



THE
POWER
TO KNOW.

SAS[®] Drug Development 3.4 Macros User's Guide

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2008. *SAS® Drug Development 3.4: Macros User's Guide*. (Second printing) Cary, NC: SAS Institute Inc.

SAS® Drug Development 3.4: Macros User's Guide

Copyright © 2009, SAS Institute Inc., Cary, NC, USA

All rights reserved. Produced in the United States of America.

For a hard-copy book: No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

For a Web download or e-book: Your use of this publication shall be governed by the terms established by the vendor at the time you acquire this publication.

U.S. Government Restricted Rights Notice: Use, duplication, or disclosure of this software and related documentation by the U.S. government is subject to the Agreement with SAS Institute and the restrictions set forth in FAR 52.227-19, Commercial Computer Software-Restricted Rights (June 1987).

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st electronic book, April 2008

SAS® Publishing provides a complete selection of books and electronic products to help customers use SAS software to its fullest potential. For more information about our e-books, e-learning products, CDs, and hard-copy books, visit the SAS Publishing Web site at support.sas.com/publishing or call 1-800-727-3228.

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

Contents

Chapter 1 Introduction 1

Audience 1

Typographic and Syntax Conventions Used in This Guide 1

Chapter 2 Installing the Macros 3

Overview 3

Requirements 3

Installation Instructions for Microsoft Windows 3

Installation Instructions for UNIX 4

Verifying the Installation 6

Chapter 3 SAS Drug Development Macros 7

Introduction 9

Macro Return Codes 9

Macros and System Policies 10

The Macros 10

Examples 30

Appendix 1 Information for Users of the Command Facility 35

Correspondence of Command Facility Macros to SAS Drug Development Macros 35

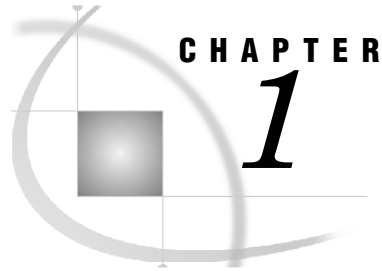
Appendix 2 Reference 37

Name of an Object Type When Specified as a Macro Parameter 37

Editable Object Properties 38

Editable User Account Properties 42

System Policies 43



CHAPTER

1

Introduction

<i>Audience</i>	1
<i>Typographic and Syntax Conventions Used in This Guide</i>	1

Audience

This guide is intended for users who want to develop applications with the SAS Drug Development macros.

You must be familiar with SAS Drug Development functionality, such as type definitions, containers, files, and access permissions. For reference information on SAS Drug Development functionality, see the SAS Drug Development online Help and user's guide.

Typographic and Syntax Conventions Used in This Guide

Throughout this guide, you will see the following typographic conventions:

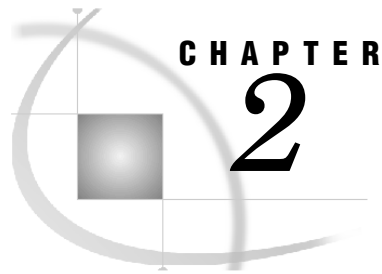
Convention	Description
monospace font	denotes code, such as a code example
monospace bold font	denotes text that you type, such as an object name
<i>monospace bold italics font</i>	denotes a value that you specify, such as your name

The following graphic explains the syntax for SAS code used in this document:

Syntax Conventions

```
PROC DATASETS <LIBRARY=libref> <MEMTYPE=(mtype-list)>  
      <DETAILS | NODetails> <other-options>;  
RENAME variable-1=new-name-1 <... variable-n=new-name-n>;
```

- 1 SAS keywords, such as statement or procedure names, appear in bold type.
- 2 Values that you must spell as they are given in the syntax appear in uppercase type.
- 3 Optional arguments appear inside angle brackets(<>).
- 4 Mutually exclusive choices are joined with a vertical bar(|).
- 5 Values that you must supply appear in italic type.
- 6 Argument groups that you can repeat are indicated by an ellipsis (...).



