

Health Insurance Claim Management System for

Chen's Chinese Medicine

Pan Chen

Syracuse University

## **Project Summary**

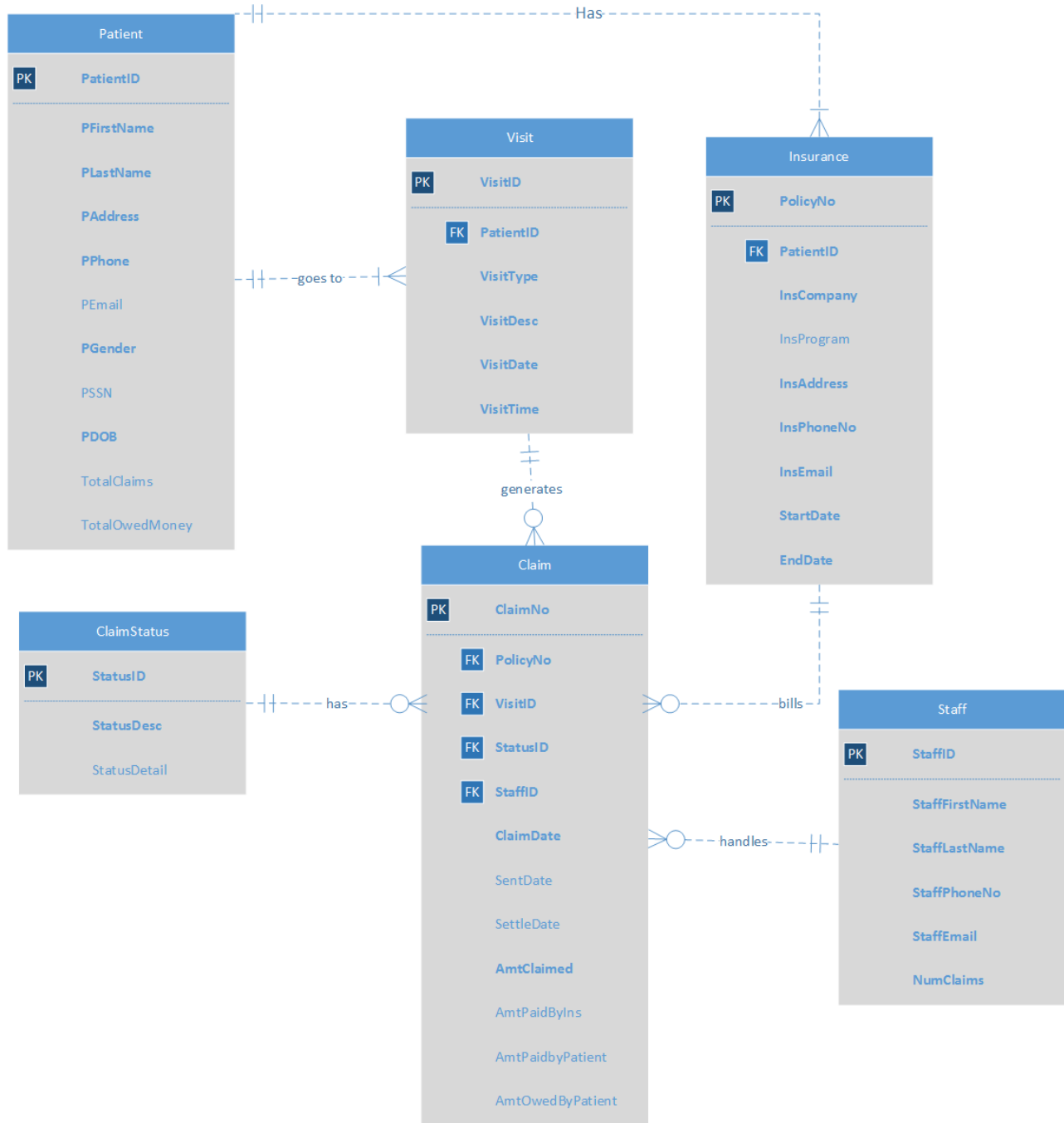
I have designed a Health Insurance Claim Management System for my aunt, who runs a clinic in Camarillo, CA called Chen's Chinese Medicine. The current one she is using is not too easy to navigate and not efficient enough: it has too much functions/options that an acupuncture clinic does not need, so it takes extra time to navigate through these redundant options input each patient's entry. Therefore, a simple design that focuses on the insurance billing might be a good start point for my project.

