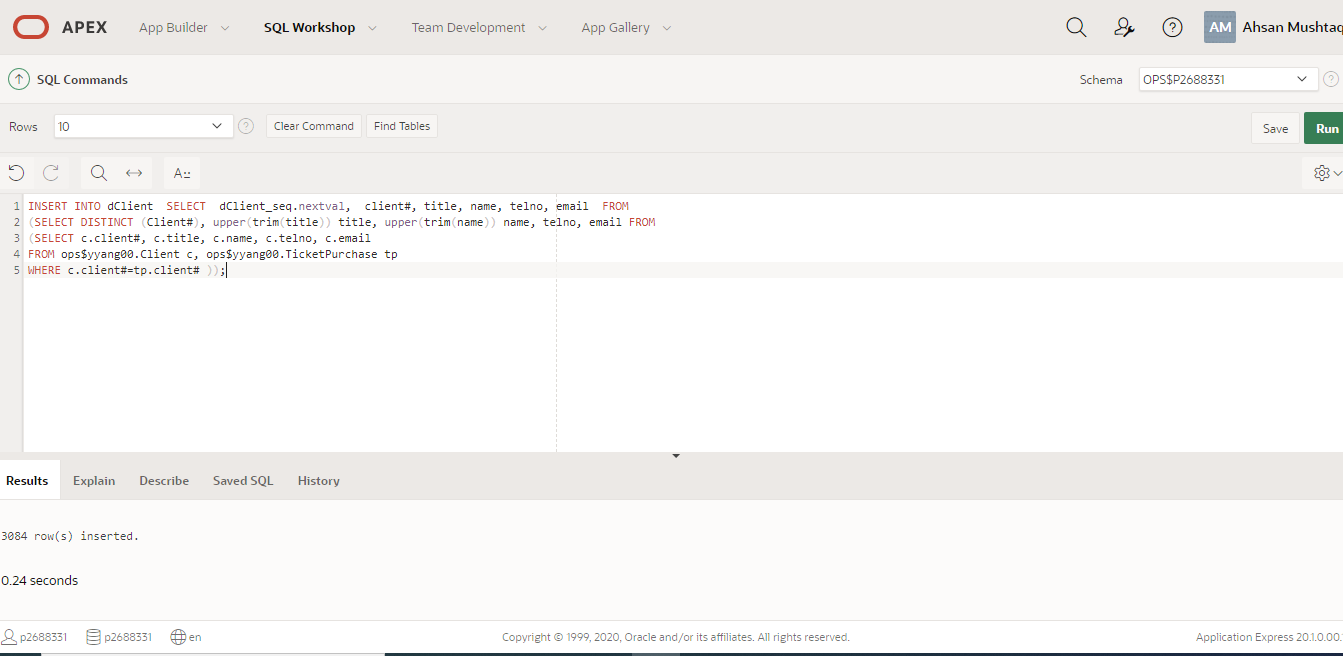
title VARCHAR2(10),

telno CHAR(11),

email VARCHAR2(50)

);

**Dimension table for production**



CREATE SEQUENCE dProduction\_seq

START WITH 1

INCREMENT BY 1

NOCACHE

NOCYCLE;

drop table dproduction;

CREATE TABLE dProduction (

production\_id number(5) PRIMARY KEY,

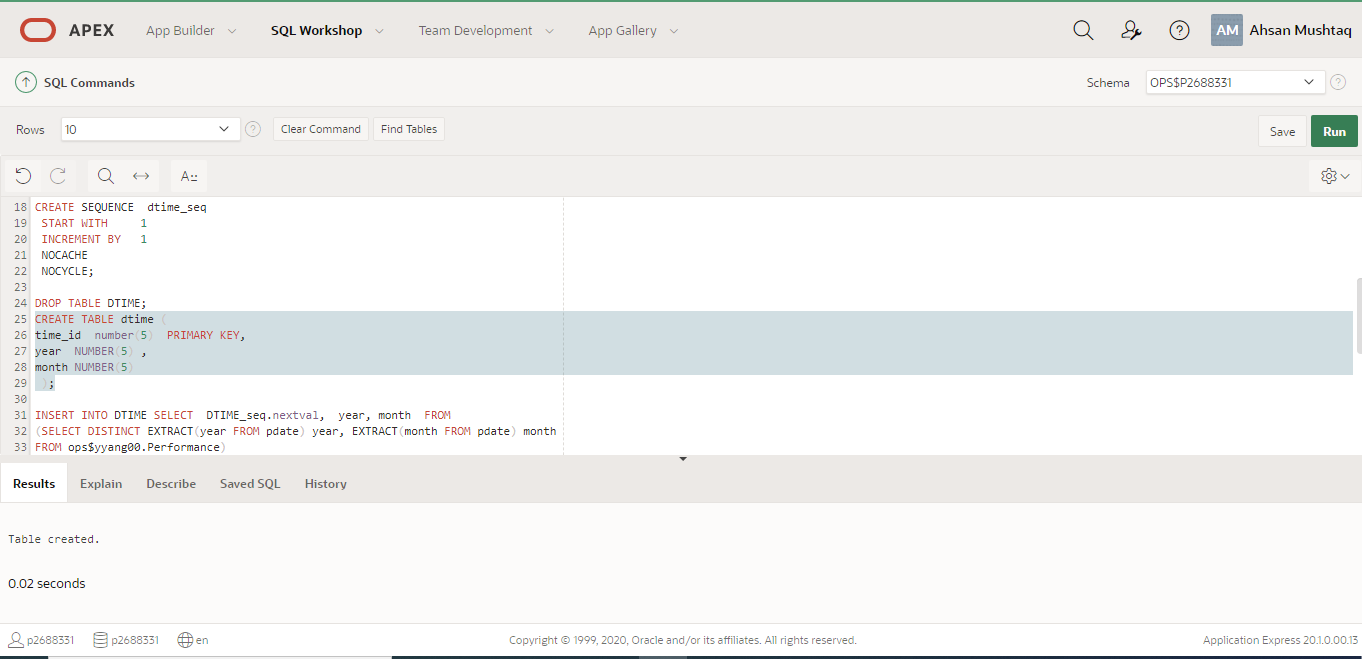
P# number(5) ,

title VARCHAR2(20),

productiondirector varchar2(20),

playauthor varchar2(20) );

