# SQL Self-Test

## Part A

## Workshop Environment

# Table of Contents

# Sample Tables

A database named SAMPLE is created in DB2's "First Steps" process. It contains nine user tables, in addition to the catalog tables. The only two tables we shall use in the workshops are DEPARTMENT and EMPLOYEE.

# DEPARTMENT Table

The DEPARTMENT table contains the following five columns:

| Column Name | Type | Nulls | Description |
|---|---|---|---|
| DEPTNO | CHAR(3) | No | Department number (primary key) |
| DEPTNAME | VARCHAR(29) | No | Name describing general activities of department |
| MGRNO | CHAR(6) | Yes | Employee number (EMPNO) of department manager |
| ADMRDEPT | CHAR(3) | No | Department (DEPTNO) to which this department reports |
| LOCATION | CHAR(16) | Yes | Name of the remote location |

MGRNO is a foreign key pointing to the primary key of the EMPLOYEE table. ADMRDEPT is a foreign key pointing to the primary key of this, the DEPARTMENT table. However, database-managed referential integrity has not been turned on, so invalid values are possible in both foreign keys.

## Contents

```
DEPTNO DEPTNAME                          MGRNO  ADMRDEPT LOCATION
------ ------------------------------ ------ -------- ----------------
A00    SPIFFY COMPUTER SERVICE DIV.    000010 A00      -
B01    PLANNING                        000020 A00      -
C01    INFORMATION CENTER              000030 A00      -
D01    DEVELOPMENT CENTER              -      A00      -
D11    MANUFACTURING SYSTEMS           000060 D01      -
D21    ADMINISTRATION SYSTEMS          000070 D01      -
E01    SUPPORT SERVICES                000050 A00      -
E11    OPERATIONS                      000090 E01      -
E21    SOFTWARE SUPPORT                000100 E01      -

  9 record(s) selected.
```

# EMPLOYEE Table

The EMPLOYEE table contains the following 14 columns:

| Column Name | Type | Nulls | Description |
|---|---|---|---|
| EMPNO | CHAR(6) | No | Employee number (primary key) |
| FIRSTNME | VARCHAR(12) | No | First name |
| MIDINIT | CHAR(1) | No | Middle initial |
| LASTNAME | VARCHAR(15) | No | Last name |
| WORKDEPT | CHAR(3) | Yes | Department (DEPTNO) in which the employee works |
| PHONENO | CHAR(4) | Yes | Telephone extension number |
| HIREDATE | DATE | Yes | Date of hire |
| JOB | CHAR(8) | Yes | Job |
| EDLEVEL | SMALLINT | No | Number of years of formal education |
| SEX | CHAR(1) | Yes | Sex (M male, F female) |
| BIRTHDATE | DATE | Yes | Date of birth |
| SALARY | DEC(9,2) | Yes | Yearly salary |
| BONUS | DEC(9,2) | Yes | Yearly bonus |
| COMM | DEC(9,2) | Yes | Yearly commission |

WORKDEPT is a foreign key pointing to the primary key of the DEPARTMENT table. However, database-managed referential integrity has not been turned on, so invalid values are possible in the foreign key.

Also, there is no database-managed domain integrity, so columns may contain values outside their domains. For example, it is possible to have a value other than M or F in the SEX column.

## Contents (Columns 1–7)

```
EMPNO  FIRSTNME     MIDINIT LASTNAME        WORKDEPT PHONENO HIREDATE
------ ------------ ------- --------------- -------- ------- ----------
000010 CHRISTINE    I       HAAS            A00      3978    01/01/1965
000020 MICHAEL      L       THOMPSON        B01      3476    10/10/1973
000030 SALLY        A       KWAN            C01      4738    04/05/1975
000050 JOHN         B       GEYER           E01      6789    08/17/1949
000060 IRVING       F       STERN           D11      6423    09/14/1973
000070 EVA          D       PULASKI         D21      7831    09/30/1980
000090 EILEEN       W       HENDERSON       E11      5498    08/15/1970
000100 THEODORE     Q       SPENSER         E21      0972    06/19/1980
000110 VINCENZO     G       LUCCHESSI       A00      3490    05/16/1958
000120 SEAN                 O'CONNELL       A00      2167    12/05/1963
000130 DOLORES      M       QUINTANA        C01      4578    07/28/1971
000140 HEATHER      A       NICHOLLS        C01      1793    12/15/1976
000150 BRUCE                ADAMSON         D11      4510    02/12/1972
000160 ELIZABETH    R       PIANKA          D11      3782    10/11/1977
000170 MASATOSHI    J       YOSHIMURA       D11      2890    09/15/1978
000180 MARILYN      S       SCOUTTEN        D11      1682    07/07/1973
000190 JAMES        H       WALKER          D11      2986    07/26/1974
000200 DAVID                BROWN           D11      4501    03/03/1966
000210 WILLIAM      T       JONES           D11      0942    04/11/1979
000220 JENNIFER     K       LUTZ            D11      0672    08/29/1968
000230 JAMES        J       JEFFERSON       D21      2094    11/21/1966
000240 SALVATORE    M       MARINO          D21      3780    12/05/1979
000250 DANIEL       S       SMITH           D21      0961    10/30/1969
000260 SYBIL        P       JOHNSON         D21      8953    09/11/1975
000270 MARIA        L       PEREZ           D21      9001    09/30/1980
000280 ETHEL        R       SCHNEIDER       E11      8997    03/24/1967
000290 JOHN         R       PARKER          E11      4502    05/30/1980
000300 PHILIP       X       SMITH           E11      2095    06/19/1972
000310 MAUDE        F       SETRIGHT        E11      3332    09/12/1964
000320 RAMLAL       V       MEHTA           E21      9990    07/07/1965
000330 WING                 LEE             E21      2103    02/23/1976
000340 JASON        R       GOUNOT          E21      5698    05/05/1947

  32 record(s) selected.
```

