mysql> SELECT \* FROM pet;

列出pet 資料表的內容

mysql> DELETE FROM pet;

刪除pet所有內容

mysql> LOAD DATA LOCAL INFILE 'pet.txt' INTO TABLE pet;

將檔名'pet.txt'的資料，匯入pet資料表內

mysql> UPDATE pet SET birth = '1989-08-31' WHERE name = 'Bowser';

更新資料表中的資料記錄的

mysql> SELECT \* FROM pet WHERE name = 'Bowser';

從名為 "pet" 的資料表中檢索符合特定條件的資料記錄

mysql> SELECT \* FROM pet WHERE birth >= '1998-1-1';

檢索出在 '1998-01-01' 或之後出生的寵物記錄。

SELECT \* FROM pet WHERE species = 'dog' AND sex = 'f';

查詢是用來檢索符合特定條件的資料記錄，條件是"species"（物種）為 'dog'（狗）且"sex"（性別）為 'f'（女性）。

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

查詢是用來檢索符合特定條件的資料記錄，條件是"species"（物種）為 'snake'（蛇）或者 'bird'（鳥）。

mysql> SELECT \* FROM pet WHERE (species = 'cat' AND sex = 'm')

OR (species = 'dog' AND sex = 'f');

檢索符合複合條件的資料記錄，條件是要麼"species"（物種）為 'cat'（貓）且"sex"（性別）為 'm'（男性），要麼"species"為 'dog'（狗）且"sex"為 'f'（女性）。

SELECT name, birth FROM pet;

查詢是用來檢索資料表 "pet" 中所有寵物的名稱（name）和出生日期（birth）的資訊

SELECT owner FROM pet;

查詢是用來檢索資料表 "pet" 中所有寵物的擁有者（owner）的資訊

mysql> SELECT DISTINCT owner FROM pet;

查詢是用來檢索資料表 "pet" 中不重複的擁有者（owner）的資訊

SELECT name, species, birth FROM pet

WHERE species = 'dog' OR species = 'cat';

查詢是用來檢索資料表 "pet" 中符合特定條件的寵物的名稱（name）、物種（species）和出生日期（birth）的資訊。具體條件是物種為 'dog'（狗）或 'cat'（貓）的寵物。

mysql> SELECT name, birth FROM pet ORDER BY birth;

查詢用來檢索資料表 "pet" 中寵物的名稱（name）和出生日期（birth）的資訊，並按照出生日期升序排序（即從最早出生日期到最晚出生日期）列出結果。

mysql> SELECT name, birth FROM pet ORDER BY birth DESC;

查詢用來檢索資料表 "pet" 中寵物的名稱（name）和出生日期（birth）的資訊，並按照出生日期降序排序（即從最晚出生日期到最早出生日期）列出結果。

mysql> SELECT name, species, birth FROM pet

ORDER BY species, birth DESC;

查詢用來檢索資料表 "pet" 中寵物的名稱（name）、物種（species）和出生日期（birth）的資訊，並按照物種升序排序，如果物種相同則按照出生日期降序排序列出結果。

mysql> SELECT name, birth, CURDATE(),

TIMESTAMPDIFF(YEAR,birth,CURDATE()) AS age

FROM pet;

查詢用來檢索資料表 "pet" 中每個寵物的名稱（name）、出生日期（birth）、當前日期（CURDATE()），以及計算每個寵物的年齡（以年為單位）。

定位資料表 "pet"。

檢索每個寵物的名稱、出生日期、當前日期（CURDATE()）。

使用TIMESTAMPDIFF函數計算每個寵物的年齡。TIMESTAMPDIFF函數將出生日期（birth）和當前日期（CURDATE()）作為參數，並計算兩者之間的差距，以年為單位，然後將結果命名為 "age"。

返回包括名稱、出生日期、當前日期和年齡的結果。

mysql> SELECT name, birth, CURDATE(),

TIMESTAMPDIFF(YEAR,birth,CURDATE()) AS age

FROM pet ORDER BY name;

查詢用來檢索資料表 "pet" 中每個寵物的名稱（name）、出生日期（birth）、當前日期（CURDATE()），以及計算每個寵物的年齡（以年為單位），並按照名稱升序排序列出結果。

定位資料表 "pet"。

檢索每個寵物的名稱、出生日期、當前日期（CURDATE()）。

使用TIMESTAMPDIFF函數計算每個寵物的年齡。TIMESTAMPDIFF函數將出生日期（birth）和當前日期（CURDATE()）作為參數，並計算兩者之間的差距，以年為單位，然後將結果命名為 "age"。