**dimension table for time**



CREATE SEQUENCE dtime\_seq

START WITH 1

INCREMENT BY 1

NOCACHE

NOCYCLE;

DROP TABLE DTIME;

CREATE TABLE dtime (

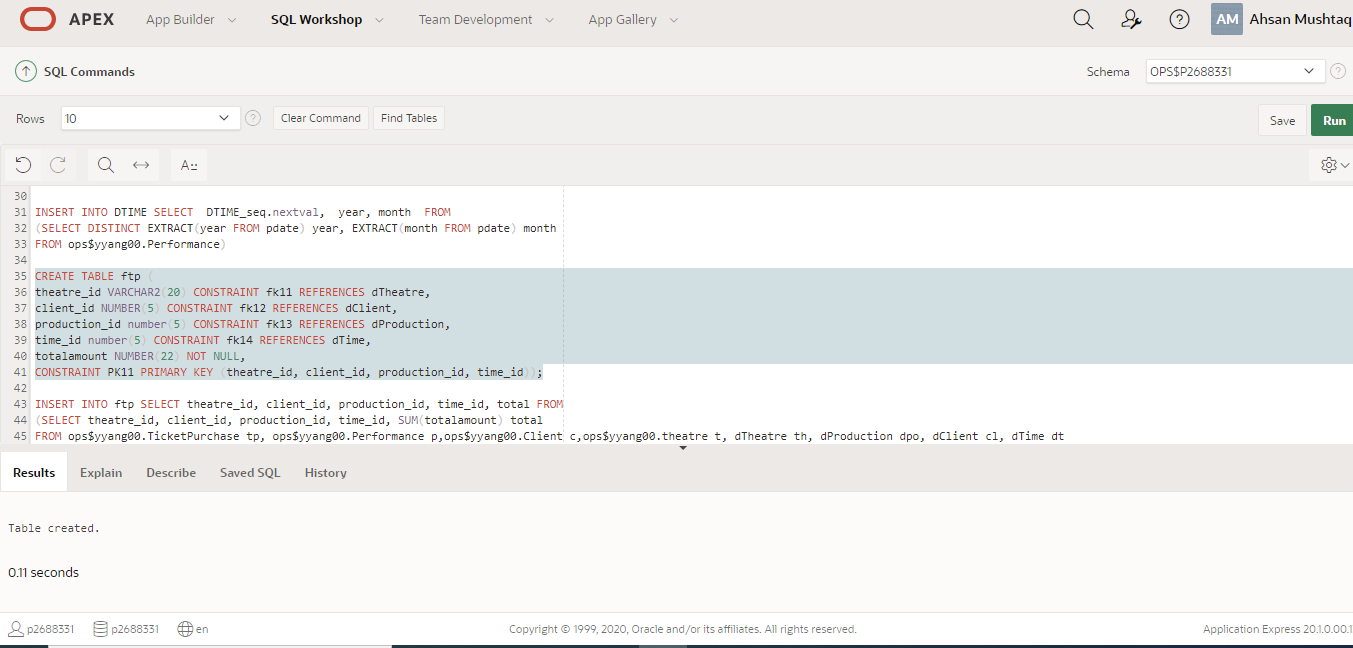
time\_id number(5) PRIMARY KEY,

year NUMBER(5) ,

month NUMBER(5)

);

**Fact table**



CREATE TABLE ftp (

theatre\_id VARCHAR2(20) CONSTRAINT fk11 REFERENCES dTheatre,

client\_id NUMBER(5) CONSTRAINT fk12 REFERENCES dClient,

production\_id number(5) CONSTRAINT fk13 REFERENCES dProduction,

time\_id number(5) CONSTRAINT fk14 REFERENCES dTime,

totalamount NUMBER(22) NOT NULL,

CONSTRAINT PK11 PRIMARY KEY (theatre\_id, client\_id, production\_id, time\_id));

**Task 6: Identify your source data from the OLTP database and design your data extraction rules. You need to give a detailed mapping and transformation list from the source to the destination.**

