Chapter 8 Problems, Part 1

1. Write the SQL code that will create only the table structure for a table named EMP\_1. This table will be a subset of the EMPLOYEE table. The basic EMP\_1 table structure is summarized in the following table. Use EMP\_NUM as the pri- mary key. Note that the JOB\_CODE is the FK to JOB so be certain to enforce ref- erential integrity. Your code should also prevent null entries in EMP\_LNAME and EMP\_FNAME.

CREATE TABLE EMP\_1(

EMP\_NUM CHAR(3) UNIQUE,

EMP\_LNAME VARCHAR(15) NOT NULL,

EMP\_FNAME VARCHAR(15) NOT NULL,

EMP\_INITIAL CHAR(1),

EMP\_HIREDATE DATE,

JOB\_CODE CHAR(3),

PRIMARY KEY(EMP\_NUM),

FOREIGN KEY(JOB\_CODE) REFERENCES JOB (JOB\_CODE) ON UPDATE CASCADE);

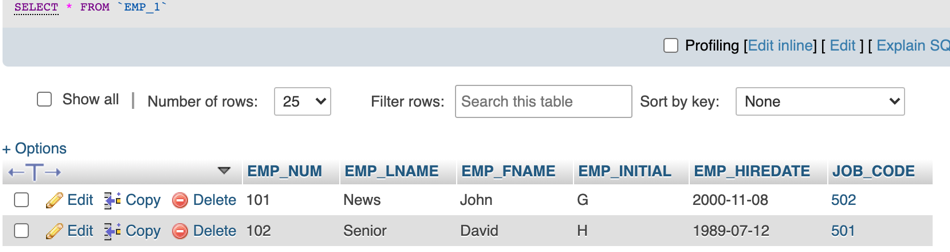
1. Having created the table structure in Problem 1, write the SQL code to enter the first two rows for the table shown in Figure P8.2. Each row should be inserted indi- vidually, without using a subquery. Insert the rows in the order that they are listed in the figure.

INSERT INTO EMP\_1

VALUES('101', 'News', 'John', 'G', '2000-11-08', '502');

INSERT INTO EMP\_1

VALUES('102', 'Senior', 'David', 'H', '1989-07-12', '501');

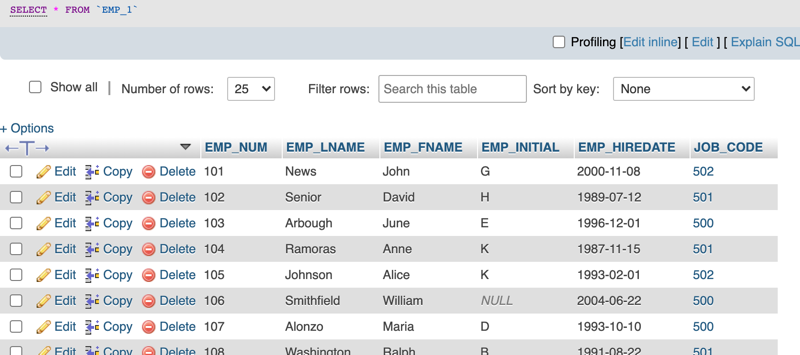


1. Using the EMPLOYEE table that already exists, use a subquery to insert the remaining rows from the EMPLOYEE table into the EMP\_1 table. Remember, your sub- query should only retrieve the columns needed for the EMP\_1 table and only the employees shown in the figure.

INSERT INTO EMP\_1 (EMP\_NUM, EMP\_LNAME, EMP\_FNAME, EMP\_INITIAL, EMP\_HIREDATE, JOB\_CODE)

SELECT EMP\_NUM, EMP\_LNAME, EMP\_FNAME, EMP\_INITIAL, EMP\_HIREDATE, JOB\_CODE

FROM EMPLOYEE

WHERE EMPLOYEE.EMP\_NUM NOT IN ('101','102');

1. Write the SQL code that will save the changes made to the EMP\_1 table (if sup- ported by your DBMS).

MySQL automatically commits changes with each command, so the COMMIT command is not needed.

