UMB Database Analyst Test

Name: Nikhil Sai Santosh Gurram

Email: ng3qn@mail.umkc.edu / gurramsaisantosh@gmail.com

Contact #: 816-529-4015

Question 1 - Select details of all patients aged less than 18 years on the visit date who were seen in more than one department in calendar year 2017.
-- The result set should have ONLY one record for each patient and department combination.

File: Answer1.sql

Code:

```
select distinct TBL_TRANSACTIONS.PATIENT_ID,
 TBL_PATIENTS.PATIENT_NAME,
  TBL_PATIENTS.BIRTH_DATE,
  TBL_DEPARTMENTS.DEPT_NAME,
  COUNT(TBL_TRANSACTIONS.VISIT_DATE) AS VISIT_COUNT
-- joining four tables together as inner join to get intersecting values in all tables
because we want to see columns from each table
 from
 (((TBL_TRANSACTIONS
 inner join TBL_PATIENTS on TBL_TRANSACTIONS.PATIENT_ID = TBL_PATIENTS.PATIENT_ID)
 inner join TBL_PHYSICIANS on TBL_TRANSACTIONS.PHYSICIAN_ID =
TBL_PHYSICIANS.PHYSICIAN_ID)
 inner join TBL_DEPARTMENTS on TBL_PHYSICIANS.DEPT_ID = TBL_DEPARTMENTS.DEPT_ID)

 where
-- Drilling down to rows with patients who are below 18 during visit
 DATEDIFF(year, BIRTH_DATE, VISIT_DATE) < 18

 and
-- Drilling down 2017 data
 (TBL_TRANSACTIONS.VISIT_DATE >= Convert(datetime, '2017-01-01') and
 TBL_TRANSACTIONS.VISIT_DATE <= Convert(datetime, '2017-12-31'))

 and

 PATIENT_NAME in (
-- selecting names of patients who are under 18 and visited multiple departments in 2017
select TBL_PATIENTS.PATIENT_NAME
-- joining four tables together as inner join to get intersecting values in all tables
because we want to see columns from each tables
from
 (((TBL_TRANSACTIONS
 inner join TBL_PATIENTS on TBL_TRANSACTIONS.PATIENT_ID = TBL_PATIENTS.PATIENT_ID)
 inner join TBL_PHYSICIANS on TBL_TRANSACTIONS.PHYSICIAN_ID =
TBL_PHYSICIANS.PHYSICIAN_ID)
 inner join TBL_DEPARTMENTS on TBL_PHYSICIANS.DEPT_ID = TBL_DEPARTMENTS.DEPT_ID)
 -- Drilling down 2017 data
 where
 (TBL_TRANSACTIONS.VISIT_DATE >= Convert(datetime, '2017-01-01') and
 TBL_TRANSACTIONS.VISIT_DATE <= Convert(datetime, '2017-12-31'))
 -- Drilling down to rows with patients who are below 18 during visit
  and
 DATEDIFF(year, BIRTH_DATE, VISIT_DATE) < 18

 -- grouping for getting departments visited count for each patient
 group by PATIENT_NAME
 -- Filtering list of people who have visited multiple departments in 2017 with age less
than 18
 having COUNT(distinct (TBL_PHYSICIANS.DEPT_ID)) > 1
)
-- grouping for getting visit count for each patient for each department
GROUP BY TBL_TRANSACTIONS.PATIENT_ID,
 TBL_PATIENTS.PATIENT_NAME,
  TBL_PATIENTS.BIRTH_DATE,
  TBL_DEPARTMENTS.DEPT_NAME
-- just for convenience - optional
order by PATIENT_ID
```

Output:

