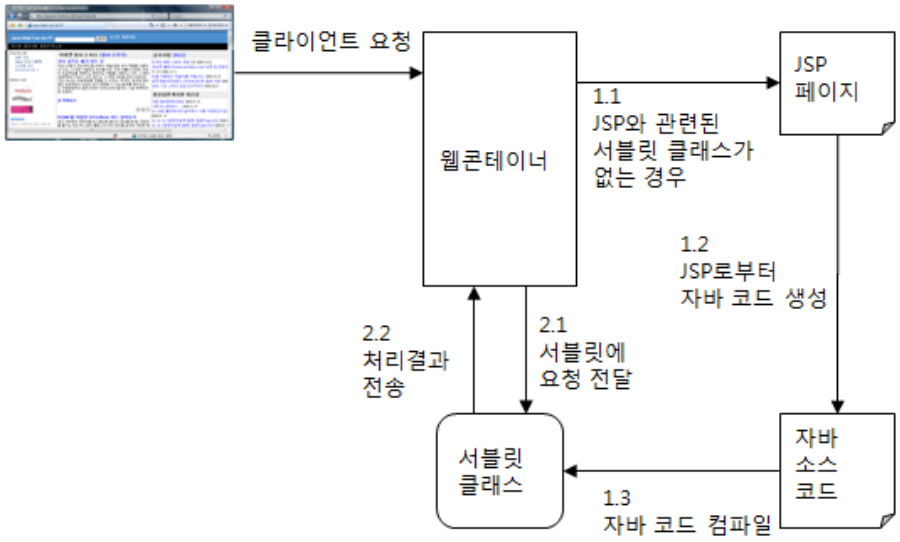
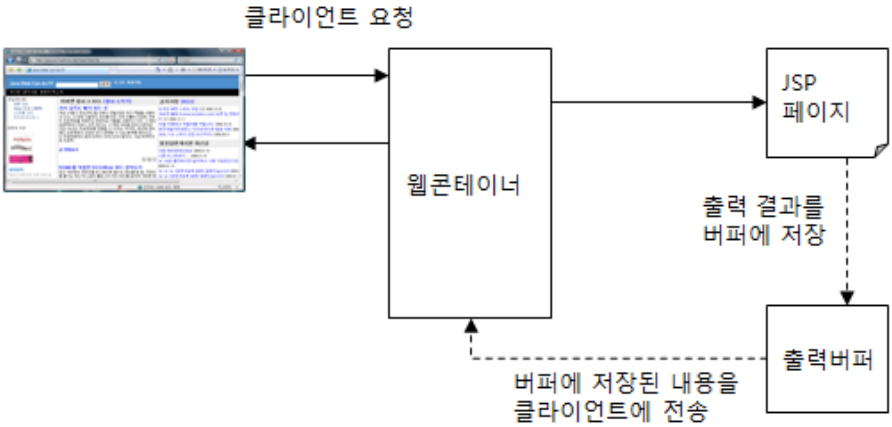
4장 필수 이해 요소

