The Relational Language SQL

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

- Originally called SEQUEL (Structured English Query Language) and pronounced "see-quail."
- Later, IBM changed it to be SQL (Structured Query Language)
- Become the standard relational database language
  - ANSI (American National Standard Institute) in 1986
  - ISO (International Organization Standardization) in 1987

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

- **SQL**
  - Direct descendent of the Tuple Relational Calculus language.
  - It says "What?" not "How?"
  - Not only the "Query" part, but also "Insert", "Delete", and "Update" data manipulation.
  - Case insensitive.

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

- **Relational Database**
  - Represented relation with **tabular format**.
  - Values in each column must belongs to only one domain (topic).
  - Each column value must be atomic.
  - No two rows are the same.
  - Rows have no order.

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

- SQL commands can be categorized into four types
  1. Data Definition Language (**DDL**)
  2. Data Manipulation Language (**DML**)
  3. Data Control Language (**DCL**)
  4. Transaction Control Language (**TCL**)

  - Some defines Data Retrieval Language (**DRL**) for the retrieval function (`SELECT statement`) since the word "manipulation" often confuses with "modification."

pakorn.wa@kmitl.ac.th

- Data Definition Language (DDL)
  - **CREATE**
  - **ALTER**
    - Allow existing database structures to be modified.
    - New column, change data type, etc.
    - Use with caution since it affects more than one application in the enterprise.
  - **DROP**
    - Use to drop tables, views, or indexes.
    - Ensure the return of the occupied spaces of the object to the public area.

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

```
CREATE TABLE "VENDOR"
(    "V#"        VARCHAR2(4),
     "VNAME"     VARCHAR2(15),
     "LOCATION"  VARCHAR2(15),
     "STATUS"    NUMBER,
     CONSTRAINT  "VENDOR_CON"        UNIQUE ("V#") ENABLE
)

CREATE TABLE "ITEM"
(    "I#"        VARCHAR2(4),
     "INAME"     VARCHAR2(15),
     "COLOR"     VARCHAR2(15),
     CONSTRAINT  "ITEM_CON" UNIQUE ("I#") ENABLE
)

CREATE TABLE "SALE"
(    "V#"        VARCHAR2(4),
     "I#"        VARCHAR2(4),
     "AMOUNT"    NUMBER,
     CONSTRAINT  "SALE_CON" UNIQUE ("V#,I#") ENABLE
)
```

A script for ORACLE®

```
CREATE TABLE VENDOR
(    V#          VARCHAR(4),
     VNAME       VARCHAR(15),
     LOCATION    VARCHAR(15),
     STATUS      SMALLINT,
     CONSTRAINT  VENDOR_CON UNIQUE (V#)
)

CREATE TABLE ITEM
(    I#          VARCHAR(4),
     INAME       VARCHAR(15),
     COLOR       VARCHAR(15),
     CONSTRAINT  ITEM_CON   UNIQUE (I#)
)

CREATE TABLE SALE
(    V#          VARCHAR(4),
     I#          VARCHAR(4),
     AMOUNT      SMALLINT,
     CONSTRAINT  SALE_CON   UNIQUE (V#,I#)
)
```

A script for MySQL

pakorn.wa@kmitl.ac.th

- Data Manipulation Language (DML)
  - **INSERT**
    - Create a new row and add them to a specified table.
  - **DELETE**
    - Remove the entire row(s) from the table.
    - Unqualified deletion means all rows in the table are deleted.
  - **UPDATE**
    - Modifies column values of existing rows.
    - Can be used on "views." The command works on the column(s) that the programmer is allowed to see.
  - **SELECT**
    - Retrieve results with 100% correct and complete precision according to the given search condition.

pakorn.wa@kmitl.ac.th

- Data Manipulation Language (DML) Cont'd.
  - `INSERT` and `DELETE`
    - Both can be used with the "views," but they must be used carefully.
    - `INSERT` operations on views can be done, and rows are inserted even though the values in the rows are not according to the view definition unless the `CHECK OPTION` declared.
    - `DELETE` operations needs even more attention!
      - The `DELETE` command removes the entire row(s).
      - In the case that a `DELETE` statement is issued by a user who is not allowed to see the entire row of a table, his `DELETE` command will remove the entire row, which includes the information in the columns that he is not allowed to see.
      - In principle, one should not be allowed to delete through view unless he is also allowed to see the entire table.

