



# xFrame5

## 다국어 개발 가이드

VERSION 1.1

서울특별시 구로구 디지털로 272, 1110 (구로동, 한신 IT 타워)  
Phone 02-2108-8030 • Fax 02-2108-8031  
[www.softbase.co.kr](http://www.softbase.co.kr)  
Copyright © 2010 SOFTBase Inc. All rights reserved

## 목차

<b>1 장: 다국어 개발 환경</b> .....	<b>4</b>
다국어 관련 개발/실행 환경.....	4
개발 환경 구성 내용.....	4
실행 파라미터 설정.....	5
언어별 사전 파일.....	6
<b>2. 다국어 개발</b> .....	<b>7</b>
메타 적용 방법.....	7
메타 데이터 관련 API.....	8
메타 데이터 로드 및 사용자 환경 설정.....	8
콜백 함수 사용 방법.....	9
실행 환경 파라미터 설정.....	9
콜백 함수 프로토타입 설명.....	9
콜백 함수 프로토타입 예시.....	10
다국어 관련 부가 파라미터 및 API.....	11
파라미터.....	11
API.....	12

# 1 장: 다국어 개발 환경

이 장에서는 다국어 개발 환경에 대한 내용을 설명한다.

- ✓ 다국어 관련 개발/실행 환경 구성
- ✓ 언어별 사전 파일

## 다국어 관련 개발/실행 환경

### 개발 환경 구성 내용

다국어 개발 환경을 구성하기 위해서는 아래와 같은 내용이 필요하다.

구분	S/W	설명
xFrame5 개발 도구	xFrame5@DevStudio	도구->옵션 메뉴의 "메타데이터" 탭 설정 필요
메타 DBMS 연계 모듈	xFrame5Meta.jsp	WAS 서버에 설치되어 xFrame5@DevStudio 와 메타 솔루션 DB 간의 연계 기능을 수행
WAS 서버	J2EE 기반 제품	J2EE 기반 WAS 서버
JDBC Driver	JDBC Driver	메타 솔루션이 사용하는 DB 에 대응되는 JDBC 드라이버
메타 솔루션 DB	DBMS	<ul style="list-style-type: none"> <li>▪ 메타 솔루션이 사용하는 DBMS이며, 언어별 메타ID 에 대응되는 정보가 저장되어 있어야 한다.</li> <li>▪ 메타 파일을 생성하기 위한 기준 데이터를 보관하고 있어야 하며, xFrame5Meta.jsp 에서 접속이 가능해야 한다.</li> </ul>

## 실행 파라미터 설정

다국어 실행 환경을 구성하기 위해서는 아래와 같은 내용이 필요하다.

USER\_LANGUAGE, SCREEN\_LANGUAGE 파라미터 정의

- SCREEN\_LANGUAGE 파라미터는 화면 개발시 사용한 언어 값을 지정한다.
- USER\_LANGUAGE는 사용자가 사용하는 언어 값을 지정한다.

USER\_LANGUAGE 파라미터 값은 메인 화면인 경우에만 xframe5.js 에 설정된 값을 로드하여 사용한다.

factory.setlocale API 를 통해, 엔진에서 사용중인 USER\_LANGUAGE 파라미터를 변경할 수 있다. 변경된 파라미터 값은 새로운 화면을 로드시부터 적용되며, 현재 로드된 화면에는 영향이 없다.

***USER\_LANGUAGE, SCREEN\_LANGUAGE 파라미터가 다른 경우에만 엔진에서 메타 처리를 수행한다.***

## 언어별 사전 파일

언어별 메타 ID 에 해당하는 정보가 포함된 텍스트 파일이며, 언어별로 존재해야 한다. 생성된 메타 파일은 screen.loadmeta API 를 통해서 로드할 수 있다.

### 사전 파일 데이터 구조

META\_ID=META\_INFORMATION

### 사전 파일 규칙

- 메타 파일 내용은 UTF-8 형식으로 저장돼야 한다.
- 각 행은 "wrwn" 문자열로 구분된다.
- "=" 문자 사이에 공백 문자가 없어야 한다.
- META\_INFORMATION에 대한 내용은 "메타 적용 대상 선정 방식"에 따라 다르다.
- 빈 문자열 또는 '#' 문자로 시작되는 행은 무시된다.

## 2. 다국어 개발

이 장에서는 다국어 개발 방식에 대한 내용을 설명한다.

- ✓ 메타 적용 방법
- ✓ 메타 데이터 관련 API
- ✓ 메타 데이터 연계 콜백 방식
- ✓ 메타 관련 부가 파라미터 및 API

### 메타 적용 방법

메타 정보 적용 대상에는 아래와 같은 적용 방식이 있다.

