# Show me the Autonumber!

*by Manohar Kamath*
*Feb 15, 1998*

You might have come across this situation; You want to add a user to the database and return an ID to the user for future reference. So you create an Autonumber field in the Access table, which increments each time you add a user.

All is fine until you add a user through the slick ASP page you created. That's when you run into this problem - how to retrieve this ID?

There are several things you can do (and believe me, I have tried them all in my evolution as an ASP developer!)

1. Use the E-mail address of the user as a key. Once you added the user to the table, query the table on e-mail field to retrieve the whole record. Now you can access this ID
2. Don't use the Autonumber at all! Use an Application variable to hold a number, which is incremented every time you want to add a user. Before you add a user, lock this variable (Using Application.Lock), get the value, unlock the variable, and use this as an ID.

In both the cases above, you have resorted to some crude ways of doing this. They work, in most cases, but either does not reflect the true ID or true way of doing it.

The solution is simple, really! Many use SQL statements to add an entry into a table. The INSERT clause is useful, especially if you have a dynamic SQL statement to construct.

And some manipulate the RecordSet object to add entries. If you are doing this, you are already half way through! Just follow the steps to retrieve the Autonumber field from your tables.

## Step 1: Create a RecordSet object

Create a RecordSet object using the Server.CreateObject method. This is the only object we will use in this example.

```
' Include the VBScript constant file
<!-- #include file="adovbs.inc" -->
Set Rs = Server.CreateObject("ADODB.RecordSet")
```

## Step 2: Open the table

Let us say we add users to a table tUsers. The fields in this table are - UserName, UserEmail and, of course, a Autonumber field called UserID. To open the table we need the

connection string. Let us use a system DSN called "AutoExample" which we need to create using ODBC manager.

We use the Open method of the RecordSet object to open the table. This method takes a few parameters (that we don't discuss here) that describe the manner in which we need to open the table.

```
' Open the table. You need the table name and the
' connection string/System DSN
Rs.Open "tUsers", "AutoExample", adOpenKeySet, adLockPessimistic,
adCmdTable
```

### Step 3: Add a new record

In our case, add a new user. This is achieved using the AddNew method in the RecordSet object.

```
Rs.AddNew
```

### Step 4: Set the field values for the table

As we mentioned earlier, the table contains UserName and UserEmail fields, apart from the UserID field. The UserID field is of type Autonumber, and can not be edited and is read-only.

For our example, we add the user via the userform.htm. This form has two input fields to enter the name and email. Those values are captured in this ASP page.

```
Rs("UserName") = Request.Form("USERNAME")
Rs("UserEmail") = Request.Form("USEREMAIL")
```

### Step 5: Update the record

This is simple.

```
Rs.Update
```

This is the method that actually "adds" the record into the physical database. The AddNew method just lets the user add a record

Now, don't panic - "But I wanted my autonumber!!!!" That step is next.

### Step 6: Retrieve the UserID

This is the trickiest part. You can get the Autonumber only after you have "Update"d the record. This is the last thing we do with this record. So, close it.

```
lUserID = Rs("UserID")
Rs.Close Set
Rs = Nothing
```

## Step 7: Display this ID (or do whatever you want)

This is a trivial step. You can do anything you want with the ID such as

- Display the ID
- E-mail the ID to the user
- Create another ID using this one

And so on.

That solves our problem of retrieving the Autonumber from tables. Remember, this works only in Access databases. Retrieving the Autonumber fields from SQL Server requires a different approach, which I don't know!

Installing the working example:

1. Extract all the files to the same directory. Make sure this directory is in your web path
2. Create a system DSN "AutoExample" using the ODBC manager in the control panel. Point the DSN to the user.mdb that comes with the sample