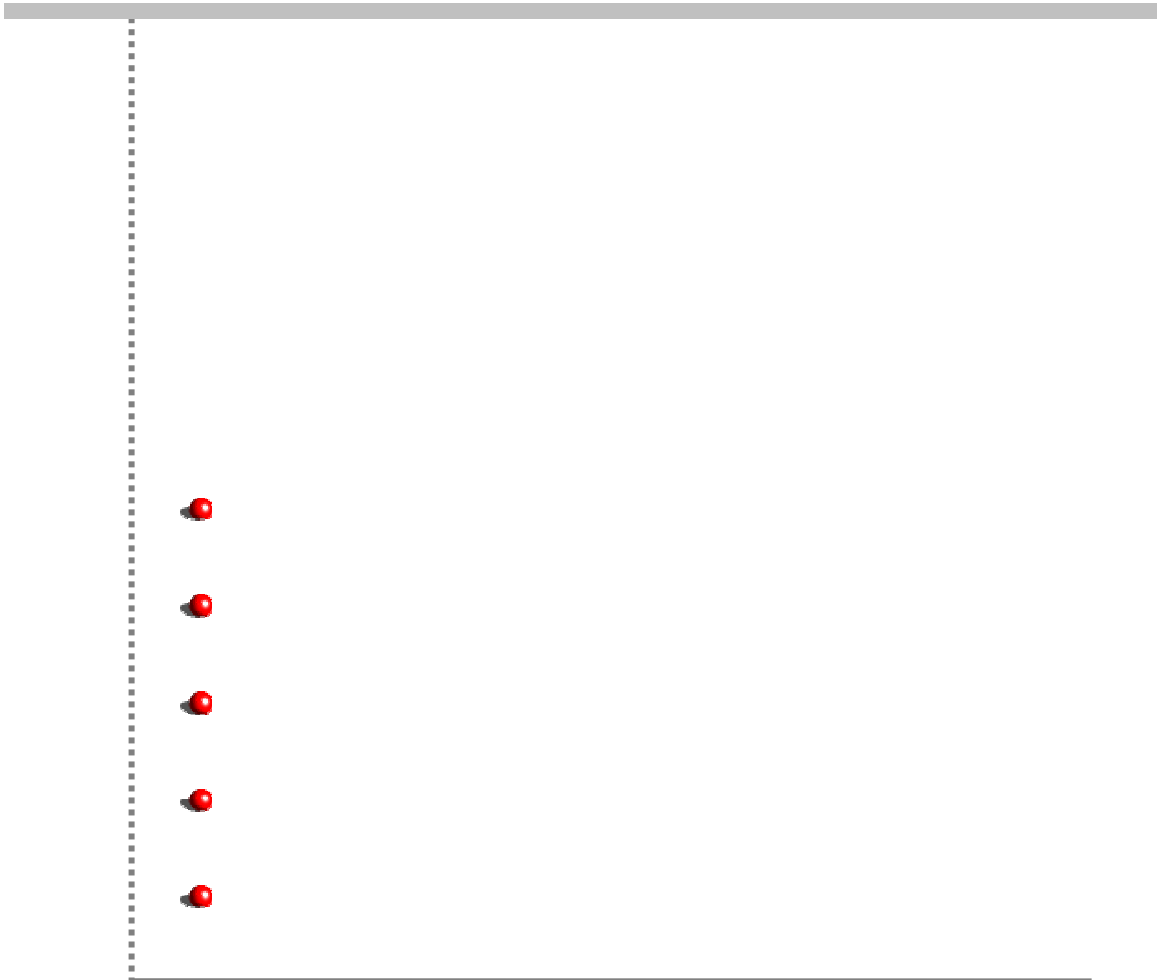


.	3
.....	4
.....	4
.....	5
.....	6
.....	6
. Features Overview	7
.....	8
Features	11
.....	15
.....	16
. Quick Tour	19
.....	20
.....	21
Step 1. ODI	22
Step 2.	29
Step 3.	46

		48
Step 1.		48
Step 2.		49
Step 3.		51
Step 4.		53
		55



본 매뉴얼은 웹 기반의 기업용 애플리케이션 개발 솔루션인 'OZ Application'을 처음 접하는 사용자에게 오즈 애플리케이션에 대한 이해와 오즈 애플리케이션 디자이너를 이용하여 프로젝트를 디자인하고 웹에 적용하는 기본적인 과정을 소개하기 위해 제작되었습니다.

본 매뉴얼은 Features Overview와 Quick Tour로 이루어져 있으며, 각 장의 내용은 다음과 같습니다.

Features Overview

오즈 애플리케이션의 특징과 기능, 아키텍처, 운영방식 등을 개괄적으로 설명합니다.

Quick Tour

오즈 애플리케이션 디자이너를 이용하여 프로젝트를 개발하고 웹으로 배포하기까지의 전 과정을 예제를 통해 살펴봅니다.

OZ Application을 포함한 관련 OZ 제품군의 매뉴얼은 아래와 같습니다.

Manual	Description
OZ Application Getting Started	오즈 애플리케이션에 대한 이해와 오즈 애플리케이션 디자이너를 이용하여 프로젝트를 만들고 웹에 적용하는 기본적인 과정을 소개합니다.
OZ Application Designer User's Guide	오즈 애플리케이션 디자이너의 화면 구성과 사용 방법 및 오즈 애플리케이션 디자이너를 이용하여 데이터와 연동하는 방법, OZC, OZS, OZF 개념과 사용 방법 등을 설명합니다.
OZ Query Designer User's Guide	오즈 쿼리 디자이너를 이용하여 데이터셋 정보를 작성하는 방법을 설명합니다.
OZ Enterprise Server Administrator's Guide	서버 관리자가 오즈 엔터프라이즈 서버를 효율적으로 관리할 수 있도록 오즈 엔터프라이즈 서버의 설치, 관리, 운영 방법을 설명합니다.
오즈 엔터프라이즈 서버 WAS 설치매뉴얼	오즈 엔터프라이즈 서버와 웹 애플리케이션 서버와의 연동 방법을 설명합니다.

본 매뉴얼을 보다 쉽게 이해하기 위해서는 다음과 같은 사전 지식이 필요합니다.

- DB 관련지식 (JDBC/ODBC, SQL 작성 등)
- 오즈 엔터프라이즈 서버와 연동될 WAS(Web Application Server)에 대한 기본 지식
- 웹페이지 제작 기본 지식(HTML, JavaScript 등)

본 제품은 웹 기반의 애플리케이션을 개발 및 운영하고자 하는 기업, 관공서 또는 단체를 대상으로 합니다.

본 제품은 웹을 기반으로 하는 모든 애플리케이션 개발에 활용 가능합니다.

대표적으로 적용되고 있는 분야는 다음과 같습니다.

- 기업내 기간 업무 시스템 (Intranet)
- 기업간 거래 시스템 (B2B)
- B2C 및 웹 포털
- 민원 서비스, 대학/병원 업무 (G2B, G2C)
- ASP/MSP 등의 서비스

. Features Overview



Features





< 오즈 제품 구성 >

오즈 애플리케이션 디자이너는 오즈 쿼리 디자이너, 오즈 엔터프라이즈 매니저, 오즈 리파지토리 매니저, 오즈 애플리케이션 뷰어, 오즈 엔터프라이즈 서버 등 다수의 모듈로 구성되어 있으며, 상호 긴밀하게 연동되어 최상의 웹 기반 솔루션 개발 환경을 제공합니다.

(OZ Application Designer)

- C/S 환경의 강력한 사용자 인터페이스를 웹에서 그대로 구현할 수 있도록 웹 애플리케이션 개발의 편리성과 유연성을 증대시키는 프로그램으로 웹 기반 애플리케이션 개발 툴입니다.
- 컴포넌트 기반의 개발 환경으로 다양하고 강력한 **Built-in** 객체와 기능을 제공하여 복잡하고 다양한 웹 애플리케이션을 유연성있게 개발하고 쉽게 유지보수할 수 있도록 합니다.
- 오즈 리포트 제품군과 호환 가능하여 다양한 보고서를 표현할 수 있습니다.

(OZ Query Designer)

- 애플리케이션 디자인에 필요한 데이터셋(ODI:OZ Data Info)을 생성할 수 있는 쿼리 디자인 도구입니다.
- DB 연결 및 쿼리 개발 과정을 비전문가도 수행할 수 있는 쿼리 도구로서, 테이블 간의 관계를 시각적으로 보여줍니다.
- Drag & Drop 방식으로 누구나 쉽게 쿼리를 디자인 할 수 있으며 디자인한 결과를 즉시 실행해 볼 수 있어 정확한 쿼리문을 생성할 수 있도록 도와 줍니다.

(OZ Enterprise Manager)

- 오즈 애플리케이션 디자이너를 이용하여 구축된 웹 애플리케이션 환경 하에서 서버의 운영 상태에 대한 모니터링 및 스케줄링 등을 통해 서비스 운용의 편의성을 극대화하는 관리자용 도구입니다.
- 다수의 오즈 엔터프라이즈 서버의 통합 관리 기능 및 서버의 재구동 없이 운영 중인 서버의 각종 설정 정보를 변경할 수 있는 Non-Stop 관리 환경을 제공합니다.

(OZ Repository Manager)

- 오즈 엔터프라이즈 서버와 연동하여 서버의 리파지토리를 편리하게 탐색 및 관리하기 위한 관리자용 도구입니다.
- 애플리케이션 프로젝트 파일, ODI 파일 등 각종 아이템 저장 및 수정에 대한 히스토리와 버전 관리, 계층적인 카테고리 관리 및 권한 설정 기능을 제공합니다.

(OZ Application Viewer)

- 오즈 애플리케이션 디자이너로 디자인한 프로젝트를 웹을 통해 보여주는 오즈 애플리케이션 전용 뷰어로서 웹 상에서 데이터 검색 기능과 사용자 입력 등의 기능을 제공합니다.
- 오즈 애플리케이션 디자이너로 개발된 User Interface를 통해 사용자 입력값을 처리하고 데이터를 조회, 입력, 수정하는 등을 수행하는 클라이언트 모듈입니다.
- Full ActiveX 방식으로 모든 컴포넌트를 한 화면에 표현할 수 있으며, 자동 업데이트 기능을 제공합니다.

(OZ Enterprise Server)

- 순수 자바 기반의 서버로서 오즈 애플리케이션 뷰어의 요청을 받아 결과 데이터 셋(**Result Set**) 및 애플리케이션 프로젝트를 뷰어에게 전달합니다.
- 리파지토리를 이용하여 사용자/사용자 그룹별 권한 관리 및 애플리케이션 프로젝트 파일의 버전 관리 기능이 지원되어 효율적인 애플리케이션 개발 및 관리가 가능합니다.
- 고성능의 다중 스레드 처리 방식의 서버 아키텍처를 채택하여 다수 사용자 동시접속 환경에서도 최상의 성능을 보장합니다.
- 독립적으로 구동되는 데몬 형식의 서버와 서블릿 엔진 위에서 구동되는 서블릿 형식의 서버를 제공함으로써 **WAS**를 비롯한 다양한 서버 실행 환경에 적용할 수 있습니다.

Features

■ 시각적인 디자인 환경

메뉴 및 툴바와 디자인 영역과 스크립트 편집 영역 등의 화면 구성으로 편리하고 직관적인 디자인 환경을 제공합니다. **Drag & Drop** 방식으로 컴포넌트 추가가 용이하며, 복잡하고 다양한 옵션 설정은 마법사 기능을 이용하여 쉽고 빠르게 구현할 수 있습니다.

■ 고수준의 컴포넌트 제공

오즈 애플리케이션은 일반 컴포넌트, 데이터 연동 컴포넌트, 리포트 연동 컴포넌트, 기타 컴포넌트 등 기업용 애플리케이션 개발을 위해 필요한 고급 컴포넌트를 기본적으로 제공합니다.



• 사용자 입력 컴포넌트

텍스트나 값을 입력하거나 선택하는 컴포넌트(TextBox, RichTextBox, MaskedTextBox, ComboBox), 숫자값을 위아래로 조절하는 컴포넌트(NumericUpDown), 검색 설정 옵션 등을 선택하는 컴포넌트(RadioButton, CheckBox, CheckListBox), 날짜 등을 선택하는 컴포넌트(DateTimePicker, MonthCalendar), 사용자 명령(확인/취소/재시도 등)을 입력받는 컴포넌트(Button) 등을 제공합니다.

• 표시 컴포넌트

텍스트를 표시하는 컴포넌트(Label), 다수의 아이템을 리스트 형태로 표시하는 컴포넌트(ListBox, CheckListBox, TreeView), 그림을 표시하는 컴포넌트(PictureBox), 진행 상태를 표시하는 컴포넌트(ProgressBar), 웹페이지를 나타낼 수 있는 컴포넌트(HTMLView) 등을 제공합니다.

• 확장 컴포넌트

오즈 애플리케이션 디자이너가 제공하지 않는 컴포넌트를 사용하는 경우를 위해 ActiveXControl, CustomControl 컴포넌트 등을 제공합니다.

- 도형 컴포넌트

선, 화살표, 원형, 사각형 등 일반적인 도형 컴포넌트를 제공합니다.



- 2D/3D Chart 컴포넌트

객은선, 막대, 원형, 산포형, 사용자 정의형 등 다양한 종류의 2D/3D Chart를 지원하며 Chart 스타일을 결정하는 다양한 조건 설정 기능과 색상, 점, 면, 라인 등을 마법사를 이용하여 쉽게 설정할 수 있습니다.

- Table 컴포넌트

쿼리 디자이너를 통해 생성된 데이터 셋의 내용을 별도의 스크립트 입력없이 마법사를 이용하여 쉽게 나타내며, 다수의 데이터를 페이지를 나누어 표시하는 페이지 나눔 기능(Horizontal Navigator, Vertical Navigator) 및 데이터 정렬 기능, 엑셀 저장 기능 등을 제공합니다.

- CrossTab 컴포넌트

쿼리 디자이너를 통해 생성된 데이터 셋의 내용을 별도의 스크립트 입력없이 마법사를 이용하여 쉽게 나타내며, 필드별 합계 기능, 드릴링 기능 등을 제공합니다.

- Grid 컴포넌트

쿼리 디자이너를 통해 생성된 데이터 셋의 내용을 나타내거나 사용자가 임의로 값을 추가, 삭제, 편집할 수 있으며, 정렬 기능, 셀 병합 기능, 엑셀 저장 기능 등을 제공합니다.



- 리포트 뷰어 호출 컴포넌트

오즈 리포트 뷰어를 호출하여 오즈 리포트 디자이너로 작성한 보고서 파일을 오즈 애플리케이션 디자이너에서 미리보기 할 수 있습니다.