## Contents (Columns 1 and 8–14)

```
EMPNO  JOB       EDLEVEL SEX BIRTHDATE  SALARY      BONUS       COMM
------ -------- ------- --- ---------- ----------- ----------- -----------
000010 PRES           18 F   08/24/1933    52750.00     1000.00     4220.00
000020 MANAGER        18 M   02/02/1948    41250.00      800.00     3300.00
000030 MANAGER        20 F   05/11/1941    38250.00      800.00     3060.00
000050 MANAGER        16 M   09/15/1925    40175.00      800.00     3214.00
000060 MANAGER        16 M   07/07/1945    32250.00      500.00     2580.00
000070 MANAGER        16 F   05/26/1953    36170.00      700.00     2893.00
000090 MANAGER        16 F   05/15/1941    29750.00      600.00     2380.00
000100 MANAGER        14 M   12/18/1956    26150.00      500.00     2092.00
000110 SALESREP       19 M   11/05/1929    46500.00      900.00     3720.00
000120 CLERK          14 M   10/18/1942    29250.00      600.00     2340.00
000130 ANALYST        16 F   09/15/1925    23800.00      500.00     1904.00
000140 ANALYST        18 F   01/19/1946    28420.00      600.00     2274.00
000150 DESIGNER       16 M   05/17/1947    25280.00      500.00     2022.00
000160 DESIGNER       17 F   04/12/1955    22250.00      400.00     1780.00
000170 DESIGNER       16 M   01/05/1951    24680.00      500.00     1974.00
000180 DESIGNER       17 F   02/21/1949    21340.00      500.00     1707.00
000190 DESIGNER       16 M   06/25/1952    20450.00      400.00     1636.00
000200 DESIGNER       16 M   05/29/1941    27740.00      600.00     2217.00
000210 DESIGNER       17 M   02/23/1953    18270.00      400.00     1462.00
000220 DESIGNER       18 F   03/19/1948    29840.00      600.00     2387.00
000230 CLERK          14 M   05/30/1935    22180.00      400.00     1774.00
000240 CLERK          17 M   03/31/1954    28760.00      600.00     2301.00
000250 CLERK          15 M   11/12/1939    19180.00      400.00     1534.00
000260 CLERK          16 F   10/05/1936    17250.00      300.00     1380.00
000270 CLERK          15 F   05/26/1953    27380.00      500.00     2190.00
000280 OPERATOR       17 F   03/28/1936    26250.00      500.00     2100.00
000290 OPERATOR       12 M   07/09/1946    15340.00      300.00     1227.00
000300 OPERATOR       14 M   10/27/1936    17750.00      400.00     1420.00
000310 OPERATOR       12 F   04/21/1931    15900.00      300.00     1272.00
000320 FIELDREP       16 M   08/11/1932    19950.00      400.00     1596.00
000330 FIELDREP       14 M   07/18/1941    25370.00      500.00     2030.00
000340 FIELDREP       16 M   05/17/1926    23840.00      500.00     1907.00

  32 record(s) selected.
```

# SQL Self-Test

## Part B

## Workshop Problems

# Table of Contents

# Workshop 1—Familiarization

## Execute SQL through the Command Center or SPUFI

1. ___ **Verify table contents.** Code full-table SELECT statements for the DEPARTMENT and EMPLOYEE tables.

   Verify that the DEPARTMENT and EMPLOYEE tables contain the contents documented in Part A. If you find any discrepancies, either code INSERT, UPDATE, and DELETE statements to bring your tables into compliance with Part A, or recognize that your actual solutions may differ from the documented solutions accordingly.

***Note:*** *If you are using workstation DB2's Command Center to do these workshops, you should find no discrepancies from Part A. If you are using mainframe DB2's SAMPLE database, however, there are some subtle differences:*

- *The tables are named DEPT and EMP instead of DEPARTMENT and EMPLOYEE.*
- *The DEPT table has five more rows than the DEPARTMENT table, representing five branch offices.*
- *The EDLEVEL column allows nulls in the EMP table but not in the EMPLOYEE table. (Do nothing to "correct" this, but be aware that INSERT statements may or may not require a value in EDLEVEL.)*

# Workshop 2—Data Retrieval

*Note:* *In all of the following SELECT statements, be sure to provide meaningful column names for the result-set columns, as shown in the expected results.*

## Execute SQL through the Command Center or SPUFI

1. ___ Show the employee number, first name, last name, and commission of all employees with a first name containing the letter E, a last name starting with S, a valid sex (test for F or M), and a commission in the 2,000's. *Code this with just four predicates!*

```
EMPNO  FIRSTNME     LASTNAME        COMM
------ ------------ --------------- -----------
000100 THEODORE     SPENSER              2092.00
000280 ETHEL        SCHNEIDER            2100.00
  2 record(s) selected.
```

2. ___ Show the employee number, last name, and gross pay (salary plus bonus plus commission) of all employees with a gross pay greater than 40,000. Show the highest-paid employees first.

```
EMPNO  LASTNAME        GROSS PAY
------ --------------- -------------
000010 HAAS                57970.00
000110 LUCCHESSI           51120.00
000020 THOMPSON            45350.00
000050 GEYER               44189.00
000030 KWAN                42110.00
  5 record(s) selected.
```

3. ___ Show the employee number, first name, middle initial, last name, and education level of all employees with *just one* E in their first name and more than 16 years of education. Sequence the results by descending education level; within education level, use ascending employee number.

```
EMPNO  FIRSTNME     MIDINIT LASTNAME        EDLEVEL
------ ------------ ------- --------------- -------
000110 VINCENZO     G       LUCCHESSI              19
000010 CHRISTINE    I       HAAS                   18
000020 MICHAEL      L       THOMPSON               18
000240 SALVATORE    M       MARINO                 17
  4 record(s) selected.
```

# Workshop 3—Data Modification

*Note:* *INSERT, UPDATE, and DELETE statements provide only an SQLSTATE as output; there is no output result set. To verify that your inserts, updates, and deletes are successful, code "before" and "after" SELECT statements with appropriate WHERE clauses to return the affected row or rows.*

### Execute SQL through the Command Center or SPUFI

1. ___ Add a row to the EMPLOYEE table for yourself. Provide an employee number, first name, middle initial (or blank), last name, and education level—these are the required (i.e., NOT NULL) columns in the table. Provide a sex and a salary, but let all the other optional (in this case, nullable) columns default to null.

2. ___ Add two rows to the DEPARTMENT table with one INSERT statement. (In mainframe DB2, you may have to use a separate INSERT statement per row of values.) Use department numbers D99 and E99. Provide appropriate department names, administrating departments, and locations, but allow both manager numbers to default to null.

3. ___ Update your row in the EMPLOYEE table. Give yourself a 10% raise and a bonus equal to half your existing salary.

4. ___ Update all rows in the DEPARTMENT table with nonnull locations (this should be just the two rows you inserted in step #2 above). Set the manager number of both to your employee number.

5. ___ Delete the two rows you inserted into the DEPARTMENT table. Code your WHERE clause carefully to ensure that it is just those two rows that get deleted.

