

CSC 675 - 01

Hospital Management Database

Oral Presentation : PoHung Wang

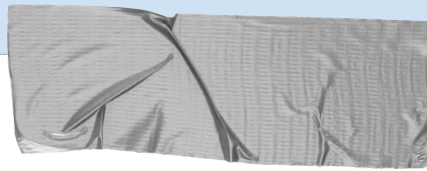
Team : Brandon Yu, Mark Soriano



Intro

We create a MySQL database to present a simple Hospital Data Management System.

Hospital database consist of hospital, doctor, patient and medical records.



Project Description

Patients are identified by **patient ID** and has **name**, **address**, and **diagnosis** as attribute.

Each patient's medical record is identified by **record id**, and keep the examination **date** and medical **problem** as attribute.

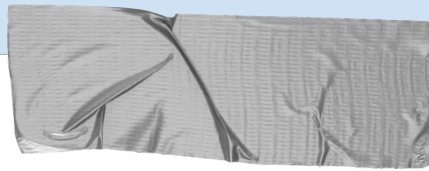
Each doctor is identified by **doctor ID** and has **name**, **specialization** and **salary** as attribute.

Each hospital is identified by **hospital ID**, and has **name**, **address**, **city** located as attribute.

Hospital has multiple patients.

Hospital has multiple doctors.

Each patient has records of examinations.



Entities and Attributes

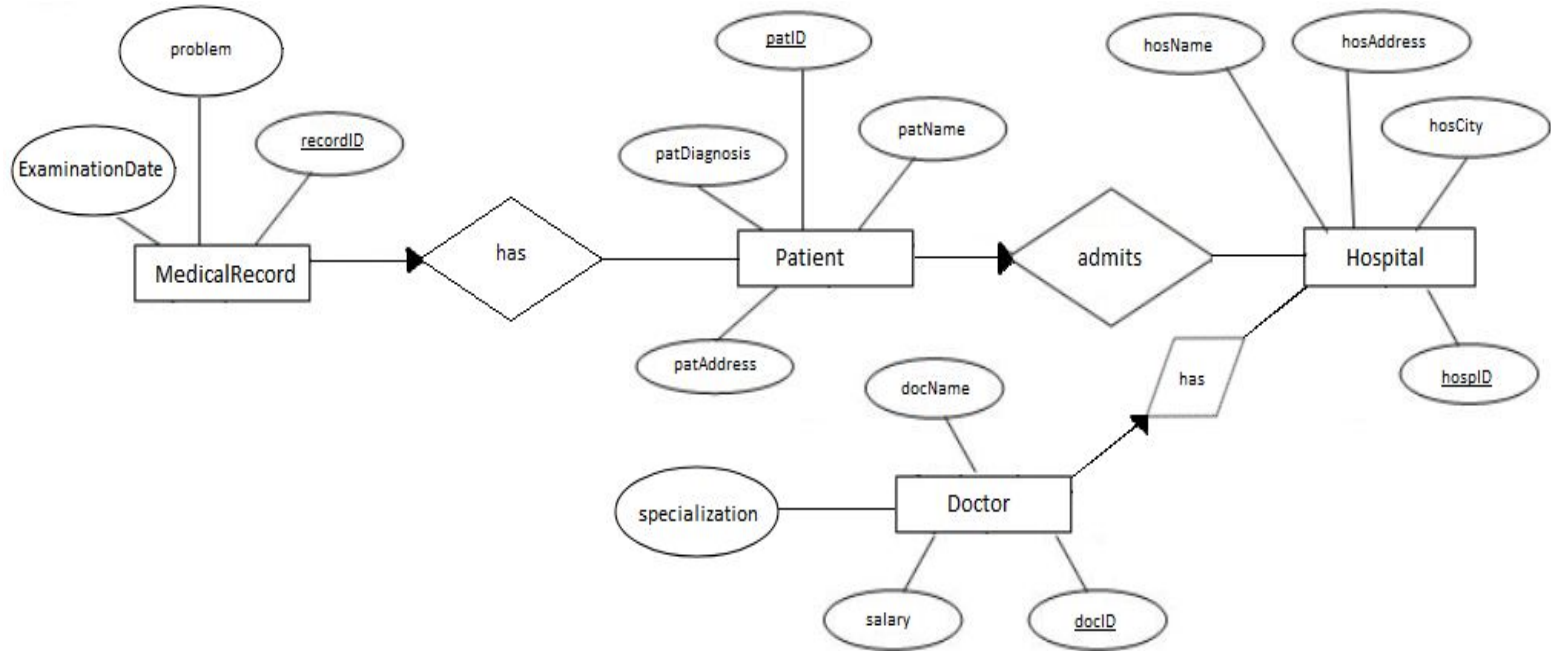
Hospital : hosplD, hosName, hosAddress, hosCity

