**Structured Query Language (SQL)**

**T.M. Rao**

**Notes-03-sql**

**SQL: Structured Query Language**

* Not a general-purpose language like Java or C++
* Developed at IBM in the 1970’s
* ANSI Standard in 1992 (SQL-92)
* All DBMS’s process SQL. Even if you do things using the GUI components, what you do is eventually translated into SQL for the DBMS to process. Good to learn SQL.
* Language for manipulating tables
* *Declarative* – Statement specifies *what* needs to be obtained, *not* *how* it is to be achieved (*e.g*., how to access data, the order of operations)
* Due to declarativity of SQL, DBMS determines evaluation strategy
  + This greatly simplifies application programs
* SQL is not case-sensitive
  + Not case sensitive for key words like: SELECT FROM …
  + Not case sensitive for column (field) names
  + But, it is case sensitive for table names
* We are using a version of sql called: mysql. Version 5.1.73

**Refer to this website:** <https://dev.mysql.com/doc/refman/5.1/en/> for more information.

**How to CREATE a table using mysql on phpmyadmin**

**Intractively:**

* Login to phpmyadmin: http://csdb.brockport.edu/phpmyadmin/
* Choose the database (say xyztest) – in the left-hand-side panel
* Click on operations tab (if the Create table option is not seen)
* Use the Create new table on database option. Enter the table name and number of fields and click go. Follow the instructions afterwards.

**Or Using SQL**

**DROP TABLE IF EXISTS `STUDENT\_2017`; #observe the backquotes**

**CREATE TABLE STUDENT\_2017(**

**Id CHAR( 9 ) ,**

**Name VARCHAR( 32 ) ,**

**Address VARCHAR( 64 ) ,**

**Status VARCHAR( 16 ) ,**

**PRIMARY KEY ( Id )**

**);**

**How to INSERT new rows into a Table?**

**Interactively:**

* Select the table (from the list of tables on the left) where you want to insert rows.
* Choose INSERT, follow instructions.

**Or using SQL:**

**INSERT INTO STUDENT\_2017 VALUES**

**('111111111', 'John Doe', '123 Main St.', 'Freshman'),**

**('666666666', 'Joseph Public', '666 Hollow Rd.', 'Sophomore'),**

**('111223344', 'Mary Smith', '1 Lake St.', 'Freshman'),**

**('987654321', 'Bart Simpson', 'Fox 5 TV', 'Senior'),**

**('023456789', 'Homer Simpson', 'Fox 5 TV', 'Senior'),**

**('123454321', 'Joe Blow', '6 Yard Ct.', 'Junior');**

**SQL Commands**

**SHOW TABLES; 🡨 Displays all tables in the current database**

**DESCRIBE <tablename>; 🡨Describes all the details about the table (Meta data)**

**Retrieving information from a table**

**SELECT** statement: Most commonly used data retrieval statement. It has a simple syntax:

SELECT <Something> 🡨 This specifies columns

FROM <One or more Tables> 🡨 This specifies tables

WHERE <some conditions> 🡨 conditions that the data values are required to satisfy

FROM and WHERE clauses are optional.

**Examples:**

1. SELECT statement can be used to do some simple calculations:

**SELECT 1 +1;**

| [Full Texts](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=&sql_query=SELECT+1+1&goto=db_sql.php&dontlimitchars=1&token=f46fe8c6aef5656405aeba4f6e79aae6) |
| --- |
| 1+1 |
| 2 |

**SELECT SIN(PI()/4), (4+1)\*5;**

| [Full Texts](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=&sql_query=SELECT+SIN(PI()/4),+(4+1)*5&goto=db_sql.php&dontlimitchars=1&token=ff02ab217907d9f518b541b4014e7bdb) | |
| --- | --- |
| SIN(PI()/4) | (4+1)\*5 |
| 0.70710678118655 | 25 |

2. We can find out the version we are using, current date, user, etc. as well. Observe that VERSION(), USER() are a pre-defined functions.

