DBI Exam 16/02/2022 Brandon Wright psybw7

1

A) Make an Entity/Relationship model for the above description and clearly identify entities, attributes, relationships, cardinality ratios.

Hospital has doctors (many) and that doctor has nurses that report to him. Nurses can provide procedures. Procedures are part of treatments, said treatments are part of the clinic and clinics are part of the hospital. Patients request treatments.

hospital -> clinics -> treatments -> procedures

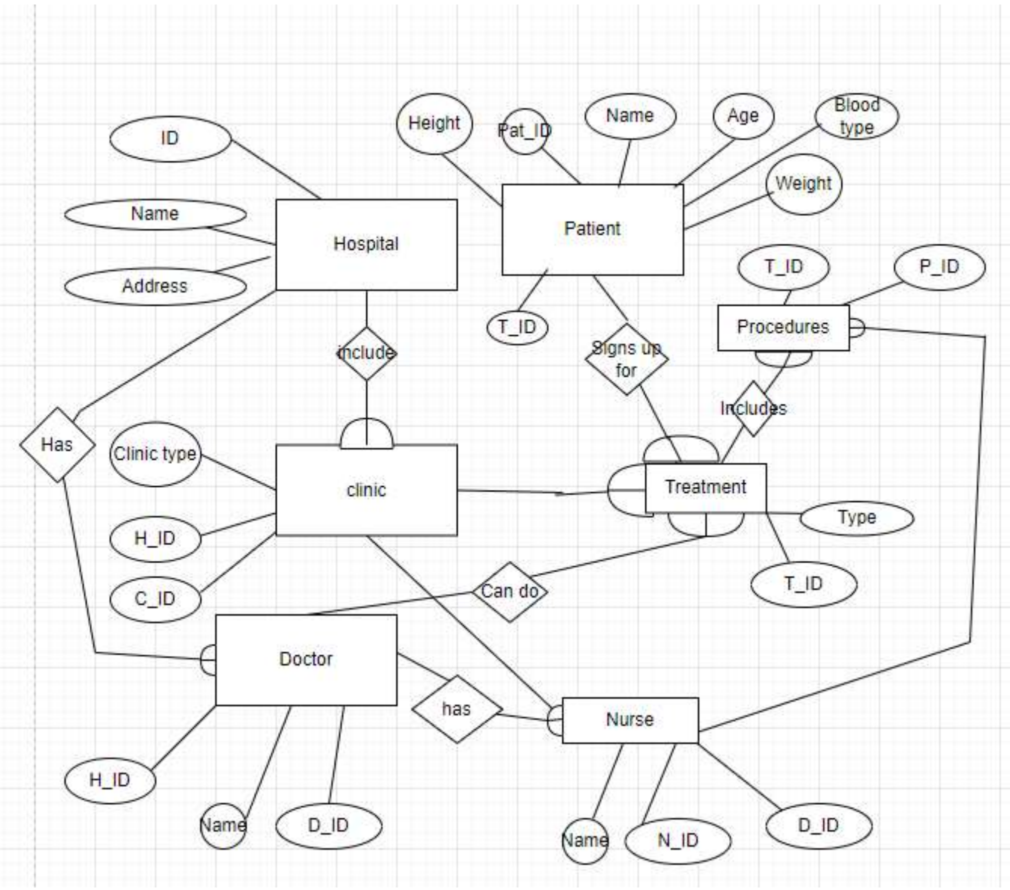
hopsital -> doctors

hospital -> nurses

doctor -> treatment

doctor -> nurses

B)



