

- 3
 - 4
 - 4
 - 5
 - 6
 - 6

- . Features Overview 7**
 - 8
 - (OZ Query Designer) 8
 - (OZ Report Designer) 9
 - (OZ Enterprise Server) 9
 - (OZ Scheduler) 10
 - (OZ Report Viewer) 10
 - (OZ Enterprise Manager) 10
 - (OZ Repository Manager) 10
- Features 11
 - 11
 - 13
 - 14
 - 16
 - 16
 - 17
 - 19

. Quick Tour	21
.....	22
.....	23
Step 1. ODI	24
Step 2.	33
Step 3.	43
.....	44
Step 1.	44
Step 2. ODI	44
Step 3.	46
Step 4.	49
.....	53



본 매뉴얼은 웹 기반의 기업용 리포팅 솔루션인 'OZ Report'를 처음 접하는 사용자들에게 웹 리포팅에 대한 이해와 오즈 리포트 디자이너를 이용하여 보고서를 제작하고 웹에 적용하는 기본적인 과정을 소개하기 위해 제작되었습니다.

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

Features Overview

OZ Report Designer의 특징과 기능, 아키텍처, 운영 방식 등을 개괄적으로 설명합니다.

Quick Tour

OZ Report Designer를 이용하여 보고서를 개발하고 웹으로 배포하기까지의 전 과정을 예제를 통해 살펴봅니다.

OZ Report를 포함한 관련 OZ 제품군의 매뉴얼은 아래와 같습니다.

Manual	Description
오즈 리포트 Getting Started	오즈 리포트의 소개와 간단한 보고서 작성을 통해 오즈 리포트의 개념을 설명합니다.
오즈 쿼리 디자이너 사용자 가이드	오즈 쿼리 디자이너를 이용해 데이터 셋 정보를 작성하는 방법을 자세히 설명합니다.
오즈 리포트 디자이너 사용자 레퍼런스	오즈 리포트에서 제공하는 컴포넌트와 마법사에 대해 자세히 설명합니다.
오즈 리포트 디자이너 사용자 가이드	오즈 리포트 디자이너 사용 방법과 보고서 작성하고 작성된 보고서를 웹에 배포하는 방법에 대해 설명합니다.
오즈 리포트 뷰어 사용자 가이드	보고서를 웹을 통해 조회하고 사용자 애플리케이션과 연동할 수 있도록 뷰어 파라미터와 활용 방법 등에 대해 설명합니다.
오즈 리파지토리 매니저 사용자 가이드	오즈 리파지토리 매니저를 사용하여 오즈 폼 및 데이터 파일 등 각종 아이템을 포괄적으로 관리하는 방법을 설명합니다.
오즈 엔터프라이즈 서버 관리자 가이드	서버 관리자가 오즈 서버를 효율적으로 관리할 수 있도록 오즈 서버의 설치, 관리, 운영 방법 등을 설명합니다.
오즈 엔터프라이즈 서버 WAS 설치 가이드	오즈 엔터프라이즈 서버와 웹 애플리케이션 서버와의 연동 방법을 설명합니다.
API 활용 가이드	오즈 제품에서 사용하는 서버 API, UDS, USL, 뷰어 API 등 사용자 애플리케이션에서 직접 제어할 수 있는 API에 대해 설명합니다.
에러 메시지 레퍼런스	오즈 제품에서 보여지는 에러 메시지에 대해 설명합니다.
스크립트 가이드	오즈 리포트에서 사용할 수 있는 스크립트의 문법과 그 활용 방법에 대해 자세히 설명합니다.

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

- JVM/JDK 설치 및 JAVA로 개발된 Application의 실행에 필요한 환경 설정 방법
- DB 관련지식 (JDBC/ODBC, SQL 작성 등)
- 해당 업무개발에 사용될 프로그래밍 언어
- 오즈 엔터프라이즈 서버와 연동될 WAS(Web Application Server)에 대한 기본 지식

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

본 제품은 웹을 기반으로 하는 모든 애플리케이션 개발 업무에 활용 가능합니다.

최근 현업에서 대표적으로 적용되고 있는 분야는 다음과 같습니다.

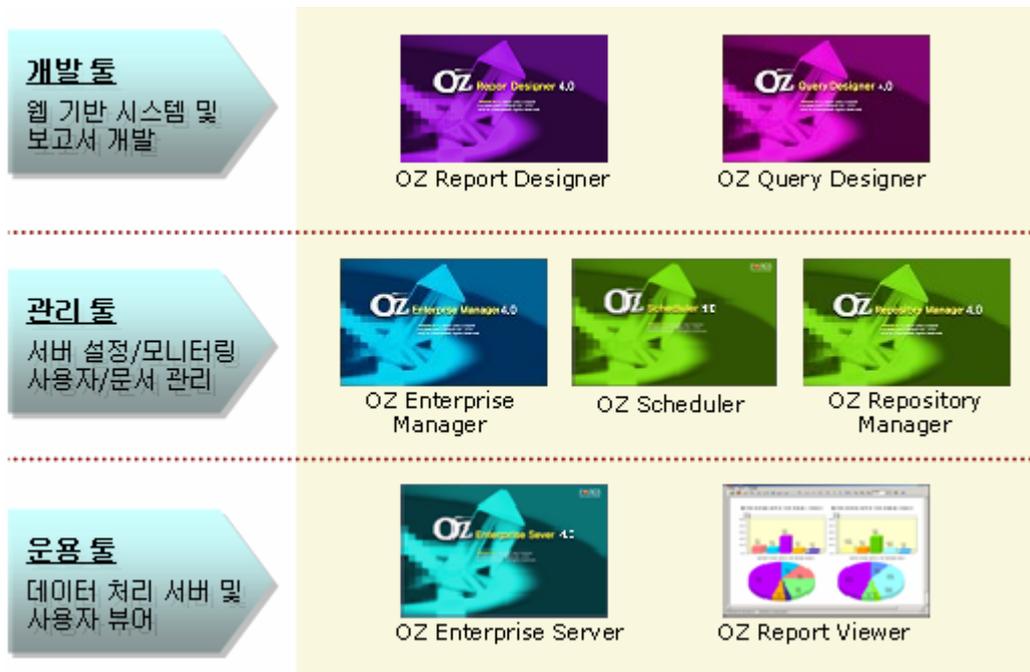
- 기업내 기간 업무 시스템 (Intranet)
- 기업간 거래 시스템 (B2B)
- B2C 및 웹 포털
- 민원 서비스, 대학/병원 업무 (G2B, G2C)
- ASP/MSP 등의 서비스

. Features Overview



 **Features**





<오즈 제품 구성>

'OZ Report'는 오즈 쿼리 디자이너, 오즈 리포트 디자이너, 오즈 엔터프라이즈 서버, 오즈 스케줄러, 오즈 리포트 뷰어, 오즈 엔터프라이즈 매니저, 오즈 리파지토리 매니저 등 다수의 기능 모듈로 구성되어 있으며 상호 긴밀한 협업을 통해 최상의 엔터프라이즈 리포팅 환경을 구현합니다.

(OZ Query Designer)

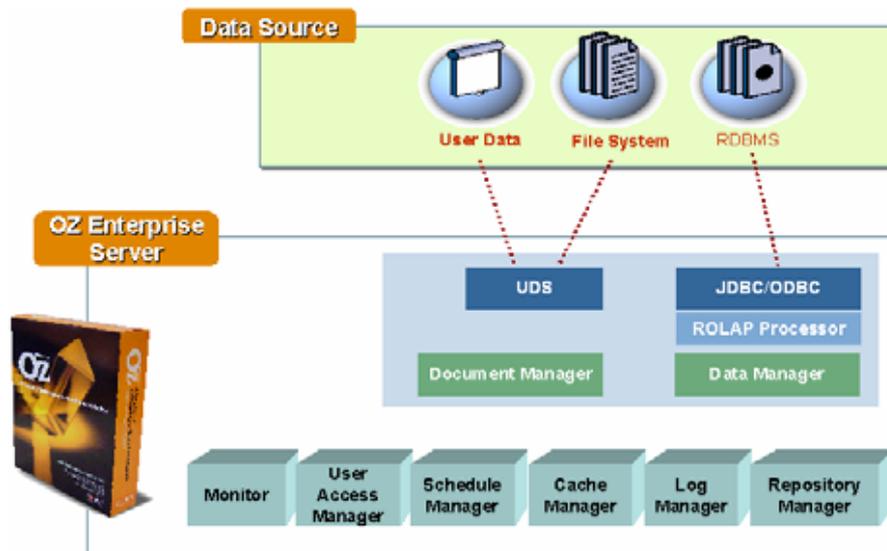
- 보고서 디자인에 필요한 데이터 셋(ODI:OZ Data Info)을 생성할 수 있는 쿼리 디자인 도구입니다.
- DB 연결 및 쿼리 개발 과정을 비전문가도 수행할 수 있도록 구현한 질의 도구로서, 테이블 간의 관계를 시각적으로 보여주며, 드래그 & 드롭(Drag and Drop)에 의한 쿼리 디자인 및 즉시 실행으로 정확한 쿼리문을 생성할 수 있도록 도와 줍니다.

(OZ Report Designer)

- 다양한 컴포넌트를 이용하여 GUI 환경에서 시각적으로(WYSIWYG : What You See Is What You Get) 보고서를 디자인할 수 있도록 해주는 리포팅 디자인 도구입니다.
- 보고서 마법사가 제공되어 마우스 클릭만으로 손쉽게 기본적인 형태의 보고서를 생성할 수 있습니다.
- 오즈 배포 마법사(OZ Publishing Wizard)는 사용자 애플리케이션에서 오즈 보고서를 호출하는 HTML, JSP 또는 ASP 코드를 자동 생성하여 개발자 편의를 극대화시켜 줍니다.

(OZ Enterprise Server)

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



(OZ Scheduler)

- 오즈 엔터프라이즈 서버와 연동하여 관리자가 미리 지정한 주기·시간에 맞춰 자동으로 보고서를 생성하고 e-mail과 웹을 통해 배포하는 기능을 담당하는 오즈 전용 작업 스케줄링 서버입니다.
- 다량의 데이터를 처리하는 배치 작업(Batch Job) 보고서 또는 주기적으로 반복되어 작성되는 보고서에 유용하며, 오즈 엔터프라이즈 매니저를 통해서 스케줄링 작업을 등록합니다.

(OZ Report Viewer)

- 최종 사용자가 웹을 통해 보고서를 '미리보기'하고, 프린터로 출력하는 기능을 담당하는 오즈 보고서 전용 뷰어입니다.
- 사용자가 조회 중인 보고서의 편집, 속성 변환, Microsoft(이하 MS) Word, MS Powerpoint, MS Excel, PDF, TIFF, HTML 등 다양한 포맷 형태의 파일 저장 기능, 모아찍기, 확대/축소 출력 등 진보된 인쇄 기능을 제공합니다.
- 동적인 사용자 입력 컴포넌트 기능과 보고서 내용에 대한 검색 기능을 제공합니다.

(OZ Enterprise Manager)

- 오즈를 이용하여 구축된 웹 리포팅 환경 하에서 서버의 운영상태에 대한 모니터링 및 스케줄링 등을 통해 서비스 운용의 편의성을 극대화하는 관리자용 콘솔입니다.
- 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 서버와 연동하여 서버 리파지토리를 편리하게 탐색 및 관리하기 위한 관리자용 도구입니다.
- 보고서 폼 파일, ODI 파일 등 각종 아이템 저장 및 수정에 대한 히스토리와 버전 관리, 계층적인 카테고리 관리 및 권한 설정 기능을 제공합니다.

Features

■ 시각적인 디자인 환경

보다 강력해진 IDE(Integrated Develop Environment)와 더 세밀해진 디자인 환경과 풍부한 컴포넌트 그리고 적재적소에 제공되는 마법사(Wizard) 기능, 데이터나 보고서 양식을 자유자재로 가공할 수 있는 자체 스크립트 언어 등 현업의 복잡하고 다양한 보고서를 쉽고 빠르게 구현할 수 있습니다.

■ 보다 손쉬운 복잡한 표 디자인을 위한 고정 테이블 컴포넌트

밴드 방식 보고서 디자인 솔루션의 한계를 극복하기 위해 개발된 고정 테이블 컴포넌트가 제공됩니다. 이 강력한 기능의 컴포넌트는 워드의 표 그리기 기능과 유사한 기능으로 셀/행의 삽입/분할/병합 등 고정 폼 형태의 보고서를 디자인을 하고자 할 때 필요한 강력한 능력을 제공합니다.

■ 보다 진보된 확장 밴드

밴드 기반의 보고서 디자인 환경을 제공합니다. 표준 밴드 외에도 사이드 밴드(Side Band), 테일 밴드(Tail Band), 더미 밴드(Dummy Band), 리전(Region) 등 특수 용도의 밴드를 추가 제공함으로써 복잡하고 특수한 형태의 보고서를 보다 효율적으로 디자인할 수 있도록 해줍니다.