**SELECT VERSION( ) , CURRENT\_DATE;**

**Note: CURRENT\_DATE is a variable**

| [Full Texts](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=&sql_query=SELECT+VERSION(),+CURRENT_DATE&goto=db_sql.php&dontlimitchars=1&token=ff02ab217907d9f518b541b4014e7bdb) | | | | |
| --- | --- | --- | --- | --- |
| **VERSION( )** | **CURRENT\_DATE** |  |
| 5.1.73 | 2016-01-28 |  |

**Refer to this website:** <https://dev.mysql.com/doc/refman/5.1/en/> for more information.

**You can enter the query in multiple lines:**

**SELECT USER(), CURRENT\_DATE;**

USER()CURRENT\_DATE

trao@localhost2016-01-28

**SELECT CURDATE() Note: CURDATE() is a function.**

| **[Full Texts](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&sql_query=SELECT+curdate()&goto=tbl_sql.php&dontlimitchars=1&token=99c51b3859c8e7b7600d91c20f65a4e3)** |
| --- |
| **curdate()** |
| 2012-01-19 |

Similarly CURTIME() is also a function. CURRENT\_TIME is an environmental variable. Shows time in hh:mm:ss format

**Suppose we have created a table called STUDENT and it has the following information:**

| [Id](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=student&token=f46fe8c6aef5656405aeba4f6e79aae6&sql_query=SELECT+%2A+FROM+student+ORDER+BY+%60student%60.%60Id%60+ASC) | [Name](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=student&token=f46fe8c6aef5656405aeba4f6e79aae6&sql_query=SELECT+%2A+FROM+student+ORDER+BY+%60student%60.%60Name%60+ASC) | [Address](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=student&token=f46fe8c6aef5656405aeba4f6e79aae6&sql_query=SELECT+%2A+FROM+student+ORDER+BY+%60student%60.%60Address%60+ASC) | [Status](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=student&token=f46fe8c6aef5656405aeba4f6e79aae6&sql_query=SELECT+%2A+FROM+student+ORDER+BY+%60student%60.%60Status%60+ASC) |
| --- | --- | --- | --- |
| 111111111 | John Doe | 123 Main St. | Freshman |
| 666666666 | Joseph Public | 666 Hollow Rd. | Sophomor |
| 111223344 | Mary Smith | 1 Lake St. | Freshman |
| 987654321 | Bart Simpson | Fox 5 TV | Senior |
| 023456789 | Homer Simpson | Fox 5 TV | Senior |
| 123454321 | Joe Blow | 6 Yard Ct. | Junior |

3. You can retrieve the entire table using SELECT \* …

Query: Show the entire contents of the table STUDENT

**SELECT \* FROM STUDENT;**

| [Id](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=student&token=f46fe8c6aef5656405aeba4f6e79aae6&sql_query=SELECT+%2A+FROM+student+ORDER+BY+%60student%60.%60Id%60+ASC) | [Name](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=student&token=f46fe8c6aef5656405aeba4f6e79aae6&sql_query=SELECT+%2A+FROM+student+ORDER+BY+%60student%60.%60Name%60+ASC) | [Address](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=student&token=f46fe8c6aef5656405aeba4f6e79aae6&sql_query=SELECT+%2A+FROM+student+ORDER+BY+%60student%60.%60Address%60+ASC) | [Status](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=student&token=f46fe8c6aef5656405aeba4f6e79aae6&sql_query=SELECT+%2A+FROM+student+ORDER+BY+%60student%60.%60Status%60+ASC) |
| --- | --- | --- | --- |
| 111111111 | John Doe | 123 Main St. | Freshman |
| 666666666 | Joseph Public | 666 Hollow Rd. | Sophomor |
| 111223344 | Mary Smith | 1 Lake St. | Freshman |
| 987654321 | Bart Simpson | Fox 5 TV | Senior |
| 023456789 | Homer Simpson | Fox 5 TV | Senior |
| 123454321 | Joe Blow | 6 Yard Ct. | Junior |

2. Using WHERE clause we can select only certain rows of the table

Query: Show me all the details of the student whose Id is 987654321

**SELECT \* FROM STUDENT WHERE Id = '987654321';**

| Id | | Name | | Address | | Status | |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 987654321 | | Bart Simpson | | Fox 5 TV | | Senior | |

3. Using \* we can show all the columns of the selected rows. If we wish to display some columns and not the others, we can specify those columns:

