

参考サイト ;

http://blog.goo.ne.jp/glass-_-onion/e/6f86c9903ce7c75ebd5d56b827724961

<http://www.hellohiro.com/customtag.htm>

- 参考サイトをもとにリンクつけつつ、特定のアクションのみパンくずに表示するよう書いてみました

前提

- strutsベース
- actionクラスは一度baseのアクションを作成し、必ず継承する。

(1)baseアクションに以下のように記述

```
//サブシステムで実装するクラス。
protected abstract String execute(ActionForm frm);

public ActionForward execute(ActionMapping map, ActionForm frm,
    HttpServletRequest request, HttpServletResponse response) {
    String result = execute(frm);

    //パンくず処理開始
    ActionStack stack = (ActionStack)request.getSession().getAttribute("ForwardStack");
    //新規作成
    if(stack == null) {
        stack = new ActionStack();
    }
    //すでに登録済みかチェック
    int index = stack.search(request.getServletPath());

    if(index == -1) {
        //-1=未登録
        //新規push
        stack.push(request.getServletPath());
    } else {
        //既登録
        //登録してある箇所から後ろを除去
        for(int i = 0; i < index - 1; i++) {
            stack.pop();
        }
    }
    //セッションに設定
    request.getSession().setAttribute("ForwardStack", stack);
    return map.findForward(result);
}
```

(2)ActionStack.javaを作成

```
import java.util.Iterator;
import java.util.Stack;

public class ActionStack {
    Stack forwardStack = new Stack();
}
```

```

Stack forwardNmStack = new Stack();
//パンくず対象
private static final String[] targetActionArr = {"/topInit.do","/search.do","/detailInit.do"};

private static final String[] targetActionNmArr = {"トップ画面","検索画面","詳細画面"};

public String pop() {
    forwardNmStack.pop();
    return (String)forwardStack.pop();
}
public void push(String path) {
    String actionNm = canPush(path);
    if(null != actionNm){
        forwardStack.push(path);
        forwardNmStack.push(actionNm);
    }
}

public boolean empty() {
    return forwardNmStack.empty();
}
public int search(String path) {
    int index = forwardStack.size();
    for(Iterator i = forwardStack.iterator(); i.hasNext(); index--) {
        String forward = (String)i.next();
        if(path.equals(forward)) {
            return index;
        }
    }
    return -1;
}
public Iterator iterator() {
    return forwardNmStack.iterator();
}
public int size() {
    return forwardNmStack.size();
}

private String canPush(String path){
    int i = 0;
    for(String targetAction:targetActionArr){
        if(targetAction.equals(path)){
            StringBuilder buf = new StringBuilder();
            buf.append("<a href= \"\".+path+\" \">");
            buf.append(targetActionNmArr[i]+ "</a>");
            return buf.toString();
        }
        i++;
    }
    return null;
}
}

```

(3)TopicPathTag.javaを作成

```

import java.io.IOException;
import java.util.Iterator;

import javax.servlet.jsp.JspException;
import javax.servlet.jsp.JspWriter;
import javax.servlet.jsp.tagext.TagSupport;

import org.apache.struts.taglib.TagUtils;
//ActionStackもインポート

@SuppressWarnings("serial")
public class TopicPathTag extends TagSupport {

    public int doStartTag() throws JspException {
        ActionStack stack = (ActionStack)TagUtils.getInstance().lookup(
            pageContext, "ForwardStack", "session");
        StringBuilder buf = new StringBuilder();
        for(Iterator i = stack.iterator(); i.hasNext();) {
            String forward = (String)i.next();
            buf.append(forward);
            buf.append(" > ");
        }
        buf.deleteCharAt(buf.lastIndexOf(">"));
        try {
            JspWriter out = pageContext.getOut();
            out.print(buf);
        } catch (IOException e) {
            throw new JspException(e);
        }
        return SKIP_BODY;
    }
}

```

(4)TLDファイルを作成(ファイル名は任意)

```

<?xml version="1.0" encoding="Shift_JIS" ?>
<!DOCTYPE taglib PUBLIC "-//Sun Microsystems, Inc.//DTD JSP Tag Library 1.1//EN" "http://jav
<taglib>
<tlibversion>1.0</tlibversion>
<jspversion>1.1</jspversion>
<shortname>org</shortname>

<tag>
<name>TopicPath</name>
<!-- タグクラスを指定 -->
<tagclass>jp.co.sample.TopicPathTag</tagclass>
<bodycontent>empty</bodycontent>
<info>パンくず</info>
</tag>

</taglib>

```