■ 강력한 자체 스크립트

자체 스크립트 언어와 다수의 내장된 함수들을 제공하여 기본 컴포넌트 만으로는 해결하기 어려운 비정형의 유동적인 보고서를 쉽게 구현합니다.

■ 종속 보고서 / 다단 보고서

보고서 내에 또 다른 여러 개의 종속 보고서(Sub-Reports)가 포함된 복합 (Composite) 보고서를 지원하며, 다단 보고서(Multiple Section Reports)를 컴포넌트 수준에서 손쉽게 구현합니다.

■ 혁신적인 데이터 그룹핑

쿼리 결과를 여러 계층의 데이터 그룹으로 재분할하고 분할된 데이터 그룹 간에 다양한 관계를 설정할 수 있는 데이터 재편성 기능을 통해 강력한 데이터 조작이 가능합니다.

■ **다이나믹 쿼리 보고서**

보고서의 내용을 구성하는 DB, 테이블 및 쿼리 정보가 최종 사용자에게 의해 런타임 시에 결정되는 Ad-Hoc 쿼리 보고서를 지원합니다.

■ **완벽한 오토사이징**

여타 리포팅 툴들이 텍스트 라벨 컴포넌트 수준에서 제한적으로 제공하는 오토 사이징 기능을 이미지, 바코드 컴포넌트는 물론 테이블이나 크로스탭 등과 같은 그룹 데이터 컴포넌트에서도 완벽히 지원합니다.

■ **다양하고 높은 수준의 컴포넌트**

웹상의 특정 디렉토리(URL)에 위치한 파일 이미지 또는 DB의 BLOB 이미지 등 다양한 형식으로 존재하는 이미지를 지원하며 각종 차트와 산업 표준 바코드 및 간트 차트 등 다른 툴들이 3rd-Party 제품에 의존하고 있는 고급 컴포넌트들을 자체 컴포넌트로 제공하여 보고서 개발 생산성을 한층 더 높여 줍니다.

2D / 3D 차트 컴포넌트

다양한 종류의 차트를 지원하며 차트 형태를 결정하는 다양한 조건 설정 기능과 색상, 점, 면, 라인 형태에 대한 사용자 정의 기능을 제공합니다.

(Y축, 복합 X축 등의 복합형 차트, 라인이나 막대, 점 등이 혼합된 사용자 정의 차트 지원)

간트 차트 컴포넌트

계획과 실적에 의한 진행상황 관리 등에 광범위하게 활용되고 있는 간트 차트 보고서 개발을 위한 간트 차트 컴포넌트와 마법사를 제공합니다.

다차원 바코드 컴포넌트

CODE39, CODE128 등 산업표준으로 사용되고 있는 40여 개 이상의 주요 바코드는 물론 PDF-417, QR 바코드 등 새로운 산업표준으로 자리잡고 있는 다차원 코드를 내장 형태로 지원합니다.

노트 컴포넌트

- Binary, BLOB, Long Varchar, CLOB 등과 같은 대용량 텍스트 데이터를 보고서에 간편하게 적용할 수 있도록 대용량 텍스트 라벨 컴포넌트를 별도로 제공합니다.
- 텍스트를 양에 따라 적절하게 페이지 단위로 구분하여 보여주기 위한 섬세한 페이지 컨트롤 및 오토사이징 기능을 제공합니다.

사용자 입력 컴포넌트

- 최종 사용자가 조회 중인 보고서에 간단한 내용을 입력하고 처리를 할 수 있는 컴포넌트를 제공합니다.
- GUI로 제공되는 입력 컴포넌트로는 버튼, 라디오버튼, 체크박스, 텍스트박스, 콤보박스가 있으며, 입력한 내용은 다른 웹페이지(ASP, JSP, PHP)를 연결하여 처리할 수 있습니다.

■ 순수 Java 기반의 솔루션

JVM(Java Virtual Machine) 설치가 가능한 모든 플랫폼에 설치, 운영 가능하며 다양한 웹 애플리케이션 서버 제품들과 완벽한 연동은 물론 EJB 등 사용자 애플리케이션과도 유연하게 통합 운영됩니다.

■ 고성능 서버 아키텍처

Multi-thread 서버 엔진, DB 및 Thread Pooling, Cache Manager와 같은 고성능 미들웨어 수준의 성능 향상 기술을 채택하여 다수 사용자 동시 접속 환경에서도 최적의 성능을 보장합니다.

■ 다양한 데이터 소스 지원

오즈는 기업 내에 다양한 형태로 존재하는 데이터를 활용하기 위한 데이터 연동 방안을 지원합니다. 대부분의 상용 DBMS(메인프레임 DB2와 Lotus Domino DB 포함)는 JDBC 또는 ODBC를 통해 연동합니다.

한편, 일반 Text 파일, XML, EJB 등의 Non-DB 데이터나 사용자 애플리케이션 데이터는 오즈의 UDS(User Data Store) 연동 모듈을 통해 처리합니다. 또한 이기종 DBMS 동시 연동이 가능하며, 하나의 보고서에 여러 DBMS에서 가져온 데이터를 이용할 수 있습니다.

Note

UDS(User Data Store)란?

일반적으로 DB에서 가져온 정보로 보고서를 만드는 방식 이외에 사용자 애플리케이션에 의해 구현된 비즈니스 로직을 기반으로 생성되는 EJB, HTTP Stream, File 등 Non-Database 소스를 오즈에 연동하는 데이터 인터페이스 기술

■ 실시간/대용량 데이터 처리 기능

데이터 소스로부터 가져온 데이터를 처리속도의 향상을 위해 실시간으로 뷰어로 전송할 수 있고, 대용량 데이터 처리 시에 메모리 관리의 효율을 위해 파일을 저장소로 사용합니다.

■ 효율적인 리파지토리 관리 기능

서버 리파지토리를 통하여 보고서의 카테고리별 관리 및 생성/수정에 따른 히스토리 기능을 제공하여 개발 편의성 및 효율성을 증대시킵니다.

■ 보고서 스케줄링

매일, 매주 또는 매월 일정 시간에 주기적으로 작성·배포되는 보고서나 대규모 데이터 처리를 수반하는 배치성 보고서에 대하여 생성에서 배포에 이르는 전 과정을 스케줄링하여 처리할 수 있습니다. 스케줄러에 의해서 생성된 보고서는 사전 등록된 사람에게 e-mail로 배포할 수 있습니다. 또한 개발자를 위해 제공된 서버 API를 이용하여 다양한 방식으로 서버 및 스케줄러를 제어할 수 있습니다. 자세한 내용은 "API 활용 매뉴얼"의 서버 API를 참조하시기 바랍니다.

■ 데이터 보안 및 인증

데이터 모듈의 위·변조 방지 기능, 사용자/사용자 그룹별 인증 기능을 제공하여 보다 강화된 보안 기능 구현이 가능합니다.

※ 참고사항 : 데이터 모듈 위변조 방지 기능은 Message Digest 방식(MD5)을 적용하여 데이터 모듈을 생성합니다. 그 외 암호화 키에 의한 데이터 모듈 암호화는 PKI 솔루션과의 연동을 통해 구현할 수 있습니다.

■ 실시간 데이터 Refresh

웹에서 보고서를 조회하고 있는 동안 동일한 보고서를 최신 데이터로 갱신하여 보여줄 수

있습니다. 이러한 데이터 실시간 갱신 기능은 추가 정보, 환율 정보 등 초 단위로 변화하는 자료의 실시간 리포팅이 가능하게 합니다.

■ **최종 사용자가 직접 커스터마이징 가능한 차트**

보고서의 차트 위치, 종류 및 형태를 사용자가 임의로 변경하거나 확대·축소할 수 있으며, 3D 차트의 경우는 자유자재로 회전이 가능합니다.

따라서, 사용자의 서로 다른 취향을 동시에 충족시킬 수 있으며 보다 효과적인 정보전달이 가능합니다.

■ **최종 보고서 편집 기능**

사용자가 완성된 최종 보고서에 새로운 문장을 추가하거나 내용, 형태 등의 변경이 가능하며 그 결과를 사용자 PC에 저장 및 인쇄하고, 메일로 다른 사용자에게도 발송 가능합니다.

※ 참고사항 : 최종 사용자 입력 기능과는 다른 기능입니다.

■ **보고서 간 HyperLink와 Tree View 지원**

서로 연관있는 별개의 보고서를 HyperLink 형태로 호출할 수 있는 기능을 제공하여 Drill-Down 보고서 셋을 구성할 수 있습니다. 구성된 보고서 간의 관계정보는 트리 구조로 제공되어 특정 보고서에 대한 빠른 참조가 가능 합니다.

■ **한 페이지로 모두 보기 지원**

다수의 페이지로 생성된 보고서를 한 페이지로 재구성해 미리보기할 수 있습니다. 하나의 표가 여러 페이지에 걸쳐 가로, 세로 방향으로 출력되는 크로스탭 보고서와 같은 경우 전체 보고서를 한 눈에 볼 수 있어 보고서 가독성을 높여줍니다.

■ **보고서 익스포트 (Export)**

생성된 보고서를 사용자 PC에 OZ Report Data File(*.ozd), Adobe PDF File(*.pdf), Microsoft Excel File(*.xls), Microsoft Word Document(*.doc), Microsoft Presentation(*.ppt), Web Page(*.html), Comma Separated Values File(*.csv), Tab Separated Text(*.txt), JPEG File(*.jpg), Tagged Image Format File(*.tif), Scalable Vector Graphics(*.svg), Hangul File(*.hml) 파일 형식으로 변환하여 저장할 수 있습니다.

■ **다양한 출력 옵션 지원**

출력용지에 맞추어 확대·축소 출력이 가능하며, 한 장의 출력용지에 최대 4페이지까지 모아찍기, 미리보기 없이 직접 프린터로 출력하는 기능 등 다양한 출력 옵션을 지원합니다.

- **뷰어의 동작을 확인할 수 있는 이벤트 기능**

웹상에서 보고서를 조회/인쇄/저장하기 위해 오즈 뷰어에서는 자바 스크립트에서 처리 결과를 받아서 사용할 수 있도록 다양한 이벤트와 결과값을 전달합니다.

- **툴바의 기능을 사용자가 제어 가능**

뷰어 OCX에서 제공하는 기본적인 툴바 기능을 사용자가 자바 스크립트를 통해 제어할 수 있는 기능을 제공합니다. 기본 툴바를 숨기고, 사용자가 만든 버튼/툴바를 이용하여 오즈 뷰어를 제어할 수 있도록 합니다.

- **대용량 데이터 처리 효율과 메모리 관리 기능 개선**

대용량 데이터 모듈의 원활한 처리와 메모리 관리의 효율을 위해 데이터 모듈을 디스크에 저장하는 기능과 전체 보고서 중 일부 페이지만을 메모리에서 관리하는 `pageque` 기능을 제공합니다.

- 일반적으로 특정 사용자 웹 애플리케이션에서 보고서를 호출하기 위해서는 각 보고서마다 오즈 뷰어 호출과 패러미터 전달을 위한 코딩 작업이 필요합니다. 오즈의 웹 배포 마법사는 호출에 필요한 ASP, JSP 또는 HTML 소스를 자동 생성해 줌으로써 보고서를 웹 상에 신속하게 적용할 수 있도록 해줍니다.

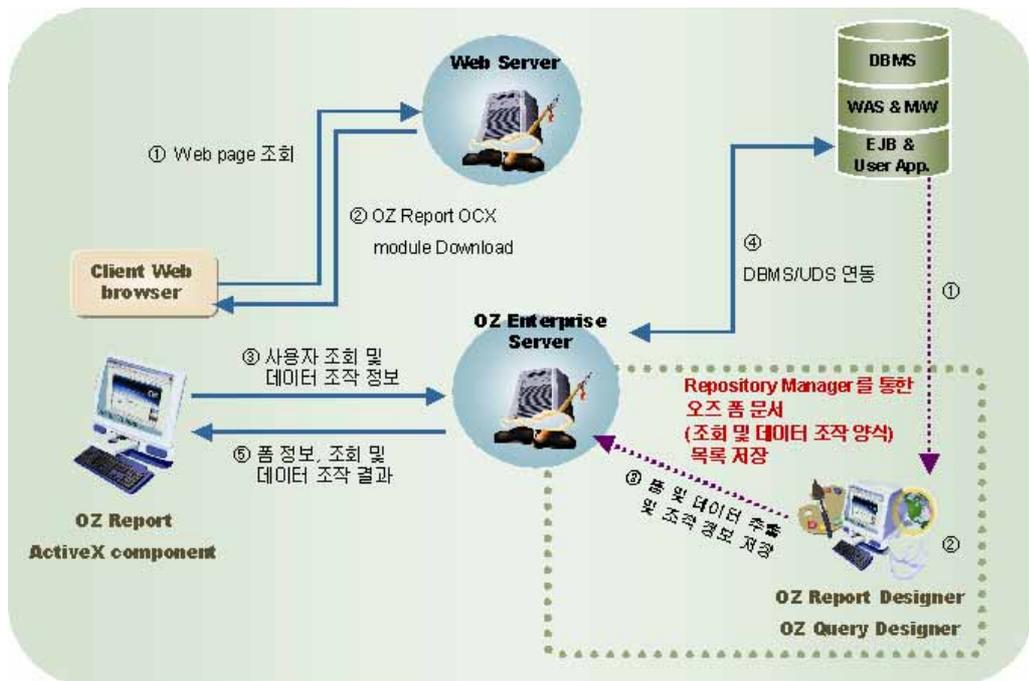
- 일정 규모 이상이거나 불특정 다수의 일반 사용자를 대상으로 서비스를 제공하는 조직에서는 운영 성능 보장이나 시스템 장애에 대비하여 다수의 오즈 서버를 이용한 전사적인 리포팅 환경을 구축합니다.