將結果按照名稱升序排序，即按照寵物的名稱字母順序排列。

返回包括名稱、出生日期、當前日期和年齡的結果，按照名稱升序排列。

mysql> SELECT name, birth, CURDATE(),

TIMESTAMPDIFF(YEAR,birth,CURDATE()) AS age

FROM pet ORDER BY age;

查詢用來檢索資料表 "pet" 中每個寵物的名稱（name）、出生日期（birth）、當前日期（CURDATE()），以及計算每個寵物的年齡（以年為單位），然後按照年齡升序排序列出結果。

定位資料表 "pet"。

檢索每個寵物的名稱、出生日期、當前日期（CURDATE()）。

使用TIMESTAMPDIFF函數計算每個寵物的年齡。TIMESTAMPDIFF函數將出生日期（birth）和當前日期（CURDATE()）作為參數，並計算兩者之間的差距，以年為單位，然後將結果命名為 "age"。

將結果按照年齡升序排序，即按照寵物的年齡從小到大排列。

返回包括名稱、出生日期、當前日期和年齡的結果，按照年齡升序排列。

mysql> SELECT name, birth, death,

TIMESTAMPDIFF(YEAR,birth,death) AS age

FROM pet WHERE death IS NOT NULL ORDER BY age;

查詢是用來檢索資料表 "pet" 中已經有死亡日期（death）的寵物的名稱（name）、出生日期（birth）、死亡日期（death），以及計算每隻已故寵物的年齡（以年為單位），然後按照年齡升序排序列出結果。

定位資料表 "pet"。

檢索每隻已經有死亡日期（death）的寵物的名稱、出生日期、死亡日期。

使用TIMESTAMPDIFF函數計算每隻已故寵物的年齡。TIMESTAMPDIFF函數將出生日期（birth）和死亡日期（death）作為參數，並計算兩者之間的差距，以年為單位，然後將結果命名為 "age"。

使用WHERE子句篩選出那些已經有死亡日期的寵物（即死亡日期不為NULL的寵物）。

將結果按照年齡升序排序，即按照已故寵物的年齡從小到大排列。

返回包括名稱、出生日期、死亡日期和年齡的結果，按照年齡升序排列。

mysql> SELECT name, birth, MONTH(birth) FROM pet;

查詢用來檢索資料表 "pet" 中每個寵物的名稱（name）、出生日期（birth），以及出生日期的月份部分。具體來說，使用了MONTH函數來提取出生日期的月份。

mysql> SELECT name, birth FROM pet WHERE MONTH(birth) = 5;

查詢用來檢索資料表 "pet" 中出生日期（birth）的月份為5（即五月）的寵物的名稱（name）和出生日期（birth）

mysql> SELECT name, birth FROM pet

WHERE MONTH(birth) = MONTH(DATE\_ADD(CURDATE(),INTERVAL 1 MONTH));

查詢用來檢索資料表 "pet" 中出生日期（birth）的月份與當前月份的下一個月相同的寵物的名稱（name）和出生日期（birth）。

定位資料表 "pet"。

使用MONTH函數和DATE\_ADD函數來比較出生日期（birth）的月份是否等於當前日期（CURDATE()）的下一個月（即加上INTERVAL 1 MONTH後的月份）的月份。

篩選出那些出生日期的月份與下一個月相同的寵物。

將符合條件的寵物的名稱（name）和出生日期（birth）檢索出來。

返回結果，包括名稱和出生日期。

mysql> SELECT name, birth FROM pet

WHERE MONTH(birth) = MOD(MONTH(CURDATE()), 12) + 1;

查詢也是用來檢索資料表 "pet" 中出生日期（birth）的月份與當前月份相同的寵物的名稱（name）和出生日期（birth）。

定位資料表 "pet"。

使用MONTH函數和MOD函數來比較出生日期（birth）的月份是否等於當前日期（CURDATE()）的月份。MONTH(CURDATE())會返回當前日期的月份，而MOD(MONTH(CURDATE()), 12) + 1則會將當前月份除以12取餘數，再加1，以確保月份循環在1到12之間。

篩選出那些出生日期的月份與當前月份相同的寵物。

將符合條件的寵物的名稱（name）和出生日期（birth）檢索出來。

返回結果，包括名稱和出生日期。

mysql> SELECT '2018-10-31' + INTERVAL 1 DAY;

