

Real Alternative DBMS ALTIBASE, Since 1999
ALTIBASE & WebLogic 연동 가이드

2010. 12



Copyright © 2000~2013 ALTIBASE Corporation. All Rights Reserved.

Document Control

Change Record

Date	Author	Change Reference
2010-12-13	durusari	Created

Reviews

Date	Name (Position)
2010-12-31	ma114(TS), balance(TS), snkim(SP)

Distribution

Name	Location

목차

개요	4
WEBLOGIC 용어 설명	5
도메인	5
WebLogic 서버 인스턴스 (관리 서버, 관리된 서버)	5
JDBC 데이터 소스	6
커넥션 풀(pool)	6
배치, 배포	6
시작하기 전에	7
WebLogic 버전	7
JDBC 스펙(spec)	7
WEBLOGIC SETUP	8
다운로드	8
설치	8
도메인 생성	10
WEBLOGIC 서버 인스턴스의 구동과 종료	12
구동	12
종료	12
JDBC 드라이버 설정	13
ALTIBASE JDBC 드라이버를 얻는 방법	13
ALTIBASE JDBC 드라이버 버전을 확인하는 방법	13
WebLogic에서 사용할 JDBC 드라이버를 설정하는 방법	13
ALTIBASE와 WEBLOGIC 연동	15
JDBC 데이터 소스 생성	15
웹 응용프로그램 생성 및 배치	20
연동 예제	21
FAIL-OVER & LOAD-BALANCING 설정	25
WebLogic 에서의 설정	25
ALTIBASE 에서의 설정	25
연동 오류	28
Cannot load driver	28
Can not make a database connection to the given URL / Invalid Altibase URL (No suitable driver)	28
Could not create pool connection	28
Client unable to establish connection / Communication link failure	28
Hang 현상	29

개요

본 문서는 ALTIBASE와 WebLogic을 연동하기 위한 가이드 문서로서 WebLogic 설치부터 시작하여 ALTIBASE와 연동을 하는 간략한 웹 응용프로그램 실행까지 다루고 있습니다.

본 문서는 Windows 플랫폼의 WebLogic 10.3.3.0 을 기준으로 작성되었으며 모든 내용은 버전에 따라 상이할 수 있으므로 WebLogic 설치 및 도메인 생성과 같은 기본환경 구축은 BEA Systems 또는 Oracle의 문서, WebLogic 설치 시 제공되는 관리 콘솔(Administration Console) 도움말을 참조하시기 바랍니다.

- [Oracle Documentation Library](#)

본 문서와 관련된 오류 및 개선사항은 ALTIBASE 기술본부 대표 메일인 support@altibase.com으로 보내주시기 바랍니다.

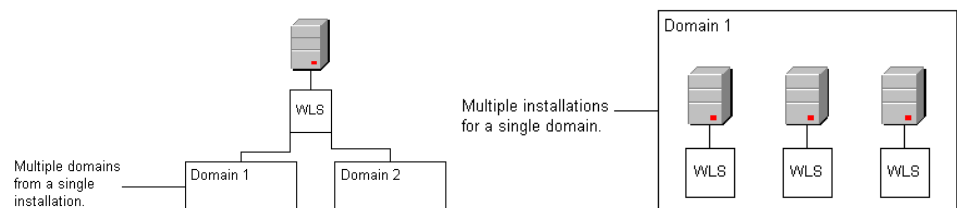
WebLogic 용어 설명

ALTIBASE와 WebLogic의 연동에 앞서 기본적으로 알아야 할 WebLogic과 관련 용어에 대해 간략하게 설명한다.

도메인

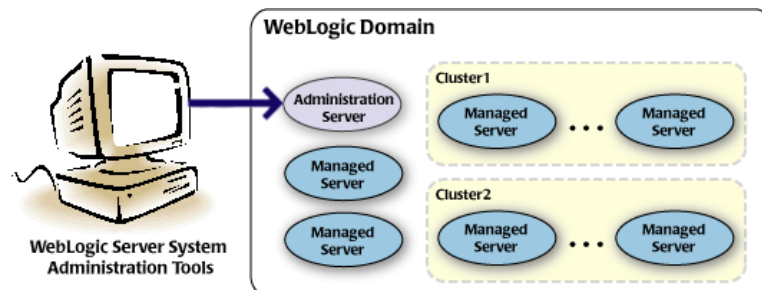
도메인이란 WebLogic 서버 인스턴스(instance)에 대한 논리적인 그룹으로 하나의 도메인은 하나 이상의 WebLogic 서버 인스턴스로 구성된다.

사용자는 하나의 WebLogic (WebLogic Server/WLS)을 설치하여 여러 개의 도메인을 구성하거나 여러 개의 WebLogic을 설치하여 하나의 도메인으로 구성할 수 있다.



WebLogic 서버 인스턴스 (관리 서버, 관리된 서버)

WebLogic 서버 인스턴스란 웹 응용프로그램(Web Application), 웹 서비스(Web Services), EJB 등을 배치(deploy)할 수 있는 하나의 프로세스(process)로서 “관리 서버”, “관리된 서버”로 구분한다.



- 관리 서버 (Administration Server)

도메인 전체를 관리할 수 있도록 관리 콘솔(Administration Console)이 함께 배치된 WebLogic 서버 인스턴스로 도메인 생성시 함께 생성된다.

- 관리된 서버 (Managed Server)

관리 서버를 제외한 모든 WebLogic 서버 인스턴스로 별도로 추가하거나 인스턴스간 클러스터링(clustering)이 가능하다.

즉, 하나의 도메인은 1 개의 관리 서버와 0 개 이상의 관리된 서버로 구성된다.

JDBC 데이터 소스

JDBC 데이터 소스란 WebLogic 서버 인스턴스가 RDBMS에 접속하기 위한 자바 객체이다.

연동할 RDBMS에 대한 JDBC 데이터 소스를 생성 후 해당 JDBC 데이터 소스를 사용할 WebLogic 서버 인스턴스를 지정하면 해당 WebLogic 서버 인스턴스에 배치된 웹 응용프로그램은 소스코드에 JDBC 데이터 소스의 JNDI (Java Naming and Directory Interface) 이름을 명시하여 대상 RDBMS와 연동을 할 수 있다.

JDBC 데이터 소스를 생성하기 위해서는 관련 JDBC 드라이버가 WebLogic에서 사용할 수 있도록 사전에 설정되어 있어야만 한다.

커넥션 풀(pool)

JDBC 데이터 소스 생성 시 자동으로 생성되는 접속 대상 RDBMS와의 접속 집합으로 “초기 용량”, “최대 용량”과 같은 다양한 속성에 의해 조절된다.

커넥션 풀, 접속 풀은 동의어로 본 문서에서는 “커넥션 풀”로 통일한다.

배치, 배포

배치, 배포는 모두 deploy 를 의미하는 동의어로 본 문서에서는 “배치”로 통일한다.

시작하기 전에

WebLogic 버전

기본적으로 최신 버전을 권장하며 최소 WebLogic 8.1.6.0 이상을 권장한다.

이보다 하위버전에서 서로 다른 버전의 ALTIBASE 를 동시에 연동하면 JDBC 드라이버 연동이 정상적으로 되지 않아 WebLogic 서버 인스턴스의 구동이 실패하거나 행(hang)과 같은 현상이 발생할 수 있다.

JDBC 스펙(spec)

ALTIBASE 5 JDBC 드라이버는 JDBC 스펙 3.0 (JRE 1.4 또는 1.5)을 기준으로 JDK 1.4 에서 제작(build)되었으며 스펙 미 지원의 경우는 “Not Supported” 또는 에러를 반환한다.

따라서, JDBC 스펙 4.0(JRE 1.6 이상)을 기준으로 작성된 응용프로그램은 스펙 미 지원으로 인한 문제가 발생할 수 있으므로 개발단계에서부터 JDBC 스펙 3.0 을 준수하도록 해야 한다.

또한, JDBC 스펙 3.0 이라 할지라도 일부 지원하지 않는 스펙이 존재하므로 사전에 “API 매뉴얼”을 참조하여 미 지원 스펙을 확인하도록 한다.

스펙 미 지원으로 인하여 발생하는 문제에 대해 ALTIBASE는 보장하지 않는다.

WebLogic Setup

WebLogic을 다운로드 및 설치 후 하나의 도메인을 생성하는 간략한 절차를 Windows 플랫폼의 WebLogic 10.3.3.0 을 예로 하여 설명한다.

다운로드

2008 년 4 월 WebLogic을 개발한 BEA Systems 가 Oracle로 인수되면서 버전 9 이상부터는 Oracle 미들웨어 제품 다운로드 페이지에서 다운로드 할 수 있다.

<http://www.oracle.com/technetwork/middleware/downloads/index.html>

해당 웹 페이지에서 라이선스 동의 후 운영체제 플랫폼에 맞는 가장 최근에 릴리즈(release)된 WebLogic을 다운로드 받도록 한다.

Oracle Fusion Middleware 11g Software Downloads

You must accept the [OTN License Agreement](#) to download this software.

☒ Accept License Agreement | ☐ Decline License Agreement

For Development:

Oracle WebLogic Server 11g Rel 1 (10.3.3) Installers

The WebLogic Server installers include Oracle Coherence and Oracle Enterprise Pack for Eclipse and supports development with other Fusion Middleware products . The zip includes WebLogic Server only and is intended for WebLogic Server development only.