- 오즈의 엔터프라이즈 매니저는 오즈로 구축된 전사적 리포팅 환경을 종합적 모니터링하고 관리할 수 있는 전용 관리 도구로서 시스템 관리자의 업무 부담을 획기적으로 절감시켜 줍니다.

오즈의 보고서 제작에서 배포에 이르는 전 과정의 운영절차를 개발 환경과 운영 환경으로 나누어 설명합니다.

개발 환경은 보고서 개발자가 오즈 쿼리 디자이너로 ODI 파일을 생성하여 오즈 리포트 디자이너에서 보고서 양식을 디자인하고, 보고서 폼과 ODI 파일을 오즈 리파지토리 매니저를 통하여 오즈 엔터프라이즈 서버의 리파지토리에 저장하는 작업과정을 의미합니다.

운영 환경은 최종 사용자가 웹 애플리케이션을 통해 보고서를 요청하여 보고서가 생성·전달되는 과정을 의미합니다.



- ① 오즈 쿼리 디자이너에서 데이터(DB, 사용자 정의 데이터 등) 정보를 담고 있는 ODI 파일을 생성합니다.
- ② 생성된 ODI 파일을 이용하여 오즈 리포트 디자이너에서는 보고서 양식을 디자인합니다. 완성된 보고서 양식은 오즈의 고유한 OZR 형식으로 저장됩니다.
- ③ 완성된 보고서 양식 파일을 오즈 리파지토리 매니저를 통해 오즈 엔터프라이즈 서버로

업로드 합니다.



- ① 사용자는 웹 브라우저를 이용하여 웹 서버에게 특정 보고서를 요청합니다.
- ② 웹 서버는 보고서를 요청한 클라이언트에게 오즈 리포트 뷰어를 다운로드 해주고, 오즈 리포트 뷰어는 클라이언트 PC에 보안 인증을 거쳐 자동 설치됩니다. 오즈 리포트 뷰어는 최초 한번만 설치되고 오즈 리포트 뷰어의 버전이 변경된 경우에만 자동으로 재설치 됩니다.
- ③ 오즈 리포트 뷰어는 오즈 엔터프라이즈 서버에 보고서 양식과 관련 데이터를 요청합니다.
- ④ 오즈 엔터프라이즈 서버는 JDBC/ODBC(DBMS)나 UDS(User Data Store)를 통해 보고서 데이터 셋을 생성합니다.
- ⑤ 오즈 리포트 뷰어에 보고서 양식과 데이터 셋을 전송합니다.
- ⑥ 오즈 리포트 뷰어는 전송 받은 보고서 양식과 데이터 셋을 결합하여 보고서를 생성하고 미리보기, 인쇄 또는 파일저장 등을 가능하게 합니다.

■ 운영 OS

JRE(Java Runtime Environment) 1.2 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(Web Application Server)

WebLogic, WebSphere, JBoss, 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



■ 지원 OS

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

■ 권장 하드웨어 사양

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

■ 지원 OS

MS Windows 98 이상

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

■ 권장 하드웨어 사양

CPU : Pentium III 500MHz 이상 권장

RAM : 64MB(128MB 이상 권장)

모니터 : 컬러 SVGA(800*600 이상의 해상도 권장)

프린터 : Windows Driver가 지원되는 프린터

■ 지원 OS

JRE(Java Runtime Environment) 1.3 이상이 지원되는 Windows NT 계열 플랫폼(Windows NT/2000/XP/2003 이상)

※ JRE 1.4이상은 별도의 제품을 제공합니다.



■ 지원 OS

MS Windows 플랫폼(Windows NT/2000/XP/2003 이상)

■ 권장 하드웨어 사양

CPU : Pentium III 500MHz 이상 권장

RAM : 128MB(256MB 이상 권장)

모니터 : 컬러 XGA(1024*768 이상의 해상도 권장)

. Quick Tour



- Step1. ODI
- Step2.
- Step3.



- Step1.
- Step2. ODI
- Step3.
- Step4.

- 윈도우 플랫폼(MS Windows NT 4.0 이상)
- 웹 서버로 인터넷 인포메이션 서버(IIS)가 설치되어 있어야 합니다.
- 오즈 엔터프라이즈 서버(데몬 타입)와 오즈 리포트 디자이너, 오즈 쿼리 디자이너, 오즈 리파지토리 매니저가 설치되어 있어야 합니다.
- 샘플 데이터베이스가 MS Access DB 파일로 제공되므로 MS Access ODBC Driver가 설치되어 있어야 합니다.
- 웹에서 보고서 서비스를 위해서 IIS 웹 서버가 구동되어 있어야 하며, 뷰어 설치 태그에서 뷰어 다운로드로 지정된 폴더에 오즈 리포트 뷰어 설치 파일이 존재하여야 합니다.



보고서를 제작하기 위해서는 우선 보고서 목적을 이해하고 보고서에 이용할 데이터 소스의 구조에 대해 충분히 숙지하고 있어야 합니다. 이후 제작하고자 하는 보고서 품에 대한 구상을 한 후 실제 디자인 작업에 들어갑니다.

그럼, 실례로 다음과 같은 보고서를 만들어 보도록 하겠습니다.



상기 보고서는 다음과 같이 크게 타이틀 부분, 크로스탭 통계표, 현황 차트, 그리고 풋터 부분으로 나눌 수 있습니다.

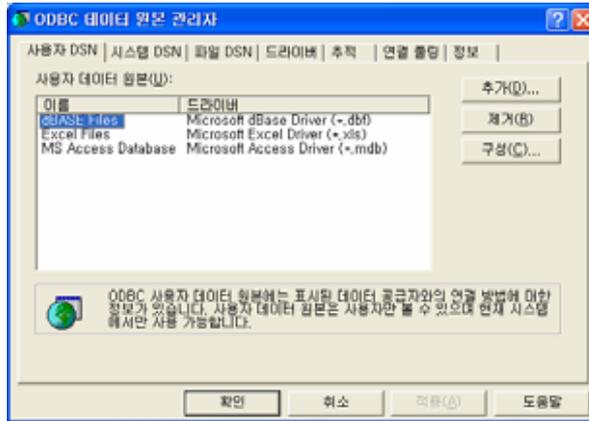
Step 1 ODI

◆ **ODBC 데이터 원본 설정**

우선 샘플 데이터베이스인 ozsample.mdb 파일을 ODBC 관리자를 통해 ODBC 데이터 소스로 등록합니다.

이용할 Microsoft Access DB 파일은 ozsample.mdb이며, DB 파일은 고객 지원 센터로 문의하시기 바랍니다. 이 ozsample.mdb를 ODBC 시스템 데이터 원본으로 등록하고 데이터 원본 이름(DSN)은 "ozsample"으로 설정합니다.

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

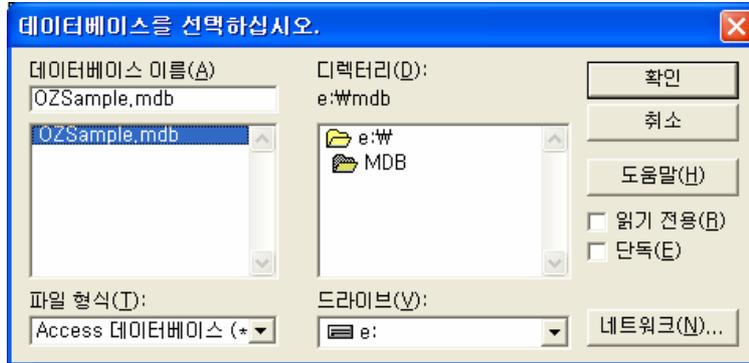


- 2) [시스템 DSN] 탭을 클릭하고 [추가] 버튼을 클릭한 후 Microsoft Access Driver(*.mdb)를 선택하고 [마침] 버튼을 클릭합니다.

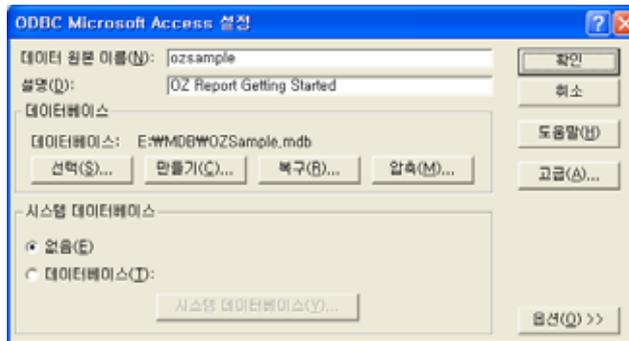


- 3) 데이터 원본 이름을 'ozsample'로 입력하고, [선택] 버튼을 클릭하여

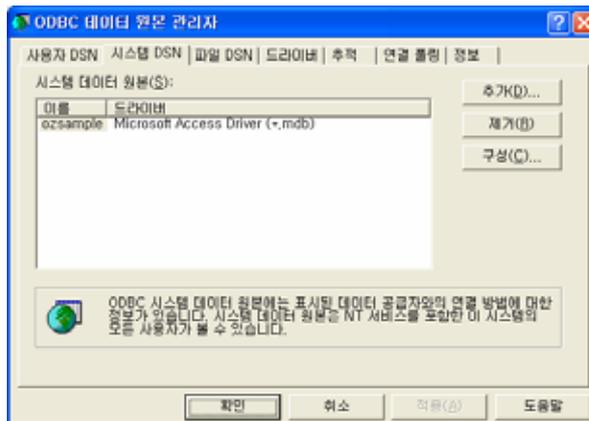
'ozsample.mdb' 파일을 선택합니다.



4) 설명 등을 추가한 후에 [확인] 버튼을 클릭합니다.



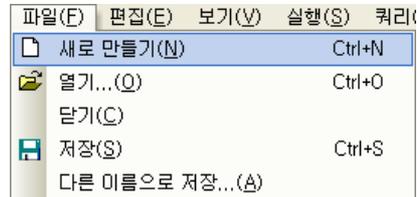
5) 'ozsample'라는 데이터 원본이 추가된 것을 확인합니다.



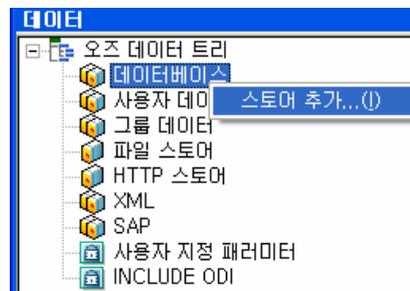
◆ **Database와의 연동**

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

- 1) [파일] → [새로 만들기]를 클릭합니다.

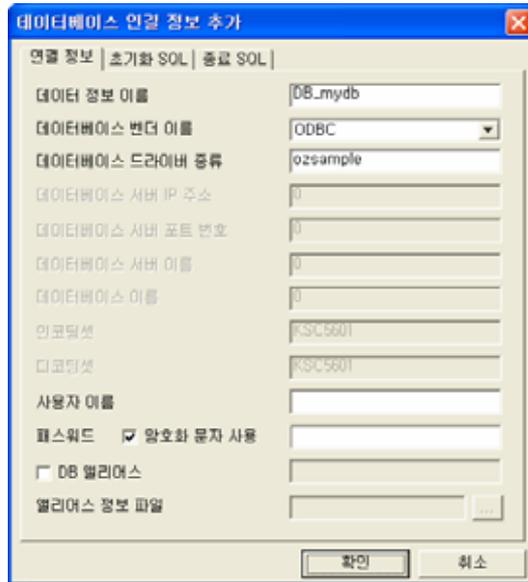


- 2) 새로운 ODI 파일 생성을 위해 데이터 트리의 [데이터베이스] 항목에서 마우스 오른쪽 버튼을 클릭하여 나타나는 팝업 메뉴에서 [스토어 추가]를 클릭합니다.



- 3) [데이터 정보 이름]은 사용자가 임의로 정할 수 있으며 본 예제에서는 'DB_mydb'로 입력합니다.

