



IB Sheet 7

개발자가이드



IBSheet 개발자 가이드

초판발행/ 2011년 7월 20일

지은이/ 현재준

펴낸이/ 정광천

펴낸곳/  (주) 아이비 리더스
Inno Biz Leaders Co., Ltd.

137-878 서울시 구로구 구로3동 한화비즈메트로 1차 1501호

전화/ (02)2621-2080

팩스/ (02)2621-2088

편집/ 현재준 인쇄/ C&G

※ 기술상담 및 내용문의는 (주)아이비리더스로 해주십시오.

<http://www.ibleaders.co.kr>



목 차

Chapter 1. IBSheet 개요	20
1. IBSheet 개요	22
1.1 일반적인 Sheet 란?	22
1.2 IBSheet 의 정의	22
1.3 IBSheet 의 적용 대상 업무	23
1.4 IBSheet 의 주요 기능	23
Chapter 2. IBSheet 의 구조	29
2. IBSheet 의 구조	31
2.1 Client Running 구조	31
2.2 데이터 조회/저장 과정	31
Chapter 3. IBSheet 의 개발환경 설정	33
3. IBSheet 의 개발 환경 설정	35
3.1 IBSheet 의 Server 설치	35
3.1.1 배포파일 및 내용	35
Chapter 4. 개발 절차	40
4. 개발 절차	42
4.1 웹 페이지에 JS 파일 링크 걸기	42
4.2 IBSheet 객체를 화면에 추가	42
4.3 초기항목설정	44
4.3.1 시트 기본 설정	45
4.3.2 헤더 정보 설정	46
4.3.3 컬럼 정보 설정	47
4.3.4 시트초기화 단순화	55
4.4 기능 구현하기	55



4.4.1 조회	55
4.4.2 입력	56
4.4.3 수정	57
4.4.4 삭제	57
4.4.5 저장	57
4.4.6 소계	57
4.4.7 엑셀다운	58
4.4.8 이벤트 사용	59
4.5 조회 XML 구현하기	59
4.6 저장 XML 구현하기	61
4.7 조회 JSON 구현하기	61
4.8 저장 JSON 구현하기	62
Chapter 5. 개발 실습	66
5. 개발 실습	68
5.1 IBSheet 객체 생성 및 기본 속성 설정하기	68
5.2 각종 기능 구현 하기	69
5.3 조회하기	70
5.4 저장하기	75
Chapter 6. 데이터 구조	82
6. 데이터 구조	84
6.1 XML	85
6.1.1 조회 XML	85
6.1.1.1 SHEET 요소	87
6.1.1.2 DATA 요소	89
6.1.1.3 TR 요소	91
6.1.1.4 TD 요소	93
6.1.1.5 ETC-DATA 요소	95
6.1.1.6 MESSAGE 요소	97
6.1.1.7 RESULT 요소	98



6.1.2 저장 XML	99
6.1.2.1 SHEET 요소	101
6.1.2.2 RESULT 요소	102
6.1.2.3 TR 요소	103
6.1.2.4 TD 요소	104
6.1.2.5 ETC-DATA 요소	106
6.2 JSON	107
6.2.1 조회 JSON	107
6.2.1.1 DATA 요소	108
6.2.1.2 TOTAL 요소	111
6.2.1.3 ETC 요소	112
6.2.1.4 MESSAGE 요소	113
6.2.1.5 RESULT 요소	114
6.2.2 저장 JSON	115
6.2.2.1 RESULT 요소	116
6.2.2.2 ETC 요소	118
6.2.1.4 MESSAGE 요소	119
6.3 Tree 기능 사용시 JSON	120
6.4 엑셀리포트 XML	122
6.4.1 IBSheetSet 요소	125
6.4.2 PageSet 요소	126
6.4.3 ColumnWidth 요소	129
6.4.4 RowHeight 요소	130
6.4.5 CustomHeader 요소	131
6.4.5.1 LeftSection, CenterSection, RightSection 요소	132
6.4.6 CustomFooter 요소	134
6.4.7 Label 요소	135
6.4.7.1 Range 요소(Label 요소의 하위)	136
6.4.7.1.1 CellFormat 요소	138
6.4.7.1.2 InputText 요소	140
6.4.7.1.3 BorderStyle 요소	141
6.4.8 DataPattern 요소	143



6.4.8.1 Standard 요소	144
6.4.7.1 Range 요소(DataPattern 요소의 하위)	146
6.4.9 BorderSet 요소	148
6.4.9.1 Range 요소(BorderSet 요소의 하위)	150
Chapter 7. IBSheet Config	153
7. IBSheet Config	155
7.1 Config 설정	155
7.2 Property List (빈번히 사용되는 속성)	156
Chapter 8. IBSheet Events	179
8. IBSheet Events	181
8.1 Event 사용	181
8.2 Event List	182
OnAfterColumnMove Event	182
OnAfterEdit Event	183
OnAfterPaste Event	184
OnAfterExpand Event	185
OnBeforeCheck Event	186
OnBeforeCheckAll Event	188
OnBeforeColumnMove Event	189
OnBeforeDownload Event	190
OnBeforeEdit Event	191
OnBeforeExpand Event	192
OnBeforeMovePage Event	193
OnBeforePaste Event	194
OnBeforeSave Event	196
OnBeforeSearch Event	197
OnButtonClick Event	198
OnCellDropEnd Event	199
OnChange Event	201



