

목 차

Chapter1. OZ e-Form 소개

1. OZ e-Form 소개	4
1.1 OZ e-Form 개요	4
1.2 OZ e-Form 주요 기능	5
1.3 OZ e-Form 서비스 프로세스	7
2. 디자이너 설치 및 실행	8
2.1 설치	8
2.2 실행	9
2.3 라이선스 적용 및 버전 확인	10
3. 입력 컴포넌트 소개	11

Chapter2. Exercise

1. 프로젝트 신청서	14
1.1 목표	14
1.2 목표 보고서 정의	17
1.3 보고서 작성 가이드	18
2. 개발 신청서	20
2.1 목표	20
2.2 목표 보고서 정의	22
2.3 보고서 작성 가이드	23
3. 주문서	30
3.1 목표	30
3.2 목표 보고서 정의	31
3.3 보고서 작성 가이드	32
4. 견적서	40
4.1 목표	40
4.2 목표 보고서 정의	42
4.3 보고서 작성 가이드	44
5. 출장보고서	49
5.1 목표	49
5.2 목표 보고서 정의	50
5.3 보고서 작성 가이드	51



목 차

Chapter3. 서버 설치 및 뷰어 테스트

1. 서버 및 뷰어 설치	64
2. 보고서 업로드	65
3. 오즈 뷰어 호출	67
3.1 목표	67
3.2 실행	69
3.3 라이선스 적용 및 버전 확인	70
※ 교재 샘플 소개	74



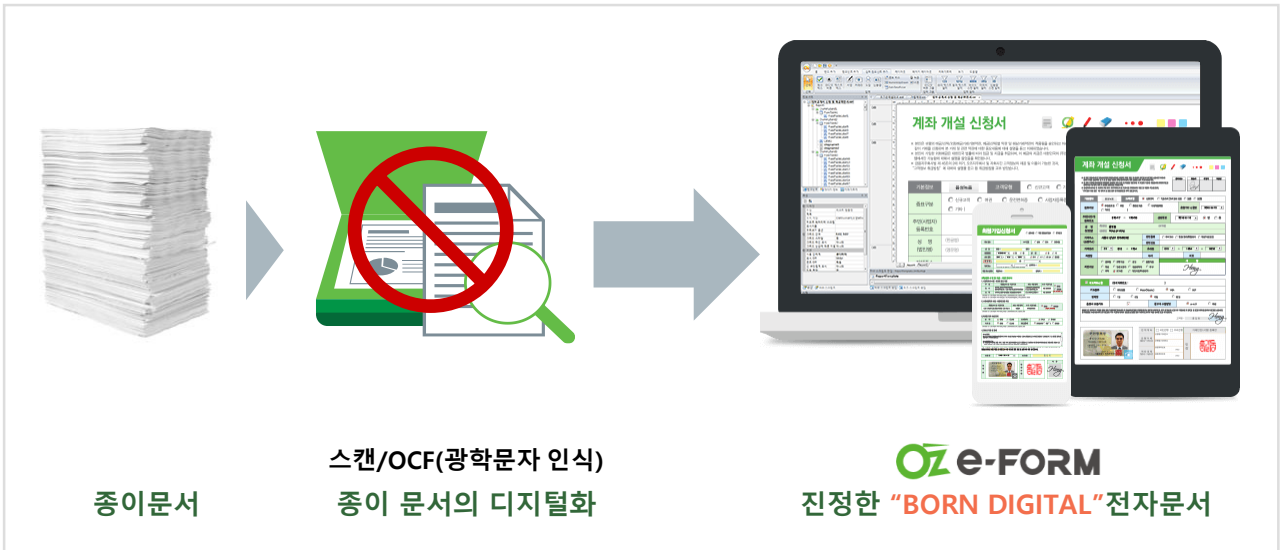
Chapter1. OZ e-Form 소개

1. OZ e-Form 개요
2. OZ e-Form 주요 기능
3. 디자이너 설치 및 실행



가. OZ e-Form 개요

- ✓ 종이를 전혀 사용하지 않는 진정한 "BORN DIGITAL" 전자문서 생성 솔루션입니다.
- ✓ 서식 제작단계부터 기업의 데이터와 연계하고, 언제 어디서나 모바일 환경에서 문서 작성이 가능합니다.
- ✓ 사진, 녹음, 바코드 인식 등 데이터 입력의 편의성과 안전한 저장을 지원하는 스마트 페이퍼리스 솔루션입니다.



[전자문서 생성 패러다임의 변화]

나. OZ e-Form 기대 효과

- ✓ 전자문서는 기업의 업무 효율성 향상과 보안 강화, 비용 절감, 녹색 성장이라는 최고의 가치를 실현합니다.



업무 효율성

- 업무 프로세스 혁신
- 원스톱 업무 처리



문서 보안 강화

- 문서 보관/폐기 시 개인정보 유출 방지
- 보안솔루션 연동, 안전한 보관



비용 절감

- 종이 문서 제작/보관 비용 절감
- 수작업 인력 비용 절감



녹색 성장

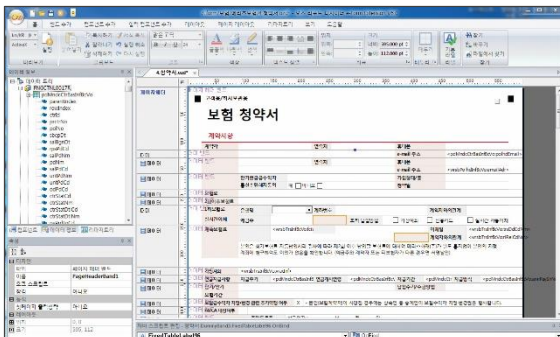
- 탄소 배출량 감소
- 환경 오염 방지

가. 프리미엄 리포팅 엔진 기반의 통합 전자문서 솔루션

- XML 서식기반 전자문서
 - ✓ 데이터 값에 따라 동적으로 변경되는 전자문서 지원
- 고성능 아키텍처
 - ✓ 서버에서 문서를 생성하지 않고 클라이언트에서 동적으로 문서를 생성하여 서버 부하 최소화
- 통계 분석 보고서 기능
 - ✓ 데이터 입력과 분석을 동시에 지원
 - ✓ 표나 차트의 드릴 다운 분석 지원
- OSMP (원소스 멀티플랫폼)
 - ✓ 하나의 서식 개발로 다중 플랫폼 & 디바이스에서 동일하게 서비스 가능
(데스크톱 및 모든 모바일 디바이스 지원)
- 문서 보안 강화
 - ✓ 강력한 서명자 인증 정보(그래픽 서명 정보, 서명 필압/속도, 서명 시간, 서명 위치, 서명자 정보)
 - ✓ 문서 변경 내용 기록, 문서 수정 시점 타임스탬프 연동, 무결성 검증을 위한 해시값 저장

나. 프리미엄 리포팅 엔진 기반의 통합 전자문서 솔루션

- 전문가를 위한 고급 서식 개발 환경
 - ✓ 다양한 로직 처리: 자바 스크립트, 다양한 이벤트 및 함수 지원
 - ✓ 강력한 데이터 처리: 입력 데이터 트랜잭션 처리, 다양한 데이터 소스 연동
 - ✓ 기존 오피스 문서를 Copy/Paste로 편리하게 디자인



다. 스마트 모바일 뷰어

iOS, Android, Windows mobile 등 모든 모바일 환경을 완벽하게 지원하며, 다양한 입력 컴포넌트와 주석 및 이미지 추가/편집기능을 통해 종이 문서보다 더 편리하게 전자문서 작성 가능

■ HTML5 & Native App 뷰어

- ✓ 모바일 환경 완벽 지원

■ 주석 및 이미지 추가/편집 기능

- ✓ 밑줄 및 형광펜 기능 제공
- ✓ 사용자 메모 기능

■ 전자 서명

- ✓ 서명 영역 자동 확대, 고해상도의 편리한 서명
- ✓ 터치스크린 단말기를 이용한 전자서명 시스템 및 그 방법

■ 신분증 스캔

- ✓ 개인정보 자동 마스킹 처리로 개인정보 보호 강화

■ 인감도장 스캔

- ✓ 스캔용지를 이용하여 실물크기로 인감도장 스캔
- ✓ 모바일 기기의 카메라 모듈을 이용한 스캔 시스템 및 그 방법



라. 다중 장치 협업 입력기술

고객 대면 창구 업무의 페이퍼리스 시스템 구현 및 다양한 장치를 이용하여 원격에서 실시간으로 동시 입력을 지원하는 기술

■ 다중 장치 제어 기능

- ✓ 업무용 데스크탑 PC와 다중 장치 (태블릿 PC나 태블릿 모니터)간 입력 값의 실시간 동기화 처리
- ✓ 입력 가이드 기능 및 사용자 관점에 따른 편리한 입력 및 서식 제어

■ 다중 언어 보고서 표시

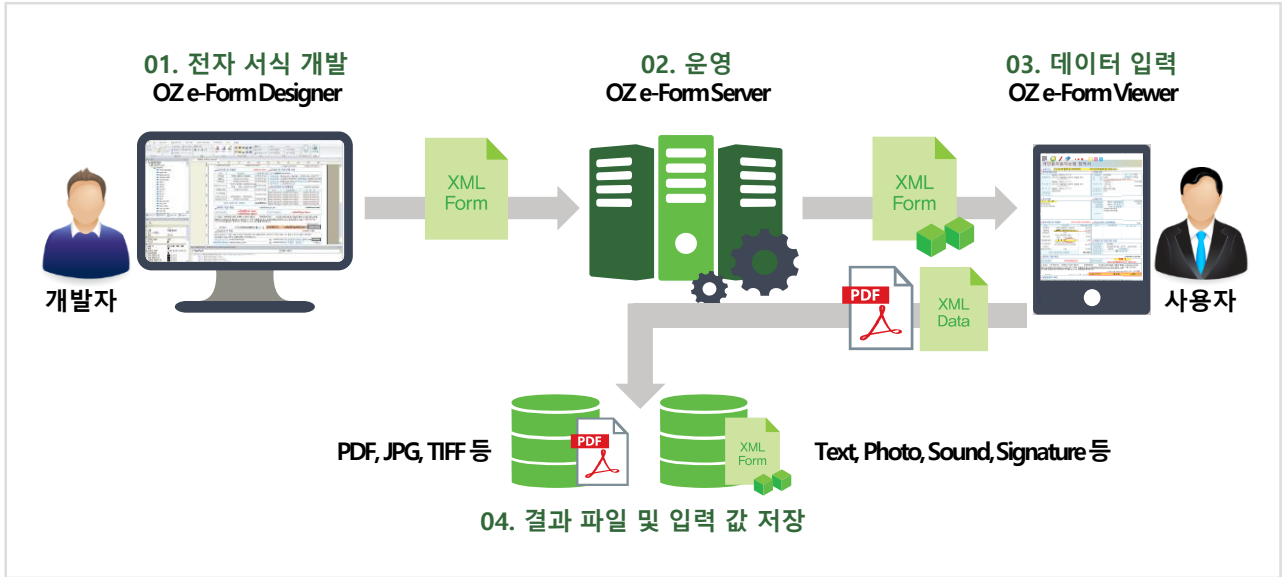
- ✓ 고객의 언어에 따라 대면 창구 업무에서 동적으로 고객에게 친숙한 언어로 서식 표시

■ 손글씨 지원

- ✓ 시스템의 국가별 언어를 자동 인식하여 손 글씨를 언어에 맞게 텍스트로 자동 변환



가. OZ e-Form 서비스 프로세스



나. 활용분야



가. 설치 방법








① 오즈 디자이너 설치 파일 실행

② '다음' 버튼 클릭하고 '사용권 계약' 동의

③ 설치 경로 확인

④ '전체 설치' 선택

⑤ 로컬 서버 타입 선택

※ 로컬 서버 타입은 오즈 서버와 같은 타입을 선택합니다.
오즈 서버 타입을 모르는 경우 'Java', '.Net' 모두 선택합니다.


⑥ 설치 완료되면 바탕화면과 시작 메뉴에 아이콘 생성됨

나. 기본 설치 경로 및 주요 파일

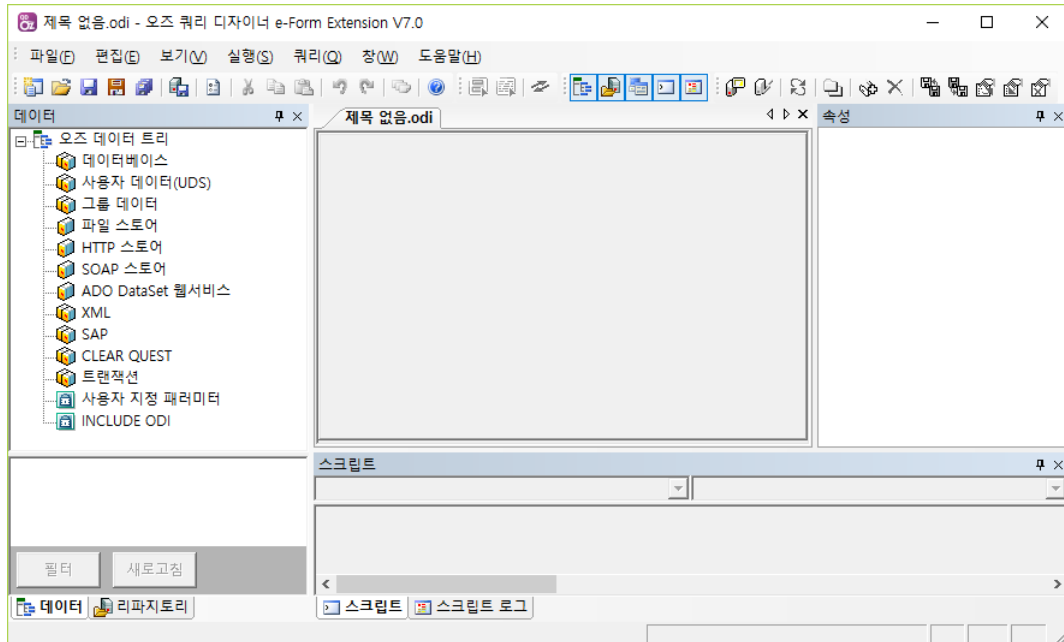
제품	기본 설치 경로
OZ e-Form Designer 7.0	%ProgramFiles%\FORCS\OZ e-Form 7.0\OZ e-Form Designer 7.0(이하 %OZD_HOME%)
OZ Query Designer	%ProgramFiles%\FORCS\OZ e-Form 7.0\OZ Query Designer 7.0(이하 %OZD_HOME%)

폴더	설명
%OZD_HOME%	- dbconfig.xml, dbconfig.xsd : DB 벤더 연결 정보 설정 파일
%OZD_HOME%\lib	XML파서, HTTPS/SSL관련 J2EE, LOG, SOAP, 메일, 압축, 오즈 서버 라이브러리 파일
%OZD_HOME%\license	- ozlicense.xml : 디자이너 라이선스 파일
%OZD_HOME%\UniversalDriver	벤더 별 JDBC Driver
%TEMP%\WOZW임시폴더	- Console.txt : 디자이너 내부 서버 로그 파일
내 문서\WOZW임시폴더	- QDConfig.ini : 쿼리 디자이너 설정 파일 - OZReportDesigner.ini : e-Form 디자이너 설정 파일

가. 쿼리 디자이너 실행


바탕 화면의 오즈 쿼리 디자이너 아이콘  을 실행합니다.

- 실행 파일 경로 : %ProgramFiles%\WFORCS\WOZ e-Form 7.0\WOZ Query Designer 7.0\bin\WOZQueryDesigner_eform.exe

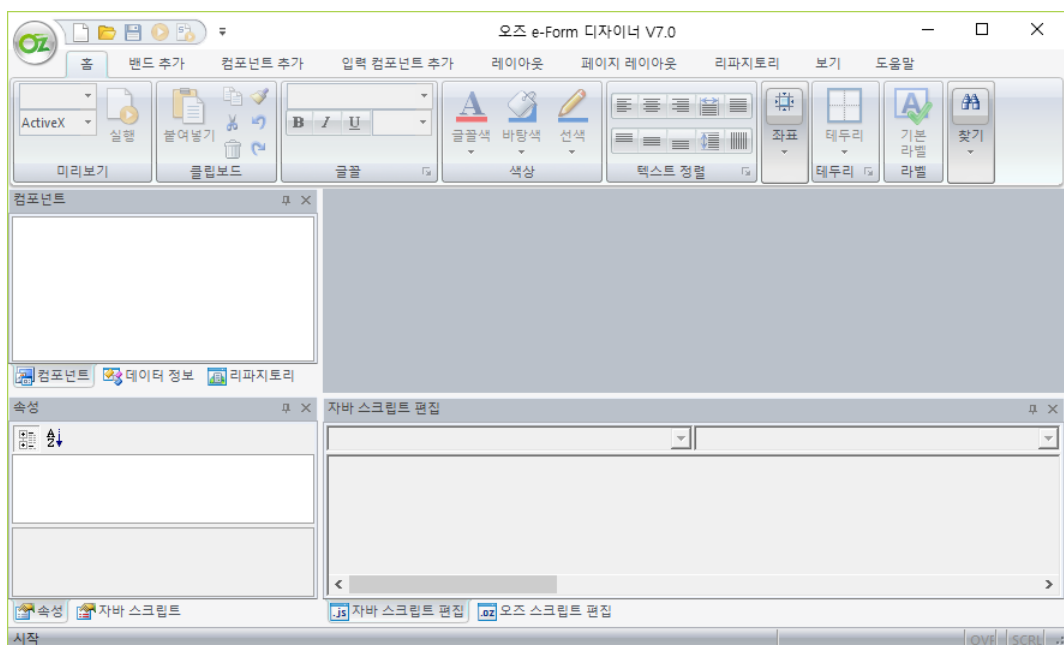


[쿼리 디자이너 실행 초기화면]

나. e-Form 디자이너 실행

바탕 화면의 오즈 e-Form 디자이너 아이콘  을 실행합니다.

- 실행 파일 경로 : %ProgramFiles%\WFORCS\WOZ e-Form 7.0\WOZ e-Form Designer 7.0\bin\WOZReportDesigner_eform.exe



[e-Form 디자이너 실행 초기화면]

가. 라이선스 정책

디자이너 최초 설치 후 라이선스 파일 없이 3개월간 사용할 수 있습니다.

