

# Report Object Reference Guide

**NetIQ AppManager**

Version 6.0



## Legal Notice

THIS DOCUMENT AND THE SOFTWARE DESCRIBED IN THIS DOCUMENT ARE FURNISHED UNDER AND ARE SUBJECT TO THE TERMS OF A LICENSE AGREEMENT OR A NON-DISCLOSURE AGREEMENT. EXCEPT AS EXPRESSLY SET FORTH IN SUCH LICENSE AGREEMENT OR NON-DISCLOSURE AGREEMENT, NETIQ CORPORATION PROVIDES THIS DOCUMENT AND THE SOFTWARE DESCRIBED IN THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. SOME STATES DO NOT ALLOW DISCLAIMERS OF EXPRESS OR IMPLIED WARRANTIES IN CERTAIN TRANSACTIONS; THEREFORE, THIS STATEMENT MAY NOT APPLY TO YOU.

This document and the software described in this document may not be lent, sold, or given away without the prior written permission of NetIQ Corporation, except as otherwise permitted by law. Except as expressly set forth in such license agreement or non-disclosure agreement, no part of this document or the software described in this document may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, or otherwise, without the prior written consent of NetIQ Corporation. Some companies, names, and data in this document are used for illustration purposes and may not represent real companies, individuals, or data.

This document could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein. These changes may be incorporated in new editions of this document. NetIQ Corporation may make improvements in or changes to the software described in this document at any time.

**Copyright © 1995-2004 NetIQ Corporation, all rights reserved.**

U.S. Government Restricted Rights: If the software and documentation are being acquired by or on behalf of the U.S. Government or by a U.S. Government prime contractor or subcontractor (at any tier), in accordance with 48 C.F.R. 227.7202-4 (for Department of Defense (DOD) acquisitions) and 48 C.F.R. 2.101 and 12.212 (for non-DOD acquisitions), the government's rights in the software and documentation, including its rights to use, modify, reproduce, release, perform, display or disclose the software or documentation, will be subject in all respects to the commercial license rights and restrictions provided in the license agreement.

ActiveAgent, ActiveAnalytics, ActiveKnowledge, ActiveReporting, ADcheck, AppAnalyzer, Application Scanner, AppManager, AuditTrack, AutoSync, Chariot, ClusterTrends, CommerceTrends, Configuration Assessor, ConfigurationManager, the cube logo design, DBTrends, DiagnosticManager, Directory and Resource Administrator, Directory Security Administrator, Domain Migration Administrator, End2End, Exchange Administrator, Exchange Migrator, Extended Management Pack, FastTrends, File Security Administrator, Firewall Appliance Analyzer, Firewall Reporting Center, Firewall Suite, Ganymede, the Ganymede logo, Ganymede Software, Group Policy Administrator, Intergreat, Knowledge Scripts, Log Analyzer, Migrate.Monitor.Manage, Mission Critical Software, Mission Critical Software for E-Business, the Mission Critical Software logo, MP3check, NetIQ, the NetIQ logo, the NetIQ Partner Network design, NetWare Migrator, OnePoint, the OnePoint logo, Operations Manager, Qcheck, RecoveryManager, Security Analyzer, Security Manager, Server Consolidator, SQLcheck, VigilEnt, Visitor Mean Business, Visitor Relationship Management, Vivinet, W logo, WebTrends, WebTrends Analysis Suite, WebTrends Data Collection Server, WebTrends for Content Management Systems, WebTrends Intelligence Suite, WebTrends Live, WebTrends Network, WebTrends OLAP Manager, WebTrends Report Designer, WebTrends Reporting Center, WebTrends Warehouse, Work Smarter, WWWorld, and XMP are trademarks or registered trademarks of NetIQ Corporation or its subsidiaries in the United States and other jurisdictions.

All other company and product names mentioned are used only for identification purposes and may be trademarks or registered trademarks of their respective companies.

---

# Contents

<b>Chapter 1</b>	<b>Report object reference</b> .....	<b>7</b>
	Introduction. ....	7
	About ADODataSource .....	8
	About the filter objects .....	8
	About the Report object .....	9
<b>Chapter 2</b>	<b>ADODataSource object</b> .....	<b>11</b>
	ADODataSource properties .....	11
	AddParameter .....	11
	Command .....	12
	Provider .....	12
	Recordset .....	13
	Timeout .....	13
	ADODataSource methods .....	14
	Execute .....	14
	Initialize .....	15
	Prepare .....	15
<b>Chapter 3</b>	<b>Datafilter objects</b> .....	<b>17</b>
	CROSSTAB object .....	18
	CROSSTAB code sample .....	19
	HISTOGRAM object .....	21
	HISTOGRAM properties .....	21
	HighLimit .....	21

LowLimit . . . . .	22
Mode . . . . .	22
Number of Bins . . . . .	23
HISTOGRAM methods . . . . .	23
AutoRange . . . . .	24
HISTOGRAM code sample . . . . .	24
PERCENT object . . . . .	26
STATISTICS object . . . . .	27
STATISTICS properties . . . . .	27
Output . . . . .	27
OutputMode . . . . .	28
Grouping . . . . .	29
TIMEFILTER . . . . .	31
TIMEFILTER properties . . . . .	31
DayOfWeekFilter . . . . .	32
Duration . . . . .	33
MaxPointsPerChart . . . . .	34
MonthAggregate . . . . .	35
TimePeriodPerPoint . . . . .	35
TIMEFILTER methods . . . . .	36
AddExclusionDateFilter . . . . .	36
AddTimeOfDayFilter . . . . .	36
RemoveColumn . . . . .	37
Description . . . . .	37
Parameters . . . . .	37
Example . . . . .	37
SetDaysOfTheWeekFilter . . . . .	37
TIMEFILTER code sample . . . . .	38

<b>Chapter 4</b>	<b>Report object</b>	<b>39</b>
	Report properties	39
	ADODataSource	42
	Chart	43
	ChartColorScheme	43
	ChartDataLabelAngle	44
	ChartDuration	44
	ChartPerColumn	45
	ChartStyle	46
	ChartThresBlue	47
	ChartThresGreen	48
	ChartThresRed	48
	ChartThresValue	49
	ChartTitle	50
	ChartTransparency	50
	ChartViewMode	51
	ChartXRot	51
	ChartXSize	52
	ChartYAxisAutoRange	52
	ChartYAxisBottomLabel	53
	ChartYAxisLabel	53
	ChartYAxisMax	54
	ChartYAxisMin	54
	ChartYAxisTopLabel	55
	ChartYRot	55
	ChartYSize	56
	ChildColumnFormat	56
	ChildSubtitle1Prefix	57

ChildSubtitle2 . . . . .	58
ColumnAlignCenter. . . . .	59
ColumnAlignLeft . . . . .	59
ColumnFormat . . . . .	60
ColumnHeadColor . . . . .	61
ColumnLineType. . . . .	61
DataSource. . . . .	62
DayOfWeekChartStart . . . . .	63
DocName . . . . .	64
EndDateTime . . . . .	64
EndDay. . . . .	65
Filter . . . . .	65
FolderPrefix . . . . .	66
FolderType . . . . .	67
FuzzTime . . . . .	67
FuzzTimeCurrent . . . . .	68
FuzzTimeType . . . . .	69
FuzzTimeUnitTime . . . . .	70
HorizontalChart. . . . .	70
Hyperlink. . . . .	71
IndexAuthor . . . . .	71
IndexCompany . . . . .	72
IndexComponent . . . . .	72
IndexCustomProp1 . . . . .	73
IndexCustomProp2. . . . .	74
IndexDescriptions . . . . .	74
IndexExpires. . . . .	75
IndexReportTitle . . . . .	75

InitialCatalog . . . . .	76
JobID . . . . .	77
Layout . . . . .	77
LayoutFolder . . . . .	78
PathAndFileName . . . . .	78
RowDetailColor . . . . .	79
RowStyle . . . . .	79
SeriesSeconds. . . . .	80
StartDateTime . . . . .	81
StartDay. . . . .	81
SubTitle1 . . . . .	82
SubTitle2 . . . . .	82
Table . . . . .	83
TextStyle . . . . .	83
VerticalSpace. . . . .	84
Report methods . . . . .	84
CreateTable . . . . .	85
FinishChart. . . . .	86
Initialize . . . . .	86
MakeChart . . . . .	87
MakeChartAndTable . . . . .	88
MakeDataTable . . . . .	89
MakeDrillDownReportV1 . . . . .	90
MakeDrillDownReportV2 . . . . .	91
TableHeader . . . . .	92
TableRow. . . . .	92
TableText. . . . .	93
Write properties . . . . .	94

**Index..... 1**



# Report object reference

This chapter covers the Report object, ADODataSource object, and filter objects properties and methods.

