IBM Software Group

Relational Databases & The Informix Detective Game

Instructions

IBM Information Management software

What is a Database?

- A database is a collection of data (values) stored in tables, where each table consists of columns (fields) and rows (records)
- We can compare a database to a filing cabinet:
  - **Database** = filing cabinet
  - **Table** = file folder within the cabinet
  - **Value** = data stored in each cell of a spreadsheet of rows (**records**) and columns (**fields**) within the file folder
- Both the filing cabinet and database provide you with a way to manage and organize data so that it can be quickly retrieved. However, because the database is automated it shortens the time required to search for data, sort data, add to the data, delete from the data, and edit the data
- With a **relational database**, such as IBM Informix, data is organized and accessed according to relationships defined between data items within the various tables. Data in the tables can be managed using the SQL (Structured Query Language) programming language

Example of a Database

- The Apple iPod uses a database to store all of your Music, Photos, etc.
- Database: Entire Music Folder
- Tables: Artist Folders (grouped by artist)
- Values: Album Names, Song Names, etc.

The History of Databases

- 1970 - IBM invents the database
  - E. F. Codd of IBM Research publishes a paper entitled “A Relational Model of Data for Large Shared Data Banks”, leading to a new way for computers to manage information
- 1974 - IBM invents the database programming language SEQUEL (or SQL for short)
  - Don Chamberlin and Ray Boyce publish "SEQUEL: A Structured English Query Language"
- 1970 to the present - IBM offers a complete family of relational database management systems (RDBMS) software
- Databases are used across all industries to manage everything from your credit card use, to bank accounts, to car insurance, to store purchases.

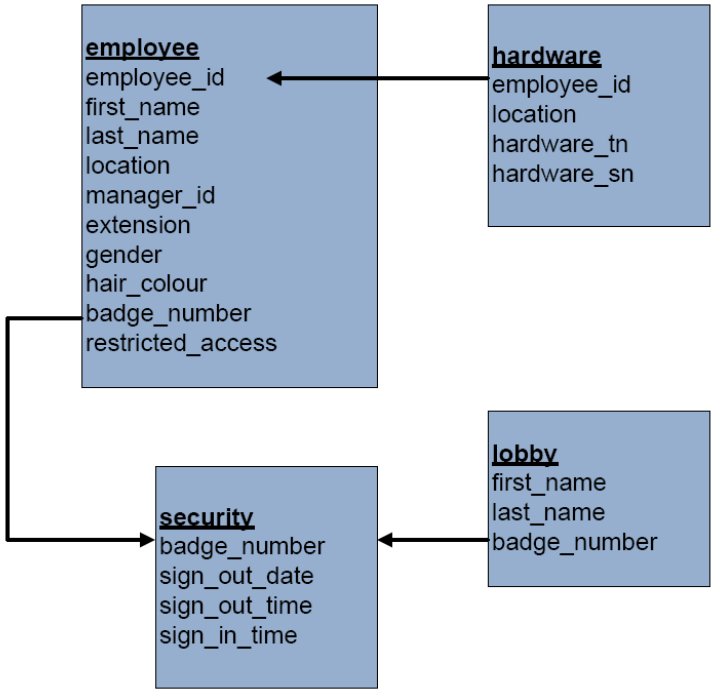
The Informix Detective Game

- Today we are going to learn about database technology by playing a fun and interactive game called “The Informix Detective Game”. This game will enable you to:
  - Understand how database data is stored and to gain an understanding of database concepts such as tables, rows (records), columns (fields), and values
  - Learn about relational databases and table joins
  - Gain a working knowledge of some of the primary SQL statements:
    - SELECT
    - UPDATE
    - INSERT
    - DELETE

Informix Detective Game

- The Informix Detective Game consists of four tables:

1. **employee** table stores information about each employee such as their employee #, name, manager, office #, phone #, badge #, etc.
2. **security** table stores information as to when each employee or visitor badged in and out of the building (date and time).
3. **lobby** table stores the badge # assigned to each visitor by name.
4. **hardware** table stores the serial # and type # for every computer along with its location and the ID of the employee who is its primary user.



