# 목 차

Ⅰ. 시작하기 전에 ..... 3
본 매뉴얼의 목적 ..... 4
본 매뉴얼의 구성 ..... 4
사전 지식 ..... 5
대상 고객 및 활용 분야 ..... 5

Ⅱ. Features Overview ..... 7
제품 구성 ..... 8
Features ..... 11
운영 흐름도 ..... 15
시스템 요구사항 ..... 16

Ⅲ. Quick Tour ..... 19
개발 환경 ..... 20
애플리케이션 개발 절차 ..... 21
Step 1. ODI 파일 만들기 ..... 21
Step 2. 보드 디자인과 스크립트 입력 ..... 29
Step 3. 완성된 프로젝트 미리보기 ..... 45

애플리케이션 운영 절차 ..... 47
Step 1. 오즈 서버 구동 ..... 47
Step 2. 오즈 서버로 파일 업로드 ..... 48
Step 3. 태그 작성 ..... 50
Step 4. 실행 ..... 51

Ⅳ. 마치면서 ..... 53

# Ⅰ. 시작하기 전에

- 본 매뉴얼의 목적
- 본 매뉴얼의 구성
- 사전 지식
- 대상 고객 및 활용 분야

## 본 매뉴얼의 목적

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

## 본 매뉴얼의 구성

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

### Features Overview

오즈 애플리케이션의 특징과 기능, 아키텍처, 운영 방식 등을 개괄적으로 설명합니다.

### Quick Tour

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

## 사전 지식

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)

## 대상 고객 및 활용 분야

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

본 제품은 웹을 기반으로 하는 모든 애플리케이션 개발 업무에 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

# Ⅱ. Features Overview

- 제품 구성
- Features
- 운영 흐름도
- 시스템 요구사항

# 제품 구성

< 오즈 애플리케이션 제품 구성>

오즈 애플리케이션은 오즈 애플리케이션 디자이너, 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.

## 오즈 애플리케이션 디자이너

- C/S 환경의 강력한 사용자 인터페이스를 웹에서 그대로 구현할 수 있도록 웹 애플리케이션 개발의 편이성과 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.

## 오즈 쿼리 디자이너

- 애플리케이션 디자인에 필요한 데이터 셋(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 디자인할 수 있으며 디자인한 결과를 즉시 실행해 볼 수 있어 정확한 쿼리문을 생성할 수 있도록 도와 줍니다.

## 오즈 엔터프라이즈 매니저

- 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.

## 오즈 리파지토리 매니저

- 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.

## 오즈 엔터프라이즈 서버

- 오즈 엔터프라이즈 서버는 오즈 뷰어의 요청을 받아 결과 데이터 셋 생성 및 폼과 데이터 셋 전송을 수행하는 고성능 다중 스레드 서버 엔진입니다.
- 오즈 엔터프라이즈 서버는 순수 자바 기반의 자바 서버와 .NET Framework 기반의 닷넷 서버 두 가지 형태로 제공하며, JRE를 지원하는 환경뿐만 아니라 .NET Framework 환경 등 다양한 서버 실행 환경에 적용할 수 있습니다.
- 리파지토리를 이용하여 사용자/사용자 그룹별 권한 관리 및 보고서 파일의 버전 관리 기능이 지원되어 효율적인 보고서 개발 및 관리가 가능합니다.
- DB Pooling, 다중 스레드, 스레드 재사용, 보고서별 캐시 설정, 디스크/메모리 혼용 캐시 기능 등 다양한 성능 향상 기법을 도입하여 다수 사용자의 동시 접속 환경에서도 최적의 성능을 보장합니다.
- 데이터의 처리 속도 향상을 위해 실시간으로 뷰어로 전송할 수 있는 동시 전송 모드 (Concurrent Mode)를 지원하며, 대용량 데이터도 실시간 처리가 가능합니다.

## 오즈 애플리케이션 뷰어

- 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 입력 값을 처리하고 데이터를 조회, 입력, 수정하는 등을 수행하는 클라이언트 모듈입니다.
- Full ActiveX 방식으로 모든 컴포넌트를 한 화면에 표현할 수 있으며, 자동 업데이트 기능을 제공합니다.

# Features

## 쉽고 간편한 웹 애플리케이션 개발 환경

■ **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.

### 일반 컴포넌트

- 사용자 입력 컴포넌트

  텍스트나 값을 입력하거나 선택하는 컴포넌트(TextBox, RichTextBox, MaskTextBox, ComboBox), 숫자 값을 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.

## 데이터 연동 컴포넌트

- 2D/3D Chart 컴포넌트

  꺾은선, 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 연동 컴포넌트

- 리포트 뷰어 컴포넌트

  오즈 리포트 뷰어를 호출하여 오즈 리포트 디자이너로 작성한 보고서 파일을 오즈 애플리케이션 디자이너에서 미리보기할 수 있습니다.
- 리포트 인쇄 및 Export 컴포넌트

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

## 기타 컴포넌트

- 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 있

는 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, JScript를 지원하여 복잡한 애플리케이션을 쉽게 구현할 수 있습니다.

## 탁월한 연동성, 대용량 데이터 처리

■ **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 DBMS에서 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.

## 강력한 모듈화 및 공유 기능

■ **OZC(OZComponent)**

오즈 애플리케이션은 ComboBox, ListBox, Chart 등 서로 다른 여러 종류의 컴포넌트를 하나의 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.

## 고성능 아키텍처, 안정적인 서비스, 강력한 보고서 서비스

- **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**

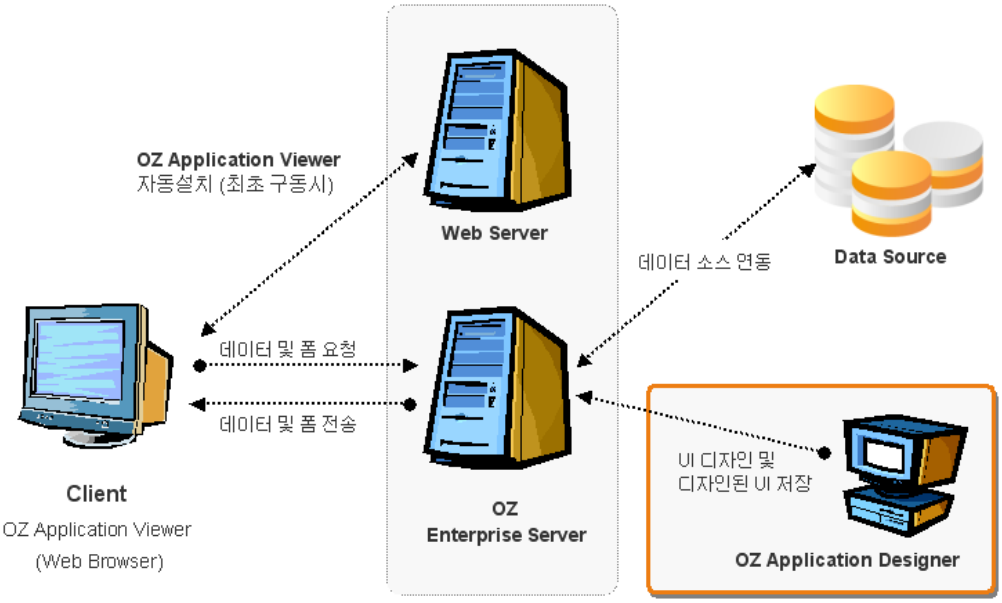
  오즈 애플리케이션은 클라이언트에서 작업을 수행하므로 네트워크 및 서버 부하를 감소시키며, 애플리케이션 프로젝트 파일의 압축 전송 기능으로 전송 효율을 극대화합니다. 또한 네트워크 전송 데이터의 위변조를 방지하여 안정적인 서비스를 제공합니다.

- **오즈 리포트와 연계된 고품질의 보고서 서비스**

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

# 운영 흐름도

오즈 애플리케이션의 개발 및 운영 흐름도는 아래와 같습니다.




**개발 과정**은 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 과정**은 최종 사용자가 웹 애플리케이션을 통해 애플리케이션 프로젝트 파일을 요청하여 애플리케이션 프로젝트 파일이 생성 및 전달되는 과정을 의미합니다.

# 시스템 요구사항

## 오즈 애플리케이션 디자이너/오즈 쿼리 디자이너

- **운영체제**

  Windows Vista 이상

  JRE 1.5 이상, .Net Framework 2.0 이상

- **권장 하드웨어**

  CPU : Dual Core 2.80GHz 이상

  RAM : 2GB 이상

  모니터 : 1920 x 1080 이상의 해상도

## 오즈 엔터프라이즈 매니저/오즈 리파지토리 매니저

- **운영체제**

  Windows Vista 이상

- **권장 하드웨어 사양**

  CPU : Dual Core 2.80GHz 이상

  RAM : 2GB 이상

  모니터 : 1920 x 1080 이상의 해상도

## 오즈 애플리케이션 뷰어

- **운영체제, 웹 브라우저**

  Windows XP 이상에 설치된 ActiveX Control을 지원하는 Internet Explorer 8.0 이상

## 오즈 엔터프라이즈 서버

■ **운영체제**

자바 서버 : JRE 1.5 이상

닷넷 서버 : .NET Framework 2.0 이상

■ **웹 서버**

HTTP를 지원하는 웹 서버

# Ⅲ. Quick Tour

- 개발 환경
- 애플리케이션 개발 절차
  - Step1. ODI 파일 만들기
  - Step2. 보드 디자인과 스크립트 입력
  - Step3. 완성된 프로젝트 미리보기
- 애플리케이션 운영 절차
  - Step1. 오즈 서버 구동
  - Step2. 오즈 서버로 파일 업로드
  - Step3. 태그 작성
  - Step4. 실행

# 개발 환경

- 웹 서버로 인터넷 정보 서비스(IIS)가 설치되어 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 운영하기 위하여 IIS 웹 서버가 구동되고 있어야 하며, 뷰어 설치 태그에서 뷰어 다운로드로 지정된 폴더에 오즈 애플리케이션 뷰어 설치 파일이 존재하여야 합니다.

## 프로젝트 개발에서 운영까지의 절차

### 개발 절차

**Step 1** ODI 파일 만들기

- ODBC 데이터 원본 설정
- 데이터베이스 정보 추가
- 데이터 셋 추가와 쿼리 생성

**Step 2** 보드 디자인과 스크립트 입력

- 새 프로젝트 생성
- ODI 파일 추가
- 보드 디자인
- 검색 조건용 UI 디자인
- 차트 디자인
- 테이블 디자인
- 스크립트 입력

**Step 3** 완성된 프로젝트 미리보기

### 운영 절차

**Step 1** 오즈 서버 구동

- 데이터베이스 정보 추가
- 오즈 서버 실행

**Step 2** 오즈 서버로 파일 업로드

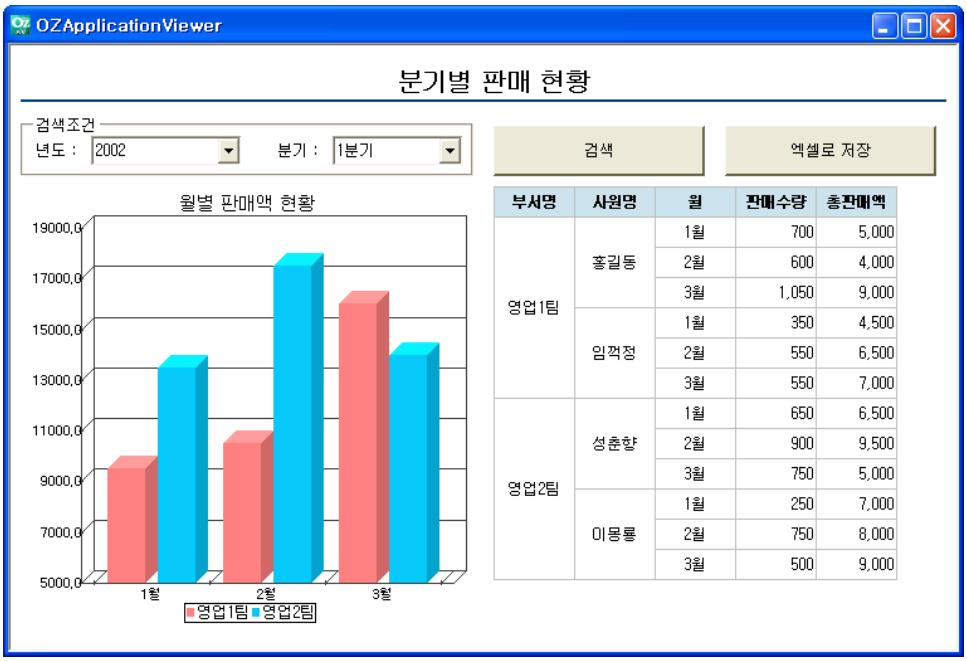
**Step 3** 태그 작성

**Step 4** 실행

# 애플리케이션 개발 절차

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




년도와 분기를 검색 조건으로 하여 분기별 판매 현황을 검색하고 검색된 내용을 차트와 테이블을 이용하여 나타냅니다.