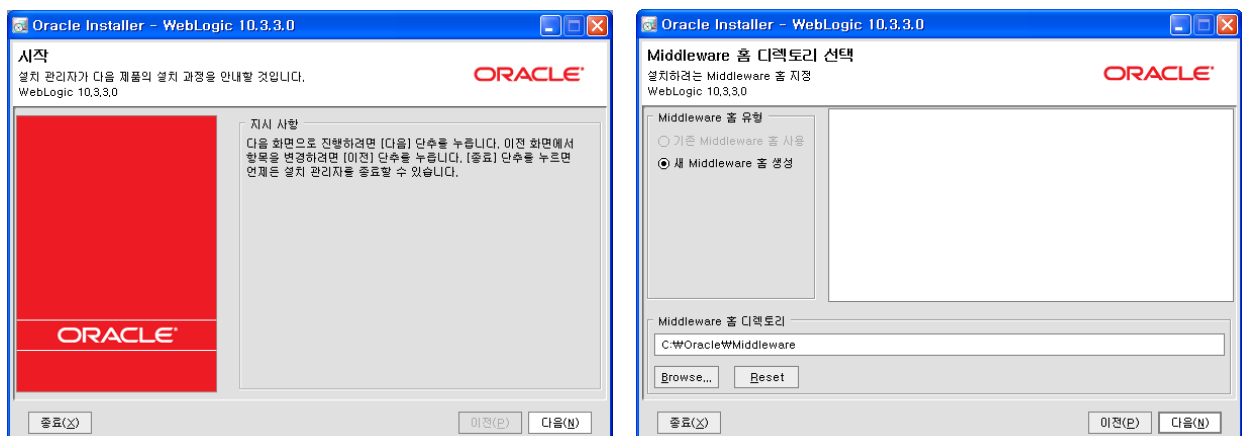
 [Linux x86 \(923 MB\)](#) |  [Windows x86 \(997 MB\)](#)

 [Zip for Windows x86, Linux x86, Mac OS X \(413 MB\)](#) | [See All](#)

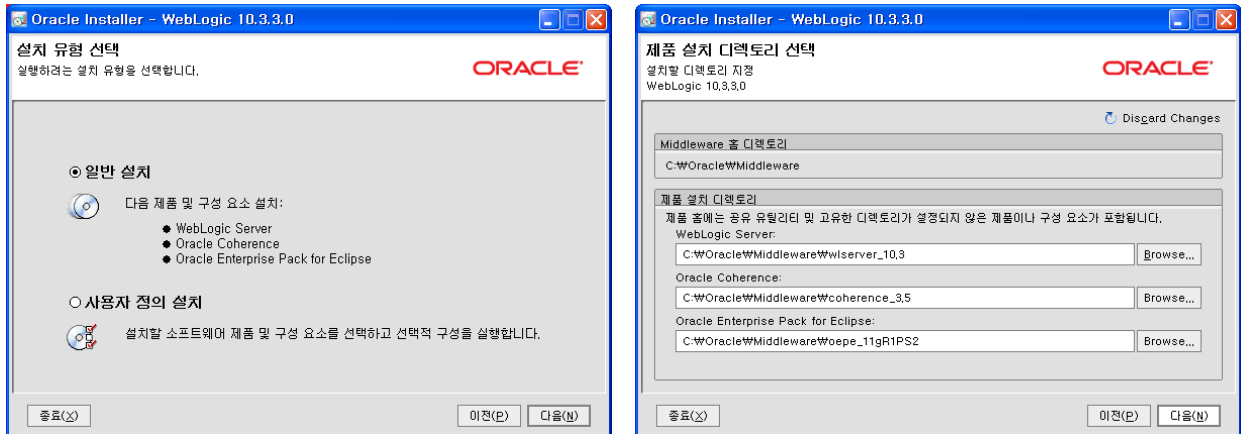
설치

다운로드 받은 파일을 실행하면 WebLogic 자동 인스톨러(installer)가 수행된다. 아래와 같은 절차로 WebLogic을 설치한다.

1. 시작 화면에서 다음을 선택 후 WebLogic 및 관련 미들웨어 제품이 설치될 경로를 지정한다.



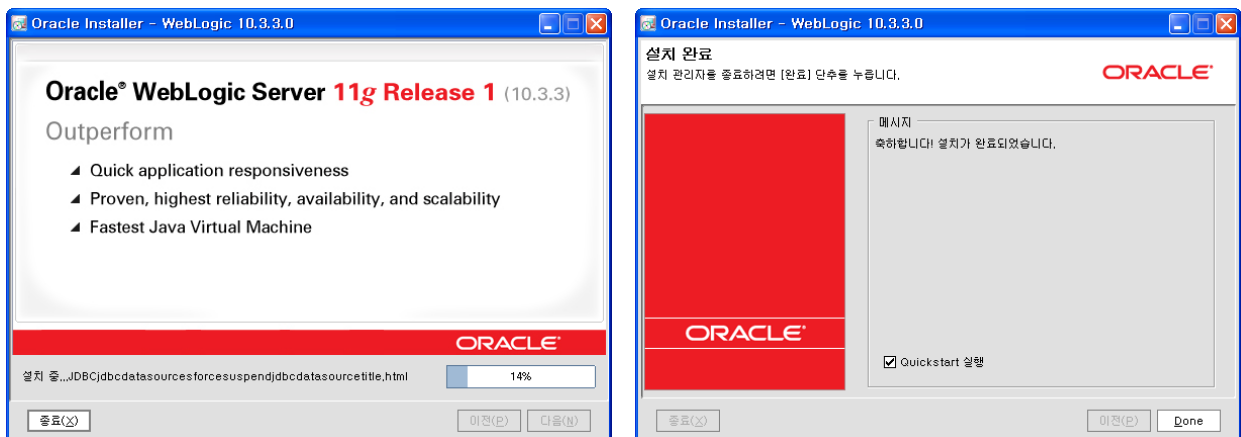
2. 설치 유형을 선택 후 WebLogic 및 관련 미들웨어 제품이 설치될 경로를 지정한다.
(여기서 지정한 WebLogic 의 설치경로는 추후 환경변수 WL_HOME 으로 식별된다.)



3. 시작 메뉴 폴더 위치를 지정 후 설치를 시작한다.



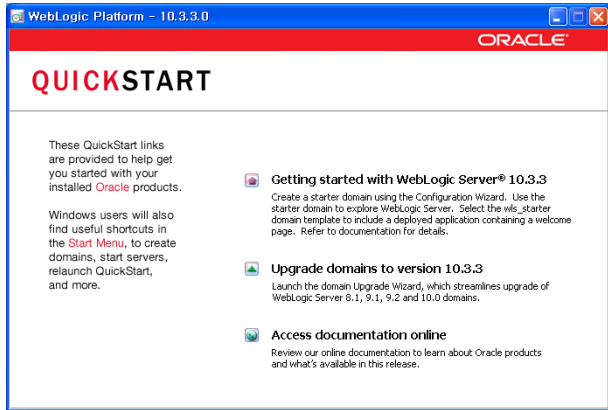
4. 설치가 모두 완료되면 이어서 수행 할 도메인 생성을 위해 “Quickstart 실행” 옵션이 체크된 상태에서 완료(Done)를 선택한다.



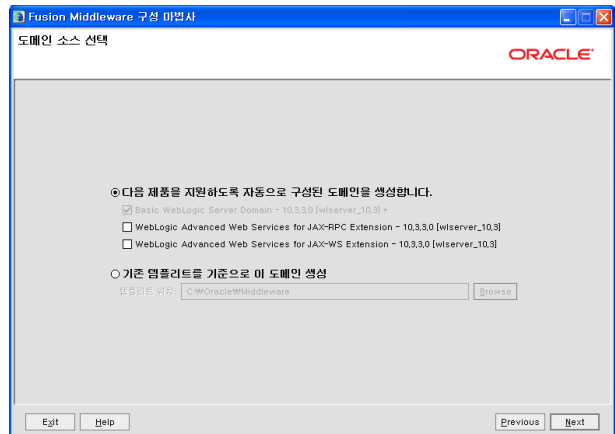
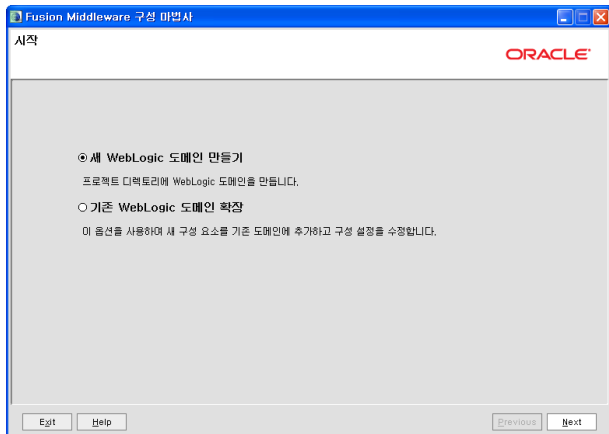
도메인 생성

설치 마법사(Configuration Wizard)를 사용하여 도메인을 생성할 수 있다. 설치 마법사 실행은 해당 파일(\$WL_HOME/common/bin/config.exe)을 직접 실행하거나 시작 메뉴의 WebLogic 메뉴 그룹에서 바로 가기를 실행하면 된다. 아래와 같은 절차로 WebLogic을 설치한다.