구분	설명	설명
텍스트만 변경	언어별 사전 파일에는 META_ID 별 메타 텍스트 정보만 저장되어 있다.	xFrame5 엔진에서 자동으로 처리한다.
다양한 속성 변경	언어별 사전 파일에 META_ID 별 다양한 속성 정보가 포함되어 있다. 속성 정보 문자열에 여러 정보를 포함하기 위해서는 구분자를 이용한 형식을 주로 사용한다.	콜백 함수를 사용해야 한다. 콜백 함수에 대한 설명은 아래의 "콜백 함수 사용 방법"을 참조한다.

## 메타 데이터 관련 API

### 메타 데이터 로드 및 사용자 환경 설정

메타 데이터 로드 및 사용자 언어 환경 설정을 위해서는 아래의 API 를 사용한다. 자세한 내용은 해당 API 에 대한 도움말을 참조한다.

#### **screen.loadmeta**

메타 데이터 URL 을 파라미터로 받아서 메타 데이터를 로드하여 글로벌 메모리에 로드한다. 글로벌 메모리에 로드한다는 말은 한번 로드된 메타 데이터를 다른 화면에서도 사용할 수 있다는 의미이다.

로드된 메타 정보는 factory.meta~로 시작되는 API 통해서 접근할 수 있다.

#### **factory.setlocale**

사용자 언어 및 국가를 지정하는 동작을 수행한다.



## 콜백 함수 사용 방법

콜백 함수를 사용하는 방법은 아래와 같다.

### 실행 환경 파라미터 설정

콜백함수는 글로벌 모듈에 정의하여 사용한다. 아래의 예시는 "SysUtil"이라는 글로벌 모듈에 "CallbackMetaData" 라는 함수를 사용한다고 설정한 예시이다.

```
// 메타 데이터를 이용하여 컴포넌트 생성시 호출될 글로벌 모듈 콜백
CALLBACK_META_DATA: "SysUtil.CallbackMetaData",
```

### 콜백 함수 프로토타입 설명

콜백 함수의 주 기능은 엔진에서 META\_ID 에 해당하는 메타 데이터 정보를 파라미터로 받아서 파라미터인 컴포넌트 생성시 사용되는 속성 정보 오브젝트의 값을 변경하는 동작을 수행하는 것이다.

```
/**
 * 메타 처리 콜백
 * @param meta_id 메타 ID
 * @param meta_value 메타 ID 에 해당하는 값
 * @param screen_url 화면 URL
 * @param object_kind 오브젝트 유형 (XFD_OBJKIND_CONTROL/XFD_OBJKIND_SHAPE)
 * @param component_kind 오브젝트 유형 (getcontrolkind, getshapekind)
 * 또는 오브젝트 세부 유형
 * @param prop_info 컴포넌트 속성 정보 오브젝트 (key: 속성명, value: 속성값)
 * @param user_language 사용자 실행 환경 언어
 * @param user_country 사용자 실행 환경 국가
 * @returns 1 - 메타 정보 처리함, 0 - 메타 정보 처리 하지 않음(엔진 기본 동작 수행)
 */
function CallbackMetaData(meta_id, meta_value, screen_url, screen_id,
                          object_kind, component_kind,
                          prop_info, user_language, user_country) {
    return 1;
}
```

## 콜백 함수 프로토타입 예시

```
function CallbackMetaData(meta_id, meta_value, screen_url, screen_id,
                          object_kind, component_kind,
                          prop_info, user_language, user_country) {
    var meta_item_arr;

    // 메타 값이 '|' 문자를 구분자로 여러개 구성되어 있는 경우 예시
    meta_item_arr = meta_value.split('|');

    if (object_kind == XFD_OBJKIND_CONTROL) {
        switch (component_kind) {
            case XFD_CTRLKIND_HYPERTEXT:
            case XFD_CTRLKIND_PUSHBUTTON:
            case XFD_CTRLKIND_CHECKBOX:
            case XFD_CTRLKIND_RADIOBUTTON:
                prop_info['text'] = meta_item_arr[0];
                prop_info['tooltip'] = meta_item_arr[0] + '_tooltip';
                break;

            case XFD_CTRLKIND_TABITEM: // 탭 아이템인 경우
                prop_info['title'] = meta_item_arr[0];
                break;
            case XFD_CTRLKIND_GRIDCOLUMNHEAD: // 그리드 컬럼 헤더인 경우
                prop_info['title'] = meta_item_arr[0];
                break;
            case XFD_CTRLKIND_GRIDCOLUMNDATA: // 그리드 컬럼 데이터인 경우
                break;
            default:
                break;
        }
    }
    else if (object_kind == XFD_OBJKIND_SHAPE) {
        switch (component_kind) {
            case XFD_SHAPEKIND_CAPTION:
                prop_info['text'] = meta_item_arr[0];
                return 1;
            default:
                break;
        }
    }

    return 0; // 엔진 기본 동작 수행
}
```

