**Hospital Management System**

**Abstract:**

Our project hospital management is to computerize the management of hospital to develop a software that is user friendly, simple, fast and cost effective.it is used to list all the hospitals and their location and it deals with the collection of patients information like patient name and their address and to store details of staff automatically. It helps us to easily delete patient, add patient, update patient and to know about patient condition.

The main function of this project is to register and store the doctor and patient details and retrieved these details when required. It also includes status of each room in the hospital and also we can easily tell that which department the particular doctor belongs to. It is very helpul to know whether the particular blood is available in the hospital or not and its charges. User can search availability of doctor and details of the patient using the id. The interface is very user friendly. The data are well protected for personal use and makes the data processing very fast. It is as simple as using the computer. It is very helpful to both patients and hospital management.

**Requirement analysis:**

let us consider traditional system where the data is stored in files. The drawback of this system is it is very difficult to retrieve the data from case files. It is difficult to retrieve the data from case files. It is difficult to handle the whole system manually and it is less accurate to keep the data in case files because the files may get destroyed. Redundancy of data may occur and this lead to the inconsistency. It is time consuming.

The proposed system is easy to operate. Speed and accuracy are the main advantages of proposed system. There is no redundancy of data. The data are stored in the computer’s secondary memories like hard disk etc. it can be easily receive and used at any time. The proposed system will easily handle all the data and the work done by the existing system. The proposed system eliminate the drawbacks of the existing system to a great extent and it provides tight security to data.

**Modules:**

**Patient:**

In this we are taking all the patient details from the hospital which we have taken in the above one, In this we will cross-check whether the patient has been admitted in the hosiptals mentioned above,if not making a new registration of the patient and storing the details of patient.

**Doctor:**

We have taken doctor information from the above hosiptals and noted down the qualification, work experience, and their address. This module contains all the information about doctor and who are the patients he is giving treatment.

**Blood Bank**

In this we are going to collect the blood from the donors and store them in the blood packets and separate the blood packets as per the blood types and give them packets id.If a person requires or request the amount related to it will be charged and noted down.

**ER diagram:**

Diagram

Description automatically generated

**Schema Refinement:**

Doctor (Did: integer, name: string, age: integer, address: string)

Patient (pid: integer, pname: string, paddress: string, disease: string)

Rooms (rid: integer, type: string, price:real)

Department (Deid: integer, dname: string, location: string)

Blood bank (bloodid: integer, group: string)

**Relational Database Schema:**

|  |  |  |  |
| --- | --- | --- | --- |
| Entity | Relationship | Entity | Cardinality |
| Doctor | Checks | Patient | One to many |
| Patient | Admitted in | Rooms | Many to many |
| Doctor | Works in | Department | One to many |
| doctor | suggest | Blood bank | One to one |

Logical Database Design:

Doctor table:

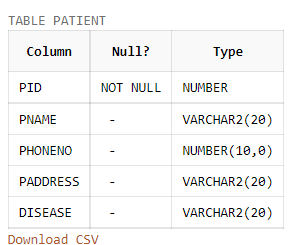
Desc Doctor;

Table

Description automatically generated

Patient table:

Desc patient;



**Room table:**

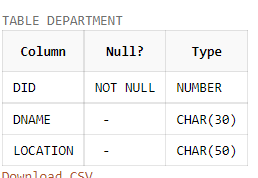
Desc Room;

Table

Description automatically generated

**Department table:**

Desc department;



**Blood bank table:**

Desc bloodbank;

Table

Description automatically generated

**Sample data:**

**DOCTOR:**

CREATE TABLE DOCTOR (DID NUMBER PRIMARY KEY, DNAME VARCHAR(30),DAGE NUMBER,DEXPERIENCE NUMBER);

INSERT INTO DOCTOR VALUES (1001, 'raju',45,9);

INSERT INTO DOCTOR VALUES (1002, 'jyothi', 30,3);

INSERT INTO DOCTOR VALUES (1003, 'sai',45,9);

INSERT INTO DOCTOR VALUES (1004, 'akhil',42,12);

INSERT INTO DOCTOR VALUES (1005, 'bhanu',45,9);

select \* from DOCTOR;