6. ___ Commit your changes to the database. Code and run the following SQL statement:

```
COMMIT;
```

*Note:* *At this point, the only change to the SAMPLE tables should be the one additional row in the EMPLOYEE table for yourself. We shall use this row in subsequent exercises, so leave it in the EMPLOYEE table for now.*

# Workshop 4—Customizing Result Sets

*Note:* *In all of the following SELECT statements, be sure to provide meaningful column names for the result-set columns, as shown in the expected results.*

## Execute SQL through the Command Center or SPUFI

1. ___ Code a SELECT statement to return the employee numbers of everyone working in department A00. Show the employee number as it appears in the table, but then also show it as an integer number, as a packed decimal number (with appropriate precision and scale), and as a floating-point number. (In mainframe DB2, you may have to use nested functions to return the integer and floating-point columns.)

```
EMPNO  EMPINT      EMPDEC   EMPDBL
------ ----------- -------- ------------------------
000010          10      10.   +1.00000000000000E+001
000110         110     110.   +1.10000000000000E+002
000120         120     120.   +1.20000000000000E+002
  3 record(s) selected.
```

2. ___ Code a SELECT statement to return the employee number, birth date, and hire date of all employees without middle initials. Display birth date in Japanese Industrial Standard (JIS) format and hire date in European (EUR) format.

```
EMPNO  JBIRTHDATE EHIREDATE
------ ---------- ----------
000120 1942-10-18 05.12.1963
000150 1947-05-17 12.02.1972
000200 1941-05-29 03.03.1966
000330 1941-07-18 23.02.1976
  4 record(s) selected.
```

3. ___ Modify your query from question #2 to show just the month and day portion of both date columns.

```
EMPNO  JBIRTHMMDD EHIREDDMM
------ ---------- ---------
000120 10-18      05.12
000150 05-17      12.02
000200 05-29      03.03
000330 07-18      23.02
  4 record(s) selected.
```

4. ___ Code a SELECT statement to return your row from the employee table. Show the employee number, salary, bonus, and commission, but if any of them is null, return zeroes instead.

```
EMPNO  SALARY       BONUS        COMM
------ ------------ ------------ ------------
123456  1100000.00   500000.00         0.00
  1 record(s) selected.
```

5. ___ Code a SELECT statement to return the employee number, first name, middle initial, and last name of all employees in department D11, but return a null instead of middle initial if middle initial is blank.

```
EMPNO  FIRSTNME     MIDINIT LASTNAME
------ ------------ ------- ---------------
000060 IRVING       F       STERN
000150 BRUCE        -       ADAMSON
000160 ELIZABETH    R       PIANKA
000170 MASATOSHI    J       YOSHIMURA
000180 MARILYN      S       SCOUTTEN
000190 JAMES        H       WALKER
000200 DAVID        -       BROWN
000210 WILLIAM      T       JONES
000220 JENNIFER     K       LUTZ
  9 record(s) selected.
```

# Workshop 5—Inner Joins

### Execute SQL through the Command Center or SPUFI

1. ___ Get a list of all employees assigned to a work department. Show the department number and name and the employee number and last name. Sequence the result by last name within department number.

```
DEPTNO DEPTNAME                          EMPNO  LASTNAME
------ --------------------------------- ------ -------------…
A00    SPIFFY COMPUTER SERVICE DIV.      000010 HAAS
A00    SPIFFY COMPUTER SERVICE DIV.      000110 LUCCHESSI
A00    SPIFFY COMPUTER SERVICE DIV.      000120 O'CONNELL
B01    PLANNING                          000020 THOMPSON
C01    INFORMATION CENTER                000030 KWAN
C01    INFORMATION CENTER                000140 NICHOLLS
C01    INFORMATION CENTER                000130 QUINTANA
D11    MANUFACTURING SYSTEMS             000150 ADAMSON
D11    MANUFACTURING SYSTEMS             000200 BROWN
D11    MANUFACTURING SYSTEMS             000210 JONES
D11    MANUFACTURING SYSTEMS             000220 LUTZ
D11    MANUFACTURING SYSTEMS             000160 PIANKA
D11    MANUFACTURING SYSTEMS             000180 SCOUTTEN
D11    MANUFACTURING SYSTEMS             000060 STERN
D11    MANUFACTURING SYSTEMS             000190 WALKER
D11    MANUFACTURING SYSTEMS             000170 YOSHIMURA
D21    ADMINISTRATION SYSTEMS            000230 JEFFERSON
D21    ADMINISTRATION SYSTEMS            000260 JOHNSON
D21    ADMINISTRATION SYSTEMS            000240 MARINO
D21    ADMINISTRATION SYSTEMS            000270 PEREZ
D21    ADMINISTRATION SYSTEMS            000070 PULASKI
D21    ADMINISTRATION SYSTEMS            000250 SMITH
E01    SUPPORT SERVICES                  000050 GEYER
E11    OPERATIONS                        000090 HENDERSON
E11    OPERATIONS                        000290 PARKER
E11    OPERATIONS                        000280 SCHNEIDER
E11    OPERATIONS                        000310 SETRIGHT
E11    OPERATIONS                        000300 SMITH
E21    SOFTWARE SUPPORT                  000340 GOUNOT
E21    SOFTWARE SUPPORT                  000330 LEE
E21    SOFTWARE SUPPORT                  000320 MEHTA
E21    SOFTWARE SUPPORT                  000100 SPENSER
  32 record(s) selected.
```

2. ___ Modify your solution to question #1. Limit the results to employees that work only in departments administered by A00.

```
DEPTNO DEPTNAME                          EMPNO  LASTNAME
------ ------------------------------ ------ -------------…
A00    SPIFFY COMPUTER SERVICE DIV.    000010 HAAS
A00    SPIFFY COMPUTER SERVICE DIV.    000110 LUCCHESSI
A00    SPIFFY COMPUTER SERVICE DIV.    000120 O'CONNELL
B01    PLANNING                        000020 THOMPSON
C01    INFORMATION CENTER              000030 KWAN
C01    INFORMATION CENTER              000140 NICHOLLS
C01    INFORMATION CENTER              000130 QUINTANA
E01    SUPPORT SERVICES                000050 GEYER
  8 record(s) selected.
```

