

Xue Bai (NUIID: 001087646)

Yanwen Duan (NUIID 001028144)

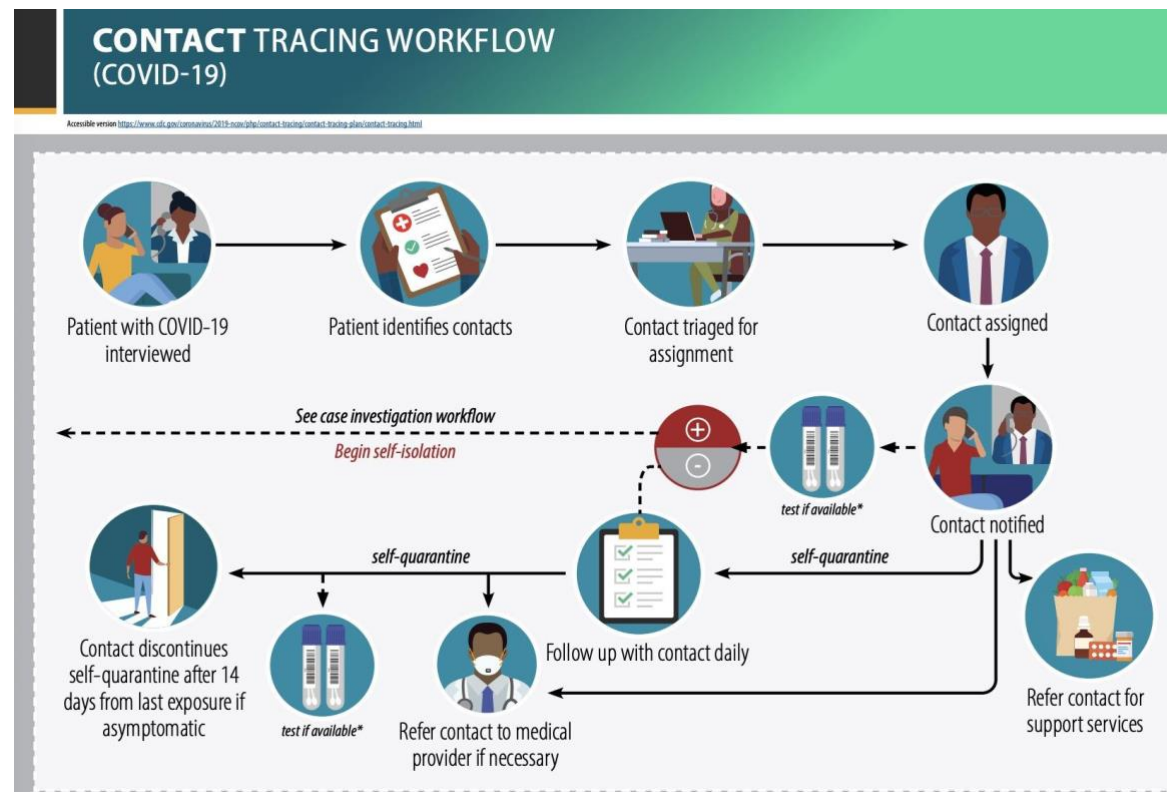
Cheng Zhao (NUIID 001013862)