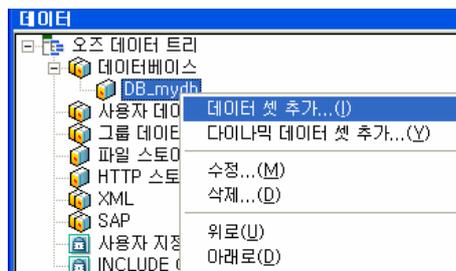
[데이터베이스 벤더 이름]에 ODBC를 선택하고, [데이터베이스 드라이버 종류]에는 ozsample.mdb에 대한 DSN 이름인 'ozsample'을 입력한 후 [확인] 버튼을 클릭합니다.



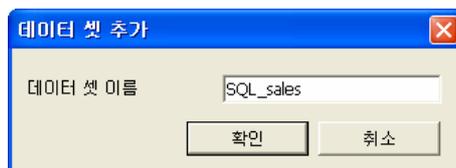
◆ 데이터 셋 추가

- 1) 데이터베이스가 연결되면 [데이터베이스] 항목 아래에 [DB_mydb] 항목이 추가됩니다.

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

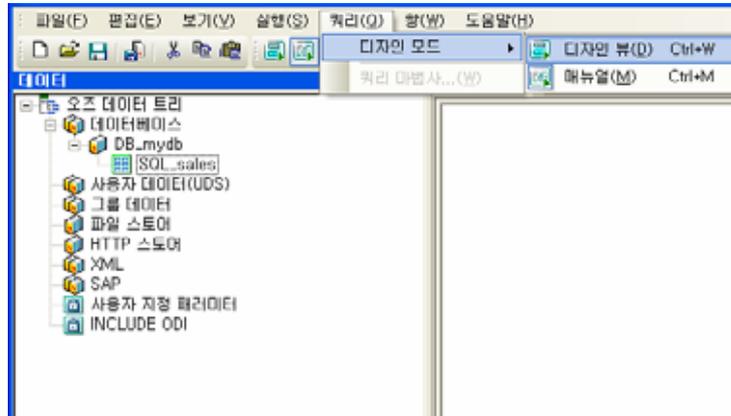


- 2) [데이터 셋 이름]으로는 'SQL_sales'를 입력하고 [확인] 버튼을 클릭합니다.



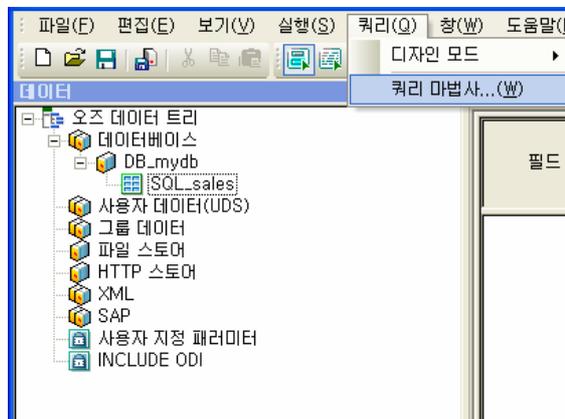
- 3) 새로운 쿼리 데이터 셋이 정상적으로 등록되었다면 디자인 뷰 모드에서 손쉽게 쿼리를 작성하기 위해 왼쪽 그림과 같이 [쿼리] 메뉴에서 [디자인 모드] →

[디자인 뷰]를 클릭합니다.



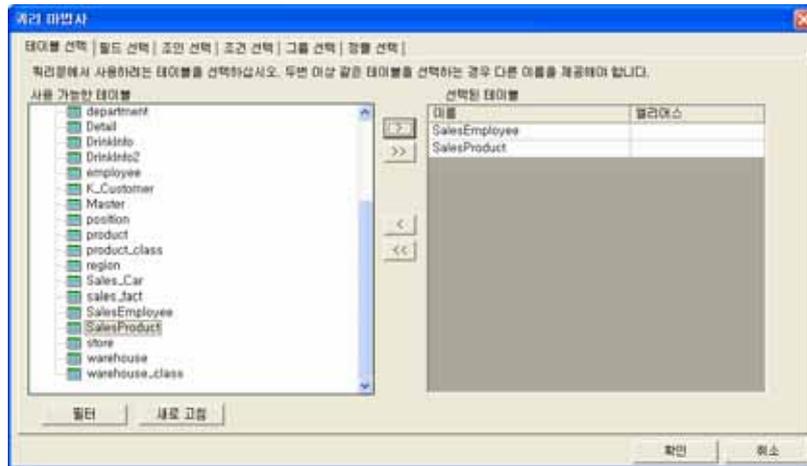
※ [파일] 메뉴의 [옵션]에서 [디자인 모드]를 '디자인 뷰'로 설정해 놓은 경우에는 오즈 쿼리 디자이너 실행시 '디자인 뷰' 모드로 적용되어 있으므로 따로 변경할 필요가 없습니다.

- 4) 화면이 '디자인 뷰' 모드로 전환되었다면 본격적으로 쿼리를 작성하기 위해 왼쪽의 화면과 같이 [쿼리] 메뉴에서 [쿼리 마법사] 메뉴를 클릭하거나 쿼리 프레임에 마우스 오른쪽 버튼을 클릭하여 나타나는 팝업 메뉴에서 다음 화면과 같이 [쿼리 마법사]를 실행합니다.



◆ 쿼리 마법사를 이용한 쿼리 생성

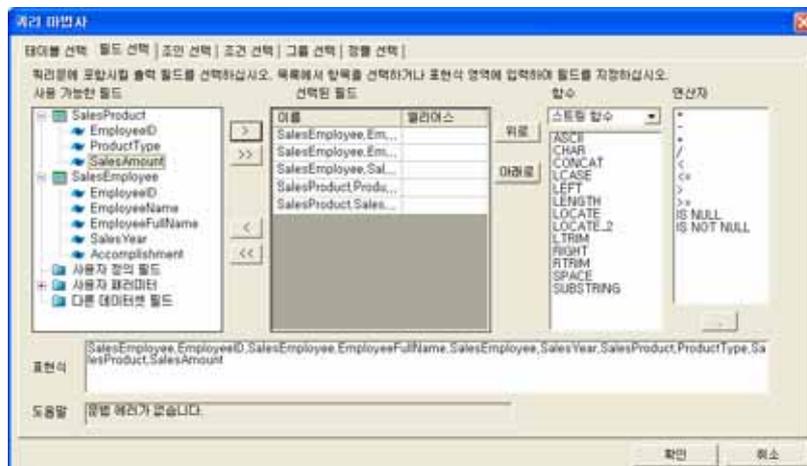
- 1) 쿼리 마법사가 실행되면 [테이블 선택] 탭의 [사용 가능한 테이블]에서 'SalesProduct' 테이블과 'SalesEmployee' 테이블을 선택한 후 [>] 버튼을 클릭하여 [선택된 테이블]로 이동시킵니다.



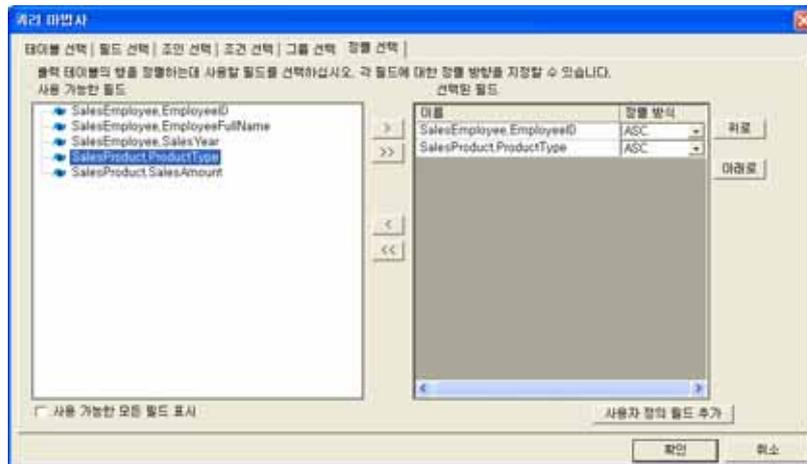
- 2) 테이블을 선택한 뒤 [필드 선택] 탭에서 실제 보고서에 출력 및 조회하고자 하는 필드를 선택합니다.

SalesEmployee.EmployeeID, SalesEmployee.EmployeeFullName,
SalesEmployee.SalesYear, SalesProduct.ProductType,

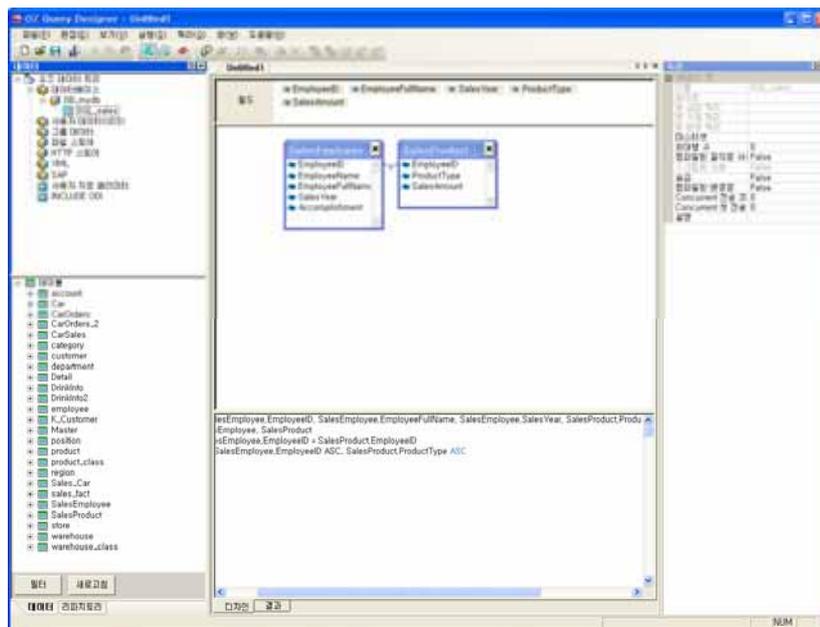
SalesProduct.SalesAmount를 [>] 버튼을 이용하여 [선택된 필드]로 이동시킵니다.



- 3) 선택한 테이블간의 조인을 선택하기 위해 [조인 선택] 탭을 클릭하면 왼쪽 화면과 같이 자동으로 기본 조인 조건을 찾아주는 것을 확인할 수 있습니다.

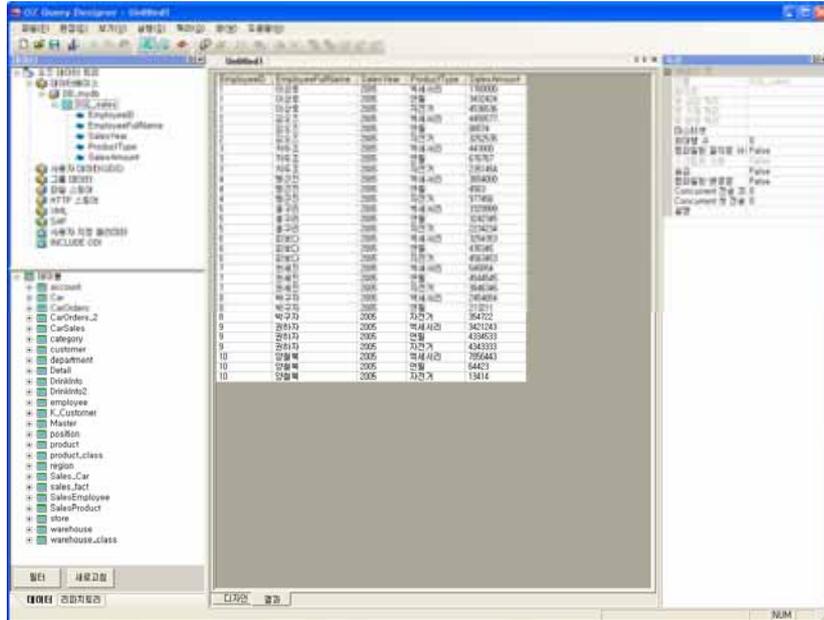


- 6) 모든 작업을 완료한 뒤 [확인] 버튼을 클릭하면 다음과 같은 화면이 표시되고 쿼리 프레임 하단에는 자동으로 생성된 쿼리문이 표시됩니다.



쿼리 프레임 상단 창에는 쿼리문으로 조회할 필드 목록이 표시되고, 가운데 창에는 테이블간의 조인 관계가 표시되며, 하단 창에는 완성된 SQL문 전체가 표시됩니다.

- 7) 화면 상단의 툴바에서 쿼리문 실행 아이콘()을 클릭하거나 <F5> 키를 누르면 다음 화면과 같이 쿼리 마법사에서 완성된 SQL문을 실행하여 결과를 조회할 수 있습니다.

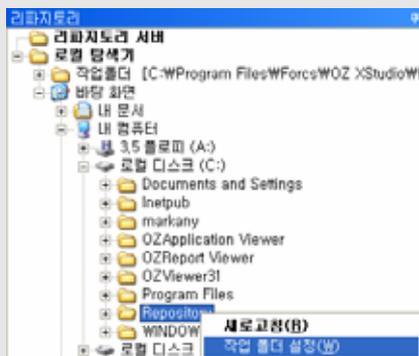


정상적으로 조회 결과가 표시된다면 [파일] 메뉴의 [저장]을 클릭하거나 <Ctrl + S> 키를 눌러 지금까지 생성한 DB 정보와 데이터 셋 정보를 'mydb.odi'라는 이름으로 작업 폴더에 저장합니다.