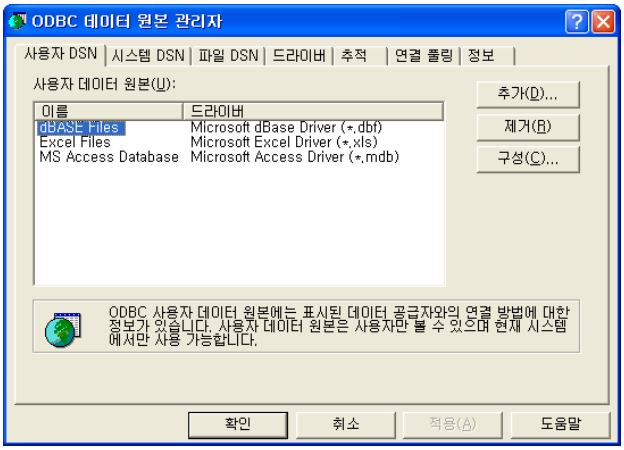
## Step 1 ODI 파일 만들기

◆ **ODBC 데이터 원본 설정**

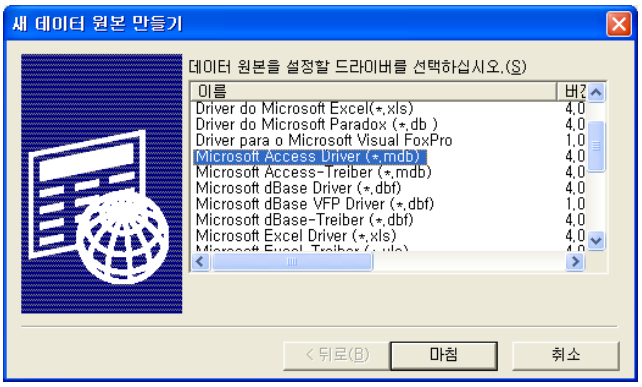
샘플 DB 파일을 ODBC 관리자를 통해 ODBC 데이터 소스로 등록합니다. 이용할 Microsoft Access DB 파일은 AppSample.mdb이며, DB 파일은 고객 지원 센터

로 문의하시기 바랍니다. AppSample.mdb를 ODBC 시스템 데이터 원본으로 등록하고 데이터 원본 이름(DSN)은 'AppSample'로 설정합니다.

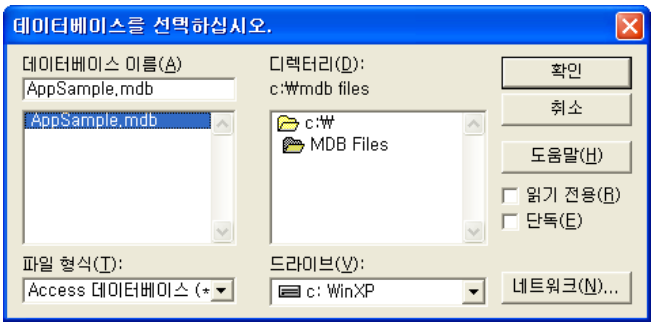
1) [제어판]의 [관리 도구]에서 [데이터 원본(ODBC)]을 선택하여 ODBC 데이터 원본 관리자를 실행합니다.



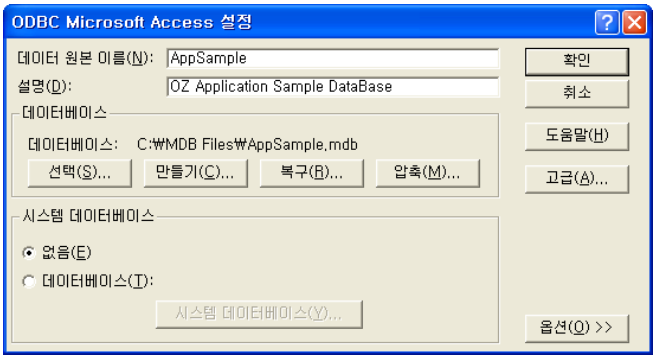

2) [시스템 DSN] 탭을 클릭하고 [추가] 버튼을 클릭한 후 Microsoft Access Driver(*.mdb)를 선택하고 [마침] 버튼을 클릭합니다.



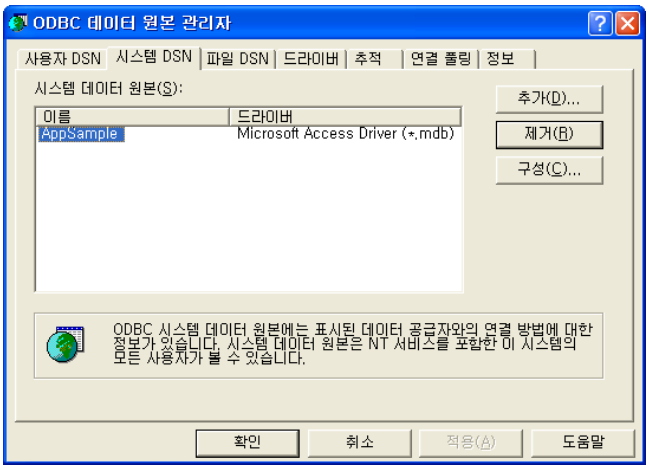

3) 데이터 원본 이름을 'AppSample'로 입력하고, [선택] 버튼을 클릭하여 'AppSample.mdb' 파일을 선택합니다.




4) 설명 등을 추가한 후에 [확인] 버튼을 클릭합니다.



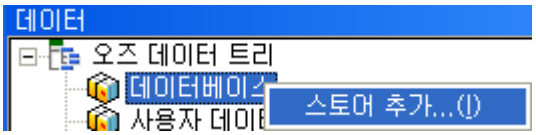

5) 'AppSample'라는 데이터 원본이 추가된 것을 확인합니다.



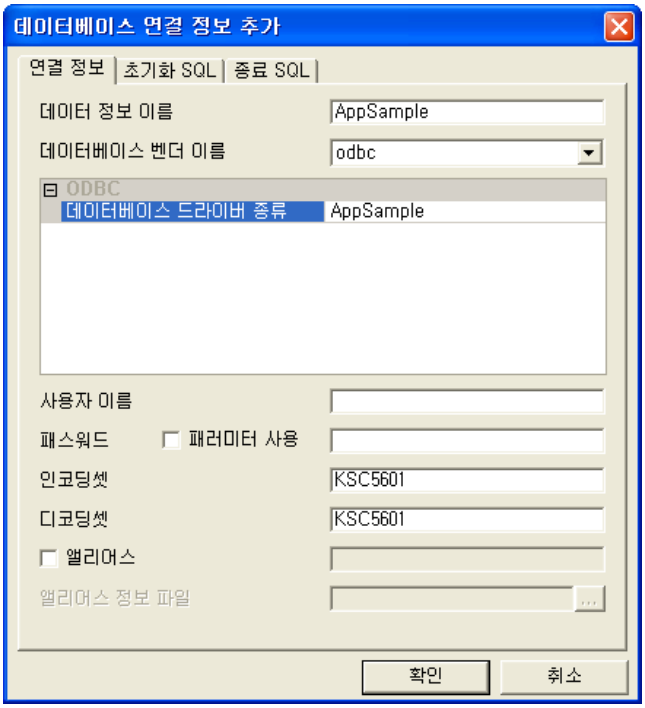

◆ 데이터베이스 정보 추가

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

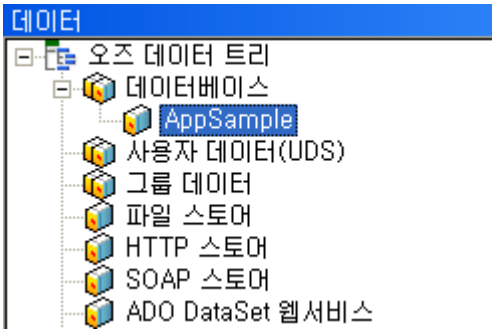
1) 오즈 쿼리 디자이너의 데이터 창의 [데이터베이스] 항목에서 마우스 오른쪽 버튼을 클릭하여 나타나는 팝업 메뉴에서 [스토어 추가]를 클릭합니다.




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.



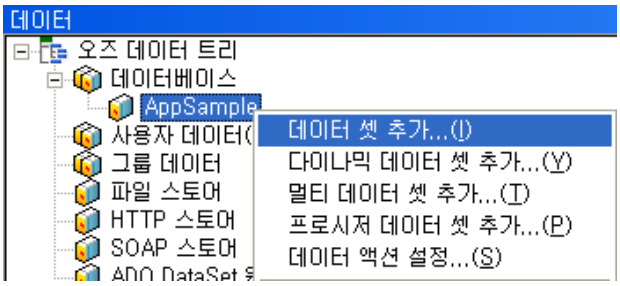

3) 추가된 데이터베이스 정보를 데이터 창에서 확인합니다.



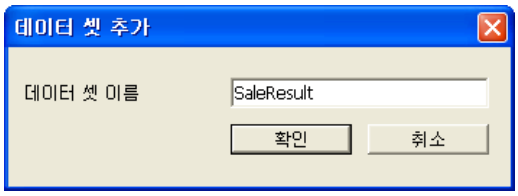
◆ 데이터 셋 추가와 쿼리 생성

데이터 셋을 추가하고 표시할 데이터를 가져올 쿼리를 생성합니다.

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

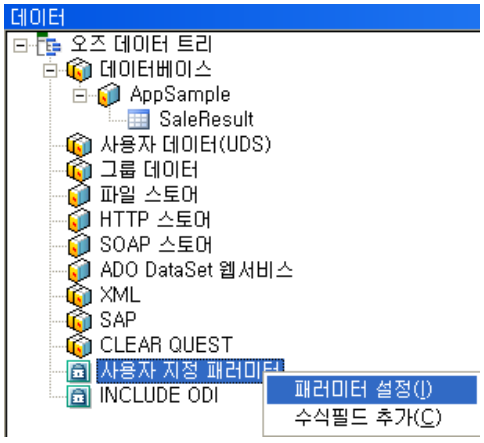



2) [데이터 셋 이름]에 'SaleResult'를 입력한 후 [확인] 버튼을 클릭하여 데이터 셋을 추가합니다.