PATIENT_ID	PATIENT_NAME	BIRTH_DATE	DEPT_NAME	VISIT_COUNT
204	KIM BOWERS	9/2/2001	CARDIOTHORACIC SURGERY	1
204	KIM BOWERS	9/2/2001	GENERAL SURGERY	1
204	KIM BOWERS	9/2/2001	ORTHOPAEDICS	2
204	KIM BOWERS	9/2/2001	PEDIATRICS	3
204	KIM BOWERS	9/2/2001	PLASTIC SURGERY	1
206	CARL MAGNUS	10/18/2012	INTERNAL MEDICINE	1
206	CARL MAGNUS	10/18/2012	ORTHOPAEDICS	1
213	TODD SWEENEY	5/12/2005	ORTHOPAEDICS	1
213	TODD SWEENEY	5/12/2005	PLASTIC SURGERY	1
216	TRUMAN CAMPBELL	6/15/2016	OTOLOARYNGOLOGY	1
216	TRUMAN CAMPBELL	6/15/2016	PEDIATRICS	1
223	CHARLOTTE GRAHAM	5/11/2015	OTOLOARYNGOLOGY	1
223	CHARLOTTE GRAHAM	5/11/2015	PEDIATRICS	3
224	PAUL MUELLER	9/5/2009	OTOLOARYNGOLOGY	1
224	PAUL MUELLER	9/5/2009	PEDIATRICS	2
226	WILLIAM HARRIS	3/14/2006	ORTHOPAEDICS	2
226	WILLIAM HARRIS	3/14/2006	PEDIATRICS	1

Question -2:

Select details of all patients with outstanding balance (CHARGES > INSURANCE_PAYMENT + PATIENT_PAYMENT) in each department, sorted DESCENDING by outstanding balance. Display Insurance as 'SELF-PAY' where INSURANCE_ID is blank.

Output fields: DEPT_NAME PATIENT_ID PATIENT_NAME INSURANCE_NAME OUTSTANDING_BALANCE

File: Answer2.sql

Code:

```
select DEPT_NAME, TBL_TRANSACTIONS.PATIENT_ID, PATIENT_NAME,
-- replace insurance name with SELF-PAY when its null
ISNULL(TBL_INSURANCES.INSURANCE_NAME, 'SELF-PAY') AS INSURANCE_NAME,
-- replace NULL insurance payment to 0
SUM(CHARGES - (ISNULL(TBL_TRANSACTIONS.INSURANCE_PAYMENT, 0) + ISNULL(PATIENT_PAYMENT,
0)) ) AS OUTSTANDING_BALANCE

from
-- inner join transactions and patients
((((TBL_TRANSACTIONS
 inner join TBL_PATIENTS on TBL_TRANSACTIONS.PATIENT_ID = TBL_PATIENTS.PATIENT_ID)
-- add inner join physicians
 inner join TBL_PHYSICIANS on TBL_TRANSACTIONS.PHYSICIAN_ID =
TBL_PHYSICIANS.PHYSICIAN_ID)
-- add inner join departments
 inner join TBL_DEPARTMENTS on TBL_PHYSICIANS.DEPT_ID = TBL_DEPARTMENTS.DEPT_ID)
-- left outer join insurances to also get the patients with no insurance id's into the
result
 left outer join TBL_INSURANCES on TBL_PATIENTS.INSURANCE_ID =
TBL_INSURANCES.INSURANCE_ID)

 -- perform analysis on the transactions that are only outstanding
where TBL_TRANSACTIONS.TX_ID IN (SELECT TX_ID FROM TBL_TRANSACTIONS
where
(TBL_TRANSACTIONS.CHARGES > ISNULL(TBL_TRANSACTIONS.INSURANCE_PAYMENT, 0) +
TBL_TRANSACTIONS.PATIENT_PAYMENT)
)
-- grouping due to details organized by department are required
GROUP BY DEPT_NAME, TBL_TRANSACTIONS.PATIENT_ID, PATIENT_NAME,
TBL_INSURANCES.INSURANCE_NAME
-- showing results in reverse order of outstanding balance in each department
ORDER BY DEPT_NAME ,OUTSTANDING_BALANCE DESC
```