1. Write the SQL code to change the job code to 501 for the person whose employee number (EMP\_NUM) is 107.

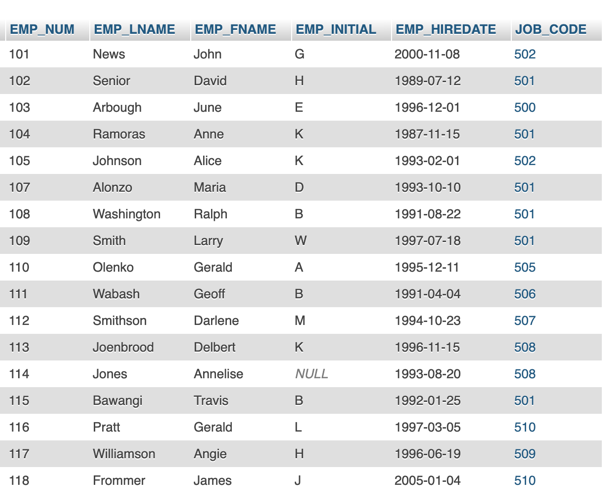
UPDATE EMP\_1

SET JOB\_CODE = '501'

/var/folders/1p/b09c6m_95m75ws9jbqh02z9r0000gn/T/com.microsoft.Word/Content.MSO/87CD31EC.tmpWHERE EMP\_NUM = '107';

1. Write the SQL code to delete the row for William Smithfield, who was hired on June 22, 2004, and whose job code is 500. (*Hint:* Use logical operators to include all of the information given in this problem. Remember, if you are using MySQL, you will have to first disable “safe mode.”)

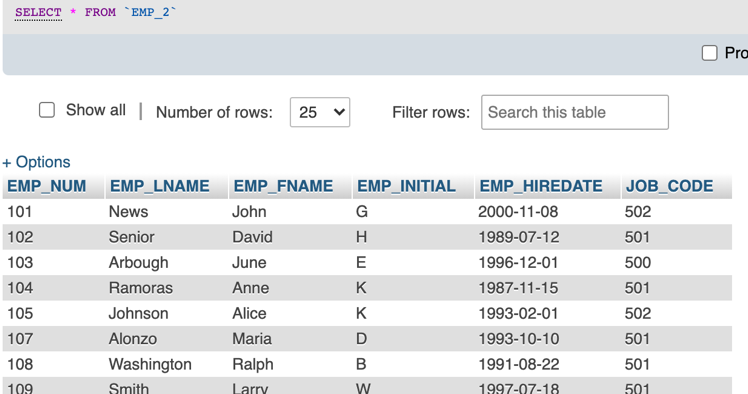
DELETE FROM EMP\_1

WHERE EMP\_LNAME = 'Smithfield' AND EMP\_FNAME = 'William';

1. Write the SQL code to create a copy of EMP\_1, including all of its data, and naming the copy EMP\_2.

CREATE TABLE EMP\_2

SELECT EMP\_NUM, EMP\_LNAME, EMP\_FNAME, EMP\_INITIAL, EMP\_HIREDATE, JOB\_CODE

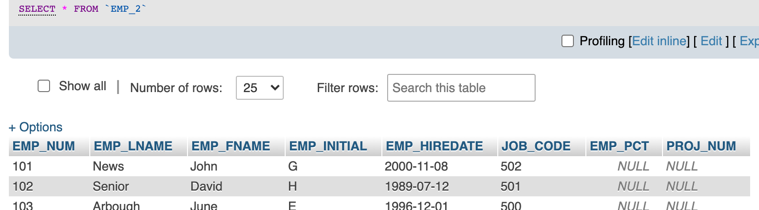
FROM EMP\_1;

1. Using the EMP\_2 table, write the SQL code that will add the attributes EMP\_PCT and PROJ\_NUM to EMP\_2. The EMP\_PCT is the bonus percentage to be paid to each employee. The new attribute characteristics are:

EMP\_PCT NUMBER(4,2) PROJ\_NUM CHAR(3)

ALTER TABLE EMP\_2

ADD EMP\_PCT NUMERIC(4,2),

ADD PROJ\_NUM CHAR(3);

1. Using the EMP\_2 table, write the SQL code to change the EMP\_PCT value to 3.85 for the person whose employee number (EMP\_NUM) is 103.

UPDATE EMP\_2

SET EMP\_PCT = 3.85

/var/folders/1p/b09c6m_95m75ws9jbqh02z9r0000gn/T/com.microsoft.Word/Content.MSO/993418C4.tmpWHERE EMP\_NUM = '103';

1. Using the EMP\_2 table, write a single SQL command to change the EMP\_PCT value to 5.00 for the people with employee numbers 101, 105, and 107.

UPDATE EMP\_2

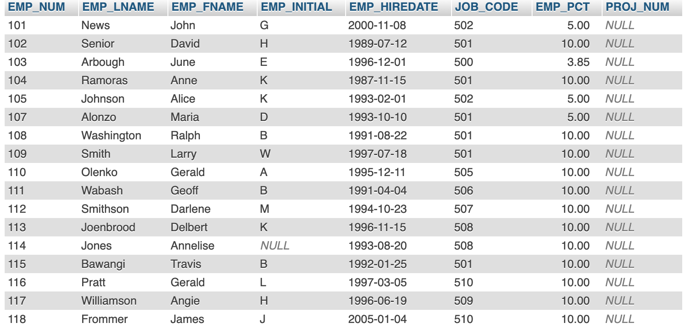
SET EMP\_PCT = 5.00

WHERE EMP\_NUM IN ('101', '105', '107');

1. Using the EMP\_2 table, write a single SQL command to change the EMP\_PCT value to 10.00 for all employees who do not currently have a value for EMP\_PCT.