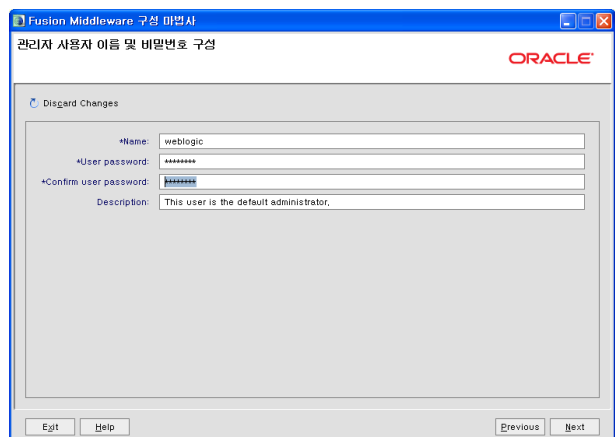
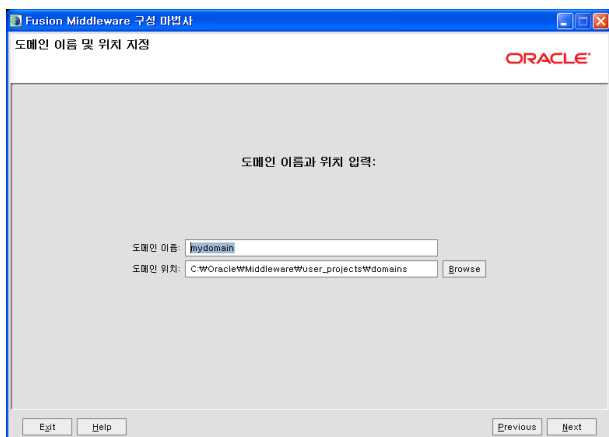
1. 설치 마법사를 실행한다. 이전에 “Quickstart”를 수행했다면 “Getting started with WebLogic Sever”를 선택한다.



2. 새 WebLogic 도메인 만들기를 선택 후 필요한 경우 추가적인 도메인 소스를 선택한다.



3. 새로이 생성할 도메인의 이름과 위치(설치경로)를 지정 후 관리 서버의 사용자 이름 및 비밀번호를 입력한다. (여기서 지정한 도메인의 설치경로는 추후 환경변수 DOMAIN_HOME 으로 식별된다.)



4. 도메인 관리 서버의 시작 모드와 JDK 설정을 한다.

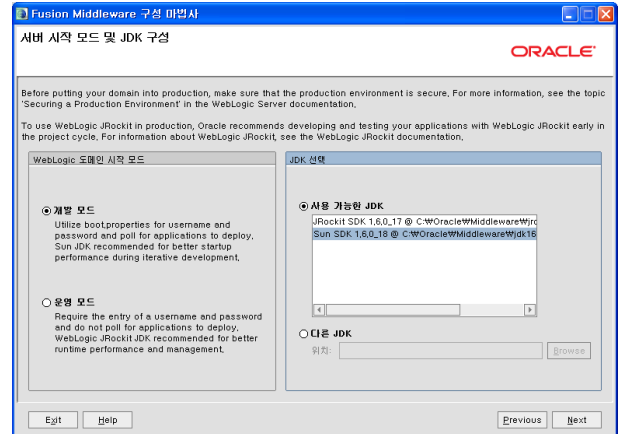
✓ 시작 모드

테스트가 목적이므로 “개발 모드”로 선택한다.

개발 모드는 \$DOMAIN_HOME/autodeploy 를 통하여 웹 응용프로그램을 자동으로 배치 및 변경할 수 있어 손쉬운 테스트가 가능하다.

✓ JDK

시작 모드를 선택하면 그에 적절한 JDK가 자동적으로 선택되므로 특별한 경우가 아니라면 별도로 선택하지 않아도 된다.

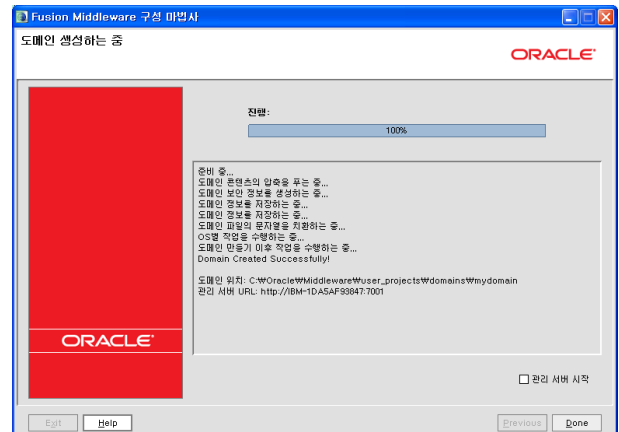


5. 관리 서버의 수신 포트와 같은 특별히 변경을 원하는 사항이 있다면 관련 항목을 선택 후 변경하도록 한다. 없다면 생략(NEXT)한다.

참고로 관리 서버는 기본적으로 7001 포트를 사용한다.



6. 도메인 생성이 모두 완료되면 “관리 서버 시작” 옵션을 체크한 상태에서 완료(Done)를 선택한다.



WebLogic 서버 인스턴스의 구동과 종료

WebLogic 서버 인스턴스를 구동하고 종료하는 여러 가지 방법 중 \$DOMAIN_HOME/bin 에 존재하는 스크립트(script or batch) 파일 실행을 통한 구동과 종료에 대해 설명한다.

스크립트 파일은 도메인 별로 관리 서버와 관리된 서버 각각에 대한 구동/종료 스크립트 파일이 별도로 존재하는데 여기서는 관리 서버의 구동과 종료에 대해서만 설명한다.

- 관리 서버 - startWebLogic.cmd / stopWebLogic.cmd
- 관리된 서버 - startManagedWebLogic.cmd / stopManagedWebLogic.cmd

구동

startWebLogic.cmd 를 실행하여 해당 도메인의 관리 서버를 구동한다. 아래는 정상적으로 구동이 완료되었을 때의 메시지 형태 예제이다.

```
$ startWebLogic.cmd
.
.
JAVA Memory arguments: -Xms256m -Xmx512m -XX:CompileThreshold=8000 -
XX:PermSize=48m -XX:MaxPermSize=128m
.
WLS Start Mode=Development
... 중략 ...

<2010. 12. 7 오후 7 시 14 분 27 초 KST> <Notice> <WebLogicServer> <BEA-000365>
<Server state changed to RUNNING>
<2010. 12. 7 오후 7 시 14 분 27 초 KST> <Notice> <WebLogicServer> <BEA-000360>
<Server started in RUNNING mode>
```

종료

stopWebLogic.cmd 를 실행하여 해당 도메인의 관리 서버를 종료한다. 아래는 정상적으로 종료가 완료되었을 때의 메시지 형태 예제이다.

```
$ stopWebLogic.cmd
Stopping weblogic Server...

Initializing WebLogic Scripting Tool (WLST) ...
Welcome to WebLogic Server Administration Scripting Shell

Type help() for help on available commands

Connecting to t3://IBM-1DA5AF93847:7001 with userid weblogic ...
Successfully connected to Admin Server 'AdminServer' that belongs to domain
'mydomain'.
... 중략 ...

Shutting down the server AdminServer with force=false while connected to
AdminServer ...
Disconnected from weblogic server: AdminServer

Exiting WebLogic Scripting Tool.

Done
Stopping Derby Server...
```

종료의 경우, 종료 스크립트 파일 실행 결과만으로 정상 종료를 보장할 수 없다. 메시지상으로 정상적인 종료를 수행하였음에도 불구하고 관련 프로세스가 여전히 존재할 시는 WebLogic 서버 인스턴스의 로그를 확인하여 조치하여야 한다.

JDBC 드라이버 설정

JDBC 데이터 소스 생성을 하기 위해서는 관련 JDBC 드라이버를 WebLogic에 설정하여야 한다. JDBC 드라이버 설정에 대한 전반적인 사항을 설명한다.

ALTIBASE JDBC 드라이버를 얻는 방법

ALTIBASE JDBC 드라이버는 Altibase.jar 라는 하나의 파일로 제공된다.

기본적으로 ALTIBASE 서버 패키지와 함께 제공되므로 ALTIBASE가 설치된 운영체제 계정의 \$ALTIBASE_HOME/lib 디렉토리 안에서 얻을 수 있다.

ALTIBASE 5 버전부터는 Altibase.jar는 물론 Altibase5.jar 라는 파일도 존재하는데, Altibase5.jar는 ALTIBASE 5 버전과 그 이하의 버전을 함께 연동할 때 사용한다.

따라서, 하나의 ALTIBASE 서버와만 연동하거나 동일한 버전의 여러 ALTIBASE 서버를 연동하는 일반적인 상황에서는 \$ALTIBASE_HOME/lib/Altibase.jar 파일을 사용하면 된다.

최신 ALTIBASE JDBC 드라이버는 ATC(<http://atc.altibase.com>) 를 통하여 얻을 수 있다.

ALTIBASE JDBC 드라이버 버전을 확인하는 방법

연동하려는 ALTIBASE 서버와 ALTIBASE JDBC 드라이버가 호환 가능한지 판단을 위해 ALTIBASE 서버와 ALTIBASE JDBC 드라이버의 버전을 비교하여 CM(Communication) 프로토콜이 일치하는지 반드시 확인하여야 한다.

ALTIBASE 서버의 버전을 확인하는 방법은 아래와 같다.

```
$ altibase -v
version 5.3.1.7 IBM_AIX_5.3-64bit-5.3.1.7-release-xtC_r (powerpc-ibm-
aix5.3.0.0) Jan 3 2010 08:38:40, binary db version 5.3.1, meta version
5.5.1, cm protocol version 5.5.1, replication protocol version 5.3.1
```

ALTIBASE JDBC 드라이버의 버전을 확인하는 방법은 아래와 같다.

```
$ java -jar Altibase.jar
JDBC Driver Info : Altibase Ver = 5.3.1.7 for JavaVM v1.4, CMP:5.5.1,
$Revision: 14502 $ Jan 3 2010 08:38:29
```

여기서 ALTIBASE 서버의 “cm protocol version”과 ALTIBASE JDBC 드라이버의 “CMP”가 일치하면 호환이 가능하다.

