Sample Template

1. Description

Available: 2004-09-15

This sample template is the recommended way of writing a sample. If you'd like to contribute, please do! Once you've written it, post a new Jira bug with component=Samples. For ideas, see XmlBeansSampleSuggestions on the wiki. Thanks for submitting!

2. Files

Zip of all files (SampleTemplate.zip)

Tar gzip of all files (SampleTemplate.tgz)

Readme file

3. Authoring Samples

If you have an idea for a sample that you'd like to share with the community, we'd be happy to review and add it to the collection! Because we want all of the samples available from this site to be useful (and you probably do, too), we'd like you to follow the guidelines on this page when writing an XMLBeans sample.

To contrubute the sample, please create a new Jira bug with component=Samples and attach the sample zip to the bug.

> **Note:**
> All sample code accepted for use on the XMLBeans web site becomes the property of the Apache Software Foundation. In other words, thanks for donating a sample!

3.1. Sample scope

Try to keep your sample reasonably small and well-defined. It's going to be more useful, and more easily found, if it's about something specific -- handling schema enums (or some other particular aspect of schema), handling validation errors, constructing mixed content with a cursor, and so on. Large samples that do lots of things are often hard to get to know and learn something useful from.

Your code will be most useful if it can be copied and pasted into someone else's app. However, schema dependencies will differ greatly from user to user, so samples that use generated types will tend to be fairly specific. Moreover, code that effectively demonstrates how a particular aspect of XMLBeans works is a good teacher, even if would take quite a bit of massaging to coerce it to another application.

3.2. Code comments

Obviously, comments are an essential part of making a sample useful. In general, the more comments your code has, the better. But here are a few guidelines that we ask you to keep in mind:

- Each sample file (including non-Java files) must include the Apache copyright notice. If it's not there when it's accepted, we'll add it.
- Each class and method should include a comment in the Javadoc style. It's not necessary to include specific Javadoc annotations (@param, @return, and so on), but your sample isn't as useful to others unless it's commented well.
- Class-level comments should describe what the class demonstrates. In particular, the comments should be sure to describe the XMLBeans technologies that the class includes. Other description text should describe what the sample's user should expect to see when they run the sample.
- Method-level comments should describe what the method does. Comments should focus on the role of XMLBeans technologies.
- Where useful, add more comments!

3.3. Code conventions

To help make XMLBeans samples as a set easier for others to use, we encourage you to follow a few simple code conventions. In this way, users can spend less time deciphering code and more time getting the things it demonstrates. Most of these are common practice anyway.

- Package name words should all be lower-case.
- Package names should begin with 'org.apache.xmlbeans.samples' and end with the sample's name, and should indicate what the sample illustrates. Examples are 'org.apache.xmlbeans.samples.anytype.*' or ' 'org.apache.xmlbeans.samples.dynamiccompilation.*'