Output: Next Page

DEPT_NAME	PATIENT_ID	PATIENT_NAME	INSURANCE_NAME	OUTSTANDING_BALANCE
CARDIOTHORACIC SURGERY	204	KIM BOWERS	UNITED HEALTH CARE	1430
CARDIOTHORACIC SURGERY	200	BENJAMIN LINCOLN	AETNA	1069
CARDIOTHORACIC SURGERY	225	STEFANIE SMITH	SELF-PAY	946
CARDIOTHORACIC SURGERY	228	HENRY HAWKING	SELECT HEALTH	419
CARDIOTHORACIC SURGERY	223	CHARLOTTE GRAHAM	AETNA	380
CARDIOTHORACIC SURGERY	213	TODD SWEENEY	MEDICARE	127
CARDIOTHORACIC SURGERY	224	PAUL MUELLER	SELECT HEALTH	10
GENERAL SURGERY	201	AARON GRAFF	BLUE CROSS BLUE SHIELD	1237
GENERAL SURGERY	205	SARAH ARMSTRONG	SELF-PAY	1056
GENERAL SURGERY	219	JAMES PITT	MEDICAID UTAH	968
GENERAL SURGERY	204	KIM BOWERS	UNITED HEALTH CARE	726
GENERAL SURGERY	229	JAMES FLEMING	BLUE CROSS BLUE SHIELD	652
GENERAL SURGERY	222	CAROLINE CRAIG	CIGNA	480
GENERAL SURGERY	230	RICHARD CASEY	BLUE CROSS BLUE SHIELD	448
GENERAL SURGERY	200	BENJAMIN LINCOLN	AETNA	284
GENERAL SURGERY	210	PETER SUTTERBERG	SELECT HEALTH MEDICARE	62
GENERAL SURGERY	203	ANDREW ROGERS	MEDICAID UTAH	59
INTERNAL MEDICINE	213	TODD SWEENEY	MEDICARE	1628
INTERNAL MEDICINE	203	ANDREW ROGERS	MEDICAID UTAH	1176
INTERNAL MEDICINE	214	JACOB MUELLER	UNITED HEALTH CARE	692
INTERNAL MEDICINE	221	MELISSA ANDREWS	SELECT HEALTH MEDICARE	654
INTERNAL MEDICINE	215	STACY NGUYEN	MEDICARE	563
INTERNAL MEDICINE	206	CARL MAGNUS	SELECT HEALTH MEDICAID	522
INTERNAL MEDICINE	201	AARON GRAFF	BLUE CROSS BLUE SHIELD	330
INTERNAL MEDICINE	218	HENRIETTA BILLINGSLEY	SELECT HEALTH	221
INTERNAL MEDICINE	229	JAMES FLEMING	BLUE CROSS BLUE SHIELD	140
INTERNAL MEDICINE	228	HENRY HAWKING	SELECT HEALTH	94
INTERNAL MEDICINE	227	LISA HOLDEN	CIGNA	59
INTERNAL MEDICINE	211	MELISSA MANHEIM	AETNA MEDICARE	52
ORTHOPAEDICS	209	CATHERINE LEWIS	SELF-PAY	953
ORTHOPAEDICS	225	STEFANIE SMITH	SELF-PAY	851
ORTHOPAEDICS	227	LISA HOLDEN	CIGNA	672
ORTHOPAEDICS	204	KIM BOWERS	UNITED HEALTH CARE	669
ORTHOPAEDICS	205	SARAH ARMSTRONG	SELF-PAY	586