## 다국어 관련 부가 파라미터 및 API

### 파라미터

실행 파라미터에 아래의 예시와 같은 다양한 정보를 설정할 수 있다.

```

VALUE_THOUSANDS_SEPARATOR: ',', // 값 저장 기준 천단위 구분자
VALUE_DECIMAL_SEPARATOR: '.', // 값 저장 기준 소수점 구분자

SCREEN_LANGUAGE: 'ko', // 화면 개발 기준 언어
SCREEN_COUNTRY: 'kr', // 화면 개발 기준 국가
SCREEN_THOUSANDS_SEPARATOR: ',', // 화면 개발 기준 천단위 구분자
SCREEN_DECIMAL_SEPARATOR: '.', // 화면 개발 기준 소수점 구분자
SCREEN_FONT: ['맑은 고딕',9,0,0,0,0], // 화면 개발 기준 폰트 정보 (xframe5.css 설정 필요)

USER_LANGUAGE: 'ko', // 사용자 언어('ko'/'en'), 파라미터에 의해서 동적 변경
USER_COUNTRY: 'kr', // 사용자 국가('kr'/'us'), 파라미터에 의해서 동적 변경
USER_THOUSANDS_SEPARATOR: ',', // 사용자 언어 기준 천단위 구분자
USER_DECIMAL_SEPARATOR: '.', // 사용자 언어 기준 소수점 구분자

IS_INPUT_RTL: false, // 데이터 입력 방향이 우측에서 좌측인지 여부(기본값: false)
HOLIDAY_COLOR: [236,109,97], // 달력 휴일 색상
SUNDAY_COLOR: [236,109,97], // 달력 일요일 색상
SATURDAY_COLOR: [104,169,219], // 달력 토요일 색상
FRIDAY_COLOR: [102,102,102], // 달력 금요일 색상
WEEKDAY_COLOR: [102,102,102], // 달력 평일 색상

// 휴일/기념일 정보 오브젝트(휴일/기념일 일자, 휴일/기념일 이름, 공휴일 여부)배열 설정
HOLIDAY_LIST: [
    { date: '****0101', title: '양력설', is_holiday: true },
    { date: '****0301', title: '3.1절', is_holiday: true },
    { date: '****0505', title: '어린이날', is_holiday: true },
    { date: '****0606', title: '현충일', is_holiday: true },
    { date: '****0815', title: '광복절', is_holiday: true },
    { date: '****1003', title: '개천절', is_holiday: true },
    { date: '****1009', title: '한글날', is_holiday: true },
    { date: '****1225', title: '기독탄신일', is_holiday: true }
],

// 요일, 월 이름 정보 배열 설정
FIRSTDAY_OF_WEEK: 0, // 0: sunday, 1:monday
DAYNAME_OF_WEEK: ['일','월','화','수','목','금','토'],
FULL_DAYNAME_OF_WEEK: ['일요일','월요일','화요일','수요일','목요일','금요일','토요일'],
MONTHNAME_OF_YEAR: ['1월','2월','3월','4월','5월','6월','7월','8월','9월','10월','11월','12월'],
FULL_MONTHNAME_OF_YEAR: ['1월','2월','3월','4월','5월','6월','7월','8월','9월','10월','11월','12월'],
SCHEDULE_DAYNAME_OF_WEEK: ['일[Sun]','월[Mon]','화[Tue]','수[Wed]','목[Thu]','금[Fri]','토[Sat]']

```

## API

메타 데이터 이외의 추가적인 정보를 설정하는 API 를 제공한다. 아래의 API 는 factory 오브젝트를 통해 제공된다.

구분	설명	비고
setlocaleseparator	천단위 및 소수점 구분자를 지정한다.	
setfirstdayofweek	달력의 첫번째 요일을 지정한다.	
setlocalecalendar	달력 관련 문자열을 지정한다.	
setpicklistbaseurl	코드 파일 기본 URL 을 지정한다.	
setholiday	휴일 정보를 설정한다.	
setinputrtl	문자열에 대한 입력 방향을 지정한다.	