IBM Information Management software

Scalar Functions and Arithmetic

Unit Objectives

After completing this unit, you should be able to:

- Use arithmetic in the SELECT and WHERE clauses
- Use basic scalar functions such as COALESCE/VALUE, DECIMAL, SUBSTR
- Use date and time scalar functions
- Use the CONCAT operator

Selecting Calculated Values

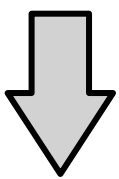
I need a list containing EMPNO, SALARY, COMM, and SALARY + COMM for employees whose salary is less than $20000, ordered by employee number

```
SELECT    EMPNO, SALARY, COMM,
          SALARY + COMM
FROM      EMPLOYEE
WHERE     SALARY < 20000
ORDER BY  EMPNO
```

EMPNO	SALARY	COMM	
000210	18270.00	1462.00	19732.00
000250	19180.00	1534.00	20714.00
000260	17250.00	1380.00	18630.00
000290	15340.00	1227.00	16567.00
000300	17750.00	1420.00	19170.00
000310	15900.00	1272.00	17172.00
000320	19950.00	1596.00	21546.00

Naming Result Columns

```
SELECT    EMPNO, SALARY, COMM, SALARY + COMM AS INCOME
FROM      EMPLOYEE
WHERE     SALARY < 20000
ORDER BY  EMPNO
```



EMPNO	SALARY	COMM	INCOME
000210	18270.00	1462.00	19732.00
000250	19180.00	1534.00	20714.00
000260	17250.00	1380.00	18630.00
000290	15340.00	1227.00	16567.00
000300	17750.00	1420.00	19170.00
000310	15900.00	1272.00	17172.00
000320	19950.00	1596.00	21546.00

Substitution of NULL Values

I need a listing containing department names and the employee number of its manager, sorted by department name.

```
SELECT   DEPTNAME
              , COALESCE (MGRNO, 'UNKNOWN')
                    AS MANAGER
FROM      DEPARTMENT
ORDER BY DEPTNAME
```

DEPTNAME	MANAGER
ADMINISTRATION SYSTEMS	000070
DEVELOPMENT CENTER	UNKNOWN
INFORMATION CENTER	000030
MANUFACTURING SYSTEMS	000060
OPERATIONS	000090
PLANNING	000020
SOFTWARE SUPPORT	000100
SPIFFY COMPUTER SERVICE DIV.	000010
SUPPORT SERVICES	000050

Arithmetic with NULL Values

I need a list of the total income (salary and commission). In the total, assume unknown commissions to be zero.

```
SELECT EMPNO, SALARY, COMM,
                SALARY + COMM
                AS "TOTAL INCOME"
FROM EMPLOYEE
```

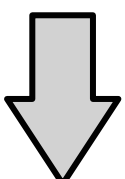
```
SELECT   EMPNO, SALARY, COMM,
                SALARY +
                COALESCE  (COMM, 0)
                AS "TOTAL INCOME"
FROM EMPLOYEE
```

EMPNO	SALARY	COMM	TOTAL INCOME
000210	18270.00	1462.00	19732.00
000260	17250.00	-	17250.00
000290	15340.00	1227.00	16567.00
000300	17750.00	-	17750.00
...	...	...	...

EMPNO	SALARY	COMM	TOTAL INCOME
000210	18270.00	1462.00	19732.00
000260	17250.00	-	-
000290	15340.00	1227.00	16567.00
000300	17750.00	-	-
...	...	...	...

Calculated Values

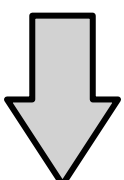
```
SELECT    EMPNO, SALARY, SALARY  * 1.0375
FROM      EMPLOYEE
WHERE     SALARY < 20000
ORDER BY  EMPNO
```



EMPNO	SALARY	
000210	18270.00	18955.125000
000250	19180.00	19899.250000
000260	17250.00	17896.875000
000290	15340.00	15915.250000
000300	17750.00	18415.625000
000310	15900.00	16496.250000
000320	19950.00	20698.125000

Decimal Representation of a Value

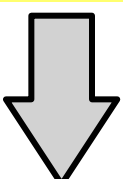
```
SELECT    EMPNO, SALARY,
          DECIMAL (SALARY * 1.0375, 8, 2)
FROM      EMPLOYEE
WHERE     SALARY < 20000
ORDER BY  EMPNO
```



