

# Barista Document Output Object

## Description

The Barista Document Output Object provides the ability to create and display reports outside of Barista in standalone mode.

Note: The Barista environment used for the session is specified by one of two ways:

1. An absolute path in the “use” statement. For example, if the object is specified using “use ::c:/Program Files/basis/barista/sys/prog/bao\_document.bbj::Document”, Barista will retrieve the configurations in the c:/Program Files/basis/barista environment.
2. Specifying the object with a relative path from a Barista environment started with the Barista configuration file found at: **barista/sys/config/<lang\_id>/barista.cfg**.

For additional information on creating reports, please see the [Document Output Tutorial](#).

## Document Output Object Methods

### Login and Security

Method	Comments
setUserID(str)	Barista user identifier. (See Barista Launch Task doc for more)
setPassword(str)	Barista user password. (See Barista Launch Task doc for more)
setFirmID(str)	Currently active Barista Firm identifier.
setLanguage(str)	Barista language code (“DEU”, “ENU”, etc.). Defaults to “ENU”.
setDBServer(str)	Default “localhost”.
setDBPort(int)	Default 2002.
setDBSSL(int)	Default 0 (non-SSL)
setDBTimeout(int)	Default 5000 (ms)

### Report Creation

Method	Comments
setDocumentID (str)	Document identifier user for Document Print settings and storing adjusted document column width. Defaults to “TEST_REPORT”.
setReportTitle(str)	Top level document title.
getReportDate(str)	Current report date in YYYYMMDD format. Defaults to system date if not supplied.
setReportTime(str)	Current report time in MMHHSS format. Defaults to system time if not supplied.
setSourceType(str)	Document system source. (B)arista for table listings, or any other user defined 1 byte code. Written to document record.
setSourceReference(str)	Related to source id (below). May include order number, journal number, etc. Written to document record.
setSourceAlias(str)	Barista table alias. Written to document record.
setSourceID(str)	Document application source reference (O)rder Proc, (G)eneral Ledger,

	etc. Written to document record.
setReportSource(str)	Document origin: (I)nternal Barista application, (E)xternal reporting (Jasper, etc.) Written to document record.
setOutputAction(str)	Output action. May contain one or more of the following: NAME[] - Predefined output document file name LST - Use single column grid w/ non proportional font FORMAT - Document contains special cell formatting OUTOPTS[] - Direct output string containing <fmt-opt>;<fmt-opt>;<fmt-opt>, where: fmt = Output format PRT - Standard printer PDF - Portable Document File (.pdf) CSV - Comma Separated Text (.csv) TXT - Tab Delimited Text (.txt) HTM - HTML Document (.htm) XLS - Spreadsheet (.xls) XML - XML Document (.xml) opt = Output type PRT - Standard Printer UNF - Unformatted output DSK - Disk file EML - Email FAX - Fax GDC - GoogleDocs repository GDF - GoogleDocs repository - Google format LCH - Launch source file after generation
addReportHeading(str)	Adds a report heading line (At least one is required)
addColumn( str <column_header>, str <data_type>, int <data_width>, int <pixel_width>, int <output_mask>, str <control_type>, str <align_code>, str <flags>)	Adds a column to the document: <ul style="list-style-type: none"> <li>● Column header - Column header text</li> <li>● Data type - Column data type: <ul style="list-style-type: none"> <li>○ "C" - Character</li> <li>○ "N" Number</li> </ul> </li> <li>● Data width - Number of data characters</li> <li>● Pixel width - Column pixel width. Overrides the automatic calculation based on data width x characters.</li> <li>● Output mask - String or numeric output mask</li> <li>● Control Type - Future use</li> <li>● Column alignment code <ul style="list-style-type: none"> <li>○ L = Left</li> <li>○ R = Right</li> <li>○ C = Center</li> </ul> </li> <li>● Flags - Additional document formatting flags: <ul style="list-style-type: none"> <li>○ B = Bold</li> <li>○ I = Italics</li> <li>○ R = Right justify</li> <li>○ L = Left justify</li> <li>○ C = Center</li> <li>○ + = Increase font size</li> <li>○ - = Decrease font size</li> </ul> </li> </ul>
setOutputData(vect!)	The main output vector containing the report row data.
setBreakData(vect!)	Vector containing the position in the main output vector where sub-heading break should occur.

setHeaderData(vect!)	Vector containing sub-heading to print for a given position in the break vector.
setTotalData(vect!)	Vector containing position in the main output vector where underlining should occur.
process()	Creates and archives the report. Launches the Barista Document Output Viewer when completed.
setSessionWait(int)	Set to 0 to continue processing immediately Set to 1 to wait for DocOut process to end before continuing (Default)

## Sample Program

The following program creates a report from the Barista Examples Item data file:

```
use ::c:/Program Files/basis/barista/sys/prog/bao_document.bbj::Document
rem Specifies the environment when generating a DocOut report from a non-
Barista environment.

use ::sys/prog/bao_document.bbj::Document
rem Used when generating a DocOut report from program within a Barista
environment.

declare Document doc!
doc! = new Document()

item_chn = unt
open(item_chn) "../apps/examples/data/ITEM"
dim item$:"item_id:c(6),description:c(30),prod_cat:c(2),stock_uom:c(3)," +
:         "cost:n(12),weight:n(12),wt_unit:c(2),price:n(12)"

doc!.setUserID("ADMIN")
doc!.setPassword("admin123")
doc!.setFirmID("01")
doc!.setLanguage("ENU")

doc!.setDocumentID("ITEM_REPORT")
doc!.setReportTitle("Item Report")
doc!.addReportHeading("Item Report")

doc!.addColumn("Item ID", "C", 6, 70, "", "E", "L", "")
doc!.addColumn("Description", "C", 30, 300, "", "E", "L", "")
doc!.addColumn("Prod Cat", "C", 6, 60, "", "E", "L", "")
doc!.addColumn("UOM", "C", 6, 60, "", "E", "L", "")
doc!.addColumn("Cost", "N", 12, 100, "###,###.00", "N", "R", "")

out!=bbjAPI().makeVector()
while 1
  readrecord(item_chn, end = *break) item$
  out!.addItem(item.item_id$)
  out!.addItem(item.description$)
  out!.addItem(item.prod_cat$)
  out!.addItem(item.stock_uom$)
  out!.addItem(item.cost$)
wend

doc!.setOutputData(out!)
```

```
doc!.process()
```

```
release
```