- 리포트 인쇄 및 Export 컴포넌트

오즈 리포트 디자이너로 작성한 보고서 파일을 미리보기를 하지 않고 바로 인쇄하거나 다른 파일로 저장할 수 있습니다.



- 그룹 컴포넌트

서로 다른 여러 종류의 컴포넌트를 그룹지을 수 있는 **GroupBox**, **Panel**, **TabControl** 등을 제공합니다.

- Invisibles 컴포넌트

팝업 메뉴를 표시할 수 있는 **ContextMenu**, 화면 상단에 주 메뉴를 표시할 수 있는 **MainMenu**, 아이템 앞에 아이콘 형태의 이미지를 표시할 수 있는 **ImageList**, 일정한 시간 간격으로 특정 명령을 실행하게 하는 **Timer**, 버튼 등의 풍선 도움말을 나타낼 수 있는 **ToolTip** 컴포넌트 등을 제공합니다.

■ 다양한 속성, 메소드, 이벤트, 상수 제공과 JScript 지원

각각의 컴포넌트별로 다양한 속성, 메소드, 이벤트, 상수 등을 제공하며, **JScript**를 지원하여 복잡한 애플리케이션을 쉽게 구현할 수 있습니다.

,

■ 다양한 데이터 소스 지원

오즈 애플리케이션은 기업 내에 다양한 형태로 존재하는 데이터를 활용하기 위한 데이터 연동 방안을 지원합니다. 일반 텍스트 파일, **XML**, **EJB** 등의 **Non-DB** 데이터나 사용자 애플리케이션 데이터는 오즈의 **UDS(User Data Store)** 연동 모듈을 통해 처리합니다. 또한 이기종 **DBMS** 동시 연동이 가능하여, 하나의 프로젝트에 여러 **DBMS**에서 가져온 데이터를 연동할 수 있습니다.

■ 실시간 대용량 데이터 처리 가능

데이터의 실시간 처리가 가능하며, 데이터 압축 전송 기능으로 대용량 데이터를 빠른 시간 내로 처리할 수 있습니다.

■ OZC(OZComponent)

오즈 애플리케이션은 **ComboBox**, **ListBox**, **Chart** 등 서로 다른 여러 종류의 컴포넌트를 하

나의 컴포넌트 즉 OZC로 모듈화하는 기능을 제공하여 복잡한 컴포넌트를 쉽게 구현할 수 있도록 합니다.

■ OZF(OZFunction)

오즈 애플리케이션은 공통적으로 쓰이는 함수를 OZF로 모듈화하여 다수 사용자간의 정보 공유를 용이하게 합니다.

■ OZS(OZStyle)

오즈 애플리케이션은 컴포넌트의 글꼴, 배경색, 크기 등과 같은 스타일을 OZS로 모듈화할 수 있으며 단순한 반복 작업을 없애 개발 시간을 단축할 수 있도록 합니다.

, ,

■ 효율적인 리파지토리 관리 기능

서버 리파지토리를 통해 애플리케이션 프로젝트 파일의 카테고리별 관리 및 생성/수정에 따른 히스토리 기능을 제공하여 관리 및 효율성을 증대시킵니다.

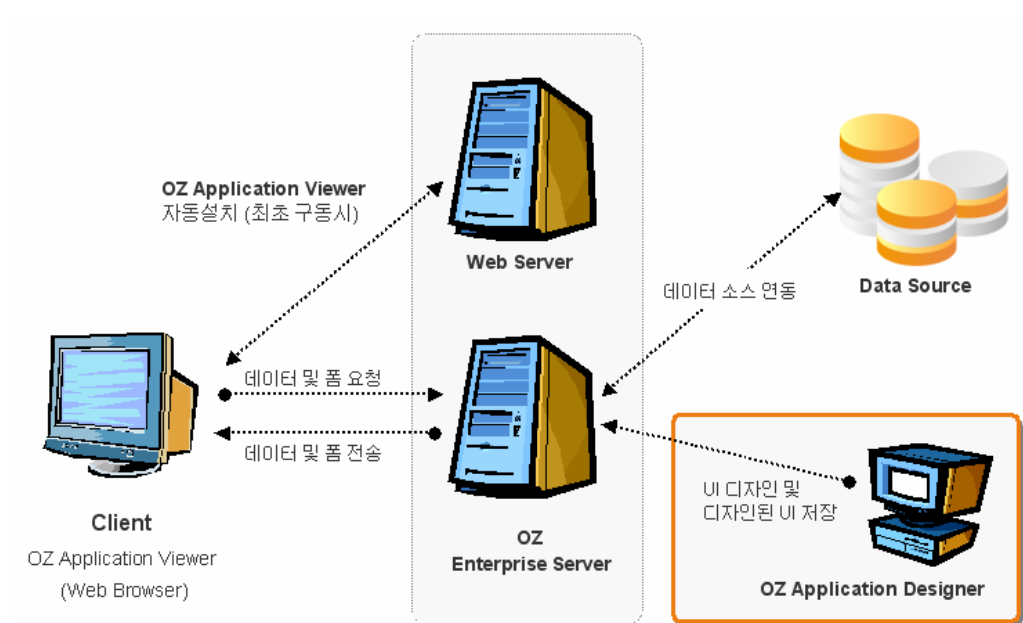
■ 대규모 사용자 환경을 위한 안정적인 서비스

오즈 애플리케이션은 클라이언트에서 작업을 수행하므로 네트워크 및 서버 부하를 감소시키며, 애플리케이션 프로젝트 파일의 압축 전송 기능으로 전송 효율을 극대화합니다. 또한 네트워크 전송 데이터의 위변조를 방지하여 안정적인 서비스를 제공합니다.

■ 오즈 리포트와 연계된 고품질의 보고서 서비스

오즈 애플리케이션은 오즈 리포트와 쉽게 연동되어 고품질의 리포팅 서비스 및 인쇄 기능을 제공합니다.

오즈 애플리케이션의 개발 및 운영 흐름도는 아래와 같습니다



개발 과정은 애플리케이션 개발자가 오즈 쿼리 디자이너로 ODI 파일을 생성하며 오즈 애플리케이션 디자이너에서 애플리케이션 프로젝트 양식을 디자인하고, 애플리케이션 프로젝트 파일과 ODI 파일을 오즈 서버의 리파지토리에 저장하는 작업 과정을 의미합니다.

운영 과정은 최종 사용자가 웹 애플리케이션을 통해 애플리케이션 프로젝트 파일을 요청하여 애플리케이션 프로젝트 파일이 생성 및 전달되는 과정을 의미합니다

■ OS

Windows NT 4.0 이상, Windows XP, Windows 2000, Windows 2003
JRE 1.3 이상

■ 하드웨어

CPU : Pentium 500MHz 이상 권장
RAM : 256MB(512MB 이상 권장)
모니터 : 컬러 XGA(1024 x 768 이상의 해상도 권장)

■ OS

Windows 98 이상

※ 유니코드는 MS Windows NT계열(NT4.0/2000/XP/2003)에서만 지원됩니다.

■ 하드웨어

CPU : Pentium 300MHz 이상 권장
RAM : 64MB(128MB 이상 권장)
모니터 : 컬러 SVGA(800 x 600 이상의 해상도 권장)

■ OS

JRE(Java Runtime Environment) 1.1.X 이상이 지원되는 모든 플랫폼(JRE 1.3.X 권장)
(Windows NT/2000/XP/2003, Linux, HP-UX, Solaris, AIX, Unixware 등)

※ JRE 1.4이상은 별도의 제품을 제공합니다.

■ 웹 서버

Internet Information Server(IIS), Netscape Enterprise Server, Apache, NCSE, CERN 등 HTTP를 지원하는 웹 서버

■ 웹 애플리케이션 서버

WebLogic, WebSphere, Jeus, Tomcat, JRun, iPlanet, SilverStream, NetAction, EAServer, iAS, AppServer, NEC WebOTX, Fujitsu Interstage 등

■ 데이터베이스

Oracle, MS SQL Server, DB2(UDB), Informix, Sybase, IngresII 등 JDBC 또는 ODBC를 지원하는 모든 상용 DBMS

/

■ OS

Windows NT 4.0 이상, Windows XP, Windows 2000, Windows 2003

■ 하드웨어

CPU : Pentium 500MHz 이상 권장

RAM : 128MB(256MB 이상 권장)

모니터 : 컬러 XGA(1024 x 768 이상의 해상도 권장)

. Quick Tour



Step1. ODI
Step2.
Step3.



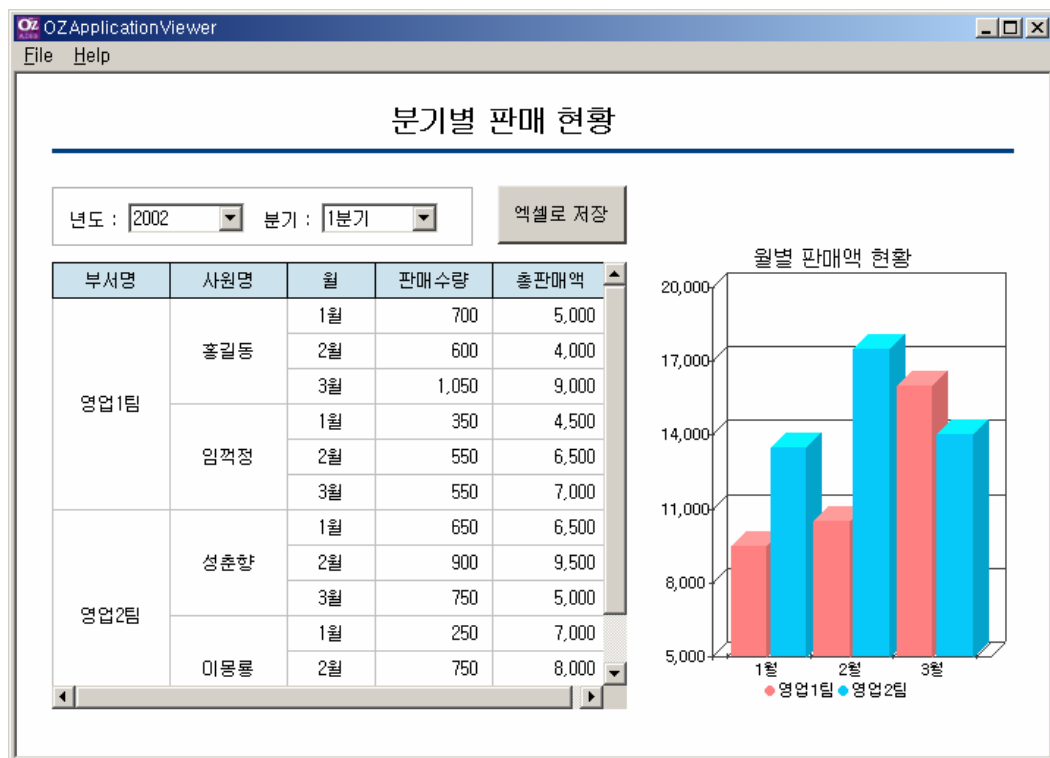
Step1.
Step2.
Step3.
Step4.

- 윈도우 플랫폼(MS Windows NT 4.0 이상)
- 웹 서버로 인터넷 정보메이션 서버(IIS) 설치가 되어 있어야 합니다.
- 오즈 엔터프라이즈 서버(데몬 타입)와 오즈 애플리케이션 디자이너, 오즈 쿼리 디자이너가 설치되어 있어야 합니다.
- 샘플 데이터베이스가 MS Access DB 파일로 제공되므로 MS Access ODBC Driver가 설치되어 있어야 합니다.
- 웹에서 애플리케이션 서비스를 위해서 IIS 웹 서버가 구동되고 있어야하며, 뷰어 설치 태그에서 뷰어 다운로드로 지정된 폴더에 오즈 애플리케이션 뷰어 설치 파일이 존재하여야 합니다.



애플리케이션 프로젝트를 만들기 위해서는 우선 애플리케이션의 목적을 이해하고 애플리케이션에 이용할 데이터 소스의 구조에 대해 충분히 숙지하고 있어야 합니다. 이후 디자인할 화면에 대해 구상을 한 후 실제 디자인 작업에 들어갑니다.

그럼, 실제로 기업의 분기별 판매현황을 분석하는 다음과 같은 애플리케이션을 만들어 보도록 하겠습니다.



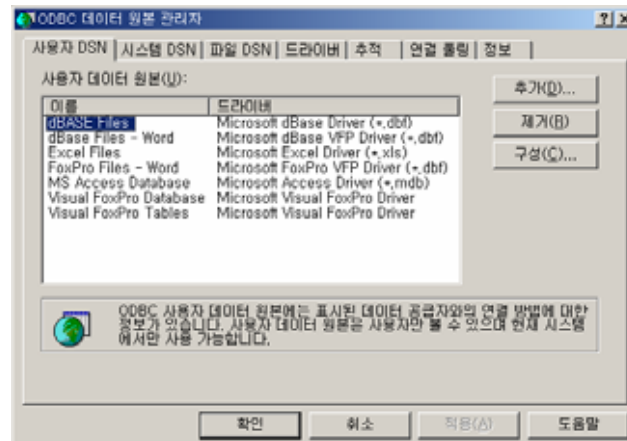
년도와 분기를 검색 조건으로하여 분기별 판매 현황을 검색하고 검색된 내용을 테이블과 차트를 이용하여 나타냅니다

Step 1 ODI

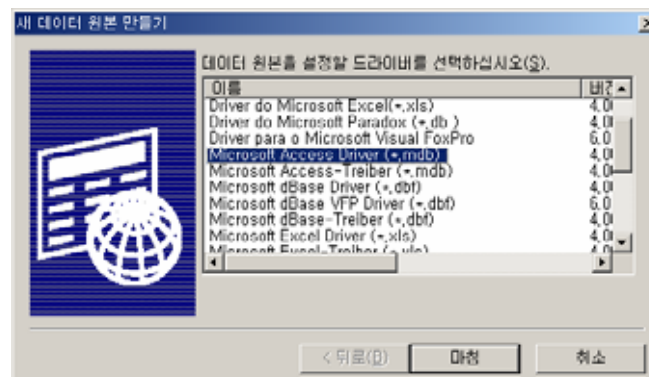
◆ ODBC 데이터 원본 설정

샘플 DB 파일을 ODBC 관리자를 통해 ODBC 데이터 소스로 등록합니다. 이용할 Microsoft Access DB 파일은 AppSample.mdb이며, DB 파일은 고객 지원 센터로 문의하시기 바랍니다. AppSample.mdb를 ODBC 시스템 데이터 원본으로 등록하고 데이터 원본 이름(DSN)은 'AppSample'로 설정합니다.

- 1) [제어판]의 [관리 도구]에서 [데이터 원본(ODBC)]을 선택하여 ODBC 데이터 원본 관리자를 실행합니다.