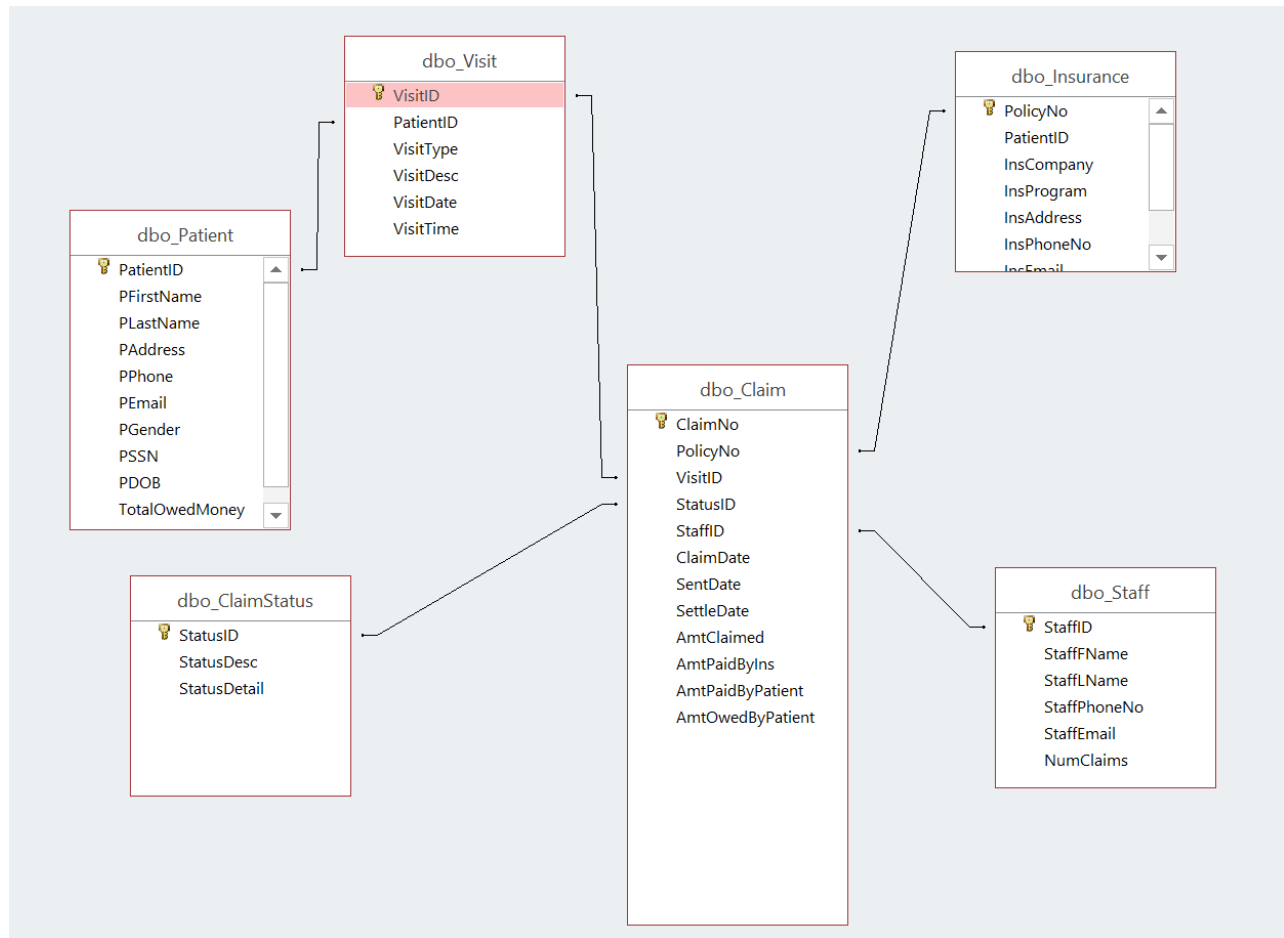
My aunt is the only practitioner in the clinic with several staff members that helps her with insurance claim filing and billing. This project is set to help the clinic manage the patient's payment to each office visit, which records the patient information and his/her insurance policy information, his/her office visits including appointment date and the description of the treatment he/she received, insurance claims information associate with his/her visits including their statuses, claim amount, and which staff member handles the claim, and the clinic staff's information.