4.1 JSP 처리 과정



4.2 출력 버퍼와 응답

■ 출력 버퍼 - JSP가 생성한 응답 결과를 임시로 저장



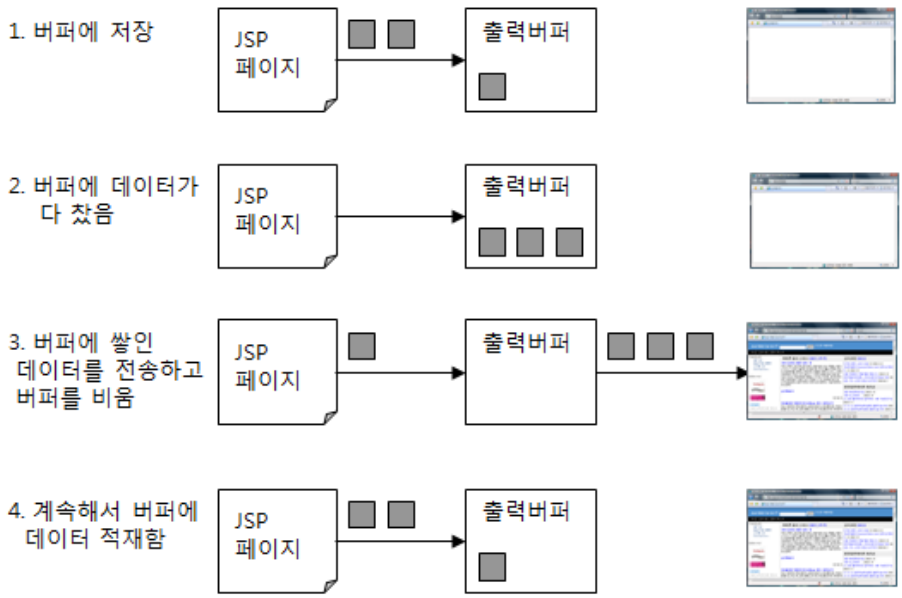
■ 출력 버퍼의 장점
- 데이터 전송 성능 향상
- 버퍼가 다 차기 전까지 헤더 변경 가능
- JSP 실행 도중 버퍼를 비우고 새 내용 전송 가능

4.2.1 page 디렉티브에서 버퍼 설정하기: buffer 속성과 autoFlush 속성

■ buffer 속성 : 버퍼 사용 여부 및 크기 지정
- <%@ page buffer="8kb" %> : 버퍼 크기를 8KB로 지정
- <%@ page buffer="none" %> : 버퍼 사용 안함

■ autoFlush 속성 : 버퍼가 다 찼을 때 처리 방식 지정
- true - 버퍼가 다 찼을 경우 버퍼를 플러시하고 계속해서 작업을 진행한다.
- false - 버퍼가 다 찼을 경우 예외를 발생시키고 작업을 중지한다.

■ 버퍼 처리 과정 : 기본적으로 버퍼가 다 차면 자동으로 데이터를 전송



[chap03\autoFlushTrue.jsp]

```
01   <%@ page contentType = "text/html; charset=utf-8" %>
02   <%@ page buffer="1kb" autoFlush="true" %>
03   <html>
04   <head><title>autoFlush 속성값 true 예제</title></head>
05   <body>
06
07   <%  for (int i = 0 ; i < 1000 ; i++) {   %>
08   1234
09   <%  }  %>
10
11   </body>
12   </html>
```

4.3 웹 어플리케이션 폴더 구성과 URL 매핑

■ 웹 어플리케이션 폴더 구조
- WEB-INF : 웹 어플리케이션 설정 정보를 담고 있는 web.xml 파일이 위치한다.
- WEB-INF\classes : 웹 어플리케이션에서 사용하는 클래스 파일이 위치한다.
- WEB-INF\lib : 웹 어플리케이션에서 사용하는 jar 파일이 위치한다.

4.4 웹 어플리케이션 배포

■ 웹 어플리케이션을 WAS에 배포하는 방법은 다음의 두 가지가 있다.
- 대상 디렉터리에 직접 복사

- war 파일로 묶어서 배포
  - 톰캣의 경우 [톰캣]\webapps에 war 파일 복사
  - war 파일의 이름이 보통 컨텍스트 경로가 됨

[꿀팁] 운영계 Apache Tomcat 설정 방법

1. Windows Service 등록

```
D:\prod\apache-tomcat-8.0.43\bin>service.bat install tomcat8_service
Installing the service 'tomcat8_service' ...
Using CATALINA_HOME:    "D:\prod\apache-tomcat-8.0.43"
Using CATALINA_BASE:    "D:\prod\apache-tomcat-8.0.43"
Using JAVA_HOME:        "C:\Program Files\Java\jdk1.8.0_91"
Using JRE_HOME:         "C:\Program Files\Java\jdk1.8.0_91\jre"
Using JVM:              "C:\Program Files\Java\jdk1.8.0_91\jre\bin\server\jvm.dl
l"
The service 'tomcat8_service' has been installed.

D:\prod\apache-tomcat-8.0.43\bin>
```