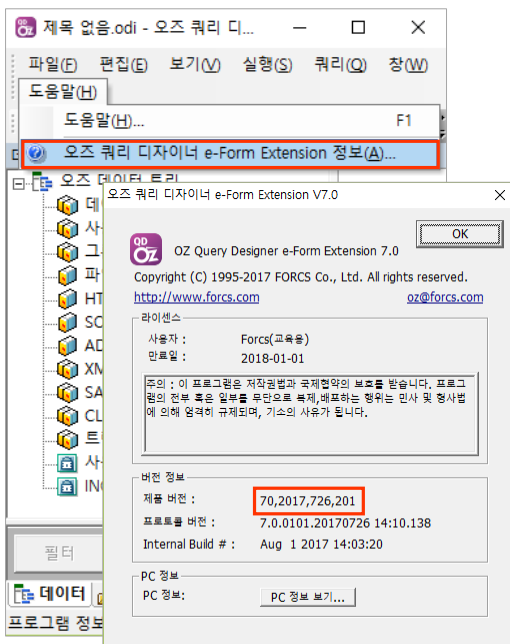
라이선스 파일을 적용하려면 설치되어 있는 각 디자이너 경로의 license 폴더에 ozlicense.xml 파일명으로 복사합니다.

- %ProgramFiles%\WForcs\OZ e-Form 7.0\OZ Query Designer 7.0\license\ozlicense.xml
- %ProgramFiles%\WForcs\OZ e-Form 7.0\OZ e-Form Designer 7.0\license\ozlicense.xml

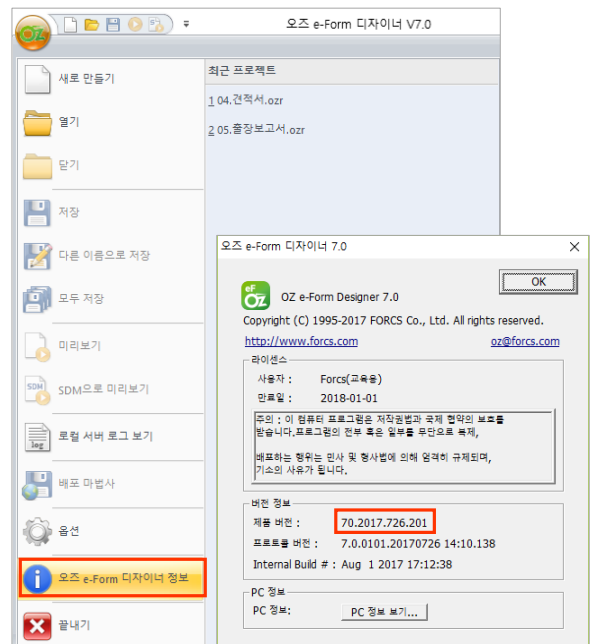
나. 디자이너 버전 확인

쿼리 디자이너 : 디자이너 실행 → '도움말' → '오즈 쿼리 디자이너 e-Form Extension 정보' 메뉴 선택

e-Form 디자이너 : 디자이너 실행 → 'OZ' 버튼 → '오즈 e-Form 디자이너 정보' 메뉴 선택




[쿼리 디자이너 버전 확인]



[e-Form 디자이너 버전 확인]

3. 입력 컴포넌트 소개

회원 가입

■ 성명	<input type="text" value="이서연"/>		
■ 생년월일	<input type="text" value="1990-10-21"/>	(<input checked="" type="radio"/> 양력 <input type="radio"/> 음력)	
■ 휴대폰번호	<input type="text" value="010"/>	<input type="text" value="1 2 3 4 - 5 6 7 8"/>	
■ 취미	<input checked="" type="checkbox"/> 여행 <input checked="" type="checkbox"/> 운동 <input type="checkbox"/> 독서		
■ 나이	<input type="text" value="26"/>	<input type="text" value="살"/>	■ 가입인사 <input type="button" value="녹음 완료!"/>
■ 사진			

2015년 12월 08일
신청인 :

- 1) CheckBox : 여러 항목 중 하나 이상 선택할 수 있는 컴포넌트입니다.
- 2) RadioButton : 여러 항목 중 하나만 선택할 수 있으며 항목별로 그룹핑하여 사용할 수 있습니다.
- 3) TextBox : 숫자나 문자열을 입력할 때 사용하는 컴포넌트입니다.
- 4) SignPad : 문구 따라 쓰기, 서명을 입력할 수 있는 컴포넌트입니다.
- 5) Camera : 사진 촬영, 도장 촬영, 신분증 촬영을 제공하는 컴포넌트입니다.
- 6) ComboBox : 다수의 아이템을 리스트 형태로 보여주는 컴포넌트입니다.
- 7) NumericUpDown : 숫자 값을 직접 입력하거나 위/아래 버튼을 클릭하여 숫자 값을 증감시키고자 할 때 사용하는 컴포넌트입니다.
- 8) DateTimePicker : 날짜 또는 시간을 입력하거나 달력을 이용하여 날짜를 입력하는 컴포넌트입니다.
- 9) VoiceRecorder : 음성을 녹음하는 컴포넌트입니다.

Chapter2. Exercise

1. 프로젝트 신청서
2. 개발 신청서
3. 주문서
4. 견적서
5. 출장보고서



e-Form 제품을 시작하면서 알아야 하는 기본 설정을 익히고, 기본이 되는 컴포넌트 기능을 습득한 후 간단한 전자 서식을 개발하는 방법을 습득합니다.

가. OZ e-Form 디자이너

❖ Designer 시작하기

Designer 실행 > 새로 만들기 > 밴드 추가 > 컴포넌트 추가 > 미리보기

❖ Repository 사용

1) 작업 폴더 설정

로컬 PC의 작업할 폴더 위치 지정

2) 리파지토리 서버에 연결

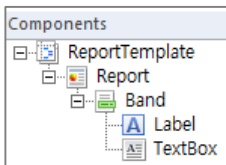
e-Form 서버로 연결

3) 보고서를 리파지토리에 업로드/다운로드

보고서를 e-Form 서버에 업로드

❖ report design 실행

■ ReportTemplate 컴포넌트의 구조

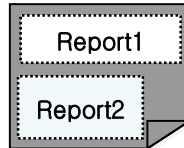


■ 리포트 타입

다중 리포트



복합 리포트



■ 기본 설정

- 리포트 단위 : ReportTemplate
- 리포트 타입, 용지 방향, 여백 : Report

나. OZ e-Form 컴포넌트

❖ 밴드

리포트 디자인은 여러 밴드로 이루어 집니다.

■ 주요 밴드

- 타이틀 밴드 : 보고서 첫 페이지 상단에 한번 출력되는 밴드입니다.
- 페이지 헤더 밴드 : 문서의 머리글처럼 매 페이지 상단에 반복 출력되는 밴드입니다.
- 데이터 밴드 : 데이터 셋의 데이터를 보여주기 위해 사용되는 밴드입니다.
- 페이지 풋터 밴드 : 문서의 바닥글처럼 매 페이지 하단에 반복 출력되는 밴드입니다.
- 써머리 밴드 : 보고서 맨 마지막 페이지에 한 번만 표시됩니다.
- 더미 밴드 : 반복 횟수 속성 값에 설정한 횟수만큼 반복하여 표시됩니다.

❖ 라벨

라벨 컴포넌트는 텍스트, 데이터 필드 값, 시스템 날짜와 시간, 이미지 등을 표현하는 컴포넌트 입니다.

■ 주요 속성

- 타입 : 데이터 형태에 따른 구분, 그리 형태에 따른 구분
- 데이터 : 텍스트, 필드 이름, 패러미터 이름
- 이미지 : 이미지 URL 설정, 이미지 삽입, 백그라운드 밴드 내 이미지
- 테두리 : 테두리 두께, 테두리 없음, 테두리 점선
- 폰트 : 글꼴, 스타일, 크기, 글꼴 색
- 포맷 : 일반, 날짜, 숫자

❖ 고정 테이블

고정 테이블은 표를 표시할 때 사용하는 컴포넌트입니다.

❖ 텍스트 박스

텍스트박스는 문자열을 입력 받을 때 사용하는 컴포넌트입니다.

■ 주요 속성

- 여러 줄 작성 : MultiLine
- 최대 글자 수
- 키보드 타입
- 패스워드 문자 표시

❖ 체크 박스

한 문항에 대해 다중 선택을 할 수 있는 컴포넌트입니다.

■ 주요 속성

- 체크 상태 : Checked
- 체크 수평 정렬
- 스타일 : 체크 스타일, 프레임 색/크기/두께/바탕 색, 체크 마크 색
- CheckedValue, UnCheckedValue

❖ 라디오 버튼

한 문항에 대해 하나만 선택할 수 있는 컴포넌트입니다.

■ 주요 속성

- 라디오 버튼 그룹 : 선택 안 함 가능
- 나머지 기능은 CheckBox와 동일

❖ NumericUpDown

NumericUpDown은 숫자 값을 입력 받는 컴포넌트입니다.

■ 주요 속성

- 증감 단위
- 최대 값, 최소 값

❖ DateTimePicker

DateTimePicker는 달력을 통해 날짜를 선택하거나 특정 포맷으로 날짜를 표현할 수 있는 컴포넌트입니다.

■ 주요 속성

- 날짜 값
- 표시 형식
- 최대 날짜, 최소 날짜

가. 목표 보고서 : 프로젝트 신청서

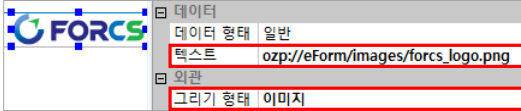
The image shows a 'Project Application Form' (프로젝트 신청서) with the FORCS logo in the top right corner. The form is divided into two main sections: '1. 신청자 정보' (Applicant Information) and '2. 프로젝트 정보' (Project Information). Section 1 includes fields for name, company address, phone number, fax number, email, gender, birth date, and age. Section 2 includes a field for the project purpose and a field for the project description. Annotations 1 through 9 point to specific features: 1 points to the FORCS logo, 2 points to the date '2017-08-03', 3 points to the phone number field, 4 points to the email field, 5 points to the gender selection, 6 points to the birth date field, 7 points to the age field, 8 points to the role selection (PM, PL, or initiator), and 9 points to the project description field.

나. 과제

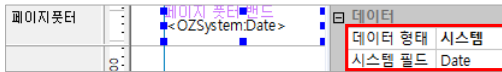
- ① 매 페이지 상단에 회사 로고 이미지 표시
- ② 매 페이지 하단에 현재 날짜, 페이지 번호(현재 페이지/전체 페이지) 표시(시스템 라벨 이용)
- ③ 전화번호, 팩스번호는 숫자 키패드로 입력(키보드 타입 속성 이용)
- ④ E-mail 주소는 이메일 키패드로 입력 (키보드 타입 속성 이용)
- ⑤ 성별은 남자, 여자 중에 하나만 선택 가능
- ⑥ 생년월일의 초기값은 1970-01-01로 설정
- ⑦ 나이는 20 ~ 70살 까지만 입력 가능하도록 설정하고 초기 값은 20으로 설정 함
- ⑧ 역할은 다중 선택 가능하며 체크박스를 오른쪽에 위치 시킴
- ⑨ 2.1, 2.2 항목에는 멀티 라인 입력이 가능하고 2.2 항목은 300자로 제한 함

다. 구현 방법

(1) 매 페이지 상단에 회사로고, 하단에 현재 날짜, 페이지 번호 표시



- 페이지 헤더 밴드에 이미지 라벨을 추가
- 이미지 라벨의 팝업 메뉴에서 '이미지 URL'을 선택한 후 이미지 파일을 선택 함



- 페이지 풋터 밴드에 시스템 라벨을 추가하여 현재 날짜, 페이지 번호를 표시

- ◆ 시스템 필드
 - Date : 현재 날짜
 - Page_Of_Total : 현재 페이지 번호/전체 페이지 번호

(2) 키패드 설정

키보드 타입	숫자
리턴키 타입	Default
팩스워드 문자 표시	아니오
텍스트 필터 이름	StringTextFilter_zipcode

- 텍스트 박스의 '키보드 타입' 속성을 '숫자' 로 선택하여 우편번호, 전화번호, 팩스번호용 키패드를 설정
- 텍스트 박스의 '키보드 타입' 속성을 '이메일' 로 선택하여 E-mail 주소용 키패드를 설정

- ◆ 텍스트 박스의 키보드 타입(속성)
 - 텍스트 박스에 텍스트 입력 시 실행할 키패드 타입을 설정
 - 알파벳, 숫자, 이메일 등으로 설정이 가능

(3) 라디오 버튼을 이용한 성별 설정

타입	라디오 버튼
이름	RadioButton1
오즈 스크립트	
장깁	아니오
Group Name	Group1
데이터	
데이터 형태	일반
텍스트	남자

- ◆ 라디오 버튼
 - 한 문항에 대해 하나만 선택할 수 있는 컴포넌트

(4) DateTimePicker를 이용한 생년월일 설정

타입	DateTimePicker
이름	DateTimePicker1
오즈 스크립트	
장깁	아니오
데이터	
데이터 형태	일반
날짜 값	01/01/1970
외관	
바탕 색	1970년 1월
투명	일 월 화 수 목 금 토
투명도	28 29 30 31 2 3
	4 5 6 7 8 9 10

- DateTimePicker의 '날짜 값' 속성의 버튼을 클릭하여 1970년 1월 1일을 선택

(5) NumericUpDown을 이용한 나이 설정

디자인	
타입	NumericUpDown
이름	NumericUpDown1
오즈 스크립트	
장김	아니오
데이터	
데이터 형태	일반
값	20
외관	
동적	
증감 단위	1
최대 값	70
최소 값	20

- NumericUpDown의 '값' 속성은 20으로 설정하고, '최대 값', '최소 값' 속성은 각 '70'과 '20'으로 설정

(6) 체크 박스를 이용한 역할 설정

속성	
디자인	
타입	체크 박스
이름	CheckBox1
오즈 스크립트	
장김	아니오
데이터	
데이터 형태	일반
텍스트	PM(Project Manager)
외관	
Checked	아니오
텍스트 수평 정렬	왼쪽
텍스트 수직 정렬	중간
체크 수평 정렬	오른쪽

- 체크 박스의 '체크 수평 정렬' 속성의 값을 '오른쪽'으로 설정
- ◆ 체크 박스
 - 한 문항에 대해 여러 항목을 선택할 수 있는 컴포넌트

(7) 텍스트 박스에 멀티 라인 및 최대 글자수 제한 설정

속성	
디자인	
타입	텍스트 박스
이름	TextBox9
오즈 스크립트	
장김	아니오
MultiLine	예
데이터	
외관	
동적	
최대 글자 수	300

- 텍스트가 여러 줄일 경우 텍스트 박스의 'MultiLine' 속성의 값을 '예'로 설정
- 텍스트 박스의 '최대 글자 수'를 '300'으로 설정하여 표시 가능한 최대 글자 수를 제한

입력 필터와 서명, 카메라, 녹음과 같은 OZ e-Form 입력 컴포넌트 기능들을 익힙니다.
동적인 전자문서 구현과 필수값 체크, 입력 컴포넌트 값 추출하는 방법에 대해 익힙니다.

가. OZ e-Form 컴포넌트

❖ 입력 필터

숫자 텍스트 필터는 텍스트 박스에 숫자만 입력될 수 있도록 하는 필터 기능을 제공합니다.

문자 텍스트 필터는 텍스트 박스에 전화번호, 주민번호, 우편번호와 같은 특정 패턴을 지정하여 입력 받을 수 있도록 하는 필터 기능을 제공합니다.

■ 주요 밴드

- 천 단위 구분자 표시, 음수 부호 입력 허용 : 숫자 텍스트 필터
- 패턴 : 문자 텍스트 필터

❖ 서명

서명은 터치 펜이나 터치 스크린을 이용하여 실제 서명을 입력 받는 컴포넌트입니다.

■ 주요 속성

- 테두리: 테두리 없음, 테두리 두께, 테두리 색
- 저장 시 사인 이미지의 사이즈 : 서명이미지 너비, 서명이미지 높이
- 서명이미지 포맷 : 32bit PNG 포맷으로 저장하면 서명 외 부분을 투명하게 처리
- 서명펜 두께, 서명펜 색 : 색상은 다른 이미지 포맷으로 적용할 수 없음
- SignedValue, UnSignedValue : 서명한 내용을 대신하여 전달할 내용
- Signature 그룹 : SetGroupName

❖ 카메라

카메라는 이미지를 표시하거나 사진 촬영을 실행하는 컴포넌트입니다.

■ 주요 속성

- PickedValue, UnPickedValue : 이미지나 촬영을 대신하여 전달할 내용
- ImagePickMode : 카메라, 도장/신분증 스캔
- SetCameraDPI

❖ 녹음

녹음은 음성을 녹음하는 컴포넌트로, 모바일 앱 뷰어에서만 동작됩니다.

❖ 버튼

버튼은 사용자의 명령을 입력 받는 버튼을 나타낼 때 사용하는 컴포넌트입니다.

■ 주요 속성

- OnCheckValidity 이벤트를 통해 필수값을 체크
: 실패 시 false 값을 리턴
: EnsureVisible : 컴포넌트가 화면에 보이도록 보고서를 스크롤 함
- OnClick 이벤트를 통해 컴포넌트의 속성을 변경. e-form의 외부로부터 OnClick 이벤트를 호출

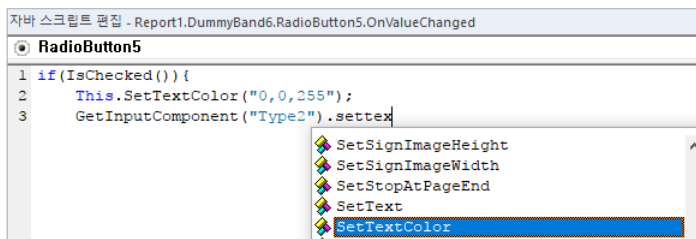
❖ 공통

- FormID(입력 컴포넌트, 라벨, 고정 테이블 라벨의 이름을 대신할 ID), Hidden, 전송 가능
- GetInputComponent, SetFormID, SetInputValue
- input data 추출 : GetInputValue(), GetInputXML(), GetInputJSON()

나. 스크립트 자동 완성 기능 및 오즈 디자이너 도움말 사용법

❖ 스크립트 자동 완성 기능

스크립트 입력 시 <Ctrl + Space> 키를 눌렀을 경우 헬프 창이 표시되며 사용할 메소드를 선택하면 대소문자가 구분되며 자동으로 메소드를 완성해줍니다.



