Practical Assignment - 1

* Name: Hardik Arora
* Branch : Btech - CS
* Program: AIML
* University Roll No. : 2215500071
* Section: 2AC
* Class Roll No. : 28

# Q. Create the following tables and fill the given data in them:-

Table-1

CREATE DATABASE kt;

USE kt;

Create Table IF NOT EXISTS College(

    cName VARCHAR(10),

    state VARCHAR(10),

    enrollment INT PRIMARY KEY NOT NULL

);

INSERT INTO College(cName, state, enrollment) VALUES('Stanford', 'CA', '15000');

INSERT INTO College(cName, state, enrollment) VALUES('Berkeley', 'CA', '36000');

INSERT INTO College(cName, state, enrollment) VALUES('MIT', 'MA', '10000');

INSERT INTO College(cName, state, enrollment) VALUES('Cornell', 'NY', '21000');

INSERT INTO College(cName, state, enrollment) VALUES('Harvard', 'MA', '50040');

SELECT \* FROM College;

OUTPUT:

# 

Table -2

Create Table IF NOT EXISTS Student(

    sID INT PRIMARY KEY,

    sName VARCHAR(10),

    GPA FLOAT,

    sizeHS INT NOT NULL,

    DoB DATE

);

INSERT INTO Student(sID, sName, GPA, sizeHS, DoB) VALUES('123', 'Amy', '3.9', '1000', '1996-06-26');

INSERT INTO Student(sID, sName, GPA, sizeHS, DoB) VALUES('234', 'Bob', '3.6', '1500', '1995-04-07');

INSERT INTO Student(sID, sName, GPA, sizeHS, DoB) VALUES('345', 'Craig', '3.5', '500', '1995-02-04');

INSERT INTO Student(sID, sName, GPA, sizeHS, DoB) VALUES('456', 'Doris', '3.9', '1000', '1997-07-24');

INSERT INTO Student(sID, sName, GPA, sizeHS, DoB) VALUES('567', 'Edward', '2.9', '2000', '1996-12-21');

INSERT INTO Student(sID, sName, GPA, sizeHS, DoB) VALUES('678', 'Fay', '3.8', '200', '1996-08-27');

INSERT INTO Student(sID, sName, GPA, sizeHS, DoB) VALUES('789', 'Gary', '3.4', '800', '1996-10-08');

INSERT INTO Student(sID, sName, GPA, sizeHS, DoB) VALUES('987', 'Helen', '3.7', '800', '1997-03-27');

INSERT INTO Student(sID, sName, GPA, sizeHS, DoB) VALUES('876', 'Irene', '3.9', '400', '1996-03-07');

INSERT INTO Student(sID, sName, GPA, sizeHS, DoB) VALUES('765', 'Jay', '2.9', '1500', '1998-08-08');

INSERT INTO Student(sID, sName, GPA, sizeHS, DoB) VALUES('654', 'Amy', '3.9', '1000', '1996-05-26');

INSERT INTO Student(sID, sName, GPA, sizeHS, DoB) VALUES('543', 'Craig', '3.4', '2000', '1998-08-27');

SELECT \* FROM Student;

OUTPUT :

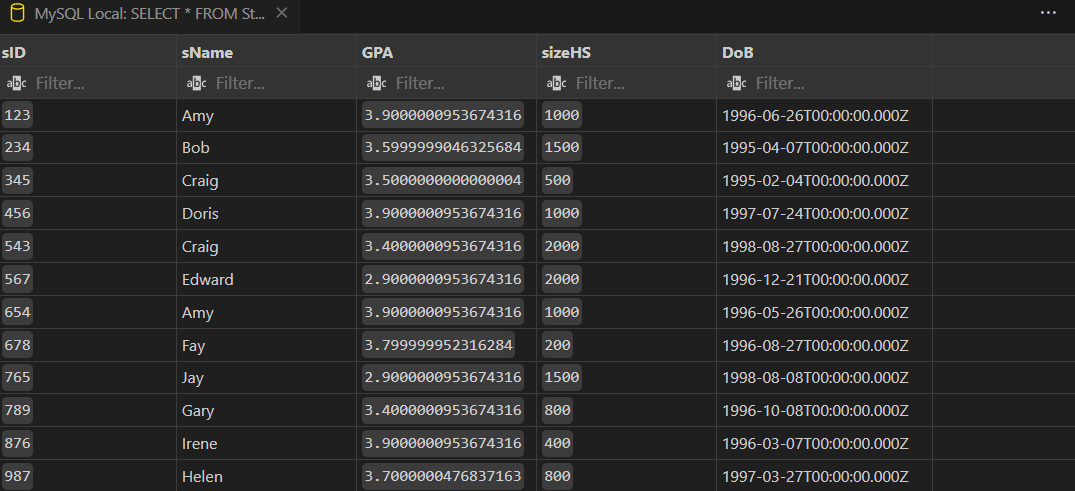


Table – 3

Create Table IF NOT EXISTS Apply(

    sID INT NOT NULL,

    cName VARCHAR(10),

    major VARCHAR(20),

    decision VARCHAR(1)