Installing the Macros

<i>Overview</i>	3
<i>Requirements</i>	3
<i>Installation Instructions for Microsoft Windows</i>	3
<i>Installation Instructions for UNIX</i>	4
<i>Verifying the Installation</i>	6

Overview

This document describes how to install the SAS Drug Development macros, which are distributed in the file SASDrugDevRemoteAPI_Macros.zip.

Requirements

The following software is required by the SAS Drug Development macros:

- SAS 9.1.3 Service Pack 4
- Java Runtime Environment Version 1.5.0*
- SAS Drug Development 3.4 remote API client

Installation Instructions for Microsoft Windows

- 1 Unzip the contents of SASDrugDevRemoteAPI_Macros.zip to C:\.

This creates the following files and folder in C:\:

- SASDrugDevRemoteAPI_Macros\sddmacros.cfg
- SASDrugDevRemoteAPI_Macros\sasv9_local.cfg
- SASDrugDevRemoteAPI_Macros\version.txt
- SASDrugDevRemoteAPI_Macros\lib\sas-drugdev-sasmacros.jar
- SASDrugDevRemoteAPI_Macros\lib\log4j.properties
- SASDrugDevRemoteAPI_Macros\sasmacro
 - This is the folder that contains the SAS Drug Development macros as .sas files.

- 2 Review the properties of the shortcut that starts SAS to ensure that the shortcut includes the option that points to the configuration file !SASROOT\sasv9.cfg.

For example:

```
-CONFIG "C:\Program Files\SAS\SAS9.1\sasv9.cfg"
```

Note: The default location for SASROOT is `C:\Program Files\SAS\SAS9.1`.

- 3 Edit the file `!SASROOT\sasv9.cfg` by adding the following line:

```
-CONFIG "C:\SASDrugDevRemoteAPI_Macros\sddmacros.cfg"
```

- 4 If the SAS Drug Development remote API client is installed in a location other than `C:\SASDrugDevRemoteAPI`, you must edit the file `sddmacros.cfg` as follows:

Caution: Back up this file before you edit it. Use extreme care when editing this file. If you have any questions, concerns, or problems, contact your on-site SAS support personnel.

- a Modify the `-JREOPTIONS` statement by editing the option `-Dsas.app.class.dirs` to include the absolute path to the `lib` folder in the SAS Drug Development remote API client installation and `C:\SASDrugDevRemoteAPI_Macros\lib`.
- b If you are connecting to an instance of SAS Drug Development that is hosted by SAS, and you are behind a firewall and proxy servers, you might need to address proxy requirements or restrictions.

If HTTPS traffic is proxied, specify the following Java system properties in the `-JREOPTIONS` statement to configure the Java runtime environment:

```
-Dhttps.proxyHost=<proxy-host-name> -Dhttps.proxyPort=<port-number>
```

Caution: Modify only these options. Ensure that you do not insert any carriage returns in the `sas.app.class.dirs` option.

The `-JREOPTIONS` statement (when using a proxy server) will look similar to the following when you are finished:

```
-JREOPTIONS=(
-Dsas.app.class.dirs=C:\RemoteAPI\lib;C:\SASDrugDevRemoteAPI_Macros\lib
-Dsas.javaobj.experimental=no
-Dhttps.proxyHost=yourProxyServer.sas.com
-Dhttps.proxyPort=8080)
```

- 5 Edit the file `!SASROOT\nls\language\sasv9.cfg` (where *language* is the language used by SAS, such as `en` for English) to change the `-SET SASAUTOS` statement to include the folder `C:\SASDrugDevRemoteAPI_Macros\sasmacro`.