Doctor : docID, docName, specialization, salary

Patient : patID, patName, patDiagnosis, patAddress

Medical record : recordID, ExaminationDate, problem

ER Diagram





Relationship Schema

Patient (patID:Integer, patName:String, patDiagnosis:String, patAddress:String)

Hospital (hosplD:Integer, hosName:String, hosAddress:String, hosCity:String)

Doctor (docID:Integer, docName:String, specialization:String, salary:Integer)

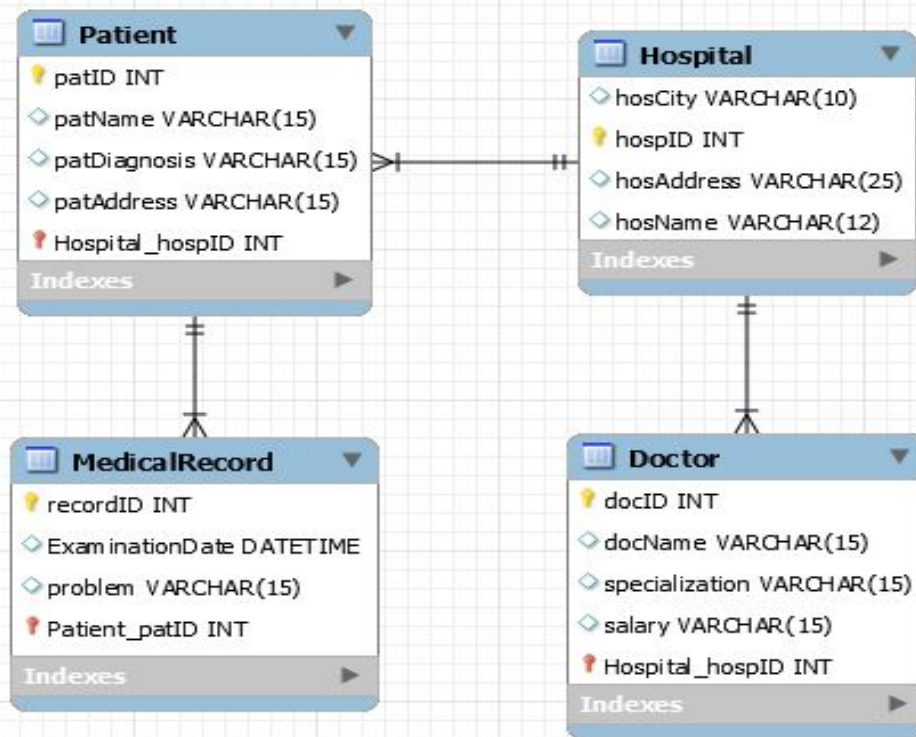
MedicalRecord (recordID:Integer, ExaminationDate:Date, problem:String)

Has_patient_record (recordID:Integer, patID:Integer)

Has_hosp_doc (hosplD:Integer, docID:Integer,)

Admits_patient_hosp (patID:Integer, hosplD:Integer)