OnChangeFilter Event	203
OnChangeSum Event	204
OnCheckAllEnd Event	205
OnClick Event	206
OnDbIcClick Event	207
OnDebugMsg Event	208
OnDecryption Event	209
OnDownFinish Event	210
OnDragStart Event	211
OnDropEnd Event	212
OnEditValidation Event	214
OnEncryption Event	215
OnExportEncryption Event	216
OnFilterEnd Event	218
OnGroupFinish Event	219
OnHScroll Event	220
OnKeyDown Event	221
OnKeyUp Event	222
OnLoad Event	223
OnLoadData Event	224
OnLoadExcel Event	225
OnLoadText Event	226
OnMessage Event	227
OnMouseDown Event	229
OnMouseMove Event	230
OnMouseUp Event	232
OnPageRequest Event	233
OnPopupClick Event	234
OnResize Event	235
OnRowSearchEnd Event	236
OnSaveEnd Event	237
OnSearchEnd Event	239



OnSelectMenu Event	241
OnSelectCell Event	243
OnSelectEnd Event	244
OnSmartResize Event	245
OnSort Event	246
OnTab Event	248
OnTreeCheckChange Event	250
OnTreeChild Event	252
OnUserResize Event	254
OnValidation Event	255
OnVScroll Event	257
OnWaitTimeOut Event	258
 Chapter 9. IBSheet Methods	 260
 9. IBSheet Methods	 262
9.1 Method 사용	262
9.2 Method List	263
ActionMenu Method	263
AllowCheck Method	266
AllowEvent4CheckAll Method	268
AllowExpand Method	269
AutoRowHeight Method	271
AutoSumPosition Method	273
BasicImeMode Method	275
CellAlign Method	277
CellBackColor Method	279
CellComboltem Method	281
CellCursor Method	283
CellEditable Method	285
CellFont Method	287
CellFontBold Method	290
CellFontColor Method	292



CellFontItalic Method	294
CellFontName Method	296
CellFontSize Method	298
CellFontStrike Method	300
CellFontUnderline Method	302
CellImage Method	304
CellSaveName Method	306
CellSearchValue Method	307
CellVAlign Method	309
CellText Method	311
CellValue Method	313
CheckAll Method	317
CheckedRows Method	319
CheckReverse Method	321
ClearHeaderCheck Method	323
ClearSelection Method	324
ClipCopyMode Method	325
ClipPasteMode Method	327
ColBackColor Method	329
ColCondProperty Method	331
ColEditable Method	333
ColDelete Method	335
ColFontBold Method	337
ColFontColor Method	339
ColFontUnderline Method	341
ColHidden Method	343
ColInsert Method	345
ColLeft Method	347
ColSaveName Method	348
ColumnSort Method	349
ColValueDup Method	351
ColValueDupRows Method	353



ColWidth Method	356
ComboOpenMode Method	358
ComputeSum Method	360
ConfirmOK Method	362
CountFormat Method	364
CountInfoElement Method	366
CountPosition Method	368
CreatePivotTable Method	371
CumulateBackColor Method	373
CurrentColInfo Method	375
Data2Clipboard Method	377
DataAlternateBackColor Method	378
DataAutoTrim Method	380
DataBackColor Method	382
DataCopy Method	384
DataFontColor Method	385
DataInsert Method	387
DataLinkMouse Method	389
DataMove Method	391
DataRowHeight Method	393
DataRowMerge Method	395
DirectDown2Excel Method	397
DirectLoadExcel Method	405
DoAllSave Method	410
DoPrint Method	413
DoRowSearch Method	414
DoSave Method	416
DoSearch Method	420
DoSearchChild Method	422
DoSearchPaging Method	425
Down2Excel Method	427
Down2ExcelUrl Method	437



Down2ExcelBuffer Method	439
Down2Pdf Method	441
Down2PdfUrl Method	446
Down2Text Method	447
Down2TextUrl Method	450
DownloadingImage Method	451
DragMode Method	453
Editable Method	455
EditEnterBehavior Method	457
EditableColorDiff Method	459
EditArrowBehavior Method	461
EditTabBehavior Method	463
Ellipsis Method	465
Enable Method	467
EnterBehavior Method	469
EtcData Method	471
EtcDataString Method	475
ExportData Method	476
ExtendLastCol Method	478
FilteredRowCount Method	480
FindCheckedRow Method	481
FindFilterRow Method	483
FindStatusRow Method	484
FindSubSumRow Method	485
FindSumRow Method	486
FindText Method	487
FitColWidth Method	489
FitSize Method	491
FitSizeCol Method	492
FocusAfterProcess Method	493
FocusAfterRowTransaction Method	495
FocusEditMode Method	497