<작업 폴더 설정 방법>

작업 폴더란 보고서에서 ODI나 이미지 파일 등을 참조할 때 기본이 되는 폴더 경로를 말합니다.

작업 폴더는 리포트 디자이너의 리파지토리창에서 원하는 폴더를 선택한 후 마우스 오른쪽 버튼을 클릭하여 나타나는 팝업 메뉴에서 [작업 폴더 설정] 메뉴를 클릭하여 설정합니다.



설정된 작업 폴더는 리파지토리창의 로컬 탐색기의 작업 폴더 옆에 나타나있는 경로를 참조하여 확인할 수 있습니다.

다음으로 "Step 2. 보고서 디자인하기"에서는 지금까지 작성된 ODI 파일을 이용하여 본격적인 보고서 디자인을 시작합니다.

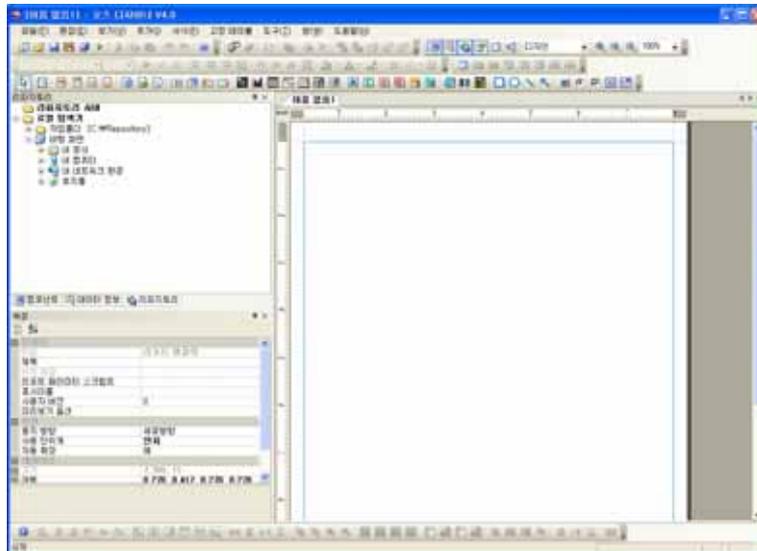
Step 2

Step 1.에서 작성한 ODI 파일은 보고서를 디자인 할 때 사용할 데이터베이스 연결정보와 쿼리 데이터 셋을 포함하고 있습니다. 그러면 지금부터 이 ODI 파일을 이용한 보고서 디자인 과정에 대해 설명합니다.

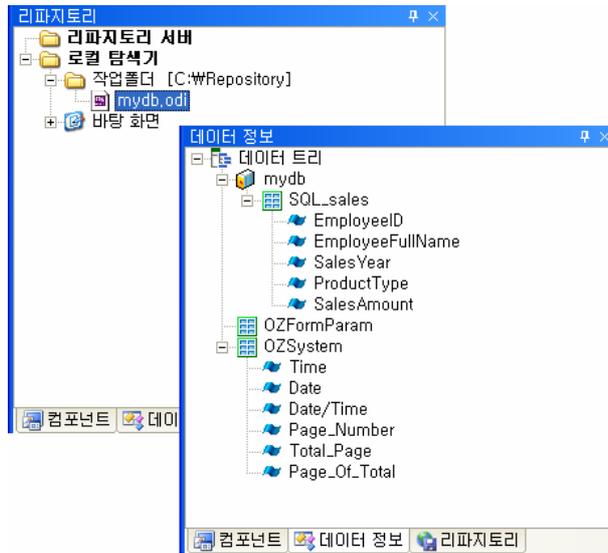
◆ **보고서 품 만들기**

작성한 ODI 파일(ozsample.odi)을 추가하고 보고서 품을 만듭니다.

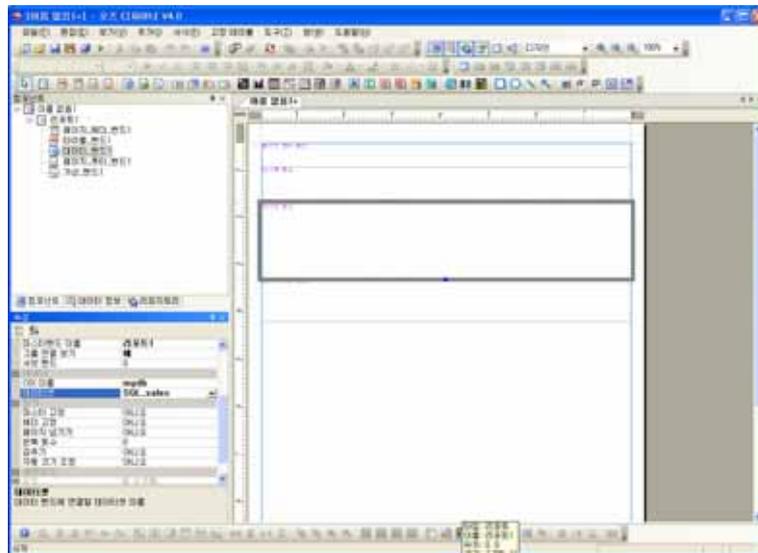
- 1) 오즈 리포트 디자이너를 실행한 후 메뉴바에서 [파일] → [새로 만들기] 메뉴 또는 툴바에서 새로 만들기 아이콘()을 클릭하여 보고서를 새로 만들기 합니다.



- 2) 리파지토리창에서 작업 폴더를 클릭하여 작성한 'mydb.odi' 파일을 더블 클릭하여 보고서에 추가합니다. 추가된 ODI 파일 정보는 데이터 정보창에서 확인할 수 있습니다.



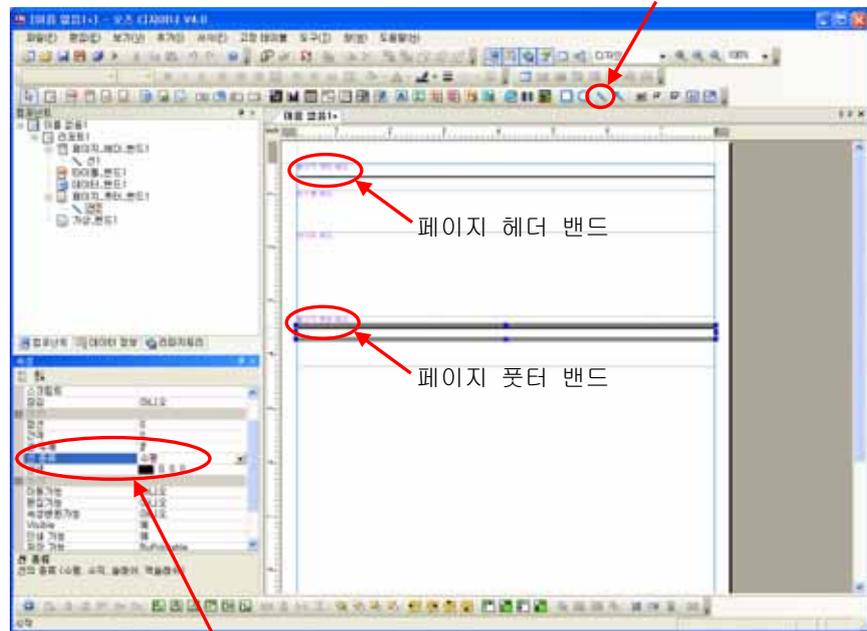
- 3) 보고서에 페이지 헤더 밴드, 타이틀 밴드, 데이터 밴드, 페이지 풋터 밴드를 추가하고, 데이터 밴드의 'ODI 이름' 속성값을 'mydb'로, '데이터 셋' 속성값은 'SQL_sales'로 설정합니다.



◆ 페이지 헤더 및 페이지 풋터 디자인

아래 그림과 같이 선 컴포넌트를 이용하여 페이지의 상단과 하단에 표시될 선을 디자인합니다.

선 컴포넌트



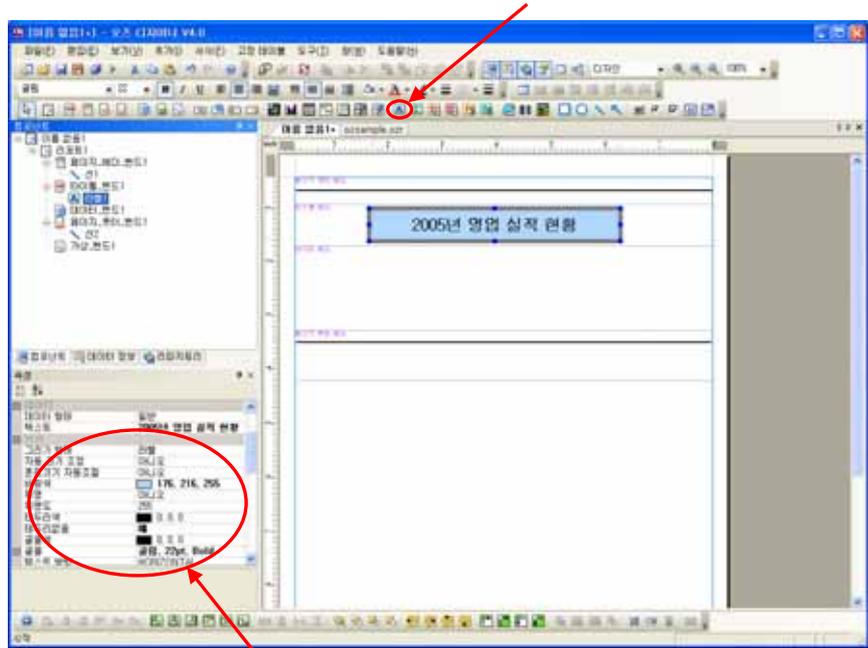
선 컴포넌트 속성 변경

툴바에서 선 컴포넌트 아이콘()을 클릭한 후 페이지 헤더 밴드와 페이지 풋터 밴드에 선을 그릴 만큼 마우스로 영역을 지정하여 추가합니다.

추가된 선 컴포넌트를 선택하여 속성창에서 '선 두께', '선 종류' 등을 적절히 설정하여 선을 디자인합니다.

◆ 보고서에 제목 달기

타이를 밴드에 라벨을 추가하여 보고서 제목으로 이용합니다.



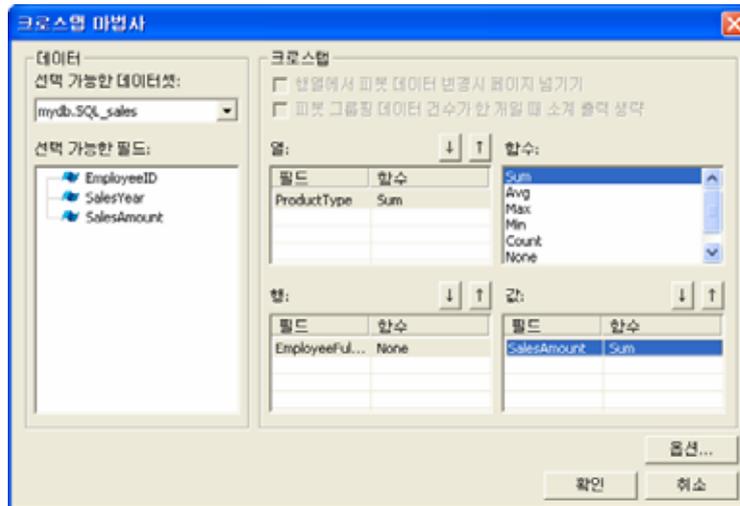
라벨 속성 변경

툴바에서 라벨 컴포넌트 아이콘()을 클릭한 후 타이틀 밴드에 라벨을 그릴 만큼 마우스로 영역을 지정하여 추가합니다.

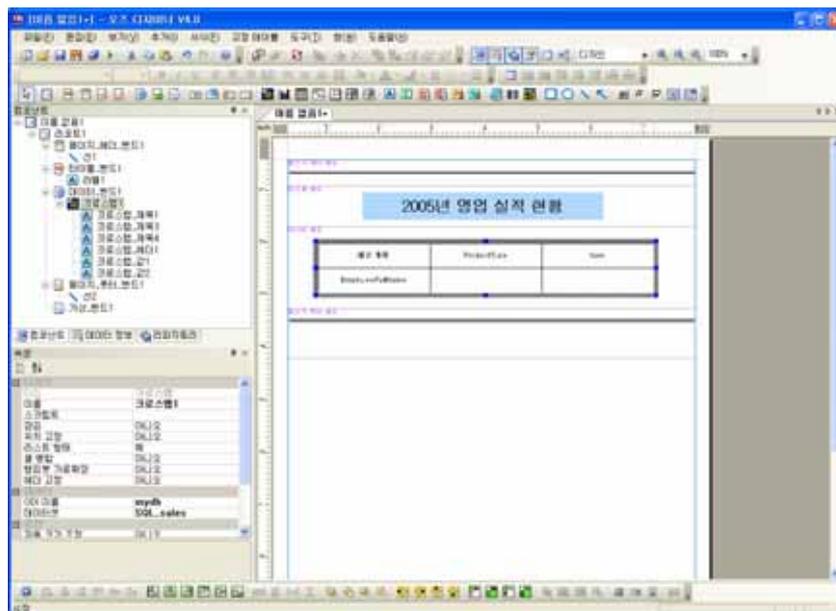
추가된 라벨 컴포넌트를 선택하여 속성창에서 '텍스트', '바탕색', '테두리없음', '글꼴' 등을 적절히 설정하여 라벨을 디자인합니다.