## Introduction

NetIQ provides several new COM objects to help you generate reports:

This object	Is used to
ADODataSource	Retrieve data from a SQL database and create a recordset of that data.
Crosstab Histogram Statistics TimeFilter Percent	Manipulate data in the recordset returned by the ADODataSource object.
Report	Format the data returned by the filter objects and generate an HTML report.

You can use these objects in the context of the NetIQ report template format to create your own report templates for use in the AppMangaer Operator Console. You can also use them in scripts that run from the command line.

The purpose of this document is to provide you with the properties of each object and the functions used with each object.

## About ADODataSource

The ADODataSource object uses the following ADO objects:

<b>This object</b>	<b>Is used to</b>
Connection	Connect to the AppManager repository.
Command	Query the AppManager repository, and return data in a Recordset object.
Recordset	Hold the data returned by the Command object.

The ADODataSource object provides the processes used to connect to a SQL database and execute stored procedures. You must pass a connection string, stored procedure, and stored procedure parameters to the ADODataSource object.

The ADODataSource object passes the connection string to the Connection object, and the parameters of the stored procedure to the Command object. Upon successful execution of the stored procedure, ADODataSource creates a Recordset object containing the data retrieved from the SQL database.

## About the filter objects

The filter objects are used to manipulate the data in the Recordset object created by ADODataSource:

<b>This filter object</b>	<b>Manipulates data in this manner</b>
Crosstab	Changes row-oriented data to column-oriented data.
Histogram	Summarizes each column of numeric data. For example, the number of occurrences of each value in a column.
Percent	For two columns of numeric data, divides the values in the first column by the corresponding values in the second column, and returns the results as percentages. For example, column one is the number of seconds a service was detected as running (150), column two is the number of seconds the service was monitored (300). $150/300 = 50\%$ (the service was running 50% of the time it was being monitored).

<b>This filter object</b>	<b>Manipulates data in this manner</b>
Statistics	Computes summary statistics on groups of data. For example, the average value of a data set.
TimeFilter	<ol style="list-style-type: none"> <li>1 Filters out unwanted data by time period. For example, excludes data collected from 12:00 AM to 6:00 AM.</li> <li>2 Organizes data by time period. The organized data is used by the statistics filter. For example, if data is collected every five minutes and the collected data is organized by hour, there are twelve values per time period; those twelve values per hour are processed by the statistics filter to provide an average hourly value.</li> <li>3 Expresses all data in a time range by a specific number of values. For example, data collected over a period of two days is expressed by twenty-four values, providing an average value for each two-hour period.</li> </ol>

You can use a sequence of filter objects, for example, the time filter and then the statistics filter.

The manipulated data is stored in a new recordset that is passed to the Report object.

## About the Report object

The Report object uses the following objects:

<b>This object</b>	<b>Is used to</b>
AMLayout	Define the location and graphical properties of a report.
ChartGenerator	Generate data charts in a report.
Table	Generate tabular data in a report.

The Report object provides the processes used to generate HTML reports based on the data in the recordset created by the filter objects.

The Report object uses the AMLayout object to specify the location of a finished report, to include images and text in a report, and to reference style sheets.

The ChartGenerator object is used to create chart images of the data passed to the Report object. ChartGenerator renders the images based on default or user-defined parameters such as data series style (area, line, bar, etc.) and 3-D rotation of the chart grid. ChartGenerator creates an appropriate number of images to illustrate the data in a report, saves the images to disk, and attaches the images to the HTML document.

The Table object is used to generate tables of the data passed to the Report object.

# ADODataSource object

Use the ADODataSource object to:

- Connect to a SQL database
- Run stored procedures to query a SQL database
- Create a Recordset object containing the result of the query

This section covers the following topics:

- [ADODataSource properties](#)
- [ADODataSource methods](#)

## ADODataSource properties

The ADODataSource object has the following properties:

Property	Description
<a href="#">AddParameter</a>	Adds parameters for the stored procedure. The code determines the subtype for the variant.
<a href="#">Command</a>	Creates the command object to execute the stored procedure.
<a href="#">Provider</a>	Name of the data provider for the ADO data source.
<a href="#">Recordset</a>	Current recordset.
<a href="#">Timeout</a>	Number of seconds to wait for the stored procedure to complete.

## AddParameter

### *Description*

Adds parameters for the stored procedure. The code determines the subtype for the variant.

**Access**

Write-only.

**Parameter**

String name of the parameter.

**Data type**

Variant.

**Default value****Syntax**

```
DS.AddParameter("@KSname") = "NT_CpuResource"
```

## Command

---

**Description**

Object that is created to execute stored procedures.

**Access**

Read-only.

**Parameter****Data type**

Command.

**Default value****Syntax**

## Provider

---

**Description**

Name of the data for the ADO data source.

***Access***

Write-only.

***Parameter******Data type***

String.

***Default value***

SQLOLEDB.

***Syntax*****Recordset**

---

***Description***

Current recordset.

***Access***

Write-only.

***Parameter******Data type***

Recordset.

***Default value******Syntax*****Timeout**

---

***Description***

Number of seconds to wait for the stored procedure to complete.

**Access**

Write-only.

**Parameter****Data type**

Integer.

**Default value**

0 (no timeout)

**Syntax****ADODataSource methods**

The ADODataSource object has the following methods:

Method	Description
<a href="#">Execute</a>	Executes the stored procedure.
<a href="#">Initialize</a>	Initializes the ADODataSource object.
<a href="#">Prepare</a>	Prepares the data source with the name of the stored procedure.

**Execute****Description**

Executes the stored procedure.

**Return**

- Type: Bool
- Value: True for success, false for failure

**Parameter****Syntax**

Flag = DS.Execute



### ***Remarks***

If the return value is true, the recordset becomes valid after this method executes.

## **Initialize**

---

### ***Description***

Initializes the ADODatasource object.

### ***Return***

- Type: Bool
- Value: True for success false for failure

### ***Parameter***

- Data type: ReportObject
- Parameter name: pReportObject

### ***Syntax***

### ***Remarks***

## **Prepare**

---

### ***Description***

Prepares the data source with the name of the stored procedure.

### ***Return***

- Type: Bool
- Value: True for success, false for failure

### ***Parameter***

- Data type: String
- Parameter name: <StoredProcedureName>

**Syntax**

```
Flag = DS.Prepare("@name of stored procedure")
```

**Remarks**

The parameters for the stored procedure are added afterward.

The previous parameter list will be cleared. The previous connection will be maintained.

# Datafilter objects

Use the data filter objects to manipulate data in ADO recordsets. Client-side recordsets are required.

Each filter object takes an existing recordset as input, and generates a new recordset as output.

The data filters do not require the Report Object, and will work with any client-side recordset.

This section covers the following topics:

- [CROSSTAB object](#)
- [HISTOGRAM object](#)
- [PERCENT object](#)
- [STATISTICS object](#)
- [TIMEFILTER](#)

## CROSSTAB object

The CROSSTAB object changes a row-oriented recordset to a column-oriented recordset. The original recordset must have at least three columns of data. The contents of the next-to-last column of data are used to identify columns of the new recordset.

For example:

Original recordset:

Date	Computer	CPU Usage
1/01/02	Server 01	50%
1/01/02	Server 02	75%
1/01/02	Server 03	100%
2/01/02	Server 01	65%
2/01/02	Server 02	90%
2/01/02	Server 03	80%

New recordset:

Date	Server 01	Server 02	Server 03
1/01/02	50%	75%	100%
2/01/02	65%	90%	80%

There is no limit to the number of columns in the original recordset, and the filter does not limit the number of columns in the recordset it generates. ADO, however, does limit the number of columns in the generated recordset to 256.

The contents of the generated recordset are from the last column of data in the original recordset. The data in the original recordset should be sorted and unique for all but the last column of data.