일반적으로 호환성을 위해 ALTIBASE 서버 패키지에 포함된 ALTIBASE JDBC 드라이버 파일을 사용하나 가급적이면 호환 가능한 최신의 ALTIBASE JDBC 드라이버 파일을 사용하는 것을 권장한다.

WebLogic에서 사용할 JDBC 드라이버를 설정하는 방법

WebLogic에서 특정 JDBC 드라이버를 사용하려면 관련 파일을 해당 도메인 관리 서버의 CLASSPATH에 추가해야 한다.

크게 두 가지 방법으로 설정이 가능하며 둘 중 어느 방법을 선택하여도 무관하나 설정 후에는 반드시 해당 도메인의 관리 서버를 재구동 해야 한다.

참고로 모든 예제들은 ALTIBASE 5 와 버전이 다른 ALTIBASE 를 동시에 연동하기 위한 설정으로 동일 버전의 ALTIBASE간의 연동 시에는 하나의 JDBC 드라이버만 설정하면 된다.

1. \$DOMAIN_HOME/lib에 JDBC 드라이버 관련 파일을 위치.

각 도메인의 기본 라이브러리 경로인 \$DOMAIN_HOME/lib에 JDBC 드라이버 관련 파일을 위치 후 해당 도메인의 관리 서버 구동하면 아래와 같은 메시지와 함께 해당 파일이 자동으로 CLASSPATH에 추가된다.

```
...
<2010. 12. 14 오후 4 시 48 분 24 초 KST> <Notice> <WebLogicServer>
<BEA-000395> <Following extensions directory contents added to the
end_of_the_classpath:C:\Oracle\Middleware\user_projects\domains\myd
omain\lib\Altibase.jar;C:\Oracle\Middleware\user_projects\domains\m
ydomain\lib\Altibase5.jar>
...
```

2. startWebLogic.cmd 파일의 CLASSPATH에 JDBC 드라이버 관련 파일을 직접추가.

각 도메인의 관리 서버 구동 스크립트 파일인 startWebLogic.cmd 를 open하여 CLASSPATH 항목에 JDBC 드라이버 관련 파일을 직접 추가 후 재구동 한다.

```
...
set CLASSPATH=%SAVE_CLASSPATH%;C:\webLogic_lib\Altibase5.jar;C:\web
Logic_lib\Altibase.jar
...
```

ALTIBASE와 WebLogic 연동

도메인 관리 콘솔을 사용하여 연동할 ALTIBASE에 대한 JDBC 데이터 소스를 생성 후 해당 JDBC 데이터 소스를 사용하는 간단한 웹 응용프로그램을 실행하는 전체적인 연동 절차를 설명한다.

JDBC 데이터 소스 생성

ALTIBASE에 대한 JDBC 데이터 소스를 생성하고 해당 JDBC 데이터 소스의 커넥션 풀을 설정하는 절차이다.

참고로 하위 버전에서는 커넥션 풀을 먼저 생성 후 해당 커넥션 풀을 사용할 JDBC 데이터 소스를 생성하는 방식이었으나 상위 버전에서는 JDBC 데이터 소스를 생성하면 자동으로 커넥션 풀이 생성된다.

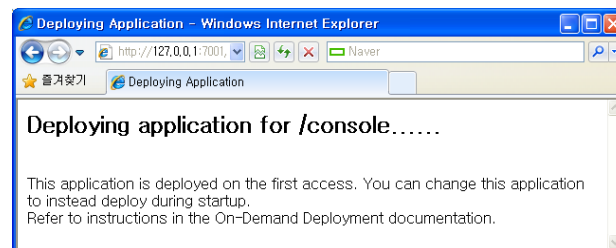
기본적으로 [WebLogic에서 사용할 JDBC 드라이버를 설정하는 방법](#) 을 참조하여 생성할 JDBC 데이터 소스에 대한 JDBC 드라이버 설정이 선행되어 있어야 한다.

JDBC 데이터 소스 생성

로컬에 설치된 ALTIBASE 5 에 대한 JDBC 데이터 소스 “altibase”를 생성한다.

1. 도메인의 관리 서버에서 제공하는 관리 콘솔로 접속한다.

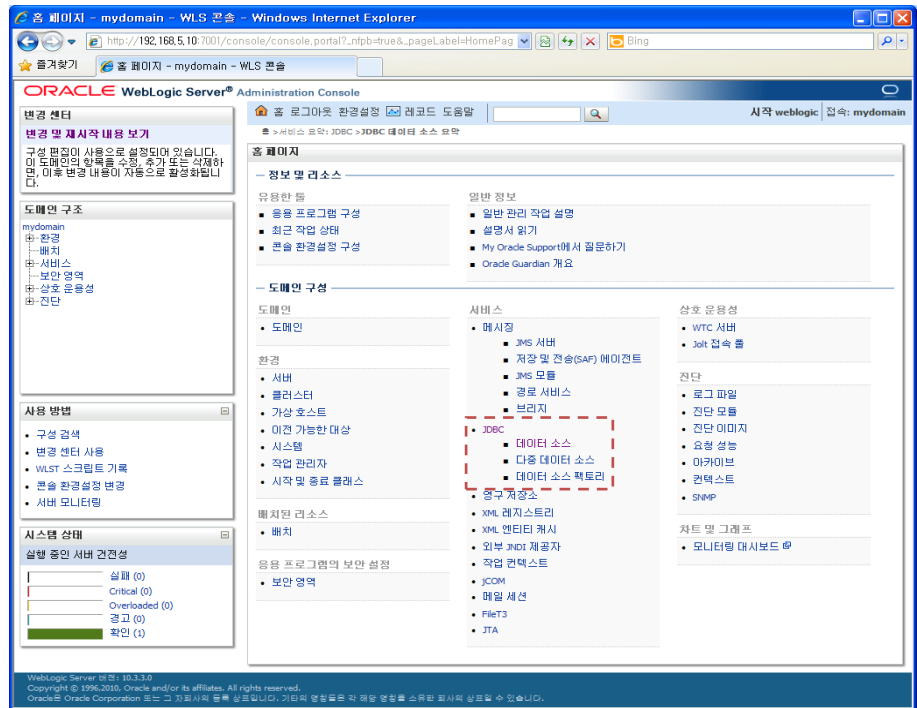
<http://127.0.0.1:7001/console>



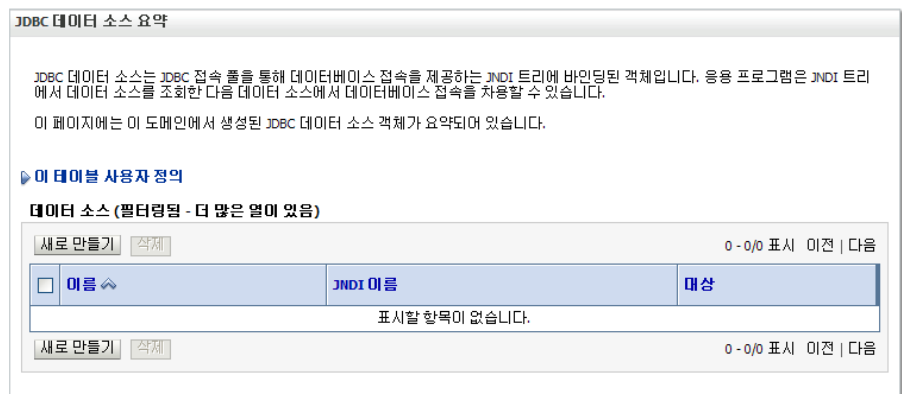
2. 관리 서버의 사용자 이름과 비밀번호와 입력하여 로그인 한다.



3. 관리 콘솔 메인 페이지에서 “데이터 소스” 항목을 선택한다.



4. [새로 만들기]를 선택한다.



5. JDBC 데이터 소스 속성을 입력 후 다음 단계로 간다.

- ✓ JDBC 데이터 소스, JNDI 이름 일반적으로 동일하게 한다.
- ✓ 데이터베이스 유형 [기타]로 선택한다.



6. 이어지는 “JDBC 데이터 소스 속성” 단계는 생략하고 다음 단계로 간다.
7. “트랜잭션 옵션”을 설정한다. 특별히 설정할 것이 없다면 다음 단계로 간다.

새 JDBC 데이터 소스 생성

뒤로 다음 완료 취소

트랜잭션 옵션

새 데이터 소스에 데이터베이스 접속을 생성할 때 사용할 비XA JDBC 드라이버를 선택했습니다.

이 데이터 소스에서 글로벌 트랜잭션을 지원합니까? 지원하는 경우 이 데이터 소스에 대한 트랜잭션 프로토콜을 선택하십시오.

☒ 글로벌 트랜잭션 지원

데이터 소스로부터의 비XA JDBC 접속이 *마지막 리소스 로깅(LR)* 트랜잭션 최적화를 사용하여 글로벌 트랜잭션에 참여할 수 있게 하려면 이 옵션을 선택합니다. 2단계 커밋 애플레이트 대신 이 옵션을 사용하는 것이 좋습니다.