ORTHOPAEDICS	208	PATRICK FULLER	CIGNA	518
ORTHOPAEDICS	226	WILLIAM HARRIS	BLUE CROSS BLUE SHIELD	516
ORTHOPAEDICS	218	HENRIETTA BILLINGSLEY	SELECT HEALTH	467
ORTHOPAEDICS	206	CARL MAGNUS	SELECT HEALTH MEDICAID	388
ORTHOPAEDICS	213	TODD SWEENEY	MEDICARE	297
OTOLOARYNGOLOGY	224	PAUL MUELLER	SELECT HEALTH	613
OTOLOARYNGOLOGY	211	MELISSA MANHEIM	AETNA MEDICARE	417
OTOLOARYNGOLOGY	215	STACY NGUYEN	MEDICARE	403
OTOLOARYNGOLOGY	223	CHARLOTTE GRAHAM	AETNA	256
OTOLOARYNGOLOGY	217	TAYLOR BARRYMORE	AETNA MEDICARE	244
OTOLOARYNGOLOGY	206	CARL MAGNUS	SELECT HEALTH MEDICAID	165
OTOLOARYNGOLOGY	207	TIMOTHY BRUNSWICK	MEDICAID UTAH	97
OTOLOARYNGOLOGY	221	MELISSA ANDREWS	SELECT HEALTH MEDICARE	68
PEDIATRICS	224	PAUL MUELLER	SELECT HEALTH	3406
PEDIATRICS	204	KIM BOWERS	UNITED HEALTH CARE	2226
PEDIATRICS	208	PATRICK FULLER	CIGNA	1974
PEDIATRICS	226	WILLIAM HARRIS	BLUE CROSS BLUE SHIELD	1945
PEDIATRICS	223	CHARLOTTE GRAHAM	AETNA	1602
PEDIATRICS	214	JACOB MUELLER	UNITED HEALTH CARE	1499
PEDIATRICS	220	KELSEY RICHARDS	SELECT HEALTH MEDICAID	1146
PEDIATRICS	206	CARL MAGNUS	SELECT HEALTH MEDICAID	685
PEDIATRICS	213	TODD SWEENEY	MEDICARE	87
PEDIATRICS	216	TRUMAN CAMPBELL	AETNA	86
PLASTIC SURGERY	206	CARL MAGNUS	SELECT HEALTH MEDICAID	1142
PLASTIC SURGERY	220	KELSEY RICHARDS	SELECT HEALTH MEDICAID	876
PLASTIC SURGERY	207	TIMOTHY BRUNSWICK	MEDICAID UTAH	525
PLASTIC SURGERY	224	PAUL MUELLER	SELECT HEALTH	522
PLASTIC SURGERY	204	KIM BOWERS	UNITED HEALTH CARE	449
PLASTIC SURGERY	212	CAROL STEWART	AETNA	425
PLASTIC SURGERY	217	TAYLOR BARRYMORE	AETNA MEDICARE	391
PLASTIC SURGERY	202	STEPHEN GRANT	SELECT HEALTH	364
PLASTIC SURGERY	218	HENRIETTA BILLINGSLEY	SELECT HEALTH	178
PLASTIC SURGERY	208	PATRICK FULLER	CIGNA	159

3. Select all transactions with visit dates in calendar year 2016 and 2017. Display Insurance as 'SELFPAY' where INSURANCE_ID is blank. Present the information in a pivot table organized by visit month and financial class. Display the total payment (Insurance Payment + Patient Payment) values in the pivot table.

Version 1: question says pivot table organized by visit_month but, it does not say month for each year or 2016 and 2017 together. So, the first version of code gives results of pivot table organized by the months together.