**Table

Description automatically generated**

**Patient table:**

CREATE TABLE PATIENT (PID NUMBER PRIMARY KEY, PNAME VARCHAR(20),PhoneNO NUMBER(10) UNIQUE, PADDRESS VARCHAR(20), disease VARCHAR(20));

INSERT INTO PATIENT VALUES (1, 'vasanth',9696969977, 'BILASPUR', 'BROKEN FINGER');

INSERT INTO PATIENT VALUES (2, 'kiriti', 9696458456, 'VIZAG', 'COLD AND COUGH');

INSERT INTO PATIENT VALUES (3, 'Nikitha', 9696969843, 'HYDERABAD', 'SKIN INFECTION');

INSERT INTO PATIENT VALUES (4, 'Rohith', 9694529843, 'DELHI', 'ACCIDENT');

INSERT INTO PATIENT VALUES (5, 'Bindu', 9687529521, 'MUMBAI', 'TOOTH PROBLEM');

INSERT INTO PATIENT VALUES (6, 'srinadh', 8594529843, 'BHOPAL', 'JOINT DISLOCATE');

SELECT \* FROM PATIENT;

**Table

Description automatically generated**

**Rooms table:**

CREATE TABLE ROOM (ROOMID NUMBER PRIMARY KEY, RTYPE VARCHAR(30), PRICE REAL);

INSERT INTO ROOM VALUES (101, 'GENERAL WARD', 3000);

INSERT INTO ROOM VALUES (102, 'GENERAL WARD AC', 5000);

INSERT INTO ROOM VALUES (103, 'SHARING WARD', 7000);

INSERT INTO ROOM VALUES (104, 'SHARING WARD AC', 9000);

INSERT INTO ROOM VALUES (105, 'SPECIAL WARD', 10000 );

INSERT INTO ROOM VALUES (106, 'SPECIAL WARD AC', 12000);

SELECT \* FROM ROOM;

**Table

Description automatically generated**

**Department table:**

create table Department(did Number primary key, dname char(30), location char(30));

insert into Department values(101, 'surgical department','vizag');

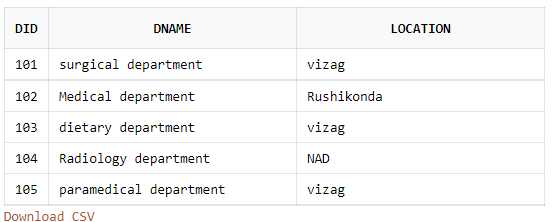
insert into Department values(102, 'Medical department','Rushikonda');

insert into Department values(103, 'dietary department','vizag');

insert into Department values(104, 'Radiology department','NAD');

insert into Department values(105, 'paramedical department','vizag');

select \* from Department;

****

**Blood Bank table:**

create table Bloodbank(bloodid int primary key, bgroup char(30));

insert into Bloodbank values(1,'O positive');

insert into Bloodbank values(2,'B positive');

insert into Bloodbank values(3,'A positive');

insert into Bloodbank values(4,'O Negative');

insert into Bloodbank values(5,'B Negative');

insert into Bloodbank values(6,'A Negative');

select \* from Bloodbank;

**Table

Description automatically generated**

**Checks in:**

create table checks(DID int, PID int primary key,Foreign key(DID) references DOCTOR, Foreign key(PID) references PATIENT);

insert into checks values (1001,1);

insert into checks values (1002,3);

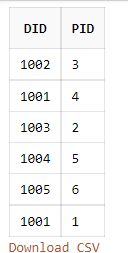
insert into checks values (1001,4);

insert into checks values (1003,2);

insert into checks values (1004,5);

insert into checks values (1005,6);

select \* from checks;

****

**Admitted:**

create table admitted (ROOMID NUMBER ,PID NUMBER,primary key(ROOMID,PID),foreign key(ROOMID) references ROOM, foreign key(PID) references PATIENT);

insert into admitted values (101,1);

insert into admitted values (102,3);

insert into admitted values (103,2);

insert into admitted values (104,4);

insert into admitted values (105,5);

insert into admitted values (106,6);

select \* from admitted;

