



## Information filled by student:

### Course basic information

Code	Course Name	Credit Hours		
		Lecture	Practice	Total
IS212	Database	2	2	3

### Research Title

(Hotel Reservation System)

**Student Name: Ahmed Hassan Ali Ahmed**

**Student ID: 16120180100193**

**Level: Second Level**

**Department: General**



## Table of Contents.

- Project title .....	(1)
- Table of contents .....	(2)
- System description .....	(3)
- List of tables in Hotel Reservation System database.....	(4) (5)
- Entity relationship Diagram .....	(6)
- Select Statements using Different Functions.....	(7)(8)
- Select statements using sub query .....	(9)
- Statements of count and Group functions.....	(9)
- Select Statements using Different Joins .....	(10)
- Insert statements .....	(11)
- Update statements .....	(12)
- Delete statements .....	(13)
- References .....	(14)

## System Description.

It is known that the best way to learn something is to practice it in a realistic scenario. Obviously, the same thing applies to creating a database. In light of this text, we will discuss creating a database structure for the hotel reservation system,

First of all, we must know what are the goals of the hotel reservation systems in general. The answer to this question includes many answers, including the following.

- To increase direct bookings for the hotels.
- To offer easy, simple and quick booking experience to the guests.
- To offers tools to hotels so that they can effectively communicate with the guests.
- View of availability of rooms in real time.
- To provide up-to-date information on the status of reservations.
- Efficient reservation, especially for frequent clients.

This system that we will explain in detail consists of 4 tables.

They are designed to facilitate data storage in terms of storage and interface.

- The first table called (Employees) contains 6 columns, the first column of them contains the primary key.
- The second table is called (Guest) and contains 5 columns. The first column of them contains the primary key.
- The third table is called (Room Type) and contains 3 columns, of which the first contains the primary key.
- The fourth and last table is called (Reservation) and contains 5 columns, the first and the second of them representing the foreign key.

## List of tables in Hotel Reservation System database.

This system consists of four main tables and also contains only two sub-tables designed to implement the conditions in the elements.

### Employees (Basic)

Field Name	Data Type	Null	Kay	Default	Extract
EmployeeID	Int	No	PRI	Null	
First Name	Varchar	No		Null	
Last Name	Varchar	No		Null	
Department	Varchar	No		Null	
Salary	Decimal	Yes		Null	
Phone	Varchar	No		Null	

### A sub of Employees It contains fake data only to apply the conditions (subsidiary)

EmployeeID	First Name	Last Name	Department	Salary	Phone
30	Ahmed	Saad	HR	1000	17556221
34	Waheed	Saad	OB	3000	1755644
44	Osama	Saad	MR	2000	1755630
66	Shawky	Saad	MH	6000	17556123
84	Saad	Saad	RH	5000	1755622
94	Adhm	Saad	SE	4000	1755633

### Guest (Basic)

Field Name	Data Type	Null	Kay	Default	Extract
Guest ID	Int	No	PRI	Null	
First Name	Varchar	No		Null	
Last Name	Varchar	No		Null	
Email	Varchar	No		Null	
PhoneNamber	Varchar	No		Null	

**A sub of Guest It contains fake data only to apply the conditions (subsidiary)**

Guest ID	First Name	Last Name	Email	PhoneNumber
59	Baher	Altawel	Baher.com	20218245
90	Shady	Fathy	Shady.com	20214520
98	Rady	Altawel	Rady.com	20218746
99	Ali	Altawel	Ali.com	20218213
239	Nail	Altawel	Nail.com	202182788
636	Shaker	Altawel	Shaker.com	20218745