FrozenCol Method	499
FrozenRows Method	501
GetCellProperty Method	503
GetChildNodeCount Method	504
GetChildRows Method	505
GetColSortInfo Method	506
GetComboInfo Method	507
GetCurrentPage Method	509
GetDataFirstRow Method	510
GetDataRows Method	511
GetDataLastRow Method	512
GetEditText Method	513
GetFilterParam Method	514
GetFirstChildRow Method	516
GetGroupCol Method	517
GetGroupRow Method	518
GetLastChildRow Method	519
GetMergedEndCell Method	520
GetMergedStartCell Method	521
GetNextSiblingRow Method	522
GetParentRow Method	523
GetPrevSiblingRow Method	524
GetSaveData Method	525
GetSaveJson Method	527
GetSaveString Method	529
GetSearchData Method	532
GetSelectionCols Method	533
GetSelectionRows Method	534
GetSheetHtml Method	535
GetSubSumInfo Method	536
GoToBaseSheet Method	537
GoToFirstPage Method	538



GoToLastPage Method	539
GoToNextPage Method	540
GoToPageNum Method	541
GoToPrevPage Method	542
GroupActionMenu Method	543
HeaderActionMenu Method	545
HeaderBackColor Method	547
HeaderCheck Method	549
HeaderCursor Method	551
HeaderFontBold Method	553
HeaderFontColor Method	555
HeaderRows Method	556
HeaderRowHeight Method	557
HideFilterRow Method	558
HideGroupRow Method	559
HideProcessDlg Method	560
HideSubSum Method	561
HighlightAfterSort Method	562
IBCloseCalendar Method	564
IBShowCalendar Method	565
ImageList Method	568
InitCellProperty Method	570
InitColumns Method	575
InitComboNoMatchText Method	600
InitHeaders Method	601
IsDataModified Method	603
IsFocused Method	604
IsHaveChild Method	605
KeyFieldImage Method	607
LastCol Method	609
LastRow Method	610
LeftCol Method	611



LoadSaveData Method	613
LoadSearchChildData Method	614
LoadSearchData Method	616
LoadExcel Method	619
LoadExcelBuffer Method	623
LoadExcelUrl Method	625
LoadText Method	626
LoadTextUrl Method	629
MergeSheet Method	630
MaximumValue Method	632
MinimumValue Method	634
MouseCol Method	636
MouseHoverMode Method	637
MousePointer Method	639
MouseRow Method	641
MouseToolTipText Method	642
MoveColumnFail Method	644
MoveColumnPos Method	645
PagingPosition Method	646
PageCount Method	648
PopupButtonImage Method	650
RangeBackColor Method	652
RangeFontBold Method	653
RangeFontColor Method	654
RangeText Method	655
RangeValue Method	657
RedrawSum Method	659
RemoveAll Method	661
RemoveFooterRow Method	662
RenderSheet Method	663
RemoveEtcData Method	665
ReNumberSeq Method	667



Reset Method	668
ReturnCellData Method	669
ReturnColumnPos Method	670
ReturnData Method	671
RowBackColor Method	672
RowBackColorD Method	674
RowBackColorI Method	676
RowBackColorU Method	678
RowCount Method	680
RowData Method	681
RowDelete Method	684
RowDraggable Method	685
RowEditable Method	686
RowExpanded Method	688
RowFontColor Method	690
RowHeight Method	692
RowHeightMax Method	694
RowHeightMin Method	696
RowHidden Method	698
RowLevel Method	700
RowMerge Method	702
RowSaveStr Method	704
RowSumable Method	705
RowTop Method	707
SaveNameCol Method	708
SavingImage Method	709
ScrollInfoFormat Method	711
ScrollInfoPosition Method	713
SearchingImage Method	715
SearchRows Method	717
SelectCell Method	718
SelectCol Method	720



SelectionMode Method	722
SelectionSummaryInfoElement Method	724
SelectRow Method	726
SendComboData Method	728
SetBlur Method	730
SetCellImageStyle Method	731
SetColProperty Method	733
SetConfig Method	739
SetDown2ExcelConfig Method	754
SetDown2TextConfig Method	755
SetEndEdit Method	756
SetFilterOption Method	757
SetFilterValue Method	759
SetFindDialog Method	761
SetFocus Method	763
SetHeaderMode Method	764
SetLoadExcelConfig Method	766
SetLoadTextConfig Method	767
SetMergeCell Method	768
SetRowHaveChildValue Method	770
SetSelectRange Method	771
SetSplitMergeCell Method	773
SheetWidth Method	774
SheetHeight Method	775
ShowButtonImage Method	777
ShowCalendar Method	780
ShowDebugMsg Method	782
ShowColumnPopup Method	783
ShowFilterRow Method	785
ShowFindDialog Method	786
ShowFooterRow Method	787
ShowGroupRow Method	789