The design is for two user groups. One is for the business owner/the practitioner, the other one is for the staff members.

Both the practitioner and staff members could check/edit the information for the patients, their insurance policy information, their clinic visit information, the statuses of the insurance claims, the report of the claims associated with each patient, and the report of the unsettled claims' details. The business owner could also check the summary of the staff members and the claims they are working on.

## Glossary and Data Model





All the required attributes are in **BOLD**

Data object	Explanation
<b>Patient</b>	Patients' personal information
<b><u>PatientID (PK)</u></b>	Either DL/ID Number or Passport Number
<b>PFirstName</b>	Patients' first name
<b>PLastName</b>	Patients' last name
<b>PAaddress</b>	Patient's address
<b>PPhone</b>	Patient's phone number
PEmail	Optional, patient's email address
<b>PGender</b>	Patient's gender
PSSN	Optional, patient's Social security number
<b>PDOB</b>	Patient's date of birth
TotalClaims	Optional, the total number of claims associated with a patient, the value is generated by a trigger.

TotalOwedMoney	Optional, the total amount of a patient's unpaid balance, the value is generated by a procedure.
<b>Insurance</b>	Patients' insurance policy information
<b><u>PolicyNo (PK)</u></b>	Patient's Insurance Policy Number
<b><u>PatientID (FK)</u></b>	Associates with PatientID in <b>Patient</b> Table, a patient could have one or more insurance policies, an insurance policy is only for one patient
<b>InsCompany</b>	
InsProgram	Insurance policy program
<b>InsAddress</b>	Insurance company address
<b>InsPhoneNo</b>	Insurance program phone number
<b>StartDate</b>	Policy Start Date
<b>EndDate</b>	Policy End Date
<b>Visit</b>	Patients' office visits information
<b><u>VisitID (PK)</u></b>	
<b>PatientID (FK)</b>	Associates with PatientID in <b>Patient</b> Table, a patient could have one or more visits, an office visit is only for one patient
<b>VisitType</b>	Type of visits, including Counseling, Medicine, Acupuncture, Cupping, and etc
<b>VisitDesc</b>	Describe the diagnosis/treatment
<b>VisitDate</b>	
<b>VisitTime</b>	
<b>Claim</b>	Details of a claim including billing information
<b><u>ClaimNo (PK)</u></b>	
<b>PolicyNo (FK)</b>	Associates with PolicyNo in <b>Policy</b> Table, a claim can only be filed under one policy, a policy can have multiple claims

<b>VisitID (FK)</b>	Associates with VisitID in <b>Visit</b> Table, a claim can only be filed for one policy, a visit can have multiple claims
<b>StatusID (FK)</b>	Associates with StatusID in <b>ClaimStatus</b> Table, a claim can only have one status, a status could be from multiple claims
<b>StaffID (FK)</b>	Associates with StaffID in <b>Staff</b> Table, a claim can only be filed by one staff, a staff can file multiple claims
<b>ClaimDate</b>	The date the claim is generated
SentDate	Optional, the date the claim is filed
SettleDate	Optional, the date the claim is settled
<b>AmtClaimed</b>	The amount asked by the clinic for the claim
AmtPaidByIns	Optional, the amount of money paid by insurance
AmtPaidbyPatient	Optional, the amount of money paid by the patient
AmtOwedByPatient	Optional, the amount of money owed by the patient, automatically calculated by AmtClaimed-AmtPaidByIns-AmtPaidbyPatient. Don't need to populate this field
<b>ClaimStatus</b>	Statuses of insurance claims
<b><u>StatusID (PK)</u></b>	Begins with either letter "N", "D", "P", "S" N for not filed yet, D for Dispute (Insurance company disagrees), P for pending (insurance company paid but the patient has not), S for settled
<b>StatusDesc</b>	Claim Status Description, which includes dispute, settled, not filed yet and pending
StatusDetail	Optional, Detailed description on what this status is about
<b>Staff</b>	Information about staff member including the doctor herself
<b><u>StaffID (PK)</u></b>	
<b>StaffFirstName</b>	
<b>StaffLastName</b>	
<b>StaffPhoneNo</b>	
<b>StaffEmail</b>	