☐ 마지막 리소스 로깅

데이터 소스로부터의 비XA JDBC 접속이 JTA를 사용하여 글로벌 트랜잭션 참여를 애플레이트할 수 있게 하려면 이 옵션을 선택합니다. 응용 프로그램에 추론 과정에 대한 대결합성이 있는 경우에만 이 옵션을 선택하십시오.

☐ 2단계 커밋 애플레이트

데이터 소스로부터의 비XA JDBC 접속이 1단계 커밋 트랜잭션 처리를 사용하여 글로벌 트랜잭션에 참여할 수 있게 하려면 이 옵션을 선택합니다. 이 옵션을 사용하면 다른 리소스는 글로벌 트랜잭션에 참여할 수 없습니다.

☐ 1단계 커밋

뒤로 다음 완료 취소

8. 연동할 ALTIBASE의 정보를 입력한다.

새 JDBC 데이터 소스 생성

뒤로 다음 완료 취소

접속 속성

접속 속성을 정의합니다.

접속할 데이터베이스 이름을 입력하십시오.

데이터베이스 이름: mydb

데이터베이스 서버의 이름 또는 IP 주소를 지정하십시오.

호스트 이름: 127.0.0.1

데이터베이스 접속에 사용되는 데이터베이스 서버의 포트를 지정하십시오.

포트: 20300

데이터베이스 접속을 생성할 때 사용할 데이터베이스 계정 사용자 이름을 지정하십시오.

데이터베이스 사용자 이름: sys

데이터베이스 접속을 생성할 때 사용할 데이터베이스 계정 비밀번호를 지정하십시오.

비밀번호:

비밀번호 확인:

뒤로 다음 완료 취소

9. ALTIBASE 접속을 위해 커넥션 풀과 관련된 JDBC 정보를 입력한다.
 - ✓ 드라이버 클래스 이름
Altibase.jar 를 사용하는 경우, Altibase.jdbc.driver.AltibaseDriver
Altibase#.jar 를 사용하는 경우, Altibase#.jdbc.driver.AltibaseDriver
 - ✓ URL
이전 단계에 입력한 ALTIBASE의 정보를 토대로 접속 URL을 입력한다.
jdbc:Altibase://127.0.0.1:20300/mydb

- ✓ 속성
ALTIBASE 4 의 경우, “**user=sys**” 형식으로 접속할 사용자를 입력해야 한다.
ALTIBASE 5 의 경우는 생략해도 무방하다.
- ✓ 테스트 테이블
일반적으로 더미(dummy) 테이블인 **dual** 을 지정한다.

새 JDBC 데이터 소스 생성

구성 테스트

뒤로

다음

완료

취소

데이터베이스 접속 테스트

제공한 데이터베이스 가용성과 접속 속성을 테스트합니다.

접속 풀에서 데이터베이스 접속을 생성하는 데에 사용되는 JDBC 드라이버 클래스의 전체 패키지 이름은 무엇입니까?
(배치할 모든 서버의 클래스 경로에 이 드라이버 클래스가 있어야 합니다.)

드라이버 클래스 이름:

접속할 데이터베이스의 URL을 지정하십시오. URL의 형식은 JDBC 드라이버에 따라 달라집니다.

URL:

데이터베이스 접속을 생성할 때 사용할 데이터베이스 계정 사용자 이름을 지정하십시오.

데이터베이스 사용자 이름:

데이터베이스 접속을 생성할 때 사용할 데이터베이스 계정 비밀번호를 지정하십시오.
(참고: 안전한 비밀번호 관리를 위해, 아래의 속성 필드 대신에 비밀번호 필드에 비밀번호를 입력합니다.)

비밀번호:

비밀번호 확인:

데이터베이스 접속을 생성할 때 JDBC 드라이버로 전달할 속성을 지정하십시오.

속성:

데이터베이스 접속을 테스트할 때 사용할 테이블 이름 또는 SQL 문을 지정하십시오.

테스트 테이블 이름:

구성 테스트

뒤로

다음

완료

취소

10. [구성 테스트]를 클릭하여 접속 테스트가 성공하면 다음 단계로 넘어간다.
테스트에 실패한 경우 우선적으로 [연동 오류](#) 섹션을 참고하도록 한다.

메시지

✓ 접속 테스트가 성공했습니다.

새 JDBC 데이터 소스 생성

[구성 테스트](#)
[뒤로](#)
[다음](#)
[완료](#)
[취소](#)

11. JDBC 데이터 소스를 배치할 서버로 관리 서버를 선택 후 [완료]를 선택한다.

새 JDBC 데이터 소스 생성

뒤로 다음 완료 취소

대상 선택

새 JDBC 데이터 소스를 배치할 대상을 한 개 이상 선택할 수 있습니다. 대상을 선택하지 않으면 데이터 소스가 생성되지만 배치되지 않습니다. 나중에 데이터 소스를 배치해야 합니다.

서버

☒ AdminServer

뒤로 다음 완료 취소

12. JDBC 데이터 소스가 성공적으로 추가된 것을 확인한다.

JDBC 데이터 소스 요약

JDBC 데이터 소스는 JDBC 접속 풀을 통해 데이터베이스 접속을 제공하는 JNDI 트리에 바인딩된 객체입니다. 응용 프로그램은 JNDI 트리에서 데이터 소스를 조회한 다음 데이터 소스에서 데이터베이스 접속을 차용할 수 있습니다.

이 페이지에는 이 도메인에서 생성된 JDBC 데이터 소스 객체가 요약되어 있습니다.

이 테이블 사용자 정의

데이터 소스 (필터링됨 - 더 많은 열이 있음)

새로 만들기 삭제 1 - 1/1 표시 이전 다음

<input type="checkbox"/> 이름	JNDI 이름	대상
<input type="checkbox"/> altibase	altibase	AdminServer

새로 만들기 삭제 1 - 1/1 표시 이전 다음

커넥션 풀 설정 변경

JDBC 데이터 소스를 생성하면 관련 커넥션 풀이 함께 생성이 된다. 커넥션 풀은 여러 가지 속성을 가지는데 필요 시 다음 절차를 참고하여 설정을 변경하도록 한다.

예제로 제시된 속성 변경은 필수는 아니나 기본적으로 권장하는 사항이다.

1. 관리 콘솔 메인 화면에서 “데이터 소스” 항목을 선택, 변경하려는 JDBC 데이터 소스를 선택 후 [구성] 탭의 [접속 풀] 탭을 선택한다.

altibase5에 대한 설정

구성 대상 모니터링 컨트롤 보안 메모

일반 접속 풀 트랜잭션 진단 ID 옵션

저장

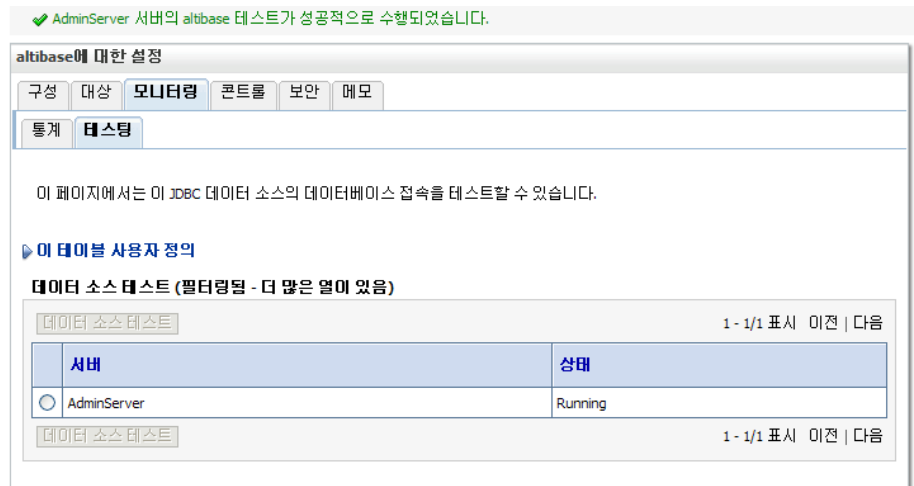
2. 아래 속성을 적절히 변경 후 [저장]을 선택한다.

- 초기 용량 (Initial Capacity)
기본적인 커넥션 풀의 개수로 기본값이 1 이나 업무 부하를 고려하여 이보다는 큰 수치를 권장한다.

확인

JDBC 데이터 소스 생성의 성공 여부를 최종적으로 확인한다.

- WebLogic 에서의 확인
관리 콘솔 메인 화면에서 “데이터 소스” 항목을 선택, 확인하려는 JDBC 데이터 소스를 클릭 후 [모니터링] 탭의 [테스팅] 탭에서 접속 성공 여부를 확인한다.



- ALTIBASE 에서의 확인
JDBC 데이터 소스의 접속 대상 ALTIBASE에서 세션 관련 쿼리를 수행, 설정한 커넥션 풀의 초기 용량만큼 세션이 존재하는지 확인한다.