ShowMsgMode Method	791
ShowPivotDialog Method	793
ShowPivotSumRatio Method	794
ShowPivotTable Method	795
ShowProcessDlg Method	797
ShowSortArrow Method	798
ShowSubSum Method	800
ShowToolTip Method	804
ShowTreeSubSum Method	805
SubSumBackColor Method	807
ShowTreeLevel Method	808
SumBackColor Method	810
SumFontBold Method	812
SumFontColor Method	814
SumRowHidden Method	816
SumValue Method	818
TabBehavior Method	820
Theme Method	822
ToolTipText Method	824
TopRow Method	826
TotalRows Method	828
TreeActionMode Method	830
TreeCheckActionMode Method	832
TreeCheckEditable Method	834
TreeCheckRowEditable Method	836
TreeCheckValue Method	838
TreeChildSort Method	840
UnicodeByte Method	841
UseDefaultTime Method	843
UserAgent Method	845
UploadingImage Method	847
ValidateFail Method	849



Visible Method	851
WaitImage Method	853
WaitImageVisible Method	855
WaitTimeOut Method	857
Chapter 10. IBSheet Global Module	859
10. IBSheet Global Module	861
10.1 Global Module 사용	861
10.2 Method List	863
IBCalendar.SetTheme Method	863
IBCalendar.Show Method	864
IBCalendar.Close Method	868
Appendix	870
1. 컬럼 타입별 정의 가능한 속성	872
1.1. 텍스트 형식 타입	872
1.2. 날짜 형식 타입	874
1.3. 숫자 형식 타입	876
1.4. 체크박스 형식 타입	879
1.5. 콤보 형식 타입	881
1.6. Html 형식 타입	883
1.7 기타 형식 타입	886
2. Fx(Formatted) 조회 모드의 허용 데이터 포맷	889
3. Theme 만들기	892
4. 엑셀 다운시 머지 적용 처리	893
5. 다중 검색어 및 범위 필터 기능 설정 및 사용법	894
6. 아이비시트에서 자주 사용되는 메서드 정리	896



(주) 아이비 리더스

Inno Biz Leaders Co., Ltd.



Chapter 1. IBSheet 개요



(주) 아이비 리더스

Inno Biz Leaders Co., Ltd.

1. IBSheet 개요

1.1 일반적인 Sheet 란?

- Sheet 혹은 Grid라는 용어는 Web 이전의 C/S 환경의 개발이 대세를 이루던 당시부터 사용되던 용어로 DBMS로부터 다량의 레코드 데이터를 조회하여 사용자가 조작하는 화면에 표시하고, 사용자가 변경한 데이터를 트랜잭션 관리를 통해 DBMS에 저장하는 역할을 주로 하였다.
- 이러한 Sheet의 역할은 대부분의 개발환경에서 반드시 필요로하여 사용되었고 현재까지도 기존보다 많은 기능이 추가되어 사용되고 있다.



1.2 IBSheet의 정의

- 크로스 브라우징, 크로스 플랫폼을 지원하는 Sheet Component
- 별도의 개발툴이 필요 없고 일반적인 Javascript를 통하여 각 기능을 설정함으로써 웹 개발자(ASP.net, JSP, PHP 등)가 쉽게 이해하고 응용하여 기능 구현이 가능.
- Ajax 통신을 통해 데이터를 XML 또는 JSON형태로 받아 화면에 로딩
- 사용자가 입력/수정하는 데이터를 관리하여 최종적으로 변경된 데이터만 레코드 단위로 모아 서버로 전달하여 저장작업을 처리
- 각 컬럼별로 다양한 데이터 타입을 제공함(체크박스, 콤보, 팝업, 이미지 등)
- 각 컬럼별로 다양한 데이터 포맷을 제공함(날짜, 정수, 실수, 주민등록번호 등)
- 합계/소계/누계 등 숫자연산 기능 제공 및 연산 관련 함수 제공
- 컬럼별 소팅 및 필터기능 제공

- 다양한 사용자 편의 기능 제공(context menu 기능, Max, Min 값 입력 제한 기능 등)
- 특정 컬럼을 트리로 지정하여 트리의 접힘/펼침에 따라 데이터를 보여주거나 감추는 트리기능 제공
- CSS를 이용한 테마 적용 기능 제공
- 메시지 파일 형식을 통한 다국어 지원 및 적용 용이
- 이전 IBSheet(ActiveX)와 하위 호환성을 갖춰, 기존 IBSheet 개발자들이 쉽게 적응 가능
- 개발자에게는 쉬운 개발을 운용자에게는 편리한 운용 방안을 제공

1.3 IBSheet의 적용 대상 업무

- 리스트를 조회하고, 정렬순서를 변경하여 다시 조회하고자 하는 경우가 많은 업무
- 리스트 조회 시 조회조건에 따라 화면 Reload가 많고 데이터 양이 많은 업무
- 업무적으로 한번에 반드시 여러 건을 입력/수정/삭제해야 하는 업무
- 컬럼 간 계산 로직이 많은 업무
- 기존 C/S환경과 비슷한 구조로 웹 환경으로 변경하고자 하는 업무

1.4 IBSheet의 주요 기능

➤ Data 조회

Ajax 통신을 통하여 서버로부터 데이터를 조회 한다. 이때 IBSheet위에 표현되어야 할 데이터만을 XML 또는 JSON 형태로 가져와 로딩함으로 기존에 페이지를 리로딩하는 것보다 효율적으로 데이터를 가져올 수 있다.