◆ 크로스탭 통계표 만들기

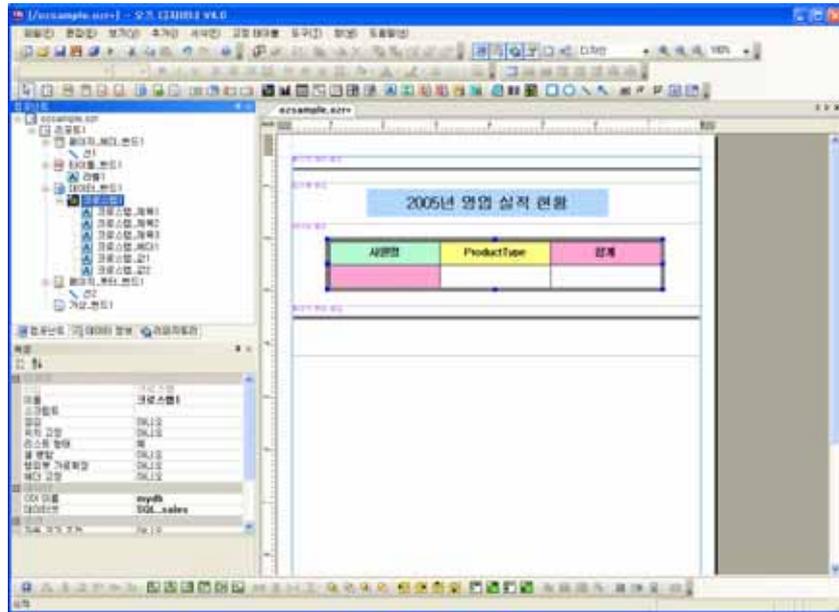
- 1) 툴바에서 크로스탭 컴포넌트 아이콘()을 클릭한 후 데이터 밴드에 마우스로 그릴 영역을 지정하면 크로스탭 마법사가 자동으로 실행됩니다.
'선택 가능한 필드'에서 필드를 선택한 후 크로스탭의 행, 열, 값으로 드래그 & 드롭하여 아래 그림과 같이 설정합니다.



- 2) 행, 열, 값을 설정한 후 [확인] 버튼을 클릭하면 아래 그림과 같이 크로스탭이 추가됩니다.

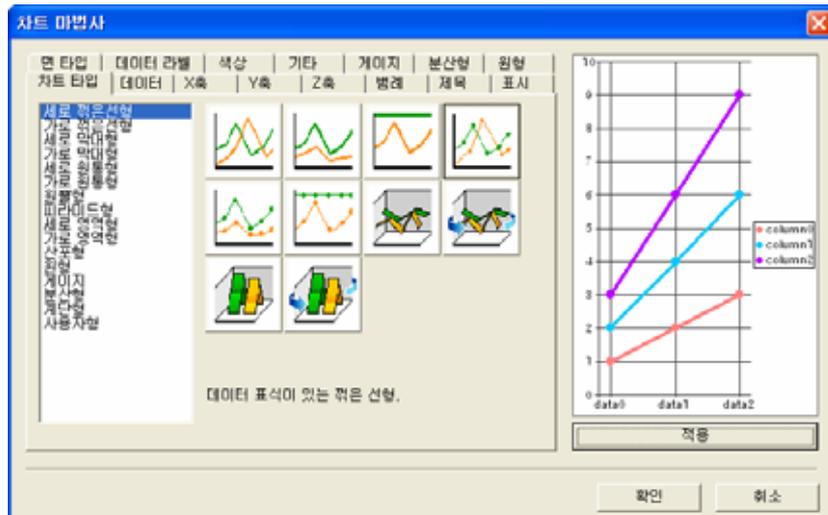


- 3) 크로스탭의 셀을 선택하여 '글꼴', '바탕색', '자동 줄바꾸기', '글자 정렬', '표시 형식' 등을 적절히 설정하여, 크로스탭을 디자인합니다.



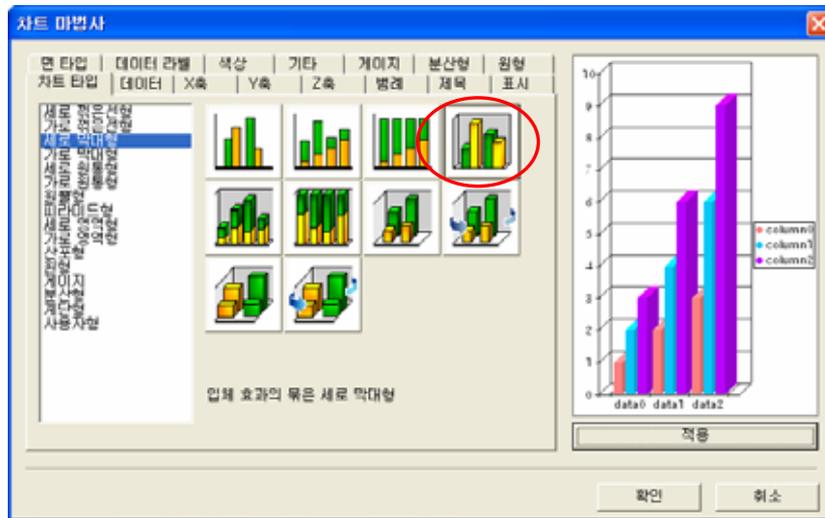
◆ 차트 만들기

- 1) 크로스탭이 추가된 데이터 밴드 아래에 새로운 데이터 밴드를 추가한 후 데이터 밴드의 'ODI 이름' 속성값을 'mydb'로, '데이터 셋' 속성값은 'SQL_sales'로 설정합니다.
- 2) 툴바에서 차트 컴포넌트 아이콘()을 클릭한 후 데이터 밴드에 마우스로 그릴 영역을 지정하면 차트 마법사가 자동으로 실행됩니다.

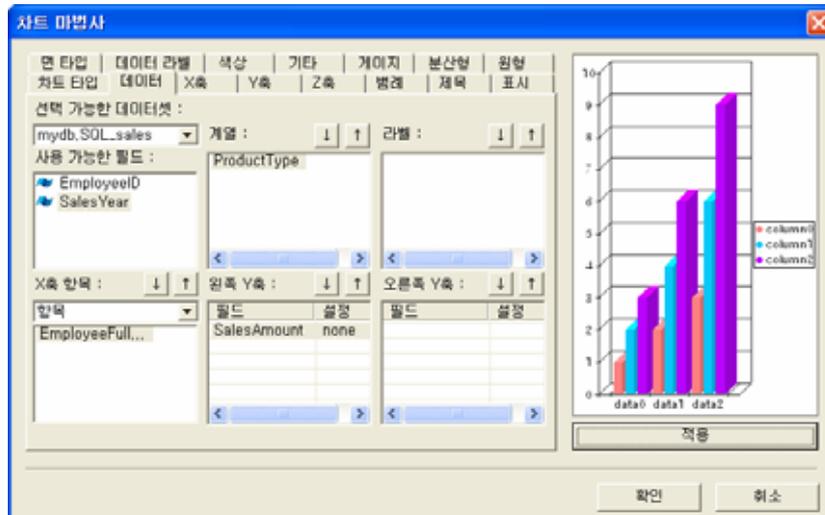


- 3) 차트의 종류를 '세로 막대형'에서 '입체 효과의 묶은 세로 막대형'을 선택합니다

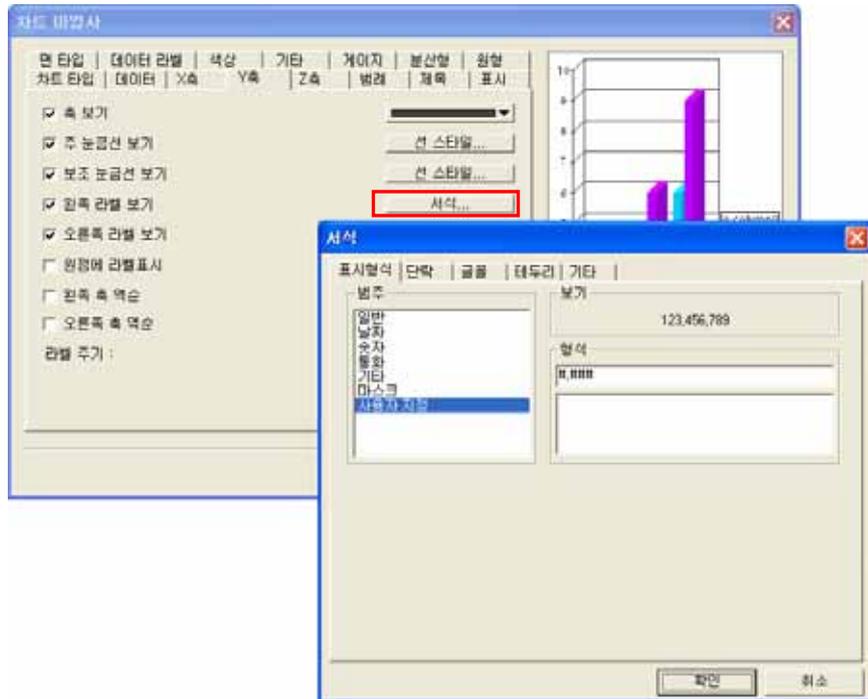
다.



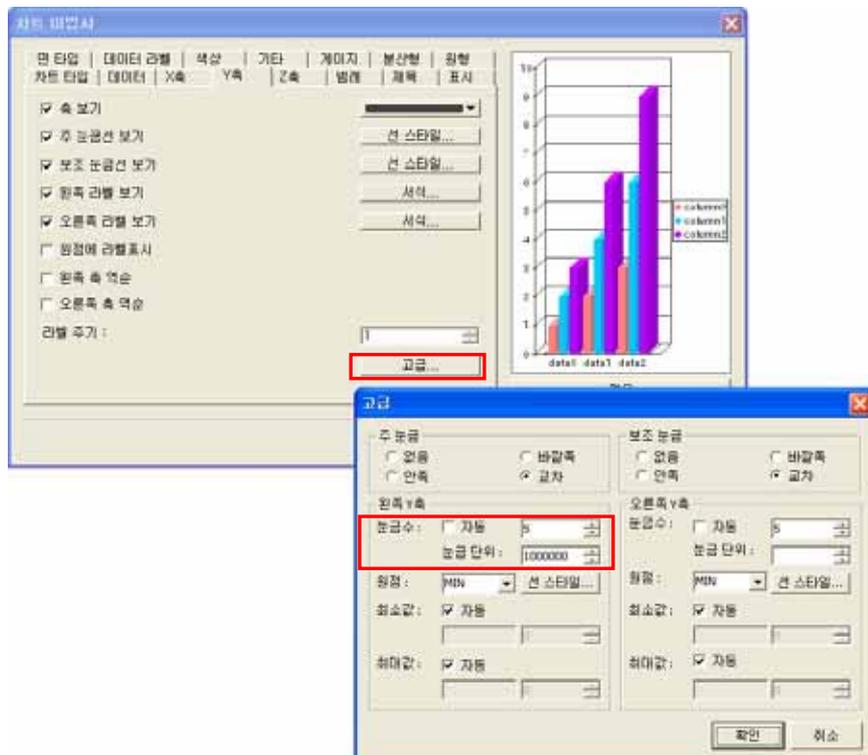
- 4) [데이터] 탭을 클릭하면 차트의 X축, Y축, 계열 등을 설정할 수 있습니다. '사용 가능한 필드'에서 필드를 선택한 후 X축 또는 Y축에 드래그 & 드롭하여 설정합니다.



