Top of Form



Bottom of Form

Top of Form



**Python 2: Getting More Out of Python  
Lesson 5, Quiz 2**

Handed in: 26 Feb 2015 07:02:55PM Graded: 2 Mar 2015 04:30:28PM

**Question 1:**

What argument to shelve.open() will activate in-memory caching?

**Your Answer:**

writeback=True

**Mentor Comments:**

Shelves do not track modifications to volatile objects, by default. So if contents of an item stored in the shelf are changed, the shelf must be updated explicitly by storing the entire item again. To automatically catch changes stored in a shelf, enable the writeback flag. Setting this flag to True causes the shelf to remember all objects retrieved from the database using an in-memory cache. Each cache object will be written back to the database when the shelf is closed. This will reduce errors and make object persistence more transparent, but it could impose a performance penalty since the cache consumes extra memory while the shelf is open and pausing to write objects back to database.

**Question 2:**

What persistent format would you recommend if the data must be read by JavaScript as well as Python programs?

**Your Answer:**

json

**Mentor Comments:**

If we want a human readable, cross-platform and cross-language serialization format, we can use JSON. JSON is actually a subset of JavaScript's object literal syntax. Although it was derived from JavaScript, json parsers exist for many languages. In fact, Python 3 comes with a built-in JSON parser.

JSON (JavaScript Object Notation), specified by RFC 4627, is a lightweight data interchange format based on a subset of JavaScript syntax. It is a text format that is completely language independent but uses conventions that are familiar to programmers of the C-family of languages (C, C++, C#, Java, Perl, Python)

The json module provides an API similiar to pickle for converting in-memory Python objects to a serialzed representation known as JSON. Unlike pickle, JSON has the benfit of being implemented in many languages. It is most widely used for communicating between the web server and the client .

JSON has become an important protocol / format with the advent of AJAX i.e. web pages that update without needing to be reloaded (example gmail). Python typically resides on the server and talks to JavaScript browsers. The

primitive data types are communicable in both directions thanks to JSON.

XML is another client-server transmission format understood across

languages. We should remember that HTML itself is close to an XML and may be served in strict XML format as a mimetype.

http://html5doctor.com/html-5-xml-xhtml-5/

**Question 3:**

What expression will serialize object x as an equivalent JSON string?

**Your Answer:**

json.dumps(x)

**Mentor Comments:**

*none*

**Overall Comments:**

Terrific work, Jason.

-Pat

**Grade:**

Great

[Take Me Back](ostreturn:)

© 2014, O'Reilly Media, Inc. All rights reserved.

Bottom of Form