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

- Data Control Language (DCL)
  GRANT and REVOKE for controlling over access permission and privileges.

GRANT	SELECT	reads only
	UPDATE	modifies existing rows
	DELETE	deletes rows
	INSERT	inserts rows
	ALTER	alters database structure definitions

E.g, GRANT SELECT, INSERT, DELETE on SALE to David;
REVOKE UPDATE on ITEM from John;

- Transaction Control Language (TCL)
  - COMMIT confirms operations in the transaction has successfully completed. All operations started from the previous sync point to the commit point are all confirmed completed. This is a "logical" confirmation.
  - ROLLBACK cancels all the operations from the previous sync point to the ROLLBACK point.
  - SET AUTOCOMMIT OFF is a sync point which is equivalent to "begin transaction."

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

The SQL SELECT Statements

A simplified SQL SELECT statement structure is shown as follows:

```
  SELECT      <column name list>  |*|  <build-in functions>
  FROM        <table names> | <view names>
[ WHERE       <row conditions>  ]
[ GROUP BY    <column name list>  ]
[ HAVING      <group condition>  ]
[ ORDER BY    <column name list> | <column position list> [ASC | DESC]  ]
```

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

For example,

```
SELECT      DEPARTMENT, COUNT(*)
FROM        STUDENT
WHERE       FACULTY = 'Engineering'
GROUP BY    DEPARTMENT
HAVING      COUNT(*) > 400
ORDER BY    2 DESC;
```

Presidential Table Template

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

PRESIDENT

PRES_NAME	BIRTH_YR	YRS_SERVE	DEATH_AGE	PARTY	STATE_BORN

STATE

STATE_NAME	ADMIN_ENTERED	YEAR_ENTERED

PRES-HOBBY

PRES_NAME	HOBBY

ADMINISTRATION

ADMIN_NR	PRES_NAME	YEAR_INAUGURATED

ADMIN-PR-VP

ADMIN_NR	PRES_NAME	VICE_PRES_NAME

PRES-MARRIAGE

PRES_NAME	SPOUSE_NAME	PR_AGE	SP_AGE	NA_CHILDREN	MAR_YEAR

ELECTION

ELECTION_YEAR	CANDIDATE	VOTES	WINNER_LOSER_INDIC

SELECT without conditions

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

- List the entire rows from the RECENT_PRESIDENTS “table”

```
CREATE VIEW RECENT_PRESIDENTS as
SELECT *
FROM PRESIDENT
WHERE BIRTH_YR > 1880;
```

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

List the entire rows from the RECENT_PRESIDENTS "table."

```
SELECT     *
FROM RECENT_PRESIDENTS;
```

PRES_NAME	BIRTH_YR	YRS_SERV	DEATH_AGE	PARTY	STATE_BORN
Roosevelt F D	1882	12	63	Democratic	Texas
Truman H S	1884	7	88	Democratic	Massachusetts
Eisenhower D D	1890	8	79	Republican	Texas
Kennedy J F	1917	2	46	Democratic	California
Johnson L B	1908	5	65	Democratic	Texas
Nixon R M	1913	5	NULL	Republican	California
Ford G R	1913	2	NULL	Republican	Nebraska
Carter J E	1924	4	NULL	Democratic	Georgia
Reagan R	1911	3	NULL	Republican	Illinois

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

If the sequence of the columns is required, column names can be explicitly listed.

```
SELECT PRES_NAME, PARTY, YRS_SERV, STATE_BORN, BIRTH_YR, DEATH_AGE
FROM recent_presidents
```

PRES_NAME	PARTY	YRS_SERV	STATE_BORN	BIRTH_YR	DEATH_AGE
Roosevelt F D	Democratic	12	Texas	1882	63
Truman H S	Democratic	7	Massachusetts	1884	88
Eisenhower D D	Republican	8	Texas	1890	79
Kennedy J F	Democratic	2	California	1917	46
Johnson L B	Democratic	5	Texas	1908	65
Nixon R M	Republican	5	California	1913	NULL
Ford G R	Republican	2	Nebraska	1913	NULL
Carter J E	Democratic	4	Georgia	1924	NULL
Reagan R	Republican	3	Illinois	1911	NULL

