



D3 on jBASE Cheat Sheet

A jBASE Getting Started Guide

Download

Visit www.zumasys.com/downloads and select the appropriate version for your operating system. Additional installations instructions can be found here.

Docker Container

The quickest way to get started with jBASE is to visit Play with Docker, a free public cloud environment utilizing Docker Containers.

```
docker run -it -d --name=jbase zumasys/jbase:latest
docker exec -it jbase bash
```

Entering jBASE

You can enter the jBASE jShell from the o/s command prompt by typing "jb". You will be prompted for an account. The default Admin account for jBASE is JBSEADM.

```
# jb
Account name or path: JBSEADM
Jsh JBSEADM ~ -->
```

Configuration

jBASE uses environment variables to configure most items. Most items are set by the profile at JBCGLOBALDIR/config/jbase_config.json. Account specific items are modified by LOGTO and what is found in the System File.

Environment Variable	Description	Set By
JBCRELEASEDIR	jBASE installation location	jb
JBCGLOBALDIR	Location for configuration items	P*
JBCDATADIR	Default location for Accounts	P
JEDIFILENAME_SYSTEM	Location of the System file.	P
JBCSPOOLERDIR	Spooler location	P
JBC_OLD_SP_EDIT	Set to 1 for D3 style SP-EDIT	P/S
JEDIENABLEQ2Q	Set to 1 for D3 style Q Pointers	P/S(36)
JEDI_ERRMSG_TRACE	Set to 1 for RUNTIME-ERRORS	P
JEDI_ERRMSG_ZERO_USED	Set to 18 for D3 Behavior	P/S(36)
JEDI_ERRMSG_NON_NUMERIC	Set to 18 for D3 behavior	P/S(36)
JEDI_ERRMSG_DIVIDE_ZERO	Set to 18 for D3 behavior	P/S(36)
Z_DISABLE_QUEUEING	Set to 1 for D3 style phantoms	P/S(36)
JBCEMULATE	Emulation to use	P/S(21)
PATH	Location to find jBASE programs	P/S(27)
JEDIFILENAME_MD	Location of the MD file to use	P/S(28)
JEDIFILEPATH	Location(s) for jBASE files	P/S(27)
JBCOBJECTLIST	Location(s) for jBASE Subroutines	S(26)
JBCDEV_BIN	Where to catalog programs	P/S(23)
JBCDEV_LIB	Where to catalog Subroutines	P/S(24)

P=JBCGLOBALDIR/profile P*=JBCRELEASEDIR/profile S=System entry for account

Accounts

An account in jBASE is a directory. All files created for that account will be in that directory. When an account is created a SYSTEM entry will be created. Emulation will be taken from the account you created it from unless you use the -e setting. The default location for accounts is JBCDATADIR.

```
jsh JBSEADM ~ -->CREATE-ACCOUNT -e d3 MYAPP
```

If you wish D3 style case insensitivity with the system file do the following

```
jsh JBSEADM ~ --> jrf -U -V <path to JEDIFILENAME_SYSTEM>
```

Files

CREATE-FILE MYFILE TYPE=JD	Creates dynamic hash file (dynamically grows)
CREATE-FILE MYDIR TYPE=UD	Creates directory file
CREATE-FILE BP TYPE=JD	Creates hash dictionary, directory data, and a BP,OBJECT directory sub file
CREATE-FILE PIBS TYPE=PROC	Create D3 like Pibs file (readonly)
jstat -v MYFILE	Information on file (istat)
jrf	Resize/Change file
jchmod	Change Attributes (Att 1 in D3)
jstat	See file distribution (istat)

Editors

jBASE offers a classic ED editor and a full screen editor jED. The jED editor is a mix of ED and a full screen editor. jED Quick Help:

Full Screen	Arrow Keys to move. Ctrl DD to delete a line. Escape to return to command prompt.
-------------	---

Command Prompt at Top	Most ED commands work. Enter or a line # from the command prompt puts you back in full screen. F1 has additions such as FIBC to compile and catalog the item.
-----------------------	---

Lists

jBASE by default stores lists in JBCGLOBALDIR/tmp/jBASEWORK. If a POINTER-FILE exists it will store lists in that file. It is recommended for D3 style to CREATE-FILE POINTER-FILE in the JBSEADM account and then add Q pointers to your other accounts.

VOC/MD

jBASE supports a MD and is defined by JBCFILENAME_MD. You can change an existing MD that is NOT case insensitive to be case insensitive with the jrf command.

Q	Classic Q pointer
PA	A paragraph (script of commands)
PQ	A Proc
N	D3 style script
F	Universe style F pointer (Q pointer alternative)

Dictionaries

jBASE supports the classic A and S style D3 dictionaries and the Universe style I and D types. D3 style subroutine calls are also supported. Compatibility with the ACCESS() var is achieved via the qbasiccommonpick include

```
SUBROUTINE CALLXPROG(VALUE)
INCLUDE qbasiccommonpick
ID=ACCESS(10)
FILEVAR=ACCESS(1)
REC=ACCESS(3)
RETURN
```

Emulation Options

Emulation in jBASE is controlled by environment variables set within the profile and \$JBCGLOBALDIR/config/Config_EMULATE. You can see your current settings with

```
jsh -> config-strings -a
```

Indexes

jBASE Indexes are created by default off dictionary items vs A items.