);

INSERT INTO Apply(sID, cName, major, decision) VALUES('123', 'Stanford', 'CS', 'Y');

INSERT INTO Apply(sID, cName, major, decision) VALUES('123', 'Stanford', 'EE', 'N');

INSERT INTO Apply(sID, cName, major, decision) VALUES('123', 'Berkeley', 'CS', 'Y');

INSERT INTO Apply(sID, cName, major, decision) VALUES('123', 'Cornell', 'EE', 'Y');

INSERT INTO Apply(sID, cName, major, decision) VALUES('234', 'Berkeley', 'biology', 'N');

INSERT INTO Apply(sID, cName, major, decision) VALUES('345', 'MIT', 'bioengineering', 'Y');

INSERT INTO Apply(sID, cName, major, decision) VALUES('345', 'Cornell', 'bioengineering', 'N');

INSERT INTO Apply(sID, cName, major, decision) VALUES('345', 'Cornell', 'CS', 'Y');

INSERT INTO Apply(sID, cName, major, decision) VALUES('345', 'Cornell', 'EE', 'N');

INSERT INTO Apply(sID, cName, major, decision) VALUES('678', 'Stanford', 'history', 'Y');

INSERT INTO Apply(sID, cName, major, decision) VALUES('987', 'Stanford', 'CS', 'Y');

INSERT INTO Apply(sID, cName, major, decision) VALUES('987', 'Berkeley', 'CS', 'Y');

INSERT INTO Apply(sID, cName, major, decision) VALUES('876', 'Stanford', 'CS', 'N');

INSERT INTO Apply(sID, cName, major, decision) VALUES('876', 'MIT', 'biology', 'Y');

INSERT INTO Apply(sID, cName, major, decision) VALUES('876', 'MIT', 'marine biology', 'N');

INSERT INTO Apply(sID, cName, major, decision) VALUES('765', 'Stanford', 'history', 'Y');

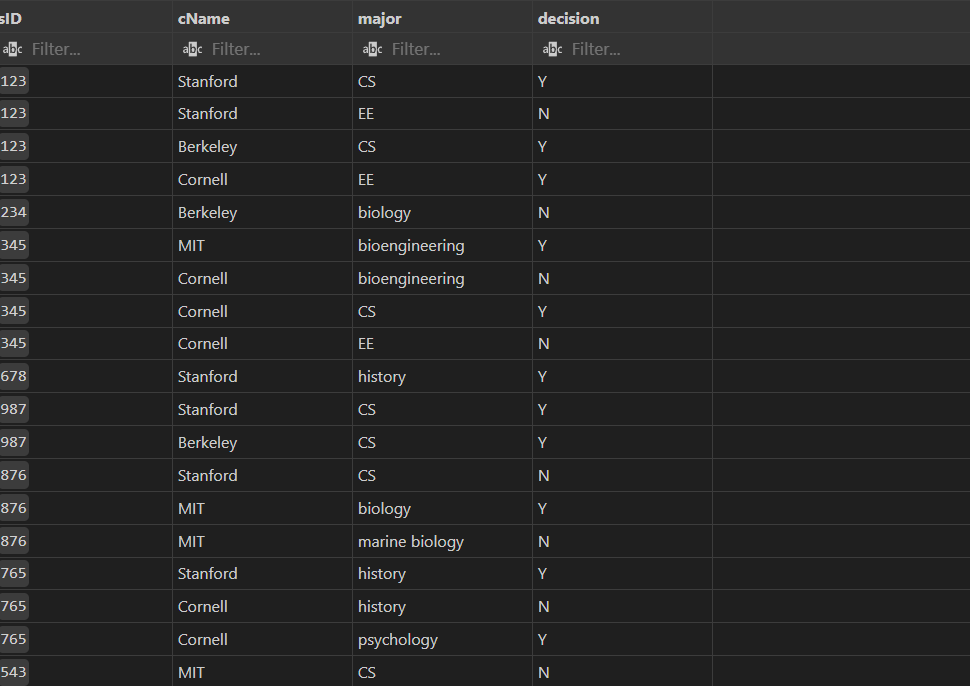
INSERT INTO Apply(sID, cName, major, decision) VALUES('765', 'Cornell', 'history', 'N');