2. d:\prod\apache-tomcat-x.x.x\conf\server.xml 설정 변경

[chap03\autoFlushTrue.jsp]

```
01  <?xml version="1.0" encoding="UTF-8"?>
02  <!--
03    Licensed to the Apache Software Foundation (ASF) under one or more
04    contributor license agreements.  See the NOTICE file distributed with
05    this work for additional information regarding copyright ownership.
06    The ASF licenses this file to You under the Apache License, Version 2.0
07    (the "License"); you may not use this file except in compliance with
08    the License.  You may obtain a copy of the License at
09
10        http://www.apache.org/licenses/LICENSE-2.0
11
12    Unless required by applicable law or agreed to in writing, software
13    distributed under the License is distributed on an "AS IS" BASIS,
14    WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
15    See the License for the specific language governing permissions and
16    limitations under the License.
17  -->
18  <!-- Note:  A "Server" is not itself a "Container", so you may not
19       define subcomponents such as "Valves" at this level.
20       Documentation at /docs/config/server.html
21   -->
22  <Server port="8005" shutdown="SHUTDOWN">
23    <Listener className="org.apache.catalina.startup.VersionLoggerListener" />
24    <!-- Security listener. Documentation at /docs/config/listeners.html
25    <Listener className="org.apache.catalina.security.SecurityListener" />
26    -->
27    <!--APR library loader. Documentation at /docs/apr.html -->
28    <Listener className="org.apache.catalina.core.AprLifecycleListener" SSLEngine="on" />
29    <!-- Prevent memory leaks due to use of particular java/javax APIs-->
30    <Listener className="org.apache.catalina.core.JreMemoryLeakPreventionListener" />
31    <Listener className="org.apache.catalina.mbeans.GlobalResourcesLifecycleListener" />
32    <Listener className="org.apache.catalina.core.ThreadLocalLeakPreventionListener" />
33
34    <!-- Global JNDI resources
```

```
35          Documentation at /docs/jndi-resources-howto.html
36      -->
37      <GlobalNamingResources>
38        <!-- Editable user database that can also be used by
39             UserDatabaseRealm to authenticate users
40        -->
41        <Resource name="UserDatabase" auth="Container"
42                  type="org.apache.catalina.UserDatabase"
43                  description="User database that can be updated and saved"
44                  factory="org.apache.catalina.users.MemoryUserDatabaseFactory"
45                  pathname="conf/tomcat-users.xml" />
46      </GlobalNamingResources>
47
48      <!-- A "Service" is a collection of one or more "Connectors" that share
49           a single "Container" Note:  A "Service" is not itself a "Container",
50           so you may not define subcomponents such as "Valves" at this level.
51           Documentation at /docs/config/service.html
52       -->
53      <Service name="Catalina">
54
55        <!--The connectors can use a shared executor, you can define one or more named thread pools-->
56        <!--
57        <Executor name="tomcatThreadPool" namePrefix="catalina-exec-"
58            maxThreads="150" minSpareThreads="4"/>
59        -->
60
61
62        <!-- A "Connector" represents an endpoint by which requests are received
63             and responses are returned. Documentation at :
64             Java HTTP Connector: /docs/config/http.html
65             Java AJP  Connector: /docs/config/ajp.html
66             APR (HTTP/AJP) Connector: /docs/apr.html
67             Define a non-SSL/TLS HTTP/1.1 Connector on port 8080
68        -->
69        <Connector port="80" protocol="HTTP/1.1"
70                   connectionTimeout="20000"
71                   redirectPort="8443" />
72        <!-- A "Connector" using the shared thread pool-->
73        <!--
74        <Connector executor="tomcatThreadPool"
75                   port="8080" protocol="HTTP/1.1"
76                   connectionTimeout="20000"
77                   redirectPort="8443" />
78        -->
79        <!-- Define a SSL/TLS HTTP/1.1 Connector on port 8443
80             This connector uses the NIO implementation. The default
81             SSLImplementation will depend on the presence of the APR/native
82             library and the useOpenSSL attribute of the
83             AprLifecycleListener.
84             Either JSSE or OpenSSL style configuration may be used regardless of