CREATE-INDEX MYFILE DICTIONARY	Create index from dictionary
CREATE-INDEX FN A1 BY 1	Create index called a1 on attribute 1
REBUILD-INDEX, VERIFY-INDEX	Rebuild index(s)
LIST-INDEX -v MYFILE	List all indexes
QUERY-INDEX	Query a Index
OPENINDEX	BP open index
SELECT,READNEXT,READPREV	BP index cruising – Replaces Root/Key

Spooler

SP-ASSIGN	Defines a form queue, spooler options and number of copies to be used by subsequent print reports
SP-NEWTAB	Creates new Spooler (clears entries)
SP-CREATE F1 PROG lp -Dprinter	Creates form/printer 1 to lp printer
SP-MODIFY F1 PROG cat > test.txt	Modifies F1 to print to file
SP-EDIT	Same
SP-STATUS	Spooler Menu Maintenance
LISTPTR	List Printers
LISTPEQS	List jobs
PEQS – does not exist as a jBASE verb	All hold jobs are O/S items in JBCSPOOLERDIR/jobs.
SETPTR	Set the line printer spooler options for a logical print channel.

Note: System(20) stores last entry #

Locks

SHOW-ITEM-LOCKS	List all record locks
LIST-LOCKS	Shows all locks
CLEAR-ITEM-LOCKS	Unlock item(s) FILENAME ITEMNAME

Development

BASIC BP PROG	Creates intermediate object. Must Catalog to use.
CATALOG BP PROG	Creates item in JBCDEV_BIN. No MD entry.
jshow -c sub	Locate subroutine
Jshow -p prog	Locate program
RUN BP PROG	Has restrictions. See docs.
EXECUTE "!"	Supported but not needed
OPEN "/tmp"	Open O/S files directly.

%open,%read,%write,%close	OPENSEQ, READSEQ/BLK, WRITESEQ, CLOSESEQ
OPEN "account,file,"	Supported
JED FIBC	Files, does a basic and catalogues.
Break on/off	Use EXECUTE "BREAK-ON/OFF"

Debugger

PROGRAM -Jd	Run program and invoke the debugger
w	Shows lines of code around break point
s/S	Single Step (S skips gosubs and subroutines)
G	Continue
#	Jump to line
/var	Show variable
V -m var	Allow changing value of Var

Administration

jlicensinginfo -v	Show Licensing
jdiag -v	Show diagnostics
where (v	Detailed information on a port
SP-EDIT	Same
SP-STATUS	Spooler Menu Maintenance
LISTPTR	List Printers
LISTPEQS	List jobs
PEQS	All hold jobs are O/S items in JBCSPoolerDir/jobs.
jsh → !<command>	Optional. Affects jsh parsing.
LIST /tmp	List local O/S files. NL conversion is per Platform.
Z,ZH,PH-START,PH-STATUS	Phantom commands
GLOBALDIR/config/PH-HISTORY	Phantom History File
Hot-Backup	Transaction Journaling/Replication
OSFI	jRFS
openDB	ODBC Jedi
ODBC Server	jAgent/ODBC
MVS.NET/FlashConnect	jAgent/jRemote/MVConnect

BASIC

emulate_d3_internal_values	If set in current emulation, some internal functions are mapped out using a d3 scheme
See also:	https://docs.zumasys.com/jbase/jbc/system-functions/
@ Variables	https://docs.zumasys.com/jbase/jbc/the-@-function/
SYSTEM(1008)	Returns SYSTEM file path
SYSTEM(1009)	Returns MD file path

Command Shells

jsh	Jshell, equivalent to TCL (F1, <Esc>OP, jshelltype jsh) Pre-processes meta characters like the asterisk (*), as expected by legacy multi-value systems. Note that only in jsh mode is JEDIFILENAME_MD examined for Q-pointers, Procs and Paragraphs.
sh	Native platform shell (F2, <Esc>OQ, jshelltype sh) Pre-processing of meta-characters depends on operating system.
msh	Mixed shell (F3, <Esc>OR, jshelltype msh) Pre-processes meta characters as a combination of jsh and sh.

Dot stacker

In jBASE, Up and Down arrows can be used, as well as Left and Right arrows.

```
jsh SandBox ~ -->.
```

User-based command stack.

Maximum number of stack entries: 500

Current number of stack entries: 98

```
?.? | .h Display help
```

```
a.(#)text Append 'text' to entry '#'  
[if '#' omitted then appends to entry 1]
```

```
c.#/s1/s2/* Change 's1' to 's2' in entry '#',  
* = change all stack entries
```

```
d.# Delete entry number '#'
```

```
f text Find first occurrence of 'text', move to top of stack
```

```
.l List first page
```

```
.ln List first 'n' lines
```

```
.lm-n List entries 'm' through 'n' inclusive
```

```
.m# Move entry number '#' to top of stack
```

```
.p# text Prepend 'text' to entry '#'
```

```
.r#/s1/s2/* Replace 's1' with 's2' in entry '#', * = replace all
```

```
s text Search and display all entries with 'text'
```

```
{x}# Execute entry '#'
```

```
[if 'x' is omitted then '#' is required]
```

```
.z Zap [clear] the command stack
```

If '#' is '0' or omitted, this implies entry 1.