Name:Kartikay Agrawal

Roll No:2148064

Exercise: Demonstration of Aggregate Functions And Views using SQL Queries.

CREATE TABLE ART\_FORM

(A\_NAME VARCHAR(20) NOT NULL CHECK(A\_NAME='JAMPADS' OR A\_NAME='PHOTOSHOOT' OR A\_NAME='STUDIO'),

A\_CODE NUMBER(1) PRIMARY KEY);

CREATE TABLE STUDIO

(S\_CODE VARCHAR(4) PRIMARY KEY,

S\_NAME VARCHAR(20),

S\_LOCATION VARCHAR(50),

S\_DURATION NUMBER(3) CHECK(S\_DURATION BETWEEN 30 AND 120),

S\_RATING NUMBER(2,1));

CREATE TABLE CUSTOMER

(C\_ID VARCHAR(4) PRIMARY KEY,

C\_NAME VARCHAR(20),

C\_EXPEREIENCE VARCHAR(10) DEFAULT 'GOOD',

STUDIO\_CODE VARCHAR(20),

ART\_CODE NUMBER(1),

FOREIGN KEY(STUDIO\_CODE) REFERENCES STUDIO(S\_CODE),

FOREIGN KEY(ART\_CODE) REFERENCES ART\_FORM(A\_CODE));

INSERT INTO ART\_FORM(A\_NAME,A\_CODE) VALUES ('JAMPADS',1);

INSERT INTO ART\_FORM(A\_NAME,A\_CODE) VALUES ('PHOTOSHOOT',2);

INSERT INTO ART\_FORM(A\_NAME,A\_CODE) VALUES ('STUDIO',3);

INSERT INTO STUDIO(S\_CODE,S\_NAME,S\_LOCATION,S\_DURATION,S\_RATING) VALUES (1432,'RocknRoll','Noida',45,4.5);

INSERT INTO STUDIO(S\_CODE,S\_NAME,S\_LOCATION,S\_DURATION,S\_RATING) VALUES (2146,'BeautynBeast','Surat',30,4);

INSERT INTO STUDIO(S\_CODE,S\_NAME,S\_LOCATION,S\_DURATION,S\_RATING) VALUES (1643,'LightsCamAction','Gurgaon',120,4.8);

INSERT INTO STUDIO(S\_CODE,S\_NAME,S\_LOCATION,S\_DURATION,S\_RATING) VALUES (1573,'LightsCamAction','Delhi',120,3);

INSERT INTO STUDIO(S\_CODE,S\_NAME,S\_LOCATION,S\_DURATION,S\_RATING) VALUES (1284,'Jamway','Bangalore',120,4.2);

INSERT INTO STUDIO(S\_CODE,S\_NAME,S\_LOCATION,S\_DURATION,S\_RATING) VALUES (1393,'DharmaProd','Mumbai',120,5);

INSERT INTO CUSTOMER(ART\_CODE,C\_ID,C\_NAME,C\_EXPEREIENCE,STUDIO\_CODE) VALUES (1,'J123','Pankaj','Excellent',1432);

INSERT INTO CUSTOMER(ART\_CODE,C\_ID,C\_NAME,C\_EXPEREIENCE,STUDIO\_CODE) VALUES (2,'P342','Uday','Bad',2146);

INSERT INTO CUSTOMER(ART\_CODE,C\_ID,C\_NAME,STUDIO\_CODE) VALUES (3,'S143','Shrey',2648);

INSERT INTO CUSTOMER(ART\_CODE,C\_ID,C\_NAME,STUDIO\_CODE) VALUES (3,'S144','Jimmy',1393);

INSERT INTO CUSTOMER(ART\_CODE,C\_ID,C\_NAME,STUDIO\_CODE) VALUES (2,'P443','Sasha',6618);

INSERT INTO CUSTOMER(ART\_CODE,C\_ID,C\_NAME,STUDIO\_CODE) VALUES (3,'S145','Shabir',2648);

Perform the following tasks:

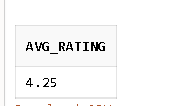
Q1. Write SQL statements to demonstrate the following Simple Aggregate functions on the

tables created for your Application domain.

AVG, SUM, COUNT, MIN, MAX, MEDIAN, CORR and STDDEV

**TO FIND THE AVERAGE RATING OF THE REGISTERED STUDIOS**

SELECT AVG(S\_RATING) AS AVG\_RATING FROM STUDIO;



**TO FIND THE TOTAL MAXIMUM DURATION OF ALL THE REGISTERED STUDIOS AFTER A SINGLE DAY OF BOOKING**

SELECT SUM(S\_DURATION) AS TOTAL\_DURATION FROM STUDIO;



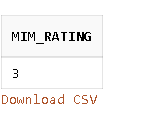
**TO FIND THE NUMBER OF DIFFERENT STUDIOS OPTED BY CUSTOMERS**

SELECT COUNT(STUDIO\_CODE) AS OPTED\_STUDIOS FROM CUSTOMER;

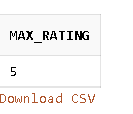


**TO FIND THE MINIMUM,MAXIMUM,MEDIAN AND STD DEV OF ALL THE RATINGS OF STUDIOS FOR ANALYSIS AND IMPROVEMENT IN THE BUSINESS CASE**

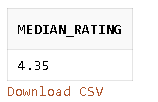
SELECT MIN(S\_RATING) AS MIN\_RATING FROM STUDIO;



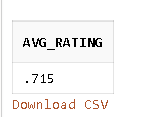
SELECT MAX(S\_RATING) AS MAX\_RATING FROM STUDIO;



SELECT MEDIAN(S\_RATING) AS MEDIAN\_RATING FROM STUDIO;



SELECT ROUND(STDDEV(S\_RATING),3) AS AVG\_RATING FROM STUDIO;



**TO FIND THE POSSIBILITY OF ANY DEPENDENCY BETWEEN RATINGS AND DURATION TIME IN A STUDIO**

SELECT CORR(S\_RATING,S\_DURATION) AS CORR\_B/W\_RATING\_N\_DUR FROM STUDIO;



Q2. Demonstrate the usage of GROUPBY and HAVING clause in compound SQL

statements using Aggregate functions on the tables created for your Application domain.