Theatre joins performance join ticket purchase

theatre#

name

address

mainTel

…

d\_date

totalamount

dTheatre

theatre#

name

address

maintel

format

dt

time\_id

year

month

fTicket\_purchase

Time\_id

Totalamount

Destination

Source Destination

Production joins performance join ticket purchase

production#

title

director

author

…

d\_date

total\_amount

dProduction

production#

title

director

author

format

dt

time\_id

year

month

fTicket\_purchase

Time\_id

Total\_amount

Source

Destination

Client joins ticket purchase

Client#

title

address

tellno

email

…

d\_date

total\_amount

dClient

Client#

title

address

tellno

email

format

dt

time\_id

year

month

fTicket\_purchase

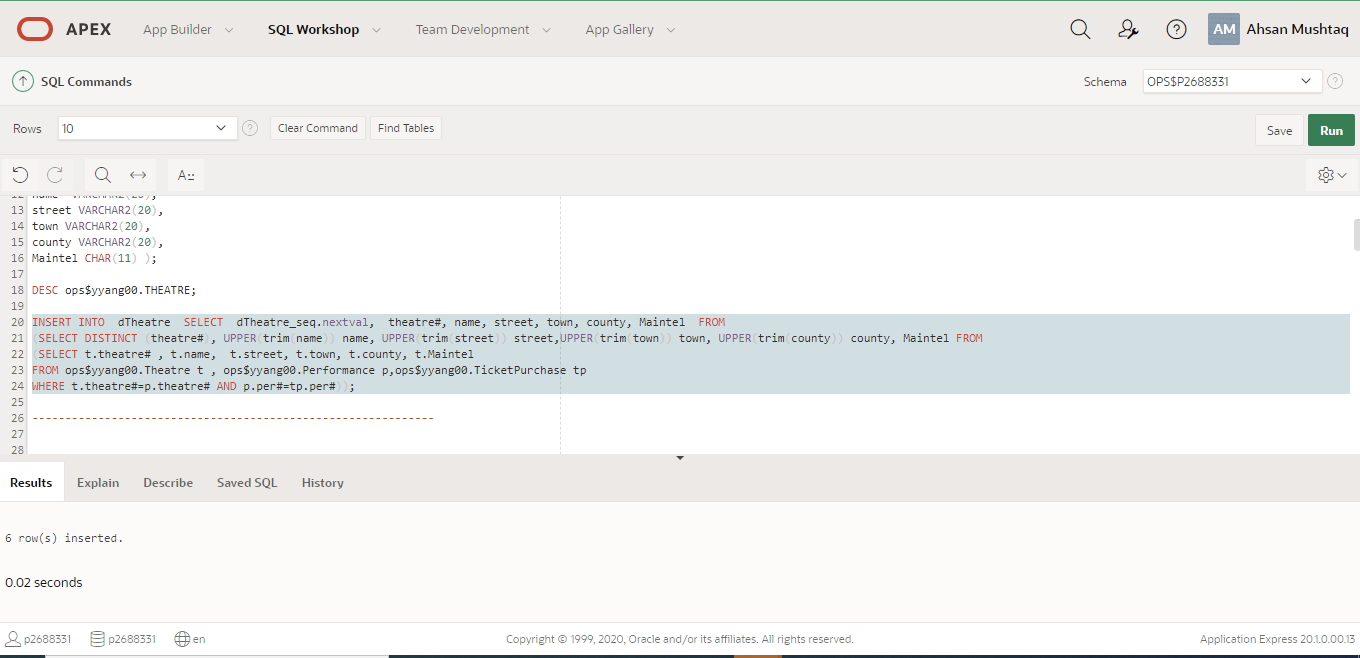
Time\_id

Total\_amount

Source Destination

**Task 7: Implement your data extraction, transformation and loading through Oracle SQL. The number of rows extracted into each dimension or fact table in your data mart should be printed from Oracle query.**

**ETL on dimension of Theatre Table**



INSERT INTO dTheatre SELECT dTheatre\_seq.nextval, theatre#, name, street, town, county, Maintel FROM

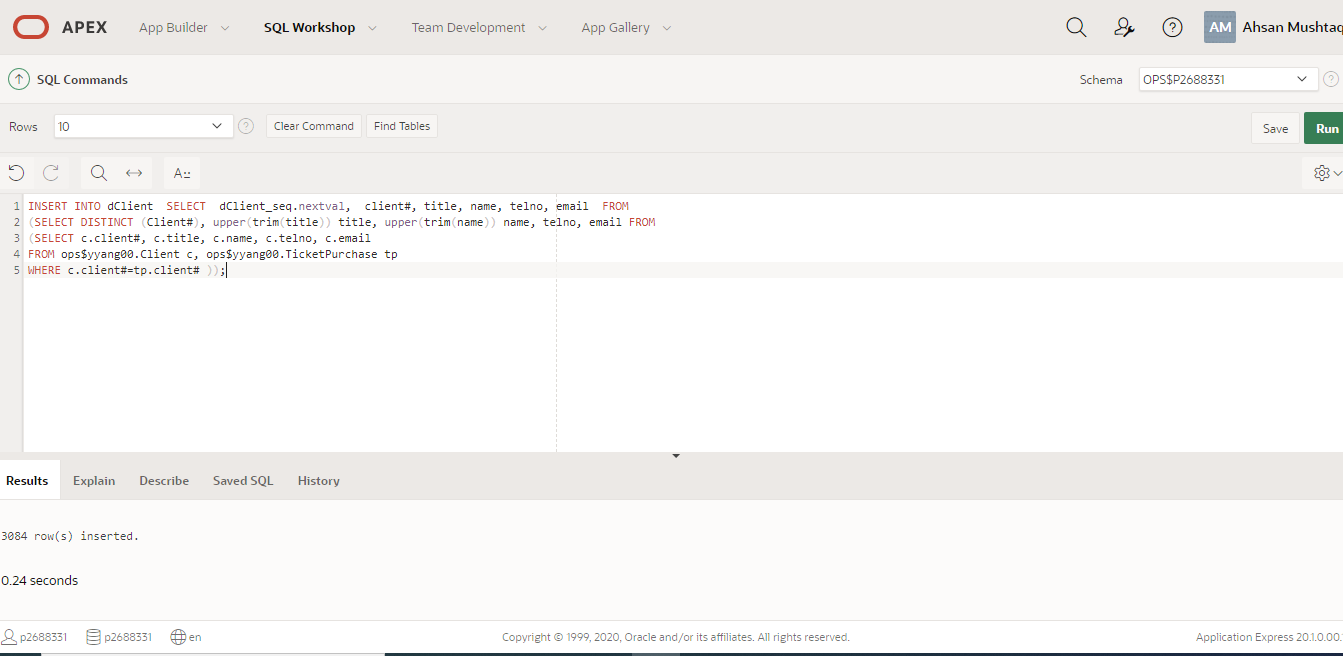
(SELECT DISTINCT (theatre#), UPPER(trim(name)) name, UPPER(trim(street)) street,UPPER(trim(town)) town, UPPER(trim(county)) county, Maintel FROM

(SELECT t.theatre# , t.name, t.street, t.town, t.county, t.Maintel

FROM ops$yyang00.Theatre t , ops$yyang00.Performance p,ops$yyang00.TicketPurchase tp

WHERE t.theatre#=p.theatre# AND p.per#=tp.per#));

**ETL on dimension of client table**



INSERT INTO dProduction SELECT dProduction\_seq.nextval, p#, title, productiondirector, playauthor FROM