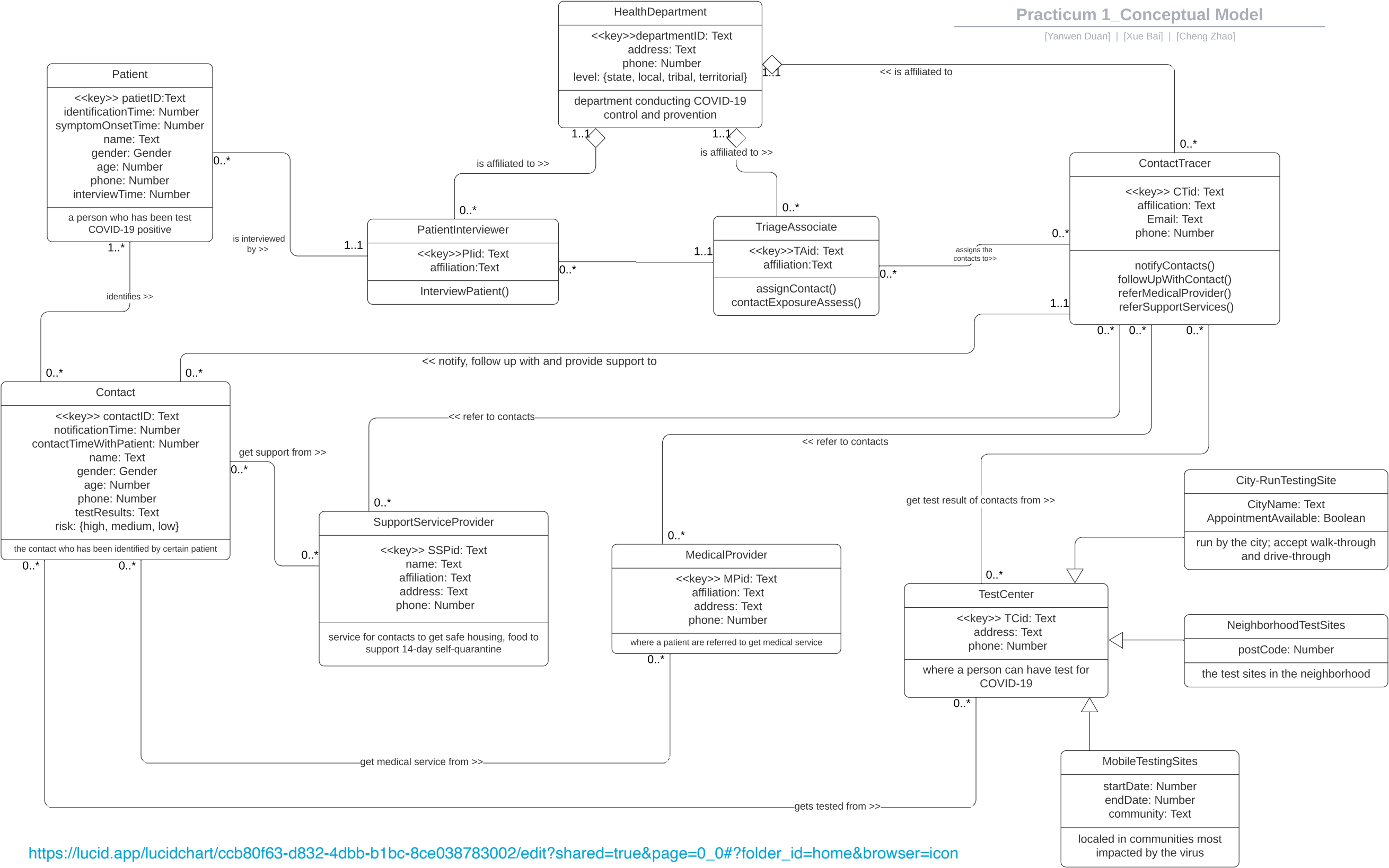
Practicum 1

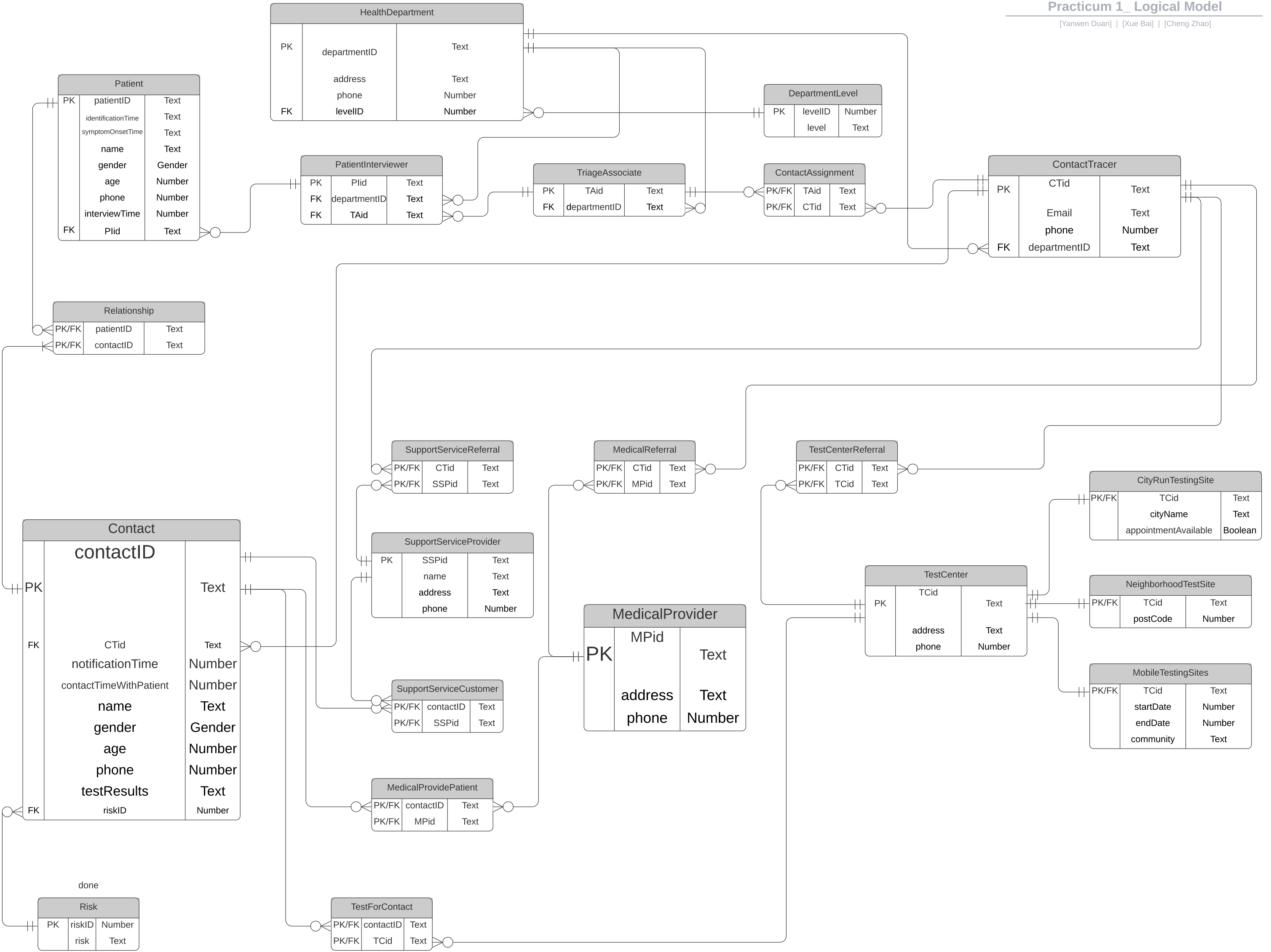
CS5200 Database

2020 Fall

Use Case: The design and the model of the database imitates the process (as shown in the following picture) of working with a person (patient) who has been diagnosed with an infectious disease to identify and provide support to people (contacts) who may have been infected through close contact with the patient. It can also be used to record and analyze activities involving people who may have been exposed to the virus, through close contact with a person diagnosed with COVID-19.