Employee Table

```
DBAccess
DISPLAY:  Next  Restart  Exit
Display next page of results.

--------- informixgame@ol_informix1170 --------- Press CTRL-W for Help --------

employee_id first_name          last_name           location manager_id

534373      William             Miller              B3-268   278354
73732       Jagadish            Dara                C2-762   278354
897575      Oleg                Glembotsky          C2-123   825474
262745      Oleg                Glembotsky          A4-324   234445
629832      Colin               Humber              A2-462   278354
847254      Simone              Tufenkjian          A1-422   825474
252254      Fernando            Rocha               B3-252   234445
243354      Nelson              Icuma               B4-222
278354      Claudinei           Santana             B1-772   234445
893454      Vlaunir             Jimenez da Silva    C1-742   825474
893445      Juliana             Santos              C1-734   825474
234445      Sissi               Pereira             D1-734
823474      Claudia             Pelster             D2-736   278354
825474      Paulo Henrique      GuimarÑes           D3-536   243354
673574      Marco               Poma                D2-524   278354
```

- In the employee table, the columns (or fields) are: **employee_id, first_name, last_name, location, manager_id, extension, gender, hair_colour, badge_number,** and **restricted_access**

- There are 140 rows (records) in the **employee** table but only a subset of the results table is shown. You need to page down (using “Next” menu option) to see additional records.

Security Table

```
DBAccess
DISPLAY:  Next  Restart  Exit
Display next page of results.

--------- informixgame@ol_informix1170 --------- Press CTRL-W for Help --------

badge_number sign_out_date sign_out_time sign_in_time

34532        04/07/2003    17:14:00      09:22:00
24623        07/04/2003    18:23:00      10:03:00
24724        07/05/2003    16:53:00      08:47:00
97586        07/03/2003    17:43:00      11:33:00
23735        07/04/2003    19:00:00      03:05:00
67254        07/04/2003    16:00:00      13:05:00
43358        07/04/2003    15:45:00      11:05:00
42358        07/04/2003    17:25:00      08:45:00
43458        07/05/2003    16:34:00      09:25:00
43238        07/04/2003    12:54:00      06:55:00
87348        07/04/2003    17:54:00      07:55:00
92033        07/04/2003    18:23:00      08:35:00
23033        07/04/2003    18:23:00      08:35:00
23045        07/04/2003    18:23:00      08:35:00
76434        07/04/2003    18:23:00      08:35:00
```

- In the security table, the columns (or fields) are: **badge_number, sign_out_date, sign_out_time,** and **sign_in_time**
- There are 151 rows (records) in the security table but only a subset of the results table is shown on each page. You need to page down (using “Next” menu option) to see additional records.

Lobby Table

```
DBAccess
SQL:   New  Run  Modify  Use-editor  Output  Choose  Save  Info  Drop  Exit
Run the current SQL statements.

--------- informixgame@ol_informix1170 --------- Press CTRL-W for Help --------

first_name          last_name           badge_number

Tuan                Ong                 V0058
Shahreyar           Qadir               V0059
Peter               Wong                V0060
T.                  Kreysel             V0061
Srinivasan          Raju                V0062
Hong                Bao                 V0063
Zhichang            Liang               V0064
Frank               Bergemann           V0065
Eric                Herber              V0066
Friedemann          Schwenkreis         V0067

 10 row(s) retrieved.
```

- In the lobby table, the columns (or fields) are: **first_name, last_name,** and **badge_number**
- There are 10 rows (records) in the lobby table.