For example:

```
-SET SASAUTOS ("C:\SASDrugDevRemoteAPI_Macros\sasmacro")
```

Installation Instructions for UNIX

- 1 Create a directory under `!SASROOT` called **RemoteAPI**. For the purposes of these instructions, SASROOT will be referred to as `/apps/sas9.1.3`.
- 2 Unzip the contents of `SASDrugDevRemoteAPI_Macros.zip` to `/apps/sas9.1.3/RemoteAPI/`.

This creates the following files and folder in `/apps/sas9.1.3/RemoteAPI/`:

- ❑ SASDrugDevRemoteAPI_Macros/sddmacros.cfg
 - ❑ SASDrugDevRemoteAPI_Macros/sasv9_local.cfg
 - ❑ SASDrugDevRemoteAPI_Macros/version.txt
 - ❑ SASDrugDevRemoteAPI_Macros/lib/sas-drugdev-sasmacros.jar
 - ❑ SASDrugDevRemoteAPI_Macros/lib/log4j.properties
 - ❑ SASDrugDevRemoteAPI_Macros/sasmacro
- This is the folder that contains the SAS Drug Development macros as .sas files.

3 To install the SAS Drug Development remote API client, unzip the contents of SASDrugDevRemoteAPI.zip to a temp area on the UNIX server. This will create a SASDrugDevRemoteAPI directory. Copy all the jars in SASDrugDevRemoteAPI/lib to /apps/sas9.1.3/RemoteAPI/SASDrugDevRemoteAPI_Macros/lib.

4 Modify the access permissions of the !SASROOT/RemoteAPI directory recursively to match the access permissions of the user ID and group that installed and runs SAS.

For example: `chown -R sasadmin:sasadmin /apps/sas9.1.3/RemoteAPI`

5 Edit the file !SASROOT/sasv9_local.cfg as follows:

Caution: Back up this file before you edit it. Use extreme care when editing this file. If you have any questions, concerns, or problems, contact your on-site SAS support personnel.

Add in the JREOPTIONS statement so that it looks like the following example:

```
-JREOPTIONS=(
-Dsas.app.class.dirs=/apps/sas9.1.3/RemoteAPI/SASDrugDevRemoteAPI_Macros/lib
-Dsas.javaobj.experimental=no
)
```

If you are connecting to an instance of SAS Drug Development that is hosted by SAS, and you are behind a firewall and proxy servers, you might need to address proxy requirements or restrictions.

If HTTPS traffic is proxied, specify the following Java system properties in the -JREOPTIONS statement to configure the Java runtime environment:

```
-Dhttps.proxyHost=<proxy-host-name> -Dhttps.proxyPort=<port-number>
```

Caution: Modify only these options. Ensure that you do not insert any carriage returns in the sas.app.class.dirs option.

The -JREOPTIONS statement will look similar to the following when you are finished:

```
-JREOPTIONS=(
-Dsas.app.class.dirs=/apps/sas9.1.3/RemoteAPI/lib;/SASDrugDevRemoteAPI_Macros/lib
-Dsas.javaobj.experimental=no
-Dhttps.proxyHost=yourProxyServer.sas.com
-Dhttps.proxyPort=8080)
```

- 6 Edit the file `!SASROOT/sasv9.cfg` to change or add the `-SASAUTOS` statement to include the path to the SAS Drug Development macros.

For example:

```
-SASAUTOS ('!SASROOT/sasautos'
'/apps/sas9.1.3/RemoteAPI/SASDrugDevRemoteAPI_Macros/sasmacro')
```

Note: You might need to enclose existing SASAUTOS settings in quotation marks.

Verifying the Installation

The following SAS code displays the settings for the `-JREOPTIONS` and verifies that the JRE is configured properly. It does not verify that the SAS Drug Development macros are installed properly.

```
proc javainfo; run;
```

The following SAS code verifies that the SAS Drug Development macros are installed. Replace `sdd-instance`, `sdd-user-ID`, and `sdd-password` with values that reflect your instance of SAS Drug Development.

```
options mprint;
proc javainfo;
run;

%sasdrugdev_login(url=%str(https://sdd-
instance/sddremote),sdduserid=%str(sdd-user-ID),sddpassword=%str(sdd-
password));

/* List the contents of the root folder in SDD */
%sasdrugdev_getobjects(sddpath=/SDD);

proc print;
title "List of Objects in /SDD";
run;

%sasdrugdev_logout;
```

The code should generate a list of the contents of the root folder in the SAS Drug Development repository.