INSERT INTO Apply(sID, cName, major, decision) VALUES('765', 'Cornell', 'psychology', 'Y');

INSERT INTO Apply(sID, cName, major, decision) VALUES('543', 'MIT', 'CS', 'N');

SELECT \* FROM Apply;

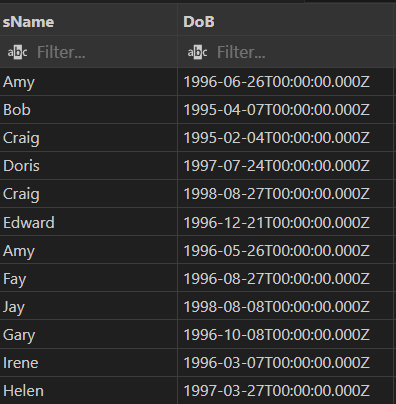
OUTPUT:



# Q. State of SQL \*PLUS Queries for each of the following:

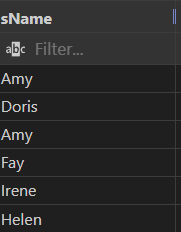
1. List the student name, dob from student table

SELECT sName, DoB FROM student;



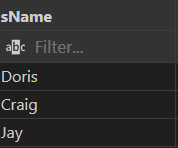
1. List of the name of student scoring more than 3.7 in GPA.

SELECT sName FROM student WHERE GPA>3.7;



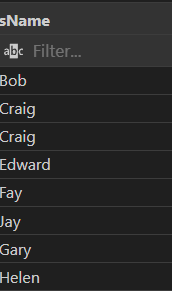
1. List the name of student whose High School size is atleast 1000 and born after 1996.

SELECT sName FROM student WHERE sizeHS >= 1000 AND DoB > '1996-12-31';



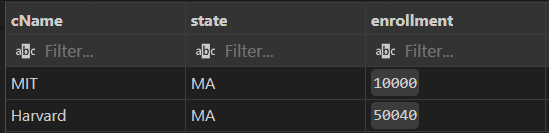
1. List the name of student who are scoring GPA between 2.9 and 3.9.

SELECT sName FROM student WHERE GPA >= 2.9 AND GPA <= 3.9;



1. List all the details of colleges who are situated in MA

SELECT \* FROM college WHERE state = 'MA' ;



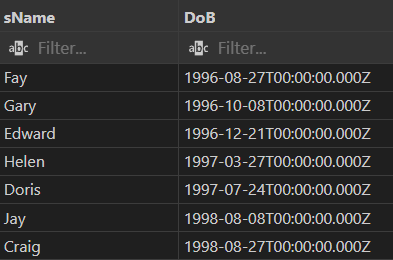
1. List the students who have scored more than 2.0 but less than 3.5.

SELECT sName FROM student WHERE GPA >= 2.0 AND GPA <= 3.5;



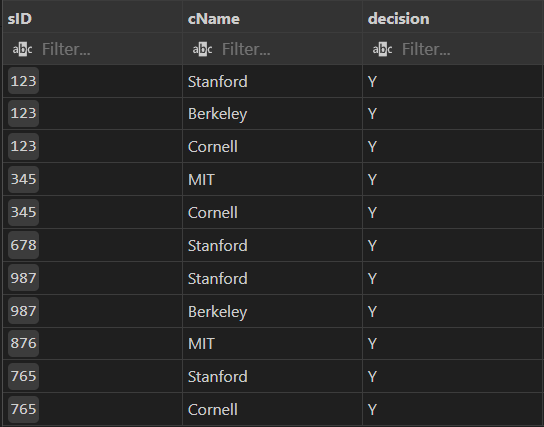
1. List the name of students who are born after 1 st Jul 1996 in the order of Date of Birth.

SELECT sName, DoB FROM student WHERE DoB > '1996-07-01' ORDER BY DoB ASC;



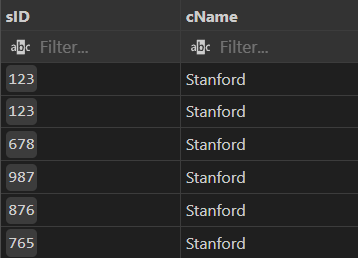
1. List the sID, cName, decision of applications that are accepted.

SELECT sID, cName, decision FROM apply WHERE decision = 'Y';



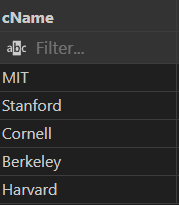
1. List the sID, cName of applications that are filed at Stanford.