3. ___ Verify the logic of your solution to question #2 by modifying it. Replace the department name with the administrating department's name. Name this column REPORTS TO.

```
DEPTNO REPORTS TO                        EMPNO  LASTNAME
------ ------------------------------ ------ -------------…
A00    SPIFFY COMPUTER SERVICE DIV.    000010 HAAS
A00    SPIFFY COMPUTER SERVICE DIV.    000110 LUCCHESSI
A00    SPIFFY COMPUTER SERVICE DIV.    000120 O'CONNELL
B01    SPIFFY COMPUTER SERVICE DIV.    000020 THOMPSON
C01    SPIFFY COMPUTER SERVICE DIV.    000030 KWAN
C01    SPIFFY COMPUTER SERVICE DIV.    000140 NICHOLLS
C01    SPIFFY COMPUTER SERVICE DIV.    000130 QUINTANA
E01    SPIFFY COMPUTER SERVICE DIV.    000050 GEYER
  8 record(s) selected.
```

4. ___ Modify your solution to question #3. Instead of limiting the results to employees that work only in departments administered by A00, limit the results to employees that work only in E-series departments.

```
DEPTNO REPORTS TO                        EMPNO  LASTNAME
------ ------------------------------ ------ -------------…
E01    SPIFFY COMPUTER SERVICE DIV.    000050 GEYER
E11    SUPPORT SERVICES                000090 HENDERSON
E11    SUPPORT SERVICES                000290 PARKER
E11    SUPPORT SERVICES                000280 SCHNEIDER
E11    SUPPORT SERVICES                000310 SETRIGHT
E11    SUPPORT SERVICES                000300 SMITH
E21    SUPPORT SERVICES                000340 GOUNOT
E21    SUPPORT SERVICES                000330 LEE
E21    SUPPORT SERVICES                000320 MEHTA
E21    SUPPORT SERVICES                000100 SPENSER
  10 record(s) selected.
```

# Workshop 6—Column Functions and Grouping

*Note:* *In all of the following SELECT statements, be sure to provide meaningful column names for the result-set columns, as shown in the expected results.*

### Execute SQL through the Command Center or SPUFI

1. ___ Find the minimum, maximum, and average commission of all employees.

```
MIN_COMM     MAX_COMM     AVG_COMM
-----------  -----------  --------------------------------
    1227.00      4220.00      2184.15625000000000000000000
  SQLWARN2=W,    SQLSTATE=01003
  1 record(s) selected.
```

   Why does this return a warning? To find help for an SQLSTATE (in workstation DB2 only), execute the question mark (?) help command followed by the SQLSTATE code, as follows:

```
? 01003
```

   _____

   _____

2. ___ Modify your solution to question #1. Calculate the same three column functions, but this time only for employees who have a commission. You should get the same results as above but without the warning.

3. ___ Find the total number of employees, the number of employees with a commission, the number of different commission values, and the number of different bonus values. (In mainframe DB2, you may not be able to return the second column.)

```
#EMPLOYEES   #W/ COMM     #COMM VALS   #BONUS VALS
-----------  -----------  -----------  -----------
          33           32           32            9
  SQLWARN2=W,    SQLSTATE=01003
  1 record(s) selected.
```

4. ___ Find the total pay (salary plus bonus plus commission) per work department. If any of the numeric columns is null, use zero in the calculation. Call the result column TOTAL PAY, round it to 0 decimal places, and return it as a large integer. Sequence the result rows in descending order of total pay.

```
WORKDEPT TOTAL PAY
-------- -----------
-            1600000
D11           244265
D21           165892
A00           141280
E11           115489
E21           104835
C01            99608
B01            45350
E01            44189
  9 record(s) selected.
```

5. ___ Modify your solution to question #4. Add the columns #EMPS and PER PERSON to the result. Sequence the result rows in descending order of per-person pay.

```
WORKDEPT #EMPS       TOTAL PAY   PER PERSON
-------- ----------- ----------- -----------
-                  1     1600000     1600000
A00                3      141280       47093
B01                1       45350       45350
E01                1       44189       44189
C01                3       99608       33203
D21                6      165892       27649
D11                9      244265       27141
E21                4      104835       26209
E11                5      115489       23098
  9 record(s) selected.
```

6. ___ Modify your solution to question #5. Limit the result to just *nonnull* work departments with a total pay greater than 100,000.

```
WORKDEPT #EMPS       TOTAL PAY   PER PERSON
-------- ----------- ----------- -----------
A00                3      141280       47093
D21                6      165892       27649
D11                9      244265       27141
E21                4      104835       26209
E11                5      115489       23098
  5 record(s) selected.
```

7. ___ Modify your solution to question #5 (not #6). Remove the PER PERSON column from the result set, but maintain the same row sequence. (This may or may not be possible in your version of mainframe DB2.)

```
WORKDEPT #EMPS        TOTAL PAY
-------- ---------- -----------
-                 1     1600000
A00               3      141280
B01               1       45350
E01               1       44189
C01               3       99608
D21               6      165892
D11               9      244265
E21               4      104835
E11               5      115489
  9 record(s) selected.
```

# Workshop 7—Subqueries and Unions

*Note:* *In all of the following SELECT statements, be sure to provide meaningful column names for the result-set columns, as shown in the expected results.*

## Execute SQL through the Command Center or SPUFI

1. ___ Find the average salary, bonus, and commission of all employees assigned to a work department.

```
SALARY                  BONUS                   COMM
----------------------- ----------------------- -----------------------
   27303.59375000...       540.62500000...         2184.15625000...
  1 record(s) selected.
```

2. ___ Now find all employees assigned to a work department, who have a salary greater than the average salary calculated in question #1. Show employee number, first and last names, and salary. Return the result rows in descending salary sequence.

```
EMPNO  FIRSTNME     LASTNAME        SALARY
------ ------------ --------------- -----------
000010 CHRISTINE    HAAS               52750.00
000110 VINCENZO     LUCCHESSI          46500.00
000020 MICHAEL      THOMPSON           41250.00
000050 JOHN         GEYER              40175.00
000030 SALLY        KWAN               38250.00
000070 EVA          PULASKI            36170.00
000060 IRVING       STERN              32250.00
000220 JENNIFER     LUTZ               29840.00
000090 EILEEN       HENDERSON          29750.00
000120 SEAN         O'CONNELL          29250.00
000240 SALVATORE    MARINO             28760.00
000140 HEATHER      NICHOLLS           28420.00
000200 DAVID        BROWN              27740.00
000270 MARIA        PEREZ              27380.00
  14 record(s) selected.
```