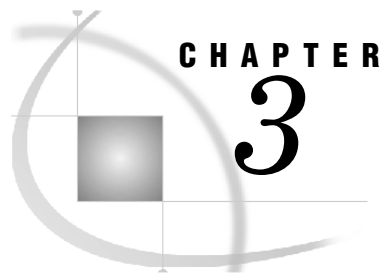
Note: By default, the root folder in the SAS Drug Development repository is `/SDD`. If the root of your SAS Drug Development repository is different, change `/SDD` in the code above.

The SAS log will contain information that might be useful for debugging the installation of the SAS Drug Development macros.

If you have problems connecting to SAS Drug Development, add the debugging parameters to `%sasdrugdev_login()`: `DEBUGLOG` and `DEBUGLEVEL`.

For example:

```
%sasdrugdev_login(url=%str(https://sdd-
instance/sddremote),sdduserid=%str(sdd-user-ID),sddpassword=%str(sdd-
password),debuglog=absolute-path-to-log,debuglevel=DEBUG);
```



CHAPTER
3

SAS Drug Development Macros

<i>Introduction</i>	9
<i>Macro Return Codes</i>	9
<i>Macros and System Policies</i>	10
<i>The Macros</i>	10
<i>Using the Ampersand Character (&) in URLs</i>	10
<i>The Proper Case for Parameter Values</i>	11
SASDRUGDEV_CONTAINEREXISTS.....	11
Description.....	11
Syntax.....	11
SASDRUGDEV_COPYCONTAINER.....	11
Description.....	11
Syntax.....	11
SASDRUGDEV_COPYFILE.....	11
Description.....	11
Syntax.....	12
SASDRUGDEV_CREATEGROUP.....	12
Description.....	12
Syntax.....	12
SASDRUGDEV_CREATELOCALFILE.....	12
Description.....	12
Syntax.....	12
SASDRUGDEV_CREATESDDCONTAINER.....	13
Description.....	13
Syntax.....	13
SASDRUGDEV_CREATESDDFILE.....	13
Description.....	13
Syntax.....	13
SASDRUGDEV_CREATEUSER.....	14
Syntax.....	14
SASDRUGDEV_DELETECONTAINER.....	14
Description.....	14
Syntax.....	14
SASDRUGDEV_DELETEFILE.....	14
Description.....	14
Syntax.....	14
SASDRUGDEV_DELETEGROUP.....	15
Description.....	15
Syntax.....	15
SASDRUGDEV_FILEEXISTS.....	15
Description.....	15
Syntax.....	15
SASDRUGDEV_GETCONTAINERPROPS.....	15
Description.....	15
Syntax.....	15
SASDRUGDEV_GETFILEPROPS.....	16
Description.....	16
Syntax.....	16
SASDRUGDEV_GETGROUPS.....	16
Description.....	16
Syntax.....	17
SASDRUGDEV_GETGROUPMEMBERS.....	17
Description.....	17
Syntax.....	17