SELECT sID,cName FROM apply WHERE cName = 'Stanford';



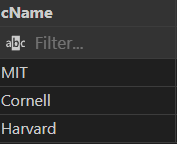
1. List all the colleges that has enrollment greater than 1000.

SELECT cName FROM college WHERE enrollment > '1000';



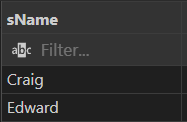
1. List the colleges not in California.

SELECT cName FROM college WHERE state != 'CA';



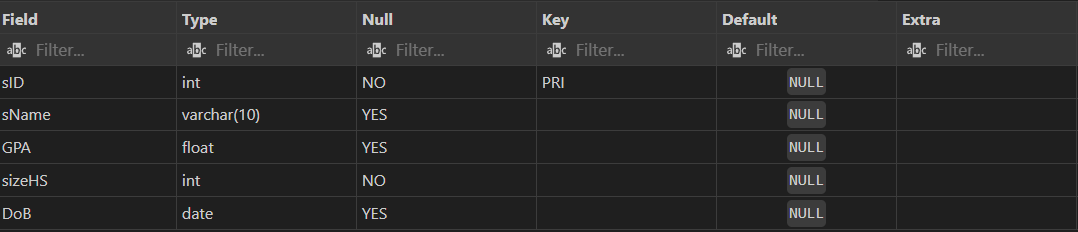
1. List names of all students who came from high school having size greater than 1700 and scored GPA less than 3.8.

SELECT sName FROM student WHERE sizeHS > '1700' AND GPA < 3.8;



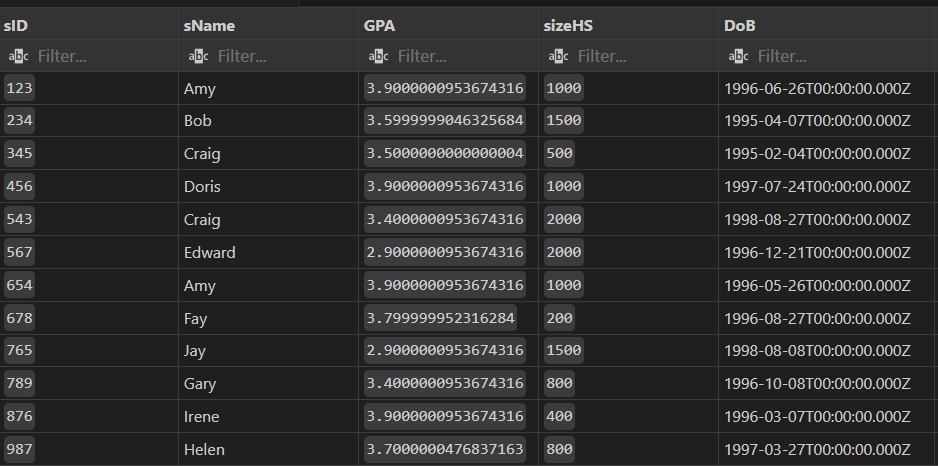
1. Display the description of student table.

DESCRIBE student;



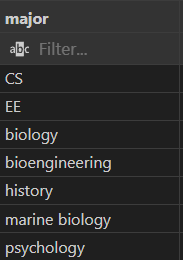
1. Display the details of all students.

SELECT \* FROM student;



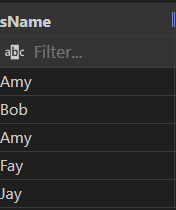
1. Display unique majors.

SELECT  DISTINCT major FROM apply;



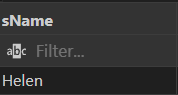
1. List the student names those are having three characters in their Names

SELECT sName FROM student WHERE CHAR\_LENGTH(sName) = 3;



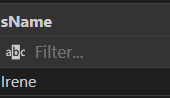
1. List the student names those are starting with ‘H’ and with five characters.

SELECT sName FROM student WHERE sName LIKE 'H%' AND CHAR\_LENGTH(sName)=5;



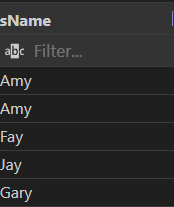
1. List the student names that are having third character and fifth character must be ‘e’.