This section covers the following topic:

- [CROSSTAB code sample](#)

## CROSSTAB code sample

```
Set recordSet = CreateObject("ADODB.RecordSet")

recordSet.Fields.Append "A", 3
recordSet.Fields.Append "B", 3
recordSet.Fields.Append "C", 3

recordSet.Open

recordSet.AddNew
recordSet.Fields("A").Value = 10
recordSet.Fields("B").Value = 12
recordSet.Fields("C").Value = 60

recordSet.AddNew
recordSet.Fields("A").Value = 10
recordSet.Fields("B").Value = 1
recordSet.Fields("C").Value = 4

recordSet.AddNew
recordSet.Fields("A").Value = 10
recordSet.Fields("B").Value = 2
recordSet.Fields("C").Value = 10

recordSet.AddNew
recordSet.Fields("A").Value = 11
recordSet.Fields("B").Value = 12
recordSet.Fields("C").Value = 50

recordSet.AddNew
recordSet.Fields("A").Value = 1
recordSet.Fields("B").Value = 2
recordSet.Fields("C").Value = 12

recordSet.AddNew
recordSet.Fields("A").Value = 4
recordSet.Fields("b").Value = 12
recordSet.Fields("C").Value = 12

recordSet.Sort = "A"

Dim crosstabfilter
Set crosstabfilter = CreateObject("NETIQFILTERS.CROSSTAB")

crosstabfilter.Recordset = recordSet
```

```
Dim ReportObj
Set ReportObj =CreateObject("NETIQREPORT.ReportObject")

ReportObj.Initialize()
ReportObj.MakeDataTable( recordset )
ReportObj.MakeDataTable( crosstabfilter.Recordset )
```

## HISTOGRAM object

The HISTOGRAM object summarizes each column of numeric data in a recordset.

The number of occurrences of each value in range bins is computed. Range bins are determined by setting the object properties, or by analysis of the limits of the data. The resulting recordset can be output as raw data, the relative percentage of each bin, or the cumulative values of the bins. If multiple columns of data are analyzed, the data should be similar in scale.

This section covers the following topics:

- [HISTOGRAM properties](#)
- [HISTOGRAM methods](#)
- [HISTOGRAM code sample](#)

### HISTOGRAM properties

This object has the following properties:

Property	Description
<a href="#">HighLimit</a>	The high limit for the histogram bins.
<a href="#">LowLimit</a>	The low limit for the histogram bins.
<a href="#">Mode</a>	The type of data that is reported.
<a href="#">Number of Bins</a>	The number of bins to make for the histogram.

### HighLimit

#### *Description*

The high limit for the histogram bins. If there are any values above this limit, a special bin is made to count them.

#### *Access*

Write-only.

***Data type***

Double.

***Valid strings***

***Default value***

## **LowLimit**

---

***Description***

The low limit for the histogram bins. If there are any values below this limit, a special bin is made to count them.

***Access***

Write-only.

***Data type***

Double.

***Valid strings***

***Default value***

## **Mode**

---

***Description***

The type of data that is reported.

***Access***

Write-only.

***Data type***

String.

***Valid strings***

- N



- Count
- Cum
- Cumulative
- Percent

***Default value***

Percent.

## Number of Bins

---

***Description***

The number of bins to make for the histogram. If the high and low limits are applied, two extra bins may be created.

***Access***

Write-only.

***Data type***

Integer.

***Valid strings***

***Default value***

10.

## HISTOGRAM methods

This object has the following method:

Method	Description
<a href="#">AutoRange</a>	Resets the histogram object to the default mode.

## AutoRange

---

### *Description*

This function resets the histogram object to the default mode. The low and high limits are determined by the source data.

### **HISTOGRAM code sample**

```
option explicit

Dim recordset
Set recordSet = CreateObject("ADODB.RecordSet")

recordset.Fields.Append "txt", 8
recordSet.Fields.Append "num", 5
recordSet.Fields.Append "B", 5
recordSET.fields.append "sam",5

recordSet.Open

Dim dt
dt = CDATE("1/1/2000")

dim j
for j = 1 to 10000
    recordSet.AddNew
    recordset.fields("txt").Value = "nothing"
    recordSet.Fields("num").Value = rnd
    recordSet.Fields("B").Value = rnd + rnd
    recordSet.Fields("sam").Value = rnd*rnd
next

Wscript.echo("Record set completed")

DIM filter
Set filter = CreateObject("NETIQFILTERS.HISTOGRAM")

recordSet.MoveFirst

filter.lowLimit = 0
filter.highLimit = 2

filter.NumberOfBins = 20
```

```
filter.Mode = "CUMULATIVE"

filter.Recordset = recordSet

Dim ReportObj
Set ReportObj = CreateObject("NETIQREPORT.ReportObject")

ReportObj.Initialize()

ReportObj.ChartYSize = 200
ReportObj.CHARTSTYLE = "LINE"
ReportObj.MakeChartAndTable( filter.Recordset)
```

## PERCENT object

The PERCENT object provides percentage values based on two adjacent columns of data.

For two columns of numeric data, divides the values in the first column by the corresponding values in the second column, and returns the results as percentages. For example, column one is the number of seconds a service was detected as running (150), column two is the number of seconds the service was monitored (300).  $150/300 = 50\%$  (the service was running 50% of the time it was being monitored).

## STATISTICS object

The STATISTICS object is used to compute summary statistics for groups of data. Non-numeric data is used to identify groups. The resulting recordset has one row of data for each group.

This section covers the following topic:

- [STATISTICS properties](#)

### STATISTICS properties

This object has the following properties:

Property	Description
<a href="#">Output</a>	Determines the type of data reported.
<a href="#">OutputMode</a>	Determines by bit mask the type of data reported.
<a href="#">Grouping</a>	Determines whether the grouping statistics are computed.

### Output

#### **Description**

Determines the type of data reported.

#### **Access**

Write-only.

#### **Parameter**

String statType (the type of statistics that are enabled).

Valid statistic types:

- All
- Average, AVG
- Box
- Change

- Close
- Count, N
- Maximum, MAX
- Minimum, MIN
- Open
- Range
- 3sigma
- Sigma, SIG
- Sum

***Data type***

Bool.

***Default value***

***Remarks***

***Example***

```
Filter.Output("Avg") = true
```

## OutputMode

---

***Description***

Determines by bit mask the type of data reported.

***Access***

Write-only.

***Bit mask value***

- statAverage = 1
- statSigma = 2
- statCount = 4
- statSum = 8

- stat3Sigma = 16
- statMinimum = 32
- statBox = 64
- statMaximum = 128
- statOpen = 256
- statClose = 512
- statChange = 1024
- statRange = 2048
- statEverything = 0xffffffff

***Parameter***

***Data type***

Long.

***Default value***

***Remarks***

## Grouping

---

***Description***

Determines whether the grouping statistics are computed.

***Access***

Write-only.

***Parameter***

***Data type***

Bool.

***Default value***

True.

***Remarks***

If the value is False, the statistics are computed for all data. One row of data is output for each data column.



# TIMEFILTER

The TIMEFILTER object performs two functions.

The first is to filter out data that does not match the day-of-the-week and the hours-of-the-day filters. The default settings are to use all data.

The second is to modify the date/time value, rounding the value down. For example, if you are generating a report of average hourly values, all values collected from 8:00 AM to 8:59:59 AM, would be considered collected at 8:00 AM.

The TIMEFILTER object operates in three different modes:

This mode	Is set by
Auto	Setting the value of the <code>MaxPointsPerChart</code> property.
Fixed-duration	Setting the value of the <code>MaxPointsPerChart</code> property <b>AND</b> Setting the value of the <code>Duration</code> property
Fixed-time-period	Setting the value of the <code>TimePeriodPerPoint</code> property.

This section covers the following topics:

- [TIMEFILTER properties](#)
- [TIMEFILTER methods](#)
- [TIMEFILTER code sample](#)

## TIMEFILTER properties

This object has the following properties:

Property	Description
<a href="#">DayOfWeekFilter</a>	Determines whether data from a particular weekday is used in the report.
<a href="#">Duration</a>	Determines the amount of data displayed in a chart.
<a href="#">MaxPointsPerChart</a>	Determines the maximum number of points to which data is aggregated for the specified time range.

Property	Description
<a href="#">MonthAggregate</a>	Aggregates the time by month.
<a href="#">TimePeriodPerPoint</a>	Determines the time range by which data is grouped.

## DayOfWeekFilter

### **Description**

Determines whether data from a particular weekday is used in the report.

### **Access**

Write-only.

### **Parameter**

String name of the weekday.

Valid weekdays:

- Sunday
- Monday
- Tuesday
- Wednesday
- Thursday
- Friday
- Saturday

### **Data type**

Bool.

### **Default value**

### **Example**

```
Filter.DayOfWeekFilter("Monday")=True
```

***Remarks***

**Duration**

---

***Description***

Determines the amount of data displayed in a chart.

***Access***

Write-only.

***Parameter***

Number of points in chart, or string name of time range represented in chart.

Valid strings:

- Hourly
- Shift
- Daily
- Weekly
- Biweekly
- 4Weeks
- Monthly
- Quarterly
- Yearly
- Auto

***Data type***

Integer or string.