<i>SASDRUGDEV_GETOBJECTS</i>	17
Description.....	17
Syntax	17
<i>SASDRUGDEV_GETPERMISSIONS</i>	18
Description.....	18
Syntax	18
<i>SASDRUGDEV_GETPASSWORDEXPIRES</i>	20
Description.....	20
Syntax	20
<i>SASDRUGDEV_GETSDDVALUES</i>	20
Description.....	20
Syntax	20
<i>SASDRUGDEV_GETUSERGROUPS</i>	21
Description.....	21
Syntax	21
<i>SASDRUGDEV_GETUSERPOLICIES</i>	21
Description.....	21
Syntax	21
<i>SASDRUGDEV_GETUSERPROPS</i>	22
Description.....	22
Syntax	22
<i>SASDRUGDEV_GETUSERS</i>	22
Description.....	22
Syntax	23
<i>SASDRUGDEV_GETUSERSWITHPOLICY</i>	23
Description.....	23
Syntax	23
<i>SASDRUGDEV_LOGIN</i>	24
Description.....	24
Syntax	24
<i>SASDRUGDEV_LOGOUT</i>	25
Description.....	25
Syntax	25
<i>SASDRUGDEV_SETCONTAINERPROPERTY</i>	25
Description.....	25
Syntax	25
<i>SASDRUGDEV_SETCONTAINERPROPS</i>	25
Description.....	25
Syntax	25
<i>SASDRUGDEV_SETFILEPROPERTY</i>	26
Description.....	26
Syntax	26
<i>SASDRUGDEV_SETFILEPROPS</i>	26
Description.....	26
Syntax	26
<i>SASDRUGDEV_SETGROUPMEMBERS</i>	27
Description.....	27
Syntax	27
<i>SASDRUGDEV_SETUSERAUTH</i>	27
Description.....	27
Syntax	27
<i>SASDRUGDEV_SETUSERPOLICIES</i>	28
Description.....	28
Syntax	28
<i>SASDRUGDEV_SETUSERPROPERTY</i>	28
Description.....	28
Syntax	28
<i>SASDRUGDEV_SETUSERPROPS</i>	29
Description.....	29
Syntax	29
<i>SASDRUGDEV_SETUSERSTATUS</i>	29
Description.....	29

<i>Syntax</i>	29
<i>SASDRUGDEV_USEREXISTS</i>	30
<i>Description</i>	30
<i>Syntax</i>	30
<i>Examples</i>	30
<i>Introduction</i>	30
<i>Create Files and Content Objects</i>	30
<i>Create a Container Object</i>	31
<i>Manipulate a SAS Data Set</i>	31
<i>Manipulate Metadata</i>	31
<i>Manipulate a User Group and Its Members</i>	32
<i>Get the User Accounts That Have a Specific System Policy</i>	32
<i>Create a User Account and Manipulate Its System Policies</i>	32
<i>Manipulate the Properties of a Content Object</i>	33

Introduction

The SAS Drug Development macros enable you to use familiar SAS syntax to make calls to SAS Drug Development. Your SAS programming skills and these macros enable you to perform certain operations.

From a programming perspective, the folders and files in the SAS Drug Development repository are containers and files. The names of the macros use this naming convention. However, in the SAS Drug Development application, “containers” are “container objects” and “files” are “content objects.” Therefore, this chapter uses the terms “container object” and “content object.”

Macro Return Codes

After you execute a macro, the global variable `_SDDRC_` contains the return code, and the global variable `_SDDMSG_` contains any additional information.

Here are the specific values that can be returned by a macro:

<code>_SDDRC_</code> Value	Explanation
-99	The result is uninitialized
-23	No user group members were specified to be updated
-22	An invalid parameter was passed
-21	No system policies were specified to be updated
-20	No value was specified for the property
-19	The property format is invalid
-18	The login credentials are invalid
-17	The user account is retired
-16	The user account is inactive
-15	The password has expired
-14	The user account could not be authenticated
-13	The URL is malformed

SDDRC Value	Explanation
-12	The property does not exist for the object
-11	The system policy does not exist
-10	The user account does not exist
-9	The user group does not exist
-8	An unexpected error was encountered
-7	The property does not exist
-6	A protected system property
-5	Invalid access permissions were encountered
-4	The object does not exist
-3	The object type is invalid for this macro
-2	No valid session exists
-1	The session is no longer valid
0	The macro executed without error
1	The object exists
2	No properties were updated

Note: A macro always returns a code, but a macro does not necessarily return every one of these codes.