❖ 도움말

오즈 e-Form 디자이너의 도움말 메뉴에서 오즈 디자이너 도움말(F1)을 선택합니다. 아래 항목을 통해 각 메소드별 사용 방법에 대해 확인할 수 있습니다.

- 오즈 e-Form Extension > 오즈 e-Form 자바 스크립트 API > Input > RadioButton > Method

가. 목표 보고서 : 개발 신청서

개발 신청서

확인자 한

개발 번호	
평가 번호	

필수값 체크 : JSON

개발 위치 정보

우편번호
주소

신청인 정보

이름
주소
전화번호
우편번호
E-mail

신청인이 본인 인가요? ☒ 예 ☐ 아니오

참고:
위의 사항에 '예'라고 선택하면, 승인 정보는 신청인에게 우편이나 메일이 발송됩니다.
이 정보를 부통산 소유주에게 전달하는 것은 귀하의 책임입니다.

개발 신청에 대한 승인 후 담당자. ☒ 소유자 ☒ 작성자

소유자 정보

이름
주소
E-mail
우편번호

작성자 정보

이름
주소
E-mail
자격증 번호

개발에 대해 설명 바랍니다.

아래 항목 중 하나를 선택하십시오.

☐ 개발 계획만 동의 ☒ 건축 규칙만 동의 ☐ 모두 동의(개발 계획 및 건축 규칙)

건축 규칙에 대한 분류

과거 분류
현재 분류

작년에 건설 산업 교육 기금은 지급 되었습니까? ☒ 예 ☐ 아니오

개발비용

신청인 이름/서명
날짜 2017/08/29
기타

자세한 내용은 1544-0181으로 문의하거나 www.forcs.com을 방문하십시오.

나. 과제

- 전화번호, 우편번호, 자격증 번호, 개발 비용은 '숫자' 키패드로 입력
- E-mail 은 Email 키패드로 입력
- 우편번호 포맷은 ###-###로 중간에 -은 자동 입력(문자 텍스트 필터)
- 자격증 번호를 입력하면 패스워드 문자 ●로 표시
- '아래 항목 중 하나를 선택하십시오.' 아래 체크된 텍스트 색상을 다르게 표시
- 예, 아니오는 둘 중에 하나만 선택 가능(라디오 버튼 그룹)
- 개발비용에 입력할 때 세자리 마다 콤마 자동 입력되고, 소수점 이하 2자리까지 표시(숫자 텍스트 필터)
- 신청인 이름/서명의 서명은 그룹 서명이 되도록 설정
- 날짜의 포맷은 yyyy/MM/dd로 설정
- 소유자를 체크하면 소유자 정보 영역이 보여지고, 체크 해제하면 소유자 정보 영역이 숨겨지도록 구현, 작성자를 체크 시에도 동일하게 동작(소유자/작성자 정보 영역을 별도의 밴드로 분리. ReBind 이용)
- 필수값 체크 버튼을 클릭했을 때 신청인 정보의 이름, 전화번호와 개발 비용의 데이터 존재 여부를 체크하고 없으면 메시지 창으로 알린 후 해당 컴포넌트로 화면 이동(OnCheckValidity, EnsureVisible, SetFocus)
- JSON 버튼을 클릭하면 입력된 데이터를 JSON 형태로 메시지 창에 출력. 이 때 신청자 사진의 촬영이 된 경우 Y, 촬영이 안된 경우 N으로 대체, 서명도 사인이 된 경우 Y, 안된 경우 N으로 대체

다. 구현 방법

(1) 자격증 번호에 패스워드 문자 ● 설정

키보드 타입	숫자
리턴키 타입	Default
패스워드 문자 표시	예

OZ Report Viewer	
E-mail	
자격증 번호	●●●●●●●●

- 텍스트 박스의 '패스워드 문자 표시' 속성을 '예' 로 선택
- 미리보기하여 자격증 번호를 입력하면, ●로 표기 됨

(2) 체크된 텍스트 색상을 다르게 설정

- 라디오버튼(FormID : Type1~Type3)의 자바 스크립트 편집 창의 OnValueChanged 이벤트에 스크립트를 작성

자바 스크립트 편집 - Report1.DummyBand6.RadioButton5.OnValueChanged

```

1 if (isChecked()) {
2     This.SetTextColor("0,0,255");
3     GetInputComponent("Type2").SetTextColor("0,0,0");
4     GetInputComponent("Type3").SetTextColor("0,0,0");
5 }
6

```

- 미리보기한 후 각 항목 선택에 따라 텍스트 색상이 다르게 표현되는지 확인

OZ Report Viewer

개발에 대해 설명 바랍니다.

아래 항목 중 하나를 선택하십시오.

☒ 개발 계획만 동의 ☐ 건축 규칙만 동의 ☐ 모두 동의(개발 계획 및 건축 규칙)

(3) 라디오 버튼 그룹을 통한 그룹핑 설정

라디오 버튼 그룹

컴포넌트 속성

타입	라디오 버튼
이름	RadioButton1
오브 스크립트	아니오
그룹 이름	RadioButtonGroup3
데이터 형태	일반
텍스트	예

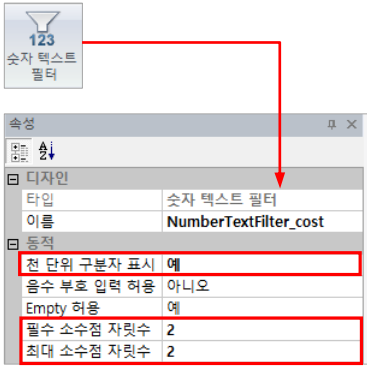
- 입력 컴포넌트 추가 탭의 입력 그룹 내 '라디오 버튼 그룹' 아이콘을 클릭하면 바로 보고서에 추가
- 여러 개의 라디오 버튼을 그룹핑 할 때 동일한 라디오 버튼 그룹을 선택

OZ Report Viewer

신청인이 본인 인가요?

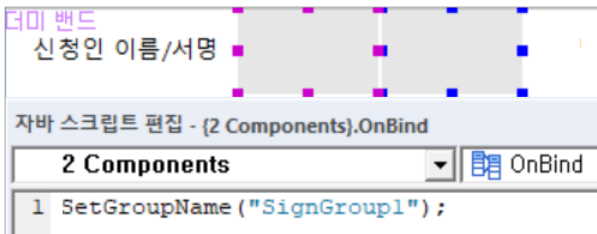
예 ☒ 아니오 ☐

(4) 숫자 텍스트 필터를 이용한 개발비용 설정



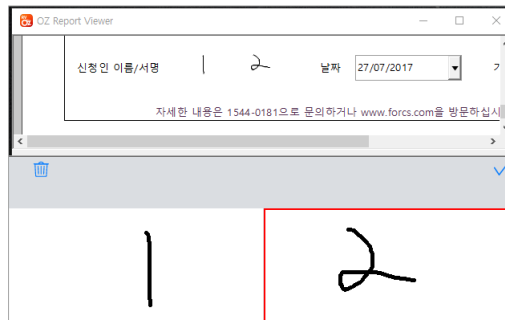
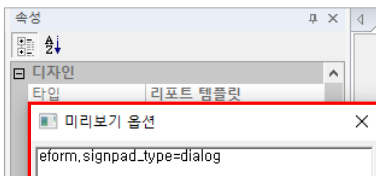
- 입력 컴포넌트 추가 탭의 입력 필터 내 '숫자 텍스트 필터' 아이콘을 클릭하면 바로 보고서에 추가
- 숫자 텍스트 필터 내 '전 단위 구분자 표시' 속성을 '예', '필수/최대 소수점 자릿수' 속성을 '2'로 설정
- 텍스트 박스의 '텍스트 필터 이름' 속성에서 생성한 숫자 텍스트 필터를 선택

(5) 신청인 이름/서명에 대해 그룹 서명 설정



- 서명을 클릭하여 서명 창이 실행될 때 같은 그룹 이름(예. "SignGroup1")으로 지정된 여러 개의 서명이 하나의 서명 창에 표시 됨
- 두 서명 컴포넌트에 자바 스크립트 편집 창의 OnBind 이벤트에 동일한 스크립트를 작성

- 리포트 템플릿의 '미리보기 옵션' 속성에 'eform.signpad_type=dialog' 로 설정하여 디자이너에서 미리보기 하여 확인 함



- ◆ 서명 그룹핑 기능을 사용하려면 Android, iOS, WinRT 뷰어에서는 서명 타입을 키패드 또는 줌 방식으로 설정하고 ActiveX 뷰어에서는 서명 타입을 다이얼로그 방식으로 설정해야 함
- ◆ 미리보기 옵션에 뷰어 패러미터를 지정하여 디자이너에서 미리보기를 할 수 있으며 서버에 보고서를 업로드 한 후에는 미리보기 옵션의 뷰어 패러미터는 반영되지 않음

(6) 날짜 포맷(yyyy/MM/dd)로 설정

속성	
디자인	
타입	DateTimePicker
이름	DateTimePicker1
오즈 스크립트	
잠김	아니오
데이터	
외관	
등적	
표시형식	date_yyyy/MM/dd
속성변환가능	아니오

- DateTimePicker의 '표시형식' 속성을 date_yyyy/MM/dd로 설정

OZ Report Viewer

날짜: 2017/08/03 기타: [녹음]

- 미리보기 하여 확인

(7) 담당자 선택 여부에 따라 디테일 정보가 보여지거나 숨겨지도록 설정

- ReBind 메소드를 이용하여 패러미터 값을 변경하여 보고서를 다시 바인딩 할 수 있음

패러미터 설정

Member(M): ShowOwnerDetail, ShowBuilderDetail

ShowBuilderDetail 속성(P):

타입	폼 패러미터 필드
데이터	
필드 이름	ShowBuilderDetail
필드 타입	Varchar
필드 값	true
필드 설명	

- 데이터 정보 창 내의 OZFormParam에 소유자, 작성자 선택 여부에 따라 디테일 정보를 보여줄 폼 패러미터 필드를 생성

- 각 담당자 별 체크박스의 Checked 속성 값을 '예'로 설정하고 자바 스크립트 편집 창의 OnValueChanged 이벤트에 스크립트를 작성

디테일 밴드 개발 신청에 대한 승인 후 담당자. ☒ 소유자 ☒ 작성자 Checked 속성: 예

SCRIPT_BEGIN .Report1.CheckBox4.OnValueChanged

var str = GetDataSetValue("OZFormParam.ShowBuilderDetail");

if(This.IsChecked()) { //소유자가 체크되면 소유자 디테일 밴드를 보여줌

var param =

"connection.pcount=2;connection.args1=ShowOwnerDetail=true;connection.args2=ShowBuilderDetail=" + str;
ReportTemplate.ReBind(ReBindTypeConst.Report, param, "");

} else { //소유자가 체크 해제되면 소유자 디테일 밴드를 숨김

var param =

"connection.pcount=2;connection.args1=ShowOwnerDetail=false;connection.args2=ShowBuilderDetail=" + str;
ReportTemplate.ReBind(ReBindTypeConst.Report, param, "");

}

SCRIPT_END

※ **SCRIPT_BEGIN, SCRIPT_END** 라인은 자바 스크립트 편집 창에 작성하지 마십시오.

디테일 밴드

개발 신청에 대한 승인 후 담당자.

☒ 소유자☒ 작성자

Checked 속성 : 예

SCRIPT_BEGIN .Report1.CheckBox5.OnValueChanged

var str = GetDataSetValue("OZFormParam.ShowOwnerDetail");

if(This.IsChecked()) {

//작성자가 체크되면 작성자 디테일 밴드를 보여줌

var param = "connection.pcount=2;connection.args1=ShowOwnerDetail=" + str +

";connection.args2=ShowBuilderDetail=true";

ReportTemplate.ReBind(ReBindTypeConst.Report, param, ";");

} else {

//작성자가 체크 해제되면 작성자 디테일 밴드를 숨김

var param = "connection.pcount=2;connection.args1=ShowOwnerDetail=" + str +

";connection.args2=ShowBuilderDetail=false";

ReportTemplate.ReBind(ReBindTypeConst.Report, param, ";");

}

SCRIPT_END

- ◆ ReBindTypeConst : 보고서의 바인딩 타입에 따라 보고서가 바인딩되는 규칙이 서로 다름

Data	패러미터 값과 패러미터 값으로 인해 변경되는 데이터 셋도 새로 고침
Label	패러미터 라벨의 텍스트만 새로 고침
Report	패러미터 값만 새로 고침하고, 패러미터 값으로 인해 변경되는 데이터 셋은 새로 고침 안 함

- 각 디테일 밴드 별 자바 스크립트 편집 창의 OnBind 이벤트에 스크립트를 작성

SCRIPT_BEGIN .Report1.DummyBand_owner.OnBind

if(This.GetDataSetValue("OZFormParam.ShowOwnerDetail") == "true") {

This.SetEnable(true); //밴드 활성화

} else {

This.SetEnable(false); //밴드 비활성화

}

SCRIPT_END

개발 신청서

작성자 정보

이름 _____

주소 _____

SCRIPT_BEGIN .Report1.DummyBand_builder.OnBind

```
if(This.GetDataSetValue("OZFormParam.ShowBuilderDetail") == "true") {
    This.SetEnable(true);           //밴드 활성화
} else {
    This.SetEnable(false);         //밴드 비활성화
}
```

SCRIPT_END

- 미리보기한 후 담당자 선택에 따라 각 밴드가 동적으로 조회 됨

OZ Report Viewer

개발 신청에 대한 승인 후 담당자. ☒ 소유자 ☒ 작성자

소유자 정보

이름 _____ E-mail _____

주소 _____ 우편 번호 _____

작성자 정보

이름 _____ E-mail _____

주소 _____ 자격증 번호 _____

Ready Message 1 / 1

OZ Report Viewer

개발 신청에 대한 승인 후 담당자. ☐ 소유자 ☒ 작성자

작성자 정보

이름 _____ E-mail _____

주소 _____ 자격증 번호 _____

개발에 대해 설명 바랍니다. _____

Ready Message 1 / 1

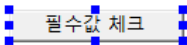
(8) 필수값 체크 버튼에 대한 설정

- 각 텍스트 박스마다 FormID 설정하고, 필수값을 체크하는 버튼의 자바 스크립트 편집창의 OnCheckValidity 이벤트에 스크립트를 작성 함

- FormID 설정

예)

이름	ApplicantName	전화번호	ApplicantPhone	개발비용	Cost
----	---------------	------	----------------	------	------



SCRIPT_BEGIN .Report1.Button2.OnCheckValidity

```

if(GetInputValue("ApplicantName") == "") {
    _MessageBox("신청인 정보의 이름을 작성하세요.", "OZ eForm", function(){
        GetInputComponent("ApplicantName").EnsureVisible();
        GetInputComponent("ApplicantName").SetFocus();
    });
    return false;
}
if(GetInputValue("ApplicantPhone") == "") {
    _MessageBox("신청인 정보의 전화번호를 작성하세요.", "OZ eForm", function(){
        GetInputComponent("ApplicantPhone").EnsureVisible();
        GetInputComponent("ApplicantPhone").SetFocus();
    });
    return false;
}
if(GetInputValue("Cost") == "") {
    _MessageBox("개발 비용을 작성하세요.", "OZ eForm", function(){
        GetInputComponent("Cost").EnsureVisible();
        GetInputComponent("Cost").SetFocus();
    });
    return false;
}
_MessageBox("Okey~!");
SCRIPT_END

```

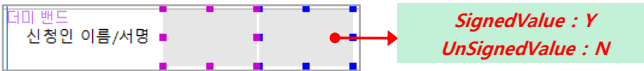
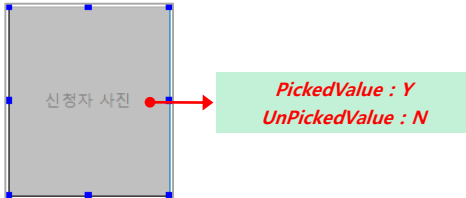
- 미리보기 시 반영되는지 확인

필수값 중 빈 값이 있으면, 아래처럼 팝업 창이 발생되고 해당 컴포넌트로 스크롤 이동 및 포커스를 설정 함

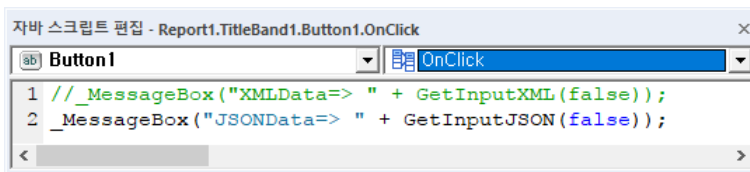


(9) JSON 버튼에 대한 설정

- 사진과 서명의 있을 경우 Y, 없을 경우 N을 보여주도록 아래와 같이 속성을 설정 함



- JSON 버튼의 자바 스크립트 편집창의 OnClick 이벤트에 스크립트를 작성 함



- 미리보기 시 반영되는지 확인



컴포넌트 이동, 메시지 박스, 콤보 박스 기능을 익힙니다.

보고서의 초기 데이터를 패러미터나 DataService로부터 가져와서 출력하는 방법을 습득합니다.

가. OZ e-Form 컴포넌트

❖ 콤보 박스

다수의 아이템을 리스트 형태로 나타내는 콤보 박스를 나타낼 때 사용하는 컴포넌트입니다.

■ 주요 속성

- item 값의 포맷 : text,value\text,value\... or text\text\...
- 입력 값을 가져오기 : text, value
- 데이터셋을 이용하여 아이템 리스트를 가져올 수 있음
- 1번째 콤보 박스에서 아이템을 선택하면 2번째 콤보 박스의 아이템 리스트가 변경 됨 : ReBind

❖ 컴포넌트 간 이동

특정 컴포넌트간 이동 : 이전 FormID, 다음 FormID

❖ 메시지 박스

메시지 확인 창을 띄웁니다.

■ 주요 속성

- _MessageBox : 메시지 확인 창을 띄웁니다.
- _ChoiceButtonBox : 확인/취소 또는 예/아니오를 선택할 수 있는 메시지 창을 띄웁니다.

❖ Functions

공통 함수를 *.OZR 파일에 정의하여 사용합니다.

ReportTemplate 의 Functions 이벤트에 입력된 사용자 정의 함수는 리포트 템플릿에 추가된 모든 컴포넌트에서 호출할 수 있습니다.

나. OZ e-Form 데이터

❖ 패러미터로 데이터 넘기기

- 폼 패러미터 : 보고서(*.ozr) 파일로 넘기는 패러미터
- ODI 패러미터 :오즈 서버가 데이터를 가져올 때 조건을 넘기는 경우 사용

❖ DataService

쿼리 디자이너에서 추가한 데이터 셋은 오즈 서버가 데이터를 조회하며, e-Form 디자이너에서 추가한 데이터 셋은 오즈 뷰어가 직접 데이터를 조회합니다.