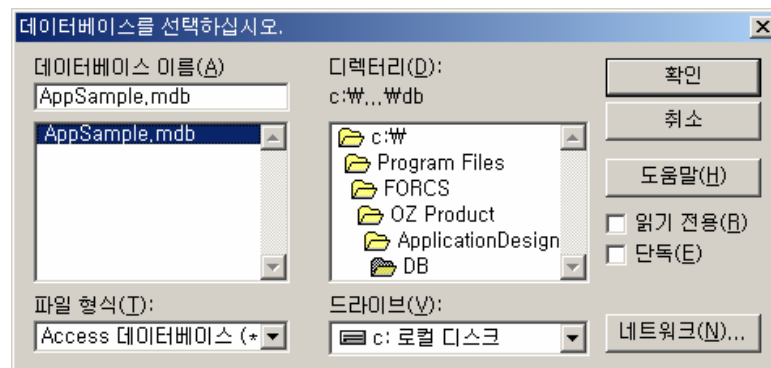


- 2) [시스템 DSN] 탭을 선택하고 [추가] 버튼을 클릭한 후 Microsoft Access Driver(*.mdb)를 선택하고 [마침] 버튼을 클릭합니다.

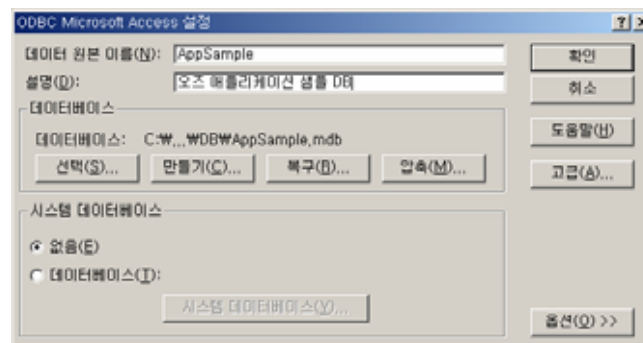


- 3) 데이터 원본 이름을 'AppSample'로 입력하고, [선택] 버튼을 눌러

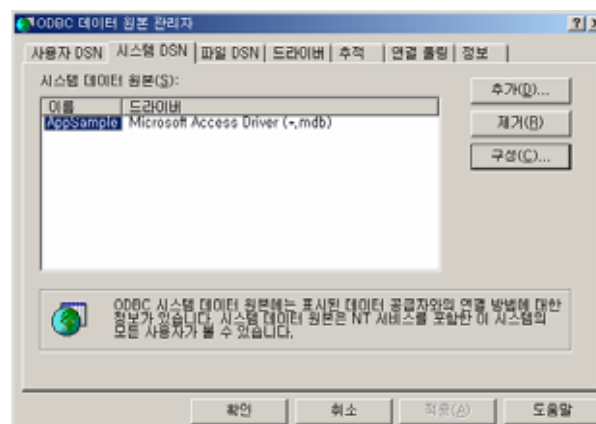
'AppSample.mdb' 파일을 선택합니다.



4) 설명 등을 추가한 후에 [확인] 버튼을 클릭합니다.



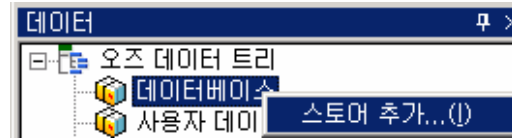
5) 'AppSample'라는 데이터 원본이 추가된 것을 확인합니다.



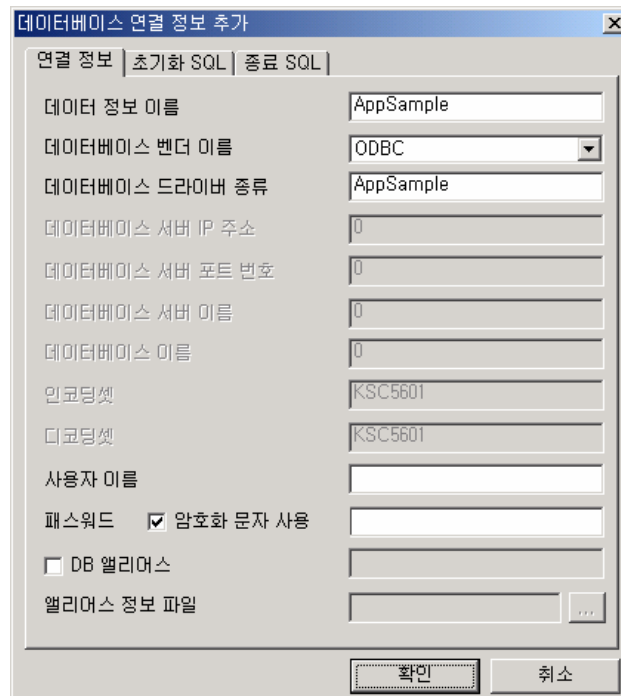
◆ 데이터베이스 정보 추가

오즈 쿼리 디자이너(OZ Query Designer)를 실행하여 연동할 데이터 베이스 정보를 추가합니다.

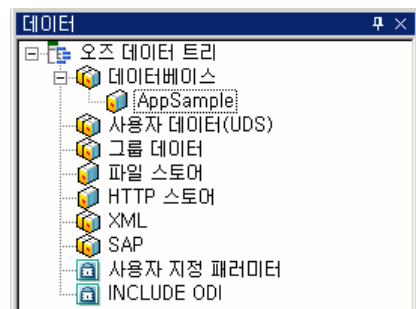
- 1) 오즈 쿼리 디자이너의 데이터창의 [데이터베이스] 항목에서 마우스 오른쪽 버튼을 클릭하여 나타나는 팝업 메뉴에서 [스토어 추가]를 선택합니다.



- 2) [데이터 정보 이름]은 사용자가 임의로 정할 수 있습니다. 본 예제에서는 'AppSample'로 입력하고 [데이터베이스 벤더 이름]은 'ODBC'로 선택한 후 [데이터베이스 드라이버 종류]는 ODBC 데이터 원본에서 설정한 'AppSample'로 입력합니다.



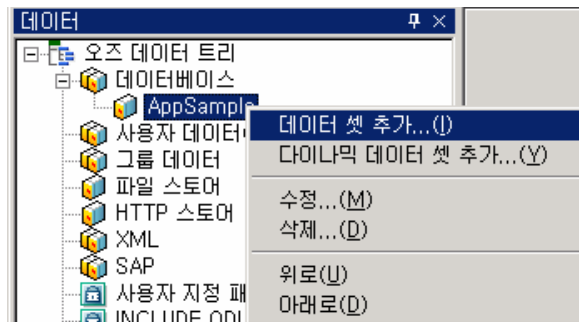
- 3) 추가된 데이터베이스 정보를 데이터창에서 확인합니다.



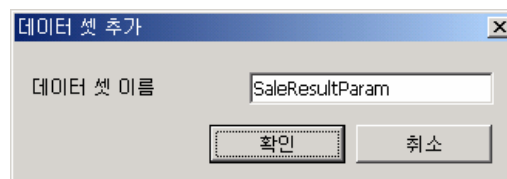
◆ 데이터 셋 추가와 쿼리 생성

데이터 셋을 추가하고 표시할 데이터를 가져올 쿼리를 생성합니다.

- 1) [AppSample] 항목에서 마우스 오른쪽 버튼을 클릭하여 나타나는 팝업 메뉴에서 [데이터 셋 추가]를 선택합니다.

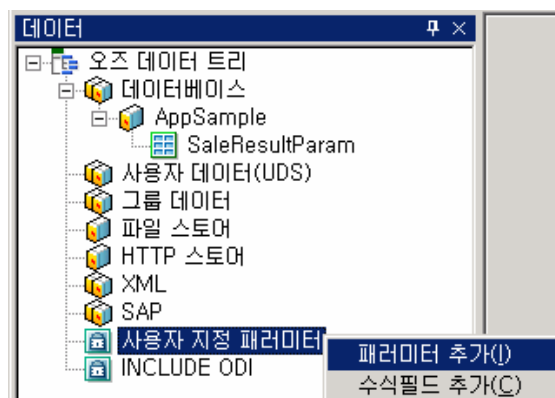


- 2) [데이터 셋 이름]에 'SaleResultParam'을 입력한 후 [확인] 버튼을 클릭하여 데이터 셋을 추가합니다.

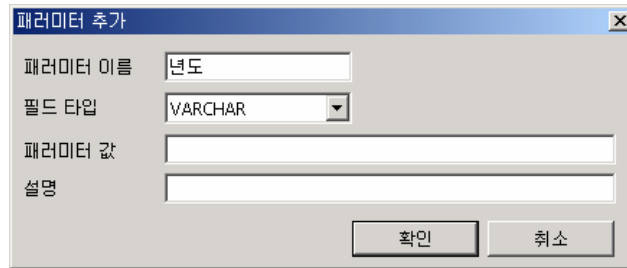


- 3) 검색 조건을 입력받는 패러미터를 추가합니다.

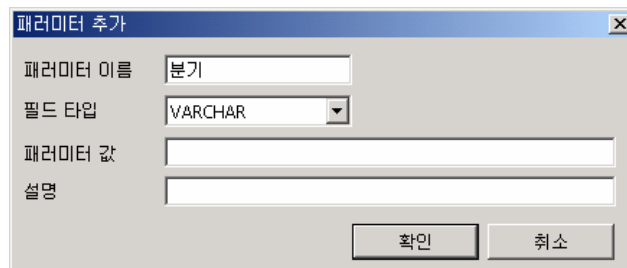
데이터창에서 [사용자 지정 패러미터]를 마우스 오른쪽 버튼으로 클릭하여 나타나는 팝업 메뉴에서 [패러미터 추가]를 선택합니다.



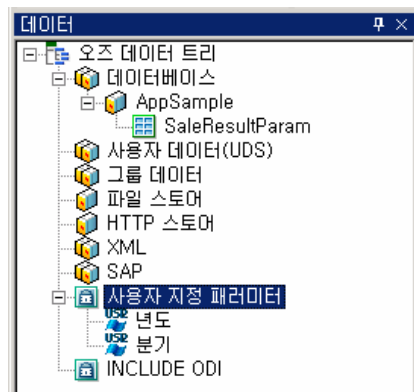
검색 조건 중 년도 값을 입력받을 패러미터의 이름을 '년도'로 입력한 후 [확인] 버튼을 눌러 추가합니다.



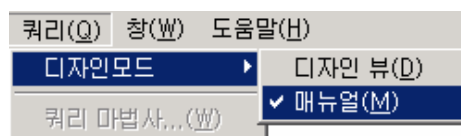
검색 조건 중 분기 값을 입력받을 패러미터의 이름을 '분기'로 입력한 후 [확인] 버튼을 눌러 추가합니다.



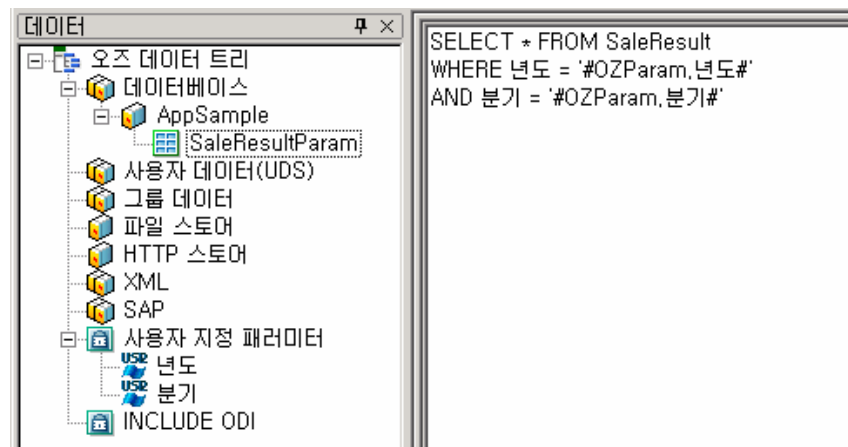
[사용자 지정 패러미터]에 패러미터가 제대로 추가되었는지 확인합니다.



- 4) 데이터 셋을 가져오는 쿼리를 직접 입력하기 위해 디자인 모드를 매뉴얼 모드로 변경합니다. 데이터창에서 'SaleResultParam'을 클릭한 후 [쿼리] 메뉴 → [디자인모드] 메뉴 → [매뉴얼] 메뉴를 선택합니다. 쿼리를 직접 입력하지 않고 디자인뷰 모드를 이용하여 쿼리를 생성하는 방법은 '오즈 쿼리 디자이너 사용자 매뉴얼'을 참조하십시오.



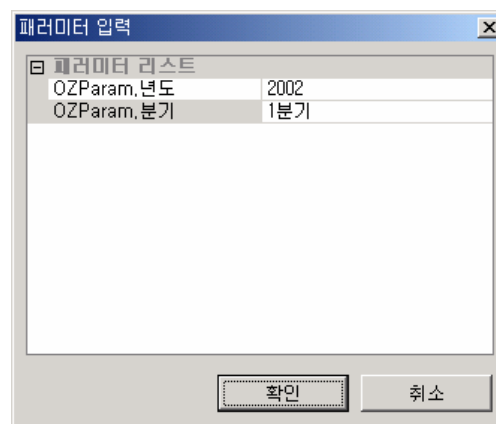
- 5) 쿼리 입력 창에 쿼리를 입력합니다.