아래는 초기 용량을 10 으로 지정한 경우의 실행 결과이다. JDBC 타입의 세션이 10 개인 것을 확인할 수 있다.

```
CMD> isql -s 127.0.0.1 -u sys -p manager -port 20300
```

```
-----
Altibase Client Query utility.
Release Version 5.3.3.12
Copyright 2000, ALTIBASE Corporation or its subsidiaries.
All Rights Reserved.
-----
```

```
ISQL_CONNECTION = TCP, SERVER = 127.0.0.1, PORT_NO = 20300
```

```
isql> set linesize 200
```

```
isql> set colsize 20
```

```
isql> select comm_name, client_type,
2 db_username, id session_id from v$session;
```

COMM_NAME	CLIENT_TYPE	DB_USERNAME	SESSION_ID
TCP 127.0.0.1:11274	JDBC	SYS	60
TCP 127.0.0.1:5126	JDBC	SYS	61
TCP 127.0.0.1:8710	JDBC	SYS	63
TCP 127.0.0.1:12294	JDBC	SYS	64
TCP 127.0.0.1:13318	JDBC	SYS	65
TCP 127.0.0.1:18694	JDBC	SYS	66
TCP 127.0.0.1:19462	JDBC	SYS	67
TCP 127.0.0.1:19974	JDBC	SYS	68
TCP 127.0.0.1:20486	JDBC	SYS	69
TCP 127.0.0.1:21254	JDBC	SYS	70
TCP 127.0.0.1:34311	CLI-32LE	SYS	72

11 rows selected.

웹 응용프로그램 생성 및 배치

연동을 위한 간략한 웹 응용프로그램을 생성하고 배치하는 예제이다.

웹 응용프로그램 생성

이름은 sample, 기본 페이지는 welcome.html 인 웹 응용프로그램을 생성한다.

1. 웹 응용프로그램의 이름으로 디렉토리를 생성 후 하위 디렉토리로 "WEB-INF" 를 생성한다.
2. WEB-INF 디렉토리에 아래 내용으로 web.xml 파일을 생성한다.

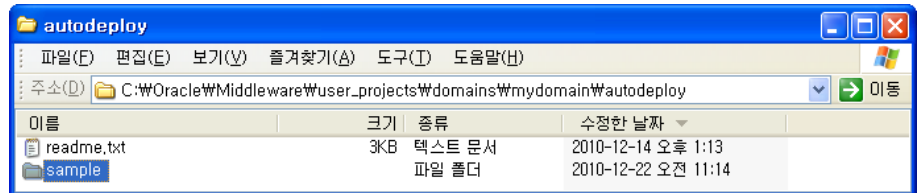
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD web Application
2.3//EN" "http://java.sun.com/dtd/web-app_2_3.dtd">
```

```
<web-app>
  <welcome-file-list>
    <welcome-file>welcome.html</welcome-file>
  </welcome-file-list>
</web-app>
```

배치

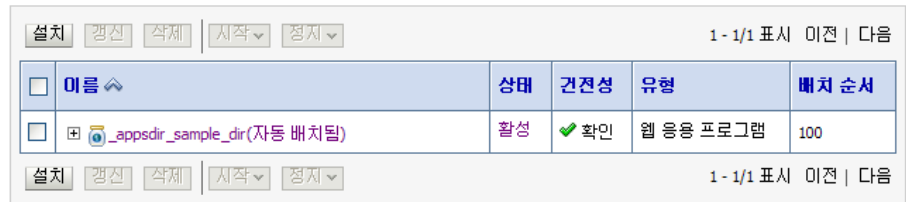
관리 콘솔에서 제공하는 배치 메뉴를 통하여 수동으로 직접 배치하여도 되나 편의상 개발 모드로 설치하였을 때 제공되는 \$DOMAIN_HOME/autodeploy 를 통한 자동 배치를 활용한다.

1. 생성한 웹 응용프로그램 디렉토리 전체를 \$DOMAIN_HOME/autodeploy 밑으로 위치 시킨다.



2. 관리 콘솔 메인 페이지 좌측에 “배치” 메뉴를 선택하여 자동배치 여부를 확인한다.

배치



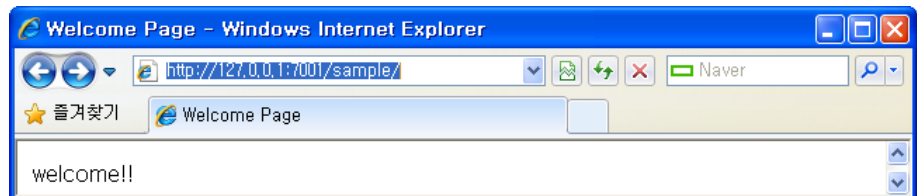
확인

웹 응용프로그램의 기본 페이지 예제를 통해 배치의 성공 여부를 최종적으로 확인한다.

1. sample 디렉토리에 아래 내용으로 **welcome.html** 를 생성한다.

```
<head><title>welcome Page</title></head>
<body>welcome!!</body>
</html>
```

2. 생성한 페이지에 접근하여 출력결과를 확인한다.



연동 예제

JDBC 데이터 소스의 JNDI를 통하여 ALTIBASE에 접속하는 예제를 수행함으로 ALTIBASE와 연동이 성공적으로 되었는지 최종적으로 확인한다.

ALTIBASE 연동

ALTIBASE 5 와 연동하는 웹 응용프로그램을 작성하여 접속 정보를 확인한다.

1. sample 디렉토리에 아래 내용으로 `singleVersion.jsp` 를 생성한다.

```
<%@ page contentType="text/html; charset=euc-kr" pageEncoding="euc-kr"%>
<%@ page import="java.sql.*, javax.naming.*, javax.sql.*"%>

<head><title>Single Version Test Page</title></head>

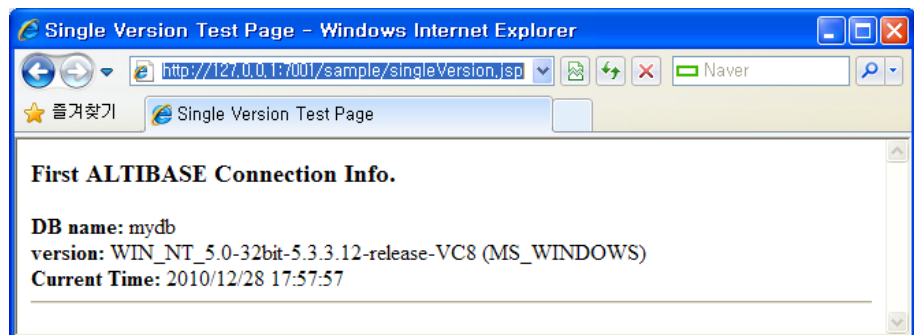
<body>
<%
    Context initCtx = new InitialContext();
    DataSource ds = (DataSource)initCtx.lookup("altibase");
    Connection sCon = ds.getConnection();
    Statement sStmt = sCon.createStatement();

    String query = "select '" +
        "<b>DB name: </b>'||trim(db_name)||'<br>" +
        "<b>version: </b>'||trim(product_signature)||'<br>" +
        "<b>Current Time: </b>" +
        "||to_char(sysdate,'yyyy/mm/dd hh24:mi:ss')||'<hr>' +
        "from v$database;";

    ResultSet sRes = sStmt.executeQuery(query);
    sRes.next();
    out.println("<font size=4><b>ALTIBASE Connection Info</b></font><p>");
    out.println(sRes.getString(1));

    sRes.close();
    sStmt.close();
    sCon.close();
%>
</body>
</html>
```

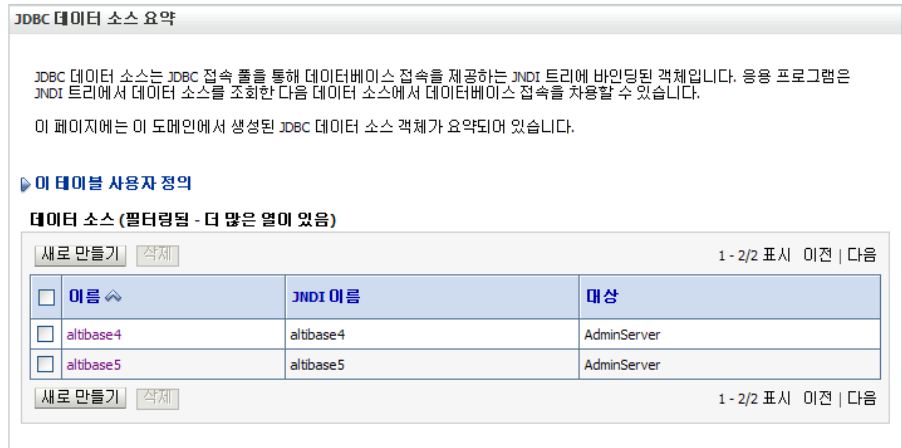
2. 생성한 페이지로 접근하여 연동 대상 ALTIBASE의 정보가 이상 없이 출력되는지 확인한다.



서로 다른 버전의 ALTIBASE 연동

ALTIBASE 4 와 ALTIBASE 5 와 동시에 연동하는 웹 응용프로그램을 작성하여 접속 정보를 확인한다.

1. [WebLogic에서 사용할 JDBC 드라이버를 설정하는 방법](#) 를 참조하여 ALTIBASE 4, ALTIBASE 5 에 각각에 대한 JDBC 드라이버 파일을 WebLogic에서 사용할 수 있도록 사전에 설정한다.
2. 각각에 대한 JDBC 데이터 소스를 생성한다. (JNDI는 JDBC 데이터 소스 이름과 같게 한다.)
 - Altibase.jar를 사용하는 ALTIBASE 4 에 대한 JDBC 데이터 소스 altibase4 (드라이버 클래스 이름: Altibase.jdbc.driver.AltibaseDriver)
 - Altibase5.jar를 사용하는 ALTIBASE 5 에 대한 JDBC 데이터 소스 altibase5 (드라이버 클래스 이름: Altibase5.jdbc.driver.AltibaseDriver)



- sample 디렉토리에 아래 내용으로 `multiVersion.jsp` 를 생성한다.