Hardware Table

```
DBAccess
DISPLAY:  Next  Restart  Exit
Display next page of results.

--------- informixgame@ol_informix1170 --------- Press CTRL-W for Help --------

employee_id location hardware_tn hardware_sn

534373      B3-268   7897-897    23-78234
73732       C2-762   7897-897    78-23672
897575      C2-123   3497-845    56-23622
262745      A4-324   3487-845    58-22352
629832      A2-462   3287-845    67-23452
847254      A1-422   3287-845    67-23232
252254      B3-252   5687-845    34-24232
243354      B4-222   5587-845    34-22332
278354      B1-772   2387-845    34-24332
893454      C1-742   2387-845    34-24331
893445      C1-734   2387-845    34-24336
234445      D1-734   2387-545    34-66336
823474      D2-736   2317-545    34-66246
825474      D3-536   2327-545    67-66246
673574      D2-524   2327-545    67-66566
```

- In the hardware table, the columns (or fields) are: **employee_id, location, hardware_tn,** and **hardware_sn**

- There are 140 rows (records) in the hardware table but only a subset of the results table is shown on each page. You need to page down (using “Next” menu option) to see additional records.

SQL Statements

- **The language of relational database technology is the Structured Query Language (SQL).** Invented by IBM in the 1970s, the SQL language continues to evolve and is the only way to access relational database data.
- This tutorial will introduce you to the following primary SQL statements:
  - **SELECT** - queries data from one or more tables
  - **UPDATE** - changes existing rows in a table
  - **INSERT** - adds data to a table
  - **DELETE** - removes rows from a table
- The following charts will describe each of these SQL statements so you will have all of the information needed to play the Informix Detective Game.

SELECT Statement

The SELECT statement is used to retrieve data. The format of this statement is:

SELECT column name(s)
FROM table name(s)
WHERE conditions for rows to meet (if any)

For example, to select all of the rows from the **lobby** table, issue the following command:

SELECT *
FROM lobby

This statement will produce the following result:

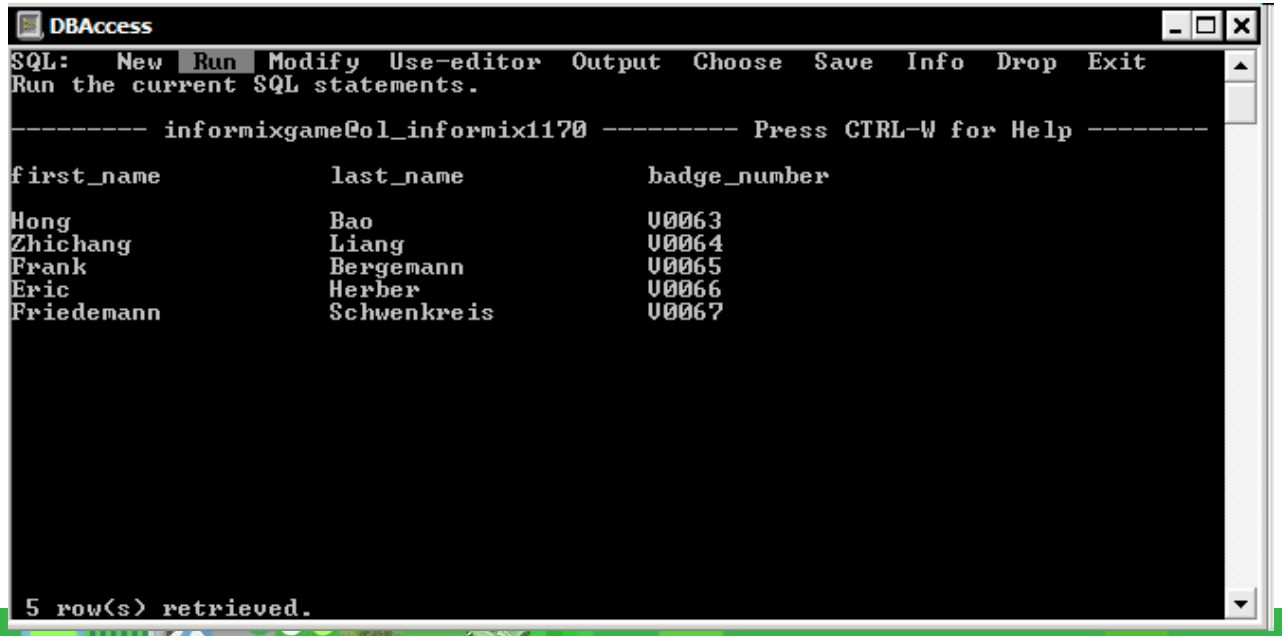
```
DBAccess
SQL:   New  Run  Modify  Use-editor  Output  Choose  Save  Info  Drop  Exit
Run the current SQL statements.

--------- informixgame@ol_informix1170 --------- Press CTRL-W for Help --------

first_name          last_name           badge_number

Tuan                Ong                 V0058
Shahreyar           Qadir               V0059
Peter               Wong                V0060
T.                  Kreysel             V0061
Srinivasan          Raju                V0062
Hong                Bao                 V0063
Zhichang            Liang               V0064
Frank               Bergemann           V0065
Eric                Herber              V0066
Friedemann          Schwenkreis         V0067

 10 row(s) retrieved.
```