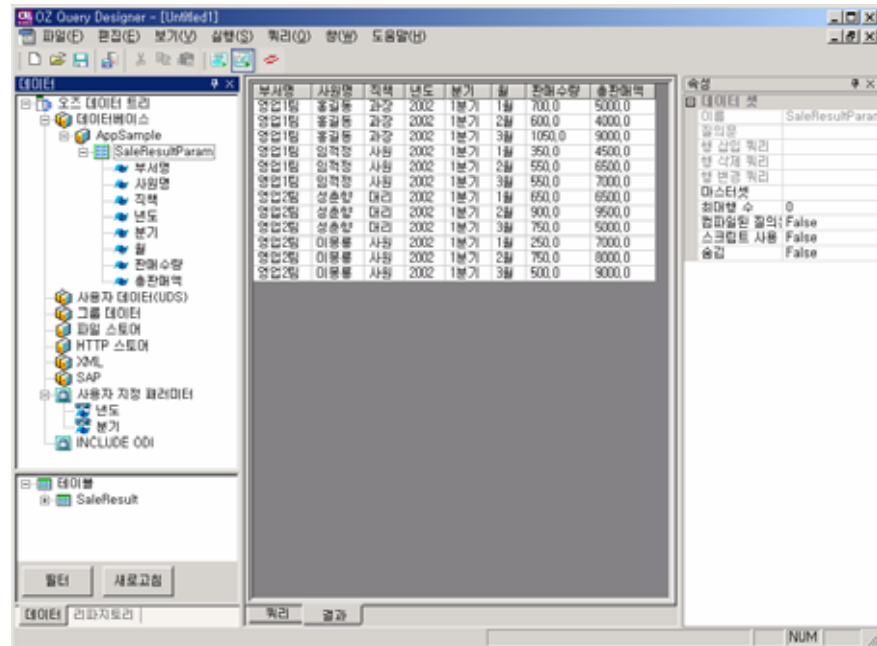
입력한 쿼리 내용은 다음과 같습니다.

```
SELECT * FROM SaleResult
WHERE 년도 = '#OZParam.년도#'
AND 분기 = '#OZParam.분기#'
```

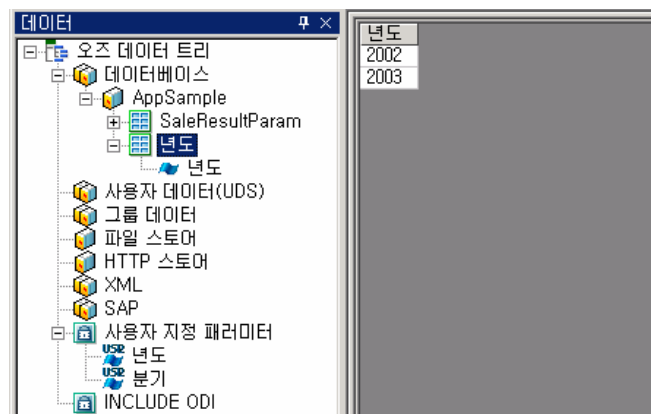
- 6) 쿼리문 실행[



아래 그림과 같이 검색 결과와 'SaleResultParam' 데이터 셋에 부서명, 사원명 등 데이터를 나타내는 필드명이 제대로 표시됨을 확인합니다.



- 7) 검색 조건을 콤보박스에 나타낼 년도와 분기를 추출할 데이터셋을 추가합니다.
- 먼저 '년도' 값을 나타낼 데이터셋을 추가합니다. 데이터창에서 'AppSample'을 마우스 오른쪽 버튼으로 클릭하여 나타나는 팝업 메뉴에서 [데이터 셋 추가] 메뉴를 선택합니다. '데이터 셋 이름'을 '년도'라고 입력하고 쿼리문을 입력(입력 방법은 위 내용 중 4), 5) 번 참조 바람)한 후 쿼리문 실행[

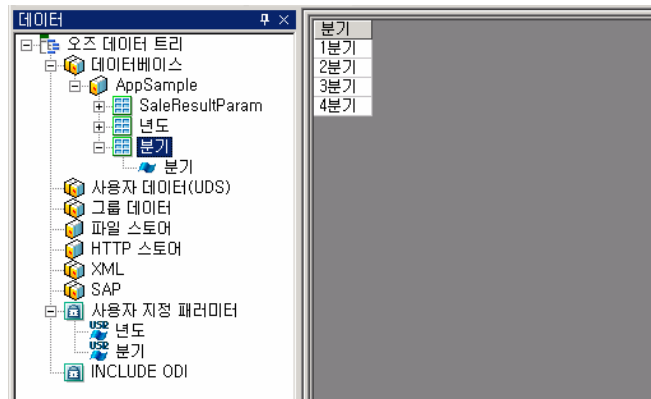


입력한 쿼리 내용은 다음과 같습니다.

```
SELECT Distinct SaleResult.년도 FROM SaleResult
```

다음으로 '분기' 값을 추출합니다. 데이터창에서 'AppSample'을 마우스 오른쪽 버튼으로 클릭하여 나타나는 팝업 메뉴에서 [데이터 셋 추가] 메뉴를 선택합니다. '데이터 셋 이름'을 '분기'라고 입력하고 쿼리문을 입력한 후 쿼리문 실행

[] 아이콘을 클릭하여 결과를 확인합니다.



입력한 쿼리 내용은 다음과 같습니다.

```
SELECT Distinct SaleResult.분기 FROM SaleResult
```

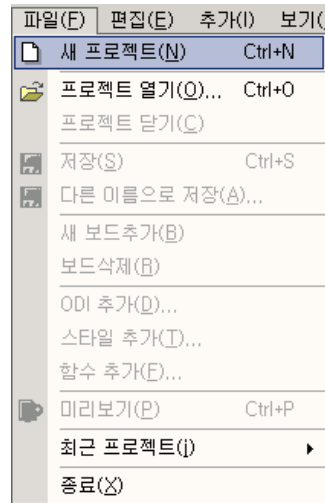
- 8) 작성한 ODI 파일을 저장합니다. [파일] 메뉴에서 [저장]을 선택하여 'AppSample.odi'라는 이름으로 리파지토리 루트 폴더에 저장합니다. 본 예제에서 리파지토리 루트 폴더는 'C:\Repository' 입니다

Step 2

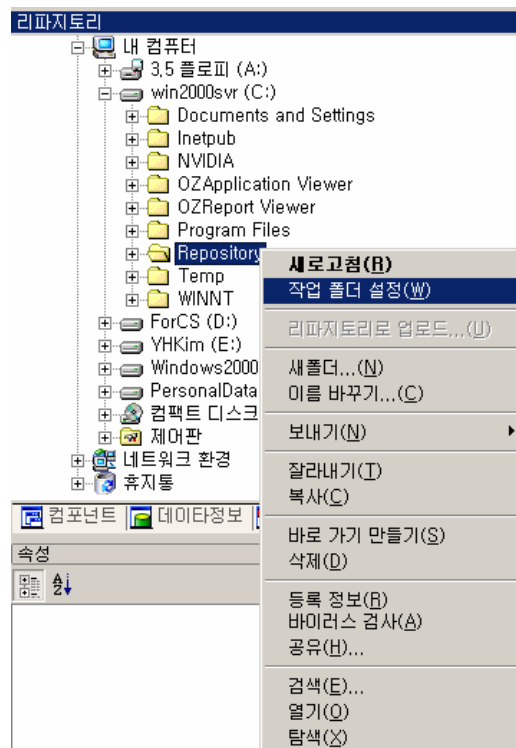
◆ 새 프로젝트 생성

오즈 애플리케이션 디자이너를 실행하여 새로운 프로젝트를 만들고 작업 폴더를 설정합니다.

- 1) [파일] 메뉴의 [새 프로젝트]를 선택하거나 톨바에서 새 프로젝트 아이콘()을 선택하여 새로운 프로젝트를 만듭니다.



- 2) 오즈 애플리케이션 디자이너 좌측 화면 상단에서 [Repository] 탭을 선택하고 작업 폴더로 설정할 폴더를 선택한 후 마우스 오른쪽 버튼을 클릭하여 나타나는 팝업 메뉴에서 [작업 폴더 설정] 메뉴를 클릭하여 해당 폴더를 작업 폴더를 설정합니다. 작업 폴더는 개발 환경마다 다르며 본 예제에서는 C:\Repository를 작업 폴더로 설정하였습니다.

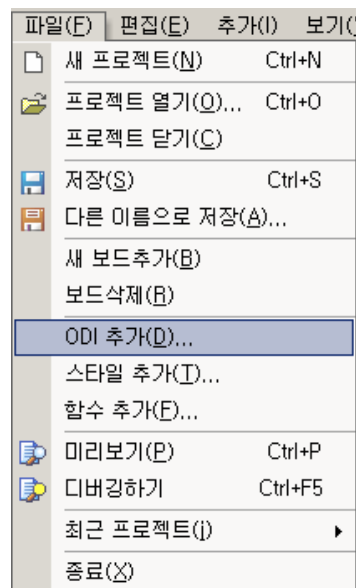


◆ ODI 파일 추가

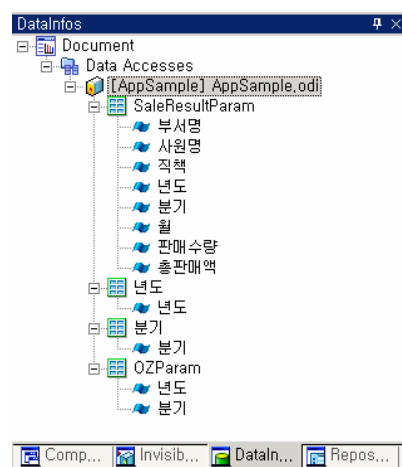
작성한 ODI 파일(AppSample.odi)을 추가합니다.

1) ODI 파일을 추가합니다.

[파일] 메뉴의 [ODI 추가] 메뉴를 선택하여 리파지토리 파일 다이얼로그에서 'AppSample.odi' 파일을 선택한 후 [확인] 버튼을 클릭하여 추가하거나 오즈 애플리케이션 디자이너 좌측 화면 상단에서 [Repository] 탭을 선택하고 작업 폴더에 있는 'AppSample.odi' 파일을 더블 클릭하여 추가합니다.



2) 오즈 애플리케이션 디자이너 좌측 화면 상단에서 [DataInfos] 탭을 선택하여 추가한 ODI 파일과 데이터 셋 내용을 확인합니다.

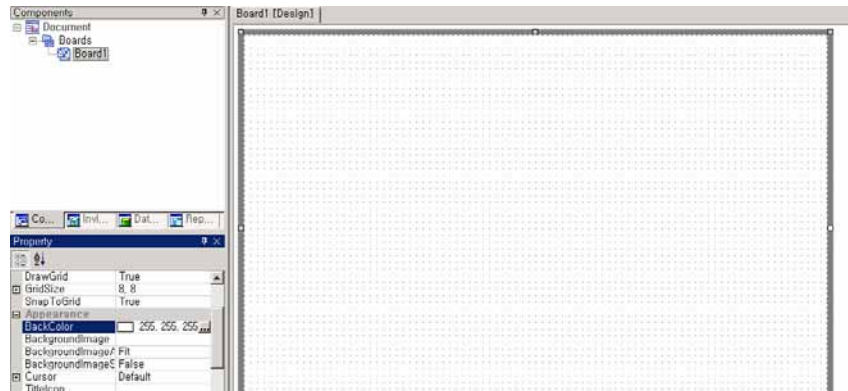


◆ 보드 디자인

보드의 배경색과 타이틀을 디자인합니다.

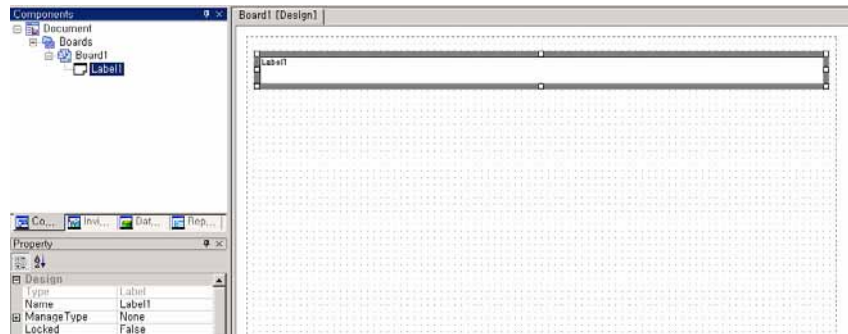
- 1) 보드의 배경색을 흰색으로 변경합니다.

[Component] 탭에서 [Board1]을 선택한 후 'Property'창에서 'BackColor'를 선택하고 색 선택 버튼[...]을 클릭하여 흰색으로 변경합니다.



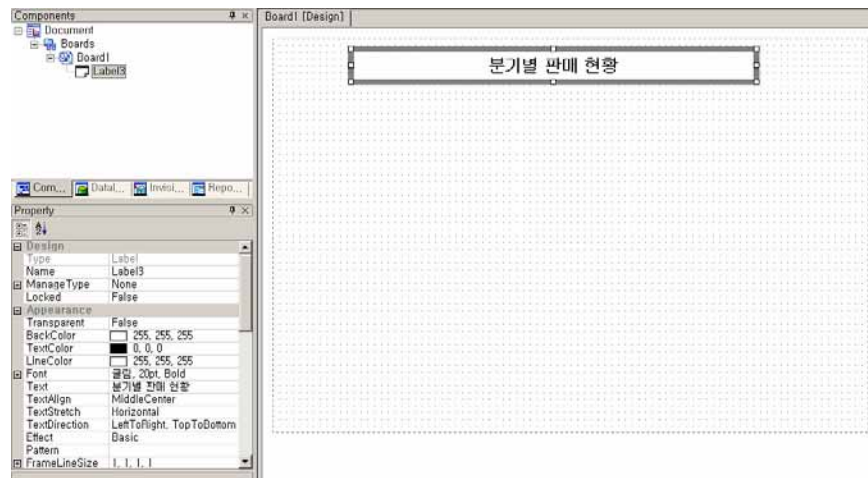
- 2) 타이틀을 표시할 라벨을 추가합니다.

[추가] 메뉴의 [컨트롤]의 [라벨]을 선택하거나 툴바에서 라벨 아이콘[A]을 선택한 후 표시하고자 하는 영역만큼 Drag & Drop하여 라벨을 추가합니다.



- 3) 추가한 라벨의 색, 글꼴, 텍스트 등을 디자인합니다.

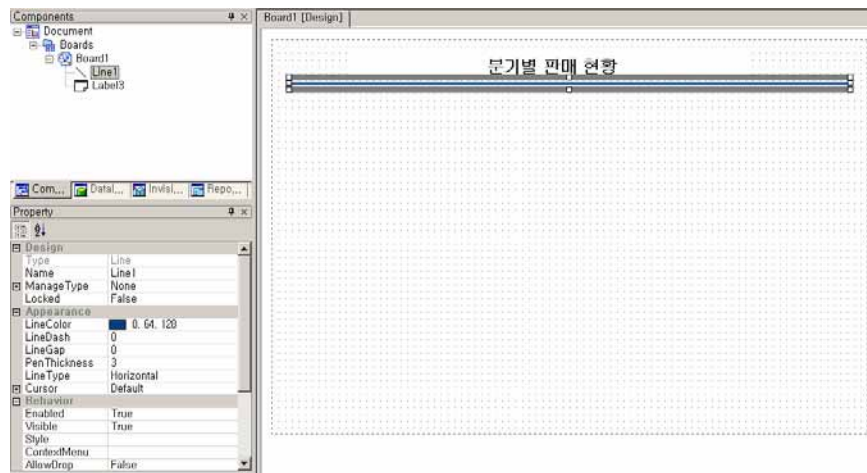
'Label1'을 선택한 후 'Property'창에서 라벨의 속성을 변경하여 디자인합니다. 'BackColor'와 'LineColor'를 흰색으로, 'Font'를 '굴림, 20pt, Bold'로, 'TextAlign'을 'MiddleCenter'로 변경하고, 'Text'를 '분기별 판매 현황'이라고 입력합니다.