C) Implement SQL statement(s) that will create the relevant tables in the Hospital database

```
 CREATE TABLE Hospital (

        H_ID INT IS NOT NULL AUTO_INCREMENT,

        Name varchar(255) IS NOT NULL,

        Address varchar(255) IS NOT NULL

        PRIMARY KEY(H_ID)

);

CREATE TABLE Clinic(

        Clinic_TYPE varchar(255) IS NOT NULL AUTO_INCREMENT,

        H_ID INT IS NOT NULL,

        C_ID INT IS NOT NULL,

        PRIMARY KEY(C_ID)

        FOREIGN KEY (H_ID) REFERENCES Hospital (H_ID)

);

CREATE TABLE Doctor(

        Name varchar(255) IS NOT NULL ,

        D_ID INT IS NOT NULL AUTO_INCREMENT,

        H_ID INT IS NOT NULL,

        PRIMARY KEY(D_ID)

        FOREIGN KEY (H_ID) REFERENCES Hospital (H_ID)

);
```

```
CREATE TABLE Nurse(

            Name varchar(255) IS NOT NULL ,

            N__ID INT IS NOT NULL AUTO_INCREMENT,

            D_ID INT IS NOT NULL,

            PRIMARY KEY(N_ID)

            FOREIGN KEY (D_ID) REFERENCES Doctor(D_ID)

);

CREATE TABLE Treatment(

            T_ID INT IS NOT NULL AUTO_INCREMENT,

            D_ID INT IS NOT NULL,

            Type varchar(255)

            PRIMARY KEY(T_ID)

            FOREIGN KEY (D_ID) REFERENCES Doctor(D_ID)

);

CREATE TABLE Procedures(

            P_ID INT IS NOT NULL AUTO_INCREMENT,

            T_ID INT IS NOT NULL,

            PRIMARY KEY(P_ID)

            FOREIGN KEY (T_ID) REFERENCES Treatment(T_ID)

);

CREATE TABLE Patient(

            Pat_ID INT IS NOT NULL AUTO_INCREMENT,

            T_ID INT IS NOT NULL,

            Name varchar(255),

            Age INT IS NOT NULL,

            PRIMARY KEY(P_ID),

            Blood_type varchar(255),

            Weight INT IS NOT NULL,

            FOREIGN KEY (T_ID) REFERENCES Treatment(T_ID)

);
```

Implement SQL statement(s) that will populate the relevant information into the appropriate tables for each of the below actions:

D)

```
INSERT INTO Patient (Name,Age,Blood_type,weight,T_ID)
Values(“Harry Lemon”,34,”O”,14,1) <- this assumes the treatment table is already populated
```

F)

```
INERT INTO Procedures(T_ID)
Values(1) <- this assumes the treatment table is already populated
```

G)

```
INSERT INTO Nurse(Name,D_ID)
VALUES(“Harriet Jones”,1) <- this assumes the treatment table is already populated
```

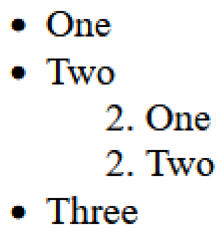
H)

```
INSERT INTO Clinic(Clinic_type,H_ID,C_ID)
VALUES(“ER”,1) <- this assumes the treatment table is already populated
```

```
SQL DELETE FROM Patient WHERE Name=”Harry lemon”;
```

2

A) Implement HTML code that displays the below lists.



- One
- Two
    2. One
    2. Two
- Three

```
<html>
<head>
<title>
hello world
</title>
</head>
<body>
<ul>
  <li>One</li>
  <li>Two</li>
            <ol>
                        <li>One</li>
                        <li>Two</li>
            </ol>
  <li>Three</li>
</ul>
</body>
</html>
```

b)

Implement HTML code that displays the table below.

```
<table>
  <tr>
    <th>00</th>
  <th>00</th>
  <th>00</th>
  <th>00</th>
  <th>00</th>
 </tr>
 <tr>
  <td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
 </tr>
 <tr>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
 </tr>
  <tr>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
 </tr>
  <tr>
<td>00</td>
<td>00</td>
<td>00</td>
<td>00</td>
```

```
<td>00</td>
  </tr>
</table>
```

B) Using CSS cascade hierarchy implement styles so that in the HTML document below the default font colour is red, the default paragraph font colour is green, and the first paragraph font colour is orange. The desired output appears after the document.

```
<!DOCTYPE html>
<html>
<head>

<style>

body{
color:red

}

p{
color:green}

</style>
</head>
<body>
This is some text.
<p style="color:orange">Text 1</p>
<p>Text 2</p>
<p>Text 3</p>
<p>Text 4</p>
<p>Text 5</p>
This is some more text.
</body>
</html>
```

```
d) <!DOCTYPE html>
<html>
<head>

<title> Radio - Hello world Brandon Wright </title>

<script>

function parseBG(form){
    let x = 0;
    for(var i = 0; i < form.length; i++){
        if(form[i].checked == true){
            x = i
            break
        }
    }
    if(x == 0){ // day

        document.getElementById("body").style.color="black"
        document.getElementById("body").style.backgroundColor="white"

    }else{ // night

        document.getElementById("body").style.color="white"
        document.getElementById("body").style.backgroundColor="black"
    }
    event.preventDefault()

    return false;
}
```

```
console.log("Hello world")

 </script>


</head>

<body id="body">

<form onsubmit="parseBG(this)">

<input type="radio" id="day"> Day </input>

<br/>

<input type="radio" id="night"> Night </input>

<br/>

<input type="submit" value="Submit">

</form>


</body>


</html>
```

E) Implement HTML and JavaScript code that displays 3 text fields and a button. When a user inputs a number in the first text field and a number in the second text field and then clicks the button, the third text field should display the average of the two numbers.