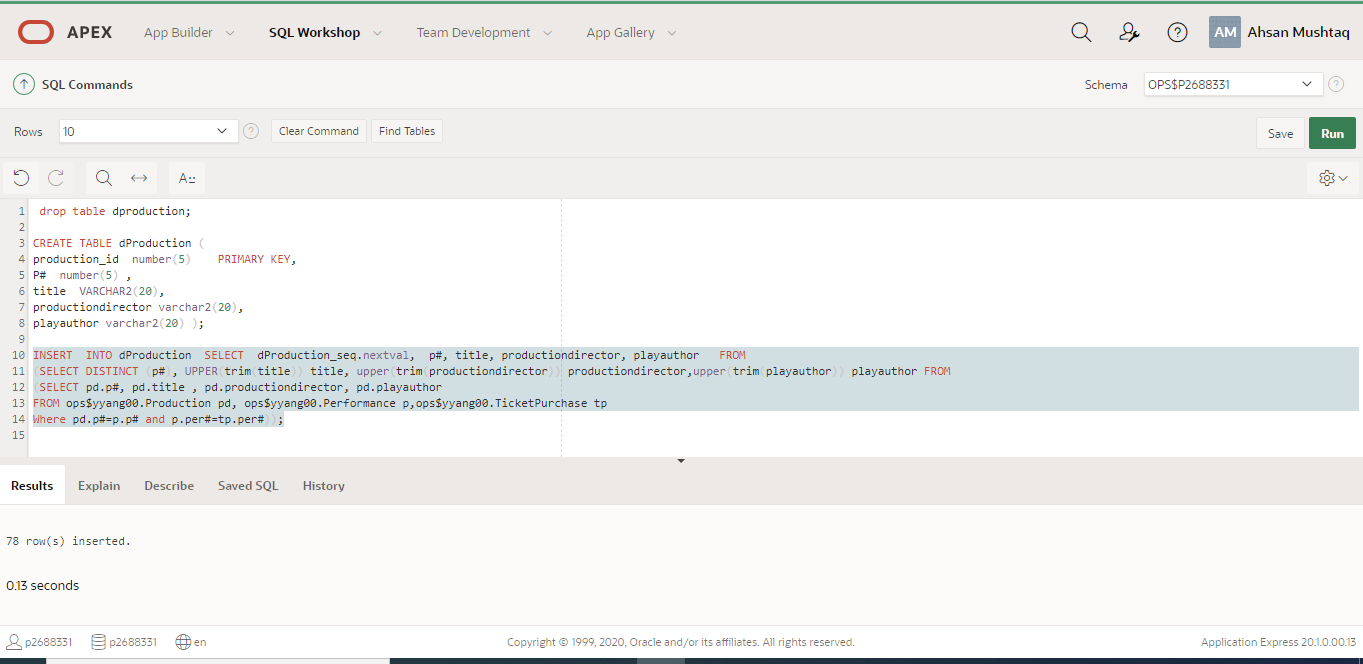
(SELECT DISTINCT (p#), UPPER(trim(title)) title, upper(trim(productiondirector)) productiondirector,upper(trim(playauthor)) playauthor FROM

(SELECT pd.p#, pd.title , pd.productiondirector, pd.playauthor

FROM ops$yyang00.Production pd, ops$yyang00.Performance p,ops$yyang00.TicketPurchase tp

Where pd.p#=p.p# and p.per#=tp.per#));

**ETL for dimention of production**



INSERT INTO dProduction SELECT dProduction\_seq.nextval, p#, title, productiondirector, playauthor FROM

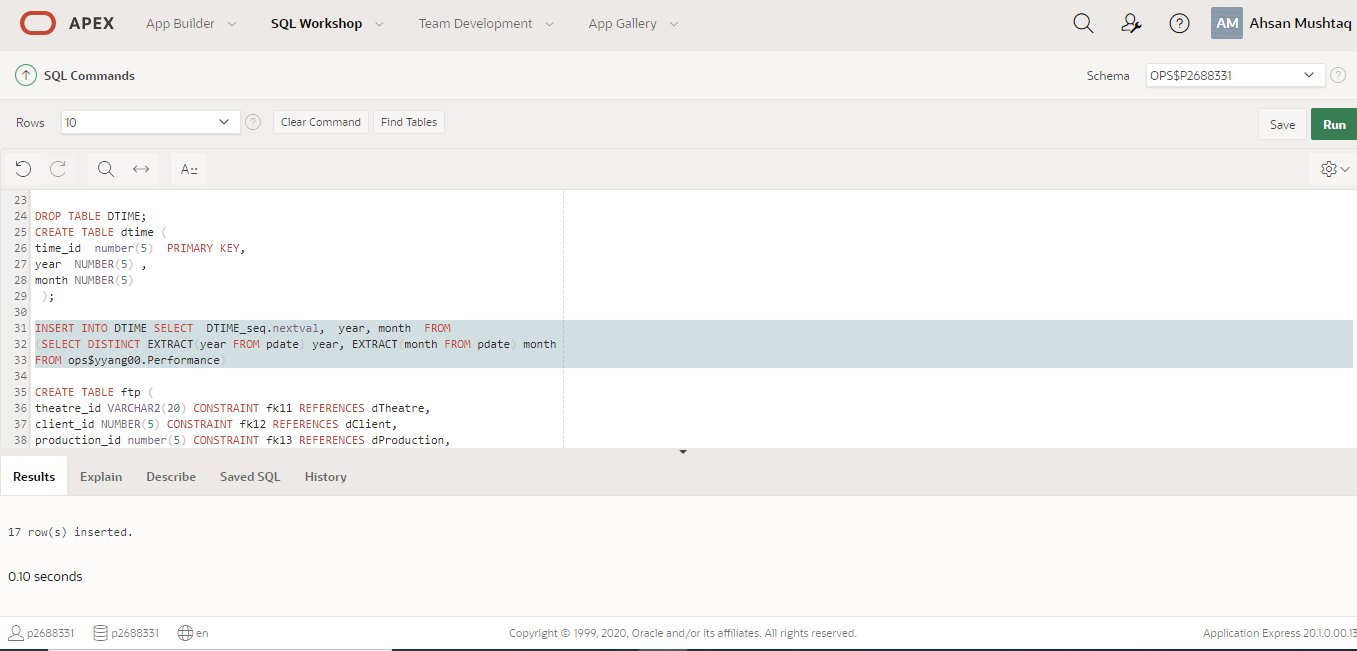
(SELECT DISTINCT (p#), UPPER(trim(title)) title, upper(trim(productiondirector)) productiondirector,upper(trim(playauthor)) playauthor FROM