SELECT sName FROM student WHERE CHAR\_LENGTH(sName) >= 5 AND SUBSTRING(sName, 3, 1) = 'e' AND SUBSTRING(sName, 5, 1) = 'e';



1. List the student names ending with ‘Y’

SELECT sName FROM student WHERE RIGHT(sName , 1) = 'Y';



1. List the students in the order of their GPA.

SELECT sName,GPA FROM student ORDER BY GPA ASC;



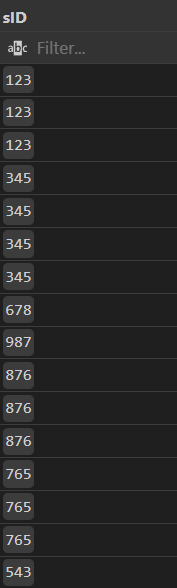
1. List the details of students in order of the ascending of GPA and descending of DoB.

SELECT \* FROM student ORDER BY GPA ASC , DoB DESC;



1. List the sIDs of students who apply in either ‘Stanford’, ‘Cornell’ or ‘MIT’ college.

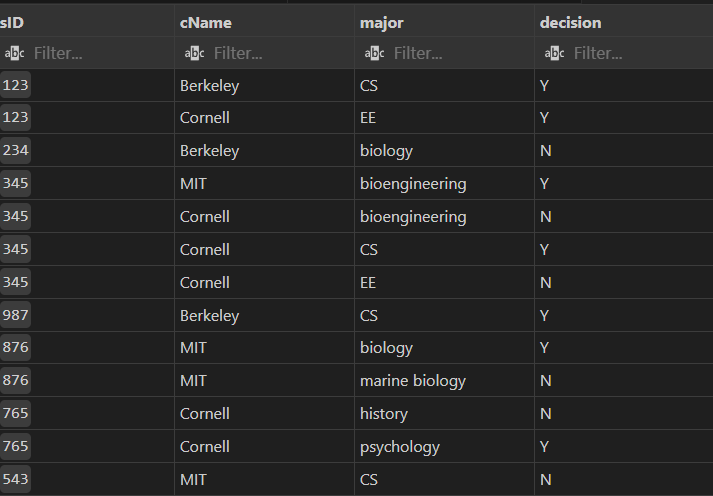
SELECT sID FROM apply WHERE cName IN ('Stanford','Cornell','MIT');



1. Delete all applications filed at Stanford.

DELETE FROM apply WHERE cName = 'Stanford';

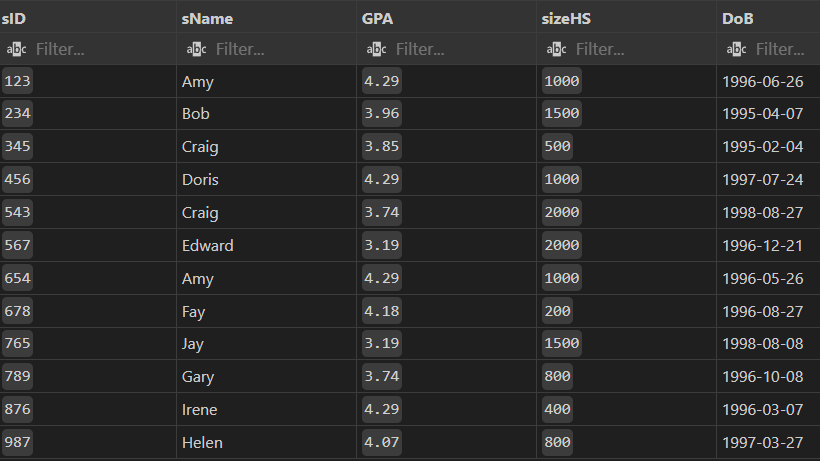
SELECT \* FROM apply;



1. Modify the GPA of all students by giving 10% raise in GPA.

UPDATE student SET GPA = GPA + (0.10\*GPA);

SELECT \* FROM student;



1. Delete the college Stanford from the table.

DELETE FROM college WHERE cName = 'Stanford';

1. Increment the GPA of students by 1.5 whose GPA is less than 3.5 and whose High School Size is greater than 1500.

Update student SET GPA = GPA + 1.5 WHERE GPA < 3.5 AND sizeHS > 1500;

SELECT \* FROM student;



1. Delete the students who have scored less than 3.2 GPA.

DELETE \* FROM student WHERE GPA < 3.2;

SELECT \* FROM student;

# Exercise:

Q. Determine the appropriate datatype:

* Deptno – INT
* Dname – VARCHAR(255)
* Loc – VARCHAR(255)

CREATE TABLE IF NOT EXISTS Dept(

    deptno INT PRIMARY KEY,

    dname VARCHAR(255),

    loc VARCHAR(255)

);