- CSV, JSON 데이터셋을 생성할 수 있음

가. 목표 보고서 : 주문서

주문 양식서

The diagram shows an order form with the following sections and callouts:

- 고객 정보 (Customer Information):**
 - 이름 (Name): Callout 3
 - 직위 (Position):
 - 회사명 (Company Name):
 - 주소 (Address):
 - 우편번호 (Postcode): Callout 1
 - 핸드폰 (Mobile Phone):
 - 생년월일 (Date of Birth): Callout 2
 - E-mail:
- 배송 정보 (Delivery Information):**
 - 배송 관련 요청사항 (Delivery request): Callout 4
 - 배송 주소 (주소 정보가 다를 경우 기재 바랍니다) (Delivery address (Please specify if different from the above)):
 - 이름 (Name):
 - 주소 (Address):
 - 핸드폰 (Mobile Phone):
 - 우편번호 (Postcode):
- 제품 타입 (Product Type):** Callout 6
- 품목 (Item):** Callout 7
- 결제 정보 (Payment Information):**
 - 무통장 입금 (Bank Transfer): Callout 9
 - 신용카드 (Credit Card):
 - 카드 번호 (Card Number):
 - 유효 날짜 (Valid Date): Callout 12
 - CVC 번호 (CVC Number):
 - 이름 (Name):
 - 서명 (Signature):
- 오른쪽 하단 완료 버튼 (Bottom Right Complete Button):** Callout 13

나. 과제

- 우편번호, 핸드폰, 생년월일은 1 자릿수마다 입력 박스 별도 추가하고 1 자리 입력하면 다음 칸으로 자동 이동
- 생년월일에 포커스가 가면 포맷(YYMMDD)이 일괄로 없어져야 함
- 고객정보의 이름, 직위는 뷰어 툴바에서 패러미터 값으로 받아서 출력(OZFormParam, 조회툴바)
- 배송정보의 배송 관련 요청사항, 주소의 경우 첫 번째 라인 입력이 채워지면 다음 라인으로 자동 이동
- 화면에 보이는 순서대로 탭 이동(이전 FormID, 다음 FormID)
- 제품 타입 콤보 박스에 JSON 데이터셋의 데이터 값을 출력
- 품목 콤보 박스에 JSON 데이터셋의 데이터 값을 출력
- 제품 타입의 값이 변경되면 관련된 품목이 자동 조회되어 품목 콤보 박스에 보여짐
- 결제 정보의 무통장 입금, 신용카드는 둘 중에 하나만 선택 가능
- 신용카드를 선택하면 하위 정보(콤보박스, 카드번호, 유효 날짜, CVC 번호)를 작성할 수 있도록 활성화하고 무통장 입금을 선택하면 신용카드의 하위 정보를 비활성화 함
- 결제 정보의 무통장 입금 선택 시 신용카드의 정보(콤보박스, 카드번호, 유효 날짜, CVC 번호)는 초기화
- 유효 날짜에 포커스가 가면 포맷(MM YY)이 일괄로 없어져야 함
- 오른쪽 하단 완료 버튼 클릭하면 "저장하시겠습니까?" 확인 메시지 나온 후 '예' 할 때만 입력 데이터 보여줌

(1) 1 자리 입력하면 자동으로 다음 칸으로 이동되도록 설정

- ReportTemplate의 자바 스크립트 편집창의 Functions 이벤트에 공통함수를 작성

```
SCRIPT_BEGIN .ReportTemplate.Functions
//Max length까지 입력되면 지정한 컴포넌트로 포커스 이동
//TextBox.OnValueChanged : autoTab(This, "nextComponentName", 1);
var autoTab = function(object, nextName, maxLength){
    if(isNaN(maxLength)) maxLength = 1;
    if(object.GetText().length == maxLength){
        object.GetInputComponent(nextName).SetFocus();
    }
}
```

- 우편번호, 핸드폰, 생년월일의 텍스트 박스를 한 자릿수마다 디자인 함
- 각 텍스트 박스마다 FormID, 최대 글자 수(1), 키보드 타입(숫자), 텍스트 필터 이름(숫자 텍스트 필터)의 속성을 변경

- FormID 설정

예)

우편번호	p1~p6	핸드폰	t1~t11	생년월일	b1~b8
------	-------	-----	--------	------	-------

- 각 텍스트 박스마다 자바 스크립트 편집 창 의 OnValueChanged 이벤트에 Functions 이벤트에 정의된 공통함수를 적용

The screenshot shows the '속성' (Properties) panel on the left and the '자바 스크립트 편집' (JavaScript Editor) on the right. In the properties panel, the '동적' (Dynamic) section has several properties highlighted with red boxes: '최대 글자 수' (Max Length) set to 1, '텍스트 입력 모드' (Text Input Mode) set to '키보드' (Keyboard), '키보드 타입' (Keyboard Type) set to '숫자' (Number), '텍스트 필터 이름' (Text Filter Name) set to 'NumberTextFilter1', and 'FormID' set to 'p1'. In the script editor, the 'OnValueChanged' event for 'TextBox_p1' is selected, and the script '1 autoTab(This, "p2");' is entered.

- ④ 과제도 공통함수를 사용하여 구현할 수 있음

The screenshot shows the '속성' (Properties) panel on the left and the '자바 스크립트 편집' (JavaScript Editor) on the right. In the properties panel, the '동적' (Dynamic) section has several properties highlighted with red boxes: '최대 글자 수' (Max Length) set to 18, '텍스트 입력 모드' (Text Input Mode) set to '키보드' (Keyboard), '키보드 타입' (Keyboard Type) set to '기본' (Default), '리턴키 타입' (Return Key Type) set to 'Default', '패스워드 문자 표시' (Password Character Display) set to '아니오' (No), '텍스트 필터 이름' (Text Filter Name) set to 'NumberTextFilter1', and 'FormID' set to 'parcel1'. In the script editor, the 'OnValueChanged' event for 'TextBox11' is selected, and the script '1 autoTab(This, "parcel2", 18);' is entered.

(2) 생년월일에 포커스가 가면 포맷(YYMMDD)이 일괄로 없어지도록 설정

- ReportTemplate의 자바 스크립트 편집창의 Functions 이벤트에 공통함수를 작성

SCRIPT_BEGIN .ReportTemplate.Functions

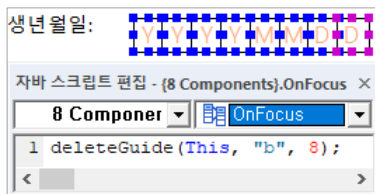
... (1) 번 스크립트

//prefix '1'부터 'lastIndex'의 입력 컴포넌트 까지 포맷 안내 문구 삭제

//TextBox.OnFocus : deleteGuide(This, "b", 8);

```
var deleteGuide = function(object, prefix, lastIndex) {
    if(isNaN(object.GetText())) {
        for(var i=1; i <= lastIndex; i++) {
            object.GetInputComponent(prefix+i).SetText("");
            object.GetInputComponent(prefix+i).SetTextColor("0,0,0");
        }
    }
}
```

SCRIPT_END



- 생년월일의 각 텍스트 박스의 자바 스크립트 편집 창의 OnFocus 이벤트에 공통함수를 적용

(3) 화면에 보이는 순서대로 탭 이동 설정

- 입력 컴포넌트는 보고서에 추가된 순서대로 이동 됨
- 입력컴포넌트의 FormID를 설정한 경우 이전/다음 FormID 설정을 통해 해당 컴포넌트로의 이동이 가능 함

생년월일:

E-mail:

FormID	b8
이전 FormID	
다음 FormID	email

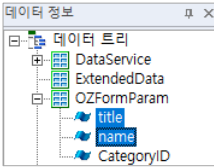
주소:

핸드폰:

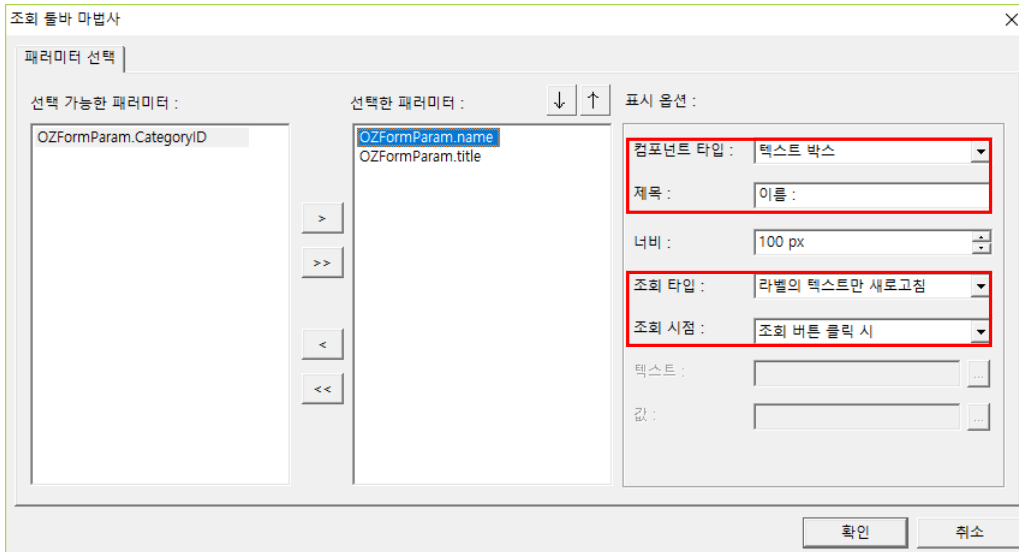
FormID	dAddress2
이전 FormID	
다음 FormID	h1

(4) 고객정보의 이름, 직위는 뷰어 툴바에서 패러미터 값으로 받아서 출력

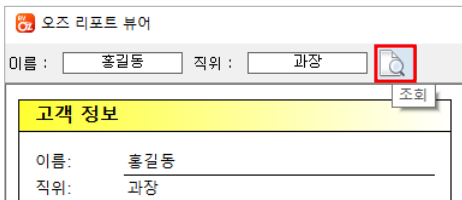
- 데이터 정보 창 내의 OZFormParam에 폼 패러미터 필드를 생성



- 컴포넌트 창 내에 ParameterToolbar1의 팝업 메뉴에서 '조회 툴바 마법사' 를 실행
- 마법사에서 뷰어 툴바에 보여줄 패러미터를 '선택한 패러미터' 로 이동한 후 각 패러미터에 대한 표시 옵션을 지정함



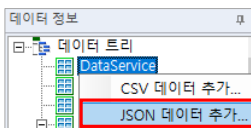
- 미리보기한 후 조회 툴바의 값을 변경하면 보고서의 데이터가 갱신 됨



(5) 제품 타입에 따른 품목 콤보박스 리스트가 변경되도록 설정

- 아래와 같이 두 개의 데이터 셋 정보를 가진 데이터 파일(OrderForm.json)로 JSON 데이터 셋을 추가 함

```
{
  "Category": [
    {"CATEGORYID": 1, "CATEGORYNAME": "AUDIO"},
    {"CATEGORYID": 2, "CATEGORYNAME": "CAMERA"},
    {"CATEGORYID": 3, "CATEGORYNAME": "TV"},
    {"CATEGORYID": 4, "CATEGORYNAME": "VIR"}
  ],
  "Order": [
    {"PRODUCTID": "SY-235-KQ", "MANUFACTURER": "SONY", "BARCODE": 846442, "PRODUCTNAME": "SY-235-KQ", "CATEGORYNAME": "AUDIO", "CATEGORYID": 1},
    {"PRODUCTID": "OP-643-KK", "MANUFACTURER": "LG", "BARCODE": 880123, "PRODUCTNAME": "OP-643-KK", "CATEGORYNAME": "AUDIO", "CATEGORYID": 1},
    ..... (생략)
  ]
}
```



- 데이터 정보 창의 DataService 팝업 메뉴 중 'JSON 데이터 추가' 메뉴를 선택하여 JSON 데이터셋을 설정 함

JSON 데이터 셋 추가

디자인 타임 데이터 셋 구성

☒ 파일이나 사용자 어플리케이션에서 가져오기(URL)

ozp://eForm/data/OrderForm.json 경로...

런 타임 데이터 셋 적용 방법

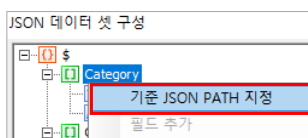
☒ 파일 경로 혹은 사용자 어플리케이션 주소(URL) ☐ 패러미터(경로) ☐ 패러미터(데이터)

ozp://eForm/data/OrderForm.json 경로... 패러미터...

JSON 데이터 셋 구성

JSON_Set1 셋 추가 셋 구성

- 디자인과 런 타임할 때의 데이터 셋을 구성



- 기본으로 추가되어 있는 JSON_Set1을 선택한 후 팝업 메뉴를 통해 셋 구성을 선택 함



- Category에 대해 기본 JSON PATH로 지정

기본 JSON PATH

\$.Category

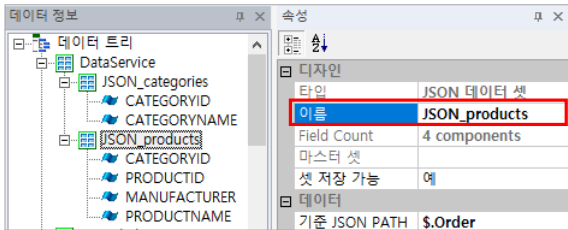
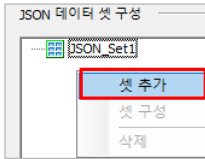
선택된 필드

	필드 이름	필드 타입	JSON PATH
1	CATEGORYID	Double	\$.Category.CATEGORYID
2	CATEGORYNA...	VarChar	\$.Category.CATEGORYNAME

- 추가할 필드를 선택한 후 팝업 메뉴를 통해 필드 추가를 함

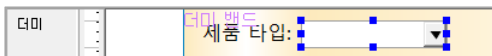
- 기본 JSON PATH에 대한 선택된 필드 리스트가 옆의 이미지처럼 변경 됨

- Order 항목에 대해서도 JSON 데이터 셋 구성에서 아래와 같이 '셋 추가' 메뉴를 선택하여 새로운 JSON 데이터 셋을 추가 함



- JSON 데이터 셋의 속성 창에서 '이름' 변경

- 제품 타입의 콤보 박스에 목록이 조회될 수 있게 자바 스크립트 편집 창의 OnBind 이벤트에 스크립트를 작성

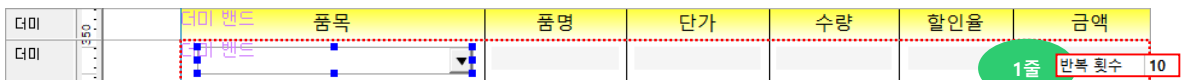


SCRIPT_BEGIN .Report1.ComboBox1.OnBind

```
var items = new Array();
for(var i=1; i <= GetDataSetRowCount("JSON_categories"); i++) {
    items[items.length] = GetDataSetValue("JSON_categories.CATEGORYNAME", i) + "," +
    GetDataSetValue("JSON_categories.CATEGORYID", i);
}
This.SetText(items.join("\n"));
```

SCRIPT_END

- 품목의 콤보 박스에 목록이 조회될 수 있게 자바 스크립트 편집 창의 OnBind 이벤트에 스크립트를 작성



SCRIPT_BEGIN .Report1.ComboBox2.OnBind

```
var items = new Array();
for(var i=1; i <= GetDataSetRowCount("JSON_products"); i++) {
    if(GetDataSetValue("OZFormParam.CategoryID") == GetDataSetValue("JSON_products.CATEGORYID", i)) {
        items[items.length] = GetDataSetValue("JSON_products.PRODUCTNAME", i) + "," +
        GetDataSetValue("JSON_products.PRODUCTID", i);
    }
}
This.SetText(items.join("\n"));
```

SCRIPT_END

- 제품 타입에 따른 품목 콤보 박스의 목록이 변경될 수 있도록 아래와 같이 설정
- 데이터 정보 창 내에 OZFormParam에 제품타입 용 품 패러미터 필드를 생성

패러미터 설정

Member(M) :

- title
- name
- CategoryID

CategoryID 속성(P):

- 디자인
 - 타입: 품 패러미터 필드
- 데이터
 - 필드 이름: CategoryID
 - 필드 타입: Varchar

- 제품 타입의 OnValueChanged 이벤트에 자바 스크립트를 작성
- 제품 타입용 품 패러미터 값에 따라 ReBind 메소드를 통해 품목 리스트를 재조회 함

Script Editor

Script

Script Name: CategoryID

Script Content:

```
SCRIPT_BEGIN .Report1.ComboBox1.OnValueChanged
var val = GetInputValue(GetFormID());
if(val != "") {
    var param = "connection.pcount=1;connection.args1=CategoryID="+val;
    ReportTemplate.ReBind(ReBindTypeConst.Data, param, ";");
}
SCRIPT_END
```

```
SCRIPT_BEGIN .Report1.ComboBox1.OnValueChanged
var val = GetInputValue(GetFormID());
if(val != "") {
    var param = "connection.pcount=1;connection.args1=CategoryID="+val;
    ReportTemplate.ReBind(ReBindTypeConst.Data, param, ";");
}
SCRIPT_END
```

- 미리보기한 후 제품 타입에 따른 품목 리스트가 동적으로 변경 됨

오즈 리포트 뷰어

이름 : [] 직위 : []

제품 타입: AUDIO

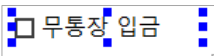
품목	품명	단가	수량	할인율	금액
SY-235-KQ					
OP-643-KK					
JK-745-GG					

(6) 결제정보 중 신용카드를 선택하면 하위 정보들을 작성할 수 있도록 활성화 설정

- 신용카드의 하위 정보 입력 컴포넌트의 Enable 속성을 '아니오'로 설정



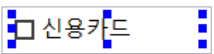
- 무통장 입금 선택 시 신용카드 하위 정보의 내용을 초기화하고 입력 컴포넌트를 비활성화
- 자바 스크립트 편집 창의 OnValueChanged 이벤트에 아래 스크립트를 작성하여 구현 함