(SELECT pd.p#, pd.title , pd.productiondirector, pd.playauthor

FROM ops$yyang00.Production pd, ops$yyang00.Performance p,ops$yyang00.TicketPurchase tp

Where pd.p#=p.p# and p.per#=tp.per#));

**ETL for time dimension table**

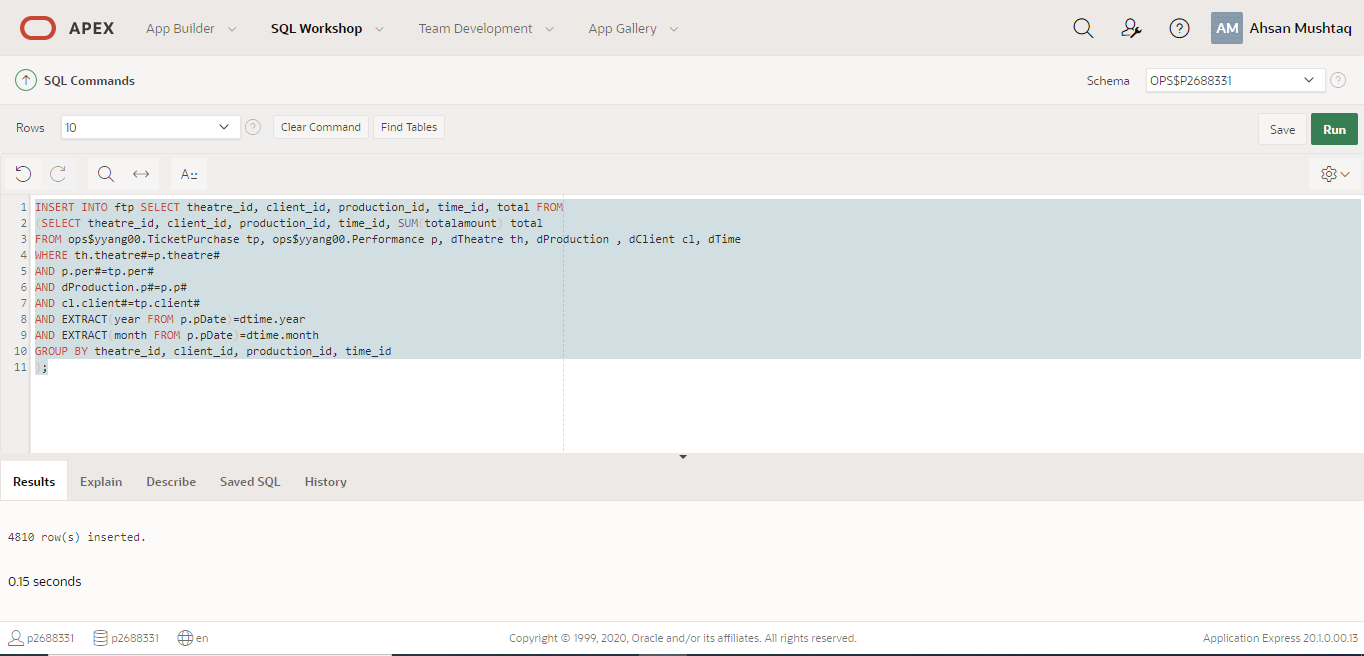


INSERT INTO DTIME SELECT DTIME\_seq.nextval, year, month FROM

(SELECT DISTINCT EXTRACT(year FROM pdate) year, EXTRACT(month FROM pdate) month

FROM ops$yyang00.Performance)

**ETL for Fact Table**



INSERT INTO ftp SELECT theatre\_id, client\_id, production\_id, time\_id, total FROM

(SELECT theatre\_id, client\_id, production\_id, time\_id, SUM(totalamount) total

FROM ops$yyang00.TicketPurchase tp, ops$yyang00.Performance p,ops$yyang00.Client c,ops$yyang00.theatre t, dTheatre th, dProduction dpo, dClient cl, dTime dt