```
<%@ page contentType="text/html; charset=euc-kr" pageEncoding="euc-kr"%>
<%@ page import="java.sql.*, javax.naming.*, javax.sql.*"%>

<head><title>Multi Version Test Page</title></head>

<body>
<%
    Context initCtx = new InitialContext();
    DataSource ds1 = (DataSource)initCtx.lookup("altibase5");
    DataSource ds2 = (DataSource)initCtx.lookup("altibase4");
    Connection sCon1 = ds1.getConnection();
    Connection sCon2 = ds2.getConnection();
    Statement sStmt1 = sCon1.createStatement();
    Statement sStmt2 = sCon2.createStatement();

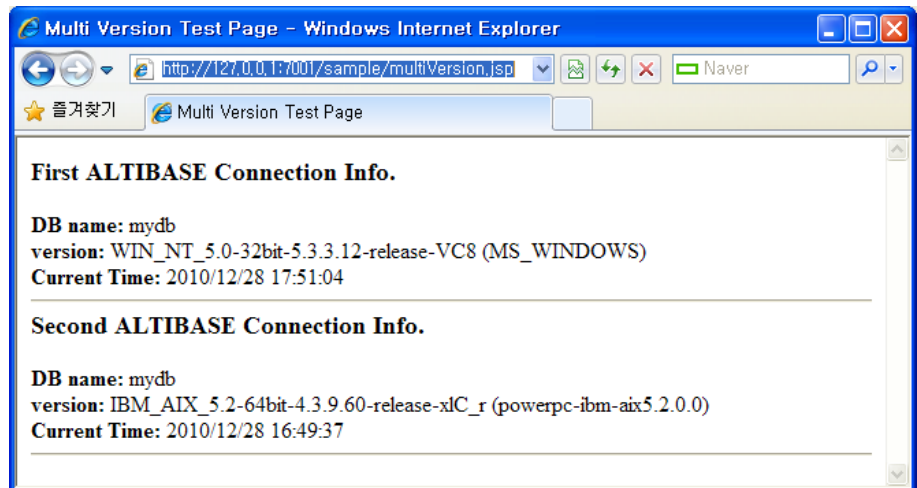
    String query = "select " +
        "<b>DB name: </b>||trim(db_name)||'<br>" +
        "<b>version: </b>||trim(product_signature)||'<br>" +
        "<b>Current Time: </b>" +
        "||to_char(sysdate,'yyyy/mm/dd hh24:mi:ss')||'<br>' +
        "from v$database;";

    ResultSet sRes1 = sStmt1.executeQuery(query);
    ResultSet sRes2 = sStmt2.executeQuery(query);
    sRes1.next();
    sRes2.next();

    out.println("<font size=4><b>First ALTIBASE Connection Info.");
    out.println("</font></b><p>" + sRes1.getString(1));
    out.println("<font size=4><b>Second ALTIBASE Connection Info.");
    out.println("</font></b><p>" + sRes2.getString(1));

    sRes1.close();
    sRes2.close();
    sStmt1.close();
    sStmt2.close();
    sCon1.close();
    sCon2.close();
%>
</body>
</html>
```

- 생성한 페이지로 접근하여 연동 대상 ALTIBASE의 정보가 이상 없이 출력되는지 확인한다.



Fail-Over & Load-Balancing 설정

WebLogic 또는 ALTIBASE 에서 Fail-Over 및 Load-Balancing 을 설정하는 방법에 대해 설명한다.

WebLogic 에서의 설정

다중 JDBC 데이터 소스 생성하면서 알고리즘으로 Fail-Over 또는 Load-Balancing 중 하나를 설정할 수 있다. 설정방법 및 상세한 내용은 관련 문서를 참조하도록 한다.

Fail-Over (장애 조치)

접속 요청을 다중 JDBC 데이터 소스 목록의 첫 번째 JDBC 데이터 소스에 전송하고, 요청이 실패할 경우 다음 JDBC 데이터 소스에 전송하는 방식으로 진행한다. 이러한 과정은 적합한 접속을 얻거나 목록의 끝에 도달할 때까지 계속되며 목록이 끝날 때까지 접속을 얻지 못하면 예외가 발생한다.

Load-Balancing (로드 균형 조정)

접속 요청을 다중 JDBC 데이터 소스 목록의 데이터 소스에 골고루 배치한다. 이 알고리즘은 Fail-Over 알고리즘을 기본적으로 제공한다.

ALTIBASE 에서의 설정

ALTIBASE 버전 5.3.3 부터 제공하는 라이브러리 계층의 Fail-Over 와 Load-Balancing을 이용하여 WebLogic과 연동하는 방법에 대해 설명한다. 설정방법 및 상세한 내용은 “ALTIBASE 이중화 매뉴얼”을 참조하도록 한다.

Fail-Over & Load-Balancing 개요

ALTIBASE는 Fail-Over로 CTF(Connection Time Fail-Over)와 STF(Service Time Fail-Over)를 제공하며 클라이언트 사이드의 Load Balancing 을 지원한다.

- Fail-Over

CTF	접속시점에 장애를 인식하여 다른 가용한 ALTIBASE 서버로 접속.
STF	접속 중에 장애 발생시 CTF 를 수행 후 세션상태(auto commit mode, optimizer mode, XA connection 등)를 복구한다. statement 에 대한 re-prepare, re-binding, execute 는 Fail-Over Callback 을 활용하여 사용자가 직접 수행하여야 한다.

- Load Balancing

접속 시점에 접속 가능한 ALTIBASE 서버 중에 하나를 무작위(Random)로 접속하는 클라이언트 사이드의 제한적인 Load Balancing.

설정 방법

JDBC에서는 관련 속성을 URL에 추가하여 설정한다. 관련 속성은 아래와 같다.

속성 명	설명
Server	기본 ALTIBASE 서버의 주소.
AlternateServers	가용 ALTIBASE 서버의 주소.
ConnectionRetryCount	가용 ALTIBASE 서버 접속실패 시 접속시도 반복횟수.

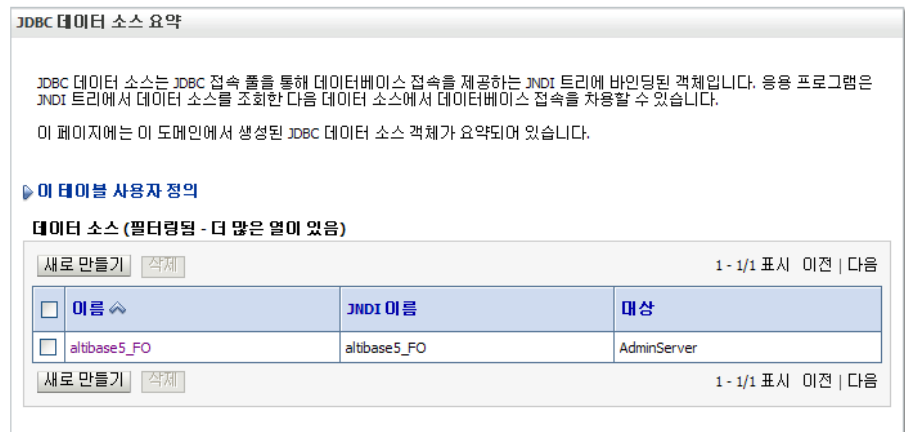
ConnectionRetryDelay	가용 ALTIBASE 서버 접속실패 시 재 접속시도 간격.(초 단위)
LoadBalance	로드밸런싱 여부. on 으로 설정하면 최초접속시도 시 기본 ALTIBASE 서버와 가용 ALTIBASE 서버를 포함하여 랜덤으로 선택한다. off 로 설정하면 최초접속시도 시에 기본 ALTIBASE 서버에 접속하고, 실패하면 가용 ALTIBASE 서버로 접속한다.
SessionFailOver	STF 를 할 것인지 여부, off 로 할 경우 CTF 만 수행된다.

설정 예제

ALTIBASE 서버 3 대(192.168.1.81, 192.168.1.82, 192.168.1.83)를 연동하는 시스템에서 장애 발생시 81, 82, 83 순으로 CTF와 STF를 사용하기 위한 예제이다.

1. JDBC 데이터 소스 altibase5_FO 를 아래와 같은 URL 로 하여 생성한다.

jdbc:Altibase://192.168.1.81:20300/mydb?AlternateServers=(192.168.1.82:20300, 192.168.1.83:20300)&ConnectionRetryCount=2&ConnectionRetryDelay=1&LoadBalance=off&SessionFailOver=on



JDBC 데이터 소스 요약

JDBC 데이터 소스는 JDBC 접속 풀을 통해 데이터베이스 접속을 제공하는 JNDI 트리에 바인딩된 객체입니다. 응용 프로그램은 JNDI 트리에서 데이터 소스를 조회한 다음 데이터 소스에서 데이터베이스 접속을 차용할 수 있습니다.

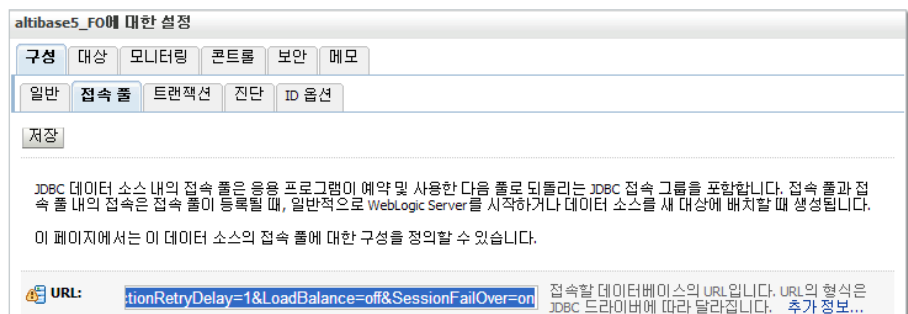
이 페이지에는 이 도메인에서 생성된 JDBC 데이터 소스 객체가 요약되어 있습니다.

이 테이블 사용자 정의

데이터 소스 (필터링됨 - 더 많은 열이 있음)

이름	JNDI 이름	대상
altibase5_FO	altibase5_FO	AdminServer

2. 생성된 JDBC 데이터 소스를 선택 후 [구성] 탭의 [접속 풀] 탭을 선택하여 URL이 끝까지 모두 정확히 인식되었는지 확인한다.



altibase5_FO에 대한 설정

구성 대상 모니터링 콘트롤 보안 메모

일반 접속 풀 트랜잭션 진단 ID 옵션

저장

JDBC 데이터 소스 내의 접속 풀은 응용 프로그램이 예약 및 사용한 다음 풀로 되돌리는 JDBC 접속 그룹을 포함합니다. 접속 풀과 접속 풀 내의 접속은 접속 풀이 등록될 때, 일반적으로 WebLogic Server를 시작하거나 데이터 소스를 새 대상에 배치할 때 생성됩니다.

이 페이지에서는 이 데이터 소스의 접속 풀에 대한 구성을 정의할 수 있습니다.

URL: jdbc:Altibase://192.168.1.81:20300/mydb?AlternateServers=(192.168.1.82:20300, 192.168.1.83:20300)&ConnectionRetryCount=2&ConnectionRetryDelay=1&LoadBalance=off&SessionFailOver=on

3. 지정한 횟수(100)만큼 연동 대상 ALTIBASE의 시간, 데이터베이스 이름, 포트번호, 버전상세정보를 출력하는데 Fail-Over가 성공적으로 수행되면 쿼리를 다시 prepare후 실패했던 작업(i) 부터 다시 수행하는 웹 응용프로그램을 생성한다.