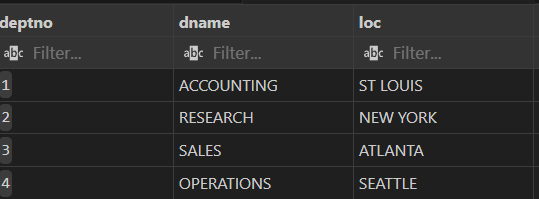
INSERT INTO Dept(deptno, dname, loc) VALUES('1', 'ACCOUNTING', 'ST LOUIS');

INSERT INTO Dept(deptno, dname, loc) VALUES('2', 'RESEARCH', 'NEW YORK');

INSERT INTO Dept(deptno, dname, loc) VALUES('3', 'SALES', 'ATLANTA');

INSERT INTO Dept(deptno, dname, loc) VALUES('4', 'OPERATIONS', 'SEATTLE');

SELECT \* FROM Dept;



Q. Determine the appropriate datatype:

* empno – INT
* ename – VARCHAR(255)
* job – VARCHAR(255)
* mgr – VARCHAR(255)
* hiredate – DATE
* sal – INT
* comm – VARCHAAR(255)
* dept - INT

CREATE TABLE IF NOT EXISTS Employee(

    empno INT PRIMARY KEY,

    ename VARCHAR(255),

    job VARCHAR(255),

    mgr VARCHAR(255),

    hierdate DATE,

    sal INT NOT NULL,

    comm VARCHAR(255),

    dept INT NOT NULL

);

INSERT INTO Employee(empno, ename, job, mgr, hierdate, sal, comm, dept) VALUES ('1', 'JOHNSON', 'ADMIN', '6', '1990-12-17', '18000', '(null)', '4');

INSERT INTO Employee(empno, ename, job, mgr, hierdate, sal, comm, dept) VALUES ('2', 'HARDING', 'MANAGER', '9', '1998-02-02', '52000', '300', '3');

INSERT INTO Employee(empno, ename, job, mgr, hierdate, sal, comm, dept) VALUES ('3', 'TAFT', 'SALES 1', '2', '1996-01-02', '25000', '500', '3');

INSERT INTO Employee(empno, ename, job, mgr, hierdate, sal, comm, dept) VALUES ('4', 'HOOVER', 'SALES 1', '2', '1990-04-02', '27000', '(null)', '3');

INSERT INTO Employee(empno, ename, job, mgr, hierdate, sal, comm, dept) VALUES ('5', 'LINCOLN', 'TECH', '6', '1994-06-23', '22500', '1400', '4');

INSERT INTO Employee(empno, ename, job, mgr, hierdate, sal, comm, dept) VALUES ('6', 'GARFIELD', 'MANAGER', '9', '1993-05-01', '54000', '(null)', '4');

INSERT INTO Employee(empno, ename, job, mgr, hierdate, sal, comm, dept) VALUES ('7', 'POLK', 'TECH', '6', '1997-09-27', '25000', '(null)', '4');

INSERT INTO Employee(empno, ename, job, mgr, hierdate, sal, comm, dept) VALUES ('8', 'GRANT', 'ENGINEER', '10', '1997-03-30', '32000', '(null)', '2');

INSERT INTO Employee(empno, ename, job, mgr, hierdate, sal, comm, dept) VALUES ('9', 'JACKSON', 'CEO', '(null)', '1990-01-01', '75000', '(null)', '4');

INSERT INTO Employee(empno, ename, job, mgr, hierdate, sal, comm, dept) VALUES ('10', 'FILLMORE', 'MANAGER', '9', '1994-08-09', '56000', '(null)', '2');

INSERT INTO Employee(empno, ename, job, mgr, hierdate, sal, comm, dept) VALUES ('11', 'ADAMAS', 'ENGINEER', '10', '1996-03-15', '34000', '(null)', '2');