Assumption for the Conceptual Model:

- Patient -- contact: Assume that each patient may identify zero or more contacts; each contact can be identified by one or more patients.
- Patient -- interviewer: Each patient would be interviewed by one interviewer, and an interviewer may receive zero or more patients.
- Interviewer -- triage: A certain interviewer would pass on the content of an interview with a certain patient to a single triage associate. Each triage associate may deal with zero or more interviewers.
- Triage -- Tracer: A triage associate will pass on info to zero (when the patient doesn't have any contacts) or multiple tracers. A tracer might interact with none or multiple triage associates.
- Interviewer -- healthDepartment: One health department may have zero or more interviewers. One interviewer can only be full-time and belongs to only one health department.
- Triage -- healthDepartment: One health department may have zero or more triage associates. One triage associate can only be full-time and belongs to only one health department.
- Tracers -- healthDepartment: One health department may have zero or more contact tracers. One contact tracers can only be full-time and belongs to only one health department.
- Tracer -- Contact: One contact can only be handled by one tracer, while one tracer can trace zero or more contacts.

- testCenter generalization: Assume that test centers are generally divided into the following groups: city-run testing sites, neighbourhood test sites and mobile test sites. (Referenced from <https://sf.gov/find-out-about-your-covid-19-testing-options>)
- TestCenter -- ContactTracer: a contact of a patient may not need a COVID-19 test if at low risk, then the tracer of this contact may not need to get the result from a test center. A test center may have no contact tracer to report test results to if no contacts are to be tested there.
- MedicalProvider -- ContactTracer: if a contact doesn't need any medical service during the process, then a contact tracer will not need to refer a medical provider and for the same reason, a medical provider may not have any contact tracers.
- supportServiceProvier -- ContactTracer: if a contact doesn't need any support service during the process/self-quarantine, then a contact tracer will not need to refer a support service provider and for the same reason, a support service provider may not have any contact tracers.
- Contact -- SupportService: if a contact doesn't need any support service during the process/self-quarantine, then a contact will not need any support service provider and for the same reason, a support service provider may not have any contacts to provide service to.

- Contact -- MedicalProvider : if a contact doesn't need any medical service during the process, then a contact will not need any medical provider and for the same reason, a medical provider may not have any contacts to perform medical operations.
- Contact -- TestCenter: a contact of a patient may not need a COVID-19 test if at low risk. For this reason, a test center may have no contacts to test.

Relational Schema:

Patient (patientID, identificationTime, symptomOnsetTime, name, gender, age, phone, interviewTime, *Plid*)

Relationship (patientID, contactID)

HealthDepartment (departmentID, address, phone, *levelID*)

DepartmentLevel (levelID, level)

PatientInterviewer (Plid, *departmentID*, *TAid*)

TriageAssociate (TAid, *departmentID*)

ContactAssignment (TAid, CTid)

ContactTracer (CTid, Email, phone, *departmentID*)

Contact (contactID, *CTid*, notificationTime, contactTimeWithPatient, name, gender, age, phone, testResults, *riskID*)

Risk (riskID, risk)

TestCenterReferral (CTid, TCid)

TestCenter (TCid, address, phone)

CityRunTestingSite (TCid, cityName, appointmentAvailable)

NeighborhoodTestSite (TCid, postCode)

MobileTestingSites (TCid, startDate, endDate, community)

TestForContact (contactID, TCid)

MedicalReferral (CTid, MPid)

MedicalProvider (MPid, address, phone)

MedicalProviderPatient (contactID, MPid)

SupportServiceCustomer (contactID, SSPid)