NumClaims	Optional, Number of claims the staff member has been working on. Generated by a procedure
-----------	---

### Business Rules:

1. Each patient must have at least one insurance policy; each policy number belongs to only one patient.
2. A patient must visit the clinic at least once; the practitioner only sees one patient for a visit.
3. A clinic visit could generate multiple insurance claims or none of it, depending on its details; an insurance claim is only associated with one clinic visit.
4. Each claim has only one claim status; a claim status could be associated with multiple claims
5. Each claim only bills one insurance policy; the clinic can file multiple policies under one insurance claim.
6. A staff member could file multiple claims; a claim could only be handled by one staff member.

### SQL script for creating tables and inserting sample data

I will demonstrate the data-insertion by creating a patient called Chris Paul, who has two claims, and he paid one of them and has not paid the other one yet.

The claim will be handled by a staff member called Jamal Crawford.

### CREATE: Patient

```
CREATE TABLE Patient
(
  PatientID CHAR(20) PRIMARY KEY,
  PFirstName VARCHAR(30) NOT NULL,
  PLastName VARCHAR(30) NOT NULL,
  PAddress VARCHAR(50) NOT NULL,
  PPhone VARCHAR(20) NOT NULL,
  PEmail VARCHAR(30),
  PGender VARCHAR(8) NOT NULL,
  PSSN CHAR(9),
  PDOB DATE NOT NULL,
  TotalClaims Int,
  TotalOwedMoney Int,
);
```

### Insert Data in PATIENT Table:

```
INSERT INTO Patient VALUES('3', 'Chris', 'Paul', '213 Staples Center', '2131234567', 'cp3@lac.com', 'male', '333333333', '1982/10/13', Null, Null);
```

Select \* from patient

	PatientID	PFirstName	PLastName	PAddress	PPhone	PEmail	PGender	PSSN	PDOB	TotalOwedMoney
1	12345	Pan	Chen	770 james	8054444444	pachen@syr.edu	M	888888888	1990-09-21	120.00
2	2	Timmy	Turner	111. atl dr.	3151243124	timmy@ss.com	M	111111111	1987-02-03	150.00
3	3	Chris	Paul	213 Staples Center	2131234567	cp3@lac.com	male	333333333	1982-10-13	NULL

## CREATE: Insurance

```
CREATE TABLE Insurance
(
  PolicyNo CHAR(15) PRIMARY KEY,
  PatientID CHAR(20) NOT NULL,
  InsCompany VARCHAR(30) NOT NULL,
  InsProgram VARCHAR(30),
  InsAddress VARCHAR(50) NOT NULL,
  InsPhoneNo VARCHAR(20) NOT NULL,
  InsEmail VARCHAR(30) NOT NULL,
  StartDate DATE NOT NULL,
  EndDate DATE NOT NULL,
  CONSTRAINT PatientID_FK FOREIGN KEY (PatientID) REFERENCES Patient(PatientID)
);
```

## Insert Data in Insurance Table:

```
/*Populate Insurance Table*/
INSERT INTO Insurance VALUES('10003', '3', 'BlueShield', 'BaksetBall Insurance', '56 Wilshire Blvd',
'8184567890', 'support@blueshield.com', '2016/08/01', '2018/08/01');
```

	PolicyNo	PatientID	InsCompany	InsProgram	InsAddress	InsPhoneNo	InsEmail	StartDate	EndDate
1	10000	12345	Aetna	Student Health	123 Waverly	3156789022	aetna@blackmail.com	2016-01-01	2017-02-03
2	10002	2	aaa	bbb	sss	212344552	ff@ff.com	2017-01-01	2017-12-31
3	10003	3	BlueShield	BaksetBall Insurance	56 Wilshire Blvd	8184567890	support@blueshield.com	2016-08-01	2018-08-01

## Create: Visit

```
CREATE TABLE Visit
(
  VisitID CHAR(15) PRIMARY KEY,
  PatientID CHAR(20) NOT NULL,
  VisitType VARCHAR(20) NOT NULL,
  VisitDesc VARCHAR(200) NOT NULL,
  VisitDate DATE NOT NULL,
  VisitTime TIME NOT NULL,
  CONSTRAINT PatientID_FK FOREIGN KEY (PatientID) REFERENCES Patient(PatientID)
);
```