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

If the sorted result is required, the ORDER BY clause can be applied.

```
SELECT PRES_NAME, PARTY, YRS_SERV, STATE_BORN, BIRTH_YR, DEATH_AGE
FROM recent_presidents
ORDER BY YRS_SERV DESC
```

PRES_NAME	PARTY	YRS_SERV	STATE_BORN	BIRTH_YR	DEATH_AGE
Roosevelt F D	Democratic	12	Texas	1882	63
Eisenhower D D	Republican	8	Texas	1890	79
Truman H S	Democratic	7	Massachusetts	1884	88
Johnson L B	Democratic	5	Texas	1908	65
Nixon R M	Republican	5	California	1913	NULL
Carter J E	Democratic	4	Georgia	1924	NULL
Reagan R	Republican	3	Illinois	1911	NULL
Kennedy J F	Democratic	2	California	1917	46
Ford G R	Republican	2	Nebraska	1913	NULL

```
SELECT PRES_NAME, PARTY, YRS_SERV, STATE_BORN, BIRTH_YR, DEATH_AGE
FROM recent_presidents
ORDER BY 3 DESC
```

PRES_NAME	PARTY	YRS_SERV	STATE_BORN	BIRTH_YR	DEATH_AGE
Roosevelt F D	Democratic	12	Texas	1882	63
Eisenhower D D	Republican	8	Texas	1890	79
Truman H S	Democratic	7	Massachusetts	1884	88
Johnson L B	Democratic	5	Texas	1908	65
Nixon R M	Republican	5	California	1913	NULL
Carter J E	Democratic	4	Georgia	1924	NULL
Reagan R	Republican	3	Illinois	1911	NULL
Kennedy J F	Democratic	2	California	1917	46
Ford G R	Republican	2	Nebraska	1913	NULL

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

List party and state born of presidents

```
SELECT PARTY, STATE_BORN
FROM recent_presidents
ORDER BY STATE_BORN
```

PARTY	STATE_BORN
Democratic	California
Republican	California
Democratic	Georgia
Republican	Illinois
Democratic	Massachusetts
Republican	Nebraska
Democratic	Texas
Republican	Texas
Democratic	Texas

```
SELECT DISTINCT PARTY, STATE_BORN
FROM recent_presidents
ORDER BY STATE_BORN
```

PARTY	STATE_BORN
Democratic	California
Republican	California
Democratic	Georgia
Republican	Illinois
Democratic	Massachusetts
Republican	Nebraska
Democratic	Texas
Republican	Texas

SELECT with Row Condition

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

List rows of President Washington

```
SELECT    *
FROM      `president`
WHERE     PRES_NAME = 'Washington G'
```

Washington G	1732	7	67	Federalist	Virginia

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

List rows of presidents who served more than 5 years as presidents

SELECT * **FROM** `president` **WHERE** YRS_SERV>5

PRES_NAME	BIRTH_YR	YRS_SERV	DEATH_AGE	PARTY	STATE_BORN
Washington G	1732	7	67	Federalist	Virginia
Jefferson T	1743	8	83	Demo-Rep	Virginia
Madison J	1751	8	85	Demo-Rep	Virginia
Monroe J	1758	8	73	Demo-Rep	Virginia
Jackson A	1767	8	78	Democratic	South Carolina
Grant U S	1822	8	63	Republican	Ohio
Cleveland G	1837	8	71	Democratic	New Jersey
Roosevelt T	1858	7	60	Republican	New York
Wilson W	1856	8	67	Democratic	Vermont
Roosevelt F D	1882	12	63	Democratic	Texas
Truman H S	1884	7	88	Democratic	Massachusetts
Eisenhower D D	1890	8	79	Republican	Texas

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

List rows of Democratic presidents who served more than 5 years as presidents

```
SELECT * FROM `president` WHERE YRS_SERV>5 AND PARTY='Democratic'
```