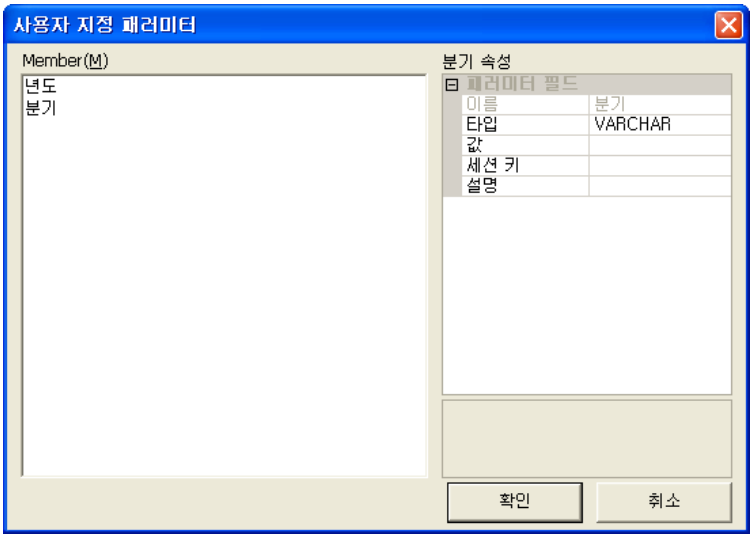



3) 검색 조건을 입력받는 패러미터를 추가합니다.

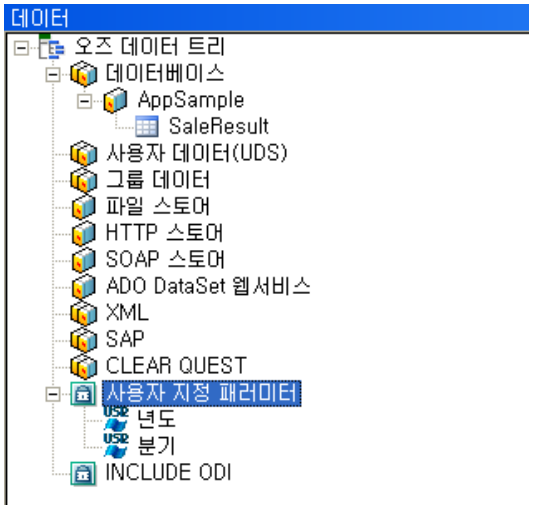
데이터 창에서 [사용자 지정 패러미터]를 마우스 오른쪽 버튼으로 클릭하여 나타나는 팝업 메뉴에서 [패러미터 설정] 메뉴를 클릭합니다.



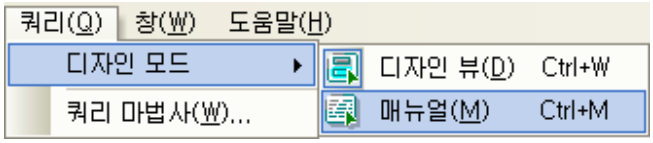

검색 조건 중 년도 값과 분기 값을 입력받을 패러미터의 이름을 '년도', '분기'로 입력한 후 [확인] 버튼을 클릭하여 추가합니다.



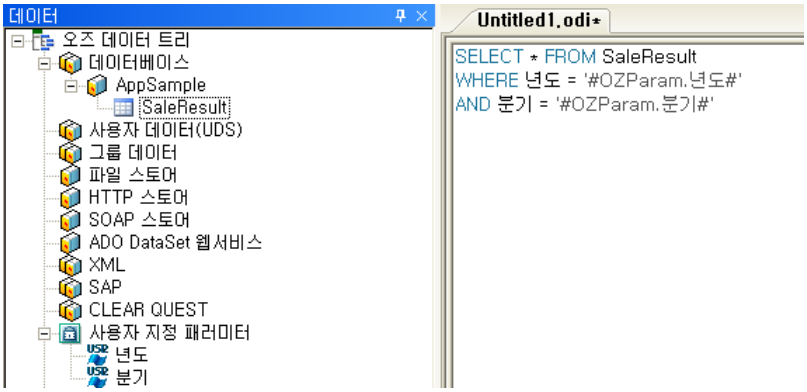

[사용자 지정 패러미터]에 패러미터가 제대로 추가되었는지 확인합니다.




4) 데이터 셋을 가져오는 쿼리를 직접 입력하기 위해 디자인 모드를 매뉴얼 모드로 변경합니다. 데이터 창에서 'SaleResult'를 클릭한 후 메뉴바에서 [쿼리] - [디자인 모드] - [매뉴얼] 메뉴를 클릭합니다. 쿼리를 직접 입력하지 않고 디자인 뷰 모드를 이용하여 쿼리를 생성하는 방법은 "오즈 쿼리 디자이너 사용자 매뉴얼"을 참조하십시오.



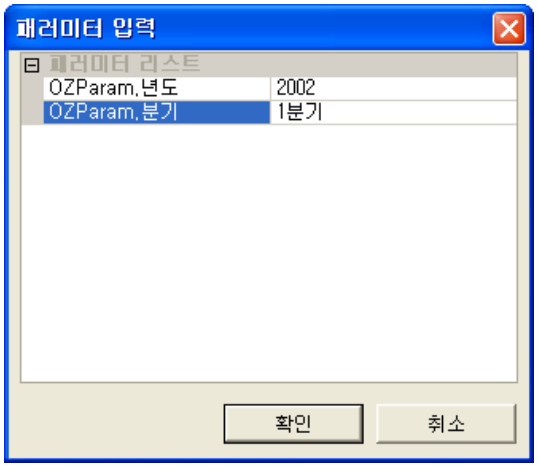
5) 쿼리 입력 창에 쿼리를 입력합니다.



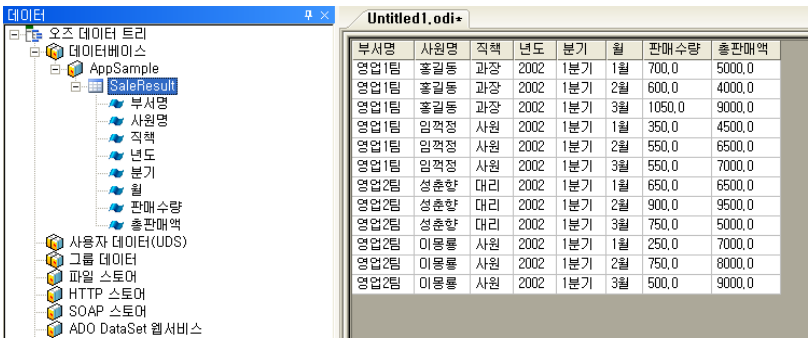

입력한 쿼리 내용은 다음과 같습니다.

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

6) 쿼리문 실행 아이콘( )을 클릭하거나 <F5>키를 눌러 나타나는 다이얼로그에 'OZParam.년도'에는 '2002'를 'OZParam.분기'에는 '1분기'라고 입력한 후 [확인] 버튼을 클릭합니다.

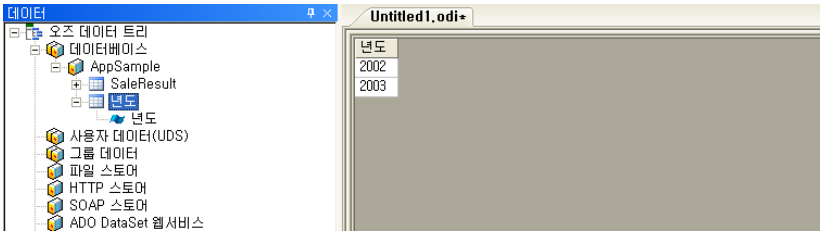



아래 그림과 같이 검색 결과와 'SaleResult' 데이터 셋에 부서명, 사원명 등 데이터를 나타내는 필드명이 제대로 표시됨을 확인합니다.




7) 검색 조건을 콤보박스에 나타낼 년도와 분기를 추출할 데이터 셋을 추가합니다.

먼저 '년도' 값을 나타낼 데이터 셋을 추가합니다. 데이터 창에서 'AppSample'을 마우스 오른쪽 버튼으로 클릭하여 나타나는 팝업 메뉴에서 [데이터 셋 추가] 메뉴를 클릭합니다. '데이터 셋 이름'을 '년도'라고 입력하고 쿼리문을 입력(입력 방법은 위 내용 중 4), 5) 번 참조 바람)한 후 쿼리문 실행 아이콘( )을 클릭하여 결과를 확인합니다.




입력한 쿼리 내용은 다음과 같습니다.

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

다음으로 '분기' 값을 추출합니다. 데이터 창에서 'AppSample'을 마우스 오른쪽 버튼으로 클릭하여 나타나는 팝업 메뉴에서 [데이터 셋 추가] 메뉴를 클릭합니다. '데이터 셋 이름'을 '분기'라고 입력하고 쿼리문을 입력한 후 쿼리문 실행 아이콘( )을 클릭하여 결과를 확인합니다.

입력한 쿼리 내용은 다음과 같습니다.

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

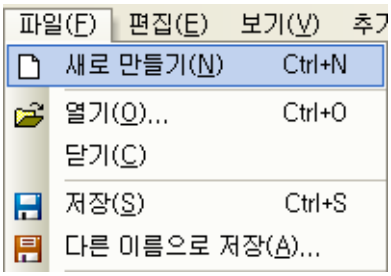
8) 작성한 ODI 파일을 저장합니다. 메뉴바에서 [파일] - [저장] 메뉴를 클릭하여 'AppSample.odi'라는 이름으로 리파지토리 루트 폴더에 저장합니다. 본 예제에서 리파지토리 루트 폴더는 'C:\Repository' 입니다.

## Step 2 보드 디자인과 스크립트 입력

### ◆ 새 프로젝트 생성

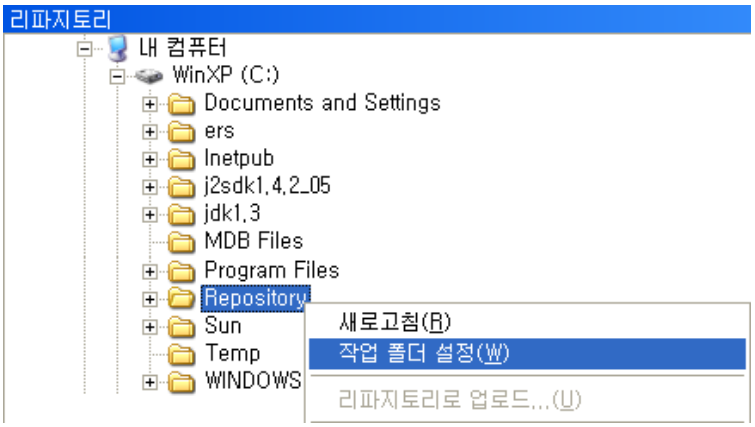
오즈 애플리케이션 디자이너를 실행하여 새로운 프로젝트를 만들고 작업 폴더를 설정합니다.

1) 메뉴바에서 [파일] - [새로 만들기] 메뉴를 클릭하거나 툴바에서 새로 만들기 아이콘( )을 클릭하여 새로운 프로젝트를 만듭니다.




2) 오즈 애플리케이션 디자이너 좌측 화면 상단에서 [리파지토리] 탭을 클릭하고 작업 폴더로 설정할 폴더를 선택한 후 마우스 오른쪽 버튼을 클릭하여 나타나는 팝업 메뉴에서 [작업 폴더 설정] 메뉴를 클릭하여 해당 폴더를 작업 폴더

로 설정합니다. 작업 폴더는 개발 환경마다 다르며 본 예제에서는 C:\Repository를 작업 폴더로 설정하였습니다.