## Insert Data in Visit Table:



```
INSERT INTO visit VALUES('365', '3', 'Physical Therapy', 'Muscle spasm', '2016/12/10', '09:30');
```

	VisitID	PatientID	Visit Type	Visit Desc	Visit Date	Visit Time
1	12	12345	Counseling	so so, lel	2016-05-06	12:30:00.0000000
2	13	12345	Outpatient	patient in in grave danger	2017-01-01	17:40:00.0000000
3	365	3	Physical Therapy	Muscle spasm	2016-12-10	09:30:00.0000000

## CREATE: Staff

```
CREATE TABLE Staff
(
  StaffID CHAR(15) PRIMARY KEY,
  StaffFName VARCHAR(20) NOT NULL,
  StaffLName VARCHAR(20) NOT NULL,
  StaffPhoneNo VARCHAR(20) NOT NULL,
  StaffEmail VARCHAR(30) NOT NULL,
  NumClaims VARCHAR (30),
);
```

## Insert Data in Staff Table:

```
INSERT INTO Staff VALUES('11', 'Jamal', 'Crawford', '2134555555', 'JCrossover@twitter.com', NULL);
```

	StaffID	StaffFName	StaffLName	StaffPhoneNo	StaffEmail	NumClaims
1	11	Jamal	Crawford	2134555555	JCrossover@twitter.com	NULL
2	24	Kobe	Bryant	3104444444	Kobe@lal.org	1
3	34	Paul	Pierce	2134444444	pp@bos.net	NULL
4	67890	Lebron	James	3151234567	Lebron@cav.org	3

## CREATE: ClaimStatus

```
CREATE TABLE ClaimStatus
(
  StatusID CHAR(15) PRIMARY KEY,
  StatusDesc VARCHAR(15) NOT NULL,
  StatusDetail VARCHAR(200) NOT NULL
);
```

## Insert Data in ClaimStatus Table:

```
INSERT INTO ClaimStatus VALUES('P2', 'Pending',
'The insurance company paid 50%, so we are waiting for the patient to pay the other 50%');
```

	StatusID	StatusDesc	StatusDetail
1	D1	In Dispute	the insurance company only agrees to pay 50% of t...
2	N1	Not Filed Yet	the claim is not filed yet due to large volume of work
3	P1	Pending	Insurance company has paid, waiting for the patien...
4	P2	Pending	The insurance company paid 50%, so we are waiti...
5	S1	Settled	Both insurance company and patient have paid

## CREATE: Claim

```
CREATE TABLE Claim
(
ClaimNo CHAR(15) PRIMARY KEY,
PolicyNo CHAR(15) NOT NULL,
VisitID CHAR(15) NOT NULL,
StatusID CHAR(15) NOT NULL,
StaffID CHAR(15) NOT NULL,
ClaimDate DATE NOT NULL,
SentDate DATE,
SettleDate DATE,
AmtClaimed DECIMAL(10,2) NOT NULL,
AmtPaidByIns DECIMAL(10,2),
AmtPaidByPatient DECIMAL(10,2),
AmtOwedByPatient AS AmtClaimed - AmtPaidByIns - AmtPaidByPatient,
CONSTRAINT PolicyNo_FK FOREIGN KEY (PolicyNo) REFERENCES Insurance(PolicyNo)
CONSTRAINT VisitID_FK FOREIGN KEY (VisitID) REFERENCES Visit(VisitID),
CONSTRAINT StatusID_FK FOREIGN KEY (StatusID) REFERENCES ClaimStatus(StatusID)
CONSTRAINT StaffID_FK FOREIGN KEY (StaffID) REFERENCES Staff(StaffID),
);
```

## Insert Data in Claim Table:

```
INSERT INTO CLAIM (CLAIMNO, policyno,visitID,StatusID,StaffID,Claimdate,sentdate,settledate, AmtClaimed,AmtPaidByIns,AmtPaidByPatient) values('31', '10003', '365', 'S1','11','2016-12-10','2016-12-11', '2016-12-11', 120,60,60);

INSERT INTO CLAIM (CLAIMNO, policyno,visitID,StatusID,StaffID,Claimdate,sentdate,AmtClaimed,AmtPaidByIns,AmtPaidByPatient) values('21', '10003', '365', 'P2','11','2017-12-10','2016-12-11',200,100,0);
```