PRES_NAME	BIRTH_YR	YRS_SERV	DEATH_AGE	PARTY	STATE_BORN
Jackson A	1767	8	78	Democratic	South Carolina
Cleveland G	1837	8	71	Democratic	New Jersey
Wilson W	1856	8	67	Democratic	Vermont
Roosevelt F D	1882	12	63	Democratic	Texas
Truman H S	1884	7	88	Democratic	Massachusetts

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

List rows of Democratic and Republican presidents

✗ `SELECT * FROM `president` WHERE PARTY='Democratic' AND PARTY='Republican'`

✓ `SELECT * FROM `president` WHERE PARTY='Democratic' OR PARTY='Republican'`

✓ `SELECT * FROM `president` WHERE PARTY IN ('Democratic' , 'Republican')`

✓
```
SELECT * FROM `president` WHERE PARTY='Democratic'
UNION
SELECT * FROM `president` WHERE PARTY='Republican'
```

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

List rows of presidents who served between 6 to 8 years

```
SELECT    *
FROM      `president`
WHERE     YRS_SERV>=6 AND YRS_SERV <=8
```

```
SELECT    *
FROM      'president'
WHERE     YRS_SERV  BETWEEN 6 AND 8
```

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

List rows of presidents whose name begins with R.

```
SELECT    *
FROM      `president`
WHERE     PRES_NAME  LIKE  'R%'
```

- % used with LIKE means string of any length including null string.
- The opposite of LIKE is NOT LIKE

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

List rows of presidents whose name begins with R and without the letter ‘g’ anywhere in the name.

```
SELECT    *
FROM      `president`
WHERE     PRES_NAME  LIKE  ‘R%’
          AND  PRES_NAME NOT LIKE ‘%g%’
```

Build-in Function Aggregates

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

- The SQL provides two types of build-in functions.
  - Ones operates on **columns**
    - **AVG**
    - **SUM**
    - **MIN**
    - **MAX**

    } Nulls are not included in the calculation.
  - The other one operates on **rows**
    - **COUNT**
- These functions are called **aggregates** since they receive multiple values as input and provide a single value output.
- Other functions can be found in the commercial DBMS products' reference manuals.

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

Find the minimum death age of presidents who already passed away

```
SELECT    MIN(DEATH_AGE)
FROM      PRESIDENT
```

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

The most common mistake is as follows

```
SELECT    PRES_NAME , MIN(DEATH_AGE)
FROM      PRESIDENT
```

The SELECT PRES_NAME returns a set of value – a list of presidents while MIN(DEATH_AGE) is an aggregate which returns the one value.

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

Set of values output and single value output cannot be together in the output list since it leads to a non-relation output.

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

The SQL logical execution sequence

1. FROM
2. WHERE
3. GROUP BY
4. Build-in Function

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

List the total number of years served by Democratic presidents

```
SELECT    SUM(YRS_SERVE)
FROM      PRESIDENT
WHERE     PARTY = 'Democratic'
```

What if we want the party name 'Democratic' to appear with the number of year?

```
SELECT    PARTY, SUM(YRS_SERVE)
FROM      PRESIDENT
WHERE     PARTY = 'Democratic'
```

Syntax error : due to the aforementioned reason.

List the total number of years served by Democratic presidents with the party name 'Democratic' to appear with the number of year?

Some tricks!

```
SELECT    MIN(PARTY) , SUM(YRS_SERVE)
FROM      PRESIDENT
WHERE     PARTY = 'Democratic'
```

However, it works if the column in the output list has only one value left after the WHERE clause.

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

List number of presidents

```
SELECT    COUNT(*)
FROM      PRESIDENT
```

List number of married presidents

```
SELECT    COUNT( DISTINCT PRES_NAME  )
FROM      PRES_MARRIAGE
```

Calculations

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

- The SQL provides basic calculation operators : + , - , * , /.
- The order of operations (called operator precedence) is the conventional * / before + - and left to right.

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

What is the average death age of the presidents?

```
SELECT    AVG(DEATH_AGE )
FROM      PRESIDENT
```

ANS: 68.8571

```
SELECT    SUM(DEATH_AGE ) / COUNT(*)
FROM      PRESIDENT
```

ANS: 61.7949

The problem is from the COUNT function which count the number of rows including the NULLs while the SUM function does not include NULL value.