Logic Schema





Index and Insert data

Inserting data to the database table

```
INSERT INTO `hospitalsys`.`doctor` (`docID`, `docName`, `specialization`, `salary`) VALUES ('12347', 'Bob', 'Gastroenterology', '131233');
```

```
INSERT INTO `hospitalsys`.`doctor` (`docID`, `docName`, `specialization`, `salary`) VALUES ('12348', 'Billy', 'Urology', '123122');
```

```
INSERT INTO `hospitalsys`.`doctor` (`docID`, `docName`, `specialization`, `salary`) VALUES ('12349', 'Ben', 'Podiatry', '90000');
```

Creating index

```
CREATE INDEX pat_id ON patient (patID);
```


```
CREATE INDEX medrec_id ON medicalrecord (recordID);
```


View

Creating the view

```
USE hospitalsys;  
CREATE VIEW patient_view  
AS SELECT patName, patAddress  
FROM patient
```

```
/* list the patients name and address on view*/  
SELECT * FROM hospitalsys.patient_view;
```

▼  patient_view

- ◆ patName
- ◆ patAddress

	patName	patAddress
	James	18 Darwin Stree
	Karl	28 Darwin Stree
	Bob	12 Kame Street
	Bill	32 Kame Street

SQL QUERY

/* find how many patients have
diagnosis of asthma */

```
1 SELECT P.patDiagnosis, count(*)
2 FROM heroku_fe27e80f5c08c1a.patient P
3 GROUP BY P.patDiagnosis
4 HAVING P.patDiagnosis = 'Asthma' ;
5
```

<	
Result Grid	
Filter Rows: <input type="text"/>	
Export:  Wrap Ce	
patDiagnosis	count(*)
Asthma	2

SQL QUERY

```
1
2
3 • /*find the doctors with over 100000 salary*/
4 SELECT D.docName, D.salary
5 FROM heroku_fe27e80f5c08c1a.doctor D
6 GROUP BY D.docName
7 HAVING D.salary > 100000;
```







<

Result Grid   Filter Rows: | Export:  | Wrap Cell Content: 

	docName	salary
	Billv	123122
	Bob	131233
	Dennv	111000
	James	341231
	Kurt	525311

SQL QUERY

```
1
2  /* Find the doctor whose salary is greater than that of some doctor name "Bob" */
3  SELECT *
4  FROM heroku_fe27e80f5c08c1a.doctor D
5  WHERE D.salary > ANY (SELECT DW.salary
6                        FROM heroku_fe27e80f5c08c1a.doctor DW
7                        WHERE DW.docName= 'Bob');
```

<					
Result Grid					
Filter Rows: <input type="text"/>					
Edit:   					
Export/Import:  					
Wrap Cell Content: 					
docID	docName	specialization	salary	Hospital_hospID	
12345	Kurt	Cardiologv	525311	123	
12346	James	Orthopedic	341231	123	
NULL	NULL	NULL	NULL	NULL	

SQL QUERY

```
1
2  /* Find names of patient whose had Fever*/
3  SELECT P.patName
4  FROM heroku_fe27e80f5c08c1a.patient P
5  WHERE P.patID IN ( SELECT M.Patient_patID
6                     FROM heroku_fe27e80f5c08c1a.medicalrecord M
7                     WHERE M.problem = "Fever");
```

<

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

patName
James
Bob

SQL QUERY

```
1  /* Find names of patient who have more than two medical record*/  
2  •  SELECT P.patName  
3     FROM heroku_fe27e80f5c08c1a.patient P  
4  WHERE Exists (( SELECT M.patID  
5                  FROM heroku_fe27e80f5c08c1a.has_patient_record M  
6                  WHERE M.patID = P.patID  
7                  Group By patID  
8                  Having count(*) > 2));
```



Result Grid



Filter Rows:

Export:



Wrap Cell Content:



patName
James

Our Database Tables



Language / Framework for Web Interface

Backend: PHP

Database (Heroku ClearDB): MySQL

Frontend: HTML, CSS



2. Web Demo Showcase

<https://paulwangcsc675.herokuapp.com>



4. Conclusion

- We learned how to use MySQL Workbench
- Working with database to integrate on a website
- Learned how to create queries on PHP and displaying data on the web.
- Learning how to design an infrastructure database system



Thank you