85             the SSLImplementation selected. JSSE style configuration is used below.
86        -->
87        <!--
88        <Connector port="8443" protocol="org.apache.coyote.http11.Http11NioProtocol"
89                   maxThreads="150" SSLEnabled="true">
90            <SSLHostConfig>
91                <Certificate certificateKeystoreFile="conf/localhost-rsa.jks"
92                             type="RSA" />
93            </SSLHostConfig>
94        </Connector>
95        -->
96        <!-- Define a SSL/TLS HTTP/1.1 Connector on port 8443 with HTTP/2
97             This connector uses the APR/native implementation which always uses
98             OpenSSL for TLS.
99             Either JSSE or OpenSSL style configuration may be used. OpenSSL style
```

```
100           configuration is used below.
101       -->
102       <!--
103       <Connector port="8443" protocol="org.apache.coyote.http11.Http11AprProtocol"
104                  maxThreads="150" SSLEnabled="true" >
105           <UpgradeProtocol className="org.apache.coyote.http2.Http2Protocol" />
106           <SSLHostConfig>
107               <Certificate certificateKeyFile="conf/localhost-rsa-key.pem"
108                            certificateFile="conf/localhost-rsa-cert.pem"
109                            certificateChainFile="conf/localhost-rsa-chain.pem"
110                            type="RSA" />
111           </SSLHostConfig>
112       </Connector>
113       -->
114
115       <!-- Define an AJP 1.3 Connector on port 8009 -->
116       <Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />
117
118
119       <!-- An Engine represents the entry point (within Catalina) that processes
120            every request.  The Engine implementation for Tomcat stand alone
121            analyzes the HTTP headers included with the request, and passes them
122            on to the appropriate Host (virtual host).
123            Documentation at /docs/config/engine.html -->
124
125       <!-- You should set jvmRoute to support load-balancing via AJP ie :
126       <Engine name="Catalina" defaultHost="localhost" jvmRoute="jvm1">
127       -->
128       <Engine name="Catalina" defaultHost="localhost">
129
130         <!--For clustering, please take a look at documentation at:
131             /docs/cluster-howto.html  (simple how to)
132             /docs/config/cluster.html (reference documentation) -->
133         <!--
134         <Cluster className="org.apache.catalina.ha.tcp.SimpleTcpCluster"/>
135         -->
136
137         <!-- Use the LockOutRealm to prevent attempts to guess user passwords
138              via a brute-force attack -->
139         <Realm className="org.apache.catalina.realm.LockOutRealm">
140           <!-- This Realm uses the UserDatabase configured in the global JNDI
141                resources under the key "UserDatabase".  Any edits
142                that are performed against this UserDatabase are immediately
143                available for use by the Realm.  -->
144           <Realm className="org.apache.catalina.realm.UserDatabaseRealm"
145                  resourceName="UserDatabase"/>
146         </Realm>
147
148         <Host name="localhost"  appBase="webapps"
149               unpackWARs="true" autoDeploy="true">
150
151             <Context path="" docBase="AdminLTE-2.4.0-rc" />
152             <Context path="ROOT" docBase="ROOT" />
153
154             <!-- <Context path="" docBase="${catalina.home}/AdminLTE-2.4.0-rc" reloadable="false" >
155       </Context> -->
156
157
158           <!-- SingleSignOn valve, share authentication between web applications
159                Documentation at: /docs/config/valve.html -->
160           <!--
161           <Valve className="org.apache.catalina.authenticator.SingleSignOn" />
162           -->
163
164           <!-- Access log processes all example.
```

```
165               Documentation at: /docs/config/valve.html
166               Note: The pattern used is equivalent to using pattern="common" -->
167          <Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs"
168                 prefix="localhost_access_log" suffix=".txt"
169                 pattern="%h %l %u %t &amp;quot;%r&amp;quot; %s %b" />
170
171        </Host>
172      </Engine>
173    </Service>
174  </Server>
```

3. war 파일 배치

"D:\prod\apache-tomcat-8.0.43\webapps\" 디렉토리에 war 파일을 배치한다.