Student#: 215800493
Sur (Family) Name: Sayal
Given Name: Jaleel Ahmad
Class: EECS-3421B
Term: Fall 2019
Project: SQL QUERIES

--Question 1

SELECT CID, NAME,CITY FROM YRB_CUSTOMER WHERE NAME LIKE '%a%e'

--Question 2

SELECT CATEGORY, MAX(PQ) AS MAX FROM (SELECT TEMP.CAT AS CATEGORY, COUNT(P.QNTY) AS PQ
FROM YRB_PURCHASE P, (SELECT O.YEAR, O.TITLE, O.CLUB, B.CAT AS CAT FROM YRB_BOOK B INNER
JOIN YRB_OFFER O ON B.TITLE = O.TITLE AND B.YEAR = O.YEAR) AS TEMP WHERE P.CLUB = TEMP.CLUB
AND P.TITLE = TEMP.TITLE AND P.YEAR = TEMP.YEAR GROUP BY TEMP.CAT) GROUP BY CATEGORY
ORDER BY MAX DESC LIMIT 1;

--Question 3

SELECT VIEW.CID, VIEW.NAME, YRB_MEMBER.CLUB FROM (SELECT FU2.CID, FU2.NAME, FU2.NUMCLUB
FROM (SELECT FU.CID, FU.NAME, COUNT(*) AS NUMCLUB FROM (SELECT YRB_CUSTOMER.CID,
YRB_CUSTOMER.NAME, TEMP.CLUB FROM (SELECT YRB_CLUB.CLUB, YRB_MEMBER.CID FROM
YRB_CLUB INNER JOIN YRB_MEMBER ON YRB_CLUB.CLUB = YRB_MEMBER.CLUB) AS TEMP,
YRB_CUSTOMER WHERE YRB_CUSTOMER.CID = TEMP.CID) AS FU GROUP BY FU.CID, FU.NAME) AS FU2
WHERE FU2.NUMCLUB <= 2) AS VIEW INNER JOIN YRB_MEMBER ON VIEW.CID = YRB_MEMBER.CID
ORDER BY VIEW.CID;

--Question 4

SELECT YRB_CATEGORY.CAT FROM YRB_CATEGORY MINUS SELECT JUSTCAT.CAT FROM (SELECT C.CID,
C.TITLE, C.YEAR, B.CAT FROM (SELECT DISTINCT P.CID, O.TITLE, O.YEAR FROM (SELECT P.CID, P.TITLE,
P.YEAR FROM YRB_PURCHASE AS P WHERE P.CID = 9) AS P INNER JOIN YRB_OFFER AS O ON P.TITLE =
O.TITLE AND P.YEAR = O.YEAR) AS C INNER JOIN YRB_BOOK AS B ON C.TITLE = B.TITLE AND C.YEAR =
B.YEAR) AS JUSTCAT;

--Question 5

SELECT * FROM YRB_OFFER WHERE PRICE=(SELECT MIN(PRICE) FROM YRB_OFFER) UNION SELECT *
FROM YRB_OFFER WHERE PRICE=(SELECT MAX(PRICE) FROM YRB_OFFER);

--Question 6

SELECT TEMP.CID, TEMP.WHEN, TEMP.TOTAL_WEIGHT, MIN(TEMP.COST) AS COST, (MIN(TEMP.COST) *
TEMP.TOTAL_WEIGHT) AS TOTAL_COST FROM (SELECT * FROM (SELECT P.CID, P.WHEN,
P.TOTAL_WEIGHT, S.COST, S.WEIGHT FROM (SELECT P.CID, P.WHEN, SUM(P.TOTAL_WEIGHT) AS

TOTAL_WEIGHT FROM (SELECT P.CID, P.TITLE, P.YEAR, P.WHEN, (P.QNTY * B.WEIGHT) AS TOTAL_WEIGHT FROM YRB_PURCHASE AS P INNER JOIN YRB_BOOK AS B ON P.TITLE = B.TITLE AND P.YEAR = B.YEAR) AS P GROUP BY P.CID, P.WHEN ORDER BY P.CID) AS P CROSS JOIN YRB_SHIPPING AS S) AS TEMP WHERE (TEMP.WEIGHT - TEMP.TOTAL_WEIGHT) > 0) AS TEMP GROUP BY TEMP.CID, TEMP.WHEN, TEMP.TOTAL_WEIGHT;

--Question 7

SELECT BOOK.TITLE, BOOK.YEAR, B.YEAR FROM YRB_BOOK AS BOOK, YRB_BOOK AS B WHERE BOOK.TITLE = B.TITLE AND BOOK.YEAR < B.YEAR;

--Question 8

SELECT O.CID, O.NAME, O.NUMBER_OF_PURCHASES FROM (SELECT * FROM (SELECT A.CID, A.NAME, A.NUMBER_OF_PURCHASES, 1 AS ONE FROM (SELECT C.CID, C.NAME, P.NUMBER_OF_PURCHASES FROM (SELECT CID, COUNT(CID) AS NUMBER_OF_PURCHASES FROM YRB_PURCHASE GROUP BY CID) AS P FULL JOIN YRB_CUSTOMER AS C ON P.CID = C.CID WHERE P.NUMBER_OF_PURCHASES IS NOT NULL ORDER BY P.NUMBER_OF_PURCHASES) AS A) UNION ALL (SELECT C.CID, C.NAME, COALESCE(P.NUMBER_OF_PURCHASES, 0) AS NUMBER_OF_PURCHASES, 2 AS ONE FROM (SELECT CID, COUNT(CID) AS NUMBER_OF_PURCHASES FROM YRB_PURCHASE GROUP BY CID) AS P FULL JOIN YRB_CUSTOMER AS C ON P.CID = C.CID WHERE P.NUMBER_OF_PURCHASES IS NULL ORDER BY C.NAME) ORDER BY ONE) AS O;

--Question 9

SELECT OUT.CLUB, OUT.NUMBER_OF_OFFERS, OUT.TOTAL_PRICE FROM (SELECT CLUB, COUNT(CLUB) AS NUMBER_OF_OFFERS, SUM(PRICE) as TOTAL_PRICE, AVG(PRICE) AS AVG_PRICE FROM YRB_OFFER GROUP BY CLUB) AS OUT WHERE OUT.AVG_PRICE > (SELECT AVG(A.AVG_PRICE) FROM (SELECT CLUB, COUNT(CLUB) AS NUMBER_OF_OFFERS, SUM(PRICE) as TOTAL_PRICE, AVG(PRICE) AS AVG_PRICE FROM YRB_OFFER GROUP BY CLUB) AS A);

--Question 10

SELECT * FROM (SELECT Y.CID,Y.NAME,SUM(Y.PRICE) AS TOTAL_PRICE FROM(SELECT R.CID, R.CLUB,R.YEAR,R.QNTY,R.TITLE,R.PRICE,C.NAME FROM(SELECT J.CID, J.CLUB,J.YEAR,J.QNTY,J.TITLE,O.PRICE FROM (SELECT P.CID, P.CLUB,P.YEAR,P.QNTY,P.TITLE FROM YRB_PURCHASE AS P INNER JOIN YRB_MEMBER AS M ON P.CID=M.CID AND P.CLUB=M.CLUB)AS J INNER JOIN YRB_OFFER AS O ON J.CLUB=O.CLUB AND J.TITLE=O.TITLE AND J.YEAR=O.YEAR) AS R INNER JOIN YRB_CUSTOMER AS C ON R.CID=C.CID) AS Y GROUP BY Y.CID,Y.NAME) AS X WHERE X.TOTAL_PRICE > 300 ORDER BY X.TOTAL_PRICE DESC;