UPDATE EMP\_2

SET EMP\_PCT = 10.00

WHERE EMP\_PCT IS NULL;

1. Using the EMP\_2 table, write the SQL command to add .15 to the EMP\_PCT of the employee whose name is Maria D. Alonzo. (Use the employee name in your command to determine the correct employee.)

UPDATE EMP\_2

SET EMP\_PCT = EMP\_PCT + .15

/var/folders/1p/b09c6m_95m75ws9jbqh02z9r0000gn/T/com.microsoft.Word/Content.MSO/75EB8ADE.tmpWHERE EMP\_LNAME = 'Alonzo' AND EMP\_FNAME = 'Maria';

1. Using a single command sequence with the EMP\_2 table, write the SQL code that will change the project number (PROJ\_NUM) to 18 for all employees whose job classification (JOB\_CODE) is 500.

UPDATE EMP\_2

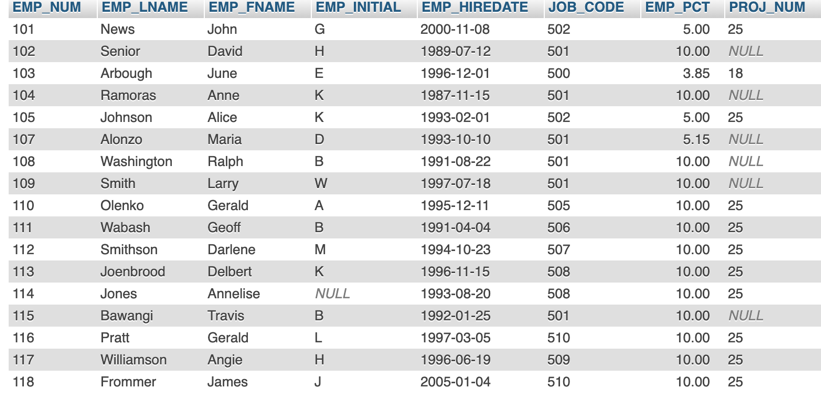
SET PROJ\_NUM = '18'

WHERE JOB\_CODE = '500';

1. Using a single command sequence with the EMP\_2 table, write the SQL code that will change the project number (PROJ\_NUM) to 25 for all employees whose job classification (JOB\_CODE) is 502 or higher.

UPDATE EMP\_2

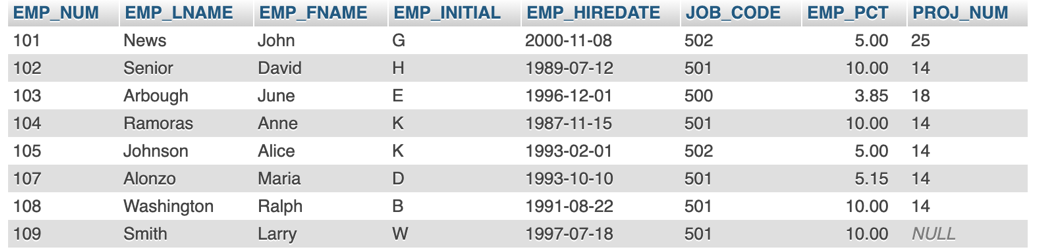
SET PROJ\_NUM = '25'

WHERE JOB\_CODE >= '502';

1. Write the SQL code that will change the PROJ\_NUM to 14 for employees who were hired before January 1, 1994, and whose job code is at least 501. When you finish Problems 7–15, the EMP\_2 table will contain the data shown in Figure P8.15.

UPDATE EMP\_2

SET PROJ\_NUM = '14'

WHERE EMP\_HIREDATE < '1994-01-01' AND JOB\_CODE >= '501';

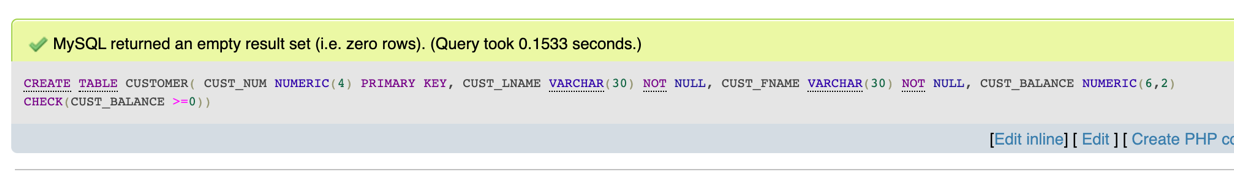
1. Create the CUSTOMER table structure illustrated in Figure P8.16. The customer number should store integer values. The name attributes should support variable length character data up to 30 characters each. The customer balance should sup- port up to six digits on the left of the decimal place and two digits to the right of the decimal place.

