XmlWriteParse仕様書

2011/09/06 garicchi バージョン1.0

XMLWriteParseクラスはXML文章の新規作成、タグの追加を実行します。または、読み込んだXml文章の指定要素を書き換えたり、属性値を書き換えたりすることができます。

・簡単な使い方

Import garicchi.xml.XmlWriteParse.XmlWriteParse;

でインポートします。

Xmlの書き換えを行う場合

XmlWriteParse write=new XmlWriteParse();

新しいインスタンスを生成します。

write.ReWriteElement();

タグ名と階層数、属性名などを引数にすることによってそのタグの要素を書き換えるメゾッドです。多数のオーバーロードがあります。詳細はドキュメントにあります。

1階層上のタグ情報まで引数に指定できます。

write.ReWriteAttributeValue();

タグ名と階層数、属性名などを引数にすることによってそのタグのしていた属性名の属性値を書き換えることができます。

サンプル

|  |
| --- |
| **package** jp.garicchi.xmlparsesample;  **import** garicchi.xml.XmlWriteParse.XmlWriteParse;  **import** java.io.IOException;  **import** org.xmlpull.v1.XmlPullParserException;  **import** android.app.Activity;  **import** android.os.Bundle;  **import** android.widget.TextView;  **public** **class** MainActivity **extends** Activity {  @Override  **public** **void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.main);  TextView **v**=(TextView)findViewById(R.id.TextView01);    //インスタンス生成  XmlWriteParse **write**=**new** XmlWriteParse();  //String型のXml文章  String **xml**="<?xml version=\"1.0\" encoding=\"UTF-8\"?><venture><company><name>楽天株式会社</name><url>http://www.rakuten.co.jp/info/</url></company></venture>";  String **reWriteXml**=**null**;  **try** {  //3階層の"name"タグの要素を"あほ"に書き換え  reWriteXml = write.ReWriteElement(xml,"1.0","UTF-8","name",3,"あほ");  } **catch** (IllegalArgumentException **e**) {  e.printStackTrace();  } **catch** (IllegalStateException **e**) {  e.printStackTrace();  } **catch** (XmlPullParserException **e**) {  e.printStackTrace();  } **catch** (IOException **e**) {  e.printStackTrace();  }    v.setText(reWriteXml);  }  } |

Xmlを新規作成する場合

Import garicchi.xml.XmlWriteParse.XmlWriteParse;

でインポートします。

Xmlの書き換えを行う場合

XmlWriteParse write=new XmlWriteParse();

新しいインスタンスを生成します。

write.NewXml();

Xmlバージョンとエンコードを指定してXmlを新規作成します。

この時xml宣言も作成してくれるのでこれ以降はタグを追加していきます

write.AddTag();

タグを追加します。要素を引数に指定する。オーバーロードメゾッドでは、単一タグとして追加し、終了タグを自動で追加します。

要素を引数に指定しないオーバーロードメゾッドではそれ以降は1階層深いタグとして、追加され、終了タグは追加されません。タグを終了するにはCloseTagメゾッドを使用します。

属性名と属性値を指定しない場合はnullを指定する。

write.CloseTag();

要素を引数に指定しないAddTagメゾッドでスタックに積み上げられた閉じられてないタグを一つ閉じます。この時1階層浅くなります。

write.CloseTagAll();

要素を引数に指定しないAddTagメゾッドでスタックに積み上げられた閉じられていないタグをルートタグまですべて閉じます。

write.GetWrittingXml();

書き込み完了したXmlを取得します。

サンプル

|  |
| --- |
| **package** jp.garicchi.xmlparsesample;  **import** garicchi.xml.XmlWriteParse.XmlWriteParse;  **import** java.util.ArrayList;  **import** android.app.Activity;  **import** android.os.Bundle;  **import** android.widget.TextView;  **public** **class** MainActivity **extends** Activity {  @Override  **public** **void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.main);  TextView **v**=(TextView)findViewById(R.id.TextView01);    ArrayList<String> **namelist**=**new** ArrayList<String>();  ArrayList<String> **valuelist**=**new** ArrayList<String>();  namelist.add("id");  namelist.add("code");  valuelist.add("01");  valuelist.add("003654");    //インスタンス生成  XmlWriteParse **write**=**new** XmlWriteParse();  write.NewXml("1.0","UTF-8");  write.AddTag("root",**null**,**null**);  write.AddTag("one","1階層",namelist,valuelist);  write.AddTag("tool",**null**, **null**);  write.AddTag("tag","深い",**null**,**null**);  write.CloseTagAll();  String **xml**=write.GetWrittingXml();  v.setText(xml);  }  } |