Table

Description automatically generated with medium confidence

**Works:**

create table works (deid Number, DID Number,primary key(DID,deid),Foreign key(deid) references Department,Foreign KEY(DID) references DOCTOR);

insert into works values(101,1001);

insert into works values(101,1002);

insert into works values(102,1003);

insert into works values(103,1004);

insert into works values(104,1005);

select \* from works;

Table, calendar

Description automatically generated

**Suggest:**

create table suggest (DID Number,bloodid int, primary key(DID,bloodid), FOREIGN KEY(DID) references DOCTOR, FOREIGN KEY(bloodid) references Bloodbank);

insert into suggest values (1001,1);

insert into suggest values (1002,1);

insert into suggest values (1001,2);

insert into suggest values (1003,4);

insert into suggest values (1004,3);

insert into suggest values (1005,6);

select \* from suggest;

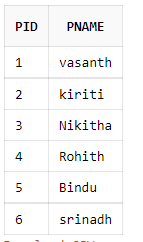
A picture containing table

Description automatically generated

**QUERY**

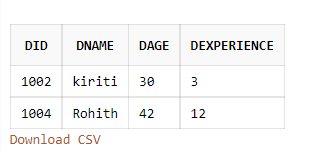
1. Print the patient id and patient name form patient table

select PID,PNAME FROM PATIENT;



1. Print the details of a doctor whose age is less than 45

select \* from DOCTOR where DAGE<45;



1. Print the patient who is suffering from BROKEN FINGER

select \* from PATIENT where DISEASE ='BROKEN FINGER';

Graphical user interface, table

Description automatically generated

1. Print the patient name and patient id who is checked by doctor vasanth

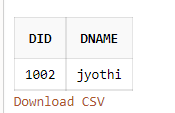
select p1.PID,p1.PNAME from PATIENT p1,DOCTOR d1, checks c1 where p1.PID=c1.PID and c1.DID=d1.DID and d1.DNAME='raju';

Graphical user interface, application, table

Description automatically generated

1. Print the doctor name and doctor id whose experience =3

select DID , DNAME from DOCTOR where DEXPERIENCE=3;



1. Print the room id and patient id and patient name who have taken special ward

select r.ROOMID,p.PID,p.PNAME from ROOM r,PATIENT p, admitted a where r.ROOMID=a.ROOMID and p.PID=a.PID and RTYPE='SPECIAL WARD';

Graphical user interface, application

Description automatically generated

1. Select the doctor who is related to surgical department.

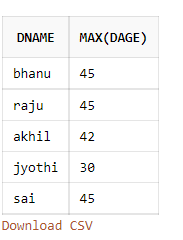
select \* from DOCTOR d,DEPARTMENT d1,works w where d.DID=w.DID and w.deid=d1.deid and d1.dname='surgical department';

Graphical user interface

Description automatically generated with medium confidence

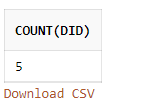
1. Select age and name of doctors using group by

select DNAME,MAX(DAGE) FROM DOCTOR group by DNAME;



1. List the number of doctors in doctor table

select count(DID) FROM DOCTOR;



1. Display all the doctors who have suggested A positive blood group.

select \* from DOCTOR d,Bloodbank b,suggest s where d.DID=s.DID and s.bloodid=b.bloodid and bgroup='A positive';

Graphical user interface, text, application, chat or text message

Description automatically generated

1. Sort the patient table according to their name.

select \* from PATIENT order by PNAME;

Table

Description automatically generated

1. Delete the patient who joined in room no 105

select PNAME from ROOM r,PATIENT p, admitted a where r.ROOMID=a.ROOMID and p.PID=a.PID and r.ROOMID=105;

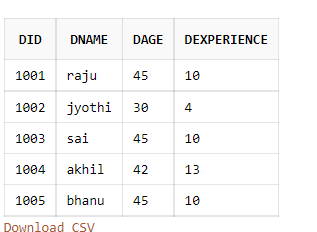
Graphical user interface, application, Word

Description automatically generated

1. update the experience of all doctors by 1

update DOCTOR SET DEXPERIENCE=DEXPERIENCE+1;

SELECT \* FROM DOCTOR;



1. create a view for doctors table

create view DOC as select DID,DNAME from DOCTOR ;

select \* from DOC;

Graphical user interface

Description automatically generated with medium confidence