Query: Show me just the name of the student whose Id is 987654321

**SELECT Name FROM STUDENT WHERE Id = '987654321';**

| Name |
| --- |
| Bart Simpson |

4. Query: Show me the Id and names of all students whose status is Senior.

**SELECT Id, Name FROM STUDENT WHERE Status = 'Senior';**

**(Note: SQL is not case-sensitive. So 'senior' would also work. But, it case-sensitive for table names!!))**

| [**Id**](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=student&token=f46fe8c6aef5656405aeba4f6e79aae6&sql_query=SELECT+Id%2C+Name+FROM+student+WHERE+Status+%3D+%27Senior%27+ORDER+BY+%60student%60.%60Id%60+ASC) | | [**Name**](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=student&token=f46fe8c6aef5656405aeba4f6e79aae6&sql_query=SELECT+Id%2C+Name+FROM+student+WHERE+Status+%3D+%27Senior%27+ORDER+BY+%60student%60.%60Name%60+ASC) | |
| --- | --- | --- | --- |
|  | 987654321 | | Bart Simpson | |
|  | 023456789 | | Homer Simpson | |

5. You can use the COUNT() function to count the number of rows that satisfy a condition.

Query: How many seniors are there in the STUDENT table?

**SELECT COUNT(\*) FROM STUDENT WHERE Status = 'Senior';**

| COUNT(\*) |
| --- |
| 2 |

6. You can use OR’s and AND’s NOT’s in the condition in the WHERE clause. You can also use comparisons like: <, >, <=, >=.

Query: What are the names of students whose Status is senior and Id > 888888888?

**SELECT Name FROM STUDENT WHERE Status = 'senior' AND Id > '888888888';**

| **Name** |
| --- |
| Bart Simpson |

7. You can use the IN operator to check the membership in an enumeration (set of values).

Query: What are the names of students who are either freshman or sophomore.

**SELECT Name FROM STUDENT WHERE Status IN ('freshman', 'sophomore');**

| [Name](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=student&token=f46fe8c6aef5656405aeba4f6e79aae6&sql_query=SELECT+Name+FROM+student+WHERE+Status+IN+%28%27freshman%27%2C+%27sophomore%27%29+ORDER+BY+%60student%60.%60Name%60+ASC) |
| --- |
| John Doe |
| Mary Smith |