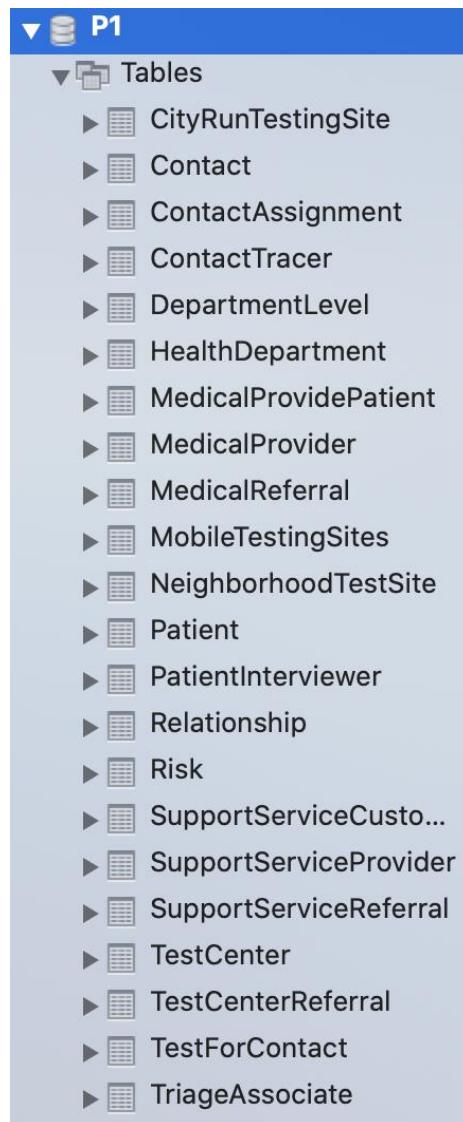
SupportServiceReferral (SSPid, CTid)

SupportServiceProvider (SSPid, name, address, phone)



Relational Schema normalization analysis:

With regard to our introduced schema, each functional dependency has a determinant which is a superkey. On the other hand, there is no non-superkey attribute that is a determinant of a functional dependency. Therefore, this subject schema is in BCNF.



Tables Generated From MySQLWorkbench





DESCRIBE DepartmentLevel;

Result Grid  Filter Rows: <input type="text" value="Search"/> Export: 							
	Field	Type	Null	Key	Default	Extra	
	levelID	int	NO	PRI	NULL		
	level	varchar(255)	YES		NULL		



DESCRIBE HealthDepartment;

Result Grid  Filter Rows: <input type="text" value="Search"/> Export: 							
	Field	Type	Null	Key	Default	Extra	
▶	departmentID	varchar(255)	NO	PRI	NULL		
	address	text	YES		NULL		
	phone	varchar(25)	YES		NULL		
	levelID	int	YES	MUL	NULL		



DESCRIBE TriageAssociate;

Result Grid  Filter Rows: <input type="text" value="Search"/> Export: 							
	Field	Type	Null	Key	Default	Extra	
▶	TAid	varchar(255)	NO	PRI	NULL		
	departmentID	text	NO		NULL		



DESCRIBE TestCenter;

Result Grid  Filter Rows: <input type="text" value="Search"/> Export: 							
	Field	Type	Null	Key	Default	Extra	
▶	TCid	varchar(255)	NO	PRI	NULL		
	address	text	YES		NULL		
	phone	varchar(25)	YES		NULL		



DESCRIBE MobileTestingSites;

Result Grid  Filter Rows: <input type="text" value="Search"/> Export: 							
	Field	Type	Null	Key	Default	Extra	
▶	TCid	varchar(255)	NO	PRI	NULL		
	startDate	varchar(255)	YES		NULL		
	endDate	varchar(255)	YES		NULL		
	community	varchar(255)	YES		NULL		



DESCRIBE CityRunTestingSite;

Result Grid  Filter Rows: <input type="text" value="Search"/> Export: 							
	Field	Type	Null	Key	Default	Extra	
▶	TCid	varchar(255)	NO	PRI	NULL		
	cityName	varchar(255)	NO		NULL		
	appointmentAvailable	tinyint(1)	NO		NULL		

DESCRIBE SupportServiceProvider;

Result Grid  Filter Rows: <input type="text" value="Search"/> Export: 							
	Field	Type	Null	Key	Default	Extra	
▶	SSPid	varchar(255)	NO	PRI	NULL		
	p_name	varchar(255)	YES		NULL		
	affiliation	text	YES		NULL		
	address	text	YES		NULL		
	phone	varchar(25)	YES		NULL		

DESCRIBE MedicalProvider;

Result Grid  Filter Rows: <input type="text" value="Search"/> Export: 							
	Field	Type	Null	Key	Default	Extra	
	MPid	varchar(255)	NO	PRI	NULL		
	affiliation	varchar(255)	YES		NULL		
	address	text	YES		NULL		
▶	phone	varchar(25)	YES		NULL		