SELECT Statement

To select all of the rows and columns from the **lobby** table where the visitor's badge number is greater than 62, issue the following statement:

```
SELECT *
FROM lobby
WHERE lobby.badge_number > 'V0062'
```

This statement will produce the following result:



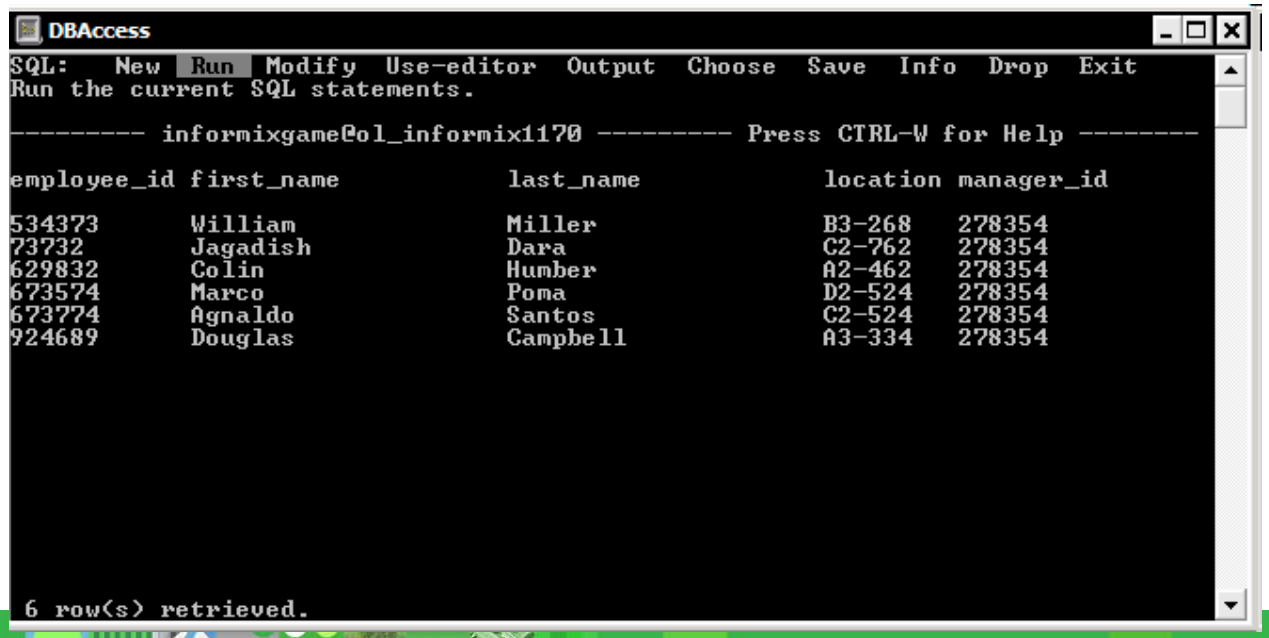
SELECT Statement

To select all of the rows and columns from the **employee** table where

a) the employee reports to Claudinei Santana (hint: employee id = 278354), and

b) the employee is male

```
SELECT *
FROM employee
WHERE employee.manager_id = '278354' AND employee.gender = 'M'
```



SELECT Statement

The SELECT statement will take any number of relational operators including:
=, >, <, >=, <=, <> (i.e. not equal to)

The SELECT statement can also take a number of predicates including:

- LIKE and NOT LIKE
- IS NULL and IS NOT NULL
- BETWEEN and NOT BETWEEN
- IN and NOT IN

In the Informix Detective Game, you will use the BETWEEN predicate.

BETWEEN Predicate

The BETWEEN predicate compares a single value to an inclusion range of values (i.e. all values BETWEEN a specified Maximum Value and Minimum Value).

The following example finds the badge numbers between 59 and 66:

```
SELECT *
FROM lobby
WHERE lobby.badge_number BETWEEN 'V0059' AND 'V0066'
```

This statement will produce the following result:

```
DBAccess
SQL:   New  Run  Modify  Use-editor  Output  Choose  Save  Info  Drop  Exit
Run the current SQL statements.

--------- informixgame@ol_informix1170 ---------- Press CTRL-W for Help --------

first_name          last_name           badge_number

Shahreyar           Qadir               V0059
Peter               Wong                V0060
T.                  Kreysel             V0061
Srinivasan          Raju                V0062
Hong                Bao                 V0063
Zhichang            Liang               V0064
Frank               Bergemann           V0065
Eric                Herber              V0066
```

UPDATE Statement

The UPDATE statement is used to change the data in a table.

With this statement, you can change the value of one or more columns for each row that satisfies the search condition of the WHERE clause. The format is:

UPDATE table name
SET column name = expression
WHERE conditions for rows to meet if any

Note: if you do not use the WHERE clause, all rows will be updated.

UPDATE Statement

For example, to change the first name of the visitor having badge number 58 to "Joan", issue the following statement:

```
UPDATE lobby
SET first_name = 'Joan'
WHERE lobby.badge_number = 'V0058'
```

The following is the result:

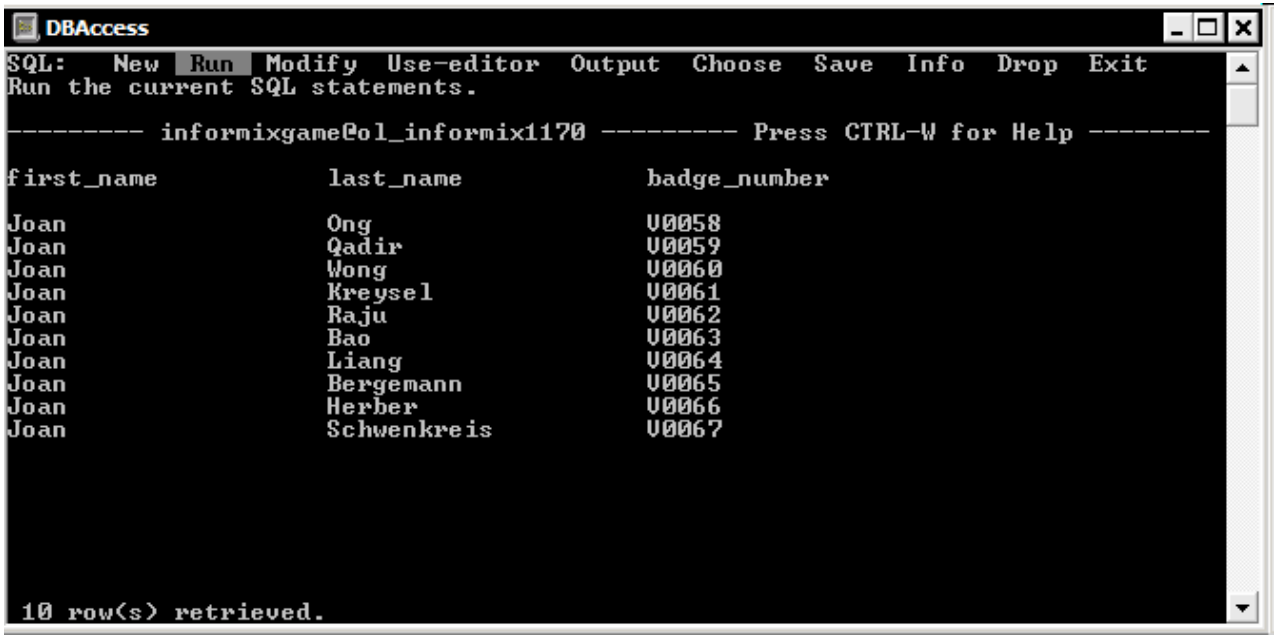
```
DBAccess
SQL:   New  Run  Modify  Use-editor  Output  Choose  Save  Info  Drop  Exit
Run the current SQL statements.

--------- informixgame@ol_informix1170 --------- Press CTRL-W for Help --------

first_name           last_name            badge_number

Joan                 Ong                  V0058
Shahreyar            Qadir                V0059
Peter                Wong                 V0060
T.                   Kreysel              V0061
Srinivasan           Raju                 V0062
Hong                 Bao                  V0063
Zhichang             Liang                V0064
Frank                Bergemann            V0065
Eric                 Herber               V0066
Friedemann           Schwenkreis          V0067

 10 row(s) retrieved.
```

UPDATE Statement

If you had omitted the WHERE clause on the previous SQL command and issued the following command instead, all records would have been updated:

```
UPDATE lobby
SET first_name = 'Joan'
```

The following is the result:



INSERT Statement

The INSERT statement is used to add data to a table. The format of this statement is:

INSERT INTO tablename (column name(s))
VALUES (value(s));

To add a visitor named “Informix Detective" to the **lobby** table, issue this command:

INSERT INTO lobby (first_name, last_name, badge_number)
VALUES (‘Informix', 'Detective', 'V0062')

This statement will produce the following result:

```
DBAccess
SQL:   New  Run  Modify  Use-editor  Output  Choose  Sa
Run the current SQL statements.

--------- informixgame@ol_informix1170 --------- Press

first_name          last_name           badge_number

Tuan                Ong                 V0058
Shahreyar           Qadir               V0059
Peter               Wong                V0060
T.                  Kreysel             V0061
Srinivasan          Raju                V0062
Hong                Bao                 V0063
Zhichang            Liang               V0064
Frank               Bergemann           V0065
Eric                Herber              V0066
Friedemann          Schwenkreis         V0067
Informix            Detective           V0062
```

Note: This insertion is the 11th record of the lobby table and that the table now has two records with badge_number = ‘V0062’

DELETE Statement

Use the DELETE statement to remove records (rows) from a table. The format is:

DELETE FROM table name
WHERE conditions for rows to meet if any

For example, to remove the record for the visitor with a last name of Detective from the **lobby** table, issue the following statement:

DELETE FROM lobby
WHERE lobby.last_name = ‘Detective'