**MORE MySQL**

**Suppose we have created another table called PET that contains the following data:**

**+----------+--------+---------+------+------------+------------+**

**| name | owner | species |Gender| BirthDate | DeathDate |**

**+----------+--------+---------+------+------------+------------+**

**| Fluffy | Harold | cat | f | 1993-02-04 | NULL |**

**| Claws | Gwen | cat | m | 1994-03-17 | NULL |**

**| Buffy | Harold | dog | f | 1989-05-13 | NULL |**

**| Fang | Benny | dog | m | 1990-08-27 | NULL |**

**| Bowser | Diane | dog | m | 1979-08-31 | 1995-07-29 |**

**| Chirpy | Gwen | bird | f | 1998-09-11 | NULL |**

**| Whistler | Gwen | bird | NULL | 1997-12-09 | NULL |**

**| Slim | Benny | snake | m | 1996-04-29 | NULL |**

**| Puffball | Diane | hamster | f | 1999-03-30 | NULL |**

**+----------+--------+---------+------+------------+------------+**

**Show me all information about pets who were born in or after 1998.**

**SELECT \* FROM PET\_2016 WHERE BirthDate >= '1998-01-01';**

**+----------+-------+---------+------+------------+-------+**

**| name | owner | species |Gender| BirthDate | DeathDate |**

**+----------+-------+---------+------+------------+-------+**

**| Chirpy | Gwen | bird | f | 1998-09-11 | NULL |**

**| Puffball | Diane | hamster | f | 1999-03-30 | NULL |**

**+----------+-------+---------+------+------------+-------+**

**Observe the use of OR in WHERE clause**

**SELECT \* FROM pet WHERE species = 'snake' OR species = 'bird';**

**+----------+-------+---------+------+------------+-------+**

**| name | owner | species |Gender| BirthDate | DeathDate |**

**+----------+-------+---------+------+------------+-------+**

**| Chirpy | Gwen | bird | f | 1998-09-11 | NULL |**

**| Whistler | Gwen | bird | NULL | 1997-12-09 | NULL |**

**| Slim | Benny | snake | m | 1996-04-29 | NULL |**

**+----------+-------+---------+------+------------+-------+**

**Observe the use of a complex boolean expression using AND, OR**

**SELECT \* FROM pet WHERE (species = 'cat' AND Gender = 'm')**

**OR (species = 'dog' AND Gender = 'f');**

**+-------+--------+---------+------+------------+-------+**

**| name | owner | species |Gender| BirthDate | DeathDate |**

**+-------+--------+---------+------+------------+-------+**

**| Claws | Gwen | cat | m | 1994-03-17 | NULL |**

**| Buffy | Harold | dog | f | 1989-05-13 | NULL |**

**+-------+--------+---------+------+------------+-------+**

**Display results in a specific order (Ascending or Descending):**

**Show me names of dates of BirthDate of all pets in the ascending order of dates of BirthDate.**

(Note: Table itself is not altered. Results are shown in the ascending order – default is ascending)

**SELECT name, BirthDate FROM pet ORDER BY BirthDate;**

**+----------+------------+**

**| name | BirthDate |**

**+----------+------------+**

**| Buffy | 1989-05-13 |**

**| Bowser | 1989-08-31 |**

**| Fang | 1990-08-27 |**

**| Fluffy | 1993-02-04 |**

**| Claws | 1994-03-17 |**

**| Slim | 1996-04-29 |**

**| Whistler | 1997-12-09 |**

**| Chirpy | 1998-09-11 |**

**| Puffball | 1999-03-30 |**

**+----------+------------+**

**Use keyword DESC if you want sorting in descending order.**

**SELECT name, BirthDate FROM pet ORDER BY BirthDate DESC;**

**+----------+------------+**

**| name | BirthDate |**

**+----------+------------+**

**| Puffball | 1999-03-30 |**

**| Chirpy | 1998-09-11 |**

**| Whistler | 1997-12-09 |**

**| Slim | 1996-04-29 |**

**| Claws | 1994-03-17 |**

**| Fluffy | 1993-02-04 |**

**| Fang | 1990-08-27 |**

**| Bowser | 1989-08-31 |**

**| Buffy | 1989-05-13 |**

**+----------+------------+**

**Working with Dates:** Date is displayed as a string 'yyyy-mm-dd'. CURDATE() returns the today’s date. YEAR(CURDATE()) gives the 4-char yyyy. MONTH(CURDATE()) gives the 2-char mm.

**Determine the age of each pet:** To determine how many years old each of your pets is, compute the difference in the year part of the current date and the birth date, then subtract one if the current date occurs earlier in the calendar year than the birth date.

Expressed in Java:

**if (RIGHT(CURDATE(),5)< RIGHT(BirthDate,5))**

**value = 1;**

**else**

**value = 0;**

**age = (YEAR(CURDATE())-YEAR(BirthDate)) – value;**

The following query shows, for each pet, the birth date, the current date, and the age in years.

Note:

* **(YEAR(CURDATE())-YEAR(BirthDate)) returns the difference between current year and year of the BirthDate. (In case of Fluffy, 2012 – 1993 = 19)**
* **RIGHT(CURDATE(),5) gives rightmost 5 digits of today’s date: 01-19**
* **(RIGHT(CURDATE(),5)<RIGHT(BirthDate,5))**

**In case of Fluffy, 01-19 < 02-04 = true. Just like in C language, true is 1, anything nonzero is true, 0 or NULL is false. In this case, true = 1, and hence the age is 19 – 1 = 18.**