➤ Data Column Sorting

IBSheet위에 조회되거나 작성된 데이터는 헤더를 클릭하는 동작만으로 서버를 거치지 않고 오름차순/내림차순으로 정렬할 수 있다.

또한 제공되는 소팅 메서드(ColumnSort())를 통하여 하나의 컬럼이 아닌 다수의 컬럼을 DB Query의 order by문장과 같이 동시에 소팅하는 작업도 가능하다.



➤ Multi-Record Transaction

IBSheet안에 변경된 데이터는 시트 내부적으로 관리된다.

따라서 IBSheet는 최초 조회시 데이터와 변경된 데이터를 동시에 갖고 있다가 저장메서드가 호출되었을 때, 전체 데이터 중에 변경된 데이터를 포함하고 있는 행(Row)를 묶어 서버로 전달하게 된다. 또한 사용자가 수정한 데이터가 올바르게 바르지 않은 경우 제공되는 메서드를 통하여 수정하기 이전의 데이터로 되돌아 가는 기능도 제공하고 있다.

➤ Tree Data

IBSheet안에 특정 컬럼을 트리로 지정하고, 조회되는 데이터도 행(Row)단위로 Level(depth)을 지정해 주면 트리기능을 사용할 수 있다.

트리기능을 사용하는 시트의 경우 행단위로 Level값을 갖게되고 이를 통하여 특정 Level까지 자동으로 접거나 펼치는 기능들을 사용할 수 있다.

단, 트리가 사용될 경우 소트 기능은 지원하지 않는다.

➤ Data Column 자동 연산 처리

IBSheet에서는 Column 간의 연산식을 등록하게 되면 Data가 조회됨과 동시에 등록된 연산식을 통하여 자동으로 연산 된 값을 출력하게 된다. 또한 사용자가 데이터를 변경시에도 연산식 적용이 가능하다.

➤ Column Freezing & Sizing

컬럼의 너비와 개수가 많아서 수평 스크롤바로 이동하며 데이터를 조회하는 경우 틀고정 컬럼의 위치를 사용자가 마우스로 이동하며 데이터를 표현할 수 있다. (Column Freezing)

또한 컬럼의 너비가 작아서 데이터가 잘려서 표시되는 경우 해당 헤더와 좌측 헤더 사이의 선을 마우스를 이용하여 너비를 사용자가 조정할수 있다. (Column Sizing)

그리고, 컬럼의 위치를 마우스를 이용하여 움직여서 사용자가 우선으로 생각하는 데이터 컬럼을 앞으로 옮겨와서 보다 편한 UI를 EndUser가 생성할 수 있다.

➤ Type

타입이란 각 컬럼의 기능이라 할 수 있다. 여기서 기능은 일반적인 문자열 입력박스 외에 체크박스, 콤보, 팝업, 이미지 등이 제공되며 각각 컬럼에 적용하는 타입에 따라 사용할 수 있는 기능메서드가 다르게 제공된다.



➤ **Data Merge**

IBSheet의 병합기능은 근접한 셀간의 Data 값이 같을 경우에는 MergeSheet 프로퍼티에 설정에 따라 자동으로 병합된다.

➤ **Data Coloring**

html에서의 Cell, Data에 대한 Back Ground Color, Font Color등을 설정할 수 있으며, 화면의 재 작성 없이 Coloring의 변화를 줄 수 있습니다

➤ **사용자 맞춤 Design**

CSS를 이용한 테마기능 또는 제공되는 함수 프로퍼티에 따라서 헤더,데이터 영역의 배경색이나 글자색,Bold와 같은 설정을 하나의 IBSheet에 단독 적용하거나 혹은 모든 화면의 IBSheet에 동일하게 적용할 수 있다.

➤ **Data Edit 기능**

IBSheet는 시트 전체 및 각 행(Row)단위, 각 컬럼(Column), 각 셀단위로 수정 가능/불가능 여부를 동적으로 변경할 수 있다.

➤ **각종 Event 기능**

IBSheet 상에서 변경되거나 동작하는 여러가지 행위는 다양한 이벤트를 통하여 캐치되어 활용될 수 있다.

예를 들어 시트에서 데이터가 변경되는 경우 발생하는 OnChange 이벤트나 특정 셀을 클릭했을 때 발생하는 OnClick 과 같은 이벤트 외에도 포커스가 이동 시 발생하는 OnSelectCell 이나 마우스가 움직일 때 마다 발생하는 OnMouseMove등 다양한 이벤트를 제공하고 있다.