- 4) 타이틀 아래에 표시할 선을 추가하고 디자인합니다.

[추가] 메뉴의 [그리기 컨트롤]의 [선]을 선택하거나 툴바에서 선 아이콘[]을 선택한 후 표시하고자 하는 영역만큼 Drag & Drop하여 선을 추가합니다.

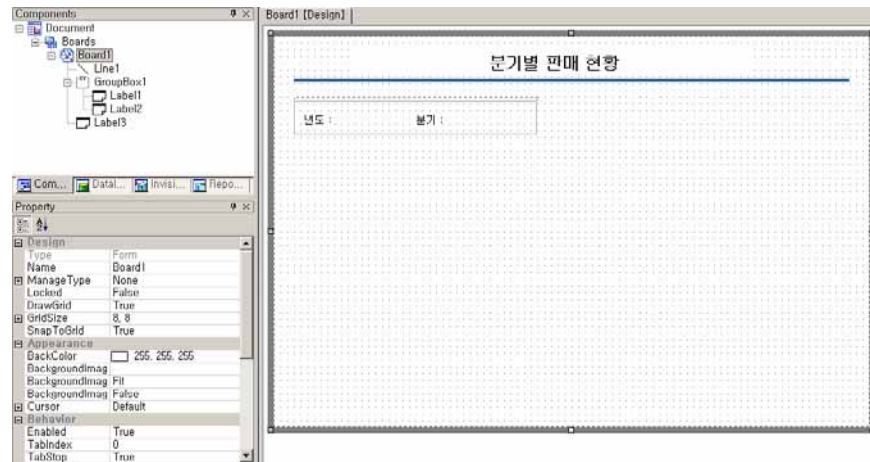
'Line1'을 선택한 후 'Property'창에서 'LineColor'를 선택하여 라인 색을 변경하고 'PenThickness'를 선택하여 라인의 두께를 적절하게 변경합니다.



◆ 검색 조건용 User Interface 디자인

그룹박스, 라벨, 버튼을 추가하고 년도와 분기를 입력 받을 콤보박스를 추가하여 해당 값을 나타내는 데이터 셋을 연결합니다.

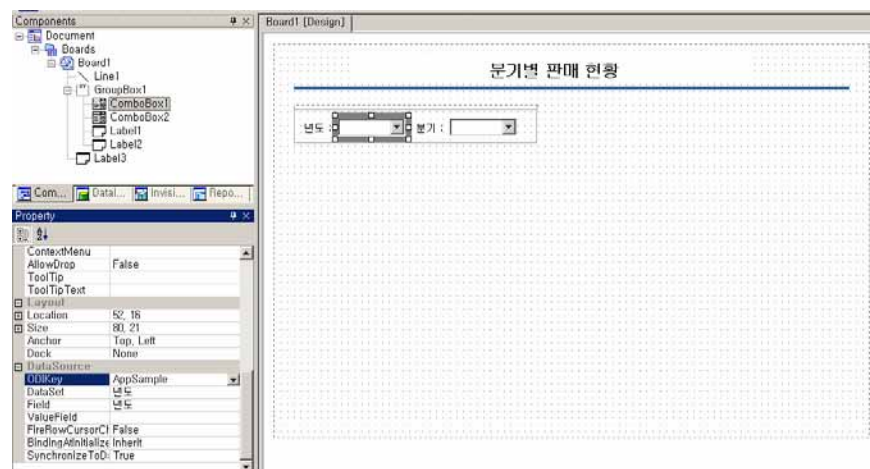
- 1) 그룹박스과 라벨을 추가하고 아래 그림과 같이 디자인합니다.



2) 콤보박스 2개를 보드에 추가합니다.

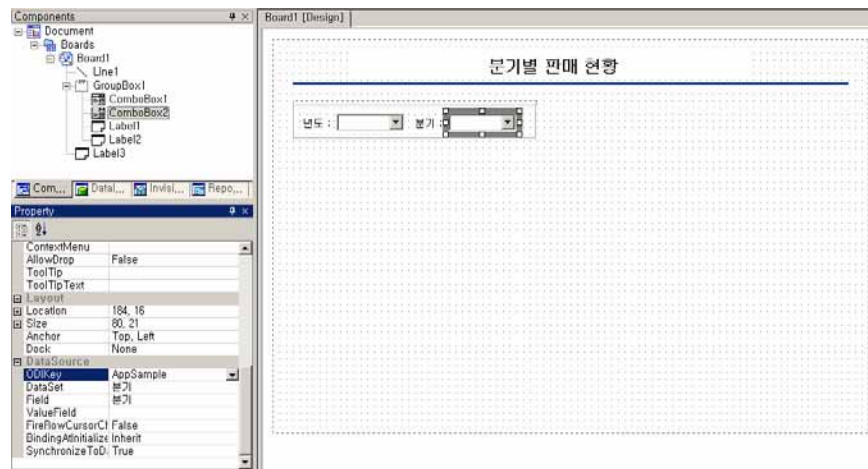
3) 콤보박스에 연동할 ODI와 데이터 셋을 설정합니다.

먼저 년도를 나타낼 콤보박스를 설정합니다. 'ComboBox1'을 선택한 후 'Property창'에서 'ODIKey'를 'AppSample'로, 'DataSet'을 '년도'로, 'Field'를 '년도'로 설정합니다.



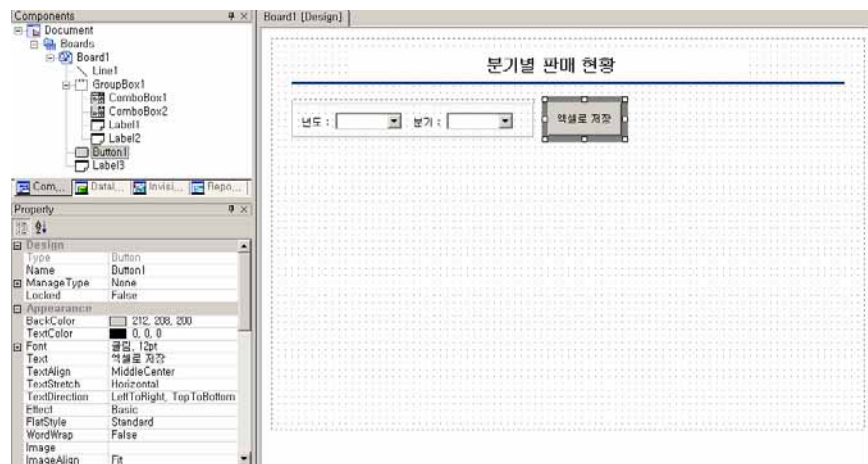
다음으로 분기를 나타낼 콤보박스를 설정합니다.

'ComboBox2'를 선택한 후 'Property창'에서 'ODIKey'를 'AppSample'로, 'DataSet'을 '분기'로, 'Field'를 '분기'로 설정합니다.



4) 테이블의 내용을 엑셀로 저장하기 위한 버튼을 추가하고 디자인합니다.

버튼을 한 개 추가한 후 추가한 버튼의 **Text**를 '엑셀로 저장'이라고 입력하고 크기와 위치를 아래 그림과 같이 디자인합니다.



◆ 테이블 디자인

ODI 파일과 데이터 셋을 테이블에 연결한 후 마법사를 이용하여 표시할 데이터를 설정하고 테이블의 모양을 디자인합니다.

1) 테이블을 보드에 추가합니다.

[추가] 메뉴의 [컴플렉스 컨트롤]의 [테이블]을 선택하거나 툴바에서 테이블 아이콘[]을 선택한 후 표시하고자 하는 영역만큼 **Drag & Drop**하여 테이블을 추가합니다.

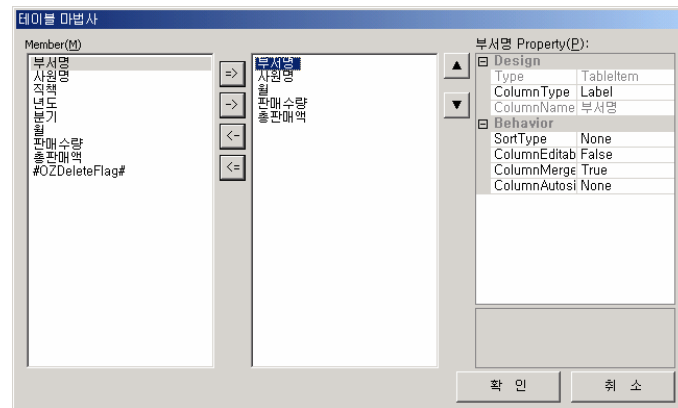
- 2) 테이블에 연동할 ODI와 데이터 셋을 설정합니다.

'Table1'을 선택한 후 'Property창'에서 'ODIKey'는 'AppSample'로, 'DataSet'은 'SaleResultParam'으로 설정합니다.

- 3) 테이블 마법사를 이용하여 표시할 필드명을 추가합니다.

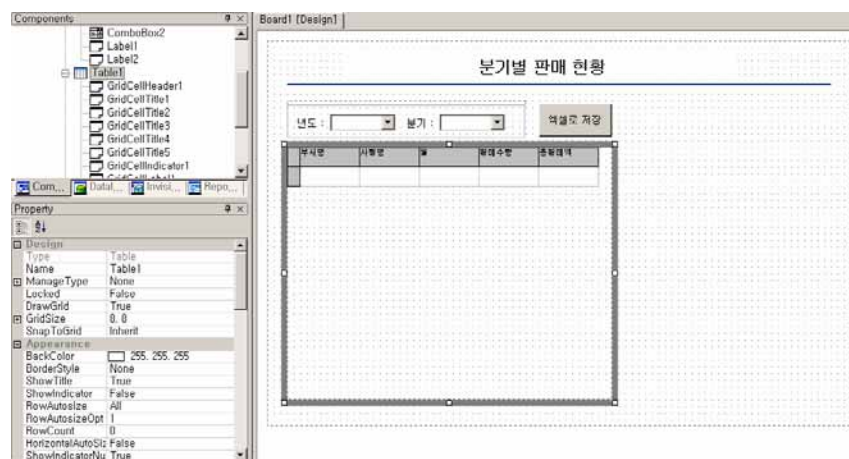
테이블을 선택하고 마우스 오른쪽 버튼을 클릭하여 나타나는 팝업 메뉴에서 [마법사]를 선택합니다.

표시할 필드를 전체/개별 추가 아이콘[=>][>]을 이용하여 추가하거나 잘못 추가된 필드는 전체/개별 삭제 아이콘[<=<]을 이용하여 삭제하여 표시할 필드명을 설정합니다.



필드명 중 '부서명'과 '사원명'은 같은 데이터인 경우 셀을 병합하기 위해 '부서명'과 '사원명'을 선택한 후 화면 오른쪽의 'Property창'에서 'ColumnMerge' 속성값을 'True'로 변경합니다.

테이블 마법사에서 [OK] 버튼을 클릭한 후의 테이블 모양은 다음과 같습니다.

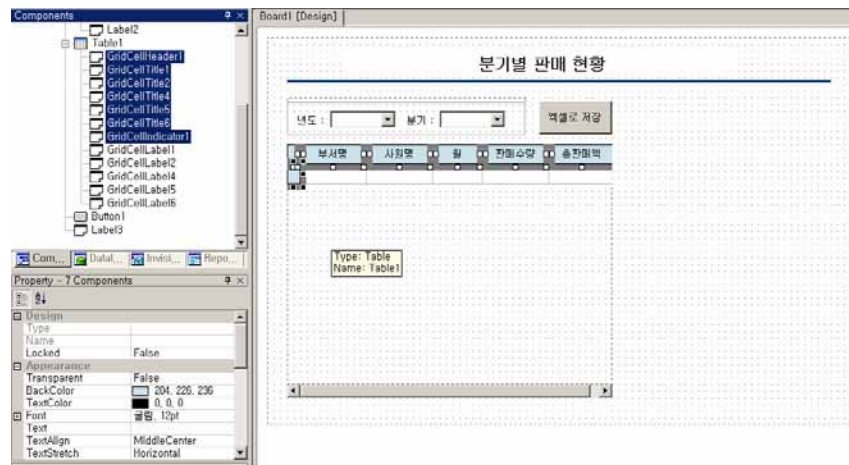


4) 테이블 모양을 디자인합니다.

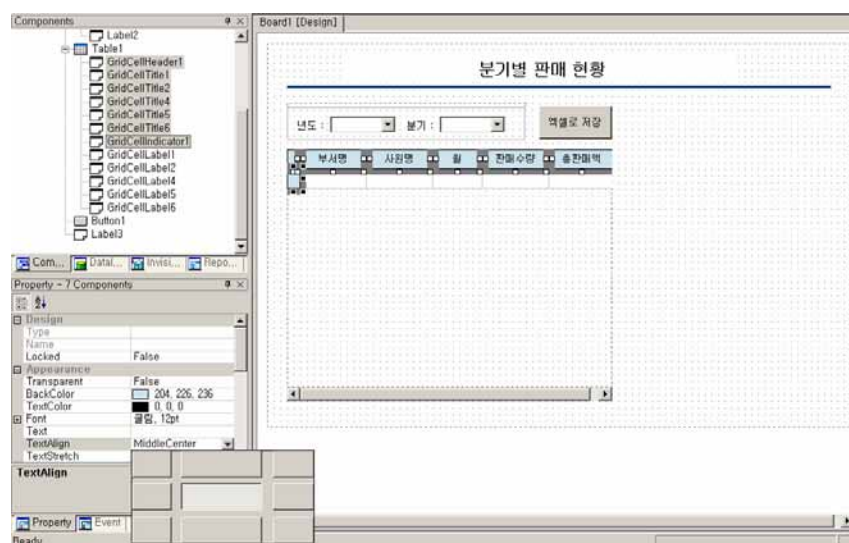
'Table1'을 선택하고 'Property창'에서 'ShowIndicator'값을 'False'로 설정합니다.

다음으로 테이블의 타이틀을 디자인합니다.

테이블 중 'GridCellHeader', 'GridCellTitle', 'GridCellIndicator'를 선택합니다. 선택하는 방법은 두 가지로 'Component창'에서 'Table1' 아래에 있는 'GridCellHeader1'을 클릭하고 키보드의 [Shift] 버튼을 누른 상태에서 'GridCellIndicator1'을 클릭하여 선택하는 방법이 있고, 보드에 디자인된 테이블에서 해당 셀을 클릭하는 방법이 있습니다.



위 그림처럼 선택한 후 'BackColor'는 원하는 색으로(본 예제에서는 '204,226,236'임), 'Font'는 '굴림,12pt'로, 'TextAlign'은 'MiddleCenter'로 설정합니다.