CREATE TABLE CUSTOMER(

CUST\_NUM NUMERIC(4) PRIMARY KEY,

CUST\_LNAME VARCHAR(30) NOT NULL,

CUST\_FNAME VARCHAR(30) NOT NULL,

CUST\_BALANCE NUMERIC(6,2) CHECK(CUST\_BALANCE >=0));

1. Create the INVOICE table structure illustrated in Figure P8.16. The invoice number should store integer values. The invoice date should store date values. The invoice amount should support up to 8 digits to the left of the decimal place and two digits to the right of the decimal place.

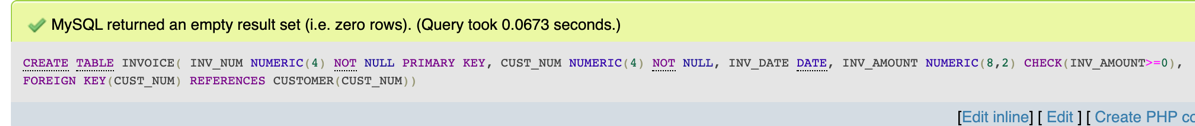
CREATE TABLE INVOICE(

INV\_NUM NUMERIC(4) NOT NULL PRIMARY KEY,

CUST\_NUM NUMERIC(4) NOT NULL,

INV\_DATE DATE,

INV\_AMOUNT NUMERIC(8,2) CHECK(INV\_AMOUNT>=0),

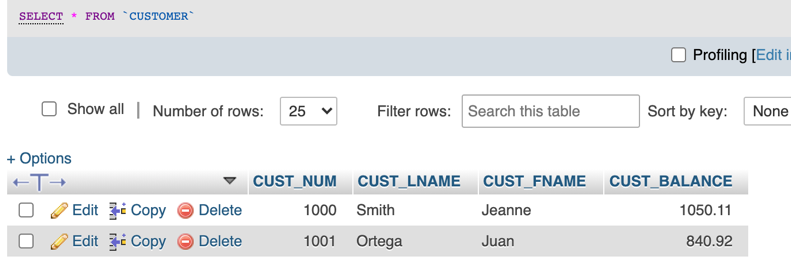
FOREIGN KEY(CUST\_NUM) REFERENCES CUSTOMER(CUST\_NUM));

1. Write the set of SQL commands necessary to insert the data into the CUSTOMER table you created in Problem 16, as illustrated in Figure P8.16.

INSERT INTO CUSTOMER

VALUES(1000, 'Smith', 'Jeanne', 1050.11);

INSERT INTO CUSTOMER

 VALUES(1001, 'Ortega', 'Juan', 840.92);

1. Write the set of SQL commands necessary to insert the data into the INVOICE table you created in Problem 17, as illustrated in Figure P8.16.

INSERT INTO INVOICE

VALUES(8000, 1000, '2016-03-23', 235.89);

INSERT INTO INVOICE

VALUES(8001, 1001, '2016-03-23', 312.82);

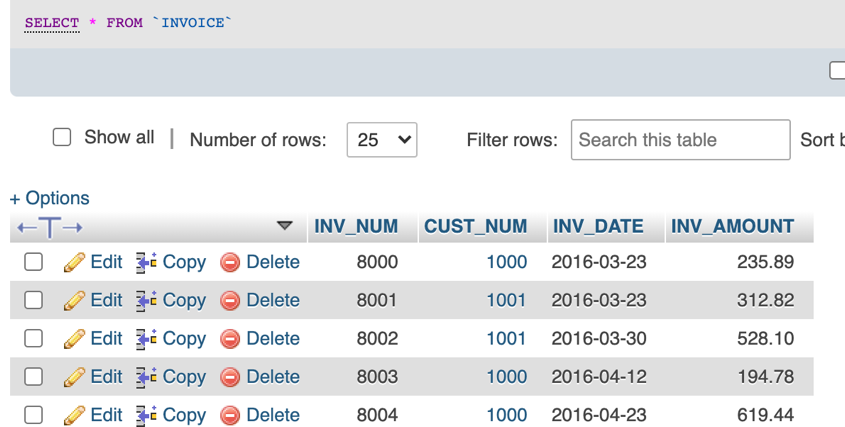
INSERT INTO INVOICE

VALUES(8002, 1001, '2016-03-30', 528.10);

INSERT INTO INVOICE

VALUES(8003, 1000, '2016-04-12', 194.78);

INSERT INTO INVOICE

 VALUES(8004, 1000, '2016-04-23', 619.44);