**SELECT Name, BirthDate, CURDATE(),**

**(YEAR(CURDATE())-YEAR(BirthDate)) – (RIGHT(CURDATE(),5)<RIGHT(BirthDate,5))**

**AS age**

**FROM PET;**

| [Full Texts](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&sql_query=+++++++SELECT+Name,+BirthDate,+CURDATE(),%0d%0a+(YEAR(CURDATE())-YEAR(BirthDate))%0d%0a+-+(RIGHT(CURDATE(),5)%3cRIGHT(BirthDate,5))%0d%0a+AS+age%0d%0a+FROM+PET&goto=tbl_sql.php&dontlimitchars=1&token=99c51b3859c8e7b7600d91c20f65a4e3) | | | |
| --- | --- | --- | --- |
| ([Name](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=+++++++SELECT+Name%2C+BirthDate%2C+CURDATE%28%29%2C%0A+%28YEAR%28CURDATE%28%29%29-YEAR%28BirthDate%29%29%0A+-+%28RIGHT%28CURDATE%28%29%2C5%29%3CRIGHT%28BirthDate%2C5%29%29%0A+AS+age%0A+FROM+PET+ORDER+BY+%60PET%60.%60Name%60+ASC) | [BirthDate](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=+++++++SELECT+Name%2C+BirthDate%2C+CURDATE%28%29%2C%0A+%28YEAR%28CURDATE%28%29%29-YEAR%28BirthDate%29%29%0A+-+%28RIGHT%28CURDATE%28%29%2C5%29%3CRIGHT%28BirthDate%2C5%29%29%0A+AS+age%0A+FROM+PET+ORDER+BY+%60PET%60.%60BirthDate%60+ASC) | [CURDATE()](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=+++++++SELECT+Name%2C+BirthDate%2C+CURDATE%28%29%2C%0A+%28YEAR%28CURDATE%28%29%29-YEAR%28BirthDate%29%29%0A+-+%28RIGHT%28CURDATE%28%29%2C5%29%3CRIGHT%28BirthDate%2C5%29%29%0A+AS+age%0A+FROM+PET+ORDER+BY+CURDATE%28%29+ASC) | [age](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=+++++++SELECT+Name%2C+BirthDate%2C+CURDATE%28%29%2C%0A+%28YEAR%28CURDATE%28%29%29-YEAR%28BirthDate%29%29%0A+-+%28RIGHT%28CURDATE%28%29%2C5%29%3CRIGHT%28BirthDate%2C5%29%29%0A+AS+age%0A+FROM+PET+ORDER+BY+%60age%60+ASC) |
| Fluffy | 1993-02-04 | 2012-01-19 | 18 |
| Claws | 1994-03-17 | 2012-01-19 | 17 |
| Buffy | 1989-05-13 | 2012-01-19 | 22 |
| Fang | 1990-08-27 | 2012-01-19 | 21 |
| Bowser | 1979-08-31 | 2012-01-19 | 32 |
| Chirpy | 1998-09-11 | 2012-01-19 | 13 |
| Whistler | 1997-12-09 | 2012-01-19 | 14 |
| Slim | 1996-04-29 | 2012-01-19 | 15 |
| Puffball | 1999-03-30 | 2012-01-19 | 12 |

(Remember: Today’s date is 2012-01-19)

**What are all the pets that were born in May?**

**SELECT Name, BirthDate FROM PET WHERE MONTH(BirthDate) = 5;**

| Name | BirthDate |
| --- | --- |
| Buffy | 1989-05-13 |

**NULL is a special value**

For any value, you can check if it IS NULL or IS NOT NULL

SELECT 1 IS NULL, 1 IS NOT NULL;

Will return

| **[Full Texts](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&sql_query=SELECT+1+IS+NULL,+1+IS+NOT+NULL&goto=tbl_sql.php&dontlimitchars=1&token=99c51b3859c8e7b7600d91c20f65a4e3)** | |
| --- | --- |
| **1 IS NULL** | **1 IS NOT NULL** |
| 0 | 1 |

Note:

Two NULL values are regarded as equal in a GROUP BY. When doing an ORDER BY, NULL values are presented first if you do ORDER BY ... ASC and last if you do ORDER BY ... DESC.

**Pattern Matching:**

The underscore character (‘\_’) is the wildcard that matches any single character.

The percent character (‘%’) is the wild card that matches 0 or more characters.

Use LIKE and NOT LIKE to do comparison.