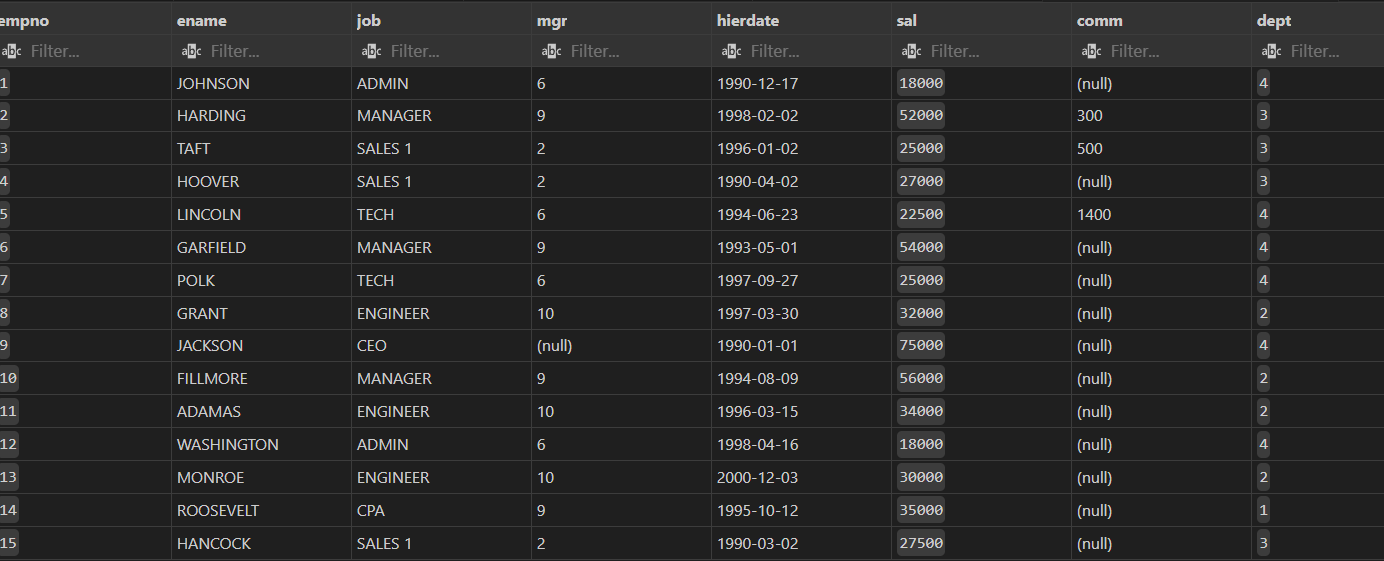
INSERT INTO Employee(empno, ename, job, mgr, hierdate, sal, comm, dept) VALUES ('12', 'WASHINGTON', 'ADMIN', '6', '1998-04-16', '18000', '(null)', '4');

INSERT INTO Employee(empno, ename, job, mgr, hierdate, sal, comm, dept) VALUES ('13', 'MONROE', 'ENGINEER', '10', '2000-12-03', '30000', '(null)', '2');

INSERT INTO Employee(empno, ename, job, mgr, hierdate, sal, comm, dept) VALUES ('14', 'ROOSEVELT', 'CPA', '9', '1995-10-12', '35000', '(null)', '1');

INSERT INTO Employee(empno, ename, job, mgr, hierdate, sal, comm, dept) VALUES ('15', 'HANCOCK', 'SALES 1', '2', '1990-03-02', '27500', '(null)', '3');

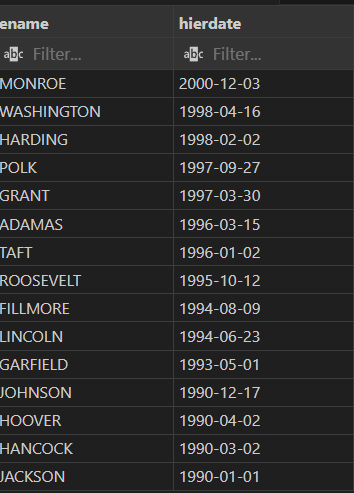
SELECT \* FROM employee;



## Solve the following queries:

Q1. Employee Name and Hire Date sorted by Hire Date(Recent to Old).

SELECT ename,hierdate FROM employee ORDER BY hierdate DESC;



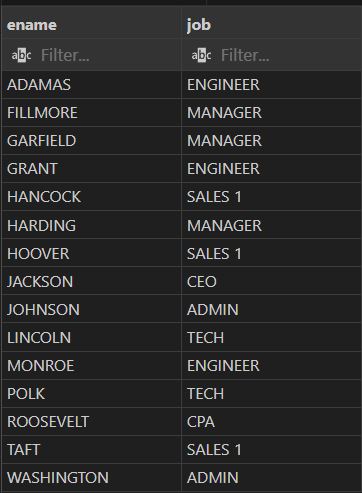
Q2. Employee Name and Job sorted by Job(Alphabetically).

SELECT ename,job FROM employee ORDER BY job ASC;



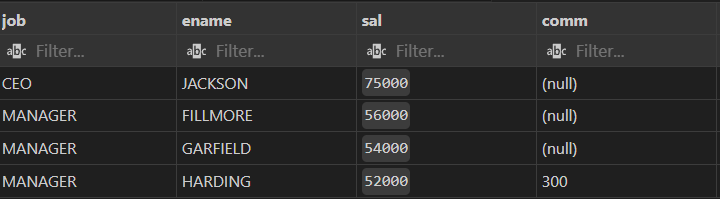
Q3. Employee Name and Job for all Engineers, sorted by Employee Name Alphabetically.