➤ **편리한 User Interface**

- 숫자 입력 시 자동 콤마 표현
- 숫자 컬럼 문자 입력 제한
- 날짜, 주민등록번호, 사업자번호, 카드번호, 우편번호 등 입력 시 자동 Masking 기능



- 자동 입력 길이 제한
- 한글 키보드, 영문 키보드 자동 변환 기능
- 필수입력 컬럼 표시
- 기존 행 복사 기능
- 툴팁 기능
- WordWrap 기능





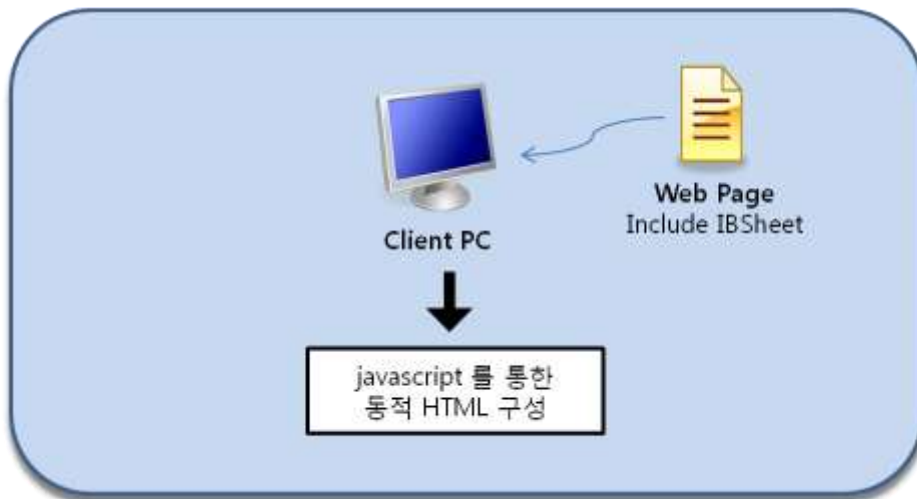
Chapter 2. IBSheet의 구조



2. IBSheet의 구조

2.1 Client Running 구조

➤ Load 과정



IBSheet가 포함된 페이지를 PC에서 열 경우 별도의 프로그램 또는 컨테이너 없이 Sheet 가 동적 HTML로 구성되어 Load 된다.

2.2 데이터 조회/저장 과정

- ① Data Request : IBSheet의 조회/저장 함수를 이용하여 데이터를 요청한다.
- ② DB Data Request : 요청을 받은 서버는 FrameWork를 통해 SQL QueryString을 구성하여 DBMS에 데이터를 요청한다.
- ③ DBMS는 QueryString을 해석하여 DataSet객체를 FrameWork에 반환한다.
- ④ FrameWork은 받은 DataSet을 정적인 Collection객체(List,Map등)로 변환하여 XML 또는 JSON 형태를 생성해주는 Page(혹은 servlet이나 dll)로 전달한다.
- ⑤ XML 또는 JSON 형태로 데이터 생성이 완료되면 IBSheet로 반환한다.
- ⑥ IBSheet는 받은 XML 또는 JSON 데이터를 파싱하여 화면에 표시한다.





Chapter 3. IBSheet의 개발환경 설정





3. IBSheet의 개발 환경 설정

3.1 IBSheet의 Server 설치

3.1.1 배포파일 및 내용

IBSheet를 이용한 개발 환경을 구축하기 위해서 아래와 같이 제공되는 파일들을 웹사이트의 특정 폴더에 복사하여 설치한다.

위치는 다음을 기준으로 한다.

₩Sheet₩
₩Sheet₩ibleaders.js
₩Sheet₩ibsheet.js
₩Sheet₩ibsheetinfo.js
₩Sheet₩ibmsg
₩Sheet₩ibsheet.cfg
₩Sheet₩Main(폴더)

jsp 파일들 (ibsheet.cfg에 위치를 설정)

₩Sheet₩Down2Excel.jsp
₩Sheet₩LoadExcel.jsp
₩Sheet₩DirectLoadExcel.jsp
₩Sheet₩DirectDown2Excel.jsp
₩Sheet₩Down2Text.jsp
₩Sheet₩LoadText.jsp

.Net

₩Sheet₩Down2Excel.aspx
₩Sheet₩LoadExcel.aspx
₩Bin₩IBSheet7.dll



₩Bin₩Syncfusion.Compression.Base.dll

₩Bin₩Syncfusion.Core.dll

₩Bin₩Syncfusion.XlsIO.Base.dll

각각 파일의 내용을 설명하자면 아래와 같다.



