

ALIAS in SQL Server

ALIASES in SQL Server is used to create temporary names (called aliases) for columns or tables.

ALIASES in SQL Server is used to create temporary names (called aliases) for columns or tables.

1. COLUMN ALIASES is used to create column headers in results for easy viewing
2. TABLE ALIASES is used to shorten SQL for easier readability or when you need to manually connect (for example, listing the same table more than once in the FROM clause).

Syntax for ALIASES alias

Syntax to set alias for column

```
ten_cot [ AS ] bi_danh
```

or set the alias for the table

```
ten_bang [AS] bi_danh
```

Variable name or variable value

ten_cot

The original name of the column to which you want the alias

ten_bang

The original name of the table to which you want the alias

AS

option. Most programmers use the keyword AS when setting the alias for the column but not when setting the table. Whether it is used or not, it does not affect the alias in MySQL. Unlike other databases, this is an optional option in MySQL. (The examples below will use AS when setting aliases for columns and removing AS when setting tables).

bi_danh

temporary names set for columns or tables.

Note

1. If bi_danh contains spaces, it must be placed in quotation marks.
2. You can use spaces when setting aliases for columns. But often, do not use spaces when setting tables.
3. The bi-name is only valid in SQL statements.

For example - set the alias for the column

Often aliases are used to create column headers in results that are easy to see.

```
SELECT nhanvien_id, ten + ho AS NAME
FROM nhanvien
WHERE ten = 'Sarah';
```

In this example, we set the alias for the second column (ie, combining the first and last name) as NAME. In the result, NAME will be the title of the second column. Because the above bi_danh has no spaces, no quotes are needed. But if you want, use this mark.

```
SELECT nhanvien_id, ten + ho AS 'NAME'
FROM nhanvien
WHERE ten = 'Sarah';
```

This is another example that needs to put bi_danh in quotation marks.

```
SELECT nhanvien_id, ten + ho AS 'TEN NHAN VIEN'
FROM nhanvien
WHERE ten = 'Sarah';
```

In this example, the second column in the result is set to TEN NHAN VIEN.

For example - set the alias for the table

Setting the alias for the table can be used to list a table more than once in the FROM clause (or self-connect) or shorten the table name to make it easier to read.

```
SELECT s.sanpham_ten, hangtonkho.chatluong
FROM sanpham s
INNER JOIN hangtonkho
ON s.sanpham_id = hangtonkho.sanpham_id
ORDER BY s.sanpham_ten ASC, hangtonkho.chatluong DESC;
```

The above example creates an alias for the sanpham table which is s. Now in the SQL statement, it is possible to call the sanpham table s. When creating a table alias, it is not necessary to create an alias for all tables given in the FROM clause. For example, create an alias for the hangtonkho table as shown below.

```
SELECT s.sanpham_ten, h.chatluong
FROM sanpham s
INNER JOIN hangtonkho h
ON s.sanpham_id = h.sanpham_id
ORDER BY s.sanpham_ten ASC, h.chatluong DESC;
```

The hangtonkho table is called h and the sanpham table is called s.

Previous article: [Conditions NOT in SQL Server](#)

Next lesson: [JOIN in SQL Server](#)

You finished reading the article "**ALIAS in SQL Server**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.
