SQL AND ORACLE

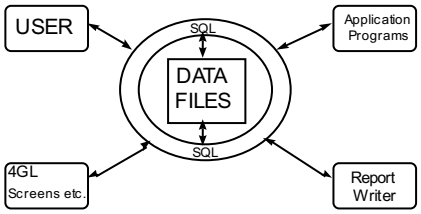
1

SIT103 Lecture 2

STRUCTURED QUERY LANGUAGE

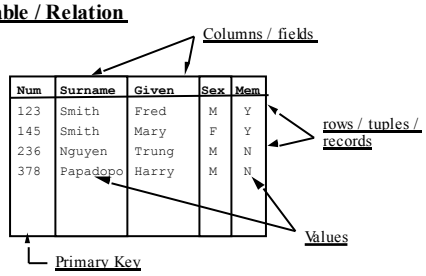
- A complete interface language for a database system.
  - *Data Definition*
    ie. Creating & Defining files, fields etc.
  - *Data Manipulation*
    ie. Inserting, Updating & Deleting records
  - *Data Extraction*
    ie. Getting data from the database (queries).

STRUCTURE OF SQL DATABASE INTERFACE



RELATIONAL DATABASES & SQL

Table / Relation



RELATIONAL / SQL DATABASE RULES

- No ordering or table joining information is stored.
- All access to the database is through SQL. Both for users and programs!
- Multiple table operations involve joining tables on fly.

SQL TABLES - STUDENT & PROGRAMME

STUDENT

Name	Null?	Type
STUDENT_NO	NOT NULL	CHAR(8)
SURNAME		VARCHAR2(30)
GIVEN		VARCHAR2(30)
DOB		DATE
SEX		CHAR(1)
ADDR1		VARCHAR2(40)
ADDR2		VARCHAR2(40)
ADDR3		VARCHAR2(40)
POSTCODE		NUMERIC(4)
PHONE		VARCHAR2(20)
PROGRAMME_CODE		VARCHAR2(6)
PROGRAMME_LOAD		VARCHAR2(1)

PROGRAM ME

Name	Null?	Type
PROGRAMME_CODE	NOT NULL	VARCHAR2(6)
PROGRAMME_NAME		VARCHAR2(50)
DEPT_NO		NUMERIC

SQL TABLE - COURSE & STUD_COURSE

COURSE

Name	Null?	Type
COURSE_CODE	NOT NULL	CHAR(5)
COURSE_NAME		VARCHAR2(50)
DEPT_NO		NUMERIC
FEE		NUMERIC(10,2)

STUDENT_COURSE

Name	Null?	Type
STUDENT_NO	NOT NULL	CHAR(8)
COURSE_CODE	NOT NULL	CHAR(5)
COURSE_PROGRAMME		VARCHAR2(6)
SEM_START		NUMERIC(1)
SEM_DURATION		NUMERIC(1)
RESULT		VARCHAR2(4)

SOME SIMPLE QUERIES

SELECT SURNAME, GIVEN
FROM STUDENT;

SURNAME	GIVEN
UNCHERNRAT	ROSEMARY ANN
SCRIVENER	ROBERT CRAWFORD R
BULSOOK	JORGE

SELECT SURNAME, GIVEN, PROGRAMME_CODE
FROM STUDENT;

SURNAME	GIVEN	PROGRAMME_CODE
UNCHERNRAT	ROSEMARY ANN	167001
SCRIVENER	ROBERT CRAWFORD	162010
BULSOOK	JORGE	165701
FERONJA	TRUC THUNG	165000

SOME SIMPLE QUERIES

SELECT SEX
FROM STUDENT;

SEX
M
M
F
M
F
F

SELECT DISTINCT SEX
FROM STUDENT;

SEX
F
M

SELECT *
FROM STUDENT;

Select All fields, from all tables in FROM clause.

CONCANTENATION QUERIES

SELECT SURNAME | | ', ' | | GIVEN
FROM STUDENT;

```
SURNAME||',   '||GIVEN
------------------------------
RAJOO,  THINH  MINH
PHONGWATCHARARUK,  JERZY  MIROSLAW
CHOW,  SOW LIM
```

SELECT 'Student: ' | | SURNAME, 'Programme: ' | |
PROGRAMME_CODE
FROM STUDENT;

```
'STUDENT:'||SURNAME              'PROGRAMME:'||PRO
-------------------------------  ----------------
Student:  RAJOO                  Programme:  167402
Student:  PHONGWATCHARARUK       Programme:  161005
Student:  CHOW                   Programme:  165000
```

- Note: Concatenation can be with fields and/or string literals.