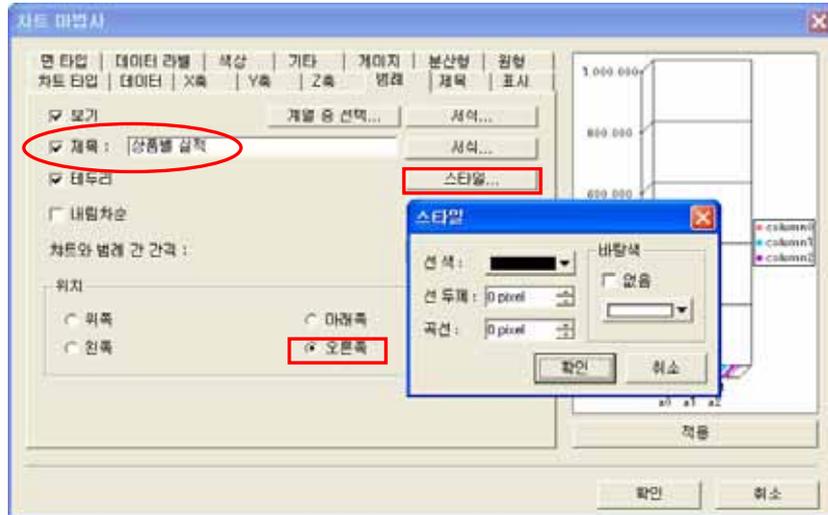
- 5) 아래 그림과 같이 설정한 후 [Y축] 탭을 클릭합니다.
 [Y축] 탭을 선택한 후 '왼쪽 라벨 보기' 옆의 [서식...] 버튼을 클릭하여 나타나는 서식 다이얼로그에서 서식을 '사용자 지정'으로 선택하고 '#,###'으로 입력한 후 [확인] 버튼을 클릭합니다.



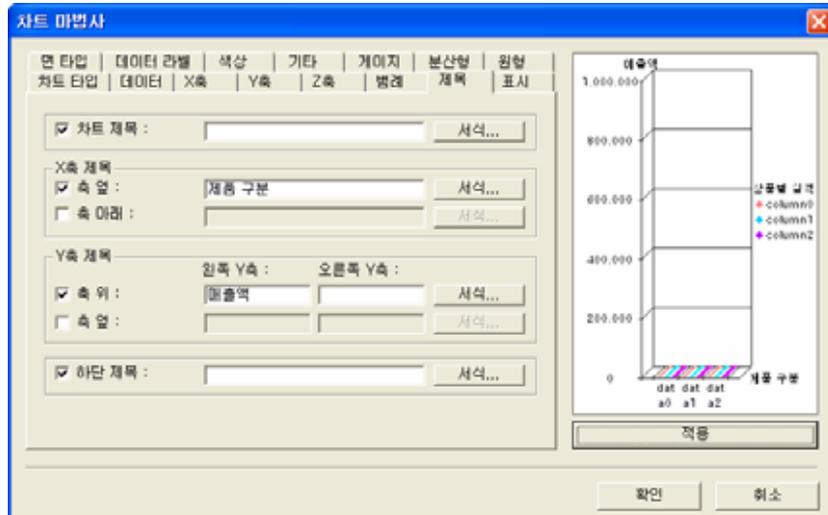
- 6) [Y축] 탭의 [고급...] 버튼을 클릭하여 나타나는 고급 옵션 설정 다이얼로그에서 왼쪽의 '눈금 단위'를 '1000000'으로, '눈금수'를 '5'로 설정한 후 [확인] 버튼을 클릭합니다.



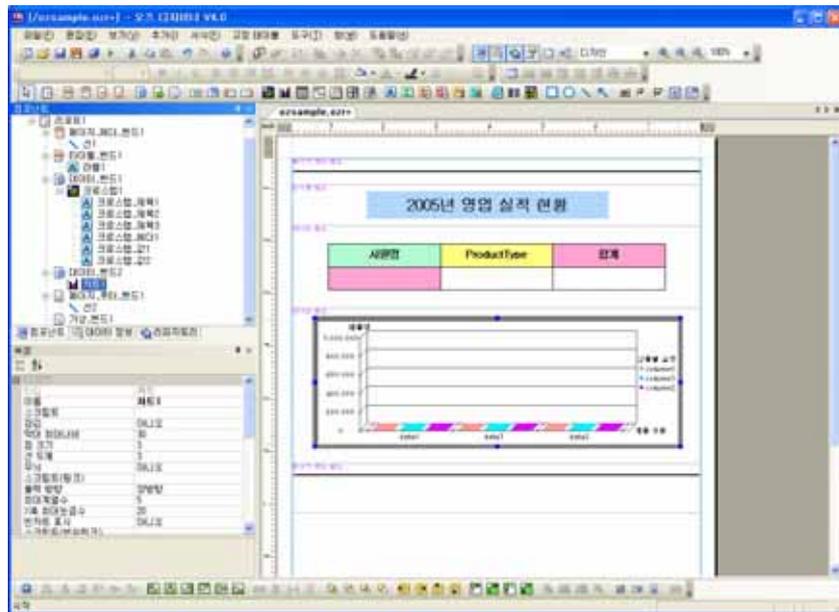
- 7) [범례] 탭을 선택하여 '제목'을 '상품별 실적'으로 입력하고, '위치'를 '오른쪽'으로 선택한 후 [스타일] 버튼을 클릭하여 스타일 다이얼로그를 띄웁니다. 사각형 스타일 다이얼로그에서 '선 두께'를 '0'으로 설정하고 '바탕색'을 원하는 색으로 선택한 후 [확인] 버튼을 클릭합니다.



- 8) [제목] 탭에서 X축의 제목을 '제품 구분'으로, Y축의 제목을 '매출액'으로 설정합니다.

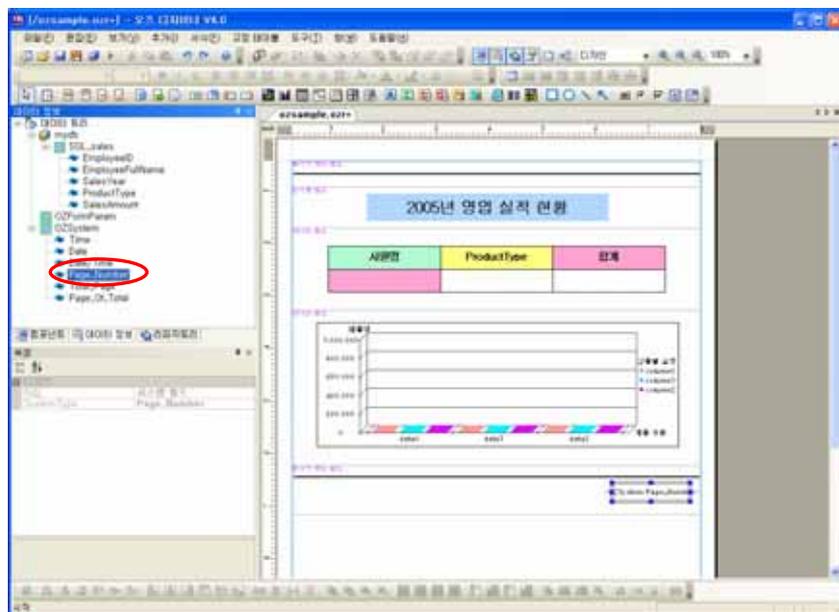


- 9) 차트 마법사에서 [확인] 버튼을 클릭하면 보고서에 차트가 추가됩니다.



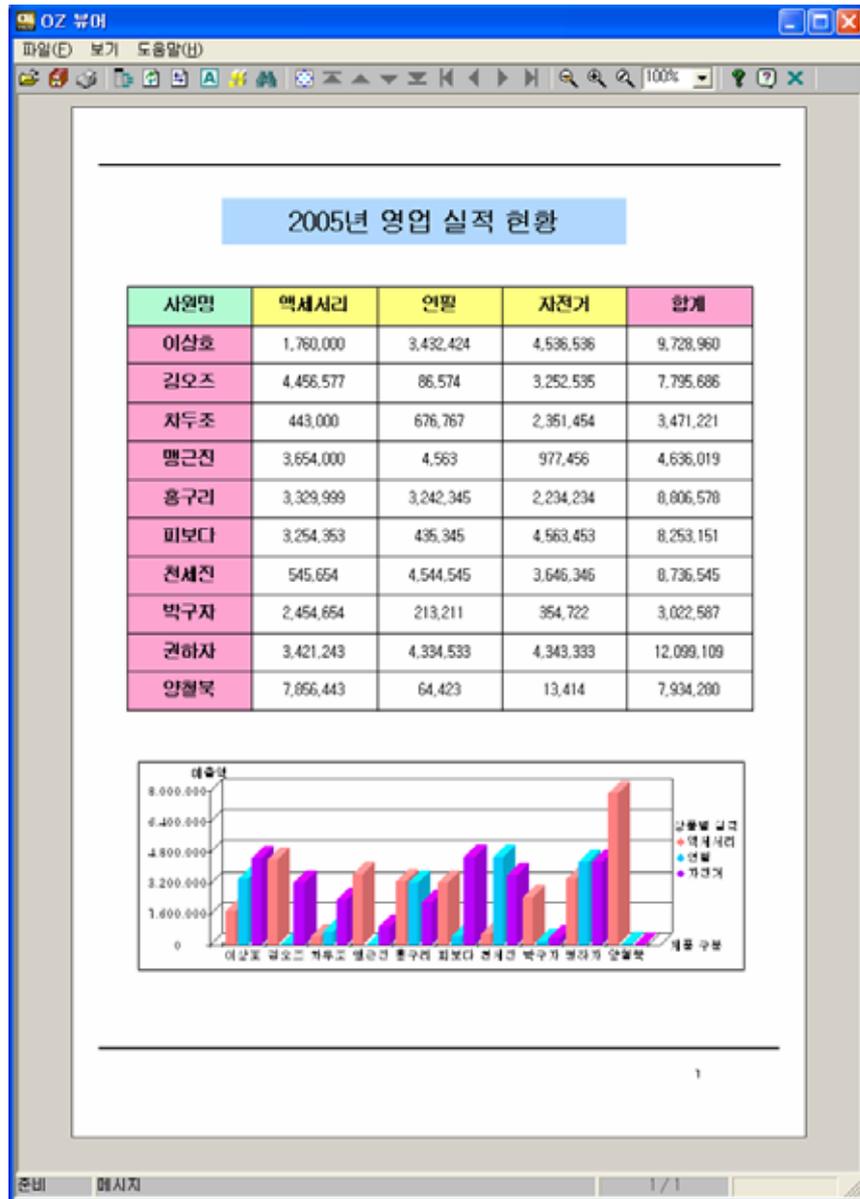
◆ 페이지 풋터에 페이지 번호 넣기

데이터 정보창에서 'Page_Number'를 선택하여 페이지 풋터 밴드의 오른쪽 하단에 드래그 & 드롭하여 추가합니다.



Step 3

툴바의 미리보기 아이콘()을 클릭합니다. 미리보기시에 나타난 화면은 실제 웹을 통해 배포되는 것과 동일합니다.



정상적으로 보고서가 보여진다면 [파일] 메뉴의 [저장] 메뉴를 클릭하거나 <Ctrl + S> 키를 눌러 리파지토리 루트 폴더에 "ozsample.ozr"이라는 이름으로 저장합니다.

Step 1

◆ 데이터베이스 정보 추가

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

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

◆ 오즈 서버 실행

오즈 서버의 설치에 대한 자세한 내용은 "설치 가이드"를 참고하시기 바라며 여기에서는 C:\OZServer 폴더에 설치되어 있다고 가정합니다.

오즈 서버가 정상적으로 설치되어 있다면 C:\OZServer\bin\ozstart.bat를 Command Line에서 실행시켜 오즈 서버를 실행합니다.

※ 오즈 서버를 실행시키기 위한 JRE 버전으로는 1.3 이상을 권장하며, 윈도우즈 환경에서 오즈 서버를 운영할 때에는 오즈 서버와 함께 제공되는 jre 폴더의 JRE를 사용하는 것이 좋습니다.

자세한 내용은 "오즈 엔터프라이즈 서버 관리자 매뉴얼"을 참고하시기 바랍니다.

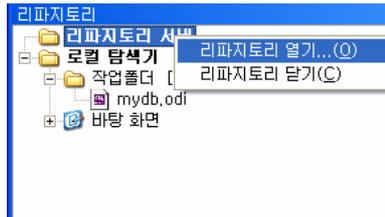
Step 2

ODI

◆ 오즈 리파지토리 클라이언트 사용하기

오즈 리포트 디자이너에 내장된 리파지토리 클라이언트 기능을 이용하여 보고서 품 및 ODI 파일을 오즈 서버로 전송합니다.

- 1) 리포트 디자이너의 리파지토리창에서 '리파지토리 서버'를 마우스 오른쪽 버튼을 클릭하여 나타나는 팝업 메뉴에서 [리파지토리 열기] 메뉴를 클릭합니다.

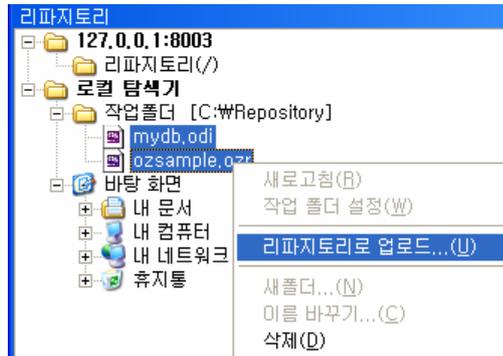


- 2) '서버리스트' (Server List) 다이얼로그에서 [서버 추가] (Add Server) 버튼을 클릭하여 접속할 서버 정보를 아래 그림과 같이 입력하고 [저장 후 접속] (Save and Connect) 버튼을 클릭합니다. (비밀번호는 기본값인 'admin' 입니다.)