DESCRIBE NeighborhoodTestSite;

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Field	Type	Null	Key	Default	Extra
▶	TCid	varchar(255)	NO	PRI	NULL	
	postCode	int	YES		NULL	

DESCRIBE PatientInterviewer;

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Field	Type	Null	Key	Default	Extra
▶	PIid	varchar(255)	NO	PRI	NULL	
	departmentID	varchar(255)	YES	MUL	NULL	
	TAid	varchar(255)	YES	MUL	NULL	

DESCRIBE Patient;

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Field	Type	Null	Key	Default	Extra
▶	patientID	varchar(255)	NO	PRI	NULL	
	identificationTime	varchar(255)	YES		NULL	
	symptomOnsetTime	varchar(255)	YES		NULL	
	name	varchar(255)	YES		NULL	
	gender	varchar(255)	YES		NULL	
	age	int	YES		NULL	
	phone	varchar(25)	YES		NULL	
	interviewTime	varchar(255)	YES		NULL	
	PIid	varchar(255)	YES	MUL	NULL	

DESCRIBE ContactTracer;

Result Grid						
		Filter Rows:			Export:	Wrap Cell Content:
	Field	Type	Null	Key	Default	Extra
▶	CTid	varchar(255)	NO	PRI	NULL	
	Email	varchar(255)	NO	UNI	NULL	
	phone	varchar(25)	NO		NULL	
	departmentID	varchar(255)	NO	MUL	NULL	

DESCRIBE Risk;

Result Grid						
		Filter Rows:			Export:	Wrap Cell Content:
	Field	Type	Null	Key	Default	Extra
▶	riskID	int	NO	PRI	NULL	
	risk	varchar(255)	NO		NULL	

DESCRIBE Contact;

Result Grid						
Filter Rows:			Export:		Wrap Cell Content:	
	Field	Type	Null	Key	Default	Extra
	contactID	varchar(255)	NO	PRI	NULL	
▶	CTid	varchar(255)	YES	MUL	NULL	
	notificationTime	varchar(255)	NO		NULL	
	contactTimeWithPatient	varchar(255)	NO		NULL	
	name	varchar(255)	NO		NULL	
	gender	enum('Male','Female','Other')	YES		NULL	
	age	int	YES		NULL	
	phone	varchar(25)	NO		NULL	
	testResults	tinyint(1)	NO		NULL	
	riskID	int	YES	MUL	NULL	

DESCRIBE Relationship;

Result Grid						
Filter Rows:			Export:		Wrap Cell Content:	
	Field	Type	Null	Key	Default	Extra
▶	patientID	varchar(255)	NO	PRI	NULL	
	contactID	varchar(255)	NO	PRI	NULL	

```
1 describe TestCenterReferral;
```

300%



29:1

Result Grid



Filter Rows:



Search

Export:



	Field	Type	Null	Key	Default	Extra	
▶	CTid	varchar(255)	NO	PRI	NULL		
	TCid	varchar(255)	NO	PRI	NULL		

```
1 describe SupportServiceReferral;
```

300%



33:1

Result Grid



Filter Rows:



Search

Export:



	Field	Type	Null	Key	Default	Extra	
▶	SSPid	varchar(255)	NO	PRI	NULL		
	CTid	varchar(255)	NO	PRI	NULL		

```
1 describe SupportServiceCustomer;
```

300% 33:1

Result Grid Filter Rows: Search Export:

Field	Type	Null	Key	Default	Extra	
contactID	varchar(255)	NO	PRI	NULL		
SSPid	varchar(255)	NO	PRI	NULL		

```
1 describe MedicalReferral;
```

300% 25:1

Result Grid Filter Rows: Search Export:

Field	Type	Null	Key	Default	Extra	
CTid	varchar(255)	NO	PRI	NULL		
MPid	varchar(255)	NO	PRI	NULL		

```
1 describe ContactAssignment;
```