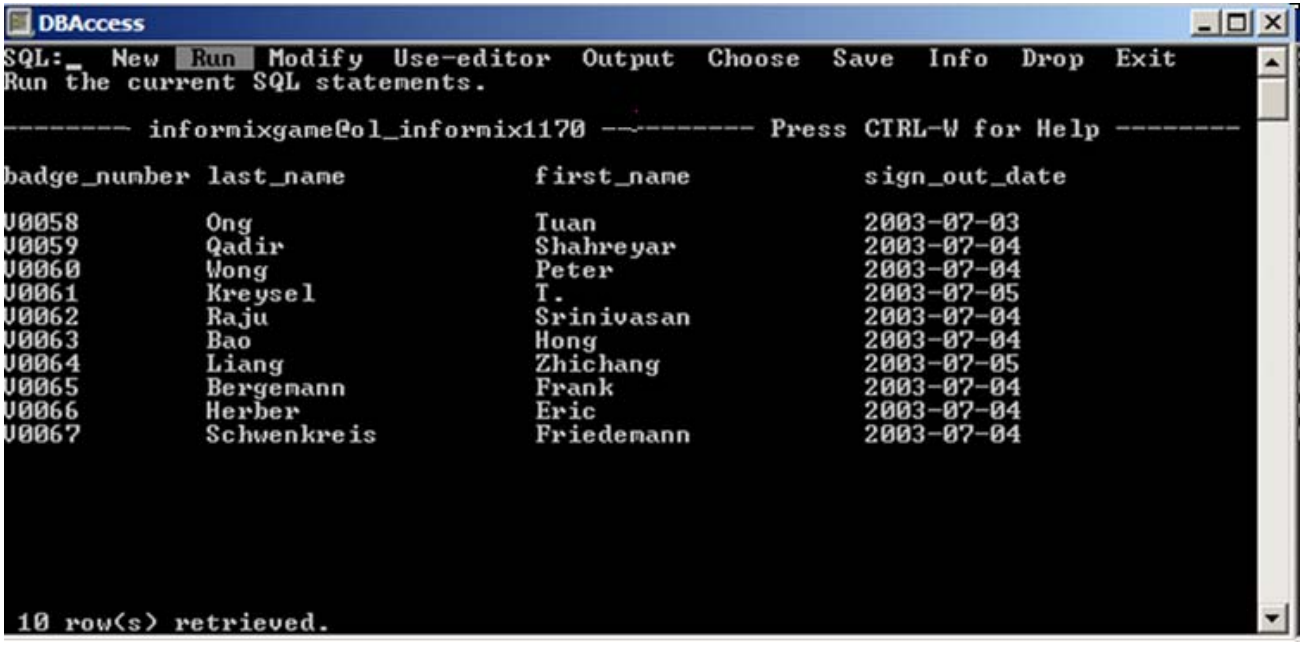
```
DBAccess
SQL:   New  Run  Modify  Use-editor  Output  Choose  Save  Info  Drop  Exit
Run the current SQL statements.

--------- informixgame@ol_informix1170 --------- Press CTRL-W for Help --------

first_name          last_name           badge_number

Tuan                Ong                 V0058
Shahreyar           Qadir               V0059
Peter               Wong                V0060
T.                  Kreysel             V0061
Srinivasan          Raju                V0062
Hong                Bao                 V0063
Zhichang            Liang               V0064
Frank               Bergemann           V0065
Eric                Herber              V0066
Friedemann          Schwenkreis         V0067

 10 row(s) retrieved.
```

Note: Following this deletion there are only 10 records in the lobby table and only one record with **badge_number = ‘V0062’**

Joining tables

The process of combining data from two or more tables is called joining tables.
The columns involved in the join condition do not have to be identical; however, they must be compatible.
To join the **lobby** table to the **security** table, issue the following command:

```
SELECT *
FROM lobby, security
WHERE security.badge_number = lobby.badge_number
```



Note: The fields shown are from both the lobby and security tables.

Recall: The Security table has 151 rows but only 10 that satisfy the join condition.

