LIST OF ALL QUERY FROM THE PPT

FOR EXCERCISE -4 (question - 4)

Database Management Systems , Unit-III, Lecture 2 Introduction to SQL

Query 1. create table

Query 2. Adding primary key foreign key

Query 3. Updates to tables [insert]

Query 4 :- Delete

Query 5:- Drop Table

Query 6:- Alter (alter table _____ add ____)

Query 7:- alter table r drop A

END OF THE PPT(1st)

Database Management Systems , Unit-III, Lecture 3 Introduction to SQL

Query 8:- select _____ from _____ where_____

Query 9:- select _____ from _____

Query 10 :- select distinct _____ from _____

Query 11:- select all _____ from _____

Query 12:- select * from instructor

Query 13 :- select '437

Query 14 :- select '25MCMC34 ' as SHIVAM

Query 15:- select 'A' from _____;

Query 16 :- select ____, ____, salary/12 from _____;

Query 17:- select ___ , ____ ,salary/12 as monthly_salary From _____;

Query 18:- select _____ from _____ where _____= '_____'.

Query 19:- select ____ from _____ where _____= '___.' and salary > 70000

Query 20:- select from_____, _____;

Query 21:- select ____ , _____ from _____ , _____ where _____.____ = _____._____;

Query 22:- select ____,_____ from ____ , _____ where _____.__= ____.____ and _____._____ = '____';

END OF THE PPT (2nd)

Database Management Systems , Unit-III, Lecture 4 Introduction to SQL

Query 23 :- select A.___, B.___ from ____ as A, _____ as B where A.___= B.___;

Query 24 :- #Self join

select distinct A. $$$ from $$$ as A, $$$ as B where A.@@@ > b.@@@ and A._____ ='_____ ';

Query 25 :- upper(s) -- converting strings to uppercase

Query 26:- Lower(s) -- converting strings to lowercase

Query 27:- TRIM

Query 28:- concatenation (using "||")

Query 29:- select ____from ____ where _____ like '%$$$%

Query 30 :- USING MATCHONG PATTERN ['@@@%']

Query 31 :- USING MATCHING PATTERN ['%@@@%']

Query 32 :- USING MATCHING PATTERN ['_ _ _']

Query 33 :- USING MATCHING PATTERN ['_ _ _ _ _%']

Query 34 :- USING not like keyword ['_ _ _ _ _%']

Query 35 :- select distinct _____from _____ order by ____

Query 36 :- select distinct _____from _____ order by ____ DESC;

Query 37 :- select _____from ____ order by ____ , _____;

Query 38 :- select _____from ____ order by ____ ASC , _____ DESC;

Query 39 :- select _____ from _____ where salary between 1200000 and 1800000

Query 40 :- not between comparison operator

Query 41 :- select ____ , ____ from ____, ____ where (____ . ____ , ___) = (____ . ___, '____');

Query 41 :- UNION OPERATION

Query 42 :- INTERSECTION OPERATION

Query 43 :- EXCEPT OPERATION

END OF THE QUERY UPTO Set Operations - continued

(IF ANY QUERY I MISSED PLEASE ADD YOURSELF AND ALSO INFORM ME)

CONTENT COPIED FROM THE PPT

- create table instructor (ID char(5), name varchar(20), dept_name varchar(20), salary numeric(8,2))
- create table instructor (ID char(5), name varchar(20) not null, dept_name varchar(20), salary numeric(8,2), primary key (ID), foreign key (dept_name) references department)
- create table department (dept name varchar (20), building varchar (15), budget numeric (12,2), primary key (dept_name))
- create table course (course_id varchar (7), title varchar (50), dept name varchar (20), credits numeric (2,0), primary key (course_id), foreign key (dept_name) references department)
- create table section (course_id varchar (8), sec_id varchar (8), semester varchar (6), year numeric (4,0), building varchar (15), room_number varchar (7), time_slot_id varchar (4), primary key (course_id, sec_id, semester, year), foreign key (course_id) references course)

- create table teaches (ID varchar (5), course_id varchar (8), sec_id varchar (8), semester varchar (6), year numeric (4,0), primary key (ID, course_id, sec_id, semester, year), foreign key (course_id, sec_id, semester, year) references section, foreign key (ID) references instructor)
- create table student (ID varchar(5), name varchar(20) not null, dept_name varchar(20), tot_cred numeric(3,0), primary key (ID), foreign key (dept_name) references department)
- create table takes (ID varchar(5), course_id varchar(8), sec_id varchar(8), semester varchar(6), year numeric(4,0), grade varchar(2), primary key (ID, course_id, sec_id, semester, year), foreign key (ID) references student, foreign key (course_id, sec_id, semester, year) references section)
- insert into instructor values ('10211', 'Smith', 'Biology', 66000)
- delete from student
- drop table r
- alter table r add A D
- alter table r drop A

Database Management Systems , Unit-III, Lecture 3 Introduction to SQL

- select A_{1},A_{2},...,A_{n} from r_{1},r_{2},...,r_{m} where P
- select name from instructor
- select distinct dept_name from instructor
- select all dept_name from instructor
- select * from instructor
- select '437'
- select '437' as FOO
- select A' from instructor
- select ID, name, salary/12 from instructor
- select ID, name, salary/12 as monthly_salary from instructor
- select name from instructor where dept_name = 'Comp. Sci.'
- select name from instructor where dept_name = 'Comp. Sci.' and salary > 70000
- select * from instructor, teaches
- select name, course_id from instructor, teaches where instructor.ID = teaches.ID

- select name, course_id from instructor, teaches where instructor.ID = teaches.ID and instructor. dept_name = 'Art'

Database Management Systems , Unit-III, Lecture 4 Introduction to SQL

- select T.name, S.course_id from instructor as T, teaches as S where T.ID=S.ID;
- select distinct T.name from instructor as T, instructor as S where T.salary > S.salary and S.dept_name ='Biology';
- select name from instructor where name like '%dar%'
- select distinct name from instructor order by name
- select name from instructor where salary between 90000 and 100000
- select name, course_id from instructor, teaches where (instructor.ID, dept_name) = (teaches.ID, 'Biology');
- (select course_id from section where semester = 'Fall' and year=2017) union (select course_id from section where semester = 'Spring' and year =2018)
- (select course_id from section where semester = 'Fall' and year=2017) intersect (select

course_id from section where semester = 'Spring' and year=2018)

- (select course_id from section where semester = 'Fall' and year =2017) except (select course_id from section where semester = 'Spring' and year=2018)