***Default value***

Auto.

### ***Example***

### ***Remarks***

The string names represent the following time ranges:

String name	Time range
Hourly	1 hour
Shift	8 hours
Daily	1 day
Weekly	7 days
Biweekly	14 days
4Weeks	28 days
Monthly	31 days
Quarterly	93 days
Yearly	365 days

## **MaxPointsPerChart**

---

### ***Description***

Determines the maximum number of points to which data is aggregated for the specified time range. The time range is determined by the input data, or the value of the Duration property.

### ***Access***

Write-only.

### ***Parameter***

### ***Data type***

Integer.

### ***Default value***

200.

### ***Example***

### ***Remarks***

## **MonthAggregate**

---

### ***Description***

Aggregates the time by month. If set to TRUE, the time values are set to the first day of the month (for example, March 1, 2002).

### ***Access***

Write-only.

### ***Data type***

BOOL.

### ***Default value***

FALSE.

## **TimePeriodPerPoint**

---

### ***Description***

Determines the time range by which data is grouped.

### ***Access***

- Write: Overrides the computed time period per point. Allows you to directly set the value. A value of 0 resets this to an automatic setting.
- Read: The value of the property.

### ***Parameter***

### ***Data type***

Integer (time in seconds).

**Default value**

**Example**

**Remarks**

## TIMEFILTER methods

This object has the following methods:

Method	Description
<a href="#">AddExclusionDateFilter</a>	Filters out data in the specified time range.
<a href="#">AddTimeOfDayFilter</a>	Adds the valid time of weekdays.
<a href="#">RemoveColumn</a>	Specifies that a source column can be removed from the resulting recordset.
<a href="#">SetDaysOfTheWeekFilter</a>	Sets the value for each weekday to True or False.

### AddExclusionDateFilter

**Description**

Filters out data in the specified time range.

**Parameters**

- date lowDate
- date highDate

**Example**

### AddTimeOfDayFilter

**Description**

Adds the valid time of weekdays.

**Parameters**

- date lowTime
- date highTime

### ***Example***

## **RemoveColumn**

---

### **Description**

---

Specifies that a source column can be removed from the resulting recordset. This can include the date/time column.

### **Parameters**

---

String <column name to be removed>

### **Example**

---

## **SetDaysOfTheWeekFilter**

---

### ***Description***

Sets the value for each weekday to True or False.

### ***Parameters***

- Bool Sunday
- Bool Monday
- Bool Tuesday
- Bool Wednesday
- Bool Thursday
- Bool Friday
- Bool Saturday

### ***Example***

```
Filter.SetDaysOfTheWeekFilter(false, true, true, true, _  
true, true, false)
```

## TIMEFILTER code sample

```
Dim ReportObj
Set ReportObj = CreateObject("NETIQREPORT.ReportObject")

ReportObj.Initialize()

Set recordSet = CreateObject("ADODB.RecordSet")

recordSet.Fields.Append "fld2:", 135
recordSet.Fields.Append "fld3", 5

Dim dt
dt = #8/10/2001

Dim j
for j = 1 to 10000
    recordSet.AddNew
    recordSet.Fields("fld2").Value = dt + (j/96.0) + .001
    recordSet.Fields("fld3").Value = rnd*rnd*100
next

Dim filter
Set filter = CreateObject("NETIQFILTERS.TIMEFILTER")
filter.MaxPointsPerChart = 50
`filter .duration = "MONTHLY"

filter.TimePeriodPerPoint = 3600*24
filter.AddTimeOfDayFilter #3:00 pm#, #4:00 pm#
filter.AddTimeOfDayFilter #1:00 am#, #6:00 am#
filter.AddExclusionDateFilter #9/5/2001#, #9/15/2001#
filter.SetDaysOfTheWeekFilter 0,1,1,1,1,1,0
filter.recordSet = recordSet

Dim filterStats
Set filterStats = CreateObject("NETIQFILTERS.STATISTICS")

filterStats.Output("AVG") = true
filterStats.Output("MAX") =true
filterStats.Output("MIN") = true
filterStats.recordSet = filter.recordSet
ReportObj.CHARTSTYLE = "Line"
ReportObj.MakeChartAndTable(filterStats.recordSet)
```



# Report object

Use the Report object to:

- Specify the location of a finished report
- Generate data charts for a report
- Generate images of data charts
- Attach data chart images to HTML document
- Generate the data table for a report

This section covers the following topics:

- [Report properties](#)
- [Report methods](#)
- [Write properties](#)

## Report properties

The Report object has the following properties:

Property	Description
<a href="#">ADODataSource</a>	Returns a reference to an ADODataSource object.
<a href="#">Chart</a>	Returns a reference to a ChartGenerator object.
<a href="#">ChartColorScheme</a>	Sets the name of the color scheme used for the chart.
<a href="#">ChartDataLabelAngle</a>	Sets the angle at which X-axis labels are written.
<a href="#">ChartDuration</a>	Sets a chart to a specific number of points or a specific time frame.
<a href="#">ChartPerColumn</a>	Controls whether a separate chart is created for each column in a recordset.
<a href="#">ChartStyle</a>	Sets the graphical style of a chart.

Property	Description
<a href="#">ChartThresBlue</a>	Sets the blue value for the color of the threshold indicator.
<a href="#">ChartThresGreen</a>	Sets the green value for the color of the threshold indicator.
<a href="#">ChartThresRed</a>	Sets the red value for the color of the threshold indicator.
<a href="#">ChartThresValue</a>	Determines whether a high threshold indicator is added to the chart.
<a href="#">ChartTitle</a>	Sets the title of the chart.
<a href="#">ChartTransparency</a>	Determines whether all series in a chart are displayed as partially transparent.
<a href="#">ChartViewMode</a>	Sets the view mode (perspective or orthographic) for the chart.
<a href="#">ChartXRot</a>	Sets the degree of X-axis rotation of the chart.
<a href="#">ChartXSize</a>	Sets the X-axis size (in pixels) of the chart image.
<a href="#">ChartYRot</a>	Sets the degree of Y-axis rotation of the chart.
<a href="#">ChartYAxisAutoRange</a>	Turns Y-axis auto range on or off.
<a href="#">ChartYAxisBottomLabel</a>	Sets a label at the bottom of the Y-axis.
<a href="#">ChartYAxisLabel</a>	Sets a label on the Y-axis, typically used to indicate units of measure.
<a href="#">ChartYAxisMax</a>	Sets the Y-axis maximum scaled value.
<a href="#">ChartYAxisMin</a>	Sets the Y-axis minimum scaled value.
<a href="#">ChartYAxisTopLabel</a>	Sets a label at the top of the Y-axis.
<a href="#">ChartYSize</a>	Sets the Y-axis size (in pixels) of the chart image.
<a href="#">ChildColumnFormat</a>	Specifies formats for data tables in child reports created by the <code>MakeDrillDownReportv#</code> functions.
<a href="#">ChildSubtitle1Prefix</a>	Adds a subtitle to a linked page of a drill-down report.
<a href="#">ChildSubtitle2</a>	Adds a second subtitle to a linked page of a drill-down report.
<a href="#">ColumnAlignCenter</a>	Specifies the string used to center-align data in a table.
<a href="#">ColumnAlignLeft</a>	Specifies the string used to left-align data in a table.
<a href="#">ColumnFormat</a>	Specifies special formats for data tables.

Property	Description
<a href="#">ColumnHeadColor</a>	Specifies the string used to set the column header color.
<a href="#">ColumnLineType</a>	Sets the graphical style of a data series in a chart (the data series represents a column of data in a recordset).
<a href="#">DataSource</a>	Name of the SQL Server used for the database.
<a href="#">DayOfWeekChartStart</a>	Controls at what day the chart begins.
<a href="#">DocName</a>	Sets the name of the top-level HTML document in a report.
<a href="#">EndDateTime</a>	Returns the end date/time for a report.
<a href="#">EndDay</a>	Sets the end date for a report.
<a href="#">Filter</a>	Sets the value for which filter object to use.
<a href="#">FolderPrefix</a>	Sets the prefix used for the report output folder.
<a href="#">FolderType</a>	Sets the naming convention of the report output folder.
<a href="#">FuzzTime</a>	Sets the use of fuzzy time.
<a href="#">FuzzTimeCurrent</a>	Sets the start time of report job as the end date/time for data in report.
<a href="#">FuzzTimeType</a>	Sets the fuzzy time increment (hour, day, month, etc.).
<a href="#">FuzzTimeUnitTime</a>	Sets the number of fuzzy time increments.
<a href="#">HorizontalChart</a>	Changes the chart from vertical to horizontal mode.
<a href="#">Hyperlink</a>	Returns a hyperlink to a finished report.
<a href="#">IndexAuthor</a>	Sets the author name by which report is indexed.
<a href="#">IndexCompany</a>	Sets the company name by which report is indexed.
<a href="#">IndexComponent</a>	Sets the monitored component by which report is indexed.
<a href="#">IndexCustomProp1</a>	Sets the custom property by which report is indexed.
<a href="#">IndexCustomProp2</a>	Sets the custom property by which report is indexed.
<a href="#">IndexDescriptions</a>	Sets the description of report.
<a href="#">IndexExpires</a>	Sets the number of days before report expires.
<a href="#">IndexReportTitle</a>	Sets the text in title bar of report.
<a href="#">InitialCatalog</a>	Sets the name of the database to query for a report.
<a href="#">JobID</a>	Appends the job ID to the name of the folder where the report is generated.