EMPNO	SALARY	
000210	18270.00	18955.12
000250	19180.00	19899.25
000260	17250.00	17896.87
000290	15340.00	15915.25
000300	17750.00	18415.62
000310	15900.00	16496.25
000320	19950.00	20698.12

Decimal Values - Truncation and Rounding

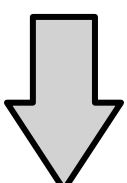
```
SELECT    EMPNO, SALARY,
          DECIMAL (SALARY * 1.0375 + 0.005, 8, 2)
FROM      EMPLOYEE
WHERE     SALARY < 20000
ORDER BY  EMPNO
```



EMPNO	SALARY	
000210	18270.00	18955.13
000250	19180.00	19899.25
000260	17250.00	17896.88
000290	15340.00	15915.25
000300	17750.00	18415.63
000310	15900.00	16496.25
000320	19950.00	20698.13

Condition on Calculated Values

```
SELECT      EMPNO, COMM, SALARY, (COMM/SALARY) * 100
FROM        EMPLOYEE
WHERE       (COMM/SALARY) * 100 > 8
ORDER BY    EMPNO
```



EMPNO	COMM	SALARY	
000140	2274.00	28420.00	8.001400
000210	1462.00	18270.00	8.002100
000240	2301.00	28760.00	8.000600
000330	2030.00	25370.00	8.001500

Date and Time

- DATE, TIME, TIMESTAMP data internally stored as packed decimal, without sign

DataType	Internal Format	Internal Length
DATE	yyyymmdd	4 bytes
TIME	hhmmss	3 bytes
TIMESTAMP	yyyymmddhhmmssnnnnnn	10 bytes

- Program uses an external format, that is,

Format	Time Format	Length	Date Format	Length
ISO	hh.mm.ss	8 bytes	yyyy-mm-dd	10 bytes
USA	hh:mm AM hh:mm PM	8 bytes 8 bytes	mm/dd/yyyy	10 bytes
EUR	hh.mm.ss		dd.mm.yyyy	10 bytes
JIS	hh:mm:ss	8 bytes	yyyy-mm-dd	10 bytes
LOCAL	???	???	???	???
TIMESTAMP DATA: yyyy-mm-dd-hh.mm.ss.nnnnnn				26 bytes

CHAR Function

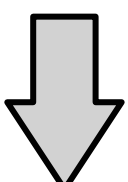
- **CHAR controls the external format of date / time data**

SELECT CHAR (TIMECOL, USA), CHAR (TIMECOL, ISO)...

03:30 PM 15.30.00

Comparison with Dates

```
SELECT      EMPNO, LASTNAME, BIRTHDATE
FROM        EMPLOYEE
WHERE       BIRTHDATE >= '1955-01-01'
ORDER BY    BIRTHDATE
```



EMPNO	LASTNAME	BIRTHDATE
000160	PIANKA	1955-04-12
000100	SPENCER	1956-12-18

DATE / TIME Arithmetic

- Subtraction only
  - time - time → time duration (decimal (6,0))
  - date – date → date duration (decimal (8,0))
  - timestamp – timestamp → timestamp duration (decimal (20,6))
- Labeled durations: YEARS, MONTHS, DAYS, HOURS, MINUTES, SECONDS, MICROSECONDS
  - time ± labeled duration → time
  - date ± labeled duration → date
  - timestamp ± duration → timestamp

Subtraction of Dates

I need a listing containing the ages of all employees 72 years old or more, sorted by age in descending sequence.

```
SELECT    EMPNO, LASTNAME, CURRENT_DATE -
          BIRTHDATE AS DIFFER
FROM      EMPLOYEE
WHERE     CURRENT_DATE - BIRTHDATE >= 720000
ORDER BY  DIFFER DESC
```

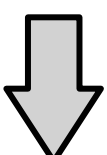
EMPNO	LASTNAME	DIFFER
000130	QUINTANA	790727.
000050	GEYER	790727.
000340	GOUNOT	781126.
000110	LUCCHESSI	750607.
000310	SETRIGHT	740021.
000320	MEHTA	720901.