	ClaimNo	PolicyNo	VisitID	StatusID	StaffID	ClaimDate	SentDate	SettleDate	AmtClaimed	AmtPaidByIns	AmtPaidByPatient	AmtOwedByPatient
1	1	10000	12	S1	67890	2016-08-12	NULL	NULL	100.00	30.00	0.00	70.00
2	2	10000	12	D1	67890	2016-12-12	2016-12-13	NULL	500.00	400.00	50.00	50.00
3	21	10003	365	P2	11	2017-12-10	2016-12-11	NULL	200.00	100.00	0.00	100.00
4	3	10002	13	P1	24	2017-08-09	2017-08-10	NULL	400.00	300.00	0.00	100.00
5	31	10003	365	S1	11	2016-12-10	2016-12-11	2016-12-11	120.00	60.00	60.00	0.00
6	4	10002	13	S1	67890	2017-09-10	2017-09-10	NULL	300.00	250.00	0.00	50.00
7	5	10002	13	N1	67890	2017-09-10	2017-09-10	NULL	300.00	250.00	0.00	50.00

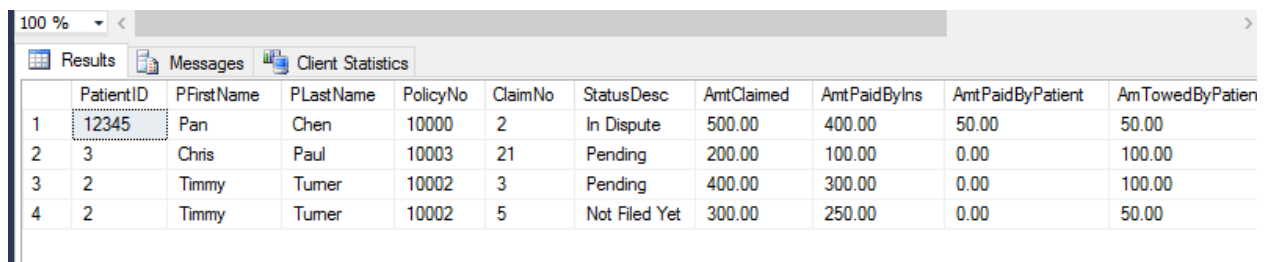
**Section V: SQL statements for answering major data questions using reports.**  
**Required to have at least one Data Question with Aggregate Functions (2 points)**

## Major Data Questions:

1. Check all the unsettled claims with detail

```
/*Check all the unsettled claims*/  
SELECT Patient.PatientID, PFirstName, PLastName, Cl.PolicyNo, ClaimNo, StatusDesc, AmtClaimed,  
AmtPaidByIns, AmtPaidByPatient, AmTowedByPatient FROM Patient  
INNER JOIN INSURANCE INS  
ON Patient.patientID=Ins.patientID  
Inner Join Claim cl  
ON INS.policyNo=cl.policyNo  
Inner Join ClaimStatus cs  
ON cs.StatusID=cl.StatusID  
WHERE StatusDesc not like 'Settled';
```

## Report:



	PatientID	PFirstName	PLastName	PolicyNo	ClaimNo	StatusDesc	AmtClaimed	AmtPaidByIns	AmtPaidByPatient	AmTowedByPatient
1	12345	Pan	Chen	10000	2	In Dispute	500.00	400.00	50.00	50.00
2	3	Chris	Paul	10003	21	Pending	200.00	100.00	0.00	100.00
3	2	Timmy	Turner	10002	3	Pending	400.00	300.00	0.00	100.00
4	2	Timmy	Turner	10002	5	Not Filed Yet	300.00	250.00	0.00	50.00