파일명	내용/역할
ibleaders.js	IB Sheet 7 제품 군 공통 js 파일
ibsheet.js	IBSheet 엔진 js 파일
ibsheetinfo.js	시트 초기화시 필요한 함수 및 개발시 유용하게 사용할 수 있는 js함수를 담아둔 파일 (지속적 업데이트 예정)
ibmsg	IBSheet에서 사용하는 시스템 메시지 파일 다국어 사용시 해당 파일을 복사하여 ibmsg.[lang] 로 복사한 후 메시지 내용을 번역하여 사용한다. (예) ibmsg.en
ibsheet.cfg	모든 시트에 전역으로 설정할 값을 정의하기 위한 파일. Down2Excel.jsp와 LoadExcel.jsp 파일에 대한 경로가 정의되어 있다.
Main	기본 테마 정보를 담고 있는 폴더
Down2Excel.jsp	엑셀다운로드 전용인 Down2Excel 메소드를 적용하기 위한 서버쪽 응용 프로그램 파일 ※ 한글이 깨질경우 jsp 내에서 UTF8 설정을 변경해야 한다. ※ 엑셀 다운로드 문서내에 이미지가 포함되는데 이 경로가 가상폴더가 포함된다면, WebRoot 변수에 서버의 물리적 디렉토리 경로를 설정해야 한다.
LoadExcel.jsp	엑셀로딩 전용인 LoadExcel 메소드를 적용하기 위한 서버쪽 응용 프로그램 파일 ※ 한글이 깨질경우 jsp 내에서 UTF8 설정을 변경해야 한다. ※ 엑셀로딩에서 임시사용될 서버쪽 폴더경로를 설정해야 한다.
DirectLoadExcel.jsp	엑셀로딩 전용인 LoadExcel 메소드와 달리 DirectLoadExcel 메소드와 연계되는 서버쪽 파일. LoadExcel 처럼 엑셀 파일을 서버로 올리는 것은 동일하지만, 엑셀 파일의 분석 결과를 그리드로 보내지 않고, DB 에 바로 적용할 수 있다. 엑셀파일의 내용은 forward 페이지로 List(Map)형태로 전달되고 forward 페이지에서 비즈니스 로직을 수행하면 된다.
DirectDown2Excel.jsp	대용량의 데이터를 엑셀파일형태로 내리기 위해서, 시트에서 데이터를 서버로 전달하지 않고, DB에서 조회한 데이터를 바로 엑셀파일로 내리게 하기 위한 jsp 파일



Down2Text.jsp	시트의 파일을 txt 형태로 내리게 하기 위한 jsp 파일
LoadText.jsp	txt 파일의 내용을 시트위에 올리기 위한 jsp 파일

jsp 파일들은 사용하는 서버의 encoding 설정이나 서버 경로에 대한 설정을 해줘야 정상적으로 다운로드 된다. 파일을 열면 소스 상단에 수정이 가능한 전역 변수가 몇개 있는데 이 부분을 수정해 주면 된다.

엑셀 다운로드 및 업로드의 기능을 원활히 지원하기 위해서는 WAS 단에 poi 라이브러리가 설치되어 있어야 한다. 아래와 같이 lib 폴더 안에 파일을 설치한 경우 WAS 서버를 재부팅해야 적용된다.

(예)

/Tomcat60/lib/poi-3.9-20121203.jar// 다운로드 전용

/Tomcat60/lib/ poi-ooxml-3.9-20121203.jar // 업로드 전용

/Tomcat60/lib/ poi-ooxml-schemas-3.9-20121203.jar // 업로드 전용

/Tomcat60/lib/ dom4j-1.6.1.jar // 업로드 전용

/Tomcat60/lib/ xmlbeans-2.3.0.jar // 업로드 전용

.Net

Down2Excel.aspx 파일을 열면 소스에 사용자가 설정이 가능한 함수 몇개 있는데 이 부분을 수정해 주면 된다.



(주) 아이비 리더스

Inno Biz Leaders Co., Ltd.



Chapter 4. 개발 절차



(주) 아이비 리더스

Inno Biz Leaders Co., Ltd.



4. 개발 절차

4.1 웹 페이지에 JS 파일 링크 걸기

IBSheet 객체를 생성하고, 생성된 객체에 여러가지 기능 동작 수행을 위한 두가지 js 파일을 링크 건다.(**ibsheet.css** 파일은 **ibsheet.js** 파일에서 알아서 참조함으로 별도로 화면상에 링크를 넣을 필요는 없다)

```
<script language="javascript" src="/Sheet/ibleaders.js"></script>
<script language="javascript" src="/Sheet/ibsheetinfo.js"></script>
<script language="javascript" src="/Sheet/ibsheet.js"></script>
```

위 두개 파일 중 ibsheet.js 파일은 IBSheet 엔진 js 파일이다. 그리고 ibsheetinfo.js 파일은 시트 초기화시 필요한 상수 및 개발시 유용하게 사용할 수 있는 js함수를 담아둔 파일이다.

4.2 IBSheet 객체를 화면에 추가

body태그 안에서 시트객체 생성

js파일을 추가한 뒤에 IBSheet 객체를 화면에 추가한다.

추가하는 방법은 **ibsheetinfo.js** 에 있는 **createIBSheet(sheetid,width,height,[lang])** 함수를 이용하여 추가하게 된다.

```
<body>
  <table class='sheet_table'>
    <tr>
      <td>
        <script language="javascript">
          //IBSheet 객체 생성
          createIBSheet("mySheet", "750px", "400px");
        </script>
```



```
</td>
</tr>
</table>
</body>
```

위와 같이 공통 스크립트를 사용하면 모든 화면에 공통적으로 동일한 IBSheet 객체를 추가할 수 있다.

다국어 사용하는 경우 ibmsg 파일을 언어별로 생성한 후 createIBSheet 함수를 이용하여 설정이 가능 하다.

```
// IBSheet 객체 생성
createIBSheet("mySheet", "750px", "400px", "en");
```

위와 같이 "en"을 인자로 넣는 경우 IBSheet에서 사용하는 메시지 및 포맷형태를 ibmsg.en 파일의 내용으로 사용하게 된다.(인자가 없는 경우는 ibmsg 파일 사용)

※ 인자를 사용하는 경우 인자를 확장자로 하는 ibmsg 파일이 반드시 존재하여야 한다.