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

For the presidents who served as presidents for more than 10% of their lives, list the president name and the percentage. Round up the percentage to two decimal digits and sort the output by the percentage in ascending order.

```
SELECT      PRES_NAME, ROUND(100.0 * YRS_SERV / DEATH_AGE, 2)
FROM        PRESIDENT
WHERE       YRS_SERV > 0.1 * DEATH_AGE
            AND  DEATH_AGE  is  NOT NULL
ORDER BY    2
```

The GROUP BY Clause

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

- Allows the rows to be grouped together according to a specified column list.
- After the grouping, built-in functions are then applied to each group.

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

For how many years that each party rules the country?

```
SELECT      SUM(YRS_SERV)
FROM        PRESIDENT
WHERE       PARTY='Demo-Rep'
UNION
SELECT      SUM(YRS_SERV)
FROM        PRESIDENT
WHERE       PARTY='Democratic'
UNION
SELECT      SUM(YRS_SERV)
FROM        PRESIDENT
WHERE       PARTY='Federalist'
UNION
```

```
SELECT      SUM(YRS_SERV)
FROM        PRESIDENT
WHERE       PARTY='Republican'
UNION
SELECT      SUM(YRS_SERV)
FROM        PRESIDENT
WHERE       PARTY='Whig'
```

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

For how many years that each party rules the country?

```
SELECT      SUM(YRS_SERV)
FROM        PRESIDENT
GROUP BY    PARTY
```

```
SUM(YRS_SERV)
28
73
11
67
6
```

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

For how many years that each party rules the country?
Show the party along with the number of years in order.

```
SELECT      PARTY , SUM(YRS_SERV)
FROM        PRESIDENT
GROUP BY    PARTY
ORDER BY    2  DESC
```

PARTY	SUM(YRS_SERV)
Democratic	73
Republican	67
Demo-Rep	28
Federalist	11
Whig	6

The HAVING Clause

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

- The SQL logical execution sequence is (as aforementioned) the FROM → the WHERE → the GROUP BY → the built-in functions.

- What if there are conditions on built-in functions (aggregates)?

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

Consider only the Democratic and Republican parties, list the party names and the total number of years that parties rule the country is more than 70 years.

```
SELECT      PARTY , SUM(YRS_SERV)
FROM        PRESIDENT
WHERE       PARTY   IN   ('Democratic', 'Republican')
GROUP BY    PARTY
HAVING      SUM(YRS_SERV) > 70
```

PARTY	SUM(YRS_SERV)
Democratic	73

pakorn.wa@kmitl.ac.th

The difference between the WHERE and the HAVING clause is that the WHERE clause is for row conditions, mostly on column names, but the HAVING clause is for group conditions, mostly on aggregate built-in functions.

Join Operation

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

- Some queries require output columns from more than one table using JOIN operations.
- Types of JOIN operations
  - **Cross join** (or the Cartesian product)
  - **Inner join** (or equijoin)
  - **Natural join**
  - **Outer join** (**Left** or **Right**)
  - Etc.

List president name, party and hobbies of presidents who had hobbies.

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

```
SELECT      RECENT_PRESIDENTS.PRES_NAME , PARTY , HOBBY
FROM        RECENT_PRESIDENTS, PRES_HOBBY
WHERE       RECENT_PRESIDENTS.PRES_NAME
            = PRES_HOBBY.PRES_NAME

SELECT      RECENT_PRESIDENTS.PRES_NAME , PARTY , HOBBY
FROM        RECENT_PRESIDENTS  INNER JOIN  PRES_HOBBY ON
            RECENT_PRESIDENTS.PRES_NAME
            = PRES_HOBBY.PRES_NAME

SELECT      PRES_NAME, PARTY, HOBBY
FROM        RECENT_PRESIDENTS  NATURAL JOIN  PRES_HOBBY
```

Note that non-match rows are eliminated.

List president name, party and hobbies of presidents who had hobbies.