2. Check claims associated with each patient, and how much does each of the patient owe the clinic

Totalclaims is an aggregate field

```
/*total claims filed by a particular patient*/  
UPDATE patient  
SET Totalclaims = claims.count2  
FROM(  
Select pat.PatientID, Count(claimNo) AS "count2" from Patient pat  
INNER JOIN INSURANCE INS  
ON pat.patientID=Ins.patientID  
Inner Join Claim cl  
ON INS.policyNo=cl.policyNo  
GROUP BY pat.PatientID) as claims
```

totalowed is a stored procedure

```

/*how much money a patient owes the clinic*/
CREATE PROCEDURE totalOwedMoney
AS
BEGIN
    UPDATE Patient
    SET TotalOwedMoney = totalowed.count2
    FROM
        (Select pat.PatientID, sum(amtowedbypatient) AS "count2" from Patient pat
        INNER JOIN INSURANCE INS
        ON pat.patientID=Ins.patientID
        Inner Join Claim cl
        ON INS.policyNo=cl.policyNo
        GROUP BY pat.PatientID)
        AS totalowed
    WHERE patient.patientID = totalowed.patientID
END;

EXEC totalOwedMoney;

```

After running the function, the report looks like :

Patient Claim Report									
PatientID	PFirstName	PLastName	Totalclaims	TotalOwed	ClaimNc	AmtOwed	Status	StaffID	
12345	Pan	Chen	2	120	1	70	Sett	67890	
					2	50	In C	67890	
2	Timmy	Turner	2	200	3	100	Per	24	
					4	50	Sett	67890	
					5	50	Not	67890	
3	Chris	Paul	2	100	21	100	Per	11	
					31	0	Sett	11	

Homepage

3. Check what claims is each of the staff member working on, and how many claims each of the staff member is working on

## Staff and Claim

StaffID	StaffFName	StaffLName	NumClaims	ClaimNo	Status	SentDate	SettleDate
11	Jamal	Crawford		21	Pending	2016-12-11	
				31	Settled	2016-12-11	2016-12-11
24	Kobe	Bryant	1				
				3	Pending	2017-08-10	
67890	Lebron	James	3				
				1	Settled		
				2	In Dispute	2016-12-13	
				4	Settled	2017-09-10	
				5	Not Filed Yet	2017-09-10	

[Homepage](#)

Saturday, December 10, 2016

Page 1 of 1

NumClaims is an aggregate procedure

```

/* Write a procedure that updates staff table with the total claims handled by each staff member
*/
CREATE PROCEDURE totalclaimsStaff
AS
BEGIN
    UPDATE staff
    SET NumClaims = totalclaims.count1
    FROM
        (Select sta.StaffID, Count(claimNo) 'count1' from Staff sta
        Inner Join Claim cl
        ON cl.staffid=sta.staffid
        GROUP BY sta.staffID) AS totalclaims
    WHERE staff.staffid = totalclaims.staffID
END;

EXEC totalclaimsStaff;
```

After running the procedure, the report looks like:

## Staff and Claim

StaffID	StaffFName	StaffLName	NumClaims	ClaimNo	Status	SentDate	SettleDate
11	Jamal	Crawford	2	21	Pending	2016-12-11	
				31	Settled	2016-12-11	2016-12-11
24	Kobe	Bryant	1				
67890	Lebron	James	4	3	Pending	2017-08-10	
				1	Settled		
				2	In Dispute	2016-12-13	
				4	Settled	2017-09-10	
				5	Not Filed Yet	2017-09-10	

Homepage


Saturday, December 10, 2016

Page 1 of 1

### Interfaces

Create forms (with at least one combo box) and reports for users to perform data input and output easily. Include the screen copy of all forms and reports. Add captions to explain the purpose of each form or report. You should generate reports for the major data questions in Section VI.

This is the homepage for my system, both the practitioner and staff member could use it, although the staff member is not able to access the general staff performance report

**Homepage** 

Patient Info	Patient Visit Info
Insurance Info	Clinic Staff Info
Claim Info	Edit Claim Statuses
Generate Patient Claim Report	Generate Staff Performance Report

Click on "Patient Info" -> The patient information form: view and edit each patient's detailed personal information

## Patient Information

PatientID	12345
PFirstName	Pan
PLastName	Chen
PAddress	770 James
PPhone	8054444444
PEmail	pachen@syr.edu
PGender	M
PSSN	888888888
PDOB	1990-09-21

[Homepage](#)

Click on Insurance info -> The policy information form: view and edit information of patients' policies.

