Rexx Vs Python

by H. Fosdick

For those who know one of these languages and want to learn about the other.

The Basics

	Rexx	Python
Easy to learn, use, and maintain	Yes	Yes
Very powerful	Yes	Yes
Open source	Yes	Yes
Portable	Yes	Yes
Runs on all platforms	Yes	Yes
Interfaces to tons of tools	Yes	Yes
ANSI or ISO Standard	Yes (ANSI-1996)	No (de facto standard by PEPs)

Profiles

	Rexx	Python
Dialects	TRL-2, ANSI, Mainframe, ooRexx, NetRexx	Version 2.x, Version 3.x, specialized implementations
Unique Usage	* Default scripting language for mainframes and several minor platforms * Interfaces to all mainframe environments and address spaces	* Most popular programming language in the world * Most widely taught language * Ships with many OS's
Programming paradigms	Procedural, object-oriented (ooRexx and NetRexx), functional, scripting	Procedural, object-oriented, functional, scripting
OOP: classes, objects, multi-inheritance, polymorphism, encapsulation	Yes, in ooRexx and NetRexx	Yes
User Group	Rexx Language Association	Python community
Quick Online Lookup	Quick Lookup	QuickRef
Cheat Sheet (printable PDF)	ANSI Rexx, Mainframe Rexx	Python 3, Cheatography
Forum	RexxLA forum	Python forums
Further information	RexxInfo.org	Python.org

Language Comparison

	ANSI Rexx	Python
Formatting	Free form	Based on indentation rules
Case-sensitive	No	Yes
Comments	Enclose inside /* and */	Start line with # or enclose inside triple quotes (""")
Line Continuation	, (comma)	\ (backslash) or implicit
Statement Separator	; (semi-colon)	; (semi-colon)
Code Blocks	Define by do - end	Define by : (colon) plus indentation
Undefined Variables	Allowed. Use SYMBOL to determine if a variable has been defined	Undefined variable raises NameError. Circumvent this by assigning special constant None.
Assignment Operators	=	= += -= *= /= %= //= **= := Bitwise: &= |= ^= >>= <<=
Arithmetic Operators	+ - * / % ** //	+ - * / % ** // Compound: += -= *= /= %= //= **=
Comparison Operators	== \== >> << >>= \<< <<= \>> = \ = <> >< > < >= \< <= \> (\ can be replaced with ¬ in any of these)	== != > < >= <= Object Identity: is is not
Logical Operators	& | && \ (prefix) ¬ (prefix)	and or not
Concatenation Operators	|| Or concatenate with blank between Or, concatenate by abuttal (no blank)	+ += * (multiple copies of a string) Concatenate literal strings with blank beween. Use f-strings. Use join or format methods
Bitwise Operators	Use built-in functions	& | ^ ~ << >> Compound: &= |= ^= <<= >>=
Membership Operators	Unsupported	in not in
Regular Expressions	Use RexxRE Regular Expressions external Library	Use the re module
Built-in Functions	About 70 functions	About 70 functions
Data Types	Everything's a string, types are reflected in usage	A dynamically typed language with these data types: text (str), numeric (int, float, complex), sequence (list, tuple, range), mapping (dict), set (set, frozenset), boolean (bool), binary (bytes, bytearray, memoryview)
Function to Check Data Type	datatype	type

Collections of Variables	Use compound variables	list, tuple, range, set, frozenset, dict, bytearray
Associative Arrays	Use compound variables	Use a dictionary (dict)
Multidimensional Arrays	Use compound variables	Use NumPy library
Stack & Queue Operations	Yes (push, pull, parse pull, queue, queued)	Yes (queue module and deque from collections)
Decimal Arithmetic	Default	Use the decimal module
Flow of Control	if, do, select, call, exit, return, iterate, leave, signal, nop	if, for, while, break, continue, pass, return, exit, quit, match-case
Trace Script Execution	trace (instruction), trace (function)	Use the trace or pdb modules
Terminate Process	exit	exit()
Get User Input	say "Enter your name:" parse pull name	name = input("Enter your name: ")
Exception Handling	signal	try-except-else-finally, raise, with
Standard Exceptions	novalue, error, failure, halt, notready, syntax, lostdigits	SyntaxError, TypeError, NameError, IndexError, KeyError, ValueError, AttributeError, IOError, ZeroDivisionError, ImportError
Run an Operating System Command	Just issue the command string (Rexx passes unrecognized strings to the default active environment)	Use the subprocess module (or older os module)

Based on *Rexx Programmer's Reference*, *Python Tutorial*, and *Python Language Reference (v 3.12.3)*.