

Exercise Number 3

Title of the Exercise Hospital

Date of the Exercise

Create a table Hospital with the fields (doctor_id, doctor_name, department, qualification, experience).

Write the queries to perform the following.

- a) Insert 5 records
- b) Display the details of doctors
- c) Display the details of doctors who have the qualification 'MD'
- d) Display all doctors who have more than 5 years experience but do not have the qualification 'MD'
- e) Display the doctors in 'Skin' department
- f) Update the experience of doctor with doctorid = 'D003' to 5
- g) Delete the doctor with DoctorID = 'D005'

Table Structure

Table name : hospital

Sl.No	Field Name	Data type	constraints
1	doctor_id	varchar(5)	not null, unique
2	doctor_name	varchar(15)	
3	department	varchar(15)	
4	qualification	varchar(10)	
5	experience	integer	

Commands

Create table hospital(doctor_id varchar(5) not null, unique,
doctor_name varchar(15), department varchar(15),
qualification varchar(10), experience integer);

CREATE TABLE

- a) Insert into hospital values ('D001', 'Aslam', 'Skin', 'MBBS', 7), ('D002', 'Adam', 'ENT', 'MBBS, MD', 7), ('D003', 'Nikhil', 'cardiology', 'MBBS', 4), ('D004', 'Ramesh', 'Skin', 'MBBS, MD', 6), ('D005', 'Smitha', 'neurology', 'MBBS', 5);

Insert 05

b) select * from hospital;

doctor-id	doctor-name	department	qualification	experience
D001	Aslam	Skin	MBBS	7
D002	Adam	ENT	MBBS, MD	7
D003	Nikhil	cardiology	MBBS	4
D004	Ramesh	Skin	MBBS, MD	6
D005	Smitha	neurology	MBBS	5

c) select * from hospital where (qualification like '%MD');

doctor-id	doctor-name	department	qualification	experience
D002	Adam	ENT	MBBS, MD	7
D004	Ramesh	Skin	MBBS, MD	6

d) select * from hospital where (experience > 5 and qualification not like '%MD');

doctor-id	doctor-name	department	qualification	experience
D001	Aslam	Skin	MBBS	7

e) select doctor-name from hospital where department = 'Skin';

doctor-name

Aslam

Ramesh

8) Update hospital set experience = '5' where doctor-id = 'D003';

UPDATE 1

Select * from hospital;

doctor-id	doctor-name	department	qualification	experience
D001	Aslam	Skin	MBBS	7
D002	Adam	ENT	MBBS, MD	7
D004	Ramesh	Skin	MBBS, MD	6
D005	Smitha	neurology	MBBS	5
D003	Nikhil	cardiology	MBBS	5

9) Delete from hospital where doctor-id = 'D005';

DELETE 1

Select * from hospital;

doctor-id	doctor-name	department	qualification	experience
D001	Aslam	Skin	MBBS	7
D002	Adam	ENT	MBBS, MD	7
D004	Ramesh	Skin	MBBS, MD	6
D003	Nikhil	cardiology	MBBS	5

Exercise Number : 4

Title of the Exercise : Bank

Date of the Exercise :

Create the following tables

Bank-customer (accno primary key, cust-name, place)

Deposit (accno foreign key , deposit-no, clamount)

Loan (accno foreign key, loan-no, Lamount)

Write the following queries

- a) Display the details of the customers.
- b) Display the customers along with deposit amount who have only deposit with the bank
- c) Display the customers along with loan amount who have only loan with the bank.
- d) Display the customers they have both loan and deposit with the bank.
- e) Display the customers who have neither a loan nor a deposit with the bank.

1	acc-no	integer	not null unique.
2	customer-name	varchar(10)	not null.
3	place	varchar(25)	

Table name : Deposit

Sl. No	Field Name	Datatype	constraints
1	acc-no	integer	foreign key.
2	deposit-no	integer	
3	d.amount	integer	

Table name : Loan.

Sl. No	Field Name	Datatype	constraints
1	acc-no	integer	foreign key.
2	loan.no	integer	
3	l.amount	integer	

Command

Create table bank.customer (acc.no integer, not null unique primary key, cust.name varchar(15) not null, place varchar(25));

CREATE TABLE

Insert into bank.customer values (11101, 'Ram', 'Pulpally'),
(11102, 'Manu', 'Bathery'), (11103, 'Achu', 'Koliyadi'), (11104,
'Hari', 'Kuppadi'), (11105, 'Ann', 'Bathery);

Insert O 5

Select * from bank.customer;

acc.no	cust.name	place.
11101	Ram	Pulpally
11102	Manu	Bathery
11103	Achu	Koliyadi
11104	Hari	Kuppadi
11105	Ann	Bathery.

Create table deposit(acc.no integer references bank.customer
(acc.no), deposit-no integer, cl.amount integer);

CREATE TABLE

Insert into deposit values (11101, 11201, 50000), (11103, 11203, 25000), (11105, 11205, 30000);

Insert 0 3

Select * from deposit;

acc-no	deposit-no	d-amount
11101	11201	50000
11103	11203	25000
11105	11205	30000

Create table loan (acc-no integer references bank-customers (acc-no), loan-no integer, l-amount integer);

CREATE TABLE

Insert into loan values (11101, 11301, 10000), (11102, 11302, 25000), (11105, 11305, 15000);

Insert 0 3

Select * from loan;

acc-no	loan-no	l-amount
11101	11301	10000
11102	11302	25000
11105	11305	15000

- a) select * from bank-customer;
- b) select cust-name, d-amount from bank-customer, deposit where bank-customer.acc-no = deposit.acc-no and deposit.acc-no NOT IN (select acc-no from loan);

<u>cust-name</u>	<u>d-amount</u>
Achu	25000

- c) select cust-name, l-amount from bank-customer, loan where bank-customer.acc-no = loan.acc-no and loan.acc-no NOT IN (select acc-no from deposit);

<u>cust-name</u>	<u>l-amount</u>
Manu	25000

- d) select cust-name from bank-customer, deposit, loan where bank-customer.acc-no = deposit.acc-no AND deposit.acc-no = loan.acc-no;

<u>cust-name</u>
Ram
Ann

- e) select cust-name from bank-customer where bank-customer.acc-no NOT IN (select acc-no from deposit UNION select acc-no from loan);

<u>cust-name</u>
Hari

Exercise Number : 5

Title of the Exercise : Employee

Date of the Exercise :

Create a table employee with fields (EmpID, CName, Salary, Department and Age). Insert some records.