**TO FIND THE MOST FAMOUS STUDIO ALONG WITH THE ART FORM IT IS LINKED WITH**

SELECT COUNT(STUDIO\_CODE),ART\_CODE FROM CUSTOMER GROUP BY ART\_CODE HAVING COUNT(STUDIO\_CODE)>1;



Q3. Demonstrate the creation and processing of SQL Views on the tables created for your

Application Domain.

Specification:

a. The Operations should be performed on tables with logical sense and mention

valid justification.

b. At least 1 Compound SQL Query with a suitable aggregate function should be

demonstrated for all the Tables

c. At least 3 views should be created which holds the records from single table and

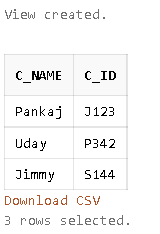
multiple tables.

d. Perform the insert, delete and update operations on the Views. State the

inferences.

CREATE VIEW CUSTOMERS AS SELECT C\_NAME,C\_ID FROM CUSTOMER;

SELECT \* FROM CUSTOMERS;

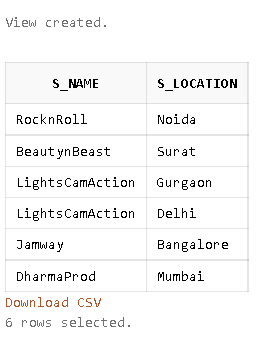


INSERT INTO CUSTOMERS(C\_NAME,C\_ID) VALUES('AJAY','S123');

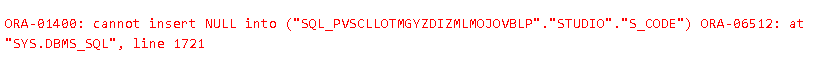


CREATE VIEW STUDIOS AS SELECT S\_NAME,S\_LOCATION FROM STUDIO;

SELECT \* FROM STUDIOS;



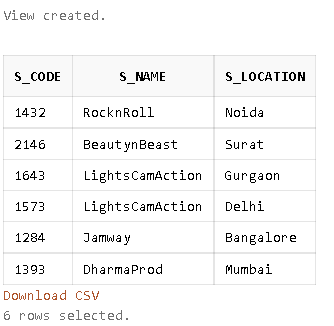
INSERT INTO STUDIOS(S\_NAME,S\_LOCATION) VALUES ('DUNK','GURGAON')



***Since the view doesn’t have a primary key we cannot insert values into it.Hence we’ll have to make a view with the primary key.***

CREATE VIEW STUDIOS\_IMP AS SELECT S\_CODE,S\_NAME,S\_LOCATION FROM STUDIO;

SELECT \* FROM STUDIOS\_IMP;



INSERT INTO STUDIOS\_IMP(S\_CODE,S\_NAME,S\_LOCATION) VALUES (3253,'DUNK','GURGAON')

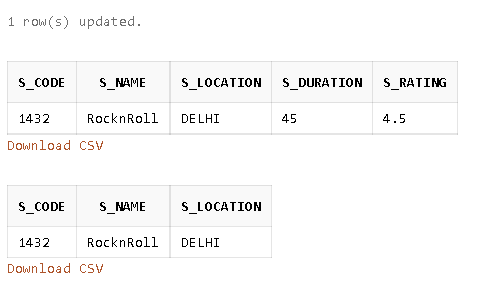


**CHANGING THE LOCATION OF STUDIO WITH CODE 1432 DUE TO SHIFTING**

UPDATE STUDIOS\_IMP SET S\_LOCATION='DELHI' WHERE S\_CODE=1432;

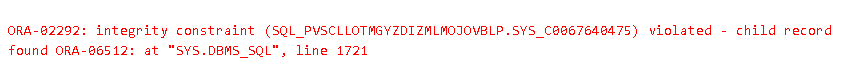
SELECT \* FROM STUDIO WHERE S\_CODE=1432;

SELECT \* FROM STUDIOS\_IMP WHERE S\_CODE=1432;



***THE ROWS GET UPDATED IN BOTH TABLE AND VIEW***

DELETE FROM STUDIOS\_IMP WHERE S\_CODE=1393;



***We cannot delete an item from the table/view by deleting it from the view as they have a parent child relation.It will get deleted only when we delete it from the parent table.***

**COMBINIG AND COMPARING THE CUSTOMER EXPEREINCE AND STUDIO RATINGS**

CREATE VIEW FEEDBACK AS(SELECT CUSTOMER.C\_EXPEREIENCE,STUDIO.S\_RATING FROM CUSTOMER INNER JOIN STUDIO ON S\_CODE=STUDIO\_CODE);

SELECT \* FROM FEEDBACK;