Tips

- Use DBAccess to execute SQL statements
- Enter SQL commands in the “Query-language” tab from the DBAccess main menu
- From the Query-language menu:
  - “New” starts a new SQL statement,
  - “Run” executes the SQL statement,
  - “Modify” modifies your last SQL statement
- When viewing your results, press “Next” to scroll through the entire list of results.
- To see the count of the number of records for an SQL statement, select Next until the number of rows retrieved is displayed.
- If the width of the fields to be displayed is less than 80 characters wide (including the field name), the results are displayed in a table format.
- If the width of the fields to be displayed is greater than 80 characters wide (including the field name), the results for each record is displayed over multiple lines, as defined by the table structure.

Additional Tips

- Note the format of data in the tables
  - gender field contains a single uppercase letter (M or F)
  - hair_colour field contains lowercase entries only (brown, black, blonde, or red)
  - restricted_access field contains a single uppercase letter (Y or N)
  - sign_out_time and sign_in_time fields use a 24-hour format (e.g., 14:00:00)
  - sign_out_date field uses year-month–day format (e.g., 2007-11-07)

    (Note: use the SET DBDATE=y4md- command prior to starting DBAccess)

- The * in place of the column name(s) of the SQL query is equal to identifying all column names

```
SELECT *
FROM lobby
```

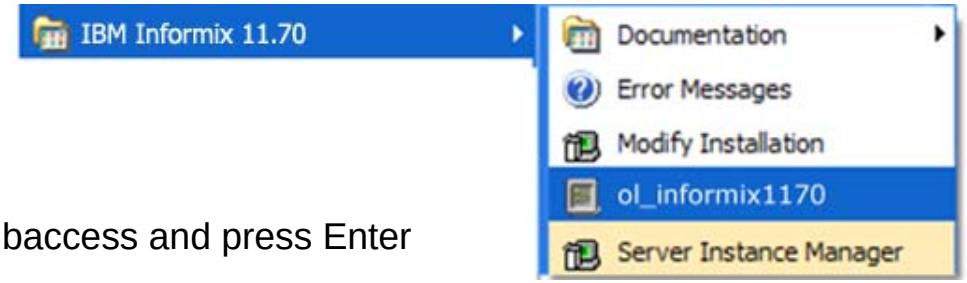
  The above query selects all columns from the lobby table.

- AND is used to create compound conditions in a SELECT statement

```
SELECT *
FROM employee
WHERE employee.manager_id = '278354'
AND employee.gender = 'M'
```

Before you start playing the game

1. Start Informix
   a) Start Informix command prompt
      - Start → **all Programs → IBM Informix Dynamic Server 11.70 → ol_informix1170**



   b) Start DBAccess
      - At the command prompt, type in dbaccess and press Enter

```
C:\Program Files\IBM\IBM Informix\11.70>dbaccess
```

2. Start Informix Detective clue program
   - Click on InformixGame shortcut on your windows desktop

3. Select “Query-language” from the DBAccess main menu

```
DBACCESS:_ Query-language  Connection  Database  Table  Session  Exit
Use SQL query language.

------------------------------------------------ Press CTRL-W for Help --------
```

Before you start playing the game

4. Select “New” from the Query-language menu and press Enter

```
SQL:   New  Run  Modify  Use-editor  Output  Choose  Save  Info  Drop  Exit
Enter new SQL statements using SQL editor.

--------- informixgame@ol_informix1170 --------- Press CTRL-W for Help --------
```

5. Try a test command → SELECT * FROM lobby
(hit ESC; Run)

```
NEW:    ESC     = Done editing       CTRL-A = Typeover/Insert     CTRL-R = Redraw
        CTRL-X  = Delete character   CTRL-D = Delete rest of line

--------- informixgame@ol_informix1170 --------- Press CTRL-W for Help --------

select * from lobby
```

6. You are now ready to play the game!

Your Mission…

- Now that you have an understanding of some relational database concepts and SQL commands, your mission is to use database technology to solve the case of the missing ThinkPad. Good luck!!!

Case Solved!