Property	Description
<a href="#">Layout</a>	Provides access to the AMLayout object.
<a href="#">LayoutFolder</a>	Sets the name of the folder where a report is generated.
<a href="#">PathAndFileName</a>	Returns the path to a report, and the file name of a report.
<a href="#">RowDetailColor</a>	Sets the background color of the data table.
<a href="#">RowStyle</a>	Sets the size of table rows.
<a href="#">SeriesSeconds</a>	Controls how the tick marks used to plot date-time based data are plotted along the X-axis.
<a href="#">StartDateTime</a>	Returns the start date/time for a report. Used as a parameter when calling the stored procedure.
<a href="#">StartDay</a>	Sets the start date for a report.
<a href="#">SubTitle1</a>	Sets the first subtitle of a report.
<a href="#">SubTitle2</a>	Sets the second subtitle of a report.
<a href="#">Table</a>	Provides access to the Table object.
<a href="#">TextStyle</a>	Sets HTML text formatting for a report.
<a href="#">VerticalSpace</a>	Adds a line vertical space in a report.

## ADODataSource

### ***Description***

Returns a reference to the ADODataSource object. ADODataSource is created if needed.

### ***Access***

Read-only.

### ***Parameter***

### ***Data type***

ADODataSource.

***Valid strings***

***Default value***

***Example***

## **Chart**

---

***Description***

Returns a reference to the ChartGenerator object. ChartGenerator is created automatically.

***Access***

Read-only.

***Parameter***

***Data type***

ChartGenerator.

***Valid strings***

***Default value***

***Example***

## **ChartColorScheme**

---

***Description***

Sets the name of the color scheme used for the chart.

***Access***

Write-only.

***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

Default.

***Example***

## **ChartDataLabelAngle**

---

***Description***

Sets the angle at which X-axis data labels are written.

***Access***

Write-only.

***Parameter***

***Data type***

Integer

***Default value***

Default.

## **ChartDuration**

---

***Description***

The value of this property is used to control whether multiple charts are generated by the MakeChart, MakeChartAndTable, and MakeDrillDownReportV# methods. Setting the value to **Auto** generates a single chart for all data. Setting the value to a number limits each chart to that number of points. Once the set number of points is illustrated in a chart, a new chart is generated. If the value is set to one of the valid strings, a single chart is generated to represent that time frame.

**Access**

Write-only.

**Parameter****Data type**

String or number.

**Valid strings**

- Hourly
- Daily
- Weekly
- Biweekly
- 4weeks
- Monthly
- Quarterly
- Yearly
- Auto

**Default value**

Auto.

**Example****ChartPerColumn**

---

**Description**

The value of this property is used to control whether multiple charts are generated by the `MakeChart`, `MakeChartAndTable`, and `MakeDrillDownReportV#` methods. If the value is `true`, a chart is generated for each column of data in the recordset.

**Access**

Write-only.

**Parameter****Data type**

- Type: Bool
- Value: True or false
- Hourly
- Daily
- Weekly
- Biweekly
- 4weeks
- Monthly
- Quarterly
- Yearly
- Auto

**Default value**

False.

**Example****ChartStyle**

---

**Description**

Sets the graphical style of chart data streams. The value of this property is used by the MakeChart, MakeChartAndTable, and MakeDrillDownReportV# methods.

If the value `pie` is used, a separate chart is made for each row in the recordset. The data values in each column of the row are used to determine the segments of the pie chart.



**Access**

Write-only.

**Parameter****Data type**

String.

**Valid strings**

- Area
- AreaStacked
- Bar
- Line
- Pie
- Ribbon

**Default value**

Bar.

**Example****ChartThresBlue**

---

**Description**

Sets the blue value for the color of the threshold indicator.

Used only if ChartThresValue = 1.

**Access**

Write-only.

**Parameter****Data type**

Long.

***Valid strings***

***Default value***

0.

***Example***

## **ChartThresGreen**

---

***Description***

Sets the green value for the color of the threshold indicator.

Used only if ChartThresValue = 1.

***Access***

Write-only.

***Parameter***

***Data type***

Long.

***Valid strings***

***Default value***

0.

***Example***

## **ChartThresRed**

---

***Description***

Sets the red value for the color of the threshold indicator.

Used only if ChartThresValue = 1.

**Access**

Write-only.

**Parameter****Data type**

Long.

**Valid strings****Default value**

0.

**Example****ChartThresValue**

---

**Description**

Determines whether a high threshold indicator is added to the chart.

**Access**

Write-only.

**Parameter****Data type**

Long.

**Valid strings****Default value**

0.

***Example***

## **ChartTitle**

---

***Description***

Sets a title for the chart.

***Access***

Write-only.

***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

Chart Title.

***Example***

## **ChartTransparency**

---

***Description***

Sets whether all series in a chart are displayed as partially transparent.

***Access***

Write-only.

***Parameter***

***Data type***

Long.

***Valid strings***

***Default value***

0.

***Example***

## **ChartViewMode**

---

***Description***

Sets the view mode (perspective or orthographic) for the chart.

***Access***

Write-only.

***Parameter***

***Data type***

Long.

***Valid strings***

***Default value***

1.

***Example***

## **ChartXRot**

---

***Description***

Sets the degree of X-axis rotation of the chart.

***Access***

Write-only.

***Parameter***

***Data type***

Long.

***Valid strings***

***Default value***

0.

***Example***

## **ChartXSize**

---

***Description***

Sets the X-axis size (in pixels) of the chart image.

***Access***

Write-only.

***Parameter***

***Data type***

Long.

***Valid strings***

***Default value***

640.

***Example***

## **ChartYAxisAutoRange**

---

***Description***

Turns Y-axis auto range on or off.

**Access**

Write-only.

**Parameter****Data type**

BOOL.

**Default value**

True.

## ChartYAxisBottomLabel

---

**Description**

Sets a label at the bottom of the Y-axis.

**Access**

Write-only.

**Parameter****Data type**

String.

**Default value**

## ChartYAxisLabel

---

**Description**

Sets a label on the Y-axis, typically used to indicate units of measure.

**Access**

Write-only.

***Parameter***

***Data type***

String.

***Default value***

## **ChartYAxisMax**

---

***Description***

Sets the Y-axis maximum scaled value. Use in conjunction with ChartYAxisMin. Turns off ChartYAxisAutoRange.

***Access***

Write-only.

***Parameter***

***Data type***

Double.

***Default value***

## **ChartYAxisMin**

---

***Description***

Sets the Y-axis minimum scaled value. Use in conjunction with ChartYAxisMax. Turns off ChartYAxisAutoRange.

***Access***

Write-only.

***Parameter***

***Data type***

Double.



***Default value***

## **ChartYAxisTopLabel**

---

***Description***

Sets a label at the top of the Y-axis.

***Access***

Write-only.

***Parameter***

***Data type***

String.

***Default value***

## **ChartYRot**

---

***Description***

Sets the degree of Y-axis rotation of the chart.

***Access***

Write-only.

***Parameter***

***Data type***

Long.

***Valid strings***

***Default value***

0.

### ***Example***

## **ChartYSize**

---

### ***Description***

Sets the Y-axis size (in pixels) of the chart image.

### ***Access***

Write-only.

### ***Parameter***

### ***Data type***

Long.

### ***Valid strings***

### ***Default value***

480.

### ***Example***

## **ChildColumnFormat**

---

### ***Description***

Specifies formats for data tables in child reports created by the MakeDrillDownReportV# functions. Some data need special formatting. The user must specify the column number or column number range that uses the special formatting. The last column number can be greater than the actual number of columns.

