Rexx Vs CLIST

by H. Fosdick

These charts are for those who know one of these languages and want to learn about the other. They may also be useful for conversions.

The Basics

	Rexx	CLIST
Easy to learn, use, and maintain	Yes	Yes
Very powerful	Yes	No (lacks common language features)
Open source	Yes	No
Portable	Yes	No
Runs on all platforms	Yes	No
Runs as the OS Shell	No	Yes
Interfaces to tons of tools	Yes	No (intended to issue OS commands)
ANSI or ISO Standard	Yes (ANSI-1996)	No

Profiles

	Rexx	CLIST
Dialects	TRL-2, ANSI, Mainframe, ooRexx, NetRexx	CLIST
Unique Usage	* Default scripting language for mainframes and several minor platforms * Interfaces to all mainframe environments and address spaces	* Available on all z/OS mainframes
Programming paradigms	Procedural, scripting, object-oriented (ooRexx and NetRexx), functional	Procedural, scripting
OOP: classes, objects, multi-inheritance, polymorphism, encapsulation	In ooRexx and NetRexx	Unsupported
User Group	Rexx Language Association	SHARE
Quick Online Lookup	Quick Lookup	IBM TSO CLISTs Manual
Cheat Sheet (printable PDF)	ANSI Rexx, Mainframe Rexx	Only this one
Forum	RexxLA forum	Expert Forum @ IBMMainframes.com
Further information	RexxInfo.org	IBM TSO CLISTs Manual

Language Comparison

	ANSI Rexx	CLIST
Format	Free form	Free form
Case-sensitive	No	Yes
Variable Names	Valid Symbol that does not start with a digit (0-9) or a period (.)	Start with & Next character must be one of: A-Z, (a-z), _, #, $, @
Comments	Enclose inside /* and */	Enclose inside /* and */. Or put comment at end of a line by starting it with /*
Line Continuation	, (comma)	- (minus sign) or + (plus sign)
Statement Separator	; (semi-colon)	None (implied by line end)
Repeated line scanning for substitution	No (use INTERPRET instruction)	Yes
Code Blocks	Define by do - end	DO-END, SELECT-END (Also DATA-ENDDATA, DATA PROMPT-ENDDATA)
Undefined Variables	Allowed. Use SYMBOL to determine if a variable has been defined	Allowed until misused
Assignment Operators	=	SET =
Arithmetic Operators	+ - * / % ** //	+ - * / ** //
Comparison Operators	== \== >> << >>= \<< <<= \>> = \= <> >< > < >= \< <= \> (\ can be replaced with ¬ in any of these)	= EQ ¬= NE < LT > GT <= LE >= GE ¬> NG ¬< NL
Logical Operators	& | && \ (prefix) ¬ (prefix)	AND && OR |
Concatenation Operators	|| Or, concatenate with blank between Or, concatenate by abuttal (no blank)	By abuttal: SET VARIABLE=&VAR1&VAR2
Bitwise Operators	Use built-in functions	Unsupported
Membership Operators	Unsupported	Unsupported
Regular Expressions	Use RexxRE Regular Expressions external Library	Unsupported
Built-in Functions	About 70 functions	15 functions, plus about 55 Control Variables
Data Types	Everything's a string, types are reflected in usage	Defined by usage
Function to Check Data Type	datatype	&DATATYPE
Collections of Variables	Use compound variables	Unsupported, you would have to program this

Associative Arrays	Use compound variables	Unsupported
Multidimensional Arrays	Use compound variables	Unsupported
Stack & Queue Operations	Yes (push, pull, parse pull, queue, queued)	Unsupported as a generalized feature, but can manage the terminal input buffer as a stack
Decimal Arithmetic	Default	Unsupported
Flow of Control	if, do, select, call, exit, return, iterate, leave, signal, nop	IF-THEN-ELSE, DO, SELECT, GOTO, EXIT, RETURN, SYSCALL, ATTN, ERROR, TERMIN, TERMING
GOTO	none (use signal)	GOTO
Calling Subroutine	call	SYSCALL
Manage Scope of Variables to Subroutines	procedure expose	Use PROC
Subroutine End	return	END
Getting Return Code	special variable RC RESULT set by subroutine RETURN	&LASTCC, &MAXCC for highest return code set
Trace Script Execution	trace (instruction), trace (function)	Use CONTROL statement
Show Full Execution	trace a	CONTROL LIST
Exception Handling	signal	ERROR, ATTN
Standard Exceptions	novalue, error, failure, halt, notready, syntax, lostdigits	ATTN (attention key pressed), ERROR (traps any of more than 100 error conditions)
Attention Routines	signal on halt	ATTN
Run an Operating System Command	Just issue the command string (Rexx passes unrecognized strings to the default active environment)	Just issue the command
Terminate Process	exit	EXIT
End with Return Code	exit 8	EXIT(8)
Get User Input	say "Enter your name:" parse pull name	WRITE Enter your name: READ NAME
Read Input Record	linein execio (mainframe only, not ANSI)	GETFILE
Write Output Record	lineout execio (mainframe only, not ANSI)	PUTFILE
Array Read/Write	execio (mainframe only, not ANSI)	Unsupported
Detecting File EOF	Sets return code	Treats as a detected error
Write without line feed	Unsupported	WRITENR

Get Date and Time	date and time functions	&SYSDATE, &SYSTIME
Get Length of a String	length("MYSTRING")	&LENGTH(MYSTRING)
Get a Substring	substr("MYSTRING",1,4)	&SUBSTR(1:4,MYSTRING)

Based on *Rexx Programmer's Reference* and *IBM TSO CLISTs Manual*.