SELECT ename,job FROM employee ORDER BY ename ASC;



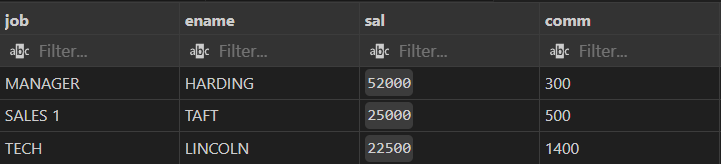
Q4. Job, Employee Name , Salary and Commission for employees with salary over 50000 sorted by Salary (Largest to Smallest).

SELECT job,ename,sal,comm FROM employee WHERE sal > '50000' ORDER BY sal DESC;



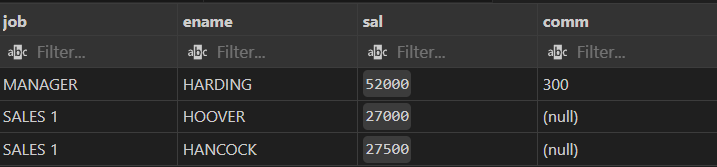
Q5. Job, Employee Name, Salary and Commission for employees with a Commission sorted by Salary (Largest to Smallest).

SELECT job,ename,sal,comm FROM employee WHERE comm != '(null)' ORDER BY sal DESC;



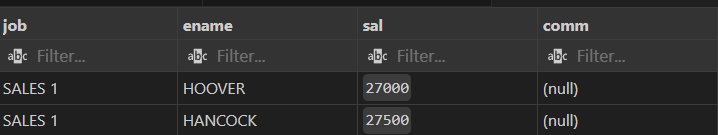
Q6. Job, Employee Name, Salary and Commission for employees whose name starts with the letter ‘H’.

SELECT job,ename,sal,comm FROM employee WHERE SUBSTRING(ename, 1, 1) = 'H';



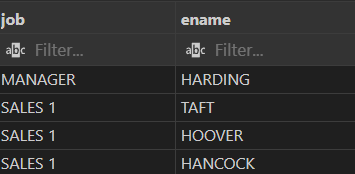
Q7. Job, Employee Name, Salary and Commission for employees whose name starts with the letter ‘H’ and who do not get any Commission.

SELECT job,ename,sal,comm FROM employee WHERE SUBSTRING(ename, 1, 1) = 'H' AND comm = '(null)';



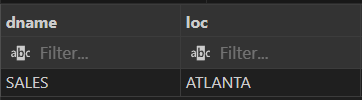
Q8. Job, Employee Name for employees in Dept No. 3.

SELECT job,ename FROM employee WHERE dept = '3';



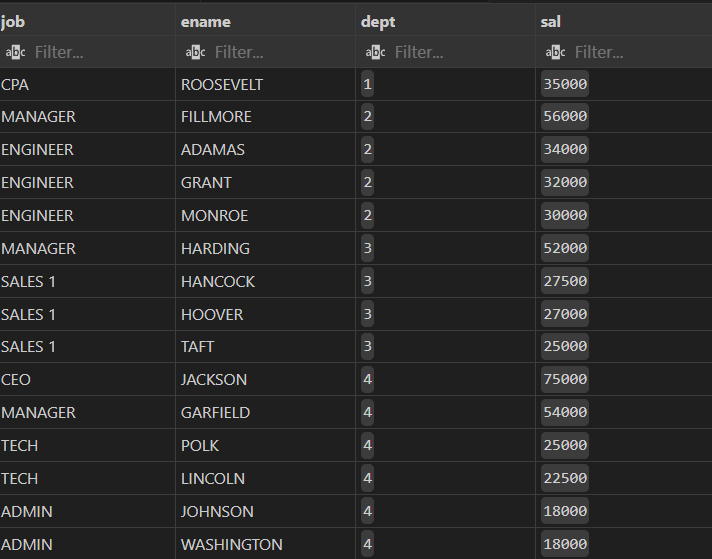
Q9. Dept Name and Loc for employees in Dept No. 3.

SELECT dname,loc FROM dept WHERE deptno = '3';



Q10. Job, Employee Name, Dept, Salary sorted first by Dept(Smallest to Largest) and then Salary(Largest to Smallest).

SELECT job,ename,dept,sal FROM employee ORDER BY dept ASC,sal DESC;



* Submitted By : Hardik Arora
* Branch : Btech - CS
* Program: AIML
* University Roll No. : 2215500071
* Section: 2AC
* Class Roll No. : 28
* Submitted to: Ayushi Mam