Write SQL queries using aggregate functions and group by clause

- a) Display the total number of employees.
- b) Display the name and age of the oldest employee of each department
- c) Display the average age of employee of each department
- d) Display departments and the average salaries
- e) Display the lowest salary in employee table.
- f) Display the number of employees working in purchase department.
- g) Display the highest salary in sales department.
- h) Display the difference between highest and lowest salary.

Table structure

Table name: employee

Sl.no	Field Name	Data type	constraints
1	Gmp ID	integer	not null, unique
2	EName	varchar(15)	
3	Salary	integer	
4	Department	varchar(15)	not null
5	Age	integer	not null

Command.

Create table employee (GmpID integer not null unique primary key, EName varchar(15), salary integer, department varchar(15) not null, age integer not null);

CREATE TABLE.

Insert into employee values (1001, 'Bopu', 30000, 'sales', 54), (1002, 'Arun', 25000, 'purchase', 49), (1003, 'Ajith', 20000, 'purchase', 25), (1004, 'Varon', 25000, 'sales', 35), (1005, 'Ram', 32000, 'purchase', 58), (1006, 'Kumar', 28000, 'sales', 45);

INSERT O 6

select * from employee;

Emp ID	Employee Name	salary	Department	Age
1001	Gopu	30000	sales	54
1002	Arun	25000	purchase	49
1003	Djith	20000	purchase	25
1004	Varan	25000	sales	35
1005	Ram	32000	purchase	58
1006	Kumar	28000	sales	45

a) select count(*) EmpID from employee;

Emp ID
6

b) select Employee, Age from employee where age in (select max(age) from employee group by department);

Employee	Age
Gopu	54
Ram	58

c) select avg(age) from employee group by department;

avg.
44.000000
44.666667

d) select department, avg (salary) from employee group by department

department	avg.
purchase	25666.666667
sales	27666.666667

e) select min (salary) from employee;

min
20000

f) select count (*) from employee where department = 'purchase';

count
3

g) select max (salary) from employee where department = 'sales';

max
30000

h) select max (salary) - min (salary) difference from employee;

difference
12000

Exercise Number

6

Title of the Exercise

Product

Date of the Exercise :

Create a table product with the fields (Product-Code primary key, Product-Name, Category, Quantity, Price).

Insert some records. Write the queries to perform the following.

- a) Display the records in the descending order of Product Name.
- b) Display Product-Code, Product-Name with price between 20 and 50.
- c) Display the details of products which belongs to the categories of 'bath soap', 'paste' or 'washing powder'.
- d) Display the products whose quantity less than 100 or greater than 500.
- e) Display the products whose names starts with 'S'.
- f) Display the products which not belongs to the category 'paste'.
- g) Display the products whose second letter is 'u' and belongs to the category 'washing powder'.

Table structure

Table name : product

Sl.No	field_Name	Datatype	constraints.
1	Product_code	Integer	Primary key, not null
2	Product_name	varchar (20)	
3	Category	varchar (20)	
4	Quantity	Integer	
5	Price	Integer	

Command.

Create table product (Product_code integer primary key not null, Product_name varchar (20), category varchar (20), quantity integer, price integer);

CREATE TABLE

Insert into product values (1544, 'himalaya', 'facewash', 75, 70), (4565, 'colgate', 'paste', 100, 90), (5543, 'vival', 'bathsoap', 250, 45), (7045, 'riel', 'washing powder', 600, 67), (9054, 'surf exel', 'washing powder', 700, 120);

Insert 05

Select * from product;

Product-code	Product-name	Category	Quantity	Price
1544	himalaya	facwash	75	70
4565	colgate	paste	100	90
5543	vival.	bathsoap	250	45
7045	riel	washing powder	600	67
9054	surf exel	washing powder	700	120

a) Select * from product order by Product-name desc;

Product-code	Product-name	category	Quantity	Price
5543	Vival	bathsoap	250	45
9054	Surf exel	washing powder	700	120
1544	himalaya	face wash	75	70
4565	colgate	paste	100	90
7045	riel	washing powder	600	67

b) select Product-code, Product-name, price from product
where (price > 20 and price < 50);

Product-code	Product-name	Price
5543	Vival	45

c) select * from product where category in ('bathsoap', 'washing powder', 'paste');

Product-code	Product-name	Category	Quantity	Price
4565	colgate	paste	100	90
5543	vival.	bathsoap	250	45
7045	riel	washing powder	600	67
9054	surf excel	washing powder	700	120

d) select product-name from product where (Quantity < 100
or quantity > 500);

Product-name
himalaya
riel
surf excel.

e) select product-name from product where product-name like '%s';

Product-name
surf excel

6) select product-name from product where category
not like 'paste';

Product-name
himalaya
vival
riel
Surf excel.

9) Select product-name from product where product name
like '-u%' and category = 'washing powders';

Product-name
Surf excel.