Python Data Structures

Single-value data structures (all immutable): int float bool

(We often treat **string** information—textual data—as a single value, though it's technically multiple-value as shown below.)

Type	Constant values	Selected operators	Selected functions	Uses
int	0 -17 238653489	+ - * / % // ** == != > < >= <=	abs() max() min()	Counting things; indexing strings or lists
float	3.0 -2.00053 3.5e19	+ - * / % // ** == != > < >= <=	abs() max() min()	Calculating real-world quantities
bool	True False	and or not == !=		Results of comparisons, yes-or-no questions

Multiple-value data structures: string list namedtuple file dict set tuple

Type	Mutable (can change in place)?	Sequence (are items in order)?	Constant values	Selected operators and functions	Selected methods	Main characteristics	Uses

string	No	Yes	'' 'Four score and 7 ...'
			"We the people ..."
			''' Multi-line string can
			span many lines '''
				in + * [ ] [:]
				len()
				== != > < >= <=
				int() float()
					S.find()
					S.split() S.strip()
					S.format()
					S.translate()
						Indexable sequence of characters, can take "slices" (substrings)	Text, names, words, "ASCII graphics" like
							************

list (also called array)	Yes	Yes	[ ] [1, 2, 3] ['Hello!']	in + * [ ] [:]
				len() min() max()
				== != > < >= <=
					L.count() L.sort()
					L.append()
					L.extend()
						Indexable sequence, can take "slices"	General collection of same-type objects

namedtuple (also called structure or record)	No	Yes	Restaurant('Taillevent',
			'French', '333-4444',
			'Gateau Marjolaine',
			33.00)
				. [to specify field/attribute]
				print(R.name, 'has',
				R.dish, 'for $',
				R.price)
					R = R._replace
					(name='Noma')
						Access to components by attribute/field name	from collections
							import namedtuple
							Grouping components of an object (e.g., student name, ID, major, GPA)

file	No	Yes		inf = open('blah.txt',
				'r')
				outf = open('ans.txt',
				'w')
					nextline =
					inf.readline()
					linelist =
					inf.readlines()
					outf.writelines()
					outf.close()
						Files give you **persistence:** Their contents stay around after the program stops running.	Saving data from one run to the next; preparing input to another program; packaging data to send over the net

dict (also map, table, dictionary)	Yes	No	{ }
			{'Joe':25, 'Jan': 47,
			'Jill':44, 'Jim': 45}
				in [ ]
				len()
				== != > < >= <=
					d.items() #view
					d.keys() #view
					d.values() #view
						No order; look up value by key	Programmer-defined indexes, fast lookup

set	Yes	No	set() {'a', 'b', 'c'}	in len() >= <= | &	S.add()
					S.remove()
						No order, no duplicates	Eliminate duplicates

tuple	No	Yes	( ) ('Joe', 'Smith', 19, 'Informatics') ('Donald',) # One item	in [ ] len() == != > < >= <=	T.count()	Grouped data, access by position	Short multi-part data packages where field names not needed