3. ___ Modify your solution to question #2. Find all employees with a salary greater than the average salary of everyone *in their same work department*.

```
EMPNO  FIRSTNME     LASTNAME        SALARY
------ ------------ --------------- -----------
000010 CHRISTINE    HAAS               52750.00
000110 VINCENZO     LUCCHESSI          46500.00
000030 SALLY        KWAN               38250.00
000070 EVA          PULASKI            36170.00
000060 IRVING       STERN              32250.00
000220 JENNIFER     LUTZ               29840.00
000090 EILEEN       HENDERSON          29750.00
000240 SALVATORE    MARINO             28760.00
000200 DAVID        BROWN              27740.00
000270 MARIA        PEREZ              27380.00
000280 ETHEL        SCHNEIDER          26250.00
000100 THEODORE     SPENSER            26150.00
000330 WING         LEE                25370.00
000150 BRUCE        ADAMSON            25280.00
000170 MASATOSHI    YOSHIMURA          24680.00
000340 JASON        GOUNOT             23840.00
  16 record(s) selected.
```

4. ___ Verify your solution to question #3. Remove first name from the result set. Add work department and departmental average salary, rounded and changed to DECIMAL(9,2). Code *at least* two solutions, one involving two scalar fullselects, and another involving a join to a nested table expression.

```
EMPNO  LASTNAME        WORKDEPT SALARY      AVERAGE
------ --------------- -------- ----------- -----------
000010 HAAS            A00         52750.00    42833.33
000110 LUCCHESSI       A00         46500.00    42833.33
000030 KWAN            C01         38250.00    30156.67
000070 PULASKI         D21         36170.00    25153.33
000060 STERN           D11         32250.00    24677.78
000220 LUTZ            D11         29840.00    24677.78
000090 HENDERSON       E11         29750.00    20998.00
000240 MARINO          D21         28760.00    25153.33
000200 BROWN           D11         27740.00    24677.78
000270 PEREZ           D21         27380.00    25153.33
000280 SCHNEIDER       E11         26250.00    20998.00
000100 SPENSER         E21         26150.00    23827.50
000330 LEE             E21         25370.00    23827.50
000150 ADAMSON         D11         25280.00    24677.78
000170 YOSHIMURA       D11         24680.00    24677.78
000340 GOUNOT          E21         23840.00    23827.50
  16 record(s) selected.
```

5. ___ The following three SELECT statements return three different results. The first returns detail-level information, the second returns subtotals by department, and the third returns a grand-total row. Combine these three results into a single result. Hint: The result sets must be made compatible—the same number of columns, with positionally compatible data types.

```
SELECT WORKDEPT, EMPNO, SALARY
  FROM EMPLOYEE
  WHERE WORKDEPT IS NOT NULL
  ORDER BY WORKDEPT, EMPNO;
SELECT WORKDEPT, SUM(SALARY)
  FROM EMPLOYEE
  WHERE WORKDEPT IS NOT NULL
  GROUP BY WORKDEPT
  ORDER BY WORKDEPT;
SELECT SUM(SALARY)
  FROM EMPLOYEE
  WHERE WORKDEPT IS NOT NULL;
```

Following are the expected combined results. Note the sequence of the result rows as well as the nulls in the subtotal and grand-total rows. Make sure that the SALARY column is defined as DECIMAL(11,2).

```
WORKDEPT EMPNO  SALARY
-------- ------ -------------
A00      000010      52750.00
A00      000110      46500.00
A00      000120      29250.00
A00      -          128500.00
B01      000020      41250.00
B01      -           41250.00
C01      000030      38250.00
C01      000130      23800.00
C01      000140      28420.00
C01      -           90470.00
D11      000060      32250.00
D11      000150      25280.00
D11      000160      22250.00
D11      000170      24680.00
D11      000180      21340.00
D11      000190      20450.00
D11      000200      27740.00
D11      000210      18270.00
D11      000220      29840.00
D11      -          222100.00
D21      000070      36170.00
D21      000230      22180.00
D21      000240      28760.00
```

```
D21      000250      19180.00
D21      000260      17250.00
D21      000270      27380.00
D21      -          150920.00
E01      000050      40175.00
E01      -           40175.00
E11      000090      29750.00
E11      000280      26250.00
E11      000290      15340.00
E11      000300      17750.00
E11      000310      15900.00
E11      -          104990.00
E21      000100      26150.00
E21      000320      19950.00
E21      000330      25370.00
E21      000340      23840.00
E21      -           95310.00
-        -          873715.00
  41 record(s) selected.
```

# SQL Self-Test

## Part C

## Workshop Solutions

# Table of Contents

# Workshop 1—Familiarization

## Execute SQL through the Command Center or SPUFI

1. ___ **Verify table contents.** Code full-table SELECT statements for the DEPARTMENT and EMPLOYEE tables.

```
SELECT *
  FROM DEPARTMENT;
SELECT *
  FROM EMPLOYEE;
```

Verify that the DEPARTMENT and EMPLOYEE tables contain the contents documented in Part A. If you find any discrepancies, either code INSERT, UPDATE, and DELETE statements to bring your tables into compliance with Part A, or recognize that your actual solutions may differ from the documented solutions accordingly.

***Note:*** *If you are using workstation DB2's Command Center to do these workshops, you should find no discrepancies from Part A. If you are using mainframe DB2's SAMPLE database, however, there are some subtle differences:*

- *The tables are named DEPT and EMP instead of DEPARTMENT and EMPLOYEE.*

```
CREATE SYNONYM DEPARTMENT FOR [schemaname.]DEPT;
CREATE SYNONYM EMPLOYEE   FOR [schemaname.]EMP;
```

- *The DEPT table has five more rows than the DEPARTMENT table, representing five branch offices.*

```
DELETE FROM DEPARTMENT
  WHERE DEPTNO > 'E21';
```

- *The EDLEVEL column allows nulls in the EMP table but not in the EMPLOYEE table. (Do nothing to "correct" this, but be aware that INSERT statements may or may not require a value in EDLEVEL.)*

# Workshop 2—Data Retrieval

***Note:*** *In all of the following SELECT statements, be sure to provide meaningful column names for the result-set columns, as shown in the expected results.*

## Execute SQL through the Command Center or SPUFI

1. ___ Show the employee number, first name, last name, and commission of all employees with a first name containing the letter E, a last name starting with S, a valid sex (test for F or M), and a commission in the 2,000's. *Code this with just four predicates!*

```
SELECT EMPNO, FIRSTNME, LASTNAME, COMM
  FROM EMPLOYEE
  WHERE FIRSTNME LIKE '%E%'
    AND LASTNAME LIKE 'S%'
    AND SEX IN ('F', 'M')
    AND COMM BETWEEN 2000.00 AND 2999.99;
```

2. ___ Show the employee number, last name, and gross pay (salary plus bonus plus commission) of all employees with a gross pay greater than 40,000. Show the highest-paid employees first.

```
SELECT EMPNO, LASTNAME
     , SALARY + BONUS + COMM AS "GROSS PAY"
  FROM EMPLOYEE
  WHERE SALARY + BONUS + COMM > 40000.
  ORDER BY "GROSS PAY" DESC;
```