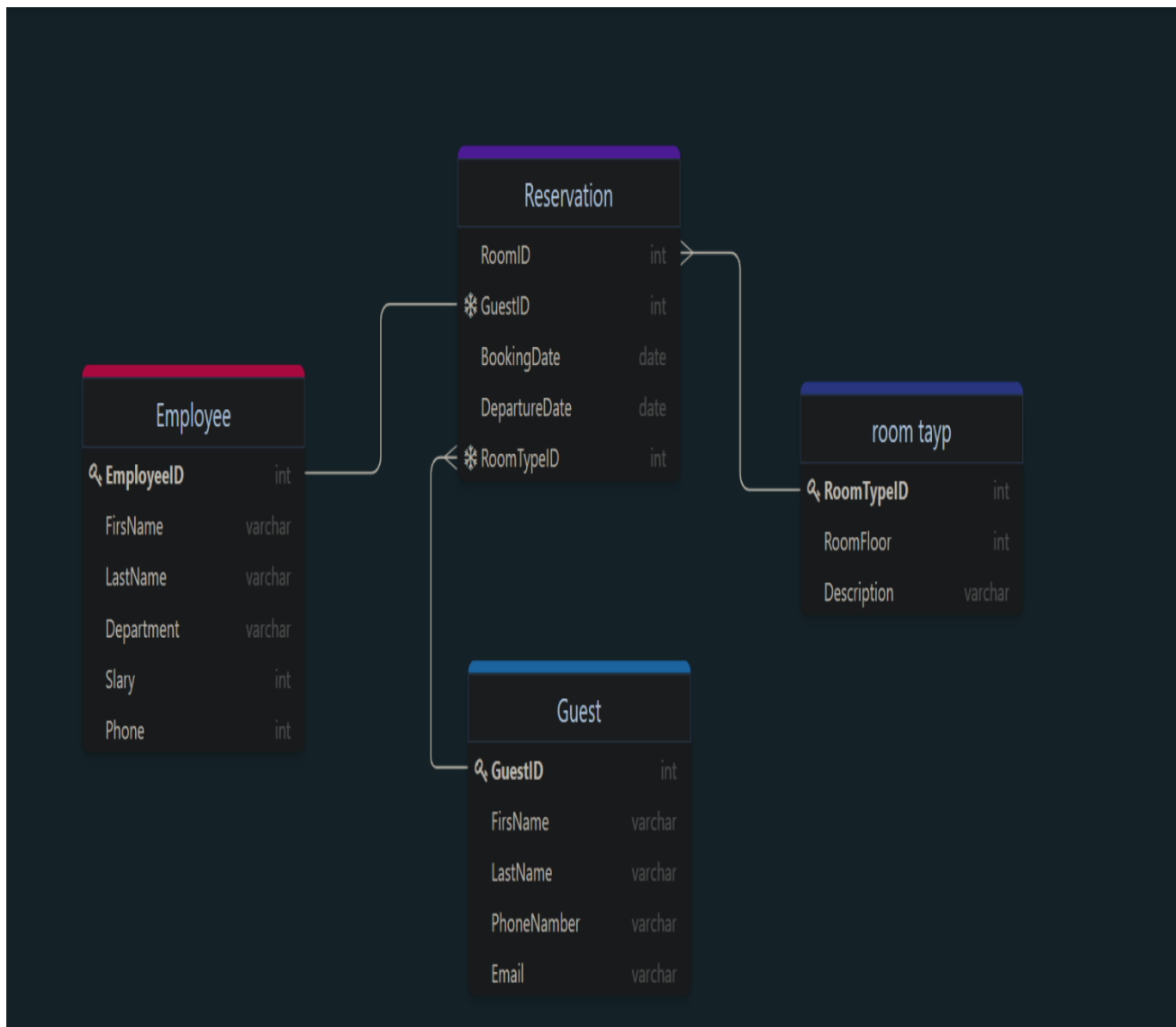
### **Room Type (Basic)**

Field Name	Data Type	Null	Kay	Default	Extract
RoomTypeID	Int	No	PRI	Null	
Room Floor	Varchar	No		Null	
Description	Varchar	Yes		Null	

### **Reservation (Basic)**

Field Name	Data Type	Null	Kay	Default	Extract
Guest ID	Int	No	MUL	Null	
RoomTypeID	Int	No	MUL	Null	
Room ID	Int	No		Null	
Booking Date	Date	No		Null	
DepartureDate	Date	No		Null	

