

SQL Server - Searching for Particular Date Values and Ranges

All applications need to retrieve data in **SQL Server** tables based on **DATETIME** and/or **SMALLDATETIME** columns. In your particular application, you may need to select records that were entered on a particular date. On the other hand, you might need to select a set of records that have a **DATETIME** column value for a particular month, or year. In other case, you might want to find all the records between two different dates. Possibly, you might need to find the first, or last record entered in a given month, day, or year. This article will discuss selecting records from a database table based on values in a **DATETIME**, or **SMALLDATETIME** column.

Prior to discussing selecting records for a particular **DATETIME** value, let's review what specific values are stored in a given **DATETIME** and **SMALLDATETIME** column.

You should recall that a **DATETIME** column contains a date and time value, where time is accurate to milliseconds and **SMALLDATETIME** columns hold a date and time value, but the time portion is only accurate to one minute.

Since these **date/time** columns store the time portion you will need to consider this when searching for records where the column holds a specific date. You will need to provide the date and time portion in the search criteria or you may not return any records or the records you wish to return. If you are not sure of the exact time associated with the records you want to retrieve you should search based on a date and/or time range. Let's go through a couple of examples to show you what I am talking about.

DATE SAMPLE Table

In order to show you different methods of searching **SQL Server** tables, I will need a sample table. The table I will be using is a very simple table called **DATE_SAMPLE** and here is a list of records in that table.

RECORD	SAMPLE_DATE
1	2001-11-08 00:00:00.000
2	2002-04-08 16:00:00.000
3	2003-04-12 16:59:00.000
4	2003-04-09 00:00:00.000
5	2003-04-09 08:00:00.000
6	2003-04-09 14:58:00.000
7	2003-04-09 23:59:00.997
8	2003-04-10 00:00:00.000
9	2003-04-12 00:00:00.000
10	2003-05-10 00:00:00.000

Common Mistakes When Searching for Dates:

When searching for dates there are a number of common mistakes that new **SQL Server** programmers sometimes make. In this section, I will show you two common date/time pitfalls.

The intent of this first example is to select all the records in the **DATE_SAMPLE** table that have a **SAMPLE_DATE** equal to **'2003-04-09'**. Here is the code:

```
SELECT * FROM DATE_SAMPLE WHERE SAMPLE_DATE = '2003-04-09'
```

When this code is run only record 4 is returned. Why are records 5, 6 & 7 not returned?

Can you tell why? Remember **DATETIME**, or **SMALLDATE** columns contain not only the date but also the time. In this particular example **SAMPLE_DATE** is a **DATETIME** column, so all the dates store contain a time, down to the milliseconds. When you specify a search criteria that only contains a date, like the above example, SQL Server needs to first convert the string expression **'2003-04-09'** to a date and time value, prior to matching the string with the values in the **SAMPLE_DATE** column. This conversion creates a value of **'2003-04-09 00:00:00.000'**, which matches with only record 4.

Another common mistake is to use the **BETWEEN** verb like so:

```
SELECT * FROM DATE_SAMPLE WHERE SAMPLE_DATE between '2003-04-09'  
AND '2003-04-10'
```

When using the **BETWEEN** verb all records that are between or equal to the dates specified are returned. Now if in my example above I only wanted to return records that have a **SAMPLE_DATE** in **'2003-04-09'**. This example returns all the records that have a **SAMPLE_DATE** in **'2003-04-09'** (records 4 - 7), but also returns record 8 that has a **SAMPLE_DATE** of **'2003-04-10'**. Since the **BETWEEN** clause is inclusive of the two dates specified, record 8 is also returned.

Now if you really desire to select all the records in the **DATE_SAMPLE** table that have a **SAMPLE_DATE** sometime in **'2003-04-09'** you have a couple of options. Let me go through each option and then explain why one might be better than another might.

Using the Convert Function:

This first example selects all records from the **DATE_SAMPLE** where the date portion of the **SAMPLE_DATE** is equal to **'2003-04-09'**.

```
SELECT * FROM DATE_SAMPLE  
WHERE CONVERT(CHAR(10), SAMPLE_DATE, 120) = '2003-04-09'
```

The reason this example works, and the first example above does not, is because this example removes the time portion of the **SAMPLE_DATE** column prior to the comparison with string **'2003-04-09'** being performed. The **CONVERT** function removes the time portion by truncating the value of the **SAMPLE_DATE** field to only the first 10 characters.