SIMPLE SORTING QUERIES

- Ascending order

SELECT SURNAME, GIVEN
FROM STUDENT
ORDER BY SURNAME;

```
SURNAME          GIVEN
---------------  ---------------------
ABELA            VENUS
ABERTON          SARAH  JANE
ALBERT           ROSLYN  ISABEL
ALLEN            BARTHOLOMEUS  DIAZ
ALLURI           ZENO
```

SIMPLE SORTING QUERIES

- Descending order

SELECT SURNAME, GIVEN, PROGRAMME_CODE
FROM STUDENT
ORDER BY PROGRAMME_CODE DESC;

```
SURNAME          GIVEN                PROGRA
---------------  -------------------  ------
MARTIN           CLARE  ALLISON       167730
RAJOO            THINH  MINH          167402
LACHASAREEKUL    GIN  HOCK            167401
ALBERT           ROSLYN  ISABEL       167400
THOMAS           CLAIRE  JOY          167331
```

Multiple Sort Fields

```
SELECT SURNAME, GIVEN
FROM STUDENT
ORDER BY SURNAME, GIVEN;
```

SURNAME	GIVEN
ABELA	VENUS
ABERTON	SARAH JANE
ABERTON	ZENO
ALLEN	BARTHOLOMEU S
ALLEN	PETER
ALLEN	STEPHEN GRAHAM

- Rows are initially sorted by the first sort field.
- Those rows with the same value in the first sort field are then sorted between themselves using the second sort field.

Change Column Headings

```
SELECT SURNAME AS "Last Name", DOB AS "Date Birth"
FROM STUDENT;
```

Last Name	Date Birth
RAJOO	01-JUL-73
PHONGWATCHARARUK	01-JAN-73
CHOW	02-DEC-74

```
SELECT SURNAME || ', ' || GIVEN AS "Full Name"
FROM STUDENT
ORDER BY "Full Name";
```

```
SELECT SURNAME || ', ' || GIVEN AS "Full Name", SEX AS
"Gender", DOB
FROM STUDENT
ORDER BY 2;
```

SELECT … WHERE

- In order to select specific records from the resulting output of a SELECT clause you can do this with WHERE.

```
SELECT SURNAME, GIVEN, SEX, PROGRAMME_CODE, DOB
FROM STUDENT
WHERE PROGRAMME_CODE = 165000;
```

SURNAME	GIVEN	S	PROGRA	DOB
CHOW	SOW LIM	F	165000	02-DEC-74
ALLURI	ZENO	M	165000	24-SEP-74
DWYER	ANDREW ANTHONY	M	165000	08-OCT-74

SELECTION CRITERIA

- All the usual comparison operators.
  ie. = < > <= >= <>
- Usual compound operators.
  ie. AND OR and NOT
- Brackets can be used:
  NOT(((cond1) AND (cond2)) OR cond3)

SELECT … WHERE

```
SELECT SURNAME, GIVEN, SEX, PROGRAMME_CODE,
DOB
FROM STUDENT
WHERE SEX = 'M' AND POSTCODE = 3000;
```

SURNAME	GIVEN	S	PROGRA	DOB
TSUNG	KOK WAH	M	166701	06-JUN-72

```
SELECT SURNAME, GIVEN, SEX, PROGRAMME_CODE,
DOB
FROM STUDENT
WHERE NOT(SEX = 'M');
```

SURNAME	GIVEN	S	PROGRA	DOB
CHOW	SOW LIM	F	165000	02-DEC-74
SAMARAWICKRAMA	JANE	F	162010	06-FEB-70
GILTRAP	ELIZABETH JANE	F	164002	15-SEP-72

WILDCARDS

```
SELECT SURNAME
FROM STUDENT
WHERE SURNAME LIKE 'S%';
```

SURNAME
SMITH
SANDERS
SOMERS

```
SELECT SURNAME
FROM STUDENT
WHERE SURNAME LIKE '%TH';
```

SURNAME
COATH
SMITH
SMITH

```
SELECT SURNAME
FROM STUDENT
WHERE SURNAME LIKE '%ON%';
```

SURNAME
PHONGWATCHARAKUL
WHEATON
ONG

Selection Criteria Shortcuts

```
SELECT SURNAME, POSTCODE
FROM STUDENT
WHERE POSTCODE BETWEEN 3000 AND 4000;

SELECT SURNAME
FROM STUDENT
WHERE SURNAME IN ('JONES', 'SMITH', 'ADAMS');

SELECT SURNAME, PROGRAMME_CODE
FROM STUDENT
WHERE PROGRAMME_CODE IN (165000, 161000, 164000);
```

SURNAME	PROGRA
CHOW	165000
ALLURI	165000
DWYER	164000

NULL operator

- A field is NULL if it is empty.
- NULL is different to an empty string!
- NOT version also possible

```
SELECT SURNAME, GIVEN
FROM STUDENT
WHERE PROGRAMME_CODE IS NULL;
```

Query Calculations

- Queries can use arithmetic calculations and built in functions.

```
SELECT 'Fee $AUD' || FEE, 'Fee $US' || FEE * 2.2
FROM COURSE;

SELECT INVOICEID, PRODNAME, QTY * UNITPRICE
FROM SALE;

SELECT SURNAME, ((SYSDATE-DOB)/365.25)
FROM STUDENT;
```

SURNAME	((SYSDATE-DOB)/365.25)
RAJOO	28.426404
PHONGRATCHARARUK	28.921955
CHOW	27.0054594

SYSTEM VARIABLES

- System variables can be used in place of field names.
- SYSDATE - Current System Date

```
SELECT SURNAME, FLOOR((SYSDATE-DOB)/365)
FROM STUDENT;
```

SURNAME	FLOOR((SYSDATE-DOB)/365)
RAJOO	28
PHONGWATCHARARUK	28
CHOW	27