

分散システム

第五回レポート課題

課題 1

【課題内容】

ファイルから XML 文書を読み込んで DOM 木を生成し、その DOM 木のすべてのノードを探索して「ノードの名前」と「ノードの値」を出力するプログラムを作成せよ。ただし、

- 作成したソースコード
- 読み込ませた XML 文書ファイル(小さいものでよい)
- 実行結果(一部でもよい。スクリーンダンプにより)

を示すこと。また、実行結果には、実行した時刻を出力すること。

余力があれば、空白記号の処理方法を変更して、結果がどのように変わるかを確かめよ。

(DocumentBuilderFactory クラスの `setIgnoringElementContentWhitespace(boolean)` を利用)

【作成したソースコード】

DOMsample.java

```
import javax.xml.parsers.*;
import org.w3c.dom.*;

public class DOMsample {
    public static void main ( String[] args ) {
        System.out.println("current time = " + new java.util.Date());
        try {
            DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
            factory.setIgnoringElementContentWhitespace(true);
            DocumentBuilder parser = factory.newDocumentBuilder();
            //DOM 木を解析
            Document doc = parser.parse(args[0]);
            //ルート要素の取得
            Element e = doc.getDocumentElement();
            //深さ優先探索
            traverseTree(e);
        } catch ( Exception e ) {
            e.printStackTrace();
        }
    }
    static void traverseTree (Node node) {
        //ノードの名前と値を取得
        String name = node.getNodeName();
        String value = node.getNodeValue();
        //標準出力に表示
        System.out.println(name + " : " + value );
        Node n = node.getFirstChild();
        //子供がいるかチェック
        while (n != null) {
            traverseTree(n);
            n = n.getNextSibling();
        }
    }
}
```

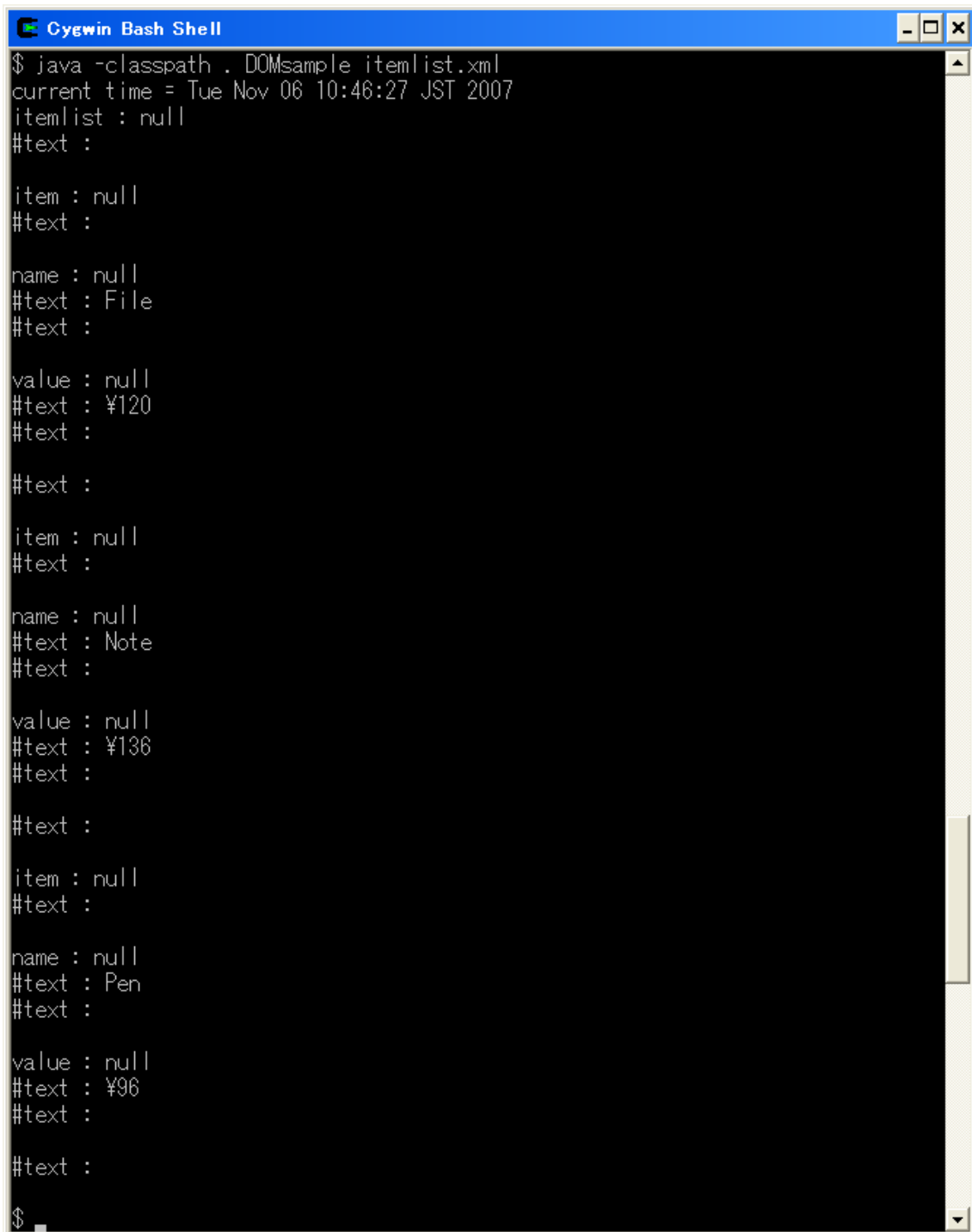
【読み込ませた XML 文書ファイル】

itemlist.xml

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE ItemList SYSTEM "itemlist.dtd">