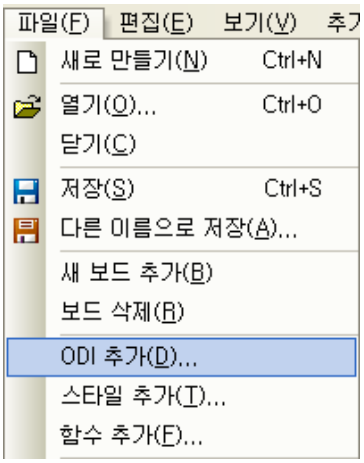



### ◆ ODI 파일 추가

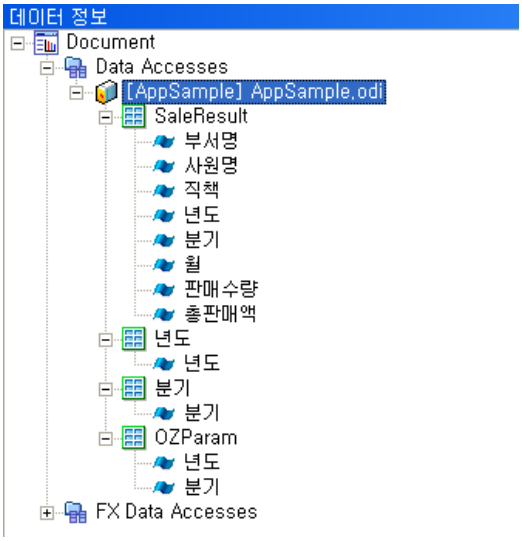
작성한 ODI 파일(AppSample.odi)을 추가합니다.

1) ODI 파일을 추가합니다.

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




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

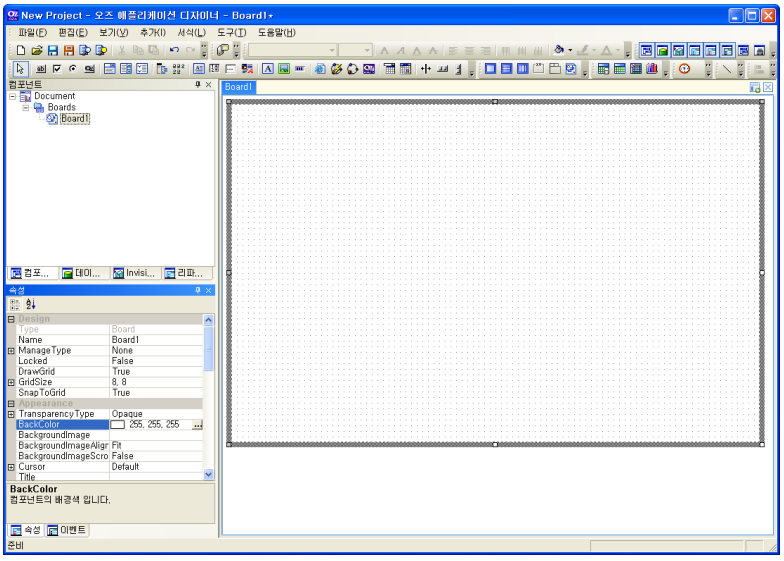



◆ 보드 디자인

보드의 배경색과 타이틀을 디자인합니다.

1) 보드의 배경색을 흰색으로 변경합니다.

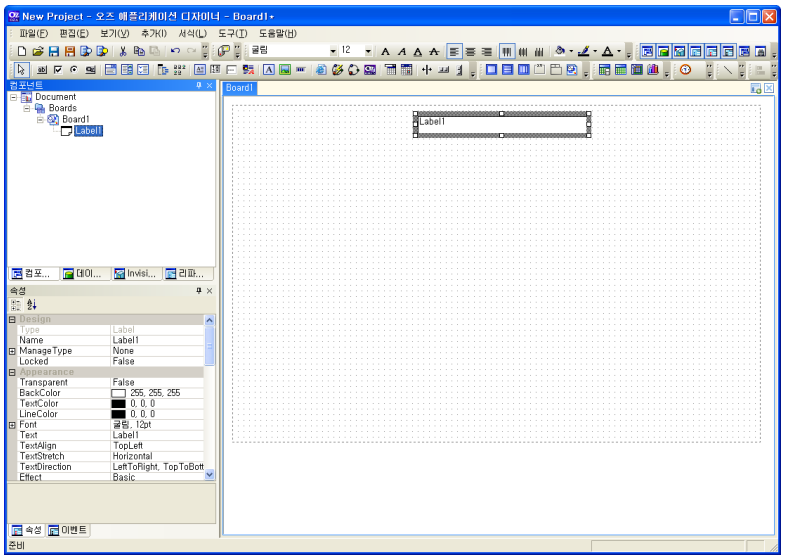
[컴포넌트] 탭에서 [Board1]을 선택한 후 속성 창에서 'BackColor'를 선택하고 색 선택 버튼[...]을 클릭하여 흰색으로 변경합니다.



2) 타이틀을 표시할 라벨을 추가합니다.

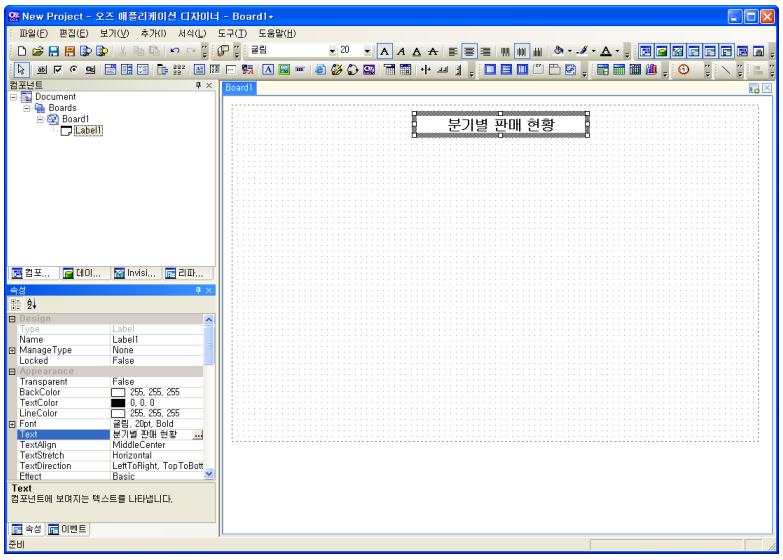
메뉴바에서 [추가] - [컨트롤] - [라벨] 메뉴를 클릭하거나 툴바에서 라벨 아이

콘( )을 클릭한 후 표시하고자 하는 영역만큼 Drag & Drop하여 라벨을 추가합니다.



3) 추가한 라벨의 색, 글꼴, 텍스트 등을 디자인합니다.

'Label1'을 선택한 후 속성 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.

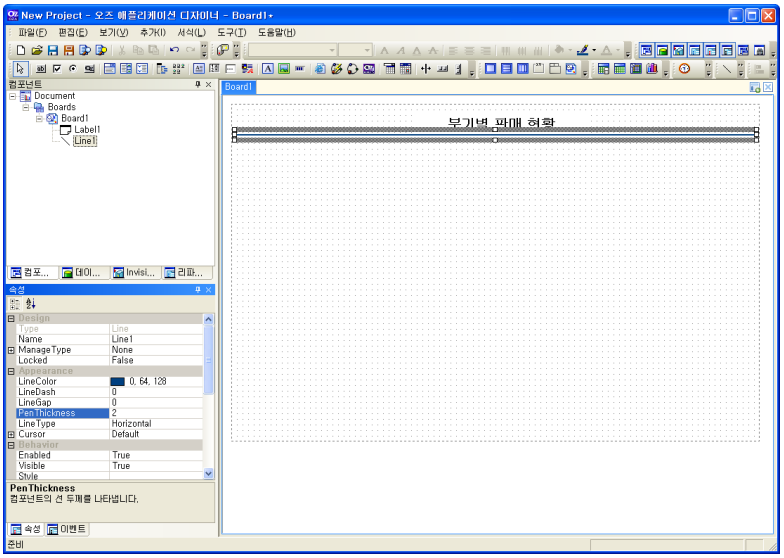



4) 타이틀 아래에 표시할 선을 추가하고 디자인합니다.

메뉴바에서 [추가] - [그리기 컨트롤] - [선] 메뉴를 클릭하거나 툴바에서 선

아이콘( )을 클릭한 후 표시하고자 하는 영역만큼 Drag & Drop하여 선을 추가합니다.

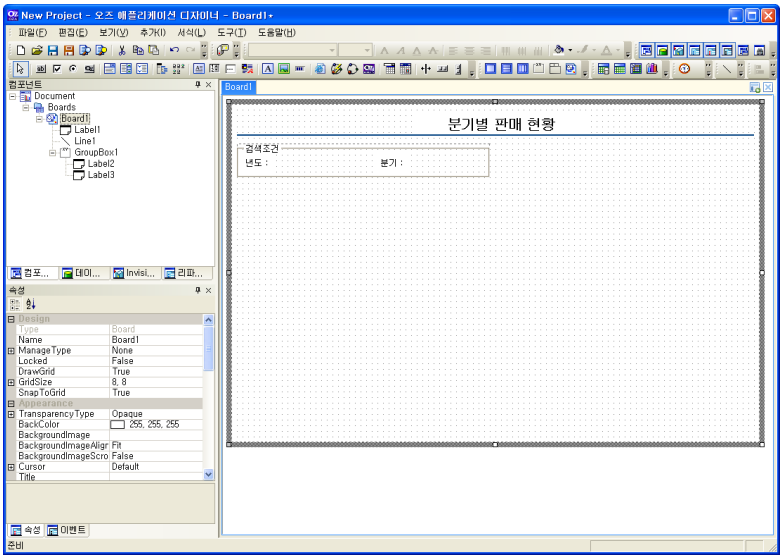
'Line1'을 선택한 후 속성 창에서 'LineColor'를 선택하여 라인 색을 변경하고 'PenThickness'를 선택하여 라인의 두께를 적절하게 변경합니다.




◆ **검색 조건용 UI 디자인**

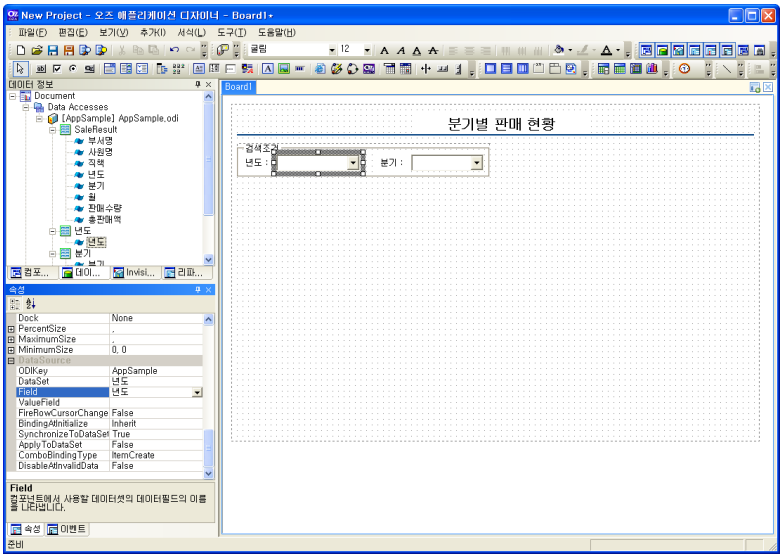
그룹박스, 라벨, 버튼을 추가하고 년도와 분기를 입력 받을 콤보박스를 추가하여 해당 값을 나타내는 데이터 셋을 연결합니다.

1) 그룹박스와 라벨을 추가하고 아래 그림과 같이 디자인합니다.