## Entity relationship Diagram.



## Select Statements using Different Functions.

1. Display records of Employees table  
select \* from Employees;
2. Display records of Guest table  
select \* from Guest;
3. Find the first name of Employees  
Select firstname from Employees;
4. Find the first name and Last name of Employees  
Select firstname, Lastname from Employees;
5. Find the first name of Guest whose First name starts with 'S'  
Select \* From Guest Where Firstname like 'S%';
6. Find the first name and salary of Employees whose Who has a salary greater than 2000  
Select firstname, salary from Employees Where salary>2000
7. Find the first name and salary of Employees whose They work in a department (HR)  
Select firstname, salary from Employees Where department='HR'
8. Find all the employees working in a department (MR)  
Select \* from employees Where department='MR'
9. Find the salary of Employees whose Their salaries range from 3,000 to 5,000  
Select \* from employees Where salary>=3000 and salary<=5000
10. Find the EmployeeID of Employees whose ID is 34  
Select \* from employees Where EmployeeID=34

11. Find the first name of Employees named 'ahmed'  
Select firstname From Employees Where firstname='ahmed';
12. Find a list of all EmployeeIDs  
Select EmployeeID From Employees;
13. Find a list of all employees 'phone numbers  
Select Phone from Employees;
14. Find the all of Employees who's in their first name there is a letter'S'  
Select \* From Employees Where Firstname like '%S%';
15. Find all employees whose second letter from his name is "a"  
Select \* From Employees Where Firstname like '\_a%';
16. Find all employees whose third letter from his name is "a"  
Select \* From Employees Where Firstname like '\_\_a%';
17. Find the Last name of Guest named 'Fathy '  
Select \* From Guest Where lastname= 'Fathy';
18. Find the GuestID of Employees whose ID is 636  
Select \* from Guest Where GuestID =636
19. Find the first and last name of the guest who owns an email named 'Nail.com'  
Select firstname, lastname from Guest Where Email = 'Nail.com'
20. Find the first and last name whose have Telephone number '20218213' in Guest  
Select first name, last name from Guest Where PhoneNumber =' 20218213'





### **Select statements using sub query.**

1. Find all employees whose salary is 4,500 or greater

```
SELECT * FROM Employees
WHERE EmployeeID IN (SELECT EmployeeID
FROM Employees WHERE salary > 4500);
```

2. Find the lowest salary taken by an employee

```
SELECT First Name, Last Name, Department FROM Employees
WHERE salary = (SELECT MIN (salary) from Employees);
```

3. Find the largest salary taken by an employee

```
SELECT First Name, Last Name, Department FROM Employees
WHERE salary = (SELECT Max (salary) from Employees);
```

### **Select Statements using Count and Group Functions.**

1. Find the number of each employee in each department

```
SELECT COUNT(EmployeeID), Department
FROM Employees
GROUP BY Department;
```

2. Find a list of all employee codes and their salary in order

```
SELECT COUNT(EmployeeID), salary
FROM Employees
GROUP BY salary
ORDER BY COUNT(EmployeeID) DESC;
```

## Select Statements using Different Joins.

1. Collect all the data that are shared between the room type and the date of reservation.

```
SELECT Room Type. RoomTypeID, Room Type. Reservation.  
Booking Date FROM (SELECT `RoomTypeID`, `Room Type`) AS Room Type  
JOIN (SELECT RoomTypeID, Booking Date) AS Reservation  
ON Room Type. RoomTypeID = Reservation. Booking Date
```

2. Find all data of employees who registered all guests.

```
SELECT Guest. GuestID, Employees. EmployeeID, Guest. FirstName  
FROM Guest INNER JOIN Employees ON Guest. EmployeeID=Employees.  
EmployeeID;
```

3. Find all data for the guest who booked his room code 50.

```
SELECT GuestID FROM Guest JOIN Reservation  
ON GuestID = Room ID WHERE Room ID=50;
```

4. Return all records from the left table (`Guest`) and matching records from the right table (Employees), if any.

```
SELECT Guest FROM Employees  
LEFT JOIN Guest ON Employee. First name = Guest. Employee id;
```

5. Return all records from the right table (Employee) and matching records from the right table (Guest).

```
SELECT Employee id FROM Guest  
RIGHT JOIN Employee ON Guest. Employee id = Employee. first name;
```