查詢將日期 '2018-10-31' 增加1天，結果顯示為 '2018-11-01'。這表示在原始日期 '2018-10-31' 的基礎上增加了一天，所以結果是下一天的日期 '2018-11-01'。這是一個常見的日期計算操作，用於在日期上添加或減去特定的時間間隔。

mysql> SELECT '2018-10-32' + INTERVAL 1 DAY;

查詢嘗試將日期 '2018-10-32' 增加1天，然而，這不是一個有效的日期，因為10月只有31天，無法增加到32號，所以結果為NULL。在日期計算中，如果嘗試處理無效的日期，MySQL通常會返回NULL或引發錯誤，因為無效日期在計算上是無法處理的。因此，結果顯示為NULL，並且可能會生成相關的警告，以指出日期值無效。

mysql> SHOW WARNINGS;

在前面的SQL查詢中，嘗試將日期 '2018-10-32' 增加1天，這是一個無效的日期計算，因為10月只有31天，無法增加到32號。當MySQL執行這個操作時，它會生成一個警告，指出日期值無效。您可以使用SHOW WARNINGS;來檢索生成的警告消息，以查看有關警告的更多詳細信息。

mysql> SELECT 1 IS NULL, 1 IS NOT NULL;

這個SQL查詢用來測試1是否為NULL和1是否不為NULL的布林表達式。

mysql> SELECT 1 = NULL, 1 <> NULL, 1 < NULL, 1 > NULL;

在SQL中，要注意 NULL 有特殊的性質，無法使用普通的比較運算符（如=、<>、<、>）來與 NULL 進行比較。因為 NULL 表示未知的值，所以無法確定它是否等於、不等於、大於或小於其他值。

在您的查詢中，嘗試將1與NULL進行比較，結果如下：

1 = NULL：這個表達式試圖比較1是否等於NULL，但由於NULL的特性，比較結果始終為NULL，表示結果未知。

1 <> NULL：這個表達式試圖比較1是否不等於NULL，但同樣由於NULL的特性，比較結果也始終為NULL，表示結果未知。

1 < NULL 和 1 > NULL：這兩個表達式試圖比較1是否小於或大於NULL，同樣由於NULL的特性，比較結果始終為NULL，表示結果未知。

總結，無法確定1與NULL的比較結果，因為NULL代表未知的值，無法與其他值進行直接的比較。在SQL中，要使用IS NULL或IS NOT NULL來測試是否為NULL或不為NULL，而不是使用普通的比較運算符。例如：

1 IS NULL：檢查1是否為NULL。

1 IS NOT NULL：檢查1是否不為NULL。

mysql> SELECT 0 IS NULL, 0 IS NOT NULL, '' IS NULL, '' IS NOT NULL;

+-----------+---------------+------------+----------------+

| 0 IS NULL | 0 IS NOT NULL | '' IS NULL | '' IS NOT NULL |

+-----------+---------------+------------+----------------+

| 0 | 1 | 0 | 1 |

+-----------+---------------+------------+----------------+

在SQL中，對於數值和字串的情況，我們可以使用 IS NULL 和 IS NOT NULL 運算符來測試是否為 NULL 或不為 NULL。

在您的查詢中，分別測試了數值0和空字串''（即雙引號之間沒有字符）：

0 IS NULL：這個表達式測試數值0是否為 NULL，結果是FALSE，因為0不是NULL。

0 IS NOT NULL：這個表達式測試數值0是否不為 NULL，結果是TRUE，因為0確實不是NULL。

'' IS NULL：這個表達式測試空字串''是否為 NULL，結果是FALSE，因為空字串不是NULL。

'' IS NOT NULL：這個表達式測試空字串''是否不為 NULL，結果是TRUE，因為空字串確實不是NULL。

結果表明，數值0和空字串''都不是NULL，因此 IS NULL 的結果為FALSE，而 IS NOT NULL 的結果為TRUE。

mysql> SELECT \* FROM pet WHERE name LIKE 'b%';

+--------+--------+---------+------+------------+------------+

| name | owner | species | sex | birth | death |

+--------+--------+---------+------+------------+------------+

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

| Bowser | Diane | dog | m | 1989-08-31 | 1995-07-29 |

這個SQL查詢用來檢索資料表 "pet" 中名稱（name）以字母 "b" 開頭的寵物記錄。

mysql> SELECT \* FROM pet WHERE name LIKE '%fy';

+--------+--------+---------+------+------------+-------+

| name | owner | species | sex | birth | death |

+--------+--------+---------+------+------------+-------+

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

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