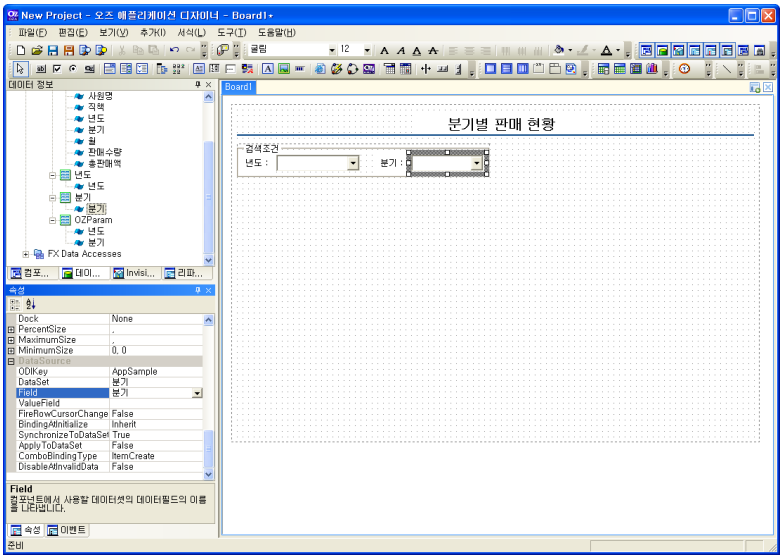


2) 콤보박스 2개를 그룹박스에 추가합니다.

3) 콤보박스에 연동할 ODI와 데이터 셋을 설정합니다.
먼저 년도를 나타낼 콤보박스를 설정합니다. 'ComboBox1'을 선택한 후 속성 창에서 'ODIKey'를 'AppSample'로, 'DataSet'을 '년도'로, 'Field'를 '년도'로 설정합니다.

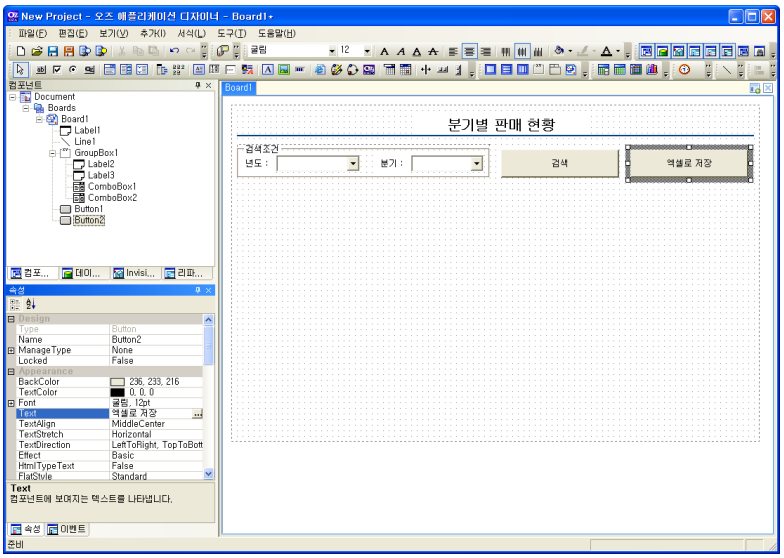



다음으로 분기를 나타낼 콤보박스를 설정합니다.
'ComboBox2'를 선택한 후 속성 창에서 'ODIKey'를 'AppSample'로, 'DataSet'을 '분기'로, 'Field'를 '분기'로 설정합니다.



4) 표시할 데이터를 검색하기 위한 검색 버튼과 테이블의 내용을 엑셀로 저장하기 위한 버튼을 추가하고 디자인합니다.

버튼을 두 개 추가한 후 추가한 버튼의 Text를 '검색'과 '엑셀로 저장'이라고 입력하고 크기와 위치를 아래 그림과 같이 디자인합니다.




◆ 차트 디자인

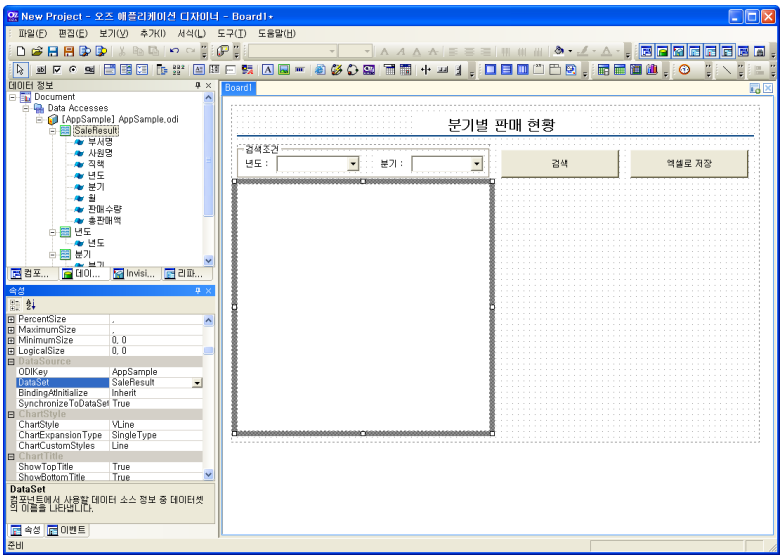
ODI 파일과 데이터 셋을 차트에 연결한 후 마법사를 이용하여 표시할 데이터를 설정하고 차트의 모양을 디자인합니다.

1) 차트를 보드에 추가합니다.

메뉴바에서 [추가] - [컴플렉스 컨트롤] - [차트] 메뉴를 클릭하거나 툴바에서 차트 아이콘( )을 클릭한 후 표시하고자 하는 영역만큼 Drag & Drop하여 차트를 추가합니다

2) 차트에 연동할 ODI와 데이터 셋을 설정합니다.

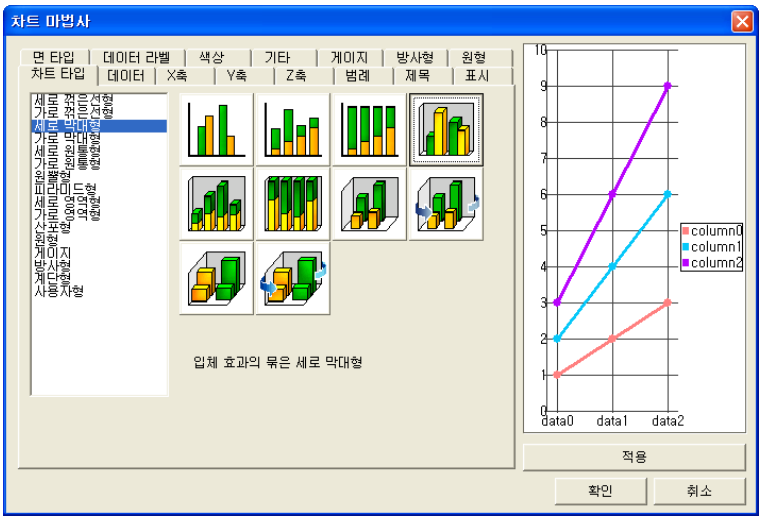
'Chart1'을 선택한 후 속성 창에서 'ODIKey'는 'AppSample'로 'DataSet'은 'SaleResult'로 설정합니다.



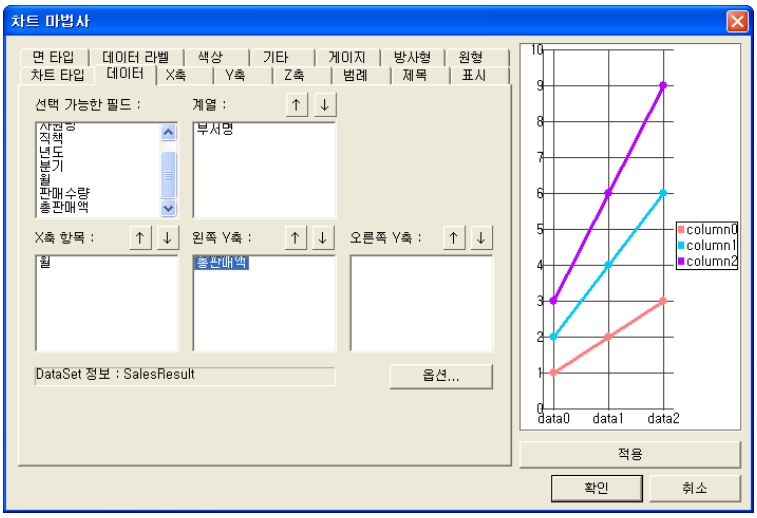

3) 차트 마법사를 이용하여 표시할 데이터 및 차트의 형태를 설정합니다.

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

[차트 타입] 탭을 클릭하여 차트의 형태를 세로 막대형의 '입체효과의 묶은 세로 막대형'으로 설정합니다.

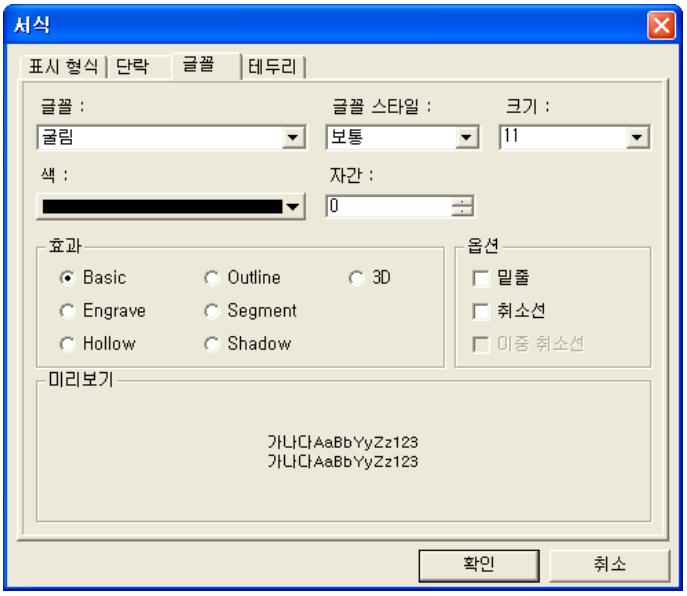



4) [데이터] 탭을 클릭한 후 X축, Y축, 범례에 표시하고자 하는 필드명을 마우스로 Drag & Drop합니다.

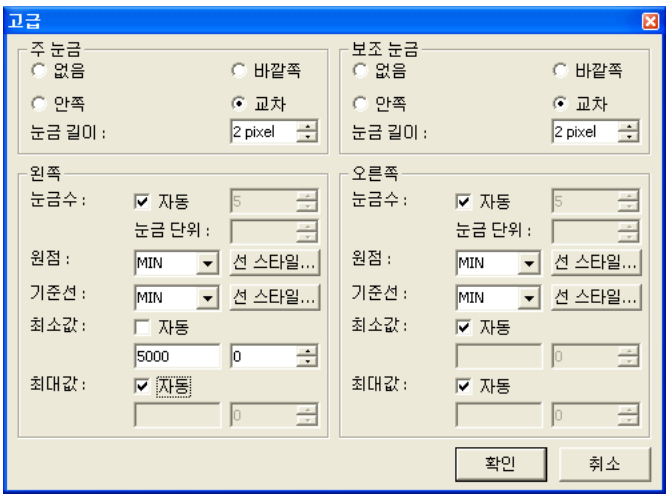



5) 차트 마법사를 이용하여 제목, 범례, X축의 글꼴, Y축의 최소/최대 값 등 차트의 모양을 디자인합니다.

