

## Re: Access query

---

*Source:* <http://www.tech-archive.net/Archive/Data/microsoft.public.data.odbc/2007-11/msg00006.html>

---

- *From:* Bob Hairgrove <[NoSpamPlease@xxxxxxx](mailto:NoSpamPlease@xxxxxxx)>
  - *Date:* Tue, 06 Nov 2007 22:42:49 +0100
- 

[This actually has nothing to do with ODBC, does it?]

On Mon, 5 Nov 2007 18:03:00 -0800, sath  
<[sath@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:sath@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx)> wrote:

I have a Accounts table in the access database. It has 60000 records. There are only three columns. Access added key , account ID and a flag. The account ID has duplicate values and it is not sorted. There are 10000 duplicates.

From the rest of your description I presume that the 'flag' column is Yes/No type? If so, you have a table where the only column providing the uniqueness of a row is the Access-generated AutoNumber column -- is that correct? What purpose does this table actually provide?

I want to update the flag to 'Y' for the duplicate values only for the first account ID.  
Say account ID has value of 35000 twice. I want to update the 35000 record Flag only once.

Also I want to get distinct account ID data with out sorting. If I try SELECT Distinct, it sorts the account ID which I do not want.

First of all, about your last question: A relational table has no concept of sorting; it is actually just a kind of bucket for storing the data. Only when the output of some query is generated will there be any sorting perceived. IOW, when you look at a table in table view, you are actually performing "SELECT \* FROM some\_table;" where the sort order of what you see is determined by whatever the primary key column(s) is (or are). If you sort on some other column by clicking on one of the column headers, you set an internal filter on the table view which changes the sort order. However, the underlying data might be stored in vastly different areas of your hard disk drive and you would never know it -- nor should you be concerned with it.

## Re: Access query

So — what is it you need to do with "SELECT DISTINCT"? If you really want to get a random ordering, you would have to add an additional column to your table, fill it with random values, and have the query output sorted on that column. But in order to perform "SELECT DISTINCT" at all, every database in existence would first need to sort the data on that column by doing a full index or table scan ... how else would you do it? It is really the most efficient way (perhaps the only way?) to select distinct rows.

Assuming that your Access-generated AutoNumber key should ideally correspond to the order in time in which a row was inserted (in real life it might not, depending on whether the database was ever compacted or not, or whether someone ran an insert query with values for that column which didn't already exist...), then in order to fetch the "first" of each duplicate account ID I would use:

```
SELECT account_id, MIN(key_col) AS first_row_key  
FROM your_table GROUP BY account_id;
```

Of course, you should change the names I have used to reflect your actual table and column names.

Save this as a new query. Next, you would create another select query by joining the original table to the new query on "key\_col" (or whatever you are calling it). It should be an inner join, not left or right join. Make sure that all columns of the original table are in the query output. After you are satisfied that the correct rows are being fetched, change this query to an update query and update the flag column to "Y".

Now, what was your ODBC question...?

--

Bob Hairgrove  
NoSpamPlease@xxxxxxxxx

.