WHERE th.theatre#=p.theatre#

AND p.per#=tp.per#

AND dpo.p#=p.p#

AND c.client#=tp.client#

AND EXTRACT(year FROM p.pDate)=dt.year

AND EXTRACT(month FROM p.pDate)=dt.month

GROUP BY theatre\_id, client\_id, production\_id, time\_id

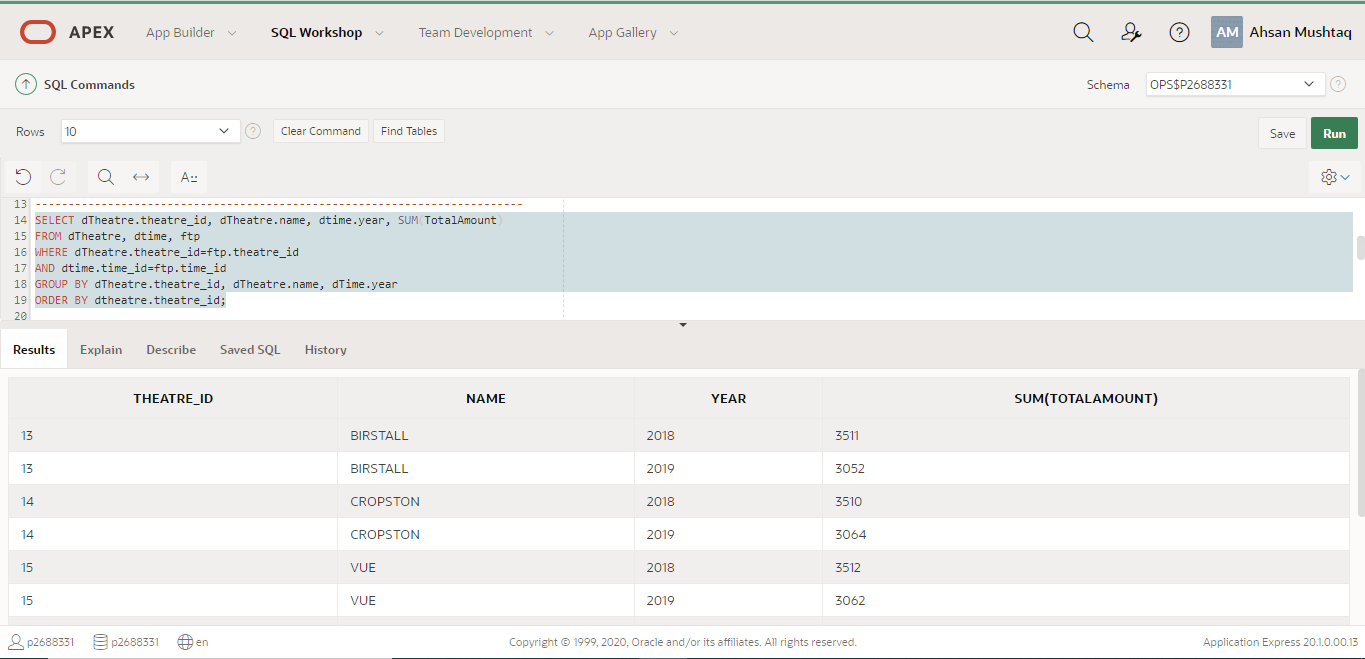
);

**Task 8: Comment on how your data mart satisfies the requirements of MT. Implement the required data analysis requirements for both the data mart and the original OLTP database. Compare your queries and comments on the advantages of a data mart in analysis operations.**

**SQL for required queries (both Data Mart and the relational model)**

* **Yearly total sale for each theatre**

**Data Mart Query**



SELECT dTheatre.theatre\_id, dTheatre.name, dtime.year, SUM(TotalAmount)

FROM dTheatre, dtime, ftp

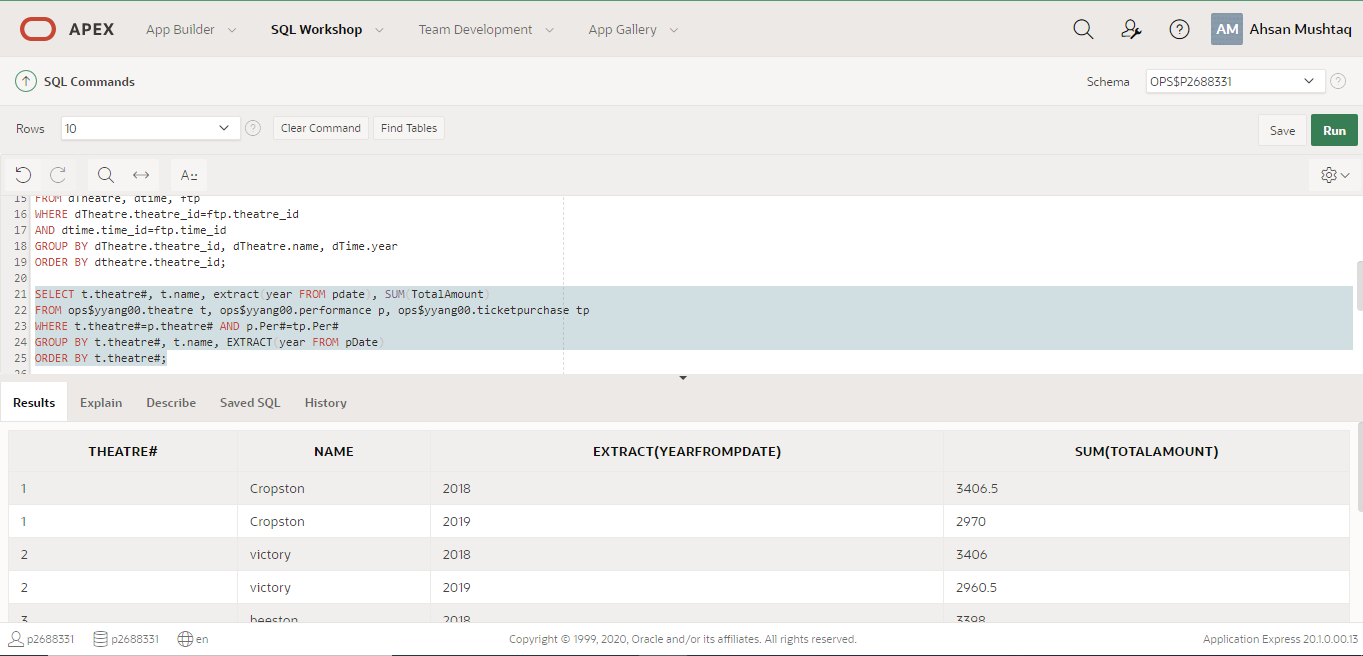
WHERE dTheatre.theatre\_id=ftp.theatre\_id

AND dtime.time\_id=ftp.time\_id

GROUP BY dTheatre.theatre\_id, dTheatre.name, dTime.year

ORDER BY dtheatre.theatre\_id;

**Analysis for original OLTP Relational database**



SELECT t.theatre#, t.name, extract(year FROM pdate), SUM(TotalAmount)

FROM ops$yyang00.theatre t, ops$yyang00.performance p, ops$yyang00.ticketpurchase tp

