

# DSGetProperty Function

## Description

Gets the specified DataSet Object property; the DataSet Object is specified by the passed handle.

## Syntax

*flag* = **DSGetProperty** (*hDS*, *property*, *value*, *arg*)

## Parameters

The DSGetProperty function has the following parameters:

Parameter	Description
<i>hDS</i>	Handle to a DataSet returned by DSInstance.
<i>property</i>	See <i>properties</i> below.
<i>value</i>	Property specific; see properties below.
<i>Arg</i>	Property specific; see properties below.

Property	Description	Value	Argument
DS_ARG\$	Returns the current value for the specified argument.	<i>value</i> [out]	Argument value (@FM-delimited list).
		<i>arg</i> [in]	Argument ID.
DS_ARGCOUNT\$	Returns the number of arguments in the DataSet.	<i>value</i> [out]	Argument count.
DS_ARGID\$	Returns the index of the passed argument name.		
DS_COLCOUNT\$	Returns the number of columns in the DataSet.	<i>value</i> [out]	Count of columns in DataSet.
DS_COLDESCRIPT\$	Returns the specified column's name, type, precision, scale, nullability, OI type, and key flag.	<i>value</i> [out]	Column name, type, precision, scale, nullability, OI type, and key flag @VM-delimited.
		<i>arg</i> [in]	Column number.
DS_COLID\$	Returns the column index for the specified column name.	<i>value</i> [out]	Column index (or @VM-delimited list).
		<i>arg</i> [in]	Column name (or @VM-delimited list).
DS_COLISKEY\$	Checks if the specified column is a key column.	<i>value</i> [out]	TRUE\$ if the column is a key column.
		<i>arg</i> [in]	Column number.
DS_COLNAME\$	Returns the specified column's name.	<i>value</i> [out]	Column name.
		<i>arg</i> [in]	Column number.
DS_COLNULLABLE\$	Returns the specified column's nullability.	<i>value</i> [out]	Nullable flag.
		<i>arg</i> [in]	Column number.
DS_COLOI TYPE\$	Returns the specified column's OI type.	<i>value</i> [out]	OI type.
		<i>arg</i> [in]	Column number.
DS_COLPRECISION\$	Returns the specified column's precision.	<i>value</i> [out]	Column precision.
		<i>arg</i> [in]	Column number.
DS_COLSCALE\$	Returns the specified column's scale.	<i>value</i> [out]	Column scale.
		<i>arg</i> [in]	Column number.