SCRIPT_BEGIN .Report1.RadioButton1.OnValueChanged

```
if(This.IsChecked()){
    //무통장 입금 선택 시
    //신용카드 하위 정보의 내용을 초기화
    GetInputComponent("cardtype").SetSelectedIndex(-1);
    GetInputComponent("cardnumber").SetText("");
    GetInputComponent("Exp1").SetText("MM");
    GetInputComponent("Exp1").SetTextColor("255, 172, 132");
    GetInputComponent("Exp2").SetText("YY");
    GetInputComponent("Exp2").SetTextColor("255, 172, 132");
    GetInputComponent("cvc").SetText("");
    //신용카드 하위 정보를 비활성화 함
    GetInputComponent("cardtype").SetEnable(false);
    GetInputComponent("cardnumber").SetEnable(false);
    GetInputComponent("Exp1").SetEnable(false);
    GetInputComponent("Exp2").SetEnable(false);
    GetInputComponent("cvc").SetEnable(false);
}
SCRIPT_END
```

- 신용카드 선택 시 신용카드 하위 정보의 입력 컴포넌트를 활성화
- 자바 스크립트 편집 창의 OnValueChanged 이벤트에 아래 스크립트를 작성하여 구현 함



SCRIPT_BEGIN .Report1.RadioButton2.OnValueChanged

```
if(This.IsChecked()){
    //신용카드 선택 시
    //신용카드 하위 정보를 활성화 함
    GetInputComponent("cardtype").SetEnable(true);
    GetInputComponent("cardnumber").SetEnable(true);
    GetInputComponent("Exp1").SetEnable(true);
    GetInputComponent("Exp2").SetEnable(true);
    GetInputComponent("cvc").SetEnable(true);
}
SCRIPT_END
```

(7) 완료 버튼 클릭 시 메시지 창 띄운 후 '예' 할 때만 입력 컴포넌트의 데이터를 보여주도록 설정

- 완료 버튼의 자바 스크립트 편집 창의 OnClick 이벤트에 _ChoiceButtonBox 메소드를 사용하여 스크립트를 작성 함

SCRIPT_BEGIN .Report1.Button1.OnClick

```
_ChoiceButtonBox("저장하시겠습니까?", "저장 여부 확인", MessageBoxButtonsConst.YesNo,
MessageBoxOptionsConst.IconQuestion, function(UserActionResult){
    if(UserActionResult) { // Yes
        var jsonString = GetInputJSON(false);
        _MessageBox("JSONData=> " + jsonString);
        Global._TriggerOCXUserEvent(jsonString);
    } else { // No
    }
});
```

SCRIPT_END

- 미리보기 한 후 메시지 창에서 '예' 버튼을 클릭 했을 때 JSON 데이터가 보여지는지 확인 함



공통함수를 파일로 관리하고, 컴포넌트의 값을 변경한 직후 다른 컴포넌트의 값이 변경되도록 합니다.
 입력 컴포넌트의 값을 JSON 문자열로 뷰어 패러미터를 통해 전달합니다.
 FormID 속성 값을 배열로 작성하여 추출된 입력 데이터 값이 배열로 넘어오는지 확인합니다.

가. OZ e-Form 컴포넌트

❖ viewer.external_functions_path

공통 함수를 각 *.OZR 파일에 정의하지 않고, 외부 파일로 정의하여 여러 개의 *.OZR 파일에서 사용할 수 있습니다.

뷰어 패러미터로 공통함수의 파일 경로를 설정하고, 보고서가 바인딩될 때 해당 함수를 로딩합니다.

e-Form 디자이너에서는 아래와 같이 미리보기 옵션 설정을 통해 공통 함수를 사용할 수 있습니다.

● 예.

```
viewer.external_functions_path=ozp://eForm/04.견적서.js
```

❖ connection.inputjson

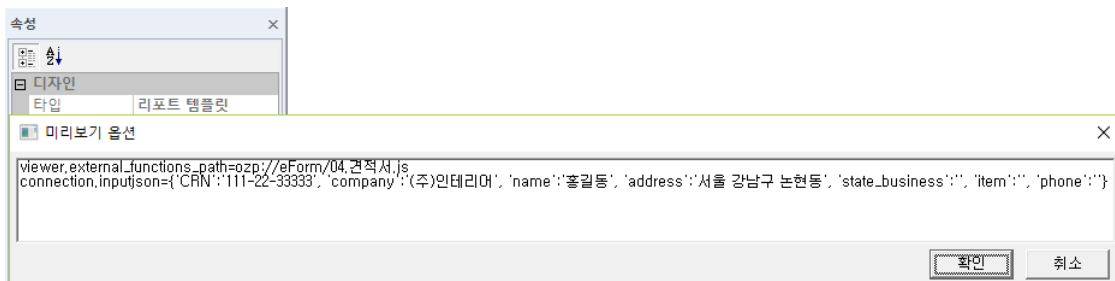
입력 컴포넌트의 값을 뷰어 패러미터로 설정할 수 있습니다.

보고서가 바인딩될 때 패러미터로 설정한 값이 입력 컴포넌트 값으로 적용되며 입력 컴포넌트의 값은 JSON 형식의 문자열로 'FormID':'컴포넌트 값' 형태로 설정하면 됩니다.

e-Form 디자이너에서는 아래와 같이 미리보기 옵션 설정을 통해 공통 함수를 사용할 수 있습니다.

● 예.

```
connection.inputjson={'CRN':'111-22-33333', 'company':'(주)인테리어', 'name':'홍길동',  
'address':'서울 강남구 논현동', 'state_business':'', 'item':'', 'phone':''}
```



❖ 자바 스크립트 이벤트

- OnKillFocus : 컴포넌트에 있던 포커스가 사라질 때 발생하는 이벤트
- OnBind : 컴포넌트가 바인딩 될 때 발생하는 이벤트
- OnEndBind : 컴포넌트가 바인딩 완료된 직후에 발생하는 이벤트

❖ FormID

더미 밴드의 반복 횟수 값이 2 이상일 때 더미 밴드 내의 입력 컴포넌트의 FormID 속성 값을 배열로 작성할 수 있습니다.

- 예.

- FormID 속성 값 : `item[].name_product`
- 자바스크립트(OnBind) : `This.SetFormID("item["+index+"].name_product");`

가. 목표 보고서 : 견적서

No.

견 적 서

작성일자 :

귀하

아래와 같이 견적합니다.

공 급 자	등록번호	111-22-33333		
	상호(법인명)	(주)인테리어	성명	홍길동
	사업장주소	서울 강남구 논현동		
	업태	종목		
	전화번호			

⑦	합 계 금 액 (공급가액 + 세액)	638,000 원 (₩ 638,000)					
	품명	규격	수량	단가	공급가액	세액	비고
	벽타일	250*400	10	15,000	150,000	15,000	
	바닥타일	200*200	3	10,000	30,000	3,000	⑤
	세면기		1	100,000	100,000	10,000	
	인건비		3	100,000	300,000	30,000	
					0		
					0		
					0		
	합 계				580,000	58,000	⑥

⑫

③

⑨

⑩

나. 과제

- ① 공급자 항목의 각 입력 컴포넌트 데이터(등록번호, 상호, 성명 등)를 JSON 형식의 문자열로 뷰어 패러미터를 통해 전달하도록 설정(connection.inputjson)
- ② 품명 리스트 건수를 폼 패러미터 필드로 제어(SetRepeatNumber)
- ③ 품명, 규격, 수량, 단가, 공급가액, 세액, 비고 별로 FormID를 배열로 설정하여 각 리스트마다 서로 다른 FormID 값을 부여할 수 있도록 설정(예. item[].name_product)
- ④ 수량, 단가는 숫자만 입력 가능(숫자 텍스트 필터)
- ⑤ 품명 리스트의 공급가액은 수량 X 단가, 세액은 (수량 X 단가) X 0.1 로 계산하고 원단위 미만 반올림
- ⑥ (하단) 합계의 공급가액과 세액은 이전까지의 공급가액과 세액을 더해줌
- ⑦ 합계금액은 ⑥번의 공급가액과 세액을 더해줌
- ⑧ ⑤~ ⑦번은 품명 리스트의 수량이나 단가를 작성할 때 동작되도록 설정(OnKillFocus, OnEndBind)
- ⑨ '품목 추가' 버튼을 클릭하면 기존 품명 리스트 건수의 마지막 행에 하나를 더해서 보여줌
- ⑩ '품목 삭제' 버튼을 클릭하면 기존 품명 리스트 건수의 마지막 행을 삭제해서 보여줌
- ⑪ 수량과 금액과 관련된 숫자는 천 단위마다 콤마가 보여지도록 함(_Format)
- ⑫ '입력 데이터 조회' 버튼을 클릭하여 팝업 창으로 데이터를 확인(_MessageBox)

※ 공통함수(/eForm/04.견적서.js)의 내용

함수	설명
_getSubTotal	공급가액(품명 별 단일)
_getTotalPrice	공급가액(합계)
_getSubTax	세액(품명 별 단일)
_getTax	세액(합계)
_getSum	합계(원)

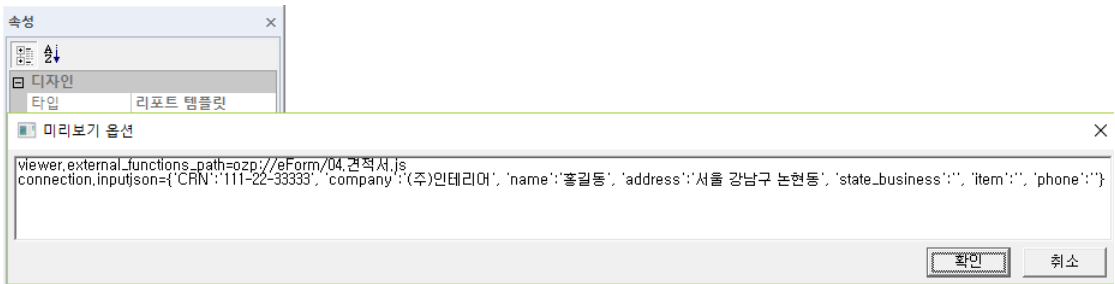
다. 구현 방법

(1) 공급자 항목의 값을 JSON 문자열로 뷰어 패러미터를 통해 전달하도록 설정

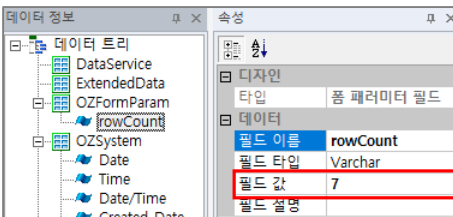
- 공급자 항목 별 서로 다른 FormID를 설정(아래는 예시)

등록번호	상호(법인명)	성명	사업자주소	업태	종목	전화번호
CRN	company	name	address	state_business	item	phone

- 리포트 템플릿의 '미리보기 옵션' 속성에 아래 내용을 작성
- connection.inputjson='{ "CRN": "111-22-33333", "company": "(주)인테리어", "name": "홍길동", "address": "서울 강남구 논현동", "state_business": "", "item": "", "phone": "" }'



(2) 품명 리스트 건수를 폼 패러미터 필드로 제어



- 데이터 정보 창 내의 OZFormParam에 더미 밴드의 반복 횟수를 설정할 필드를 생성 함
- 해당 폼 패러미터 필드의 '필드 값' 속성의 값에 따라 미리보기 시 보여지는 더미 밴드의 수가 결정 됨

- 더미 밴드 내 품명 리스트를 보여줄 표는 한 줄만 디자인 하고, 자바 스크립트 편집 창 의 OnBind 이벤트에 아래 스크립트를 작성

더미	더미 밴드	품명	규격	수량	단가	공급가액	세액	비고
더미	더미 밴드							

1 줄

```
SCRIPT_BEGIN .Report1.DummyBand6.OnBind
```

```
var cnt = This.GetDataSetValue("OZFormParam.rowCount");
```

```
This.SetRepeatNumber(cnt);
```

```
SCRIPT_END
```

//폼 패러미터 값

//반복 횟수 설정

(3) 품명 항목 별로 서로 다른 FormID를 배열로 설정



- 품명 리스트를 보여주는 더미 밴드의 인덱스 값을 정의할 전역 변수를 선언

- 품명 항목 별 서로 다른 FormID를 배열로 설정(아래는 예시)

더미	...	더미 밴드	품명	규격	수량	단가	공급가액	세액	비고
더미	...	더미 밴드							

품명	규격	수량	단가	공급가액	세액	비고
item[].name_product	item[].standard	item[].amount	item[].price_unit	item[].price_supply	item[].price_tax	item[].note

- 품명 항목 별로 아래 스크립트와 같이 FormID의 배열 내 [index] 값을 반영

더미 밴드	품명	규격	수량	단가	공급가액	세액
더미 밴드						

SCRIPT_BEGIN .Report1.FixedTableTextBox30.OnBind

```
var index = 0;
if(GetGlobal("rowIndex") != null)index = GetGlobal("rowIndex");
This.SetFormID("item["+index+"].name_product");
SCRIPT_END
```

- 품명 리스트가 추가될 때 반영할 인덱스 값을 증가시킴

규격	수량	단가	공급가액	세액	비고

SCRIPT_BEGIN .Report1.FixedTableTextBox36.OnBind

```
var index = 0;
if(GetGlobal("rowIndex") != null)index = GetGlobal("rowIndex");
This.SetFormID("item["+index+"].note"); //현재 index 값으로 FormID를 설정
SetGlobal("rowIndex" , parseInt(index)+1); //index 값 증가하여 전역 변수로 선언
SCRIPT_END
```

(4) 품명 리스트의 수량이나 단가를 작성할 때 금액이 계산되도록 설정

- 수량과 단가의 자바 스크립트 편집 창 내에 OnKillFocus, OnEndBind 이벤트에 자바 스크립트를 작성

품명	규격	수량	단가	공급가액

자바 스크립트 편집 - [2 Components].OnKillFocus

2 Components OnKillFocus

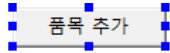
```
var index = This.GetFormID();
index = index.replace(/[^0-9]/g, "");
```

```
_getSubTotal("amount","price_unit","price_supply",index); //공급가액(품명별 단일)
_getTotalPrice("amount", "price_unit", "supply_total"); //공급가액(합계)
_getSubTax("amount","price_unit","price_tax",index); //세액(품명별 단일)
_getTax("amount", "price_unit", "tax_total"); //세액(합계)
_getSum("supply_total", "tax_total", "price_total_01"); //합계(원)
_getSum("supply_total", "tax_total", "price_total_02"); //합계
```

- 공통 함수에 대해 정의되어 있는 외부 파일 내 함수를 사용하여 ⑤~ ⑦ 번의 과제를 구현
- 리포트 템플릿의 '미리보기 옵션' 속성에 viewer.external_functions_path= ozp://eForm/04.견적서.js 를 작성

(5) 품목 추가/삭제 버튼

- 품목 추가 버튼을 클릭 했을 때 더미 밴드 하단에 밴드가 추가되도록 스크립트를 작성



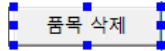
```

SCRIPT_BEGIN .Report1.Button2.OnBind
This.SetPrintable(false);           //버튼의 프린터 인쇄 및 저장을 하지 않을 경우
SCRIPT_END

SCRIPT_BEGIN .Report1.Button2.OnClick
var num = parseInt(This.GetDataSetValue("OZFormParam.rowCount"));
num = num+1;
var param;
param = "connection.pcount=1;connection.args1=rowCount="+num;
ReportTemplate.ReBind("Data", param, ";");
SCRIPT_END

```

- 품목 삭제 버튼을 클릭 했을 때 더미 밴드 맨 마지막 행이 삭제되도록 스크립트를 작성



```

SCRIPT_BEGIN .Report1.Button1.OnBind
This.SetPrintable(false);           //버튼의 프린터 인쇄 및 저장을 하지 않을 경우
SCRIPT_END

SCRIPT_BEGIN .Report1.Button1.OnClick
var num = parseInt(This.GetDataSetValue("OZFormParam.rowCount"));
num = num-1;

if(num == 0){
    This.GetInputComponent("supply_total").SetText(0);
    This.GetInputComponent("tax_total").SetText(0);
    This.GetInputComponent("price_total_01").SetText(0);
    This.GetInputComponent("price_total_02").SetText(0);
}

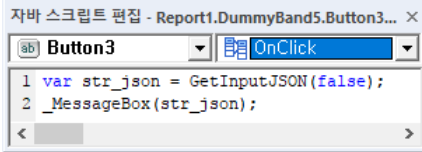
if(num < 0){
    _MessageBox("더 이상 삭제할 품목 리스트가 존재하지 않습니다.");
    num = 0;
}

var param;
param = "connection.pcount=1;connection.args1=rowCount="+num;
ReportTemplate.ReBind("Data", param, ";");
SCRIPT_END

```

(6) 입력 데이터 조회 버튼

- 자바 스크립트 편집 창의 OnClick에 스크립트를 작성



- 미리보기 시 아래 이미지와 같이 뜬 팝업 창을 통해 품명 리스트의 데이터를 조회



시스템 필드(OZSystem) 사용방법에 대해 익히고, OZFormParam의 값에 따라 표시 여부를 설정하고 고정테이블의 셀을 재배치하는 스크립트를 실습합니다.

가. OZ e-Form 컴포넌트

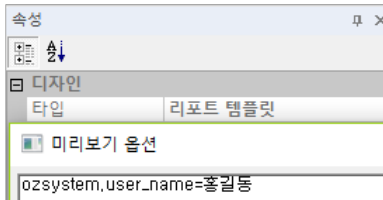
❖ ozsystem.user_name

시스템 필드 중 User_Name(사용자 이름)과 Creator_Name(보고서를 생성한 사용자 이름) 값을 설정합니다.

Ozsystem.user_name 패러미터 값을 'A'로 설정한 상태에서 OZR 파일을 미리보기 한 후 OZD 파일로 저장한 후, ozsystem.user_name 패러미터 값을 'B'로 변경하고 OZD 파일을 미리보기하면 Create_Name은 'A'로 User_Name은 'B'로 표기 됩니다.

E-Form 디자이너에서는 아래와 같이 미리보기 옵션을 통해 시스템 필드 값을 설정할 수 있습니다.