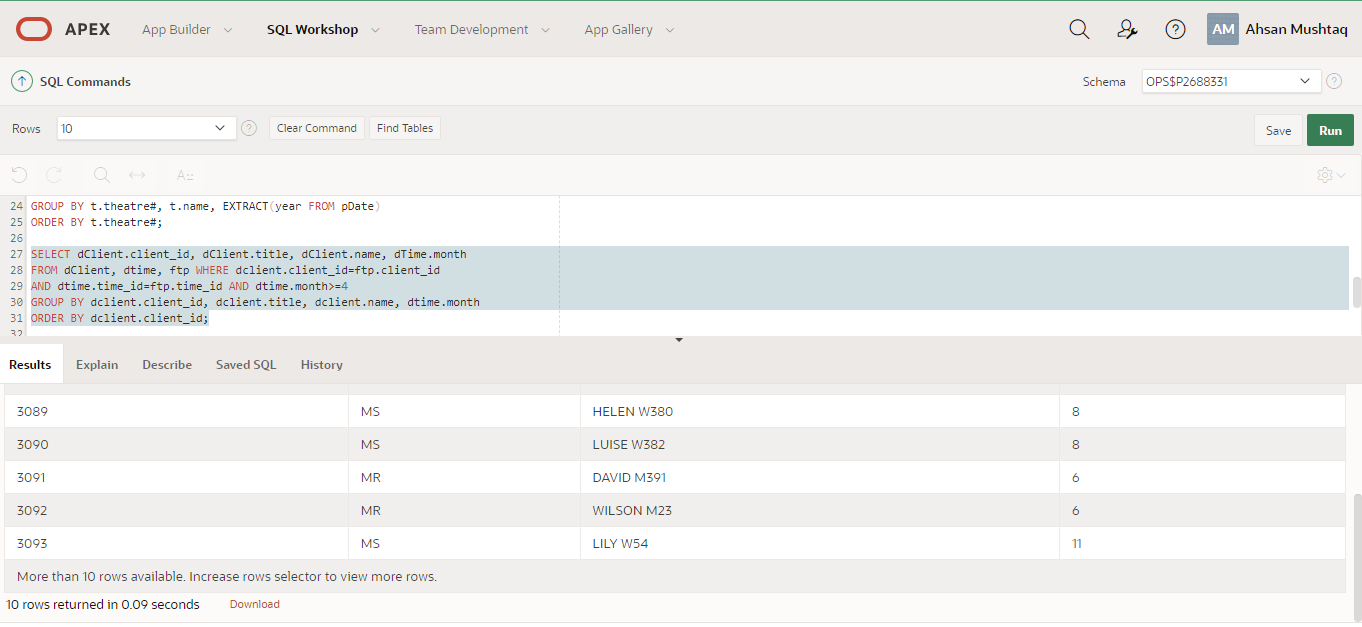
WHERE t.theatre#=p.theatre# AND p.Per#=tp.Per#

GROUP BY t.theatre#, t.name, EXTRACT(year FROM pDate)

ORDER BY t.theatre#;

* **All clients who visited MT theatres in at least 4 different months in a year.**

**Data mart Query**



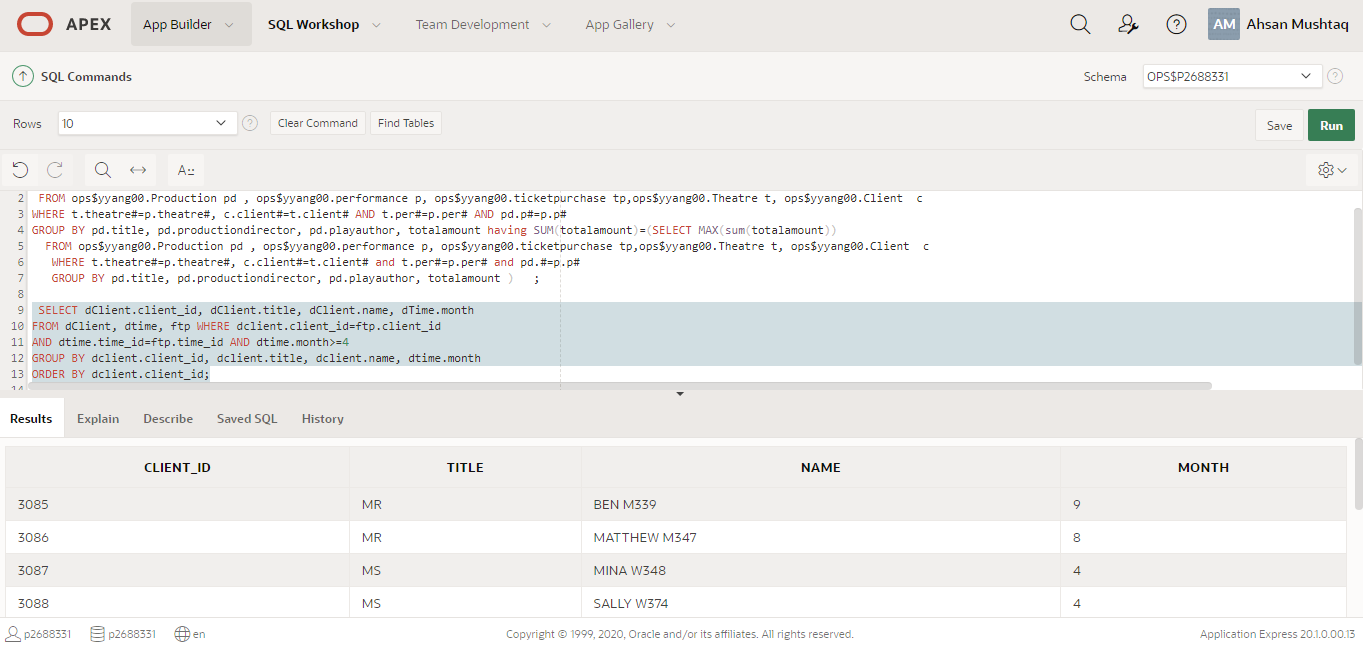
SELECT dClient.client\_id, dClient.title, dClient.name, dTime.month