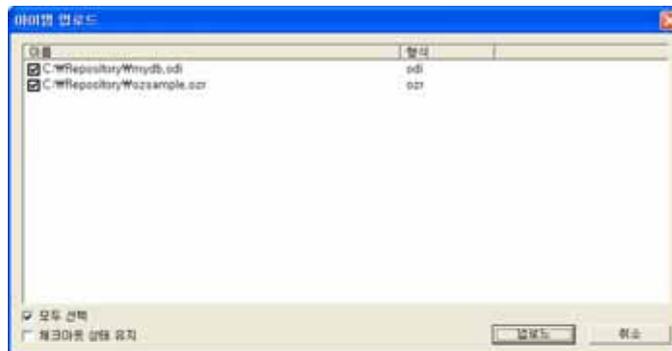


'서버리스트' 다이얼로그에 대한 자세한 내용은 "오즈 리포트 디자이너 사용자 매뉴얼"의 서버 리파지토리 사용법을 참조하시기 바랍니다.

- 3) 작업 폴더에서 업로드할 OZR과 ODI 파일을 마우스 오른쪽 버튼을 클릭하여 나타나는 팝업 메뉴에서 [리파지토리로 업로드] (Upload to Repository) 메뉴를 클릭합니다.



4) 아이템 업로드 다이얼로그에서 업로드할 파일을 체크한 후 [업로드] 버튼을 클릭하여 업로드합니다.



Step 3

◆ 리포트 뷰어 설치 태그 작성

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

- ztransferx.js

```
var ZTransferXObjectElement = document.createElement('<OBJECT width =
"0" height = "0" ID="ZTransferX" CLASSID="CLSID:C7C7225A-9476-47AC-
B0B0-FF3B79D55E67"
codebase="http://127.0.0.1/OZReportViewer/ZTransferX.cab#version=2,1,0,
2"></OBJECT>');
var ZTransferXParamElement1 = document.createElement('<PARAM
NAME="download.Server" VALUE="http://127.0.0.1/OZReportViewer/">');
var ZTransferXParamElement2 = document.createElement('<PARAM
NAME="download.Port" VALUE="80">');
var ZTransferXParamElement3 = document.createElement('<PARAM
NAME="download.Instruction" VALUE="ozrviewer.idf">');
var ZTransferXParamElement4 = document.createElement('<PARAM
NAME="install.Base" VALUE="<PROGRAMS>/Forcs">');
var ZTransferXParamElement5 = document.createElement('<PARAM
NAME="install.Namespace" VALUE="CustomNameSpace">');
ZTransferXObjectElement.appendChild(ZTransferXParamElement1);
ZTransferXObjectElement.appendChild(ZTransferXParamElement2);
ZTransferXObjectElement.appendChild(ZTransferXParamElement3);
ZTransferXObjectElement.appendChild(ZTransferXParamElement4);
ZTransferXObjectElement.appendChild(ZTransferXParamElement5);
OZEmbedControlLocation.appendChild(ZTransferXObjectElement);
```

※ 주의사항

웹에서 리포트 서비스를 위해서 IIS 웹 서버가 구동되고 있어야 하며, IIS의 루트 폴더 하위에 다운로드로 설정한 폴더가 있고 그 폴더에 오즈 리포트 뷰어 설치 파일이 존재하여야 합니다. (위 그림처럼 설정한 경우에는 IIS 루트 폴더 → 'OZReportViewer' 폴더임)

◆ 리포트 파일 호출 태그 작성

웹페이지 제작용 툴이나 메모장 등을 실행하여 리포트 파일 호출 태그를 다음과

같이 작성합니다.

- sample.html

```
<div id="OZEmbedControlLocation">
<script id="ZTransferX" src="ztransferx.js"></script>
<script LANGUAGE="Javascript">
var tag = '<OBJECT id = "OZReportViewer" CLASSID="CLSID:0DEF32F8-
170F-46f8-B1FF-4BF7443F5F25" width="800" height="100%"></OBJECT>';
var paramTag = new Array();
paramTag[paramTag.length] = '<param name="connection.server"
value="127.0.0.1">';
paramTag[paramTag.length] = '<param name="connection.port"
value="8003">';
paramTag[paramTag.length] = '<param name="connection.reportname"
value="/ozsample.ozr">';
paramTag[paramTag.length] = '<param name="viewer.namespace"
value="CustomNameSpace\ozviewer">';
paramTag[paramTag.length] = '<param name="viewer.isframe"
value="false">';
paramTag[paramTag.length] = '<param name="odi.odinames"
value="mydb">';
oz_activex_build(OZEmbedControlLocation, tag, paramTag);
</script>
</div>
```

'connection.reportname'에는 리포트 파일의 이름을 설정합니다. 본 예제에서는 'ozsample.ozr'입니다.

◆ HTML 작성

리포트 파일 호출 태그를 HTML 문서 규격에 맞게 삽입하여 작성한 ozsample.html 파일을 저장합니다.

```

appsample.html - 메모장
[편집] [문자집] [서식] [보기] [도움말]
<html>
<head>
<title>oz Application Sample</title>
<script src="oz_activex.js"></script>
</head>
<body>
<div id="BZEmbedControlLocation">
<script id="ZtransferX" src="ztransferx.js"></script>
<script LANGUAGE="JavaScript">
var tag = {OBJECT id = "BZApplicationViewer" CLASSID="CLSID:9B7A88F3-7398-4672-9318-368596587884" width="800" height="400";
var paramtag = new Array();
paramtag[paramtag.length] = {param name="repository_agent_type" value="FROM_REMOTE"};
paramtag[paramtag.length] = {param name="repository_agent_server_name" value="127.0.0.1"};
paramtag[paramtag.length] = {param name="repository_agent_server_port" value="8082"};
paramtag[paramtag.length] = {param name="zadoc_path" value="AppSample.oz"};
paramtag[paramtag.length] = {param name="viewer_namespace" value="CustomNamespace@BZ"};
oz_activex_build(BZEmbedControlLocation, tag, paramtag);
</script>
</div>
</body>
</html>
    
```

※ 주의사항

리포트 뷰어 설치 태그를 작성한 'ztransferx.js' 파일과 'oz_activex.js' 파일은 리포트 파일 호출 태그를 작성한 HTML 파일과 같은 위치에 있어야 합니다.

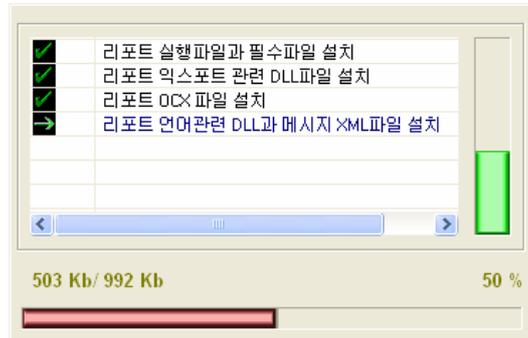
Step 4

오즈를 웹에서 서비스하기 위해서는 기본적으로 오즈 서버와 웹 서버가 구동되어 있어야 합니다.

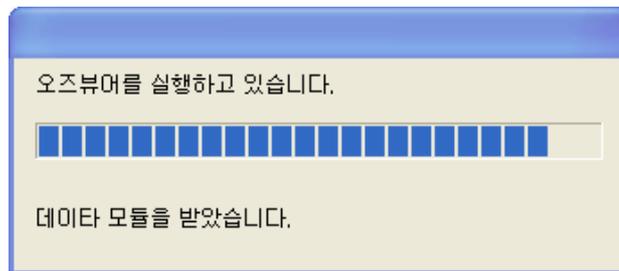
- 1) 리포트 파일 호출 태그를 삽입한 HTML 파일(ozsample.html)을 웹 브라우저로 엽니다.
- 2) 인증 창이 나타나면서 Ztransfer Control을 설치할 것인지를 물어봅니다.



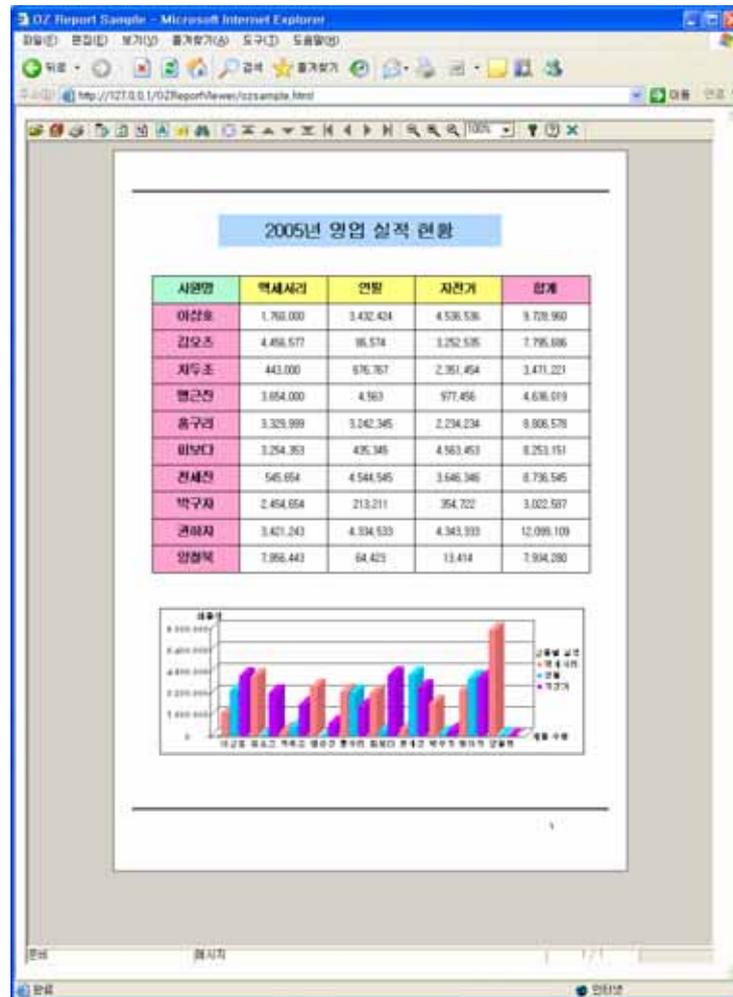
- 3) Ztransfer가 설치되면 오즈 ActiveX Viewer를 다운로드해서 설치합니다.



- 4) 설치가 완료되면 데이터 모듈과 폼을 받고 바인딩하는 진행사항을 볼 수 있는 프로그래스 바가 나타납니다.



- 5) 프로그래스 바의 진행이 완료되면 다음과 같이 오즈 뷰어가 삽입된 브라우저 화면이 나타납니다.

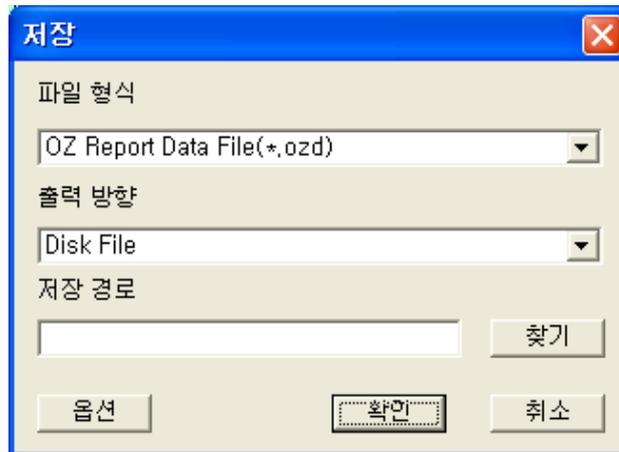


■ 인쇄하기

인쇄 아이콘()을 클릭하여 오즈 리포트 뷰어를 통해 웹에서 미리보기한 보고서를 프린터로 인쇄할 수 있습니다.

■ 저장하기

데이터 저장 아이콘()을 클릭하면 오즈 보고서를 다양한 형식의 파일로 전환하여 사용자 PC에 저장할 수 있습니다.





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

첫째로 오즈 리포트 디자이너의 전체적인 기능을 Overview를 통해서 사용자가 오즈 리포트 디자이너를 통해 어떠한 기능들을 구현할 수 있는가를 설명하였고, 둘째로 Quick Tour에서는 처음 오즈 리포트 디자이너를 접하는 사용자에게 데이터 연결 설정부터 보고서 작성까지 예제를 중심으로 설명하였습니다.

오즈 리포트 디자이너의 세부적인 기술 정보를 원하신다면 Getting Started외의 세분화된 매뉴얼을 참조하시기 바랍니다.

오즈 리포트 디자이너의 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