예. ozsystem.user_name=홍길동



❖ 자바 스크립트 이벤트

- setVisible : 표시 여부를 설정
- setPrintable : 출력 여부를 설정
- setLeft : 왼쪽 위치를 설정
- setTop : 위쪽 위치를 설정
- setWidth : 너비를 설정
- setHeight : 높이를 설정

가. 목표 보고서 : 출장 보고서

출장 보고서

② 담당 (부서장 임원 사장)
 (sign2) (sign3) (sign4)
 8 / 14 / / /

③ 출장종

소속부서 마케팅팀 출장자 홍길동

출장지 싱가포르 ⑧ 출장기간 2017-08-14 ~ 2017-08-21

일정별 업무 실적	
일 자	업 무 내 용
2017-08-14	인천국제공항 출발 싱가포르 도착 16:00 : 포시엑스 싱가포르 지사 담당자 미팅
2017-08-15	싱가포르 IT 전시회 참석
2017-08-16	싱가포르 IT 전시회 참석
2017-08-17	싱가포르 IT 전시회 참석
2017-08-18	싱가포르 IT 전시회 참석

일정별 업무 실적

⑨

일 자	업 무 내 용
2017-08-19	주말
2017-08-20	주말
2017-08-21	싱가포르 출발 인천국제공항 도착

⑧

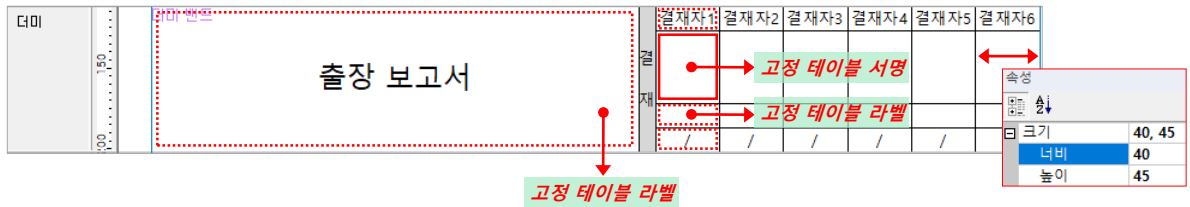
나. 과제

- ① 결재란 열은 '직급, 서명, 이름, 날짜'로 구성되며 폼 패러미터 필드로 보여줄 결재란의 수를 최소 2개에서 최대 6개까지 설정이 가능하도록 하며 결재란 개수에 따라 자동으로 오른쪽으로 정렬되도록 설정(SetVisible, SetPrintable, SetLeft)
- ② 결재란 개수는 폼 패러미터 필드로 4로 설정하고 ' 직급 ' 은 폼 패러미터 필드를 통해 ' 담당,부서장,임원,사장 ' 으로 설정
- ③ 결재란 '이름, 날짜'는 폼 패러미터 필드를 통해 표시 여부를 결정하고, '이름'은 편집이 되도록, '날짜'는 편집이 되지 않도록 속성을 설정(편집 가능)
- ④ 결재란 '서명'을 작성한 직후 '이름'에는 OZSystem.User_Name의 값을, '날짜'에는 오늘 날짜를 'M / d' 패턴으로 자동 표기되도록 설정
- ⑤ 결재란 '이름, 날짜'의 '글꼴 크기' 속성을 10pt으로 설정하고, 위 ④번이 실행될 때 ' 글꼴 크기 ' 가 8pt 가 되도록 스크립트로 표현
- ⑥ 결재란 '서명' 데이터를 초기화하면 '이름, 날짜' 데이터가 초기화 되도록 설정
- ⑦ '출장자'에는 결재란 '담당' 이름과 동일하게 표기되도록 설정(OZSystem.Creator_Name)
- ⑧ '출장기간'에 시작일과 종료일을 설정하면 일정별 업무 실적의 '일자'가 동적으로 변경되도록 설정
- ⑨ 페이지 마다 표의 제목(일정별 업무 실적)을 반복 출력
- ⑩ '출장기간'의 시작일을 종료일보다 큰 값으로 설정할 경우 종료일을 시작일과 동일한 값으로 설정
- ⑪ '출장기간'의 종료일을 시작일보다 작은 값으로 설정한 경우 "시작일보다 이전 날짜는 선택하실 수 없습니다." 팝업창을 띄우고, 종료일을 시작일과 동일한 값으로 설정

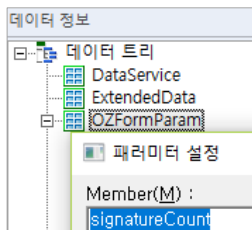
다. 구현 방법

(1) 결재란 개수를 가변적으로 설정하고 그 개수에 따라 자동으로 오른쪽으로 정렬되도록 설정

- 디자인 화면



- 결재란의 개수를 가변적으로 설정할 수 있도록 폼 패러미터 필드를 생성



- 결재란 수를 최소 2개에서 최대 6개까지 설정할 수 있도록 더미 밴드 자바 스크립트 편집 창의 OnBind 이벤트에 스크립트를 작성
- 밴드의 너비와 결재란 개수를 글로벌 변수로 선언

SCRIPT_BEGIN .Report1.DummyBand2.OnBind

```
var bandWidth = This.GetWidth();
SetGlobal("bandwidth",bandWidth);
```

//밴드의 너비를 글로벌 변수로 선언

//결재란 개수를 가변적으로 설정할 변수

```
var signatureCount = parseInt(This.GetDataSetValue("OZFormParam.signatureCount"));
```

```
if(signatureCount < 2){
```

```
    signatureCount =2;
```

//결재란 최소 2개

```
}else if(signatureCount > 6 || isNaN(signatureCount)){
```

```
    signatureCount =6;
```

//결재란 최대 6개까지 설정

```
}
```

```
SetGlobal("signatureCount",signatureCount);
```

//결재란 개수를 글로벌 변수로 선언

- 각 결재란 셀의 OnBind 이벤트에 자바 스크립트를 작성하여 자동으로 오른쪽으로 정렬되도록 설정

	결재자1	결재자2	결재자3	결재자4	결재자5	결재자6
직급						

```

SCRIPT_BEGIN .Report1.FixedTableSignPad7.OnBind
switch(GetGlobal("signatureCount")){
    case 2:
        This.SetLeft(GetGlobal("bandwidth")-80);
        break;
    case 3:
        This.SetLeft(GetGlobal("bandwidth")-120);
        break;
    case 4:
        This.SetLeft(GetGlobal("bandwidth")-160);
        break;
    case 5:
        This.SetLeft(GetGlobal("bandwidth")-200);
        break;
    case 6:
        This.SetLeft(GetGlobal("bandwidth")-240);
        break;
    default :
        break;
}

```

```

This.SetText("(sign1)");
This.SetBackgroundColor("246, 246, 246");
This.SetTextColor("196, 189, 151");
SCRIPT_END

```

	결재자1	결재자2	결재자3	결재자4	결재자5	결재자6
직급						

```

SCRIPT_BEGIN .Report1.FixedTableLabel17.OnBind
if(GetGlobal("signatureCount")<4){
    This.SetVisible(false);
    This.SetPrintable(false);
}else{
    switch(GetGlobal("signatureCount")){
        case 4:
            This.SetLeft(GetGlobal("bandwidth")-40);
            break;
        case 5:
            This.SetLeft(GetGlobal("bandwidth")-80);
            break;
        case 6:
            This.SetLeft(GetGlobal("bandwidth")-120);
            break;
        default :
            break;
    }
}

```

```

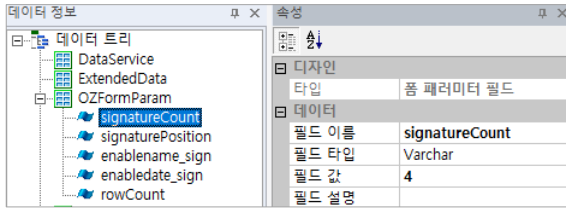
This.SetText("(sign4)");
This.SetBackgroundColor("246, 246, 246");
This.SetTextColor("196, 189, 151");
SCRIPT_END

```

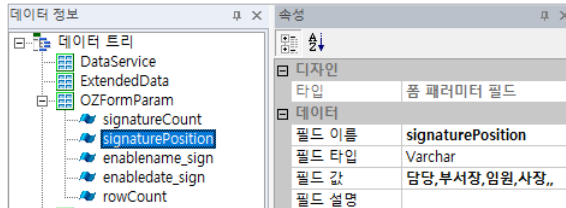
※ 결재란 개수에 따라 너비와 위치 조정이 필요한 그 외의 항목들(보고서명과 결재란의 '직급, 이름, 날짜')에 대해서도 자동으로 너비 조정 및 위치 정렬이 되도록 자바 스크립트를 작성 해야 함

(2) 결재란 개수와 '직급'을 폼 패러미터 필드로 가변적으로 설정

- 결재란의 개수와 직급을 가변적으로 설정할 수 있도록 폼 패러미터 필드를 생성



- 각 폼 패러미터 필드의 '필드 값' 속성 내에 왼쪽 이미지와 같이 테스트 값을 작성



- 밴드의 자바 스크립트 편집 창 내에 직급에 대해 폼 패러미터로 전달 받은 값을 배열과 글로벌 변수로 선언

SCRIPT_BEGIN .Report1.DummyBand2.OnBind

... (1) 스크립트

//직급명에 대해 배열로 선언 및 글로벌 변수로 선언

var str_position = This.GetDataSetValue("OZFormParam.signaturePosition");

var temp = new Array();

temp = str_position.split(",");

```
for(var i=0; i<6; i++){
    if(temp[i] == null){
        temp[i] = "approval" + (i+1);
    }
}
```

SetGlobal("position1",temp[0]);

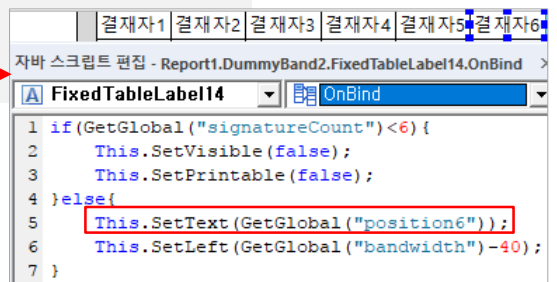
SetGlobal("position2",temp[1]);

SetGlobal("position3",temp[2]);

SetGlobal("position4",temp[3]);

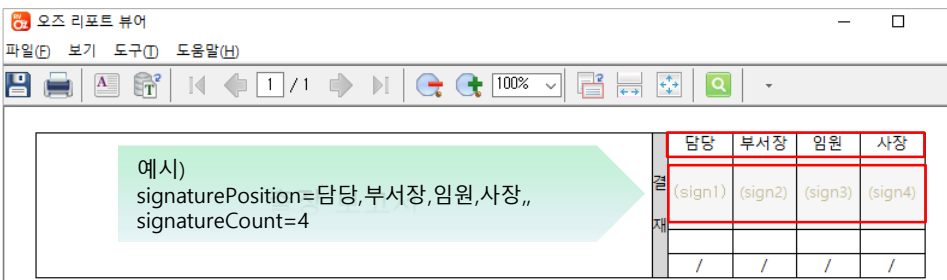
SetGlobal("position5",temp[4]);

SetGlobal("position6",temp[5]);

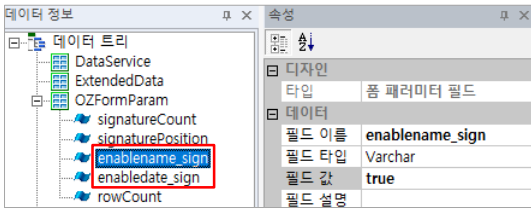


- 직급이 표현되는 각 셀에 글로벌 변수의 값을 가져와 SetText()로 표현

- 폼 패러미터 필드로 설정한 값이 미리보기 시 반영되는지 확인



(3) 결재란 '이름, 날짜'는 폼 패러미터 필드로 표시 여부를 결정하고, '이름'은 편집이 되도록 설정



- 결재란의 이름, 날짜의 표시 여부를 결정할 폼 패러미터 필드를 생성

- 밴드의 자바 스크립트 편집 창 내에 이름, 날짜에 대해 폼 패러미터로 전달 받은 값을 글로벌 변수로 선언

SCRIPT_BEGIN .Report1.DummyBand2.OnBind

.. '결재란 관련 스크립트'

var enablename = This.GetDataSetValue("OZFormParam.enablename_sign");

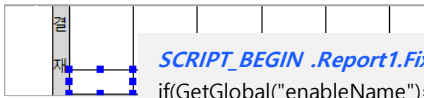
enablename= enablename.toLowerCase();

SetGlobal("enableName",enablename);

var enabledate = This.GetDataSetValue("OZFormParam.enabledate_sign");

enabledate= enabledate.toLowerCase();

SetGlobal("enableDate",enabledate);



SCRIPT_BEGIN .Report1.FixedTableLabel21.OnBind

if(GetGlobal("enableName")==="false"){

This.SetVisible(false);

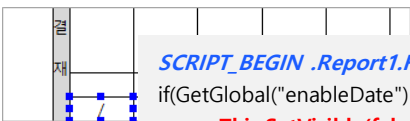
This.SetPrintable(false);

}

..(생략)

- 이름이 표현되는 각 셀의 OnBind 이벤트에 자바 스크립트를 작성

- 폼 패러미터 값이 false일 때 해당 셀을 숨기고 인쇄/저장 되지 않도록 설정



SCRIPT_BEGIN .Report1.FixedTableLabel28.OnBind

if(GetGlobal("enableDate")==="false"){

This.SetVisible(false);

This.SetPrintable(false);

}

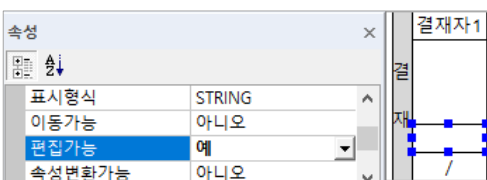
if(GetGlobal("enableName")==="false"){

This.SetTop(60);

}

..(생략)

- 날짜가 표현되는 각 셀의 OnBind 이벤트에 자바 스크립트를 작성



- 결재란 이름의 '편집가능' 속성 값을 '예'로 설정하여 편집이 되도록 설정

- 폼 패러미터 필드로 설정한 값이 미리보기 시 반영되는지 확인

예시)
enablesign_name = true
enablesign_date = true

※ 결재란 이름, 날짜의 표시 여부에 따라 높이, 위쪽 좌표 설정이 필요한 셀에 대해서도 자바 스크립트 작성이 필요 함

(4) 결재란 '서명' 을 작성한 직후 '이름, 날짜' 가 자동으로 표기되도록 설정

서명	signature_01	signature_02	signature_03	signature_04	signature_05	signature_06
이름	name_sign_01	name_sign_02	name_sign_03	name_sign_04	name_sign_05	name_sign_06
날짜	date_sign_01	date_sign_02	date_sign_03	date_sign_04	date_sign_05	date_sign_06

- 서명, 이름, 날짜에 대해 FormID 속성을 설정
- ❖ 입력 컴포넌트와 더불어 라벨과 고정 테이블 라벨에 FormID 설정이 가능

- ReportTemplate의 Functions 이벤트에 공통함수를 작성

SCRIPT_BEGIN .ReportTemplate.Functions

//글짜 크기를 설정

```
var _oz_shrinkFontSizeToFit = function (comp, fontSize) {
    var newSize = fontSize;
    if (newSize < 1) newSize = 1;
    if (newSize != comp.GetFontSize()) {
        comp.SetFontSize(newSize);
    }
};
```

//텍스트를 설정

```
var _oz_setText = function (comp, text, fontSize) {
    if (comp.GetText() == text)
        return;
    comp.SetText(text);
    if (fontSize > 0)
        _oz_shrinkFontSizeToFit(comp, fontSize);
};
```

... (이어서)

//서명 작성 직후 이름이 자동으로 표기될 때 사용

//SignPad.OnValueChanged : _oz_updateCompEditor(This, "name_sign_01", 8);

```
var _oz_updateCompEditor = function (comp, labelName, fontSize) {
    var revision = This.GetDataSetValue("OZSystem.User_Name");
    if (revision)
        _oz_setText(This.GetInputComponent(labelName), revision, fontSize);
    else
        _oz_setText(This.GetInputComponent(labelName), "", 0);
};
```

//서명 작성 직후 날짜가 자동으로 표기될 때 사용

//SignPad.OnValueChanged : _oz_updateCompEditedDate(This, "date_sign_01", "M / d", 8);

```
var _oz_updateCompEditedDate = function (comp, labelName, format, fontSize) {
    var revision = new Date().getTime();
    if (revision) {
        var formattedDate = _FormatDate(revision, format);
        _oz_setText(This.GetInputComponent(labelName), formattedDate, fontSize);
    } else {
        _oz_setText(This.GetInputComponent(labelName), "", 0);
    }
};
```

SCRIPT_END

- 각 결재란 셀의 OnValueChanged 이벤트에 자바 스크립트를 작성하여 '이름, 날짜'가 자동으로 표기되도록 공통 함수를 사용하여 설정
- 글자 크기(8pt) 및 날짜 패턴(M / d)을 설정
- 서명 데이터가 초기화 될 때 이름, 날짜 데이터도 초기화 되도록 설정

	결재자1	결재자2	결재자3	결재자4	결재자5	결재자6
재						

SCRIPT_BEGIN .Report1.FixedTableSignPad7.OnValueChanged

if(GetInputValue(GetFormID()) == ""){ //서명 데이터가 초기화 될 때

GetInputComponent("name_sign_01").SetText("");

GetInputComponent("date_sign_01").SetText("/");

}else{

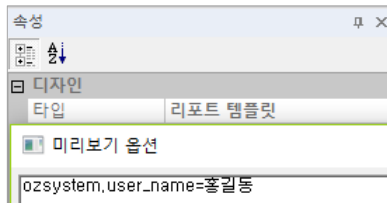
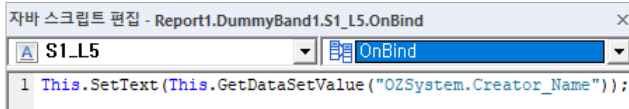
_oz_updateCompEditor(This, "name_sign_01", 8);

_oz_updateCompEditedDate(This, "date_sign_01", "M / d", 8);