### ***Access***

Write-only.

### ***Parameter***

<first column>, <last column>.

The first column can be expressed as the name of the column, or the index to the column.

### **Data type**

String or number.

### **Default value**

-1.

### **Valid values**

- percent -- multiplies the data by 100 and adds the % sign after the value.
- sec2hour -- converts a value in seconds to the output format **xxx hours, yyy minutes**.
- auto -- automatically formats the data
- number for fixed point -- C-style format specification where xxxx is the printf-style format specification for either a string or a double, depending on the type of data (for example, %.3f)

The column formats can be reset by assigning the property with no parameters or values.

### **Example**

```
ReportObj.ColumnFormat(4) = "sec2hour"  
ReportObj.ColumnFormat(5, 20) = 4  
ReportObj.ColumnFormat("xyz") = "percent"
```

## **ChildSubtitle1Prefix**

---

### **Description**

Adds a subtitle to a linked page of a drill-down report.

The value of this property is used by the `MakeDrillDownReportV#` method. The value from the summary table is appended to this value to create a subtitle. If this value is not defined, no subtitle is added to the linked page.

**Access**

Write-only.

**Parameter****Data type**

String.

**Valid strings****Default value****Example****ChildSubtitle2**

---

**Description**

Adds a second subtitle to a linked page of a drill-down report.

The value of this property is used by the `MakeDrillDownReportV#` method to add a subtitle after the one specified by `ChildSubtitle1Prefix`. If this value is not defined, the second subtitle is not added to the linked page.

**Access**

Write-only.

**Parameter****Data type**

String.

***Valid strings***

***Default value***

***Example***

## **ColumnAlignCenter**

---

***Description***

Specifies the string used to center-align data in a table.

***Access***

Write-only.

***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

style="TEXT-ALIGN: center"

***Example***

## **ColumnAlignLeft**

---

***Description***

Specifies the string used to left-align data in a table.

***Access***

Write-only.

**Parameter****Data type**

String.

**Valid strings****Default value**

style="TEXT-ALIGN: left"

**Example**

## ColumnFormat

---

**Description**

Sets special formatting for columns in data tables. The column number or the range of column numbers must be specified. The last column number in a range can be greater than the actual number of columns.

**Access**

Write-only.

**Parameter**

(<first column>, <last column>)=<property value>

For example:

```
ReportObj.ColumnFormat(5, 20)=4
```

**Data type**

String or number.

**Valid strings**

- Percent
- Sec2hour
- Auto

- <number>

**Percent** multiplies the data by 100 and adds the % sign after.

**Sec2hour** converts a value in seconds to XXX hours, YYY minutes.

**<number>** controls the number of significant digits printed.

***Default value***

style="TEXT-ALIGN: left"

***Example***

## ColumnHeadColor

---

***Description***

Specifies the string used to set the header color for a table.

***Access***

Write-only.

***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

class=TableHeader

***Example***

## ColumnLineType

---

***Description***

Sets the graphical style used to represent the data values in a column of a recordset. A separate graphical style can be used for each column of a recordset.

The value of this property is used by the MakeChart, MakeChartAndTable, and MakeDrillDownReportV# methods.

**Access**

Write-only.

**Parameter**

<column number>.

For example:

```
ReportObj.ColumnLineType(4) = 'Bar'
```

**Data type**

String.

**Valid strings**

- Area
- AreaStacked
- Bar
- Cylinder
- Line
- Ribbon

**Default value**

Bar.

**Example**

## **DataSource**

---

**Description**

Name of the SQL Server used for the database being queried.

**Access**

Read-/Write-only.



***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

***Example***

## **DayOfWeekChartStart**

---

***Description***

Controls at what day the chart begins when the ChartDuration value is set to weekly, biweekly, or 4weeks.

***Access***

Write-only.

***Parameter***

***Data type***

String.

***Valid strings***

- Sunday
- Monday
- Tuesday
- Wednesday
- Thursday
- Friday
- Saturday

***Default value***

Sunday.

***Example***

**DocName**

---

***Description***

Sets the name of the top-level HTML document in a report.

***Access***

Write-only.

***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

Default.htm.

***Example***

**EndDateTime**

---

***Description***

Returns the end date/time for the report. Used as a parameter when calling the stored procedure.

***Access***

Read-only.

***Parameter***

***Data type***

String (SQL-safe format).

***Valid strings***

***Default value***

***Example***

## **EndDay**

---

***Description***

Sets the end date for a report.

***Access***

Write-only.

***Parameter***

***Data type***

Date.

For example:

```
ReportObj.EndDay = Date("1/1/2002")
```

***Valid strings***

***Default value***

***Example***

## **Filter**

---

***Description***

Sets the value for which filter object to use. The value is used by the MadeDrillDownReportV# method.

A filter takes a recordset from ADODataSource and generates a new recordset based on the filter implementation.

**Access**

Write-only.

**Parameter**

**Data type**

RecordsetFilter.

**Valid strings**

**Default value**

**Example**

```
Dim CrossTableObj
    Dim StatsFilterObj
    Dim TimeFilterObj
Set CrossTableObj=CreateObject ("NETIQFILTERS.CROSSTAB")
Set StatsFilterObj=CreateObject ("NETIQFILTERS.STATISTICS")
Set TimeFilterObj=CreateObject ("NETIQFILTERS.TIMEFILTER")
With TimeFilterObj
    .TimePeriodPerPoint = PRM_TIMEPERIOD * 3600
    .PreFilter           = CrossTableObj
End With
```

## FolderPrefix

---

**Description**

Sets the prefix used for the report output folder.

**Access**

Write-only.

**Parameter**

**Data type**

String.

***Valid strings***

***Default value***

Prefix.

***Example***

## **FolderType**

---

***Description***

Sets the naming convention of the report output folder.

***Access***

Write-only.

***Parameter***

***Data type***

Integer.

***Valid strings***

- 0 (FolderType\_Unique)
- 1 (FolderType\_UniqueWPrefix)
- 2 (FolderType\_Specific)

***Default value***

0.

***Example***

## **FuzzTime**

---

***Description***

Determines whether to use fuzzy time (sliding time range) for a report.

**Access**

Write-only.

**Parameter****Data type**

- Type: String
- Value: Y to use fuzzy time, N to use fixed time

**Valid strings****Default value**

Y.

**Example****FuzzTimeCurrent**

---

**Description**

Sets the start time of report job as the end date/time for the report data (if the value of the property is Y).

If the value of the property is N, then midnight of the day on which the report jobs starts is the end date/time for the report data.

**Access**

Write-only.

**Parameter****Data type**

- Type: String
- Value: Y to use start time of report job as end date/time, N to use midnight as end date/time

***Valid strings***

***Default value***

Y.

***Example***

## **FuzzTimeType**

---

***Description***

Sets the fuzzy time increment (hour, day, month, etc.).

***Access***

Write-only.

***Parameter***

***Data type***

Integer.

***Valid strings***

- 201 (year)
- 202 (month)
- 203 (week) currently not implemented
- 204 (day)
- 205 (hour)
- 206 (minute)

***Default value***

204.

***Example***

## **FuzzTimeUnitTime**

---

***Description***

Sets the number fuzzy time increments to use.

***Access***

Write-only.

***Parameter***

***Data type***

Integer.

***Valid strings***

***Default value***

1.

***Example***

## **HorizontalChart**

---

***Description***

Changes the chart from vertical to horizontal mode.

***Access***

Write-only.

***Parameter***

***Data type***

BOOL



***Default value***

FALSE

## **Hyperlink**

---

***Description***

Returns a blank string, or a hyperlink to a report.

***Access***

Read-only.

***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

***Example***

## **IndexAuthor**

---

***Description***

Sets the author name by which a report is indexed.

This value is used by the Layout object to generate XML index information.

***Access***

Write-only.

***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

Author.

***Example***

## **IndexCompany**

---

***Description***

Sets the company name by which a report is indexed.

This value is used by the Layout object to generate XML index information.

***Access***

Write-only.

***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

NetIQ.

***Example***

## **IndexComponent**

---

***Description***

Sets the monitored component name by which a report is indexed.

This value is used by the Layout object to generate XML index information.

***Access***

Write-only.

***Parameter******Data type***

String.

***Valid strings******Default value***

Component.

***Example***

## **IndexCustomProp1**

---

***Description***

Sets a custom property by which a report is indexed.

This value is used by the Layout object to generate XML index information.

***Access***

Write-only.

***Parameter******Data type***

String.

***Valid strings***

***Default value***

***Example***

## **IndexCustomProp2**

---

### ***Description***

Sets a custom property by which a report is indexed.