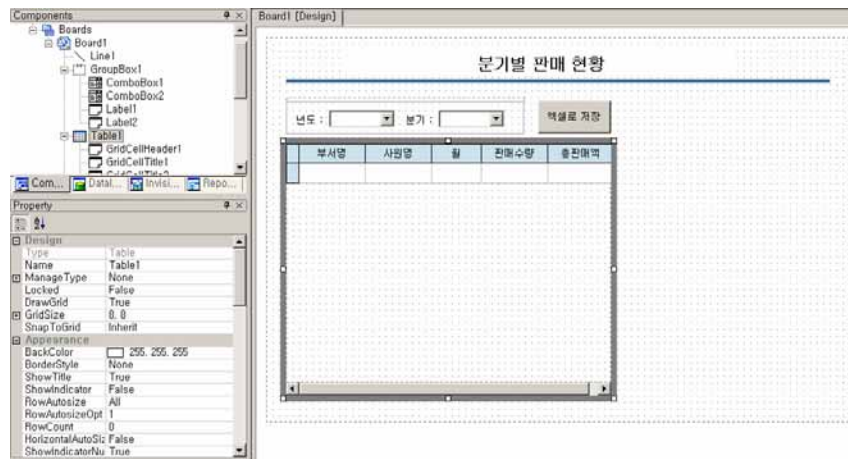


다음으로 데이터가 표시될 라벨을 디자인합니다.

테이블의 'CellLabel' 중 부서명, 사원명, 월에 해당하는 라벨을 선택(선택하는 방법은 타이틀 선택 방법을 참조)한 후 'Font'는 '굴림,12pt'로, 'TextAlign'은 'MiddleCenter'로 설정합니다

테이블의 'CellLabel' 중 숫자 데이터 즉 판매수량, 총판매액에 해당하는 라벨을 선택하여 'Font'는 '굴림,12pt'로, 'TextAlign'은 'MiddleRight'로, 'Pattern'은 패턴 설정 버튼[]을 클릭하여 '숫자형'으로, 'InternalMargin'은 '0,0,10,0'으로 설정합니다.

설정이 완료된 후 화면은 다음과 같습니다.



◆ 차트 디자인

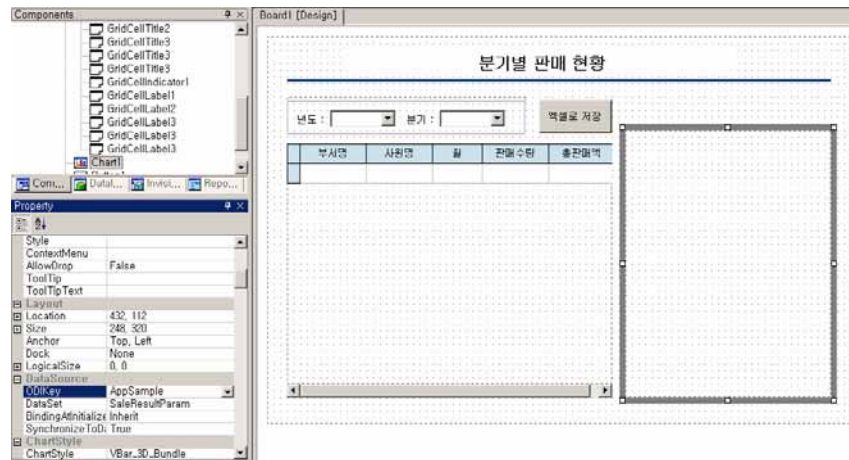
ODI 파일과 데이터 셋을 차트에 연결한 후 마법사를 이용하여 표시할 데이터를 설정하고 차트의 모양을 디자인합니다.

- 1) 차트를 보드에 추가합니다.

[추가] 메뉴의 [컴플렉스 컨트롤]의 [차트]를 선택하거나 툴바에서 차트 아이콘 []을 선택한 후 표시하고자 하는 영역만큼 Drag & Drop하여 차트를 추가합니다

- 2) 차트에 연동할 ODI와 데이터 셋을 설정합니다.

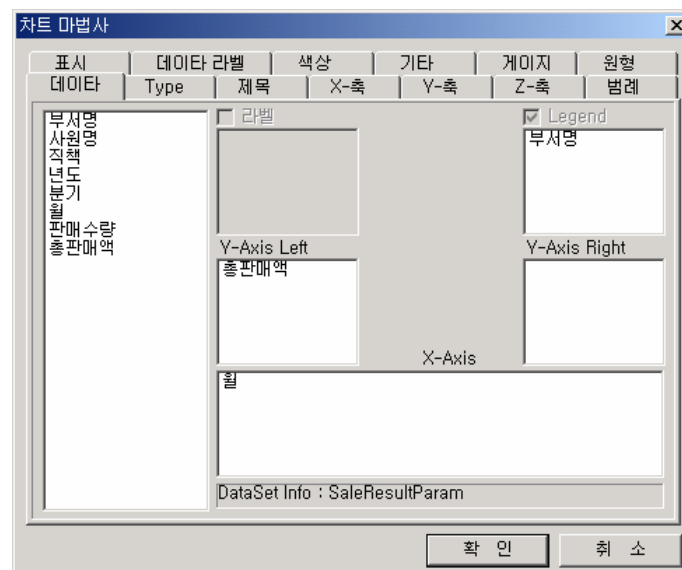
'Chart1'을 선택한 후 'Property창'에서 'ODIKey'는 'AppSample'로 'DataSet'은 'SaleResultParam'으로 설정합니다.



3) 차트 마법사를 이용하여 표시할 데이터를 설정합니다.

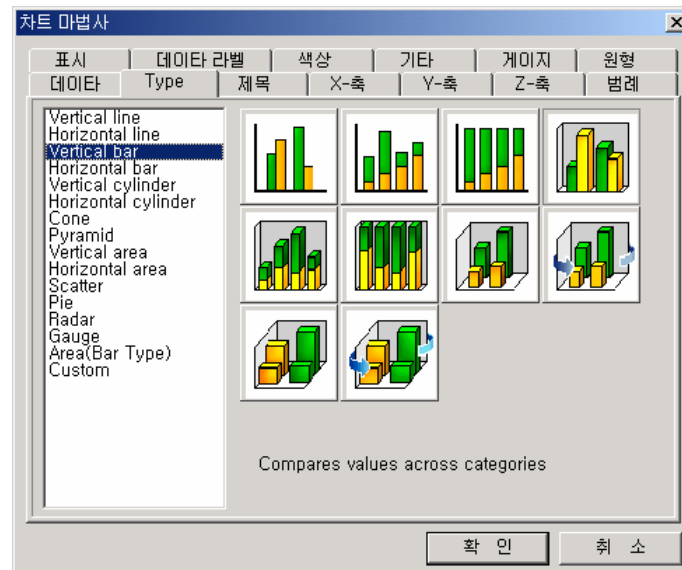
차트를 선택하고 마우스 오른쪽 버튼을 클릭하여 나타나는 팝업 메뉴에서 [마법사]를 선택합니다.

[데이터] 탭을 선택한 후 X축, Y축, 범례에 표시하고자 하는 필드명을 마우스로 Drag & Drop합니다.

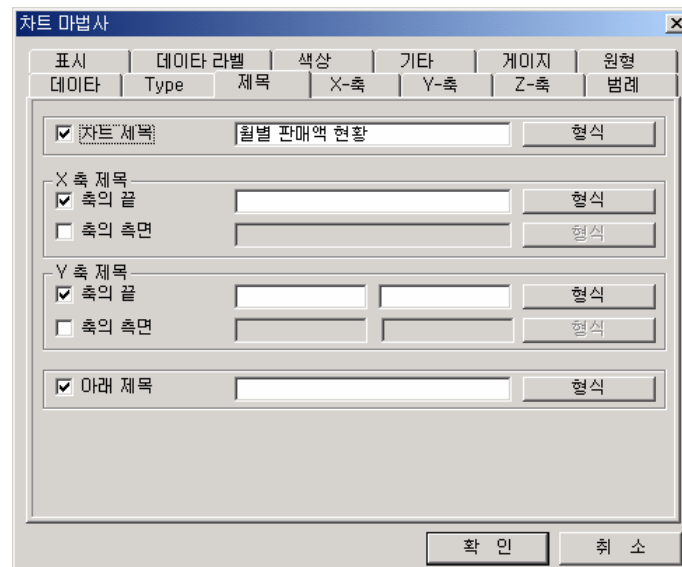


4) 차트 마법사를 이용하여 차트 형태, 제목, 범례, X축의 글꼴, Y축의 최소/최대값 등 차트의 모양을 디자인합니다.

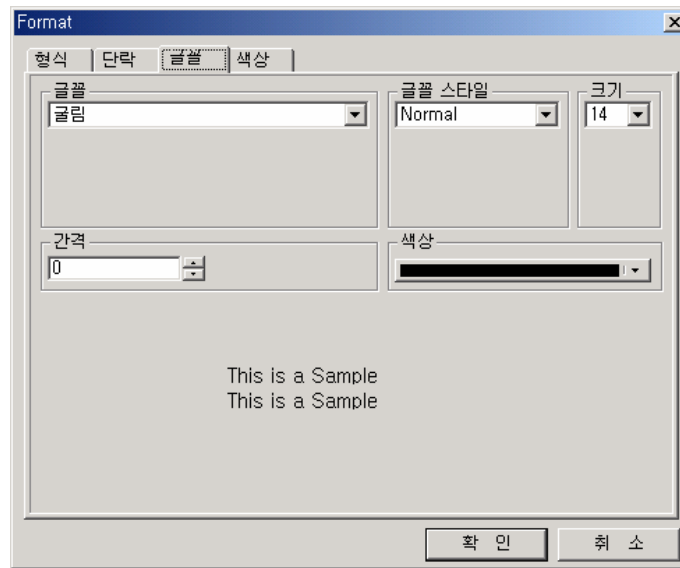
차트 마법사를 실행한 후 [Type] 탭을 선택하여 차트의 형태를 3차원 막대 그래프로 설정합니다.



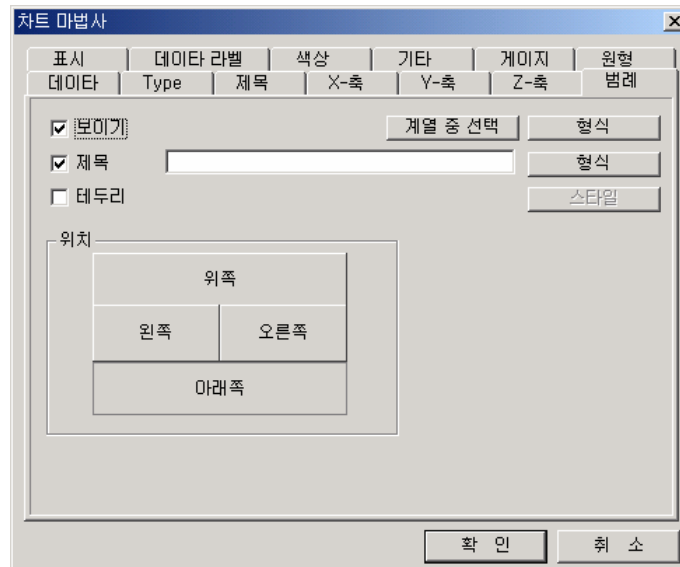
[제목] 탭을 선택하여 차트의 제목을 입력합니다.



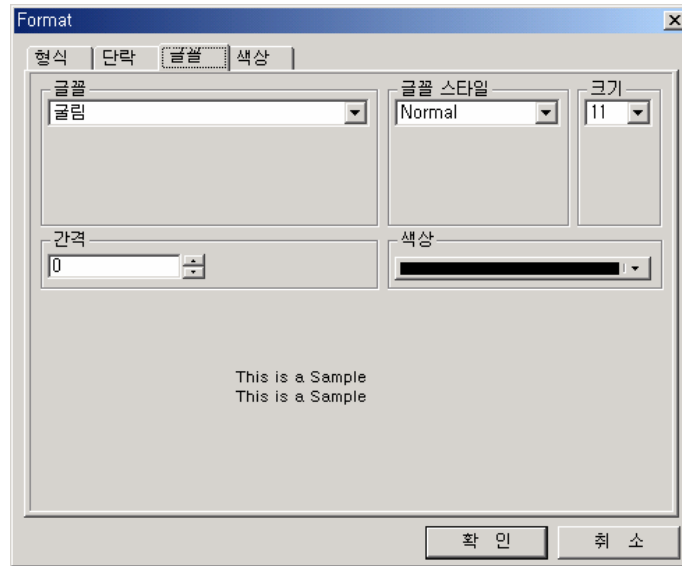
제목을 입력한 후 [형식] 버튼을 클릭하여 제목의 글꼴을 설정합니다.



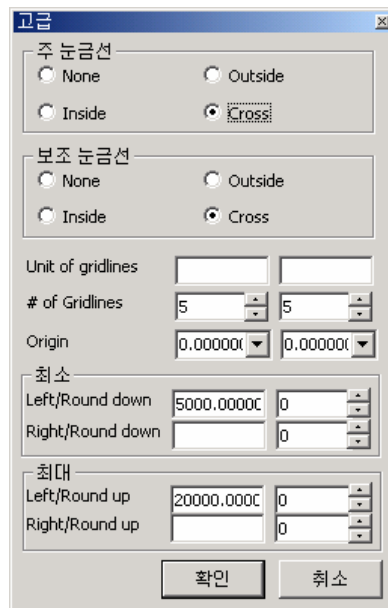
[범례] 탭을 선택한 후 범례의 위치를 '아래쪽'으로 설정합니다.



[X-축] 탭을 선택한 후 [형식] 버튼을 클릭하여 X축의 글꼴을 설정합니다.



[Y-축] 탭을 선택한 후 [고급] 버튼을 클릭하여 Y축의 최소/최대값을 설정합니다.