FROM dClient, dtime, ftp WHERE dclient.client\_id=ftp.client\_id

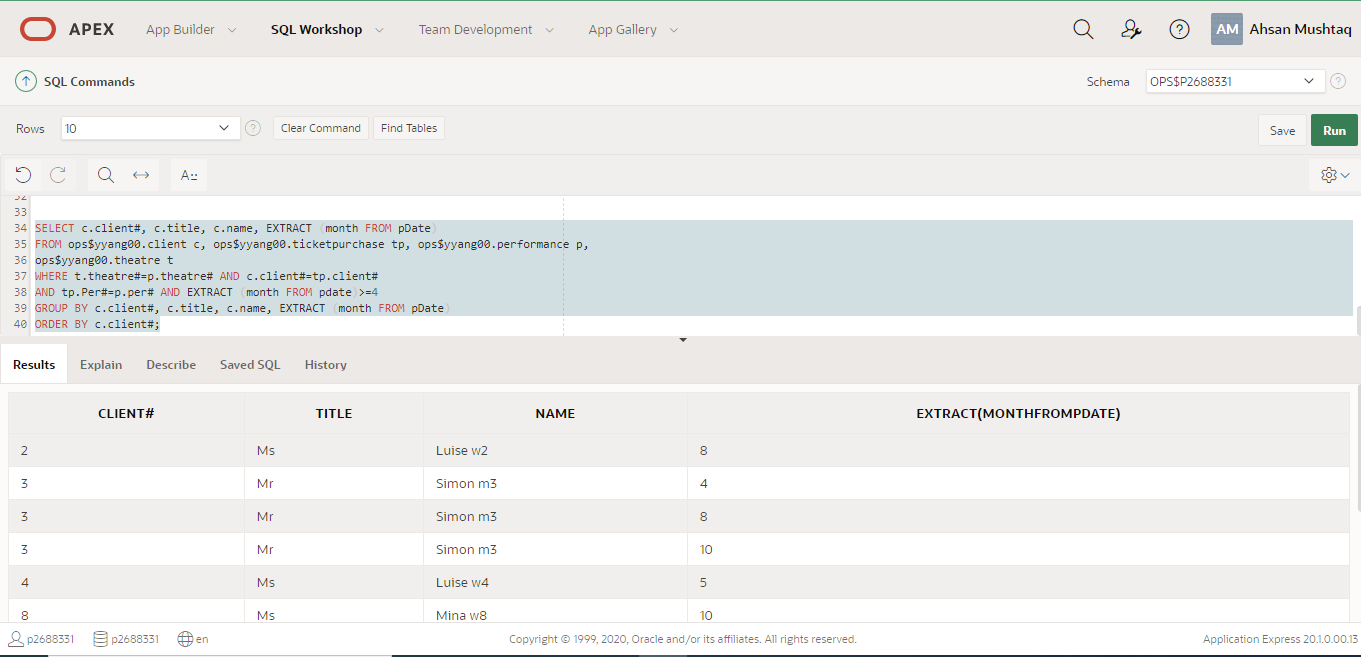
AND dtime.time\_id=ftp.time\_id AND dtime.month>=4

GROUP BY dclient.client\_id, dclient.title, dclient.name, dtime.month

ORDER BY dclient.client\_id;



* **Analysis for original OLTP Relational database**



SELECT c.client#, c.title, c.name, EXTRACT (month FROM pDate)

FROM ops$yyang00.client c, ops$yyang00.ticketpurchase tp, ops$yyang00.performance p,

ops$yyang00.theatre t

WHERE t.theatre#=p.theatre# AND c.client#=tp.client#

AND tp.Per#=p.per# AND EXTRACT (month FROM pdate)>=4

GROUP BY c.client#, c.title, c.name, EXTRACT (month FROM pDate)

ORDER BY c.client#;

* **List the titles, production directors and play authors of all products with the highest total sale.**

**Data Mart Query**

SELECT dproduction.title, dproduction.productiondirector, dproduction.playauthor, SUM(TotalAmount)

FROM dproduction ,ftp

WHERE t.theatre#=p.theatre#, c.client#=t.client# AND t.per#=p.per# AND pd.p#=p.p#

GROUP BY dproduction.title, dproduction.productiondirector, dproduction.playauthor, totalamount having SUM(totalamount)=(SELECT MAX(sum(totalamount))

FROM dproduction ,ftp

WHERE t.theatre#=p.theatre#, c.client#=t.client# and t.per#=p.per# and pd.#=p.p#

GROUP BY pd.title, pd.productiondirector, pd.playauthor, totalamount );

**Relational Query:**

SELECT pd.title,pd.productiondirector, pd.playauthor, SUM(TotalAmount)

FROM ops$yyang00.Production pd , ops$yyang00.performance p, ops$yyang00.ticketpurchase tp,ops$yyang00.Theatre t, ops$yyang00.Client c

WHERE t.theatre#=p.theatre#, c.client#=t.client# AND t.per#=p.per# AND pd.p#=p.p#

GROUP BY pd.title, pd.productiondirector, pd.playauthor, totalamount having SUM(totalamount)=(SELECT MAX(sum(totalamount))

FROM ops$yyang00.Production pd , ops$yyang00.performance p, ops$yyang00.ticketpurchase tp,ops$yyang00.Theatre t, ops$yyang00.Client c

WHERE t.theatre#=p.theatre#, c.client#=t.client# and t.per#=p.per# and pd.#=p.p#

GROUP BY pd.title, pd.productiondirector, pd.playauthor, totalamount );

**Comparison between Data Mart and relational models**

1)As we can relational databases are first step to ETL process Extraction transformation and load and the data mart are the last step in the ETL process.

2) In relational database the data is unprocessed and in the data mart it is cleaned and processed and validated for reporting.

3) A database is related to one specific subject activities and aspects on the other hand Data mart having multiple subjects

4) One big difference in that relational database is OLTP online transaction data repository While data mart is an OLAP online Analytical data repository.