```
<? page contentType="text/html; charset=euc-kr" pageEncoding="euc-kr"%>
<? page import="java.sql.*, javax.naming.*, javax.sql.*"%>

<head><title>FailOver Test Page</title></head>
<body>
<%
    Context          initCtx = new InitialContext();
    DataSource        ds      = (DataSource)initCtx.lookup("altibase5_FO");
```

```

Connection      sCon  = ds.getConnection();
PreparedStatement sStmt = null;
ResultSet       sRes  = null;

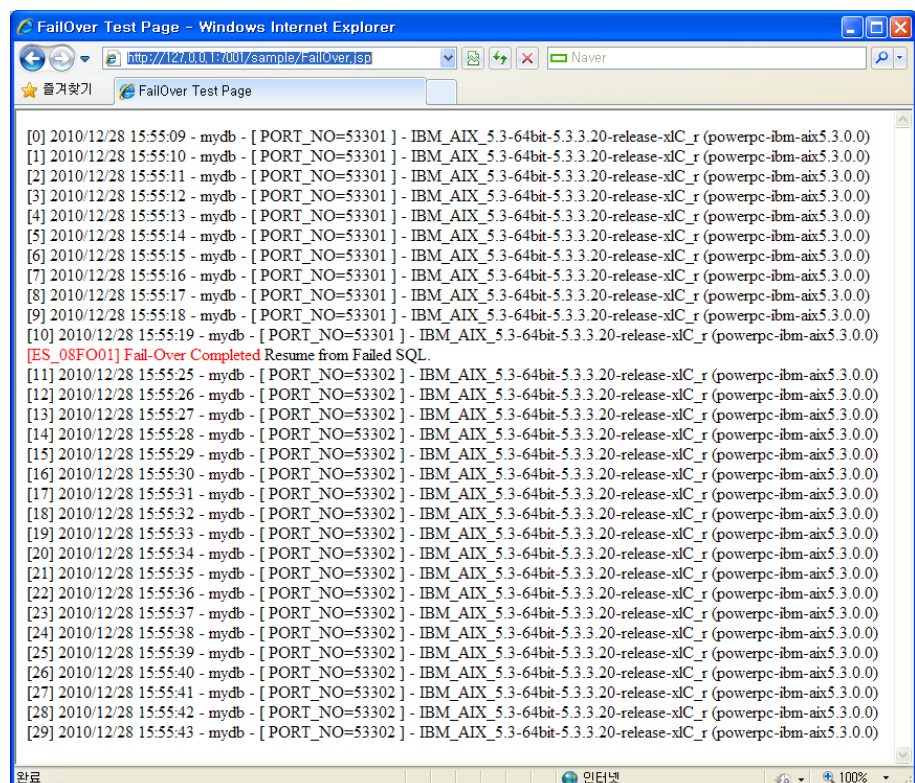
String query = "select to_char(sysdate,'yyyy/mm/dd hh24:mi:ss' || ' - ')" +
               "||trim(db_name)||' - '" +
               "||('select '[ PORT_NO='||value1||' ] - '" +
               "from v$property where name='PORT_NO')" +
               "||trim(product_signature)||<br> '" +
               "from v$database;";

sStmt = sCon.prepareStatement(query);

for (int i = 0; i < 100; i++) {
    try {
        sRes = sStmt.executeQuery();
        if ( sRes.next() )
        {
            out.println("["+i+"] "+sRes.getString(1));
            out.flush();
        }
        Thread.sleep(1000);
    } catch (SQLException e)
    {
        out.println("<font color=red>["+e.getSQLState()+"]" +
"+e.getMessage());
        if (e.getSQLState().equals("ES_08F001"))
        {
            out.println("</font> Resume from Failed SQL.<br>");
            sStmt = sCon.prepareStatement(query);
            i--;
            continue;
        }
        else break;
    }
}
sStmt.close();
sRes.close();
sCon.close();
%>
</body>
</html>

```

4. 생성한 페이지로 접근하여 연동 대상 ALTIBASE의 정보가 이상 없이 출력되는지 확인 후 현재 연동 대상인 ALTIBASE 를 종료시켰을 때 Fail-Over가 잘 되는지 확인한다.



연동 오류

ALTIBASE와 WebLogic 연동 시 발생할 수 있는 일반적인 오류에 대해서 설명한다.

Cannot load driver

JDBC 드라이버 설정이 정상적으로 완료되지 않은 상태에서 접속을 시도할 경우 발생한다.

[WebLogic에서 사용할 JDBC 드라이버를 설정하는 방법](#) 을 참조하도록 한다.

Can not make a database connection to the given URL / Invalid Altibase URL (No suitable driver)

아래와 같이 접속 URL에 오류가 있을 경우 오류가 발생한다.

- 잘못된 형식, 오타
- jdbc 항목을 “Altibase”가 아닌 다른 이름으로 설정한 경우 (이 경우는 버전에 따라 No suitable driver 가 발생하기도 한다.)

```
jdbc:otherDBMS://127.0.0.1:20300/mydb
```

Could not create pool connection

JDBC 데이터 소스 생성 시 커넥션 풀 설정에 오류가 있을 때 발생한다. 다양한 경우가 있으나 대표적인 경우는 아래와 같다.

- 접속 대상이 ALTIBASE 4 인 경우는 속성(property)에 “user=sys” 형식으로 접속할 사용자를 입력해야 한다.

Client unable to establish connection / Communication link failure

아래의 경우, Client unable to establish connection 오류가 발생한다. (버전에 따라 Communication link failure 가 발생하기도 한다.)

- ALTIBASE 서버가 구동되어 있지 않는 경우
- 접속 URL에서 ALTIBASE 서버의 ip 주소 또는 포트번호를 잘못 설정하여 접속이 불가능한 경우

아래는 연동할 ALTIBASE 서버의 ip 주소가 192.168.1.81 이고 포트번호가 20300 인데 다른 값으로 잘못 지정한 경우의 예이다.

```
jdbc:Altibase://192.168.1.80:20800/mydb
```

Hang 현상

ALTIBASE JDBC 드라이버와 ALTIBASE 서버의 CM 프로토콜 버전이 다른 상태에서 접속을 시도할 경우 WebLogic 버전에 따라 행(hang)과 같은 현상이 발생할 수도 있다.

이런 현상이 발생하게 되면 대부분 WebLogic 서버 인스턴스를 재 구동해야만 하므로 사전에 주의가 필요하다.

관련된 내용은 [ALTIBASE JDBC 드라이버 버전을 확인하는 방법](#) 을 참조하도록 한다.



알티베이스㈜

서울특별시 구로구 구로 3 동 182-13
대륭포스트 2 차 1008 호
02-2082-1000
<http://www.altibase.com>

대전사무소

대전광역시 서구 둔산동 921
주은리더스텔 901 호
042-489-0330

기술지원본부

서울특별시 구로구 구로 3 동 182-13
대륭포스트 2 차 908 호
02-2082-1000

기술지원센터

02-2082-1114
<http://support.altibase.com>

Copyright © 2000~2013 ALTIBASE Corporation. All Rights Reserved.

이 문서는 정보 제공을 목적으로 제공되며, 사전에 예고 없이 변경될 수 있습니다. 이 문서는 오류가 있을 수 있으며, 상업적 또는 특정 목적에 부합하는 명시적, 묵시적인 책임이 일체 없습니다. 이 문서에 포함된 ALTIBASE 제품의 특징이나 기능의 개발, 발표 등의 시기는 ALTIBASE 재량입니다. ALTIBASE는 이 문서에 대하여 관련된 특허권, 상표권, 저작권 또는 기타 지적 재산권을 보유할 수 있습니다.