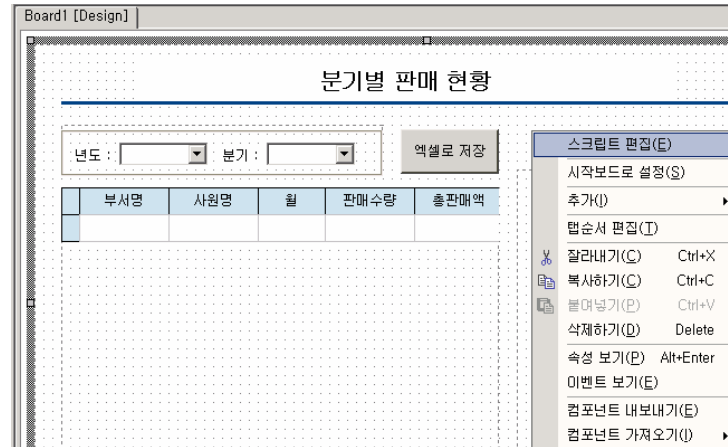


◆ 스크립트 입력

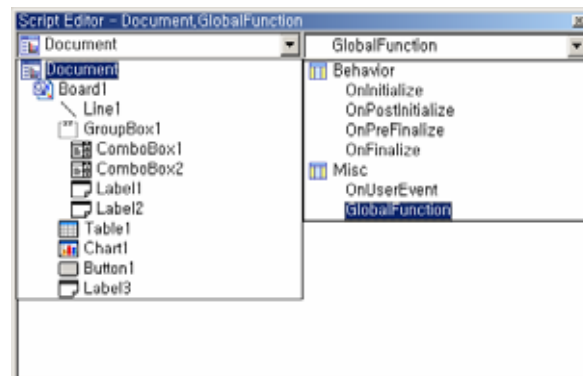
테이블과 차트 디자인을 모두 마친 후에는 검색 조건을 입력받아 데이터를 표시하고 엑셀로 저장하는 스크립트를 입력합니다.

- 1) 데이터를 표시하고 엑셀로 저장하는 스크립트를 전역 함수로 다음과 같이 입력합니다.

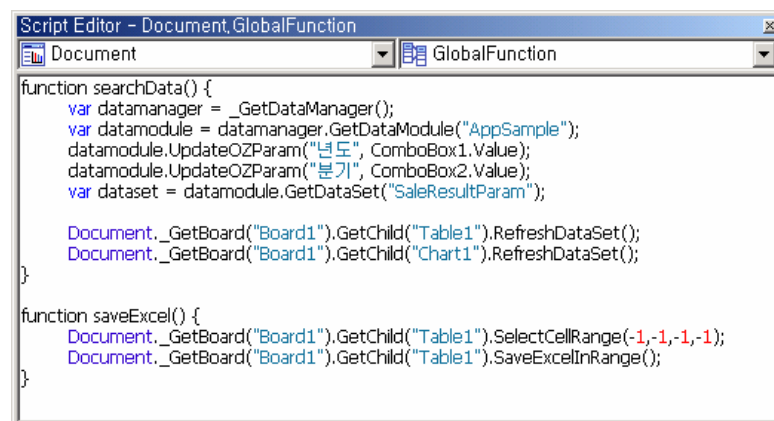
보드에서 오른쪽 버튼을 눌러 나타나는 팝업 메뉴에서 [스크립트 편집] 메뉴를 선택합니다.



스크립트 편집창의 좌측 상단 콤보박스에서 'Document'를 선택하고 우측 상단 콤보박스에서 'Global Function'을 선택합니다.

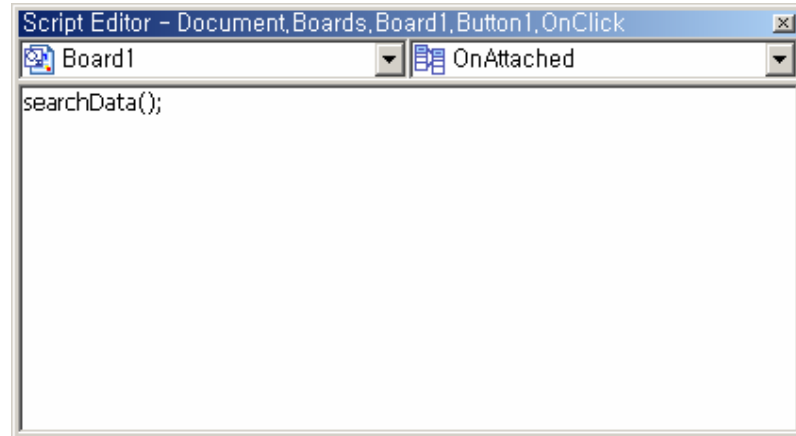


데이터를 표시하는 'searchData()' 함수와 엑셀로 저장하는 'saveExcel()' 함수를 아래와 같이 입력합니다.



- 2) 보드가 화면에 표시될 때 기본적인 값을 표시하도록 입력합니다.

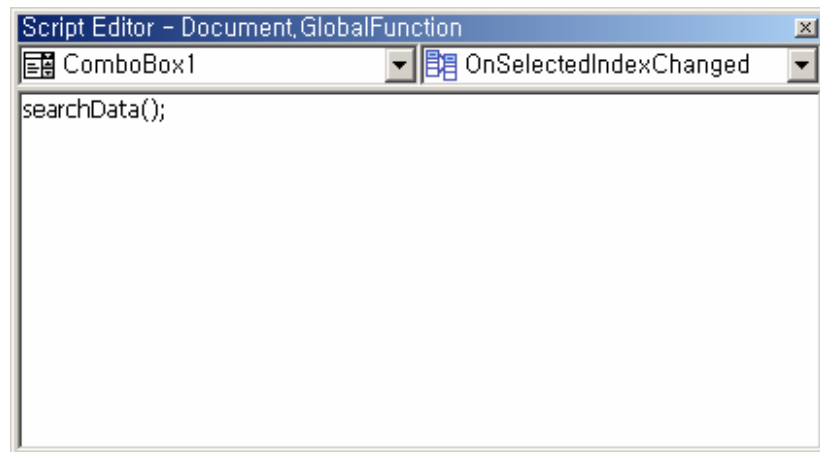
스크립트 편집창의 좌측 상단 콤보박스에서 'Board1'을 선택하고 우측 상단 콤보박스에서 'OnAttached'를 선택한 후 'searchData()' 함수를 호출하도록 아래와 같이 입력합니다.



- 3) 년도와 분기를 선택하는 콤보 박스의 선택값이 변경되었을 경우 데이터를 다시 표시하도록 입력합니다.

먼저 년도를 변경하였을 경우를 입력합니다.

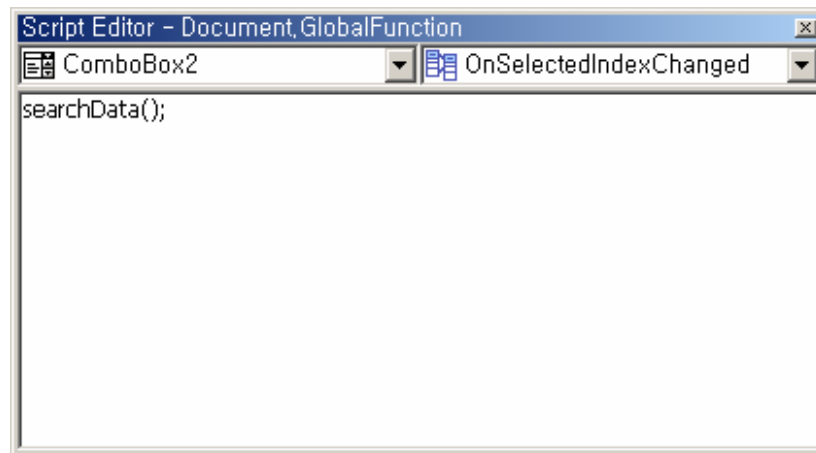
스크립트 편집창의 좌측 상단 콤보박스에서 'ComboBox1'을 선택하고 우측 상단 콤보박스에서 'OnSelectedIndexChanged'를 선택한 후 'searchData()' 함수를 호출하도록 아래와 같이 입력합니다.



다음으로 분기를 변경하였을 경우를 입력합니다.

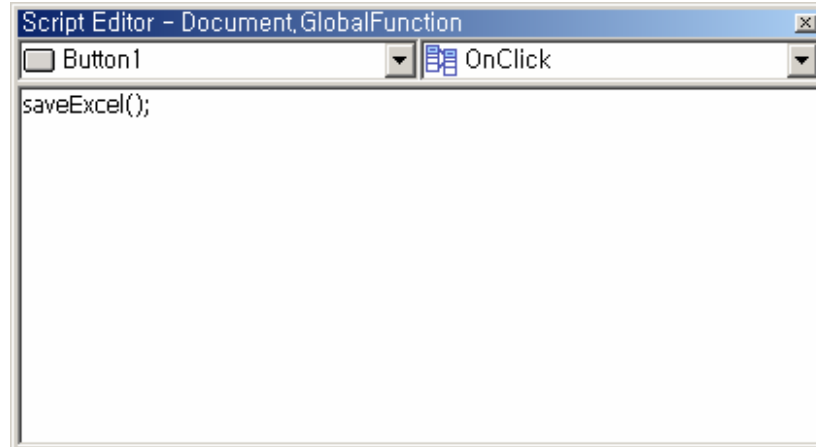
스크립트 편집창의 좌측 상단 콤보박스에서 'ComboBox2'을 선택하고 우측 상단 콤보박스에서 'OnSelectedIndexChanged'를 선택한 후 'searchData()' 함수를 호출하도록 아래와 같이 입력합니다.

수를 호출하도록 아래와 같이 입력합니다.



- 4) 엑셀로 저장' 버튼을 사용자가 클릭할 경우 테이블의 내용을 엑셀 파일로 저장하는 기능을 입력합니다.

스크립트 편집창의 좌측 상단 콤보박스에서 'Button1'을 선택하고 우측 상단 콤보박스에서 'OnClick'를 선택한 후 'saveExcel()' 함수를 호출하도록 아래와 같이 입력합니다.



- 5) 입력한 전체 스크립트 내용을 확인합니다.

[Edit] 메뉴에서 [Preview Script] 메뉴를 선택하여 나타나는 스크립트 미리보기 창에서 확인합니다.



```

SCRIPT_BEGIN Document.GlobalFunction
function searchData() {
    var datamanager = _GetDataManager();
    var datamodule = datamanager.GetDataModule("AppSample");
    datamodule.UpdateOZParam("년도", ComboBox1.Value);
    datamodule.UpdateOZParam("분기", ComboBox2.Value);
    var dataset = datamodule.GetDataSet("SaleResultParam");

    Document._GetBoard("Board1").GetChild("Table1").RefreshDataSet();
    Document._GetBoard("Board1").GetChild("Chart1").RefreshDataSet();
}

function saveExcel() {
    Document._GetBoard("Board1").GetChild("Table1").SelectCellRange(-1,-1,-1,-1);
    Document._GetBoard("Board1").GetChild("Table1").SaveExcelInRange();
}
SCRIPT_END

SCRIPT_BEGIN Document.Boards.Board1.OnAttached
searchData();
SCRIPT_END

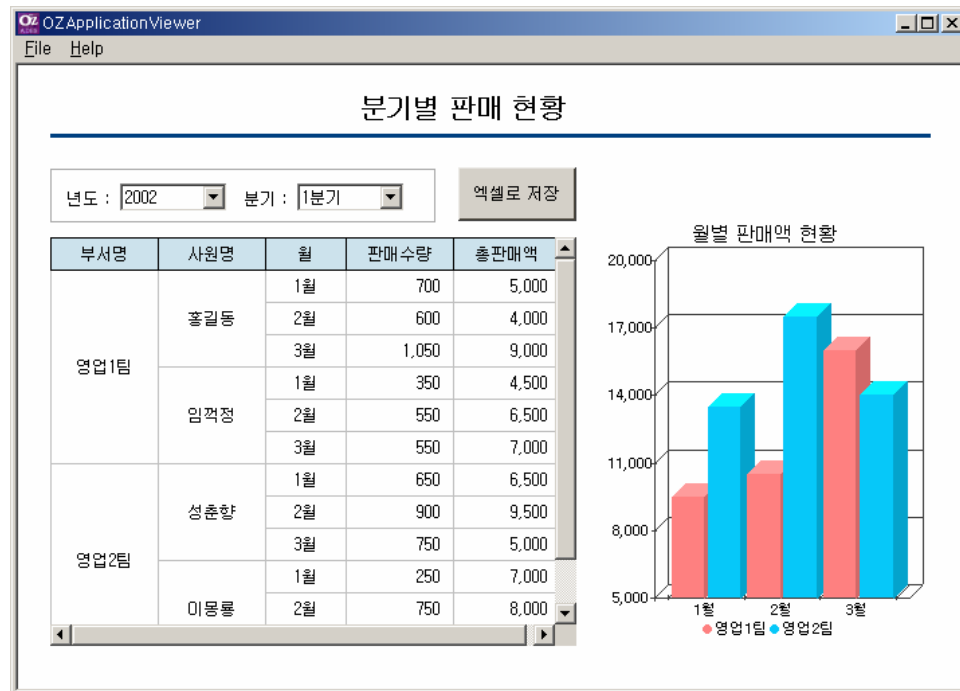
SCRIPT_BEGIN Document.Boards.Board1.ComboBox1.OnSelectedIndexChanged
searchData();
SCRIPT_END

SCRIPT_BEGIN Document.Boards.Board1.ComboBox2.OnSelectedIndexChanged
searchData();
SCRIPT_END

SCRIPT_BEGIN Document.Boards.Board1.Button1.OnClick
saveExcel();
SCRIPT_END
    
```

Step 3

오즈 애플리케이션 디자이너 [File] 메뉴의 [Preview]를 선택하거나 툴바에서 미리보기 아이콘[]을 클릭합니다. 미리보기시에 나타난 화면은 실제 웹을 통해 배포되는 것과 동일합니다.



현재 파일을 'AppSample.oz'로 저장합니다.

Step 1

◆ 데이터베이스 정보 추가

오즈 서버가 설치된 폴더의 `conf` 폴더에 있는 `db.properties` 파일을 문서 편집기로 열어서 사용할 데이터베이스 정보를 아래와 같이 추가합니다.

```
AppSample.vender = odbc
AppSample.dsn = AppSample
AppSample.user =
AppSample.password =
AppSample.maxconns = 5
AppSample.initconns = 2
AppSample.timeout = 5
```

위 내용은 ODBC 설정 예제입니다. Oracle 등의 다른 데이터베이스 정보를 추가하는 방법은 '오즈 엔터프라이즈 서버 관리자 매뉴얼'을 참고하시기 바랍니다.