Date / Time Scalar Functions

- DAY, MONTH, YEAR, HOUR, MINUTE, SECOND, MICROSECOND, DATE, TIME
  - Extract portions of a date, time, timestamp, or duration
- DAYS - Converts a date to the number of days since 12/31/0000
  - DAYS(date_1) – DAYS(date_2) gives the number of days between date_1 and date_2

Date Scalar Functions

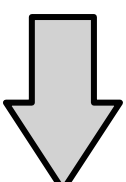
```
SELECT    LASTNAME, FIRSTNME,
          CURRENT_DATE - BIRTHDATE AS AGE,
          YEAR(CURRENT_DATE - BIRTHDATE) AS YEARS,
          MONTH(CURRENT_DATE - BIRTHDATE) AS MONTHS,
          DAY(CURRENT_DATE - BIRTHDATE) AS DAYS
FROM      EMPLOYEE
WHERE     YEAR(CURRENT_DATE - BIRTHDATE) >= 72
ORDER BY  AGE DESC, LASTNAME
```



LASTNAME	FIRSTNME	AGE	YEARS	MONTHS	DAYS
GEYER	JOHN	790727.	79	7	27
QUINTANA	DOLORES	790727.	79	7	27
GOUNOT	JASON	781126.	78	11	26
LUCCHESSI	VINCENZO	750607.	75	6	7
SETRIGHT	MAUDE	740021.	74	0	21
MEHTA	RAMLAL	720901.	72	9	1

DATE Arithmetic (1 of 2)

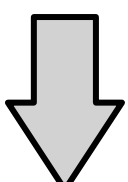
```
SELECT    PROJNO,
          DAYS(PRENDATE) - DAYS(PRSTDATE) AS DAYS
FROM      PROJECT
WHERE     DAYS(PRENDATE) - DAYS(PRSTDATE) <= 300
ORDER BY  DAYS
```



PROJNO	DAYS
PL2100	257
MA2113	289

DATE Arithmetic (2 of 2)

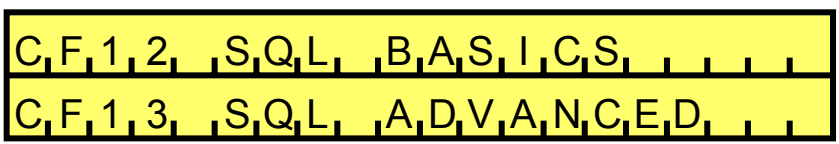
```
SELECT      PROJNO, PRENDATE,
            PRENDATE + 2 MONTHS + 15 DAYS
FROM        PROJECT
WHERE       PROJNO = 'AD3100'
ORDER BY    PROJNO
```



PROJNO	PRENDATE	
AD3100	1983-02-01	1983-04-16

Substring of Strings (1 of 2)

COURSINF

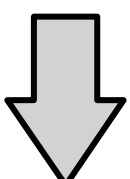



```
SELECT SUBSTR(COURSINF, 6, 15) ...
```

```
S Q L   B A S I C S
S Q L   A D V A N C E D
```

Substring of Strings (2 of 2)

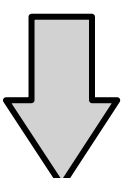
```
SELECT    SUBSTR(PROJNO,1, 2) AS PROJ_CLASS,
           PROJNAME
FROM      PROJECT
WHERE     PROJNO LIKE 'IF%'
```



PROJ_CLASS	PROJNAME
IF	QUERY SERVICES
IF	USER EDUCATION

Concatenation of Values

```
SELECT    LASTNAME CONCAT ', '
          CONCAT FIRSTNME AS NAME
FROM      EMPLOYEE
WHERE     WORKDEPT = 'A00'
ORDER BY  NAME
```



NAME

HAAS, CHRISTINE
LUCCHESSI, VINCENZO
O'CONNELL, SEAN

Checkpoint

1. True or False? If you use the DECIMAL scalar function to have two decimal positions instead of six, the result will be rounded.
2. If you subtract two dates, the format of the result will be:
   a. DD.MM.YYYY
   b. YYYY-MM-DD
   c. YYYYMMDD
3. Name several scalar functions.

Checkpoint Solutions

1. False
2. c
3. DECIMAL
   SUBSTR
   COALESCE
   VALUE
   YEAR
   MONTH
   DAY
   DAYS
   ROUND

Unit Summary

Having completed this unit, you should be able to:

- Use arithmetic in the SELECT and WHERE clauses
- Use basic scalar functions such as COALESCE/VALUE, DECIMAL, SUBSTR
- Use date and time scalar functions
- Use the CONCAT operator