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

PRES_NAME	PARTY	HOBBY
Roosevelt F D	Democratic	Fishing
Roosevelt F D	Democratic	Sailing
Roosevelt F D	Democratic	Swimming
Truman H S	Democratic	Fishing
Truman H S	Democratic	Poker
Truman H S	Democratic	Walking
Eisenhower D D	Republican	Bridge
Eisenhower D D	Republican	Fishing
Eisenhower D D	Republican	Goft
Eisenhower D D	Republican	Hunting
Eisenhower D D	Republican	Painting
Kennedy J F	Democratic	Sailing
Kennedy J F	Democratic	Swimming
Kennedy J F	Democratic	Touch Football
Johnson L B	Democratic	Riding
Nixon R M	Republican	Goft

- The SQL provides **a set of OUTER JOIN** keywords for the join operations which unmatched rows are also required.

- The LEFT [OUTER] JOIN performs an outer join of table X and Y and returns all rows from X with the matched rows from Y together with all rows in X that have no matching rows in Y – which contain null values.

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

```
SELECT  RECENT_PRESIDENTS.PRES_NAME , PARTY , HOBBY
FROM    RECENT_PRESIDENTS  LEFT OUTER JOIN  PRES_HOBBY ON
        RECENT_PRESIDENTS.PRES_NAME
        = PRES_HOBBY.PRES_NAME
```

PRES_NAME	PARTY	HOBBY
Roosevelt F D	Democratic	Fishing
Roosevelt F D	Democratic	Sailing
Roosevelt F D	Democratic	Swimming
Truman H S	Democratic	Fishing
Truman H S	Democratic	Poker
Truman H S	Democratic	Walking
Eisenhower D D	Republican	Bridge
Eisenhower D D	Republican	Fishing
Eisenhower D D	Republican	Goft
Eisenhower D D	Republican	Hunting
Eisenhower D D	Republican	Painting
Kennedy J F	Democratic	Sailing
Kennedy J F	Democratic	Swimming
Kennedy J F	Democratic	Touch Football
Johnson L B	Democratic	Riding
Nixon R M	Republican	Goft
Ford G R	Republican	NULL
Carter J E	Democratic	NULL
Reagan R	Republican	NULL

Notice that President Carter, Ford, and Reagan, who did not hobby also appear in the output results.

- The **RIGHT [OUTER] JOIN** performs an outer join of tables X and Y and returns all rows from Y with the matched rows from X together with all rows in Y that have no matching rows in X – which will contain null values.
- The **FULL [OUTER] JOIN** performs an outer join and returns all rows from X and Y, extended with nulls if they do not satisfy the join condition.

List president name, birth year, and total number of children from all marriages of Republican presidents who marries more than once and have more than three children in total. Sort the output by total number of children in descending order.

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

```
SELECT      P1.PRES_NAME, BIRTH_YR, SUM(NR_CHILDREN)
FROM        PRESIDENT P1 , PRES_MARRIAGE P2
WHERE       P1.PRES_NAME=P2.PRES_NAME
            AND PARTY='Republican'
GROUP BY    P1.PRES_NAME, BIRTH_YR
HAVING      COUNT(*) > 1
            AND SUM(NR_CHILDREN) > 3
ORDER BY    3 DESC
```

PRES_NAME	BIRTH_YR	SUM(NR_CHILDREN)
Roosevelt T	1858	6
Reagan R	1911	4

Subqueries

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

- The **subqueries** allow queries to be used inside of other queries.
- Subqueries are used when a question can be further broken down into small parts.

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

List presidential details (the entire row) of the president who died the youngest.

```
SELECT      PRES_NAME, DEATH_AGE
FROM        PRESIDENT
WHERE       DEATH_AGE = (   SELECT      MIN(DEATH_AGE)
                            FROM        PRESIDENT
                            WHERE       DEATH_AGE IS NOT NULL   )
```

PRES_NAME	DEATH_AGE
Kennedy J F	46

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

List presidential details (the entire row) of the president who died the second youngest.

```
SELECT    PRES_NAME, DEATH_AGE
FROM      PRESIDENT
WHERE     DEATH_AGE = ( SELECT   MIN(DEATH_AGE)
                        FROM     PRESIDENT
                        WHERE    DEATH_AGE  >
                                 ( SELECT      MIN(DEATH_AGE)
                                   FROM        PRESIDENT
                                   WHERE       DEATH_AGE IS NOT NULL  )  )
```

PRES_NAME	DEATH_AGE
Garfield J A	49

KMITL
พระจอมเกล้าลาดกระบัง
pakorn.wa@kmitl.ac.th

Q & A