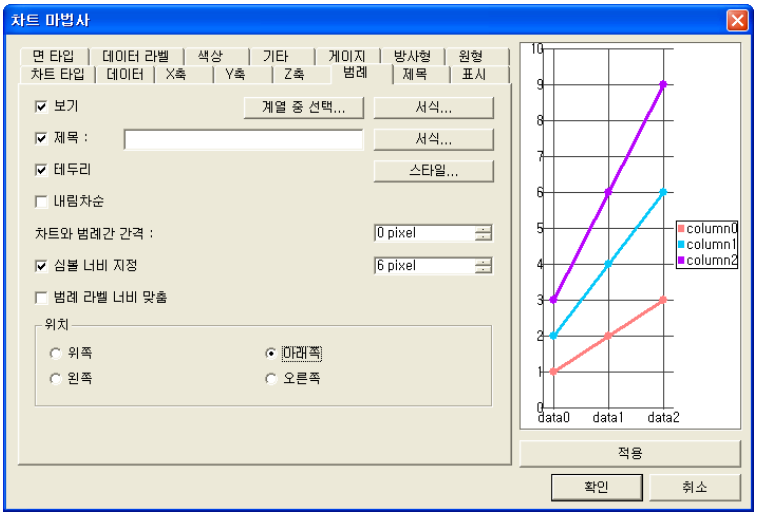
[X축] 탭을 클릭한 후 [서식] 버튼을 클릭하여 X축의 글꼴을 설정합니다.



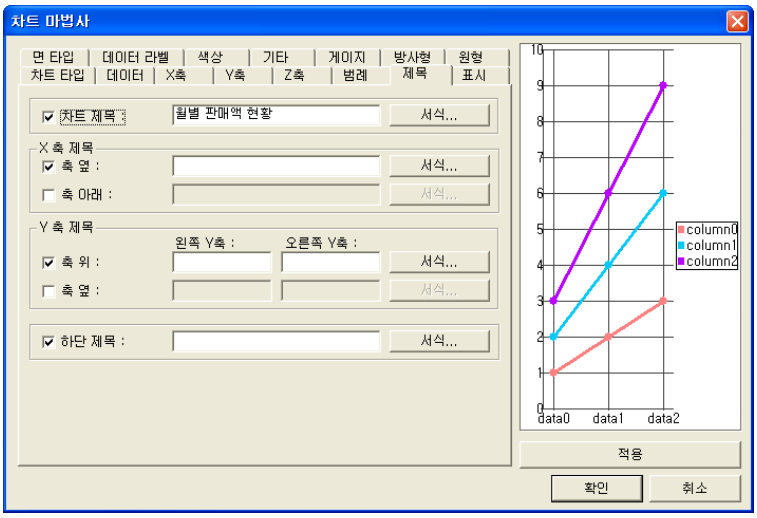

[Y축] 탭을 클릭한 후 [서식] 버튼을 클릭하여 Y축의 글꼴을 설정하고, [고급] 버튼을 클릭하여 Y축의 최소/최대값 등을 설정합니다.



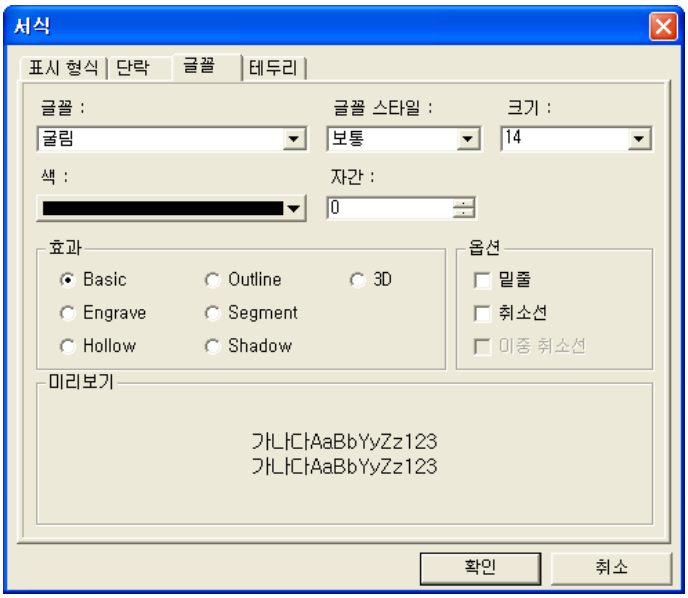

[범례] 탭을 클릭한 후 범례의 위치를 '아래쪽'으로 설정합니다.




[제목] 탭을 클릭하여 차트의 제목을 입력합니다.




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




6) [확인] 버튼을 클릭하여 차트 마법사를 종료합니다.

## ◆ 테이블 디자인

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

1) 테이블을 보드에 추가합니다.

메뉴바에서 [추가] - [컴플렉스 컨트롤] - [테이블] 메뉴를 클릭하거나 툴바에서 테이블 아이콘( )을 클릭한 후 표시하고자 하는 영역만큼 Drag & Drop 하여 테이블을 추가합니다.

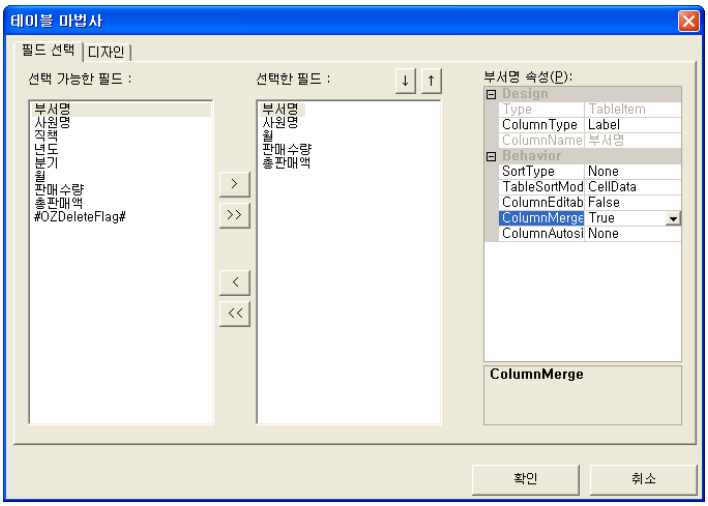
2) 테이블에 연동할 ODI와 데이터 셋을 설정합니다.

'Table1'을 선택한 후 속성 창에서 'ODIKey'는 'AppSample'로, 'DataSet'은 'SaleResult'로 설정합니다.

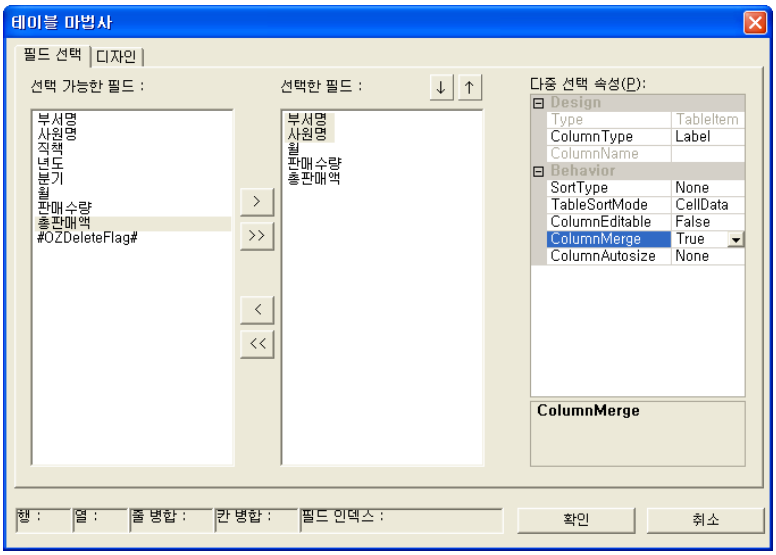
3) 테이블 마법사를 이용하여 표시할 필드명을 추가합니다.

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

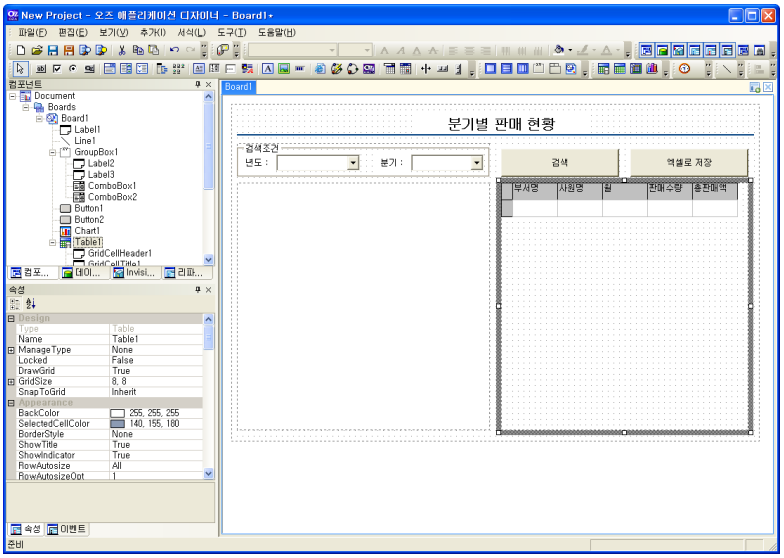
테이블 마법사에서 [ ] 버튼을 이용하여 아래 그림과 같이 필드를 추가합니다.




선택한 필드 중 '부서명'과 '사원명'은 같은 데이터인 경우 셀을 병합하기 위해 '부서명'과 '사원명'을 선택한 후 화면 오른쪽의 속성 창에서 'ColumnMerge' 속성 값을 'True'로 변경합니다.

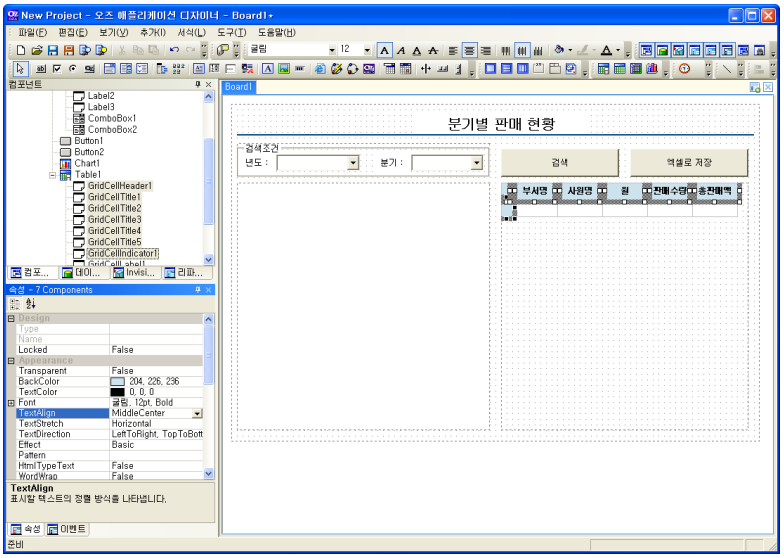



[확인] 버튼을 클릭하여 테이블이 추가된 것을 확인합니다.




4) 테이블 모양을 디자인합니다.

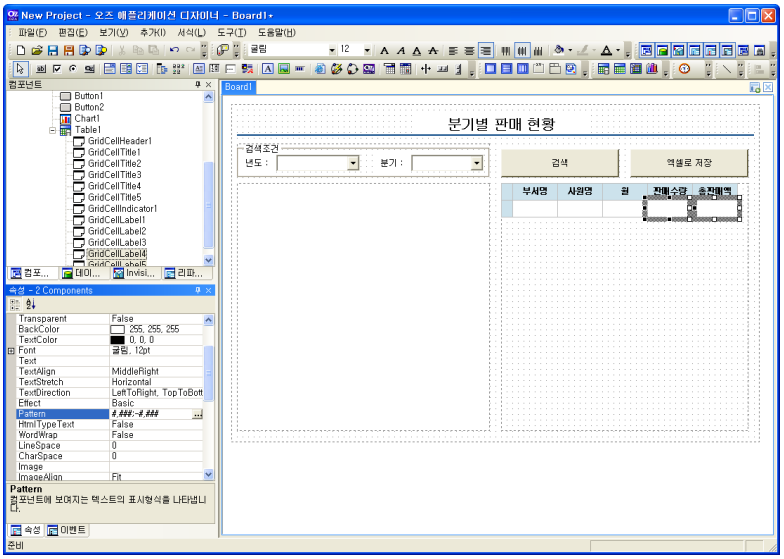
'Table1'을 선택하고 속성 창에서 'ShowIndicator' 값을 'False'로 설정합니다. 'GridCellHeader1', 'GridCellTitle1' ~ 'GridCellTitle5', 'GridCellIndicator1'을 선택한 후 'BackColor'는 '204, 226, 236', 'CellFrameColor'는 '192, 192, 192'로, 'Font'는 '굴림, 12pt, Bold'로, 'TextAlign'은 'MiddleCenter'로 설정합니다.