## Insurance Policy Information

PolicyNo	10000
PatientID	12345
InsCompany	Aetna
InsProgram	Student Health
InsAddress	123 Waverly
InsPhoneNo	3156789022
InsEmail	aetna@blackmail.com
StartDate	2016-01-01
EndDate	2017-02-03

[Homepage](#)

Click on Patient Visit Info -> The policy information form: view and edit information of patients' clinic visits

## Visit

VisitID	<input type="text" value="12"/>
PatientID	<input type="text" value="12345"/>
VisitType	<input type="text" value="Counseling"/>
VisitDesc	<input type="text" value="so so, lel"/>
VisitDate	<input type="text" value="2016-05-06"/>
VisitTime	<input type="text" value="12:30:00.0000000"/>

[Homepage](#)

Click on “Edit Claim Status” -> The Claim Status form: view and edit descriptions of different claims statuses

## ClaimStatus

StatusID	<input type="text" value="D1"/>
StatusDesc	<input type="text" value="In Dispute"/>
StatusDetail	<input type="text" value="the insurance company only agrees to pay 50% of the amount"/>

[Homepage](#)

Click on “Claim info” -> The Claim form: view and edit descriptions of different insurance claims



## Claim

ClaimNo	<input type="text" value="1"/>
PolicyNo	<input type="text" value="10000"/>
VisitID	<input type="text" value="12"/>
StatusID	<input type="text" value="S1"/>
StaffID	<input type="text" value="67890"/>
ClaimDate	<input type="text" value="2016-08-12"/>
SentDate	<input type="text"/>
SettleDate	<input type="text"/>
AmtClaimed	<input type="text" value="100"/>
AmtPaidByIns	<input type="text" value="30"/>
AmtPaidByPatient	<input type="text" value="0"/>
AmtOwedByPatient	<input type="text" value="70"/>


[Homepage](#)

Click on "Clinic Staff Info" -> The staff form: view and edit each staff member's information

## Staff

StaffID	<input type="text" value="24"/>
StaffFName	<input type="text" value="Kobe"/>
StaffLName	<input type="text" value="Bryant"/>
StaffPhoneNo	<input type="text" value="3104444444"/>
StaffEmail	<input type="text" value="Kobe@lal.org"/>
NumClaims	<input type="text" value="1"/>

[Homepage](#)

Click on  -> Patient Search: Search each patient's total owed money and their claims' statuses by selecting Patient's ID

## Patient and Claims

PatientID1	2
PFirstName	Timmy
PLastName	Turner
TotalOwedMoney	150

## Claims

ClaimNo	StatusDesc
3	Pending
4	Settled
5	Not Filed Yet

Record: 1 of 3

No Filter

Search

```
CREATE TRIGGER claimsfiled
ON claim
FOR INSERT, UPDATE, DELETE
AS
IF @@ROWCount>=0
BEGIN
UPDATE PATIENT
SET Totalclaims = claimcount.count1
FROM
(Select pat.PatientID, Count(claimNo) AS "count1" from Patient pat
INNER JOIN INSURANCE INS
ON pat.patientID=Ins.patientID
Inner Join Claim cl
ON INS.policyNo=cl.policyNo
GROUP BY pat.PatientID) AS claimcount
END;
```

```
Select * from Patient
```

100 %

Results Messages Client Statistics

	PatientID	PFirstName	PLastName	PAddress	PPhone	PEmail	PGender	PSSN	PDOB	TotalOwedMoney	Totalclaims
1	12345	Pan	Chen	770 james	8054444444	pachen@syr.edu	M	888888888	1990-09-21	120.00	2
2	2	Timmy	Turner	111, atl dr.	3151243124	timmy@ss.com	M	111111111	1987-02-03	200.00	2
3	3	Chris	Paul	213 Staples Center	2131234567	cp3@lac.com	male	333333333	1982-10-13	100.00	2

Now we delete one

```
Delete from Claim  
where claimNo='31';
```

100 %

Messages Client Statistics

(3 row(s) affected)

(1 row(s) affected)

Now the number of his claims is updated

100 %

Results Messages

	PatientID	PFirstName	PLastName	PAddress	PPhone	PEmail	PGender	PSSN	PDOB	TotalOwedMoney	Totalclaims
1	12345	Pan	Chen	770 james	8054444444	pachen@syr.edu	M	888888888	1990-09-21	120.00	2
2	2	Timmy	Turner	111, atl dr.	3151243124	timmy@ss.com	M	111111111	1987-02-03	200.00	1
3	3	Chris	Paul	213 Staples Center	2131234567	cp3@lac.com	male	333333333	1982-10-13	100.00	2