(

```
<!DOCTYPE html>

<html>

<head>


<title> Radio - Hello world Brandon Wright </title>


<script>


function parseNums(form){


   x = parseInt(form[0].value)
```

```
    y = parseInt(form[1].value)


    form[2].value = (x + y) / 2

    event.preventDefault()

}


console.log("Hello world")

 </script>


</head>

<body id="body">

<form onsubmit="parseNums(this)">

<input type="text" id="day"> Number 1 </input>

<br/>

<input type="text" id="night"> Number 2</input>

<br/>

<input type="text" id="night"> Result (DONT MODIFY THIS)</input>

<br/>

<input type="submit" value="Submit">

</form>


</body>


</html>
```

f) Add JavaScript code to the below HTML so that the text in even rows (i.e. second and fourth rows) is green while the text in odd rows (i.e., first and third rows) is brown.
(5)

```
<!DOCTYPE html>

<html>

<head>
```

```
</head>
<body>
<table id="tab">
<tr><td class="person">Anna</td></tr>
<tr><td class="person">Bob</td></tr>
<tr><td class="person">Chloe</td></tr>
<tr><td class="person">Darren</td></tr>
</table>
</body>
</html>

<script>
    var tab = document.getElementById("tab");
    for (var i in tab.rows) {
        if(i % 2 == 0){
            tab.rows[i].style.color = "green"
        }else{
            tab.rows[i].style.color = "brown"
        }

}

</script>
```

g) Add JavaScript to the HTML code below so that the button acts as an on/off switch, i.e., when the HTML first gets displayed the button value is “0”. When the button gets clicked it should change to “1” and when it is gets clicked again it should change to “0”.

```
<!DOCTYPE html>
<html>
<body>
<input type="button" id="count" value="0">
</body>
```

```
</html>

<script>

document.getElementById("count").onclick = function() {
   let x = document.getElementById("count").value
   if(x == "0"){
      document.getElementById("count").value = 1
   }else{
      document.getElementById("count").value = 0
   }

}

</script>
```

h) Add JavaScript code to the below HTML so that the paragraphs text is sorted in reverse order.

```
<!DOCTYPE html>
<html>
<body>
<h1>Reverse sorting</h1>
<p class="person">Anna</p>
<p class="person">Chloe</p>
<p class="person">Bob</p>
<p class="person">Darren</p>
</body>
</html>

<script>
   const people = document.getElementsByClassName("person");
   let arr = []
```

```
  for(let i = people.length-1; i >= 0 ; i--){


    arr.push(people[i])

  }

  for(let i = 0 ; i <= arr.length; i++){

    arr[i].remove()

    let t = document.createElement("p")

    t.appendChild(arr[i])

    document.getElementsByTagName("body")[0].appendChild(t);

  }

</script>
```

3

A) Implement a PHP script that will output a complete, structurally correct simple “Hello World” HTML page with CSS formatting for one element.

```
<html>
 <head>
  <title>PHP Test</title>
 </head>
 <body>
 <?php echo '<p style=”color:red”>Hello World</p>'; ?>
 </body>
</html>
```

b) Implement a PHP counter script that will output the first five even numbers as a complete, structurally correct HTML listing, one number per line.

```
<html>
 <head>
  <title>PHP Test</title>
 </head>
 <body>
 <?php

$counter = 0;

$f = 0;

while($counter < 5) {

  if($f % 2 == 0){

              echo ‘<p>Number ’.$f.’</p><br/>’

}
  $f++;


}

?>
 </body>
</html>
```

Look at the below HTML form. Implement a PHP script “month.php” that will read the value of “number” and generate a structurally correct HTML code displaying the name of the month it corresponds to, i.e. when “4” is submitted, “April” should be displayed.

If the user submits an invalid month value, then an appropriate error message should be displayed.

(

```
<!DOCTYPE html>

<html>

<body>

<?php


              $title = $_POST['name'];

              if($title <=0 || $title >=13){

                            echo “Wrong value submitted!”
}else{

$months = array("January", "February", "March", "April",”May”,”June”,”July”,”August”,”September”,”October”,”November”,”December”);

Echo $months[title]

}


?></body>

</html>
```