3. ___ Show the employee number, first name, middle initial, last name, and education level of all employees with *just one* E in their first name and more than 16 years of education. Sequence the results by descending education level; within education level, use ascending employee number.

```
SELECT EMPNO, FIRSTNME, MIDINIT, LASTNAME, EDLEVEL
  FROM EMPLOYEE
  WHERE FIRSTNME LIKE '%E%'
    AND FIRSTNME NOT LIKE '%E%E%'
    AND EDLEVEL > 16
  ORDER BY EDLEVEL DESC, EMPNO;
```

# Workshop 3—Data Modification

*Note:* *INSERT, UPDATE, and DELETE statements provide only an SQLSTATE as output; there is no output result set. To verify that your inserts, updates, and deletes are successful, code "before" and "after" SELECT statements with appropriate WHERE clauses to return the affected row or rows.*

### Execute SQL through the Command Center or SPUFI

1. ___ Add a row to the EMPLOYEE table for yourself. Provide an employee number, first name, middle initial (or blank), last name, and education level—these are the required (i.e., NOT NULL) columns in the table. Provide a sex and a salary, but let all the other optional (in this case, nullable) columns default to null.

```
INSERT INTO EMPLOYEE
          ( EMPNO, FIRSTNME, MIDINIT, LASTNAME, EDLEVEL
          , SEX, SALARY)
  VALUES ( '123456', 'first', 'm', 'last', 99
          , 'M', 1000000.);
```

2. ___ Add two rows to the DEPARTMENT table with one INSERT statement. (In mainframe DB2, you may have to use a separate INSERT statement per row of values.) Use department numbers D99 and E99. Provide appropriate department names, administrating departments, and locations, but allow both manager numbers to default to null.

```
INSERT INTO DEPARTMENT
          (DEPTNO, DEPTNAME, ADMRDEPT, LOCATION)
  VALUES ('D99', 'Department D99', 'A00', 'Location D99')
       , ('E99', 'Department E99', 'A00', 'Location E99');
```

*Alternative solution for older versions of mainframe DB2:*

```
INSERT INTO DEPARTMENT
          (DEPTNO, DEPTNAME, ADMRDEPT, LOCATION)
  VALUES ('D99', 'Department D99', 'A00', 'Location D99');
INSERT INTO DEPARTMENT
          (DEPTNO, DEPTNAME, ADMRDEPT, LOCATION)
  VALUES ('E99', 'Department E99', 'A00', 'Location E99');
```

3. ___ Update your row in the EMPLOYEE table. Give yourself a 10% raise and a bonus equal to half your existing salary.

```
UPDATE EMPLOYEE
  SET SALARY = SALARY * 1.1
    , BONUS = SALARY * .5
  WHERE EMPNO = '123456';
```

4. ___ Update all rows in the DEPARTMENT table with nonnull locations (this should be just the two rows you inserted in step #2 above). Set the manager number of both to your employee number.

```
UPDATE DEPARTMENT
  SET MGRNO = '123456'
  WHERE LOCATION IS NOT NULL;
```

5. ___ Delete the two rows you inserted into the DEPARTMENT table. Code your WHERE clause carefully to ensure that it is just those two rows that get deleted.

```
DELETE FROM DEPARTMENT
  WHERE DEPTNO IN ('D99', 'E99');
--WHERE LOCATION IS NOT NULL;
--WHERE MGRNO = '123456';
```
*(Any of these WHERE clauses should work.)*

6. ___ Commit your changes to the database. Code and run the following SQL statement:

```
COMMIT;
```
*(Be sure to do this step!)*

***Note:*** *At this point, the only change to the SAMPLE tables should be the one additional row in the EMPLOYEE table for yourself. We shall use this row in subsequent exercises, so leave it in the EMPLOYEE table for now.*

# Workshop 4—Customizing Result Sets

*Note:* *In all of the following SELECT statements, be sure to provide meaningful column names for the result-set columns, as shown in the expected results.*

## Execute SQL through the Command Center or SPUFI

1. ___ Code a SELECT statement to return the employee numbers of everyone working in department A00. Show the employee number as it appears in the table, but then also show it as an integer number, as a packed decimal number (with appropriate precision and scale), and as a floating-point number. (In mainframe DB2, you may have to use nested functions to return the integer and floating-point columns.)

```
SELECT EMPNO
     , INTEGER(EMPNO) AS EMPINT
     , DECIMAL(EMPNO,6) AS EMPDEC
     , DOUBLE(EMPNO) AS EMPDBL
  FROM EMPLOYEE
  WHERE WORKDEPT = 'A00';
```

*Alternative solution for older versions of mainframe DB2:*

```
SELECT EMPNO
     , INTEGER(DECIMAL(EMPNO,6)) AS EMPINT
     , DECIMAL(EMPNO,6) AS EMPDEC
     , DOUBLE(DECIMAL(EMPNO,6)) AS EMPDBL
  FROM EMPLOYEE
  WHERE WORKDEPT = 'A00';
```

2. ___ Code a SELECT statement to return the employee number, birth date, and hire date of all employees without middle initials. Display birth date in Japanese Industrial Standard (JIS) format and hire date in European (EUR) format.

```
SELECT EMPNO
     , CHAR(BIRTHDATE,JIS) AS JBIRTHDATE
     , CHAR(HIREDATE,EUR) AS EHIREDATE
  FROM EMPLOYEE
  WHERE MIDINIT = ' ';
```

3. ___ Modify your query from question #2 to show just the month and day portion of both date columns.

```
SELECT EMPNO
     , SUBSTR(CHAR(BIRTHDATE,JIS),6) AS JBIRTHMMDD
     , SUBSTR(CHAR(HIREDATE,EUR),1,5) AS EHIREDDMM
  FROM EMPLOYEE
  WHERE MIDINIT = ' ';
```

4. ___ Code a SELECT statement to return your row from the employee table. Show the employee number, salary, bonus, and commission, but if any of them is null, return zeroes instead.

```
SELECT EMPNO
     , COALESCE(SALARY,0.) AS SALARY
     , COALESCE(BONUS,0.) AS BONUS
     , COALESCE(COMM,0.) AS COMM
  FROM EMPLOYEE
  WHERE EMPNO = '123456';
```

