

SET tablename[startcolumn,startrow] {}

The syntax above adds or rewrites data in a table.

- **tablename**: The name of the table with full path.
- **startcolumn**: Sets the starting column which to set data from.
- **startrow**: Sets the starting row which to set data from.

Theoretical example

Command:

```
SET table01[1,1]
{
new record, new record, new record
new record, new record, new record
new record, new record, new record
}
```

Output:

TABLE01	0	1	2	3
0	record	record	record	record
1	record	new record	new record	new record
2	record	new record	new record	new record
3	record	new record	new record	new record

Practical example

Command:

```
SET table01[1,1]
{
new record, new record, new record
new record, new record, new record
new record, new record, new record
}
```

Output:

```
SET::TABLE01[1,1]
```

You can see that the data we had been set is fitting into the existing table. What would be the consequences of putting a wider data range into the table? The answer is:

Command:

```
SET table01[3,3]
{
new record, new record, new record
new record, new record, new record
new record, new record, new record
}
```

Output:

TABLE01	0	1	2	3	4	5
0	record	record	record	record		
1	record	record	record	record		
2	record	record	record	record		
3	record	record	record	new record	new record	new record
4				new record	new record	new record
5				new record	new record	new record

You can see that the table has been automatically resized itself. Two new columns and rows have been created.

SET tablename[startcolumn,-1] {}

The syntax above adds a new row into a table. In case of empty brackets an empty row will be added.

- **tablename:** The name of the table with full path.
- **startcolumn:** Starting point of the column for the data insertion.
- **-1:** Indicating new row creation.

Theoretical example

Command:

```
SET table01[0,-1]
{
new record, new record, new record, new record
}
```

Output:

TABLE01	0	1	2	3
0	record	record	record	record
1	record	record	record	record
2	record	record	record	record
3	record	record	record	record
4	new record	new record	new record	new record

SET tablename[-1,startrow] {}

The syntax above adds a new column into a table. In case of empty brackets an empty column will be added.

- **tablename**: The name of the table with full path.
- **-1**: Indicating new column creation.
- **startrow**: Starting point of the row of the data insertion.

Theoretical example

Command:

```
SET table01[-1,0]
{
new record
new record
new record
new record
}
```

Output:

TABLE01	0	1	2	3	4
0	record	record	record	record	new record
1	record	record	record	record	new record
2	record	record	record	record	new record
3	record	record	record	record	new record

From:
<http://wiki.catdb.eu/> - The Catalog Wiki

Permanent link:
<http://wiki.catdb.eu/doku.php/set>

Last update: **2016/11/17 13:23**