}

SCRIPT_END

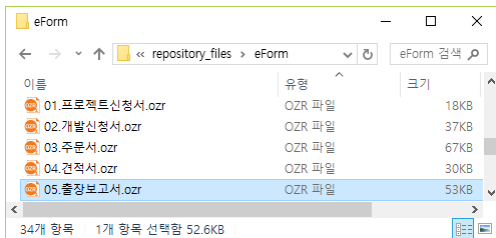
(5) '출장자'에 결재란의 '담당' 이름과 동일하게 표기되도록 설정



- '출장자'에 OZSystem.Creator_Name을 표기
- 결재란의 '담당'에 OZSystem.User_Name을 표기

- 리포트 템플릿의 '미리보기 옵션' 속성에 'ozsystem.user_name=홍길동'로 설정하여 디자이너에서 확인

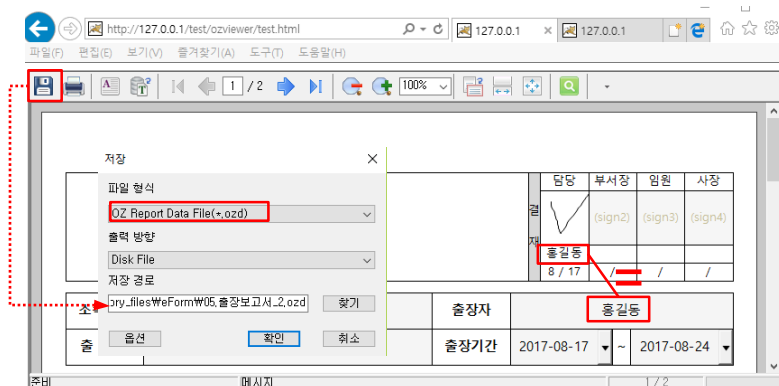
❖ 보고서 업로드 후 뷰어를 통해 확인하는 방법



- %OZ_HOME%/repository_files 폴더 내에 보고서 파일을 업로드 함

connection.servlet=http://127.0.0.1:80/test/server
connection.reportname=eForm/05.출장보고서.ozr
ozsystem.user_name=홍길동

- ozsystem.user_name 패러미터를 설정한 후 OZR 파일(connection.reportname 패러미터 사용)을 미리보기하면, '출장자'에 반영, 결재란의 '담당' 이름에도 동일하게 반영 됨
- 뷰어 내 '저장' 메뉴를 통해 OZD 확장자로 파일을 저장



```
connection.servlet=http://127.0.0.1:80/test/server
connection.openfile=ozp://eForm/05.출장보고서_2.ozd
ozsystem.user_name=홍길순
```

- ozsystem.user_name 패러미터를 변경한 후 저장한 OZD 파일(connection.openfile 패러미터 사용)을 미리보기하면, '출장자'에 그 이전 값이, 결재란의 '담당' 이름에는 변경된 값이 반영 됨

(6) '출장기간'에 시작일과 종료일을 설정하면 일정별 업무 실적의 '일자'가 동적으로 변경되도록 설정

- 일정별 업무 실적의 '일자'가 동적으로 변경될 때 더미 밴드의 반복 횟수 값을 지정할 수 있도록 패러미터 필드를 생성하고 자바 스크립트 편집 창의 OnBind 이벤트에 해당 필드를 글로벌 변수로 선언 함

```
SCRIPT_BEGIN .Report1.OnBind
```

```
SetGlobal("rowCount",0);
```

```
SCRIPT_END
```

- 출장기간의 시작일과 종료일과 관련하여, FormID 설정 및 자바 스크립트 편집 창의 OnValueChanged 이벤트에 다음의 스크립트를 각각 작성 함
- 아래는 FormID 예시

시작일	fromDate	종료일	toDate
-----	----------	-----	--------

SCRIPT_BEGIN .Report1.S1_L6.OnValueChanged

```
var input1 = This.GetValue();
var input2 = This.GetInputComponent("toDate").GetValue();
var date1 = new Date(input1.substr(0,4),input1.substr(5,2)-1,input1.substr(8,2));
var date2 = new Date(input2.substr(0,4),input2.substr(5,2)-1,input2.substr(8,2));
```

```
var interval = date2 - date1;
var day = 1000*60*60*24;
var days = parseInt(interval/day)+1;
```

```
if(interval < 0){ //시작일을 종료일보다 큰 값으로 설정할 경우 종료일을 시작일과 동일한 값으로 설정
    This.GetInputComponent("toDate").SetValue(This.GetValue());
    days = 1;
}
```

```
var param;
param = "connection.pcount=2;connection.args1=rowCount="+days;
```

```
ReportTemplate.ReBind("Data", param, ";");
```

SCRIPT_END

SCRIPT_BEGIN .Report1.FixedTableTextBox2.OnValueChanged

```
var input1 = This.GetInputComponent("fromDate").GetValue();
var input2 = This.GetValue();
var date1 = new Date(input1.substr(0,4),input1.substr(5,2)-1,input1.substr(8,2));
var date2 = new Date(input2.substr(0,4),input2.substr(5,2)-1,input2.substr(8,2));
```

```
var interval = date2 - date1;
var day = 1000*60*60*24;
var days = parseInt(interval/day)+1;
```

```
if(interval < 0){ //종료일을 시작일보다 작은 값으로 설정한 경우 팝업창 실행 및 시작일과 동일한 값으로 설정
    _MessageBox("시작일보다 이전 날짜는 선택하실 수 없습니다.");
    This.SetValue(input1);
    days = 1;
}
```

```
var param;
param = "connection.pcount=2;connection.args1=rowCount="+days;
```

```
ReportTemplate.ReBind("Data", param, ";");
```

SCRIPT_END

- 더미 밴드 내 표는 한 줄만 디자인 함

SCRIPT_BEGIN .Report1.FixedTableDateTimePicker1.OnBind

```
var index = 0;
if(GetGlobal("rowCount") != null)index = GetGlobal("rowCount");
This.SetFormID("work["+index+"].date");
```

SCRIPT_END

- ❖ OnEndBind 이벤트는 모든 컴포넌트의 OnBind 이벤트가 모두 완료된 후 발생하는 이벤트

SCRIPT_BEGIN .Report1.FixedTableDateTimePicker1.OnEndBind

```
var input1 = This.GetInputComponent("fromDate").GetValue();
var date1 = new Date(input1.substr(0,4),input1.substr(5,2)-1,input1.substr(8,2));
var day = 1000*60*60*24;
```

```
index = This.GetFormID();
index = index.replace(/^[^0-9]/g,"");
var date2 = new Date(Date.parse(date1) + (index)*day);
var year = date2.getFullYear();
```

```
if(date2.getMonth() < 9 ){
    var month = '0'+(date2.getMonth()+1);
}else{
    var month = date2.getMonth()+1;
}
```

```
if(date2.getDate() < 9 ){
    var day = '0'+date2.getDate();
}else{
    var day = date2.getDate();
}
```

```
var date = year+"-"+month+"-"+day;
This.SetValue(date);
```

SCRIPT_END

고정 테이블 텍스트 박스
work[].contents

SCRIPT_BEGIN .Report1.FixedTableTextBox3.OnBind

```
var index = 0;
if(GetGlobal("rowCount") != null)index = GetGlobal("rowCount");
This.SetFormID("work["+index+"].contents");
SetGlobal("rowCount" , parseInt(index)+1);
```

SCRIPT_END

(7) 페이지 마다 표의 제목(일정별 업무 실적)을 반복 출력

- 페이지 헤더 밴드로 구현하고 '첫 페이지 출력생략' 속성의 값을 '예' 로 설정

Chapter3. 서버 설치 및 뷰어 테스트

1. 서버 및 뷰어 설치
2. 보고서 업로드
3. 오즈 뷰어 호출



가. 사전 준비

- 1) jdk 설치 : jre 1.5 이상이 설치되어 있어야 OZ 서버 7.0 버전을 설치할 수 있습니다.
- 2) apache-tomcat 설치
- 3) OZ Enterprise Server 7.0 설치

나. 오즈 서버 설치

- 1) tomcat의 webapps 폴더 내에 oz 서버를 붙여넣기 하기

C:\Program Files (x86)\Forcs\OZ Enterprise Server 7.0\OZ Servlet 7.0 폴더를 복사하여 C:\Program Files (x86)\Tomcat7.0\webapps 폴더 밑으로 붙여넣기한 후 폴더명을 **eform**으로 변경합니다.

- 2) OZ 서버 라이선스 반영

C:\Program Files (x86)\Tomcat8.0\webapps\eform\WEB-INF\license 폴더 밑에 ozlicense.xml 파일을 붙여넣기 합니다.

- 3) tomcat 재구동 후 OZ 콘솔 화면 확인

- ① 웹 브라우저에서 oz 콘솔 URL(예 : <http://127.0.0.1:8080/eform/server>)을 실행하고 Username, Password를 임의로 설정한 후 로그인 합니다.

(보고서를 oz 서버에 업로드 할 때 Username, Password 정보를 사용 합니다.)

- ② OZ 서버가 정상적으로 구동되는지 여부는, OZ 콘솔 페이지 내에 OZ 버전 정보 및 시스템 정보가 조회되는지 여부로 파악이 가능합니다.

[참고] PC의 IP 확인하는 방법?

cmd 창에서 ipconfig 명령어를 실행하여 확인이 가능합니다.

다. 오즈 뷰어 설치

- 1) OZ 서버 폴더 내에 오즈 뷰어를 붙여넣기 하기


C:\Program Files (x86)\Forcs\OZ Enterprise Server 7.0\OZ Report ActiveX Viewer 7.0 폴더를 복사하여 C:\Program Files (x86)\Apache Software Foundation\Tomcat 8.0\webapps\eform 폴더 밑으로 붙여넣기한 후 폴더명을 **ozviewer**로 변경합니다.

가. 리파지토리의 정의

‘리파지토리’는 보고서와 관련된 파일들(ozr, odi, 이미지 등)을 보관해 둔 저장소입니다.

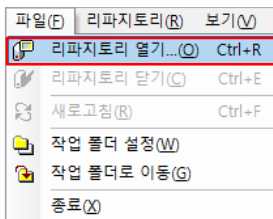
✓ 오즈 서버 리파지토리 기본 경로 : %OZ_HOME%/repository_files/

나. 리파지토리 서버 연결

1) 바탕 화면의 오즈 리파지토리 매니저 아이콘  을 실행합니다.

- 실행 파일 경로 : %ProgramFiles%\WFORCS\OZ e-Form 7.0\OZ Repository Manager 7.0\bin\OZRepositoryManager.exe

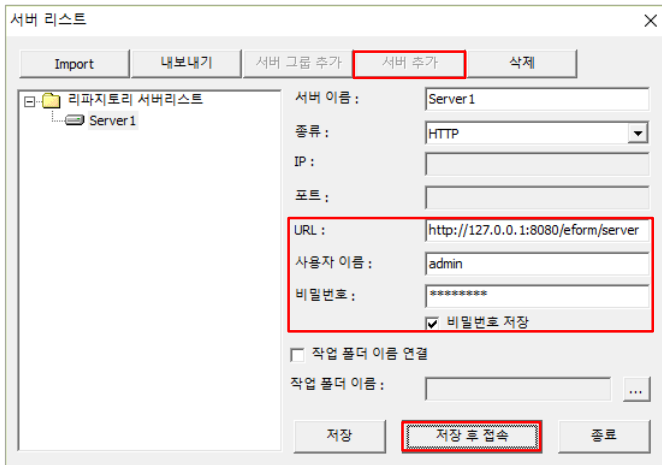
2) 파일 > 리파지토리 열기 메뉴를 클릭하면 ‘서버 리스트 다이얼로그’가 실행 됩니다.



3) ‘서버 추가’ 버튼을 클릭하여 새로운 서버 리파지토리를 추가 합니다.

4) URL과 사용자 이름, 비밀번호를 입력 후 ‘저장 후 접속’ 버튼을 클릭합니다.

5) 서버 리파지토리 연결이 완료되면 연결된 리파지토리 서버의 카테고리화 아이টে이 표시 됩니다.

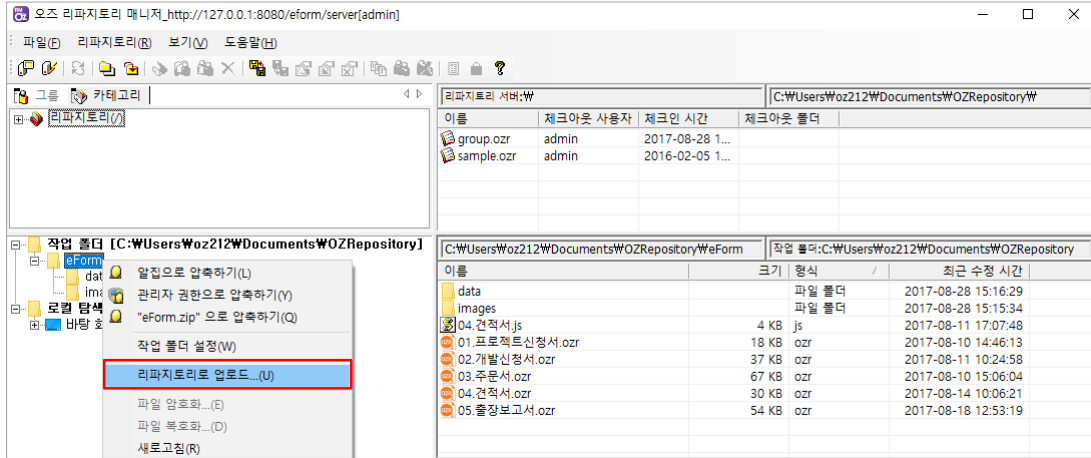


6) 서버 리파지토리 연결이 완료되면 연결된 리파지토리 서버의 카테고리화 아이টে이 표시 됩니다.

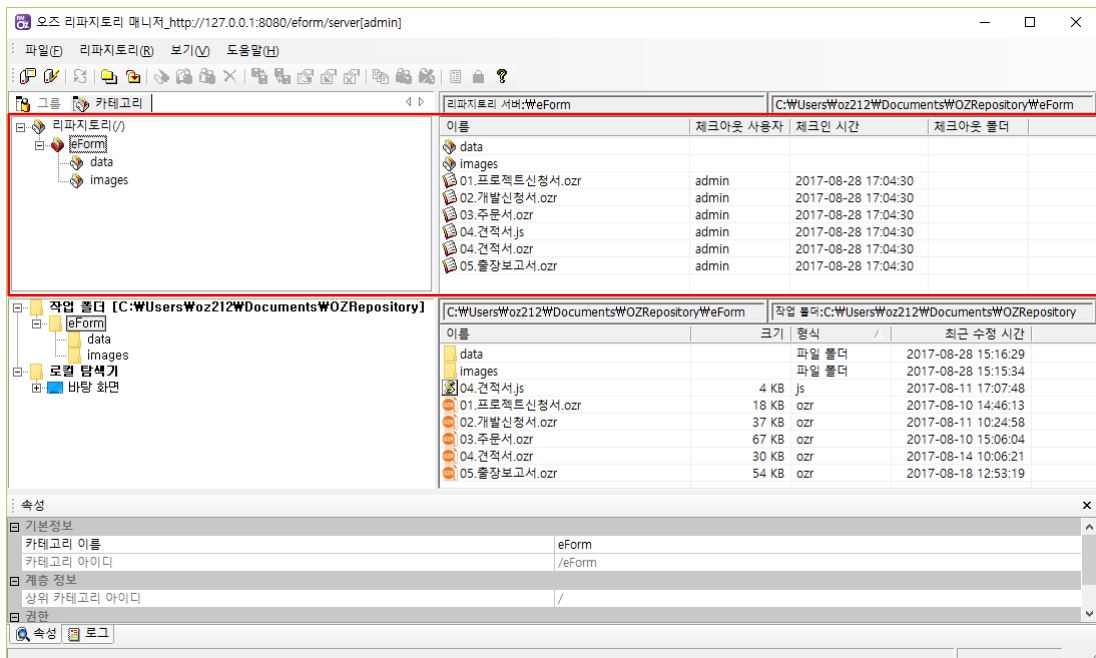
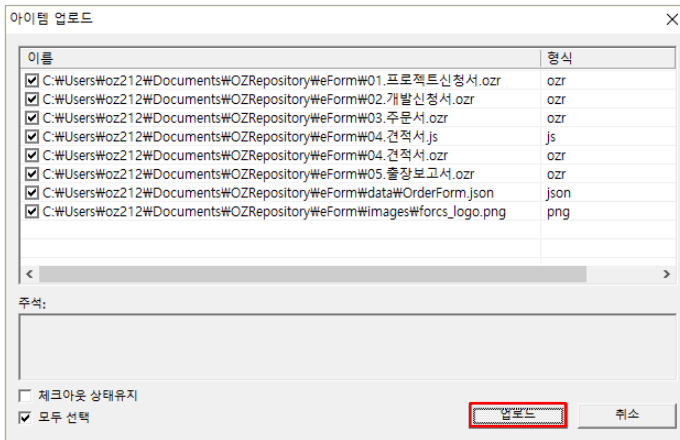
2. 보고서 업로드

다. 보고서를 리파지토리에 업로드/다운로드

- 1) 작업 폴더에서 업로드할 폴더나 파일을 선택한 후 마우스 오른쪽 버튼을 클릭하여 팝업 메뉴에서 '리파지토리 업로드'를 선택합니다.



- 2) '아이템 업로드'에서 파일을 선택한 후 하단의 '업로드' 버튼을 클릭하면 OZ 서버 내 해당 파일이 업로드 됩니다.



오즈 뷰어를 호출하는 샘플 웹페이지를 수정하여 e-Form 보고서를 실행하는 실습을 진행합니다.

❖ e-Form 뷰어 패러미터

- comment.all : 편집 모드 또는 주석 모드 아이콘 표시 여부를 설정
- pdf.savecomment : PDF 파일로 저장할 때 주석을 포함하여 저장할지 여부를 설정

❖ e-Form 뷰어 함수

1) GetInformation : 오즈 뷰어의 정보를 가져옵니다.

- INPUT_JSON_ALL : 전체 보고서의 입력 값을 합쳐서 하나의 JSON 형식 텍스트로 가져옴
- INPUT_CHECK_VALIDITY_AT=보고서 인덱스(0 인덱스부터 시작 됨)

: 지정한 보고서에서 OnCheckValidity 이벤트에 스크립트가 입력된 첫 번째 버튼을 찾아

OnCheckValidity 이벤트에 입력된 스크립트 실행 결과를 가져옴,

실행 성공인 경우 "valid"로 리턴 되고 실행 실패 또는 잘못된 보고서 인덱스를 설정한 경우 ""로 리턴 됨

2) Script : 오즈 뷰어 툴바 아이콘의 기능을 제어합니다.

