

## Recordsets

The *Recordsets* collection describes instances of all recordsets that are currently open in relation with a given **database object**.

tags:

[Collections](#)

### Syntax

```
database.Recordsets()
```

```
database.Recordsets(index)
```

```
database.Recordsets(recordsetname)
```

Object	Type	Argument	Type	Returned value
database	<b>Database object</b>	absent		A <b>Collection</b> object
		index	integer long	A <b>Recordset</b> object corresponding to the index-th item in the ~Recordsets() collection. The 1st recordset is Recordsets(0), the 2nd is Recordsets(1) and so on ... The last one is Recordsets.Count - 1.
		recordsetname	string	A <b>Recordset</b> object having the argument as name. The argument is NOT case-sensitive.

### Remarks

- The *recordsetname* argument is not case sensitive.
- The name of a recordset is not chosen by the user or the programmer. It is defined as a unique name by Access2Base. As a consequence the use of the syntax based on the recordsetname requires the name to be kept somewhere to allow the retrieval of the recordset object. See the example.

### Error messages

Argument nr. 1 [Value = '...'] is invalid
Out of array range or incorrect array size for collection Recordsets()
Recordset '...' not found

### See also ...

[QueryDefs](#)[TableDefs](#)

### Example

A recordset is created somewhere

```
Global sRSName As String
'-----
Dim oRecordset As Object
    Set oRecordset = Application.CurrentDb().OpenRecordset("SELECT * FROM [PRODUCTS]")
    sRSName = oRecordset.Name
```

... then, elsewhere, for instance in an event ... next construction is valid:

```
Dim oRecordset As Object
    Set oRecordset = Application.CurrentDb().Recordsets(sRSName)
    '...
```

Bookmark this page » » [Recordsets](#)