DS_COLTYPE\$	Returns the specified column's type.	value[output]	Column type.
		arg[in]	Column number.
DS_CONNECTION\$	Returns the Connection Object handle which the DataSet uses.	value[output]	Connection Object handle (hXO) for the DataSet.
DS_DELETE_SCRIPT\$	Returns the DataSet's delete script.	value[output]	DataSet's delete script.
DS_DFT_ISNULL\$	Returns TRUE\$ if the specified default column is NULL.	value[output]	TRUE\$ for NULL, FALSE\$ otherwise.
		arg[in]	Column ID.
DS_DFT_ROW\$	Returns the default row using the specified conversion (defaults to OI conversion).	value[output]	Row of @FM-delimited data (NULLs are returned as 0-length values).
		arg[in]	[Conversion].
DS_DFT_VALUE\$	Returns the default column value for the specified column optionally using the specified conversion.	value[output]	Column value (@FM-delimited list) (NULL is returned as an empty string).
		arg[in]	Column ID [@FM conversion].
DS_KEYLIST\$	Returns a list of the keys in the current view. This property is typically used to store the current filtered set of keys. Using DS SetProperty(DS_KEYLIST\$) with the stored set of keys will restore the filter to the state that it was in when DSGetProperty(DS_KEYLIST\$) was called.	value[output]	An @fm-delimited list of keys; multiple columns are delimited by @vm if more than one column is used as the key.
		arg [in]	FirstRow: @fm: LastRow. FirstRow specifies the first row to retrieve the key for; defaults to row 1. LastRow specifies the last row to retrieve the key for; defaults to the last row in the DataSet.
DS_INSERT_SCRIPT\$	Returns the DataSet's insert script.	value[output]	DataSet's insert script.
DS_ISNULL\$	Returns TRUE\$ if the specified value is NULL.	value[output]	TRUE\$ for NULL, FALSE\$ otherwise. If more than one column is specified, returns an @FM-delimited list of null flags.
		arg[in]	Column [@FM rowtype].
DS_MODIFIED\$	Returns TRUE\$ if the DataSet has been modified.	value[output]	TRUE\$ if the DataSet has been modified.
DS_RECORD\$	Efficiently returns multiple rows from the current view (or the entire view) in OpenInsight format. This property is significantly more efficient than the DS_ROW\$ property for multiple rows. This property is Read-Only.	value[output]	An @rm-delimited list of rows, with columns delimited by @fm.
		arg[in]	FirstRow: @fm: LastRow. FirstRow specifies the first row to retrieve; defaults to row 1. LastRow specifies the last row to retrieve; defaults to the last row in the DataSet.
DS_ROW\$	Returns a row of data, either the current row (default), the default row, or the workspace row, using either the OI conversion (default) or any other specified conversion.	value[output]	Row of @FM-delimited data (NULLs are returned as 0-length values).
		arg[in]	[Conversion [@FM rowtype]].
DS_ROWCOUNT\$	Returns the number of rows in the DataSet.	value[output]	Count of @FM-delimited rows in the current view of the DataSet.
DS_ROWID\$	Returns the current row in the DataSet.	value[output]	Current row index (1-based).
DS_QUERY\$	Returns the query handle used by the DataSet, allowing the DataSet timeout to be set using the QrySetProperty(QRY_TIMEOUT\$) property.	hDS[in]	The DataSet Object handle.
		value[output]	Query handle or 0 if none.
DS_SCRIPT\$	Returns the specified script used by the DataSet.	value[output]	The specified script from the DataSet.
		arg[in]	Script ID (defaults to SELECT).
DS_SELECT_SCRIPT\$	Returns the DataSet's select script.	value[output]	DataSet's select script.
DS_UPDATE_SCRIPT\$	Returns the DataSet's update script.	value[output]	DataSet's update script.
DS_VALID\$	Checks if the passed DataSet handle is valid.	value[output]	TRUE\$ if the passed DataSet handle is valid.
		hXO[in]	Connection Object handle.

DS_VALUE\$	Returns a column's value optionally of a specific conversion (defaults to OI conversion) and rowtype (defaults to a DataSet row).	<i>value</i> [out]	Column value (@FM-delimited) (NULL is returned as an empty string).
		<i>arg</i> [in]	Column ID [@FM conversion [@FM rowtype]].
DS_VIEW\$	Returns the current view (master, select, insert, update, delete) of the DataSet.	<i>value</i> [out]	View ID (master, select, insert, update, delete).
DS_WORK_ISNULL\$	Returns TRUE\$ if the specified workspace column is NULL.	<i>value</i> [out]	TRUE\$ for NULL, FALSE\$ otherwise.
		<i>arg</i> [in]	Column ID.
DS_WORK_ROW\$	Returns the workspace row using the specified conversion (defaults to OI conversion).	<i>value</i> [out]	Row of @FM-delimited data (NULLs are returned as 0-length values).
		<i>arg</i> [in]	[Conversion].
DS_WORK_VALUE\$	Returns the workspace column value for the specified column optionally using the specified conversion.	<i>value</i> [out]	Column value (@FM-delimited) (NULL is returned as an empty string).
		<i>arg</i> [in]	Column ID [@FM conversion]

Column and argument parameters are permitted to be either name or position (number). In addition, DSGetProperty supports @VM lists of columns, argument names/positions to be passed, or 0 to signify all.

## Returns

True for successful execution or **False** for failure.