<itemlist>
    <item>
        <name>File</name>
        <value>¥120</value>
    </item>
    <item>
        <name>Note</name>
        <value>¥136</value>
    </item>
    <item>
        <name>Pen</name>
        <value>¥96</value>
    </item>
</itemlist>
```

【実行結果】



```
Cygwin Bash Shell
$ java -classpath . DOMsample itemlist.xml
current time = Tue Nov 06 10:46:27 JST 2007
itemlist : null
#text :

item : null
#text :

name : null
#text : File
#text :

value : null
#text : ¥120
#text :

#text :

item : null
#text :

name : null
#text : Note
#text :

value : null
#text : ¥136
#text :

#text :

item : null
#text :

name : null
#text : Pen
#text :

value : null
#text : ¥96
#text :

#text :

$
```

空白処理は true にしても false にしても、出力結果に変化がなかった。

課題 2

【課題内容】

第二回レポートで作成した商品データの XML 文書を用いて、指定された商品名に対する価格を検索し、出力するプログラムを作成せよ。ただし、

- 作成したソースコード
- 読み込ませた XML 文書ファイル(小さいものでよい)
- 実行結果(一部でもよい。スクリーンダンプにより)

を示すこと。また、実行結果には、実行した時刻を出力すること。

【作成したソースコード】

DOMsearch.java

```
import javax.xml.parsers.*;
import org.w3c.dom.*;

public class DOMsearch {
    public static void main ( String[] args ) {
        System.out.println("current time = " + new java.util.Date());
        try {
            DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
            factory.setIgnoringElementContentWhitespace(true);
            DocumentBuilder parser = factory.newDocumentBuilder();

            //DOM 木を解析
            Document doc = parser.parse(args[0]);
            //標準入力の値を受け取る
            String word = args[1];
            //ルート要素の取得
            Element e = doc.getDocumentElement();

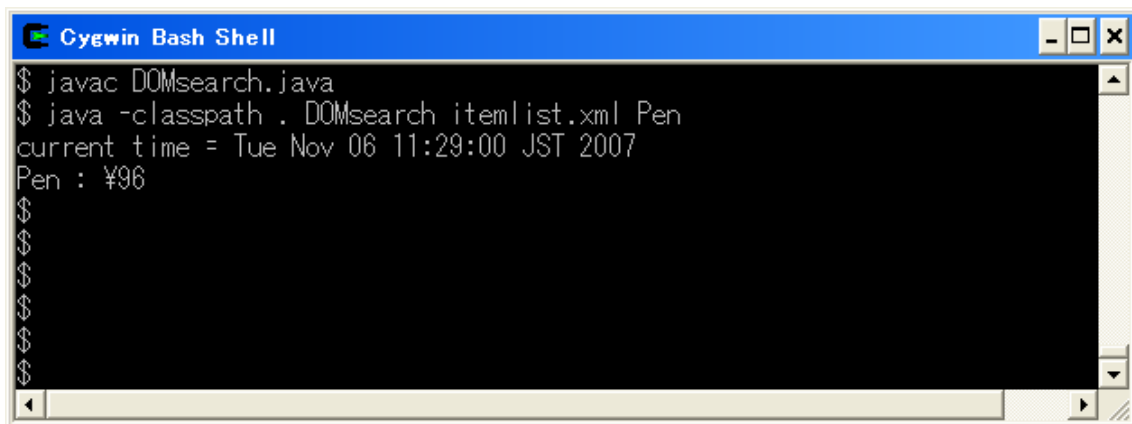
            traverseTree(e,word);
        } catch ( Exception e ) {
            e.printStackTrace();
        }
    }
    static void traverseTree (Node node, String str) {
        String value = node.getNodeValue();
        if ( str.equals ( value ) ) {
            Node cur = node.getParentNode();
            cur = cur.getNextSibling();
            String name = cur.getNodeName();
            if ( name.equals ( "value" ) ) {
                cur = cur.getFirstChild();
                value = cur.getNodeValue();
                System.out.println ( str + " : " + value );
            }
        }
        Node n = node.getFirstChild();
        //子供がいるかチェック
        while ( n != null ) {
            traverseTree(n,str);
            n = n.getNextSibling();
        }
    }
}
```

【読み込ませた XML 文書ファイル】

itemlist.xml

```
<?xml version="1.0" encoding="iso-8859-1"?>  
<!DOCTYPE ItemList SYSTEM "itemlist.dtd">  
  
<itemlist>  
  <item>  
    <name>File</name>  
    <value>¥120</value>  
  </item>  
  <item>  
    <name>Note</name>  
    <value>¥136</value>  
  </item>  
  <item>  
    <name>Pen</name>  
    <value>¥96</value>  
  </item>  
</itemlist>
```

【実行結果】



```
Cygwin Bash Shell  
$ javac DOMsearch.java  
$ java -classpath . DOMsearch itemlist.xml Pen  
current time = Tue Nov 06 11:29:00 JST 2007  
Pen : ¥96  
$  
$  
$  
$  
$  
$  
$
```