Query: Show me all the details of pets whose name begins with ‘b’

**SELECT \* FROM PET WHERE Name LIKE 'b%';**

| [Name](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27b%25%27+ORDER+BY+%60PET%60.%60Name%60+ASC) | [Owner](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27b%25%27+ORDER+BY+%60PET%60.%60Owner%60+ASC) | [Species](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27b%25%27+ORDER+BY+%60PET%60.%60Species%60+ASC) | [Gender](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27b%25%27+ORDER+BY+%60PET%60.%60Gender%60+ASC) | [BirthDate](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27b%25%27+ORDER+BY+%60PET%60.%60BirthDate%60+ASC) | [DeathDate](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27b%25%27+ORDER+BY+%60PET%60.%60DeathDate%60+ASC) |
| --- | --- | --- | --- | --- | --- |
| Bowser | Diane | dog | m | 1979-08-31 | 1995-07-29 |
| Buffy | Harold | dog | f | 1989-05-13 | *NULL* |

Query: All names ending in ‘fy’

**SELECT \* FROM PET WHERE Name LIKE '%fy';**

| [Name](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27%25fy%27+ORDER+BY+%60PET%60.%60Name%60+ASC) | [Owner](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27%25fy%27+ORDER+BY+%60PET%60.%60Owner%60+ASC) | [Species](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27%25fy%27+ORDER+BY+%60PET%60.%60Species%60+ASC) | [Gender](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27%25fy%27+ORDER+BY+%60PET%60.%60Gender%60+ASC) | [BirthDate](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27%25fy%27+ORDER+BY+%60PET%60.%60BirthDate%60+ASC) | [DeathDate](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27%25fy%27+ORDER+BY+%60PET%60.%60DeathDate%60+ASC) |
| --- | --- | --- | --- | --- | --- |
| Fluffy | Harold | cat | f | 1993-02-04 | *NULL* |
| Buffy | Harold | dog | f | 1989-05-13 | *NULL* |

Query: All names containing ‘w’

**SELECT \* FROM PET WHERE Name LIKE '%w%';**

| [Name](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27%25w%25%27+ORDER+BY+%60PET%60.%60Name%60+ASC) | [Owner](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27%25w%25%27+ORDER+BY+%60PET%60.%60Owner%60+ASC) | [Species](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27%25w%25%27+ORDER+BY+%60PET%60.%60Species%60+ASC) | [Gender](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27%25w%25%27+ORDER+BY+%60PET%60.%60Gender%60+ASC) | [BirthDate](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27%25w%25%27+ORDER+BY+%60PET%60.%60BirthDate%60+ASC) | [DeathDate](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27%25w%25%27+ORDER+BY+%60PET%60.%60DeathDate%60+ASC) |
| --- | --- | --- | --- | --- | --- |
| Claws | Gwen | cat | m | 1994-03-17 | *NULL* |
| Bowser | Diane | dog | m | 1979-08-31 | 1995-07-29 |
| Whistler | Gwen | bird | *NULL* | 1997-12-09 | *NULL* |

Query: All 5-letter names: (Use \_ 5 times)

**SELECT \* FROM PET WHERE Name LIKE '\_\_\_\_\_';**

| [**Name**](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27_____%27+ORDER+BY+%60PET%60.%60Name%60+ASC) | [**Owner**](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27_____%27+ORDER+BY+%60PET%60.%60Owner%60+ASC) | [**Species**](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27_____%27+ORDER+BY+%60PET%60.%60Species%60+ASC) | [**Gender**](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27_____%27+ORDER+BY+%60PET%60.%60Gender%60+ASC) | [**BirthDate**](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27_____%27+ORDER+BY+%60PET%60.%60BirthDate%60+ASC) | [**DeathDate**](http://csdb.brockport.edu/phpmyadmin/sql.php?db=test&table=PET&token=99c51b3859c8e7b7600d91c20f65a4e3&sql_query=SELECT+%2A+FROM+PET+WHERE+Name+LIKE+%27_____%27+ORDER+BY+%60PET%60.%60DeathDate%60+ASC) |
| --- | --- | --- | --- | --- | --- |
| **Claws** | **Gwen** | **cat** | **m** | **1994-03-17** | ***NULL*** |
| **Buffy** | **Harold** | **dog** | **f** | **1989-05-13** | ***NULL*** |