◆ 오즈 서버 실행

오즈 서버가 설치된 폴더의 `bin` 폴더에 있는 `ozstart.bat` 파일을 윈도우 시작 메뉴의 실행을 통해 실행합니다.

```

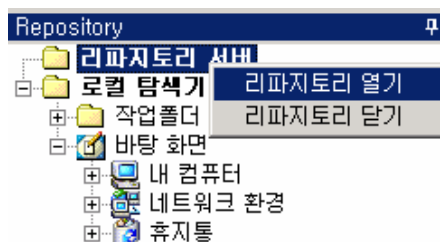
2004-10-01 16:04:50.528 2203 [main] INFO server - OZ DataModule
2004-10-01 16:04:50.528 2203 [main] INFO server - Streaming Protocol :
2040
2004-10-01 16:04:50.528 2203 [main] INFO server -
2004-10-01 16:04:50.528 2203 [main] INFO server -
2004-10-01 16:04:50.528 2203 [main] INFO server - This product includes Hyperso
nic SQL.
2004-10-01 16:04:50.528 2203 [main] INFO server -
2004-10-01 16:04:50.528 2203 [main] INFO server -
2004-10-01 16:04:50.538 2213 [main] INFO server - This product includes softwar
e developed by the
2004-10-01 16:04:50.538 2213 [main] INFO server - Apache Software Foundation (h
ttp://www.apache.org/).
2004-10-01 16:04:50.538 2213 [main] INFO server -
2004-10-01 16:04:50.538 2213 [main] INFO XML-License-Loader - [.xml] loader
2004-10-01 16:04:50.538 2213 [main] INFO XML-Parser-Configurator - try to get D
OM10_SUN parser factory...
2004-10-01 16:04:50.538 2213 [main] INFO XML-Parser-Configurator - Class.forName
e(con.sun.xml.parser.DocumentBuilderFactoryImpl).newInstance()
OZ TCP Server is ready up.
2004-10-01 16:04:50.548 2223 [main] INFO Server - OZ TCP Server is ready up.

```

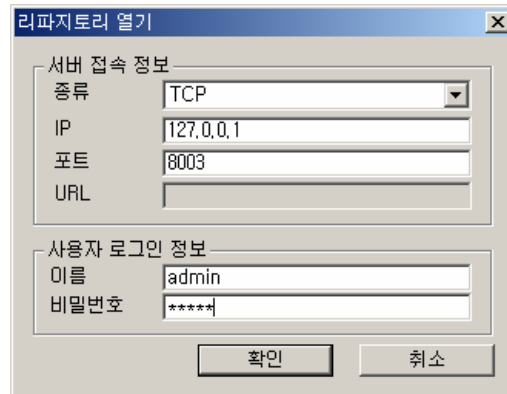
Step 2

오즈 서버가 실행되고 있는 상태에서 애플리케이션 디자이너의 Repository Server에 접속하여 애플리케이션 프로젝트 파일과 ODI 파일을 업로드합니다.

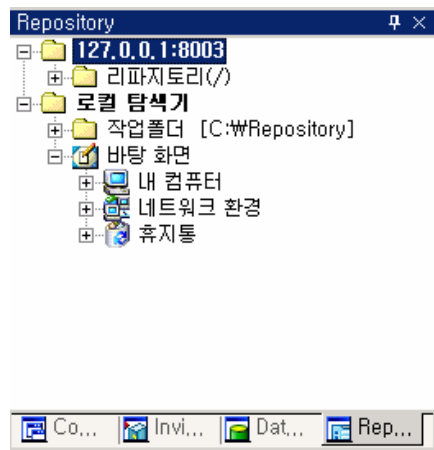
- 1) 오즈 애플리케이션 디자이너의 [Repository] 탭을 선택하고 '리파지토리 서버'에서 마우스 오른쪽 버튼을 눌러 나타나는 팝업 메뉴에서 [리파지토리 열기]를 선택합니다.



- 2) '리파지토리 열기' 다이얼로그에서 아래 그림과 같이 입력하고 [확인] 버튼을 클릭합니다. (비밀번호는 기본값은 'admin' 입니다.)

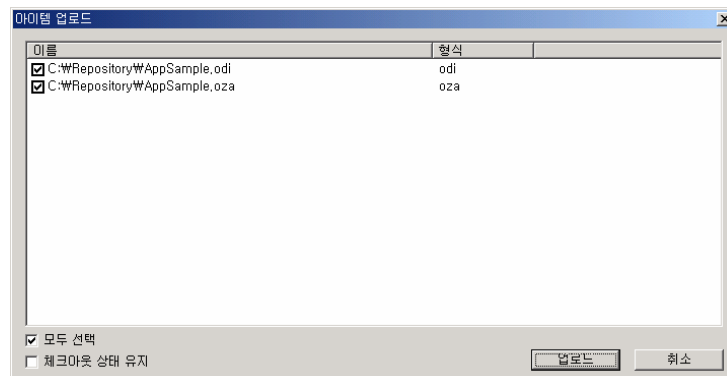


- 3) 아래 그림과 같이 리파지토리 서버에 연결됨을 확인합니다.

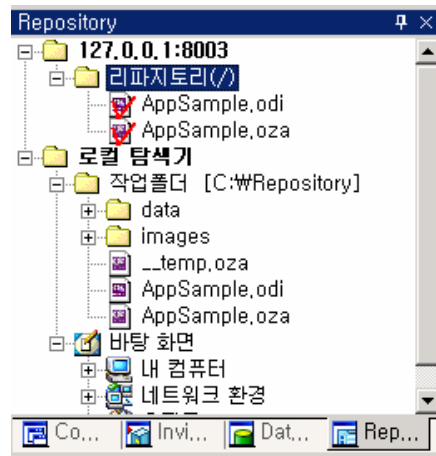


- 4) '작업 폴더'에서 파일을 선택하여 리파지토리 서버에 Drag & Drop하여 서버로 업로드합니다.

본 예제에서는 애플리케이션 프로젝트 파일인 'AppSample.oda'와 ODI 파일인 'AppSample.odi' 파일을 업로드합니다.



5) 아래 그림과 같이 리파지토리 서버에 업로드됨을 확인합니다.



Step 3

◆ 애플리케이션 뷰어 설치 태그 작성

웹페이지 제작용 툴이나 메모장 등을 실행하여 애플리케이션 뷰어 설치 태그를 다음과 같이 작성합니다.

```
<OBJECT id=ZTransferX style="LEFT: 0px; TOP: 0px"
  codeBase=http://127.0.0.1/OZApplicationViewer/ZTransferX.cab#version=2,0,1,2
  height=0 width=0
  classid="CLSID:C7C7225A-9476-47AC-B0B0-FF3B79D55E67" VIEWASTEXT>
  <PARAM NAME="download.Server" VALUE="http://127.0.0.1/OZApplicationViewer">
  <PARAM NAME="download.Port" VALUE="80">
  <PARAM NAME="download.Instruction" VALUE="OZViewer.idf">
  <PARAM NAME="install.Base" VALUE="<PROGRAMS>/Forcs">
  <PARAM NAME="install.Namespace" VALUE="AppViewer">
  <PARAM NAME="useprogress" VALUE="1">
  <PARAM NAME="debug" VALUE="-1">
</OBJECT>
```

※ 주의사항

웹에서 애플리케이션 서비스를 위해서 IIS 웹 서버가 구동되고 있어야 하며,

IIS의 루트 폴더 하위에 다운로드로 설정한 폴더가 있고 그 폴더에 오즈 애플리케이션 뷰어 설치 파일이 존재하여야 합니다. (위 그림처럼 설정한 경우에는 IIS 루트 폴더 → 'OZApplicationViewer' 폴더임)

◆ 애플리케이션 프로젝트 파일 호출 태그 작성

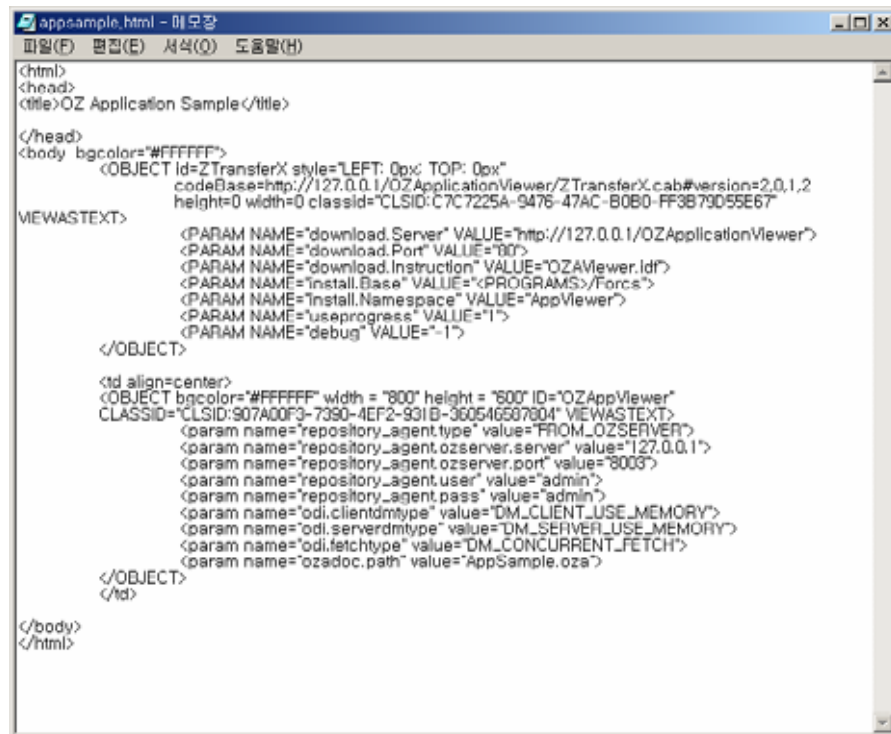
웹페이지 제작용 툴이나 메모장 등을 실행하여 애플리케이션 프로젝트 파일 호출 태그를 다음과 같이 작성합니다.

```
<OBJECT bgcolor="#FFFFFF" width = "800" height = "600" ID="OZAppViewer"
  CLASSID="CLSID:907A00F3-7390-4EF2-931B-360546587804" VIEWASTEXT>
  <param name="repository_agent.type" value="FROM_OZSERVER">
  <param name="repository_agent.ozserver.server" value="127.0.0.1">
  <param name="repository_agent.ozserver.port" value="8003">
  <param name="repository_agent.user" value="admin">
  <param name="repository_agent.pass" value="admin">
  <param name="odi.clientdmttype" value="DM_CLIENT_USE_MEMORY">
  <param name="odi.serverdmttype" value="DM_SERVER_USE_MEMORY">
  <param name="odi.fetchtype" value="DM_CONCURRENT_FETCH">
  <param name="ozadoc.path" value="AppSample.oz">
</OBJECT>
```

'ozadoc.path'에는 애플리케이션 프로젝트 파일의 이름을 설정합니다. 본 예제에서는 'AppSample.oz'입니다.

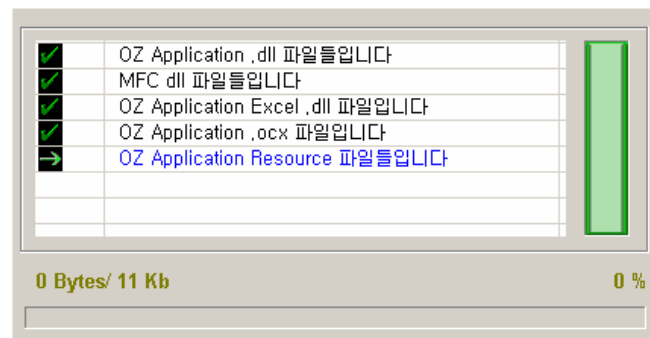
◆ HTML 작성

애플리케이션 뷰어 설치 태그와 프로젝트 파일 호출 태그를 HTML 문서 규격에 맞게 삽입하여 appsample.html 파일로 저장합니다.

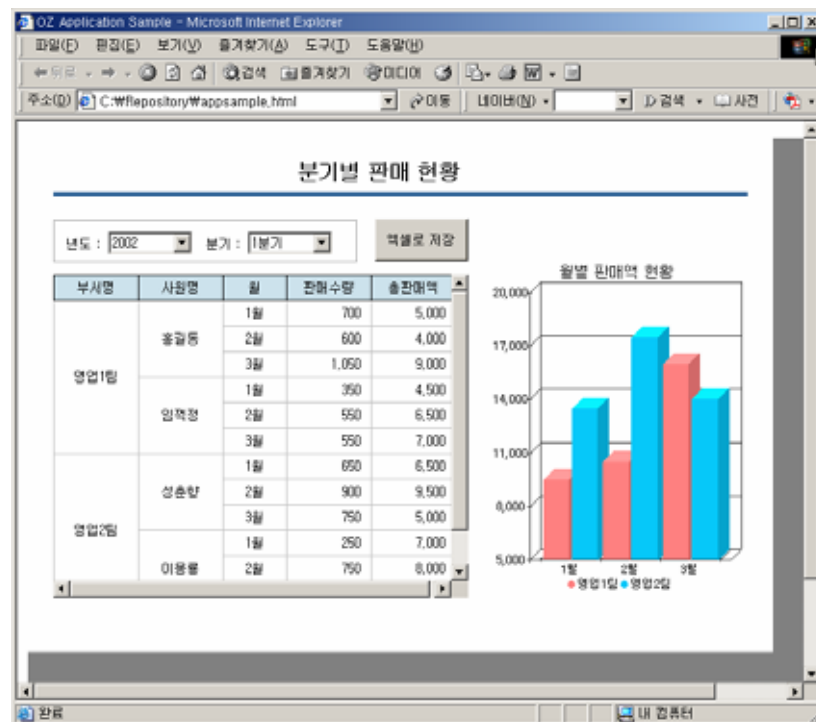
**Step 4**

오즈 애플리케이션 서비스를 위해서는 기본적으로 오즈 서버와 웹 서버가 실행되고 있어야 합니다.

- 1) 오즈 애플리케이션 뷰어 설치 태그와 프로젝트 파일 호출 태그를 삽입한 HTML 파일(appsample.html)을 웹브라우저로 엽니다.
- 2) 오즈 애플리케이션 뷰어가 설치됩니다.
애플리케이션 뷰어는 설치 태그 중 'install.Base'와 'install.Namespace'에서 설정한 폴더에 설치됩니다.



3) 아래 그림과 같이 실행 결과를 확인할 수 있습니다.





본 매뉴얼은 오즈 애플리케이션 디자이너를 처음 접하시는 분들을 위한 기초 학습서로써, 세부적인 기술 내용 보다는 오즈 제품군을 전체적으로 이해하는데 도움을 드리고자 제작되었습니다.

첫째로 사용자가 오즈 애플리케이션 디자이너를 통해 어떠한 기능들을 구현할 수 있는가를 Overview를 통해서 설명하였고, 둘째로 Quick Tour에서는 오즈 애플리케이션 디자이너를 처음 접하는 사용자에게 데이터 연결 설정 방법부터 애플리케이션 개발과 운영까지 예제를 중심으로 설명하였습니다.

오즈 애플리케이션을 포함한 오즈 제품군의 세부적인 기술 정보를 원하신다면 Getting Started 외의 세분화된 매뉴얼을 참조하시기 바랍니다.

오즈의 자세한 기술사항 문의

주식회사 포시에스 <http://www.forcs.com>

서울특별시 동작구 대방동 49-6 유한양행빌딩 12층 (156-754)

전화 : 82-2-828-1400

e-mail : oz@forcs.com