This value is used by the Layout object to generate XML index information.

### ***Access***

Write-only.

### ***Parameter***

### ***Data type***

String.

### ***Valid strings***

### ***Default value***

### ***Example***

## **IndexDescriptions**

---

### ***Description***

Sets description by which a report is indexed.

This value is used by the Layout object to generate XML index information.

### ***Access***

Write-only.

***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

Description.

***Example***

## **IndexExpires**

---

***Description***

Sets the number of days to keep a report before it can be deleted.

***Access***

Write-only.

***Parameter***

***Data type***

Integer.

***Valid strings***

***Default value***

60.

***Example***

## **IndexReportTitle**

---

***Description***

Sets a title by which a report is indexed.

This value is used by the Layout object to generate XML index information.

***Access***

Write-only.

***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

***Example***

## **InitialCatalog**

---

***Description***

Sets the database name used in the SQL query.

***Access***

Write-only.

***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

QDB.

### ***Example***

## **JobID**

---

### ***Description***

Appends the job ID to the name of the folder where the report is generated. Typically, the Report Script job ID is used to make the folder name unique, although the string value can be anything.

### ***Access***

Write-only.

### ***Parameter***

### ***Data type***

String.

### ***Default value***

## **Layout**

---

### ***Description***

Provides access to the Layout object.

### ***Access***

Read-only.

### ***Parameter***

### ***Data type***

Layout.

***Valid strings***

***Default value***

***Example***

## **LayoutFolder**

---

***Description***

Sets the name of the folder where a report is generated.

***Access***

Write-only.

***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

Folder.

***Example***

## **PathAndFileName**

---

***Description***

Returns the path to a report, and the file name of a report.

***Access***

Read-only.



***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

***Example***

## **RowDetailColor**

---

***Description***

Sets the background color of the data table.

***Access***

Write-only.

***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

class=TableBodyEven

***Example***

## **RowStyle**

---

***Description***

Sets the size of table rows.

**Access**

Write-only.

**Parameter****Data type**

String.

**Valid strings****Default value**

style='height:auto'

**Example****SeriesSeconds**

---

**Description**

Controls how the tick marks used to plot date-time based data are plotted along the X-axis.

**Access**

Write-only

**Parameter****Data type**

Long

**Default value**

3600

## StartDateTime

---

### **Description**

Returns the start date/time for a report. Used as a parameter when calling the stored procedure.

### **Access**

Read-only.

### **Parameter**

### **Data type**

String (SQL-safe format).

### **Valid strings**

### **Default value**

### **Example**

## StartDay

---

### **Description**

Sets the start date for a report.

### **Access**

Write-only.

### **Parameter**

### **Data type**

Date.

***Valid strings***

***Default value***

***Example***

`ReportObj.StartDay=Date("1/1/2002")`

## **SubTitle1**

---

***Description***

Sets the first subtitle of a report.

***Access***

Write-only.

***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

***Example***

## **SubTitle2**

---

***Description***

Sets the second subtitle of a report.

***Access***

Write-only.

***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

***Example***

## **Table**

---

***Description***

Provides access to the Table object. The Table object must be created with the MakeTable function.

***Access***

Read-only.

***Parameter***

***Data type***

Table.

***Valid strings***

***Default value***

***Example***

## **TextStyle**

---

***Description***

Sets HTML text formatting for a report.

***Access***

Write-only.

***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

style=\`font-size:12pt\`

***Example***

## VerticalSpace

---

***Description***

Adds a line vertical space in a report.

***Access***

Write-only.

***Parameter***

***Data type***

String.

***Valid strings***

***Default value***

10pt.

***Example***

## Report methods

The Report object has the following methods:

Method	Description
<a href="#">CreateTable</a>	Creates the Table object with the desired number of columns.
<a href="#">FinishChart</a>	Finishes a manually implemented chart.
<a href="#">Initialize</a>	Initializes the Report object.
<a href="#">MakeChart</a>	Creates a chart from a recordset.

Method	Description
<a href="#">MakeChartAndTable</a>	Combines the MakeChart and MakeDataTable methods.
<a href="#">MakeDataTable</a>	Creates a data table from a recordset.
<a href="#">MakeDrillDownReportV1</a>	Creates linked HTML pages from a series of recordsets.
<a href="#">MakeDrillDownReportV2</a>	Creates linked HTML pages from a set of recordsets.
<a href="#">TableHeader</a>	Specifies that the next line is the tableheader line.
<a href="#">TableRow</a>	Specifies a new table row and inserts data.
<a href="#">TableText</a>	Adds data to the current table column.

## CreateTable

### Description

Creates the Table object with the desired number of columns. Accessible from the Table parameter. Required when building a table manually.

### Return

Reference to the table object.

### Parameter

- Data type: Integer
- Parameter name: columns
- Description: Number of columns for table

### Prerequisite

### Syntax

```
MyTable = ReportObj.CreateTable(10)
```

### ***Recordset organization***

### ***Remarks***

## **FinishChart**

---

### ***Description***

Finishes a manually implemented chart. Not required when using MakeChart function.

### ***Return***

- Type: Bool
- Value: True for success, false for failure

### ***Parameter***

### ***Prerequisite***

### ***Syntax***

```
Flag = ReportObj.FinishChart
```

### ***Recordset organization***

### ***Remarks***

## **Initialize**

---

### ***Description***

Initializes the Report object.

### ***Return***

- Type: Bool
- Value: True for success, false for failure



**Parameter****Prerequisite****Syntax**

```
Flag = ReportObj.Initialize
```

**Recordset organization****Remarks****MakeChart**

---

**Description**

Creates a chart from a recordset. The chart is added to the report.

**Return**

- Type: Bool
- Value: True for success, false for failure

**Parameter**

- Data type: Recordset
- Parameter name: pRecordset
- Description: The recordset contains the Xdata and multiple columns of Ydata. The Xdata should be either a date/time field or a descriptive field.
- Data type: Integer
- Parameter name: FirstColumn
- Description: Specifies the first column, which is the column containing the Xdata. The default value is 0.

**Prerequisite****Syntax**

```
Flag = ReportObj.MakeChart (myRecordset)
```

## ***Recordset organization***

### ***Remarks***

## **MakeChartAndTable**

---

### ***Description***

Combines the functionality of the MakeChart and MakeDataTable methods.

### ***Return***

- Type: Bool
- Value: True for success, false for failure

### ***Parameter***

- Data type: Recordset
- Parameter name: pRecordset
- Description: The recordset contains the Xdata and multiple columns of Ydata. The Xdata should be either a date/time field or a descriptive field.
- Data type: String
- Parameter name: <mode>
- Description: Display mode for the chart and table. Use to determine what is contained in the report. Possible modes are `chart`, `table` or `both`. The default is `both`.

### ***Prerequisite***

### ***Syntax***

```
Flag = ReportObj.MakeChartAndTable(my Recordset,"both")
```

## ***Recordset organization***

### ***Remarks***

## **MakeDataTable**

---

### ***Description***

Uses the recordset to create the data table in the report.

### ***Return***

- Type: Bool
- Value: True for success, false for failure

### ***Parameter***

- Data type: Recordset
- Parameter name: pRecordset
- Description: Contains the recordset used to create the table
- Data type: Integer
- Parameter name: columns repeat
- Description: If the data value repeats, the number of columns in which an empty string value is printed

### ***Prerequisite***

### ***Syntax***

```
Flag = ReportObj.MakeDataTable(myRecordset)
```

## ***Recordset organization***

### ***Remarks***

## **MakeDrillDownReportV1**

---

### ***Description***

Takes a series of recordsets and generates a drill-down report consisting of a top-level HTML page with links to additional pages. The additional pages each contain a chart and table.

### ***Return***

- Type: Bool
- Value: True for success, false for failure

### ***Parameter***

- Data type: Recordset
- Parameter name: pRecordset
- Description: A series of recordsets, the first of which is a summary of the others
- Data type: String
- Parameter name: <mode>
- Description: Display mode for the chart and table. Use to determine what is contained in the report. Possible modes are `chart`, `table` or `both`. The default is `both`.

### ***Prerequisite***

### ***Syntax***

```
Flag = ReportObj.MakeDrillDownReportV1(recordset, "both")
```

### ***Recordset organization***

- Summary recordset: One row for each drill-down report.
- Drill-down report recordset: One detailed recordset (for the corresponding row in the summary recordset).