다음으로 데이터가 표시될 라벨을 디자인합니다.

'Table1'의 'GridCellLabel' 중 부서명, 사원명, 월에 해당하는 라벨을 선택한 후 'Font'는 '굴림, 12pt'로, 'TextAlign'은 'MiddleCenter'로 설정합니다.

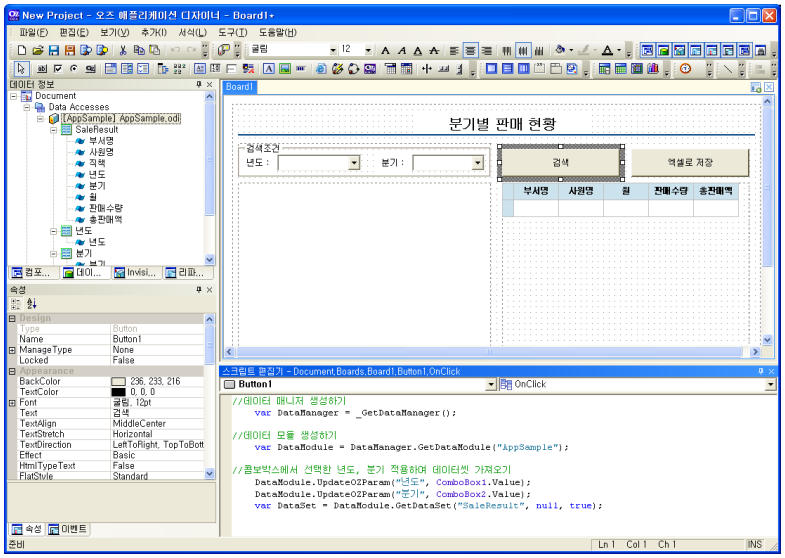
'Table1'의 'GridCellLabel' 중 판매수량, 총판매액에 해당하는 라벨을 선택하여 'Font'는 '굴림, 12pt'로, 'TextAlign'은 'MiddleRight'로, 'Pattern'은 패턴 설정 버튼[...]을 클릭하여 범주를 '숫자'로, 형식을 "#,###;-#,###"로 설정합니다.



## ◆ 스크립트 입력

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

1) [검색] 버튼을 클릭하면 년도와 분기에서 선택한 값으로 데이터가 검색되도록 스크립트를 입력합니다. 'Text' 속성 값이 '검색'인 버튼을 마우스 오른쪽 버튼으로 클릭하여 나타나는 팝업 메뉴에서 [스크립트 편집] 메뉴를 클릭하여 활성화된 스크립트 편집 창에서 이벤트를 OnClick로 선택한 후 아래와 같이 스크립트를 입력합니다.




입력한 스크립트는 다음과 같습니다.

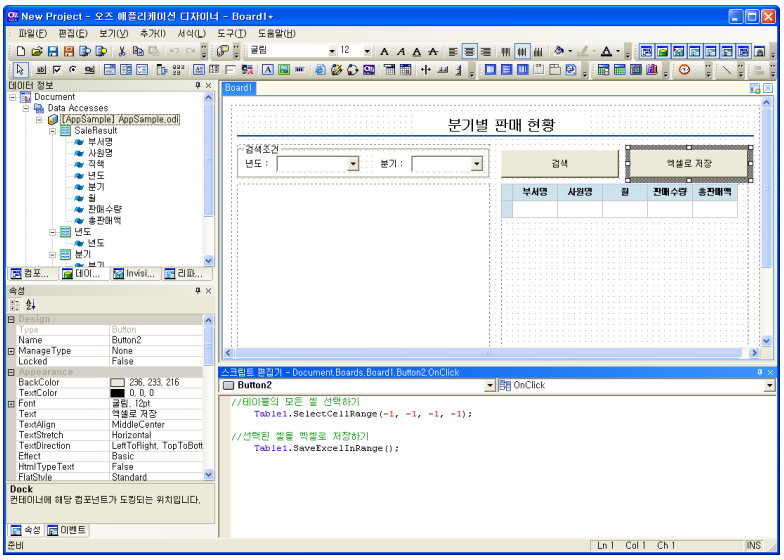
```
//데이터 매니저 생성하기
  var DataManager = _GetDataManager();

//데이터 모듈 생성하기
  var DataModule = DataManager.GetDataModule("AppSample");

//콤보박스에서 선택한 년도, 분기 적용하여 데이터셋 가져오기
  DataModule.UpdateOZParam("년도", ComboBox1.Value);
  DataModule.UpdateOZParam("분기", ComboBox2.Value);
  var DataSet = DataModule.GetDataSet("SaleResult", null, true);
```

2) [엑셀로 저장] 버튼을 클릭하면 테이블의 데이터를 엑셀 파일로 저장하도록 스크립트를 입력합니다. 'Text' 속성 값이 '엑셀로 저장'인 버튼을 마우스 오른쪽 버튼으로 클릭하여 나타나는 팝업 메뉴에서 [스크립트 편집] 메뉴를 클릭

하여 활성화된 스크립트 편집 창에서 이벤트를 OnClick로 선택한 후 아래와 같이 스크립트를 입력합니다.

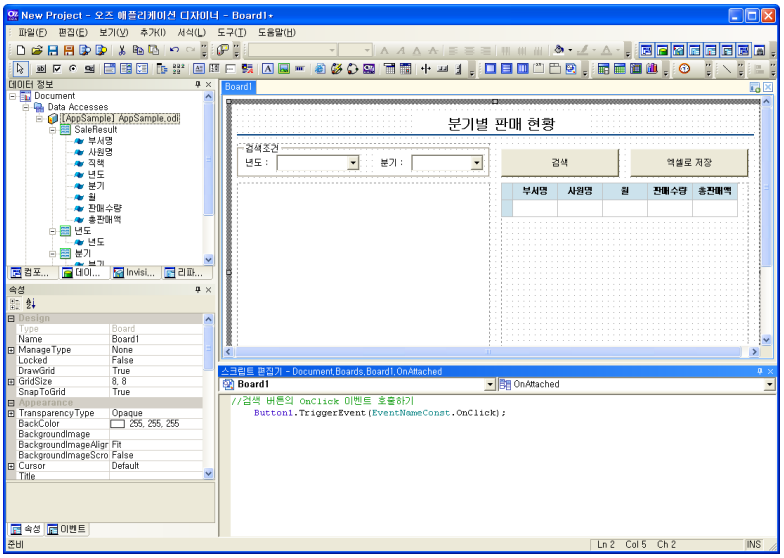



입력한 스크립트는 다음과 같습니다.

```
//테이블의 모든 셀 선택하기
  Table1.SelectCellRange(-1, -1, -1, -1);

//선택된 셀을 엑셀로 저장하기
  Table1.SaveExcelInRange();
```

3) 보드가 화면에 표시될 때 차트와 테이블에 데이터가 표시되도록 보드에 스크립트를 입력합니다. 보드를 마우스 오른쪽 버튼으로 클릭하여 나타나는 팝업 메뉴에서 [스크립트 편집] 메뉴를 클릭하여 활성화된 스크립트 편집 창에서 이벤트를 OnAttached로 선택한 후 아래와 같이 스크립트를 입력합니다.

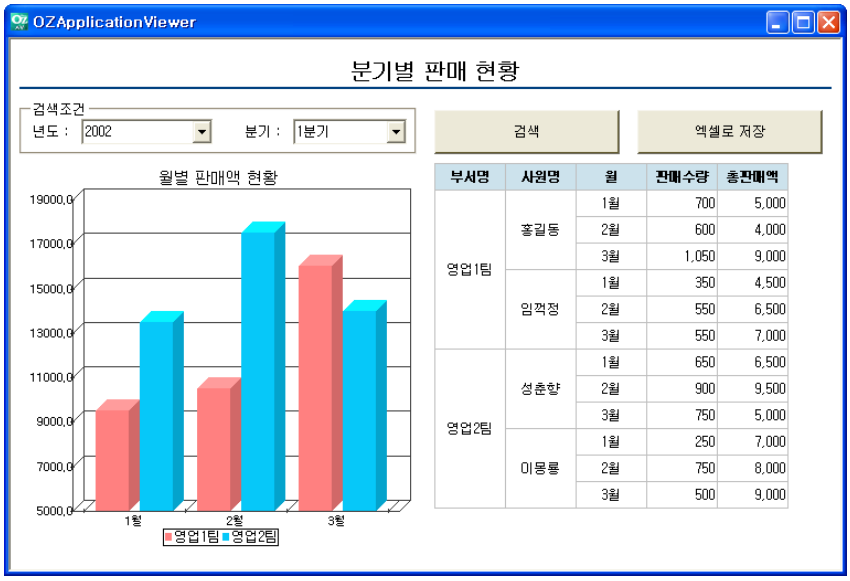



입력한 스크립트는 다음과 같습니다.

```
//검색 버튼의 OnClick 이벤트 호출하기
  Button1.TriggerEvent(EventNameConst.OnClick);
```

## Step 3 완성된 프로젝트 미리보기

오즈 애플리케이션 디자이너의 메뉴바에서 [파일] - [미리보기] 메뉴를 클릭하거나 툴바에서 미리보기 아이콘( )을 클릭합니다.




디자인한 파일을 'AppSample.oza'로 저장합니다.

# 애플리케이션 운영 절차

## Step 1 오즈 서버 구동

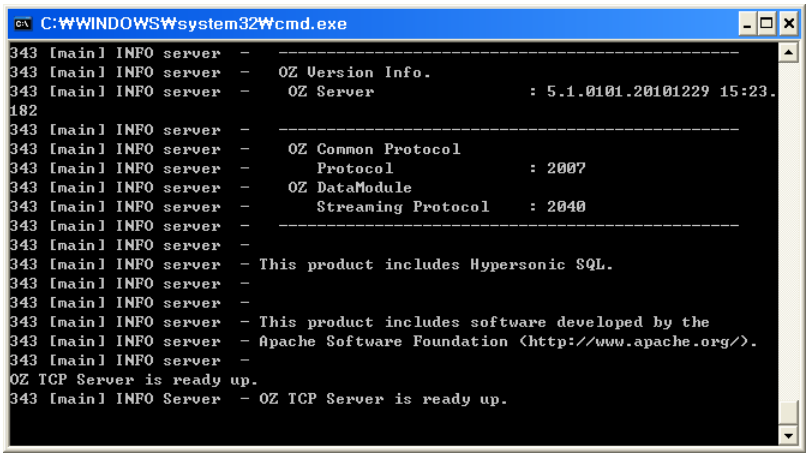
### ◆ 데이터베이스 정보 추가

오즈 서버가 설치된 폴더의 conf 폴더에 있는 db.properties 파일을 문서 편집기로 열어서 ODBC 연결 정보를 아래와 같이 추가합니다.