5. ___ Code a SELECT statement to return the employee number, first name, middle initial, and last name of all employees in department D11, but return a null instead of middle initial if middle initial is blank.

```
SELECT EMPNO
     , FIRSTNME
     , NULLIF(MIDINIT,' ') AS MIDINIT
     , LASTNAME
  FROM EMPLOYEE
  WHERE WORKDEPT = 'D11';
```

# Workshop 5—Inner Joins

### Execute SQL through the Command Center or SPUFI

1. ___ Get a list of all employees assigned to a work department. Show the department number and name and the employee number and last name. Sequence the result by last name within department number.

```
SELECT D.DEPTNO, DEPTNAME, E.EMPNO, LASTNAME
  FROM DEPARTMENT D, EMPLOYEE E
  WHERE D.DEPTNO = E.WORKDEPT
  ORDER BY DEPTNO, LASTNAME;
```

2. ___ Modify your solution to question #1. Limit the results to employees that work only in departments administered by A00.

```
SELECT D.DEPTNO, DEPTNAME, E.EMPNO, LASTNAME
  FROM DEPARTMENT D, EMPLOYEE E
  WHERE D.DEPTNO = E.WORKDEPT
    AND ADMRDEPT = 'A00'
  ORDER BY DEPTNO, LASTNAME;
```

3. ___ Verify the logic of your solution to question #2 by modifying it. Replace the department name with the administrating department's name. Name this column REPORTS TO.

```
SELECT D.DEPTNO, A.DEPTNAME AS "REPORTS TO"
     , E.EMPNO, LASTNAME
  FROM DEPARTMENT D, EMPLOYEE E, DEPARTMENT A
  WHERE D.DEPTNO = E.WORKDEPT
    AND D.ADMRDEPT = A.DEPTNO
    AND D.ADMRDEPT = 'A00'
  ORDER BY DEPTNO, LASTNAME;
```

4. ___ Modify your solution to question #3. Instead of limiting the results to employees that work only in departments administered by A00, limit the results to employees that work only in E-series departments.

```
SELECT D.DEPTNO, A.DEPTNAME AS "REPORTS TO"
     , E.EMPNO, LASTNAME
  FROM DEPARTMENT D, EMPLOYEE E, DEPARTMENT A
  WHERE D.DEPTNO = E.WORKDEPT
    AND D.ADMRDEPT = A.DEPTNO
    AND D.DEPTNO LIKE 'E%'
  ORDER BY DEPTNO, LASTNAME;
```

# Workshop 6—Column Functions and Grouping

***Note:*** *In all of the following SELECT statements, be sure to provide meaningful column names for the result-set columns, as shown in the expected results.*

### Execute SQL through the Command Center or SPUFI

1. ___ Find the minimum, maximum, and average commission of all employees.

```
SELECT MIN(COMM) AS MIN_COMM
     , MAX(COMM) AS MAX_COMM
     , AVG(COMM) AS AVG_COMM
  FROM EMPLOYEE;
```

   Why does this return a warning? To find help for an SQLSTATE (in workstation DB2 only), execute the question mark (?) help command followed by the SQLSTATE code, as follows:

```
? 01003
```

   SQLSTATE 01003: Null values were eliminated from the argument of a column function.

2. ___ Modify your solution to question #1. Calculate the same three column functions, but this time only for employees who have a commission. You should get the same results as above but without the warning.

```
SELECT MIN(COMM) AS MIN_COMM
     , MAX(COMM) AS MAX_COMM
     , AVG(COMM) AS AVG_COMM
  FROM EMPLOYEE
  WHERE COMM IS NOT NULL;
```

3. ___ Find the total number of employees, the number of employees with a commission, the number of different commission values, and the number of different bonus values. (In mainframe DB2, you may not be able to return the second column.)

```
SELECT COUNT(*) AS #EMPLOYEES
     , COUNT(COMM) AS "#W/ COMM"
     , COUNT(DISTINCT COMM) AS "#COMM VALS"
     , COUNT(DISTINCT BONUS) AS "#BONUS VALS"
  FROM EMPLOYEE;
```

4. ___ Find the total pay (salary plus bonus plus commission) per work department. If any of the numeric columns is null, use zero in the calculation. Call the result column TOTAL PAY, round it to 0 decimal places, and return it as a large integer. Sequence the result rows in descending order of total pay.

```
SELECT WORKDEPT
     , INTEGER( ROUND( SUM(
         COALESCE(SALARY,0.) +
         COALESCE(BONUS,0.) +
         COALESCE(COMM,0.)
       ) ,0) ) AS "TOTAL PAY"
  FROM EMPLOYEE
  GROUP BY WORKDEPT
  ORDER BY "TOTAL PAY" DESC;
```

5. ___ Modify your solution to question #4. Add the columns #EMPS and PER PERSON to the result. Sequence the result rows in descending order of per-person pay.

```
SELECT WORKDEPT
     , COUNT(*) AS #EMPS
     , INTEGER( ROUND( SUM(
         COALESCE(SALARY,0.) +
         COALESCE(BONUS,0.) +
         COALESCE(COMM,0.)
       ) ,0) ) AS "TOTAL PAY"
     , INTEGER( ROUND( AVG(
         COALESCE(SALARY,0.) +
         COALESCE(BONUS,0.) +
         COALESCE(COMM,0.)
       ) ,0) ) AS "PER PERSON"
  FROM EMPLOYEE
  GROUP BY WORKDEPT
  ORDER BY "PER PERSON" DESC;
```

6. ___ Modify your solution to question #5. Limit the result to just *nonnull* work departments with a total pay greater than 100,000.

```
SELECT WORKDEPT
     , COUNT(*) AS #EMPS
     , INTEGER( ROUND( SUM(
         COALESCE(SALARY,0.) +
         COALESCE(BONUS,0.) +
         COALESCE(COMM,0.)
       ) ,0) ) AS "TOTAL PAY"
     , INTEGER( ROUND( AVG(
         COALESCE(SALARY,0.) +
         COALESCE(BONUS,0.) +
         COALESCE(COMM,0.)
       ) ,0) ) AS "PER PERSON"
  FROM EMPLOYEE
  WHERE WORKDEPT IS NOT NULL (Don't put this in the HAVING clause!)
  GROUP BY WORKDEPT
  HAVING SUM( COALESCE(SALARY,0.) +
              COALESCE(BONUS,0.) +
              COALESCE(COMM,0.) ) > 100000.
  ORDER BY "PER PERSON" DESC;
```