## **Remarks**

# **MakeDrillDownReportV2**

---

## **Description**

Takes a set of recordsets and generates a drill-down report consisting of a top-level HTML page with links to additional pages. At least two recordsets are required for each report. The first recordsets are summary descriptive data displayed on the top-level page, the additional recordsets supply the content of the reports on the additional pages.

## **Return**

- Type: Bool
- Value: True for success, false for failure

## **Parameter**

- Data type: Recordset
- Parameter name: pRecordset
- Description: Sets of recordsets consisting of summary descriptive data and detailed data
- Data type: String
- Parameter name: <mode>
- Description: Display mode for the chart and table. Use to determine what is contained in the report. Possible modes are **chart**, **table** or **both**. The default is **both**.
- Data type: Int
- Parameter name: NumberOfRepeatDataSets
- Description: The number of datasets for the summary dataset. The default value is 1.

### ***Prerequisite***

### ***Syntax***

```
Flag = ReportObj.MakeDrillDownReportV2(recordset, "both")
```

### ***Recordset organization***

One-row summary recordset (displayed in top-level HTML page), and one or more detailed recordsets displayed in additional pages. This pairing of recordsets is used to display all data in a report.

### ***Remarks***

## **TableHeader**

---

### ***Description***

Specifies that the next line in the script begins the table header. Follow this function with TableText to add data to each column.

### ***Return***

### ***Parameter***

### ***Prerequisite***

CreatTable(columns) method.

### ***Syntax***

```
ReportObj.TableHeader
```

### ***Recordset organization***

### ***Remarks***

## **TableRow**

---

### ***Description***

Specifies a new table row. Inserts data into at least two columns.

***Return******Parameter***

- Data type: Variant
- Parameter name: Col1
- Description: Specifies data to insert into table at column 1
- Data type: Variant
- Parameter name: Col2
- Description: Specifies data to insert into table at column 2

***Prerequisite***

CreatTable(columns) method.

***Syntax***

ReportObj.TableRow "<label>, <value>"

***Recordset organization******Remarks*****TableText**

---

***Description***

Adds data to the current table column. Moves the column pointer forward.

***Return******Parameter***

- Data type: Variant
- Parameter name: <value>
- Description: Specifies data to insert into table at current column

***Prerequisite***

CreatTable(columns) method.

**Syntax**

ReportObj.TableText

**Recordset organization****Remarks****Write properties**

	Functions	Initialize	MakeChart	MakeChartAndTable	MakeDrillDownReportV1	MakeDrillDownReportV2	MakeDataTable
<b>Properties</b>							
ChartColorScheme		X	X	X	X		
ChartDataLabelAngle		X	X	X	X		
ChartDuration		X	X	X	X		
ChartPerColumn		X	X	X	X		
ChartStyle		X	X	X	X		
ChartThresBlue		X	X	X	X		
ChartThresGreen		X	X	X	X		
ChartThresRed		X	X	X	X		
ChartThresValue		X	X	X	X		
ChartTitle		X	X	X	X		
ChartTransparency		X	X	X	X		
ChartViewMode		X	X	X	X		
ChartXRot		X	X	X	X		
ChartXSize		X	X	X	X		



	Functions	Initialize	MakeChart	MakeChartAndTable	MakeDrillDownReportV1	MakeDrillDownReportV2	MakeDataTable
<b>Properties</b>							
ChartYAxisAutoRange			X	X	X	X	
ChartYAxisBottomLabel			X	X	X	X	
ChartYAxisLabel			X	X	X	X	
ChartYAxisMax			X	X	X	X	
ChartYAxisMin			X	X	X	X	
ChartYAxisTopLabel			X	X	X	X	
ChartYRot			X	X	X	X	
ChartYSize			X	X	X	X	
ChildColumnFormat					X	X	
ChildSubTitle1Prefix					X	X	
ChildSubTitle2					X	X	
ColumnAlignCenter				X			X
ColumnAlignLeft				X			X
ColumnFormat				X			X
ColumnHeadColor				X			X
ColumnLineType			X	X	X	X	
DataSource	X						
DayOfWeekChartStart			X	X	X	X	
DocName	X						
EndDay	X						
Filter					X	X	
FolderPrefix	X						

Functions	Initialize	MakeChart	MakeChartAndTable	MakeDrillDownReportV1	MakeDrillDownReportV2	MakeDataTable
<b>Properties</b>						
FolderType	X					
FuzzTimeCurrent	X					
FuzzTimeType	X					
FuzzTimeUnitTime	X					
FuzzyTime	X					
HorizontalChart		X	X	X	X	
IndexAuthor	X					
IndexCompany	X					
IndexComponent	X					
IndexCustomProp1	X					
IndexCustomProp2	X					
IndexDescriptions	X					
IndexExpires	X					
IndexReportTitle	X					
InitialCatalog	X					
JobID	X					
LayoutFolder	X					
RowDetailColor			X			X
RowStyle			X			X
SeriesSeconds		X	X	X	X	
StartDay	X					
Subtitle1	X					

	Functions	Initialize	MakeChart	MakeChartAndTable	MakeDrillDownReportV1	MakeDrillDownReportV2	MakeDataTable
Properties							
Subtitle2	X						
TestStyle			X				X
TitleDescription	X						
	X				X	X	



# Index

## A

### ADODataSource

- AddParameter 11
- ADO objects 8
- Command 12
- Execute 14
- filter objects 8
- Initialize 15
- methods 14
- Prepare 15
- properties 11
- Provider 12
- Recordset 13
- Timeout 13

## C

### CROSSTAB

- code sample 19
- overview 18

## D

### Data filter objects

- CROSSTAB 18
- HISTOGRAM 21
- PERCENT 26
- STATISTICS 27
- TIMEFILTER 31

## H

### HISTOGRAM

- AutoRange 24
- code sample 24
- HighLimit 21
- LowLimit 22

- methods 23
- Mode 22
- Number of Bins 23
- overview 21
- properties 21

## I

- introduction 7

## P

- PERCENT 26

## R

- Report object

- ADODataSource 42
- Chart 43
- ChartColorScheme 43
- ChartDataLabelAngle 44
- ChartPerColumn 45
- ChartStyle 46
- ChartThresBlue 47
- ChartThresGreen 48
- ChartThresValue 49
- ChartTitle 50
- ChartTransparency 50
- ChartViewMode 51
- ChartXRot 51
- ChartXSize 52
- ChartYAxisAutoRange 52
- ChartYAxisBottomLabel 53
- ChartYAxisLabel 53
- ChartYAxisMax 54
- ChartYAxisMin 54
- ChartYAxisTopLabel 55
- ChartYRot 55
- ChartYSize 56
- ChildColumnFormat 56

ChildSubtitle1Prefix 57  
ChildSubtitle2 58  
ColumnAlignCenter 59  
ColumnAlignLeft 59  
ColumnFormat 60  
ColumnHeadColor 61  
ColumnLineType 61  
CreateTable 85  
DataSource 63  
DayOfWeekChartStart 63  
DocName 64  
EndDateTime 64  
EndDay 65  
Filter 65  
FinishChart 86  
FolderPrefix 66  
FolderType 67  
FuzzTime 67  
FuzzTimeCurrent 68  
FuzzTimeType 69  
FuzzTimeUnitTime 70  
HorizontalChart 70  
Hyperlink 71  
IndexAuthor 71  
IndexCompany 72  
IndexComponent 72  
IndexCustomProp1 73  
IndexCustomProp2 74  
IndexDescriptions 74  
IndexExpires 75  
IndexReportTitle 75  
InitialCatalog 76  
Initialize 86  
JobID 77

- Layout 77
- LayoutFolder 78
- MakeChart 87
- MakeChartAndTable 88
- MakeDataTable 89
- MakeDrillDownReportV1 90
- MakeDrillDownReportV2 91
- methods 84
- object overview 9
- PathAndFileName 78
- properties 39
- RowDetailColor 79
- RowStyle 79
- SeriesSeconds 80
- StartDateTime 81
- StartDay 81
- SubTitle1 82
- SubTitle2 82
- Table 83
- TableHeader 92
- TableRow 92
- TableText 93
- TextStyle 83
- Vertical Space 84

## S

### STATISTICS

- Grouping 29
- Output 27
- OutputMode 28
- overview 27
- properties 27

## T

### TIMEFILTER

- AddExclusionDateFilter 36



- AddTimeOfDayFilter 36
  - code sample 38
- DayOfWeekFilter 32
- Duration 33
- MaxPointsPerChart 34
  - methods 36
- MonthAggregate 35
  - overview 31
  - properties 31
- RemoveColumn 37
- SetDaysOfThe WeekFilter 37
- TimePeriodPerPoint 35