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

### ◆ 오즈 서버 실행

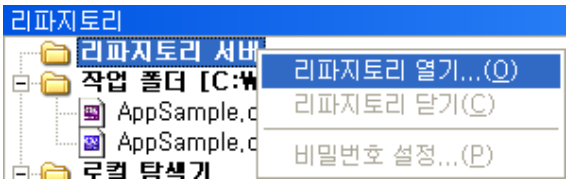
오즈 서버가 설치된 폴더의 bin 폴더에 있는 ozstart.bat 파일을 더블클릭하여 실행합니다.(오즈 서버 실행에 관한 내용은 "OZ Enterprise Server Administrator's Guide"를 참고하시기 바랍니다.)



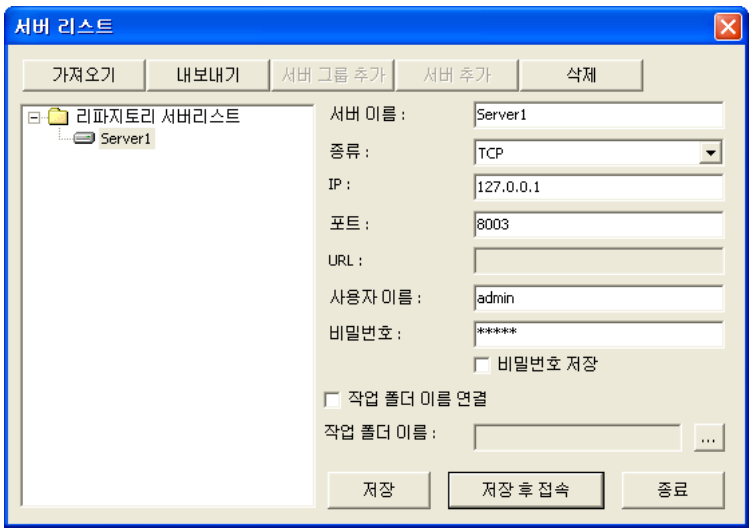
## Step 2 오즈 서버로 파일 업로드

오즈 서버가 실행되고 있는 상태에서 애플리케이션 디자이너의 리파지토리 서버에 접속하여 애플리케이션 프로젝트 파일과 ODI 파일을 업로드합니다.

1) 오즈 애플리케이션 디자이너의 [리파지토리] 탭을 클릭하고 '리파지토리 서버'에서 마우스 오른쪽 버튼을 클릭하여 나타나는 팝업 메뉴에서 [리파지토리 열기]를 클릭합니다.



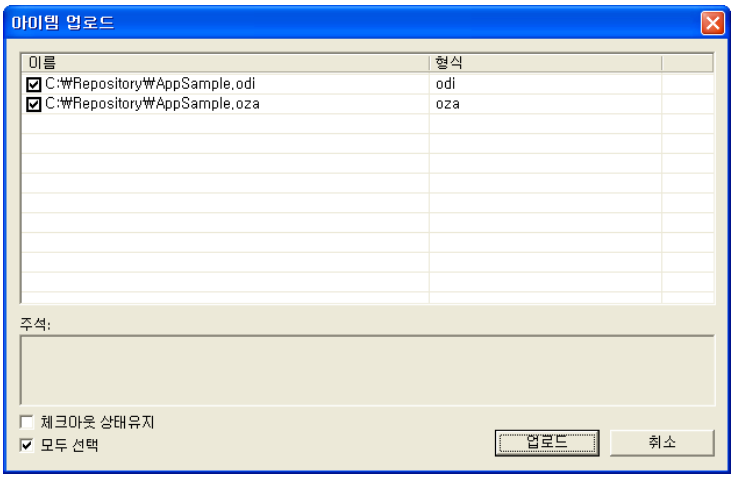

2) '서버리스트 다이얼로그'에서 [서버 추가] 버튼을 클릭하여 접속할 서버 정보를 아래 그림과 같이 입력하고 [저장 후 접속] 버튼을 클릭합니다. (비밀번호는 기본 값은 'admin' 입니다.)




'서버리스트 다이얼로그'에 대한 자세한 내용은 "오즈 애플리케이션 디자이너 사용자 매뉴얼"의 서버 리파지토리 사용법을 참조하시기 바랍니다.

3) 아래 그림과 같이 리파지토리 서버에 연결됨을 확인합니다.

4) '작업폴더'에서 파일을 선택하여 리파지토리 서버에 Drag & Drop하거나 [리파지토리로 업로드] 메뉴를 클릭하여 서버로 업로드합니다. 본 예제에서는 애플리케이션 프로젝트 파일인 'AppSample.oza'와 ODI 파일인 'AppSample.odi' 파일을 업로드합니다.




5) 아래 그림과 같이 리파지토리 서버에 업로드됨을 확인합니다.

## Step 3 태그 작성

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

```
<html>
<body>

<div id="InstallOZViewer">
    <object ID="ZTransferX" classid="CLSID:C7C7225A-9476-47AC-B0B0-FF3B79D55E67" width="0" height="0" codebase = "http://127.0.0.1/OZApplicationViewer/ZTransferX_2,2,5,1.cab#version=2,2,5,1">
    <param name="download.server" value="http://127.0.0.1/OZApplicationViewer/">
    <param name="download.port" value = "80">
    <param name="download.instruction" value = "ozaviewer.idf">
    <param name="install.base" value = "<PROGRAMS>/Forcs">
    <param name="install.namespace" value = "custom_namespace">
    </object>
</div>

<div id="RunOZViewer">
    <object ID="OZApplicationViewer" CLASSID="CLSID:907A00F3-7390-4EF2-931B-360546587804" width="100%" height="100%">
    <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="ozadoc.path" value="AppSample.oza">
    </object>
</div>

</body>
</html>
```

※ 참고사항 : 애플리케이션 뷰어를 클라이언트에 설치하기 위해서는 download.server 패러미터 값으로 설정한 폴더에 애플리케이션 뷰어 설치 파일을 복사하여야 합니다. 만일 위 태그처럼 사용하려면 IIS 루트 폴더 아래의 'OZApplicationViewer' 폴더에 애플리케이션 뷰어 설치 파일을 복사하시기 바랍니다.

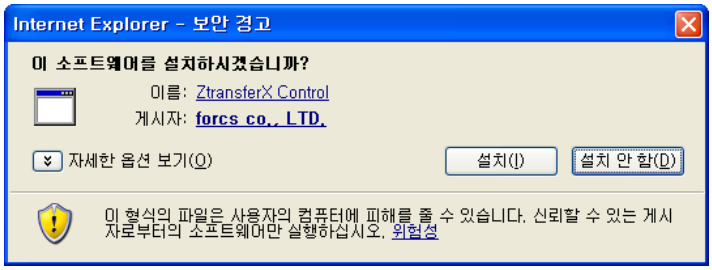
## Step 4 실행

애플리케이션 뷰어를 설치 및 실행합니다.

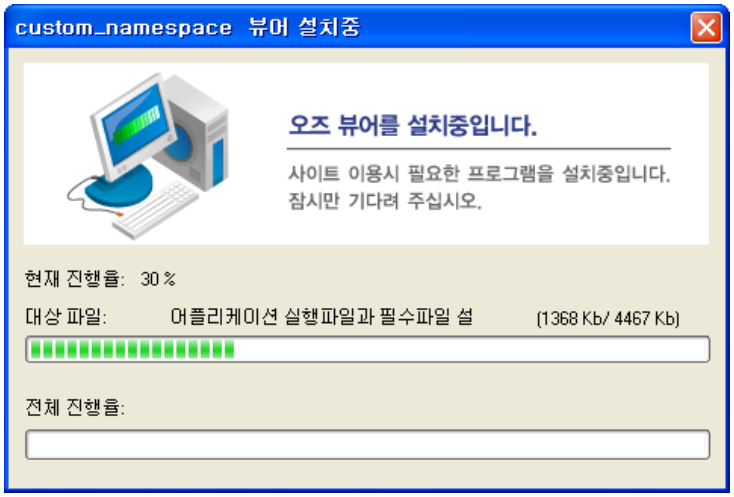
1) 오즈 서버와 웹 서버를 구동시킵니다.

2) 프로젝트 파일 호출 태그를 삽입한 HTML 파일(appsample.html)을 웹브라우저로 엽니다.

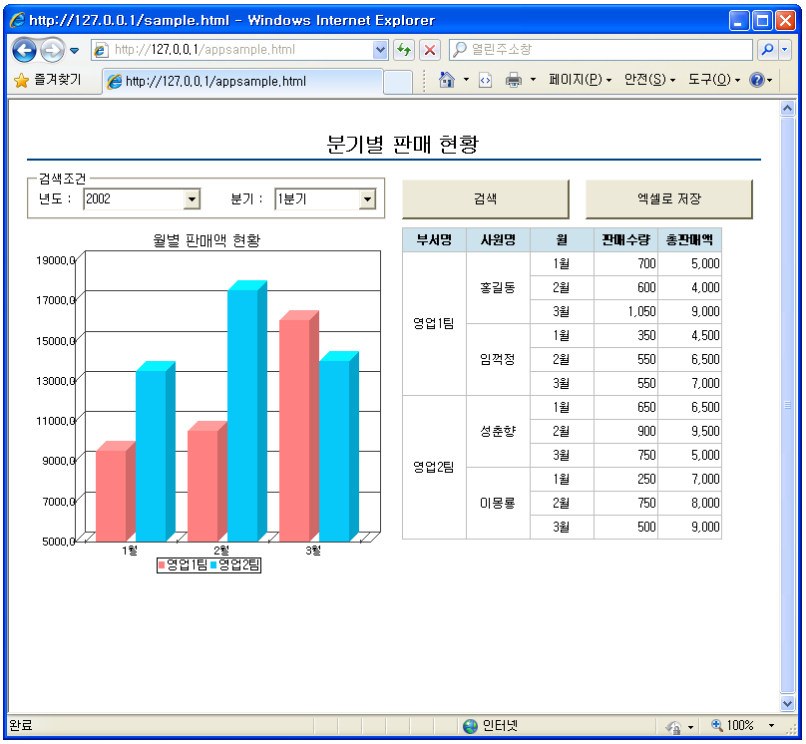
3) 인증 창이 나타나면서 Ztransfer Control을 설치할 것인지를 물어봅니다.




4) [설치] 버튼을 클릭하여 ZtransferX Control을 설치합니다. ZtransferX Control이 설치된 후 자동으로 오즈 애플리케이션 뷰어가 설치됩니다. 애플리케이션 뷰어는 설치 태그 중 'install.Base'와 'install.Namespace'에서 설정한 폴더에 설치됩니다.




5) 다음 그림과 같이 실행 결과를 확인할 수 있습니다.


http://127.0.0.1/sample.html - Windows Internet Explorer
http://127.0.0.1/appsample.html
즐겨찾기
http://127.0.0.1/appsample.html
분기별 판매 현황
검색조건
년도 : 2002
분기 : 1분기
검색
엑셀로 저장
월별 판매액 현황
19000,0
17000,0
15000,0
13000,0
11000,0
9000,0
7000,0
5000,0
1월
2월
3월
영업1팀 영업2팀
부서명 사원명 월 판매수량 총판매액
영업1팀 홍길동 1월 700 5,000
2월 600 4,000
3월 1,050 9,000
임꺽정 1월 350 4,500
2월 550 6,500
3월 550 7,000
영업2팀 성춘향 1월 650 6,500
2월 900 9,500
3월 750 5,000
이몽룡 1월 250 7,000
2월 750 8,000
3월 500 9,000
완료
인터넷
100%

# Ⅳ. 마치면서

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 Overview 를 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 강남구 논현로 646 포시에스빌딩 (우)06106**
**전화 : 82-2-6188-8282**
**e-mail : oz@forcs.com**