7. ___ Modify your solution to question #5 (not #6). Remove the PER PERSON column from the result set, but maintain the same row sequence. (This may or may not be possible in your version of mainframe DB2.)

```
SELECT WORKDEPT
     , COUNT(*) AS #EMPS
     , INTEGER( ROUND( SUM(
         COALESCE(SALARY,0.) +
         COALESCE(BONUS,0.) +
         COALESCE(COMM,0.)
       ) ,0) ) AS "TOTAL PAY"
  FROM EMPLOYEE
  GROUP BY WORKDEPT
  ORDER BY "TOTAL PAY" / #EMPS DESC;
```

*(In the syntax charts, this is known as a "sort key expression.")*

# Workshop 7—Subqueries and Unions

*Note:* *In all of the following SELECT statements, be sure to provide meaningful column names for the result-set columns, as shown in the expected results.*

### Execute SQL through the Command Center or SPUFI

1. ___ Find the average salary, bonus, and commission of all employees assigned to a work department.

```
SELECT AVG(SALARY) AS SALARY
     , AVG(BONUS) AS BONUS
     , AVG(COMM) AS COMM
  FROM EMPLOYEE
  WHERE WORKDEPT IS NOT NULL;
```

2. ___ Now find all employees assigned to a work department, who have a salary greater than the average salary calculated in question #1. Show employee number, first and last names, and salary. Return the result rows in descending salary sequence.

```
SELECT EMPNO, FIRSTNME, LASTNAME, SALARY
  FROM EMPLOYEE
  WHERE WORKDEPT IS NOT NULL
    AND SALARY > (SELECT AVG(SALARY)
                    FROM EMPLOYEE
                    WHERE WORKDEPT IS NOT NULL)
  ORDER BY SALARY DESC;
```

*(Did you test for WORKDEPT IS NOT NULL in both places?)*

3. ___ Modify your solution to question #2. Find all employees with a salary greater than the average salary of everyone *in their same work department*.

```
SELECT EMPNO, FIRSTNME, LASTNAME, SALARY
  FROM EMPLOYEE AS CURRENT_ROW
  WHERE SALARY > (SELECT AVG(SALARY)
                    FROM EMPLOYEE
                    WHERE WORKDEPT = CURRENT_ROW.WORKDEPT)
  ORDER BY SALARY DESC;
```

*(Note that the correlated reference to WORKDEPT automatically eliminates nulls, allowing both tests for WORKDEPT IS NOT NULL to be removed.)*

4. ___ Verify your solution to question #3. Remove first name from the result set. Add work department and departmental average salary, rounded and changed to DECIMAL(9,2). Code *at least* two solutions, one involving two scalar fullselects, and another involving a join to a nested table expression.

```
SELECT EMPNO, LASTNAME, WORKDEPT, SALARY
     , DECIMAL( ROUND(
         (SELECT AVG(SALARY)
            FROM EMPLOYEE
            WHERE WORKDEPT = CURRENT_ROW.WORKDEPT)
       ,2) ,9,2) AS AVERAGE
  FROM EMPLOYEE AS CURRENT_ROW
  WHERE SALARY > (SELECT AVG(SALARY)
                    FROM EMPLOYEE
                    WHERE WORKDEPT = CURRENT_ROW.WORKDEPT)
  ORDER BY SALARY DESC;
SELECT EMPNO, LASTNAME, E.WORKDEPT, SALARY, AVERAGE
  FROM EMPLOYEE AS E
     , (SELECT WORKDEPT
             , DECIMAL( ROUND( AVG(SALARY) ,2) ,9,2)
               AS AVERAGE
          FROM EMPLOYEE
          GROUP BY WORKDEPT) AS X
  WHERE SALARY > AVERAGE
    AND E.WORKDEPT = X.WORKDEPT
  ORDER BY SALARY DESC;
```

5. ___ The following three SELECT statements return three different results. The first returns detail-level information, the second returns subtotals by department, and the third returns a grand-total row. Combine these three results into a single result. Hint: The result sets must be made compatible—the same number of columns, with positionally compatible data types.

```
SELECT WORKDEPT, EMPNO, SALARY
  FROM EMPLOYEE
  WHERE WORKDEPT IS NOT NULL
  ORDER BY WORKDEPT, EMPNO;
SELECT WORKDEPT, SUM(SALARY)
  FROM EMPLOYEE
  WHERE WORKDEPT IS NOT NULL
  GROUP BY WORKDEPT
  ORDER BY WORKDEPT;
SELECT SUM(SALARY)
  FROM EMPLOYEE
  WHERE WORKDEPT IS NOT NULL;
```

Following are the expected combined results. Note the sequence of the result rows as well as the nulls in the subtotal and grand-total rows. Make sure that the SALARY column is defined as DECIMAL(11,2).

```
WORKDEPT EMPNO  SALARY
-------- ------ -------------
A00      000010      52750.00
A00      000110      46500.00
A00      000120      29250.00
A00      -          128500.00
B01      000020      41250.00
B01      -           41250.00
C01      000030      38250.00
C01      000130      23800.00
C01      000140      28420.00
C01      -           90470.00
D11      000060      32250.00
D11      000150      25280.00
D11      000160      22250.00
D11      000170      24680.00
D11      000180      21340.00
D11      000190      20450.00
D11      000200      27740.00
D11      000210      18270.00
D11      000220      29840.00
D11      -          222100.00
D21      000070      36170.00
D21      000230      22180.00
D21      000240      28760.00
```

```
D21      000250      19180.00
D21      000260      17250.00
D21      000270      27380.00
D21      -          150920.00
E01      000050      40175.00
E01      -           40175.00
E11      000090      29750.00
E11      000280      26250.00
E11      000290      15340.00
E11      000300      17750.00
E11      000310      15900.00
E11      -          104990.00
E21      000100      26150.00
E21      000320      19950.00
E21      000330      25370.00
E21      000340      23840.00
E21      -           95310.00
-        -          873715.00
  41 record(s) selected.
```

```
SELECT WORKDEPT, EMPNO, SALARY
  FROM EMPLOYEE
  WHERE WORKDEPT IS NOT NULL
UNION ALL
SELECT WORKDEPT, NULLIF(' ',' ') AS EMPNO
     , DECIMAL(SUM(SALARY),11,2) AS SALARY
  FROM EMPLOYEE
  WHERE WORKDEPT IS NOT NULL
  GROUP BY WORKDEPT
UNION ALL
SELECT NULLIF(' ',' ') AS WORKDEPT
     , NULLIF(' ',' ') AS EMPNO
     , DECIMAL(SUM(SALARY),11,2) AS SALARY
  FROM EMPLOYEE
  WHERE WORKDEPT IS NOT NULL
ORDER BY WORKDEPT, EMPNO;
```