- disable_input_all : 모든 보고서의 모든 입력 컴포넌트를 비활성화
- enable_input_all : 모든 보고서의 모든 입력 컴포넌트를 활성화

❖ 멀티보고서

- 여러 보고서를 한 뷰어에 보여주는 방식
- viewer.childcount : 하나의 오즈 뷰어에 여러 개의 보고서를 보여줄 때 메인 보고서 외에 추가되는 차일드 보고서의 개수를 설정
- 추가되는 차일드 보고서에 대한 패러미터 설정은 패러미터 그룹명 앞에 child#.(# : 차일드 보고서 번호)을 붙여줌

❖ 뷰어 함수와 이벤트

1) 함수

- Script => save_memorystream : Memory Stream으로 저장,
저장된 결과는 OZExportMemoryStreamCallBack 이벤트에서 사용 가능

2) 이벤트

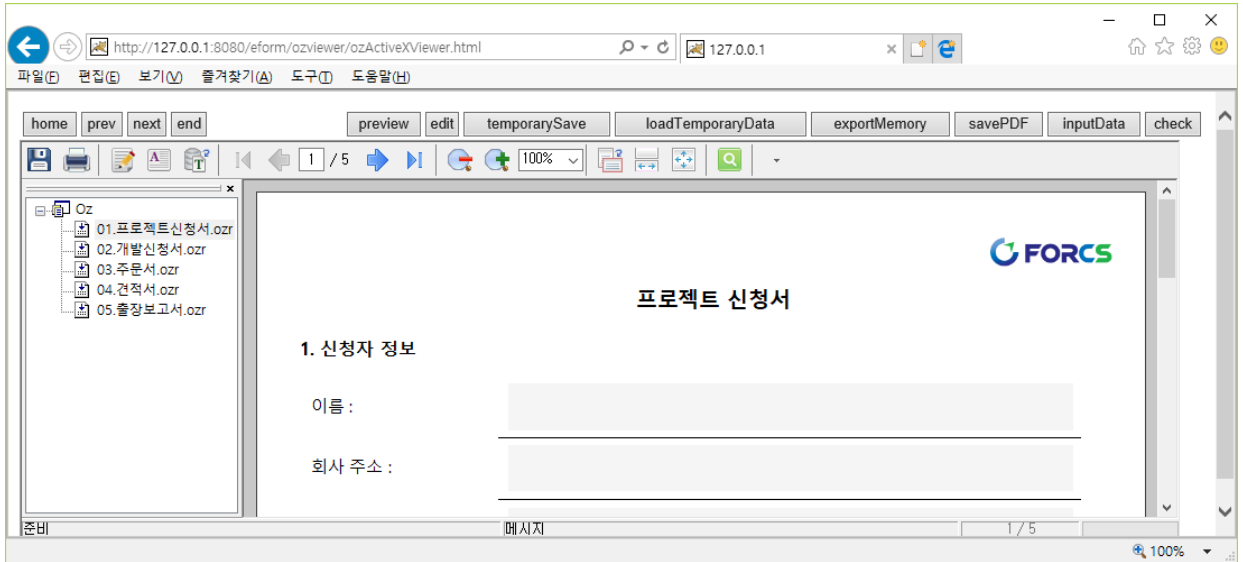
- OZExportMemoryStreamCallBack : Script, ScriptEx 함수의 명령어를 "save_memorystream"로 설정하여 실행한 경우 발생하는 이벤트, outputdata는 {"파일 전체 경로":"base64로 인코딩되어 저장된 파일의 Memory Stream"} 형식의 문자열로 리턴 됨

❖ 오즈 뷰어 도움말 활용하기

뷰어 설치와 실행하는 방법 및 뷰어 호출 시 필요한 패러미터, 함수, 이벤트 등에 대한 사용방법을 숙지할 수 있습니다.

- 오즈 e-Form 디자이너 -> 도움말 메뉴 -> 오즈 뷰어 도움말 선택

가. 샘플 웹페이지



나. 과제

- ① 실습한 보고서들을 멀티 보고서로 호출되도록 함
- ② 주석 모드 아이콘(📝)이 표시되도록 하고, 보고서에 주석을 작성한 후 주석이 포함되어 pdf 파일이 저장되도록 설정(comment.all, pdf.savecomment)
- ③ 개발신청서 보고서에 그룹 서명이 되는지 확인(eform.signpad_type)
- ④ 견적서 보고서에 connection.inputjson 뷰어 패러미터로 넘긴 값이 공급자 항목에 적용되도록 함
- ⑤ 출장 보고서에 connection.args# 뷰어 패러미터로 넘긴 값이 '결재'란에 반영되도록 설정
- ⑥ 출장 보고서에 ozsystem.user_name 뷰어 패러미터로 넘긴 값이 '출장자'에 반영되도록 함
- ⑦ preview 버튼 클릭 시 모든 보고서의 모든 입력 컴포넌트가 비활성화 되도록 설정
- ⑧ edit 버튼 클릭 시 모든 보고서의 모든 입력 컴포넌트가 활성화 되도록 설정
- ⑨ temporarySave 버튼은 뷰어에서 현재 입력컴포넌트의 값들을 임시로 저장할 때 사용하며, 버튼 클릭 시 json 형식으로 추출한 입력컴포넌트의 값을 base64로 인코딩된 값을 리턴
- ⑩ loadTemporaryData 버튼은 임시로 저장한 값을 보고서에 반영할 때 사용하며, 첫 번째 보고서를 통해 확인이 가능 함
- ⑪ exportMemory 버튼은 Memory Stream으로 저장하기 위해 사용하며, outputdata 를 팝업창을 통해 확인
- ⑫ savePDF 버튼을 클릭 시 c:\wtemp 폴더 내에 basic.pdf 파일명으로 저장 함
- ⑬ inputData 버튼을 클릭 시 전체 보고서의 입력 값을 JSON 형식 텍스트로 가져오도록 설정
- ⑭ check 버튼을 클릭 시 '개발 신청서' 보고서에 대해 필수 값을 체크 함

다. ActiveX 뷰어 호출 소스 수정

(1) C:\Program Files (x86)\Apache Software Foundation\Tomcat 8.0\webapps\eform\ozviewer\wrv_install_bind.html 복사하여 ozActiveXViewer.html 파일로 이름 변경하여 사용 함

(2) ozActiveXViewer.html 샘플 소스

● 멀티 보고서 호출 샘플

```
<html style='height:100%'>
<meta http-equiv="Content-Type" Content="text/html; charset=utf-8" />
<head>
  <script language = "JavaScript" src = "http://127.0.0.1:8080/eform/ozviewer/ztransferx_browsers.js"> </script>
  <script language = "JavaScript" src = "http://127.0.0.1:8080/eform/ozviewer/ozviewer_browsers.js"> </script>
  <script language ="JavaScript">
    function ZTInstallEndCommand_ZTransferX(param1,param2) {
      Create_Div();
      Initialize_OZViewer("OZReportViewer", "CLSID:0DEF32F8-170F-46f8-B1FF-4BF7443F5F25", "98%",
"98%", "application/OZRViewer");
      Insert_OZViewer_Param("viewer.childcount", "4");
      Insert_OZViewer_Param("viewer.showtree", "true");
      //멀티 보고서를 하나의 보고서처럼 사용할지 여부를 설정
      Insert_OZViewer_Param("global.concatpage", "true");

      Insert_OZViewer_Param("connection.servlet", "http://127.0.0.1:8080/eform/server");
      Insert_OZViewer_Param("connection.reportname", "eForm/01.프로젝트신청서.ozr");
      //편집 모드 또는 주석 모드 아이콘 표시 여부를 설정
      Insert_OZViewer_Param("comment.all", "true");
      //PDF 파일로 저장할 때 주석을 포함하여 저장할지 여부를 설정
      Insert_OZViewer_Param("pdf.savecomment", "true");

      Insert_OZViewer_Param("child1.connection.servlet", "http://127.0.0.1:8080/eform/server");
      Insert_OZViewer_Param("child1.connection.reportname", "eForm/02.개발신청서.ozr");
      Insert_OZViewer_Param("child1.eform.signpad_type", "dialog"); //그룹 서명
      Insert_OZViewer_Param("child1.comment.all", "true");
      Insert_OZViewer_Param("child1.pdf.savecomment", "true");

      Insert_OZViewer_Param("child2.connection.servlet", "http://127.0.0.1:8080/eform/server");
      Insert_OZViewer_Param("child2.connection.reportname", "eForm/03.주문서.ozr");
      Insert_OZViewer_Param("child2.comment.all", "true");
      Insert_OZViewer_Param("child2.pdf.savecomment", "true");

      Insert_OZViewer_Param("child3.connection.servlet", "http://127.0.0.1:8080/eform/server");
      Insert_OZViewer_Param("child3.connection.reportname", "eForm/04.견적서.ozr");
      Insert_OZViewer_Param("child3.viewer.external_functions_path", "ozp://eForm/04.견적서.js");
      Insert_OZViewer_Param("child3.connection.pcount", "1");
      Insert_OZViewer_Param("child3.connection.args1", "rowCount=10");
      Insert_OZViewer_Param("child3.connection.inputjson", "{ 'CRN': '111-22-33333',
'company': '(주)인테리어', 'name': '홍길동', 'address': '서울 강남구 논현동', 'state_business': '', 'item': '', 'phone': '' }");
      Insert_OZViewer_Param("child3.comment.all", "true");
      Insert_OZViewer_Param("child3.pdf.savecomment", "true");
```

```

Insert_OZViewer_Param("child4.connection.servlet", "http://127.0.0.1:8080/eform/server");
Insert_OZViewer_Param("child4.connection.reportname", "eForm/05.출장보고서.ozr");
Insert_OZViewer_Param("child4.connection.pcount", "5");
Insert_OZViewer_Param("child4.connection.args1", "signatureCount=4");
Insert_OZViewer_Param("child4.connection.args2", "signaturePosition담당,부서장,임원,사장,,");
Insert_OZViewer_Param("child4.connection.args3", "enablename_sign=true");
Insert_OZViewer_Param("child4.connection.args4", "enableddate_sign=true");
Insert_OZViewer_Param("child4.connection.args5", "rowCount=1");
Insert_OZViewer_Param("child4.ozsystem.user_name", "홍길동"); //시스템 필드 값 설정
Insert_OZViewer_Param("child4.comment.all", "true");
Insert_OZViewer_Param("child4.pdf.savecomment", "true");

Insert_OZViewer_Param("viewer.isframe", "false");
Insert_OZViewer_Param("viewer.namespace", "custom_namespaceWwwozviewer");

Start_OZViewer();
}
</script>

```

● 함수 테스트

```

<script language="JavaScript">
function home() {
    OZReportViewer.Script("home"); //첫 페이지로 이동
}
function prev() {
    OZReportViewer.Script("prev"); //한 페이지 이전으로 이동
}
function next() {
    OZReportViewer.Script("next"); //한 페이지 다음으로 이동
}
function end() {
    OZReportViewer.Script("end"); //마지막 페이지로 이동
}
function preview() {
    OZReportViewer.Script("movereport=0"); //보고서 인덱스로 이동(zero-based index)
    OZReportViewer.Script("movepage=1"); //지정된 페이지 번호로 이동
    OZReportViewer.Script("disable_input_all"); //모든 보고서의 모든 입력 컴포넌트를 비활성화
}
function edit() {
    OZReportViewer.Script("movereport=0");
    OZReportViewer.Script("movepage=1");
    OZReportViewer.Script("enable_input_all"); //모든 보고서의 모든 입력 컴포넌트를 활성화
}
function temporarySave() { //json 추출 값을 base64로 인코딩된 값이 리턴 됨
    temporaryInputValue = OZReportViewer.GetInformation("EFORM_DATA_AT=0");
    alert("임시 저장 ==> " + temporaryInputValue);
}
function loadTemporaryData() {
    OZReportViewer.Script('closeall');
    var param = "";
    param += "viewer.showtree=trueWn";
    param += "connection.servlet=http://127.0.0.1:8080/eform/serverWn";
    param += "connection.reportname=eForm/01.프로젝트신청서.ozrWn";
    param += "connection.eformdata=" + temporaryInputValue + "Wn";
    OZReportViewer.CreateReport(param);
}

```

```

//Memory Stream으로 저장, 저장된 결과는 OZExportMemoryStreamCallBack 이벤트에서 사용 가능
function exportMemory() {
    OZReportViewer.ScriptEx("save_memorystream",
"export.format=ozd;export.path=c:\\\\temp;export.mode=silent;export.filename=test.ozd;export.confirmsave=f
alse",",");
}
function OZExportMemoryStreamCallBack_OZReportViewer(outputdata) {
    if(outputdata == "{}") {
        alert("Fail");
    } else {
        //outputdata는 {"파일 전체 경로":"base64로 인코딩되어 저장된 파일의 Memory Stream"} 형식의 문자열 값
        alert(outputdata);
    }
}
function savePDF() { //pdf 파일로 저장
    OZReportViewer.ScriptEx("save",
"export.format=pdf;export.path=C:\\\\temp;export.mode=normal;export.filename=basic.pdf;export.confirmsave
=false",",");
}
function inputData() { //전체 보고서의 입력 값을 합쳐서 하나의 JSON 형식 텍스트로 가져옴
    var strInputValue = OZReportViewer.GetInformation("INPUT_JSON_ALL");
    alert("JSON ==> " + strInputValue);
}
function check() {
    //개발신청서 보고서에 대해 필수값을 체크 함(버튼의 OnCheckValidity 이벤트를 실행)
    var chkResult = OZReportViewer.GetInformation("INPUT_CHECK_VALIDITY_AT=1");
    alert("chkResult ==> " + chkResult);
}
</script>
</head>

<body style='height:100%'>
<table style="width:100%">
<tr>
<td>
        <input type="button" value="home" onclick="home()">
        <input type="button" value="prev" onclick="prev()">
        <input type="button" value="next" onclick="next()">
        <input type="button" value="end" onclick="end()">
    </td>
<td align="right">
        <input type="button" value="preview" onclick="preview()">
        <input type="button" value="edit" onclick="edit()">
        <input type="button" value="temporarySave" onclick="temporarySave()">
        <input type="button" value="loadTemporaryData" onclick="loadTemporaryData()">
        <input type="button" value="exportMemory" onclick="exportMemory()">
        <input type="button" value="savePDF" onclick="savePDF()">
        <input type="button" value="inputData" onclick="inputData()">
        <input type="button" value="check" onclick="check()">
    </td>
</tr>
</table>

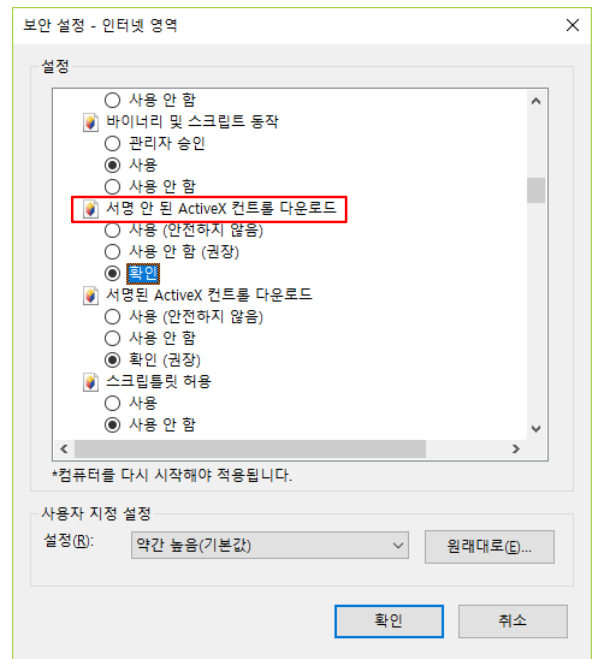
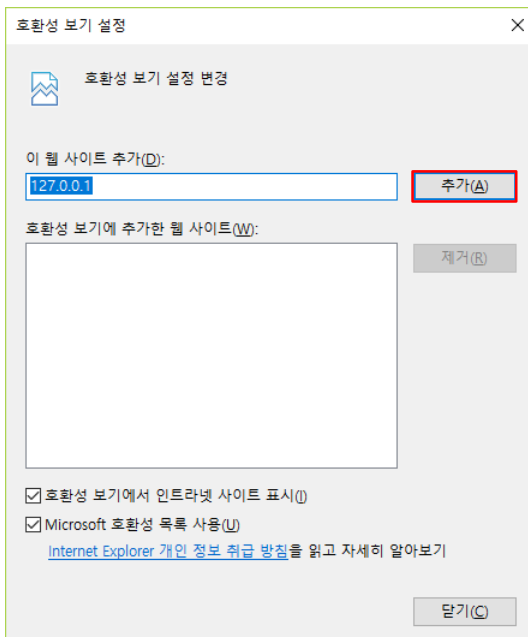
```


● 뷰어 설치

```
<div id = "InstallOZViewer">
<script language = "JavaScript">
    Initialize_ZT("ZTransferX", "CLSID:C7C7225A-9476-47AC-B0B0-FF3B79D55E67", "0", "0",
"http://127.0.0.1:8080/eform/ozviewer/ZTransferX_2,2,5,3.cab#version=2,2,5,3", "application/OZTransferX_1029");
    Insert_ZT_Param("download.server", "http://127.0.0.1/eform/ozviewer/");
    Insert_ZT_Param("download.port", "8080");
    Insert_ZT_Param("download.instruction", "ozviewer.idf");
    Insert_ZT_Param("install.base", "<PROGRAMS>/Forcs");
    Insert_ZT_Param("install.namespace", "custom_namespace");
    Start_ZT();
</script> </div>
</body>
</html>
```

(3) 인터넷 익스플로러에서 아래 URL을 입력하여 테스트를 진행 함

- URL : <http://127.0.0.1:8080/eform/ozviewer/ozActiveXViewer.html>
- ActiveX 뷰어 설치가 정상적으로 되지 않을 경우 아래 내용들을 확인합니다.



- 인터넷 익스플로러의 설정 -> 호환성 보기 설정 -> 이 웹 사이트 추가 항목에 해당 웹페이지를 '추가' 함
- 인터넷 익스플로러의 설정 -> 인터넷 옵션 -> 보안 탭 -> '사용자 지정 수준' 버튼 -> '서명 안 된 ActiveX 컨트롤 다운로드' 항목에 대해 '확인'을 선택 함

(1) 교재 샘플 폴더 구성

'eForm' 폴더는 '내 문서\OZRepository\eForm'에 복사되어야 합니다.

폴더	설명
/eForm/data/	샘플 데이터 (JSON 파일)
/eForm/images/	이미지 파일
/eForm/*.ozr	'Chapter2. Exercise' 관련된 보고서 파일
/eForm/*.js	'Chapter2. Exercise' 관련된 공통함수 파일
/eForm/overview/	'Chapter3. 서버 설치 및 뷰어 테스트' 관련 파일

※ 샘플 파일은 <http://www.oztn.net/OZEdu/OZeFormCourse> 에서 다운로드 받을 수 있습니다.