File: Answer3.sql

Code:

```
USE DBA_TEST
-- Sort the pivot table by Month and then by different financial classes
SELECT MONTH,
[COMMERCIAL], [MEDICARE], [MEDICAID], [SELF_PAYMENT]
FROM(
-- Select the specific month name from given visit date, financial class of insurance (
Self_payment when null)
-- and insurance payment + patient_payment as total_payment(0 when null)
select DATENAME(MONTH, VISIT_DATE)as MONTH, ISNULL(FINANCIAL_CLASS, 'SELF_PAYMENT') as
FINANCIAL_CLASS,
ISNULL(TBL_TRANSACTIONS.INSURANCE_PAYMENT, 0) + ISNULL(PATIENT_PAYMENT, 0) AS
TOTAL_PAYMENT



from
-- inner join transactions and patients
((((TBL_TRANSACTIONS
 inner join TBL_PATIENTS on TBL_TRANSACTIONS.PATIENT_ID = TBL_PATIENTS.PATIENT_ID)
-- add inner join physicians
 inner join TBL_PHYSICIANS on TBL_TRANSACTIONS.PHYSICIAN_ID =
TBL_PHYSICIANS.PHYSICIAN_ID)
-- add inner join departments
 inner join TBL_DEPARTMENTS on TBL_PHYSICIANS.DEPT_ID = TBL_DEPARTMENTS.DEPT_ID)
-- left outer join insurances to also get the patients with no insurance id's into the
result
 left outer join TBL_INSURANCES on TBL_PATIENTS.INSURANCE_ID =
TBL_INSURANCES.INSURANCE_ID)

-- Select transactions from only 2016 and 2017
where TX_ID in (
select TX_ID from TBL_TRANSACTIONS where VISIT_DATE between '2016-01-01' and '2017-12-31'
)
-- Consider this as Source (SRC)
 ) AS SRC

-- pivot sum(total payment) for a purticular financial class and declaring it as pivot
variable (pvt)
PIVOT
(
SUM(TOTAL_PAYMENT)
FOR [FINANCIAL_CLASS] IN ([COMMERCIAL], [MEDICARE], [MEDICAID], [SELF_PAYMENT])
) AS PVT

-- order by occuring month names for convenience
ORDER BY (
     case MONTH
     when 'January' then 0
     when 'February' then 1
     when 'March' then 2
     when 'April' then 3
        when 'May' then 4
        when 'June' then 5
        when 'July' then 6
        when 'August' then 7
        when 'September' then 8
        when 'October' then 9
        when 'November' then 10
        when 'December' then 11
     end
)
```

Output:

```
MONTH         COMMERCIAL    MEDICARE    MEDICAID    SELF_PAYMENT
January       6728          1067        832         365
February      3683          278         NULL        634
March         5549          NULL        3469        703
April         2886          2489        1680        NULL
May           4817          3784        1754        NULL
June          7930          1284        1959        NULL
July          2947          1757        2364        934
August        4089          529         1123        NULL
September     6847          2791        710         NULL
October       4050          25966       1052        474
November      5748          81009       1265        NULL
December      7275          31824       NULL        NULL
```

Version 2 : Organizes the result by the visit month of each year

File: Answer 3_2.sql

Code:

```
USE DBA_TEST
-- Sort the pivot table by Month and then by different financial classes
SELECT MONTH,
[COMMERCIAL], [MEDICARE], [MEDICAID], [SELF_PAYMENT]
FROM(
-- Select the specific month name and year from given visit date, financial class of
insurance ( Self_payment when null)
-- and insurance payment + patient_payment as total_payment(0 when null)
```

```sql
select CAST(DATENAME(MONTH, VISIT_DATE)  as varchar) + ' ' + CAST(DATEPART(YEAR,
VISIT_DATE) as varchar) as MONTH,
ISNULL(FINANCIAL_CLASS, 'SELF_PAYMENT') as FINANCIAL_CLASS,
ISNULL(TBL_TRANSACTIONS.INSURANCE_PAYMENT, 0) + ISNULL(PATIENT_PAYMENT, 0) AS
TOTAL_PAYMENT



from
-- inner join transactions and patients
((((TBL_TRANSACTIONS
 inner join TBL_PATIENTS on TBL_TRANSACTIONS.PATIENT_ID = TBL_PATIENTS.PATIENT_ID)
-- add inner join physicians
 inner join TBL_PHYSICIANS on TBL_TRANSACTIONS.PHYSICIAN_ID =
TBL_PHYSICIANS.PHYSICIAN_ID)
-- add inner join departments
 inner join TBL_DEPARTMENTS on TBL_PHYSICIANS.DEPT_ID = TBL_DEPARTMENTS.DEPT_ID)
-- left outer join insurances to also get the patients with no insurance id's into the
result
 left outer join TBL_INSURANCES on TBL_PATIENTS.INSURANCE_ID =
TBL_INSURANCES.INSURANCE_ID)

-- Select transactions from only 2016 and 2017
where TX_ID in (
select TX_ID from TBL_TRANSACTIONS where VISIT_DATE between '2016-01-01' and '2017-12-31'
)
-- Consider this as Source (SRC)
 ) AS SRC

-- pivot sum(total payment) for a purticular financial class and declaring it as pivot
variable (pvt)
PIVOT
(
SUM(TOTAL_PAYMENT)
FOR [FINANCIAL_CLASS] IN ([COMMERCIAL], [MEDICARE], [MEDICAID], [SELF_PAYMENT])
) AS PVT

-- order by occuring month names for convenience
ORDER BY (
     case MONTH
     when 'January 2016' then 0
     when 'February 2016' then 1
     when 'March 2016' then 2
     when 'April 2016' then 3
        when 'May 2016' then 4
        when 'June 2016' then 5
        when 'July 2016' then 6
        when 'August 2016' then 7
        when 'September 2016' then 8
        when 'October 2016' then 9
        when 'November 2016' then 10
        when 'December 2016' then 11
        when 'January 2017' then 12
     when 'February 2017' then 13
     when 'March 2017' then 14
     when 'April 2017' then 15
        when 'May 2017' then 16
        when 'June 2017' then 17
```

```
            when 'July 2017' then 18
            when 'August 2017' then 19
            when 'September 2017' then 20
            when 'October 2017' then 21
            when 'November 2017' then 22
            when 'December 2017' then 23
        end
)
```