Macros and System Policies

The SAS Drug Development system policies control the macros that are functional for you. For example, to change a user account with a macro, you must have the system policy **User can manage user accounts**.

If you call a macro without the required system policy, the macro fails. The macro returns a code and an error message.

For more information about the system policies, see the online Help.

The Macros

Using the Ampersand Character (&) in URLs

For a macro with a parameter that specifies a URL, such as a macro that sets properties, you cannot embed the ampersand character (&) in the URL. The ampersand character is a special character in SAS. If you embed an ampersand character, SAS will attempt to resolve the subsequent text as a macro variable.

The Proper Case for Parameter Values

Although SAS is case insensitive, the parameter values passed by the SAS Drug Development macros are case sensitive. This chapter presents the parameter values in the case that they must be passed to the macros.

SASDRUGDEV_CONTAINEREXISTS

Description

Determines whether a container object exists.

Syntax

```
%SASDRUGDEV_CONTAINEREXISTS(SDDPATH=sdd-path);
```

sdd-path is the path (starting at the root) and name of the container object in the SAS Drug Development repository.

SASDRUGDEV_COPYCONTAINER

Description

Copies a container object (and the objects it contains) from one location in the SAS Drug Development repository to another location in the repository.

Syntax

```
%SASDRUGDEV_COPYCONTAINER(SRCPATH=source-path,  
DESTPATH=destination-path);
```

source-path is the path (starting at the root) and name of the container object in the SAS Drug Development repository to copy.

destination-path is the path (starting at the root) and name of the new container object in the SAS Drug Development repository to which to copy the container object.

SASDRUGDEV_COPYFILE

Description

Copies a content object from one location in the SAS Drug Development repository to another location in the repository.

Syntax

%SASDRUGDEV_COPYFILE(SRCPATH=*source-path*, DESTPATH=*destination-path*);

source-path is the path (starting at the root) and name of the content object in the SAS Drug Development repository to copy.

destination-path is the path (starting at the root) and name of the new content object in the SAS Drug Development repository to which to copy the content object.

SASDRUGDEV_CREATEGROUP

Description

Creates a user group in SAS Drug Development.

Syntax

%SASDRUGDEV_CREATEGROUP(SDDGROUPNAME=*name*, SDDGROUPDESC=*description*);

name is the name of the user group.

description is the description of the user group.

SASDRUGDEV_CREATELOCALFILE

Description

Copies a content object from the SAS Drug Development repository to the computer on which SAS is executing.

Syntax

%SASDRUGDEV_CREATELOCALFILE(LOCALPATH=*local-path*, SDDPATH=*sdd-path*);

local-path is the absolute path and name of the file to create on the local computer on which SAS is executing.

sdd-path is the path (starting at the root) and name of the content object in the SAS Drug Development repository to copy.

SASDRUGDEV_CREATE_SDDCONTAINER

Description

Creates a container object in the SAS Drug Development repository. All parent folders must already exist.

Syntax

```
%SASDRUGDEV_CREATE_SDDCONTAINER(SDDPATH=sdd-path)  
<, TYPE=type>;
```

sdd-path is the path (starting at the root) and name of the container object to create in the SAS Drug Development repository.

type is the type of container object to create. These are the valid values:

- compound
- folder (default)
- indication
- protocol
- trashcan

For information about the valid values for the type of container object, see Appendix 2, “Reference.”

SASDRUGDEV_CREATE_SDDFILE

Description

Copies a file from the computer on which SAS is executing to the SAS Drug Development repository.

Syntax

```
%SASDRUGDEV_CREATE_SDDFILE(LOCALPATH=local-path, SDDPATH=sdd-  
path<, TYPE=type>;
```

local-path is the absolute path and name of the file.

sdd-path is the path (starting at the root) and name of the content object in the SAS Drug Development repository to which to copy the file.

type is the type of content object to create.

For information about the valid values for the type of content object, see Appendix 2, “Reference.”