300% 27:1

Result Grid Filter Rows: Search Export:

Field	Type	Null	Key	Default	Extra
TAid	varchar(255)	NO	PRI	NULL	
CTid	varchar(255)	NO	PRI	NULL	

```
1 describe TestForContact;
```

300% 25:1

Result Grid Filter Rows: Search Export:

Field	Type	Null	Key	Default	Extra
contactID	varchar(255)	NO	PRI	NULL	
TCid	varchar(255)	NO	PRI	NULL	

```
1 describe MedicalProvidePatient;
```

300%



3:1

Result Grid



Filter Rows:

Search

Export:



	Field	Type	Null	Key	Default	Extra	
▶	contactID	varchar(255)	NO	PRI	NULL		
	MPid	varchar(255)	NO	PRI	NULL		

Demo for DELETE/DROP constraints

DROP TABLE HealthDepartment;

Result Grid			Filter Rows:	Search	Export:
Level	Code	Message			
Error	3730	Cannot drop table 'healthdepartment' referenced by a foreign key constraint 'patientinterviewer_ibfk_1' on table 'PatientInterviewer'.			

Query Result in MySQLWorkbench

The screenshot displays the MySQL Workbench interface. At the top, a toolbar contains various icons for file operations, execution, and navigation. A dropdown menu is set to "Limit to 1000 rows". The main editor area contains a SQL query with line numbers 1 through 10. The query is as follows:

```
1  -- Q1. List the emails of tracers who work at tribal level health department
2  SELECT Email
3  FROM ContactTracer AS CT
4  JOIN (SELECT departmentID
5        FROM HealthDepartment AS HD
6        JOIN DepartmentLevel AS DL ON HD.levelID = DL.levelID
7        WHERE level = "tribal") AS DI ON CT.departmentID = DI.departmentID;
```

Below the query editor, the "Result Grid" tab is active. It shows a table with one column, "Email". The first row contains the email address "Phasellus@adipiscing.co.uk", and the subsequent rows are empty.

100% 1:10

Result Grid Filter Rows: Search Export:











Email
Phasellus@adipiscing.co.uk








28

1:10

[illegible]



Limit to 1000 rows



27

28

29 -- Q4. list the IDs of Patient Interveiwers who has patients aging between 18 to 50 years old

30 • SELECT PI.PIid

31 FROM PatientInterviewer AS PI

32 JOIN Patient AS P

33 ON PI.PIid=P.PIid

34 WHERE P.age > 18 AND P.age < 50;



35

36

100%


1:10

Result Grid



Filter Rows:

Export:



	Plid	
▶	164304021266	
	167511266889	
	165405291898	
	160211047832	

Limit to 1000 rows

```

37  -- Q5. Get the total number of contacts tested at city-run test center grouped by city
38  *  SELECT cityName,
39         CRTS.TCid,
40         countOfContacts,
41         AVG(countOfContacts) OVER(PARTITION BY CRTS.cityName) AS avgNumberOfTestedContacts,
42         SUM(countOfContacts) OVER(PARTITION BY CRTS.cityName) AS totalNumberOfTestedContacts
43  FROM CityRunTestingSite as CRTS
44  ⊖ JOIN (SELECT COUNT(contactID) countOfContacts, TFC.TCid
45         FROM TestForContact AS TFC
46         JOIN CityRunTestingSite AS CRTS
47         ON TFC.TCid = CRTS.TCid
48         GROUP BY TFC.TCid) AS countContact_TC
49  ON CRTS.TCid = countContact_TC.TCid;

```

100%
1:10

Result Grid
Filter Rows:

Export:

	cityName	TCid	countOfConta...	avgNumberOfTestedConta...	totalNumberOfTestedCont...	
▶	Montelupo Fiorentino	C8FA18D1-5075-9E97-A0F8-04155E978045	1	1.0000	1	
	Oostkerk	0F08900A-DEE9-739E-5096-6899654CA947	1	1.0000	1	