Output:

```
      MONTH         COMMERCIAL          MEDICARE        MEDICAID        SELF_PAYMENT
      Jan-16        4424                1067            832             NULL
      Feb-16        1221                278             NULL            NULL
      Mar-16        1276                NULL            2822            703
      Apr-16        NULL                846             1680            NULL
      May-16        2667                2680            1123            NULL
      Jun-16        3073                NULL            656             NULL
      Jul-16        1172                1757            NULL            NULL
      Sep-16        2771                2791            710             NULL
      Oct-16        1456                3341            1052            NULL
      Nov-16        3777                NULL            NULL            NULL
      Dec-16        5644                911             NULL            NULL
      Jan-17        2304                NULL            NULL            365
    Feb-17        2462                NULL            NULL            634
    Mar-17        4273                NULL            647             NULL
    Apr-17        2886                1643            NULL            NULL
    May-17        2150                1104            631             NULL
    Jun-17        4857                1284            1303            NULL
    Jul-17        1775                NULL            2364            934
    Aug-17        4089                529             1123            NULL
    Sep-17        4076                NULL            NULL            NULL
    Oct-17        2594                22625           NULL            474
    Nov-17        1971                81009           1265            NULL
    Dec-17        1631                30913           NULL            NULL
```

4) Write a brief summary analyzing the information you see in the pivot table created in question 3. Identify any anomalies and find the TX_IDs that are causing these.

Solution:

A) Summary of Pivot Table:

Total Payments from commercial insurance policies are higher in more than half of the months when compared to Medicare, Medicaid and Self Payment. But, During OCT, NOV, DEC of 2017 huge money has been received as total payment from Medicare. Almost 10 times the regular average amount has been received. This is something to investigate. And, less self-payment totals have been received in the months of 2016 when compared to 2017.

B) Anomalies in DB:

I have spotted an Insertion Anomaly in TBL_PATIENTS where INSURANCE_ID is NULL (meaning Data Entry people did not enter their insurance id in the Database). It results in no mapping of those patients with their insurance. Which cause and Insertion Anomaly.

Patients with no Insurance ID’s:

205	SARAH ARMSTRONG	11/12/1978	NULL
209	CATHERINE LEWIS	1/7/1965	NULL
225	STEFANIE SMITH	12/10/1981	NULL

Transactions with this Anomaly:

TX_ID	VISIT_DATE	PATIENT_ID	PHYSICIAN_ID	CHARGES	INSURANCE_PAYMENT	PATIENT_PAYMENT
44	1/21/2017	225	512	1311	NULL	365
56	2/3/2017	205	528	1690	NULL	634
68	10/8/2017	205	527	1060	NULL	474
76	3/15/2016	209	525	1656	NULL	703
96	7/19/2017	225	525	1785	NULL	934