## Insert Statement.

1. Add data for a new employee in the employee list. The employee called Ahmed Saad works in a HR department and takes 1000 salary and his phone number 17556221.  
insert into Employees (Employee ID, Firs Name, Last Name, Department , salary, Phone) values (30, Ahmed, Saad, HR,1000,017556221);
2. Add data for a new employee in the employee list. The employee called waheed Saad works in an OB department and takes 3000 salary and his phone number 1755644.  
insert into Employees (Employee ID, Firs Name, Last Name, Department, salary, Phone) values (34, waheed, Saad, OB,3000,1755644 );
3. Add data for a new employee in the employee list. The employee called Osama Saad works in an MR department and takes 2000 salary and his phone number 1755630.  
insert into Employees (Employee ID, Firs Name, Last Name, Department, salary, Phone) values (44, Osama, Saad, Mr,2000,1755630 ); select \* from Employees;
4. Add data for a new employee in the employee list. The employee called Shawky Saad works in an MH department and takes 6000 salary and his phone number 1755630.  
insert into Employees (Employee ID, Firs Name, Last Name, Department, salary, Phone) values (44, Shawky, Saad, Mh,6000,17556123 ); select \* from Employees;
5. Add data for a new customer in the guest schedule. The customer claims Shade Fathy and has an email address (Shady.com) and his phone number is 20214520.  
insert into Guest (Guest ID, FirstName, last Name, Email, Phone Number) values (90, shady, Fathy, shady.com,20214520); select \* from Guest;



## Update Statement.

1. Increase the salary of the employee, who claims to be Shawky, to 3300.

```
UPDATE Employees SET salary = 3300
```

```
WHERE Firs Name = Shawky;
```

```
select * from Employees;
```

2. Increase the salary of the employee with a job code that is 84 to 10,000.

```
UPDATE Employees SET salary= 10000
```

```
WHERE Employee ID = 84;
```

```
select * from Employees;
```

3. Increase all staff working in the HR department to 15,000.

```
UPDATE Employees SET salary= 15000
```

```
WHERE Department = HR;
```

```
select * from Employees;
```

4. Update the data of all employees working in the (HR) department in terms of salary increase to 18,000 and change the name of the department to (Human Resources).

```
UPDATE Employees SET salary= 18000, Department=Human Resources
```

```
WHERE Department= HR;
```

```
select * from Employees;
```

5. Update all employees in terms of salary to 20000.

```
UPDATE Employees
```

```
SET salary= 20000;
```

```
select * from Employees;
```



## Delete Statement.

1. Delete all employee data with code 66.

```
delete from Employees
where Employee ID= 66;
select * from Employees;
```

2. Delete all data of the employee who claims to be Saad.

```
delete from Employees
where Firs Name= Saad;
select * from Employees;
```

3. Delete the guest's email address (Nail.com).

```
delete from Guest
where Email= Nail.com;
select * from Guest;
```

4. Delete the data for employees whose salary is 2000 or smaller.

```
delete from Employees
where salary>2000;
select * from Employees;
```

5. Delete the data of employees working in a department='MH'

```
delete from Employees
where Department=MH;
select * from Employees;
```

## References.

**much of contents of the topic is derived from:**

1. Database Book
2. [https://www.w3schools.com/sql/sql\\_join.asp](https://www.w3schools.com/sql/sql_join.asp)
3. [https://www.w3schools.com/sql/sql\\_groupby.asp#:~:text=The%20SQL%20GROUP%20BY%20Statement,by%20one%20or%20more%20columns.](https://www.w3schools.com/sql/sql_groupby.asp#:~:text=The%20SQL%20GROUP%20BY%20Statement,by%20one%20or%20more%20columns.)
4. [www.w3schools.com](http://www.w3schools.com)
5. <https://www.w3resource.com/sql/aggregate-functions/count-with-group-by.php>
6. [www.w3resource.com](http://www.w3resource.com)
7. [https://www.w3schools.com/sql/sql\\_insert.asp](https://www.w3schools.com/sql/sql_insert.asp)

**Git-Hub Repository Link.**