

# Java Web システム ショッピングサイト

システム名	KEN ショッピングサイト
処 理 名	フロントコントローラー
プログラム名	Control
パッケージ	ken

*KEN School*

処理名	フロントコントローラー	クラス名	Control
業務概要説明			
<p>【処理概要】</p> <ul style="list-style-type: none"><li>● リクエストを受け取り、機能ごとにどのクラスを使用し具体的な処理をし、どの JSP で表示をするか等、動作を制御する</li></ul>			

処理名	フロントコントローラー	クラス名	Control		
クラス詳細（クラス設計/フィールド設計）					
【クラス設計】					
項番	項目	値			
1	package	ken			
2	class/interface/jsp	class			
3	アクセス修飾子	public			
4	修飾子	－			
5	継承元	HttpServlet			
6	実装元	－			
7	使用クラス/インタフェース	java.io.FileInputStream; java.io.IOException; java.util.Properties; javax.servlet.RequestDispatcher; javax.servlet.ServletConfig; javax.servlet.ServletContext; javax.servlet.ServletException; javax.servlet.annotation.WebServlet; javax.servlet.http.HttpServlet; javax.servlet.http.HttpServletRequest; javax.servlet.http.HttpServletResponse; ken.act.Action;			
8	アノテーション	@WebServlet("/kenshop")			
【フィールド設計】					
項番	可視性	型	変数名	初期値	備考
1	private	long	serialVersionUID	1L	static final 修飾子付与
2	private	ServletConfig	config	null	
3	private	ServletContext	context	null	

処理名	フロントコントローラー	クラス名	Control
-----	-------------	------	---------

### メソッド・コンストラクタ詳細

【メソッド設計】

メソッド名	init
-------	------

項番	項目	値
1	アクセス修飾子	public
2	修飾子	—
3	戻り値	void
4	引数	—
5	try-catch(throws)	ServletException
6	アノテーション	@Override

【処理内容】

- フィールドの config に、このクラスの getServletConfig() メソッドの戻り値を代入する
- フィールドの context に、フィールドの config が参照しているオブジェクトの getServletContext() メソッドの戻り値を代入する

処理名	フロントコントローラー	クラス名	Control
メソッド・コンストラクタ詳細			
【メソッド設計】			
メソッド名		doGet	
項番	項目	値	
1	アクセス修飾子	public	
2	修飾子	－	
3	戻り値	void	
4	引数	HttpServletRequest 型 request : リクエストオブジェクト	
	引数	HttpServletResponse 型 response : レスポンスオブジェクト	
5	try-catch/throws	ServletException, IOException	
6	アノテーション	－	
【処理内容】			
1. このクラスの doPost() メソッドを呼び出す。その際に、引数で受け取ったリクエストオブジェクトとレスポンスオブジェクトを引数として渡す			

処理名	フロントコントローラー	クラス名	Control
メソッド・コンストラクタ詳細			
【メソッド設計】			
メソッド名		doPost	
項番	項目	値	
1	アクセス修飾子	public	
2	修飾子	—	
3	戻り値	void	
4	引数	HttpServletRequest 型 req : リクエストオブジェクト	
	引数	HttpServletResponse 型 res : レスポンスオブジェクト	
5	try-catch throws	ServletException , IOException	
6	アノテーション	—	
【処理内容】			
<div>1. ローカル変数として、String 型 actName、String 型 actionClassName、Action 型 action、String 型 forwardJSP、RequestDispatcher 型 rd のローカル変数を宣言する actName、actionClassName、forwardJSP は ""(空文字) で初期化、action、rd は null で初期化する</div> <div>2. 引数で受け取ったリクエストオブジェクトの setCharacterEncoding() メソッドを呼び出し、文字エンコーディングを行う。対応する文字コードは "UTF-8" とする</div> <div>3. フィールドの変数 context が参照しているオブジェクトの getRealPath() メソッドを呼び出す。その際、引数に "/" という文字列を渡し、コンテキストまでの物理パスを表す文字列を戻り値として受け取る</div> <div>4. 3. で受け取った文字列と、"/WEB-INF/action.properties" という文字列を結合する</div> <div>5. 4. で作成した文字列を引数に、FileInputStream クラスのオブジェクトを生成する</div> <div>6. Properties クラスのオブジェクトを生成する</div> <div>7. 6. で生成した Properties クラスオブジェクトの load() メソッド呼び出す。その際に、引数として 5. で生成した FileInputStream 型オブジェクトを渡す</div> <div>8. 引数で受け取っているリクエストオブジェクトから、"act" というキーのリクエストパラメータを取得し、1. で宣言した変数 actName に代入する</div> <div>9. もし、変数 actName が null ではない場合 6. で生成した Properties クラスオブジェクトの getProperty() メソッドを呼び出す。その際、引数として変数 actName を渡す。呼び出したら戻り値を変数 actionClassName に格納する</div> <div>10. もし、9. の結果が false の場合 6. で生成した Properties クラスオブジェクトの getProperty() メソッドを呼び出す。その際、引数には "top" という文字列を渡す。呼び出したら戻り値を変数 actionClassName に格納する</div> <div>11. Class クラスの static メソッドである forName() メソッドを呼び出す。その際に、引数として 1. で宣言した変数 actionClassName を渡し、戻り値を Class&lt;?&gt; 型変数 actionClass に格納する</div> <div>12. 変数 actionClass が参照しているオブジェクトの newInstance() メソッドを呼び出し、戻り値を Action 型の変数に代入する (キャストが必要)</div> <div>13. 変数 action が参照しているオブジェクトの execute() メソッドを呼び出す。その際、引数で受け取ったリクエストオブジェクトを引数として渡し、1. で宣言した変数 forwardJSP に代入する</div>			

処理名	フロントコントローラー	クラス名	Control
メソッド・コンストラクタ詳細			
<p>【処理内容】</p> <p>14. もし、変数 forwardJSP が null でなく、かつ変数 forwardJSP が参照している文字列が ""(空文字) ではない場合            変数 context の getRequestDispatcher() メソッドを呼び出す。その際、引数に変数 forwardJSP を渡し、戻り値を 1. で宣言した変数 rd に格納する。</p> <p>15. もし、14. の結果が false の場合            引数に文字列 "/top.jsp" を渡して変数 context の getRequestDispatcher() メソッドを呼び出し、戻り値を 1. で宣言した変数 rd に格納する</p> <p>16. 上記の処理 (3 ～ 15) を try 句で囲み、Exception 例外を受け取る。catch 句には以下の処理を記述する。            16.1 例外クラスの printStackTrace() メソッドを呼び出す            16.2 変数 context の getRequestDispatcher() メソッドを呼び出す。その際、引数として "/irregular_error.jsp" を渡し、戻り値を 1. で宣言した変数 rd に代入する</p> <p>17. 変数 rd の forward() メソッドを呼び出し、JSP に遷移させる</p>			