참고1. body태그 밖에서 시트객체 생성

```
<script language="javascript">
function LoadPage(){
    //IBSheet 객체 생성
    createIBSheet2( document.getElementById("sheetObj"),
        "mySheet", "750px", "400px");
}
</script>
<body onload="LoadPage()">
    <div id="sheetObj"></div>
</body>
```

참고2. 시트에서 사용하는 파일에 대한 Base 확장자 지정 방법

3장과 이번 장에서 소개한 바와 같이 ibsheet는 기본적인 javascript(js) 파일 외에도 ibsheet.cfg 나 ibmsg.kr, license(**확장자 없음**)과 같이 일반적이지 않은 확장자를 갖은 파일을 사용하는데 이런 파일들은 web서버의 mine-type에 별도로 등록해 주지 않는 경우 문제를 일으키기도 한다.

따라서 이러한 파일에 대해서 기본확장자를 지정하는 기능을 제공하고 있다. 기본확장자를 지정하는 방법은 ibsheetinfo.js 파일에 시트를 생성하는 createlBSheet() 함수에 BaseFileExt라는 속성을 설정하는 것이다.

```
//기본 파일의 확장자를 xml로 설정
function createlBSheet(){
    ...
    Grids.Locale = locale;
    Grids.BaseFileExt = "xml";
    ...
}
```

위와 같이 설정시 ibsheet는 다음과 같이 xml확장자를 갖은 파일을 바라보게 된다.

- ibsheet.cfg.xml

- ibmsg.xml (ibmsg.kr.xml, ibmsg.en.xml, ...)

단 이 속성은 license 파일에는 적용되지 않으며, license 파일에 특정확장자를 적용하고자 하는 경우에는 ibsheet.cfg 파일에 LicenseFileName 속성을 설정하여야 한다.

4.3 초기항목설정

개체가 생성된 후 페이지가 onload 될때 시트의 초기 항목을 설정한다. 아래와 같이 Body 태그의 onload 이벤트에서 IBSheet에 대한 초기 항목을 설정할 함수를 정의 한다.

<Body onload="기본 속성을 설정할 함수명()">

Body 태그의 onload 이벤트에서 정의하는 것은 페이지의 로드가 완료 되는 시점에 IBSheet의 객체도 생성이 완료되므로 그때 IBSheet의 제공 기능들을 사용할 수 있기 때문이다. 만약 객체 생성 전에 IBSheet 기능을 사용하려고 한다면 자바스크립트 오류가 발생 할 수 있다.

초기 항목 설정에 필요한 처리 항목은 아래와 같다.

기본 코드 절차	IBSheet 제공 함수	호출 횟수
1. 시트 정보 설정	SetConfig Method	1
2. 헤더 정보 설정	InitHeaders Method	1
3. 컬럼 정보 설정	InitColumns Method	1

아래의 절차에 따라 페이지 설정해 보도록 한다.

4.3.1. 시트 기본 설정

IBSheet객체.SetConfig(cfg);

전달되는 인자의 형식은 JSON 타입으로 설정 하고자 하는 정보를 JSON 형태로 구성하여 전달한다.

예) var cfg= {SearchMode:1, Page:20};

IBSheet객체.SetConfig(cfg);

이 함수를 통하여 시트의 기본적인 환경을 설정한다.

설정변수	타입	설명
DataRowMerge	Boolean	전체행의 가로머지 허용여부 (Default=0)
FrozenCol	Integer	고정컬럼의 수
MergeSheet	Integer	머지 종류를 설정 ➤ msNone : 머지없음 ➤ msHeaderOnly : 헤더만 머지 ➤ msPrevColumnMerge : 데이터영역 앞컬럼기준 머지 ➤ msAll : 전체 머지 ➤ msHeaderOnly+msPrevColumnMerge : 헤더영역 머지 + 데이터영역 앞컬럼 머지
Page	Integer	SearchMode가 1또는 2인 경우 한번에 표시



		할 행의 개수. 이 개수를 기준으로 머지된 행이 나누어 진다.
RefreshHeaderMode	Integer	헤더 갱신 모드 설정 (Default=1) ➤ 0: 헤더 리프레쉬 안함 ➤ 1: 컬럼이동시 헤더 리프레쉬함
SearchMode	Integer	조회 방식 설정 ➤ smGeneral : 전체보기조회 (권장하지 않음) ➤ smClientPaging : 페이징 조회 ➤ smLazyLoad : 스크롤 조회(권장, 기본값) ➤ smServerPaging : 서버 스크롤 페이징 조회 (5000건 이상의 대용량시 권장)
SizeMode	Integer	사이즈 방식 설정 ➤ 0: 사이즈 고정 ➤ 1: 높이를 스크롤 없이 자동 설정 ➤ 2: 너비를 스크롤 없이 자동 설정 ➤ 3: 높이, 너비를 스크롤 없이 자동 설정 ➤ 4: 설정한 높이 이전까지 스크롤 없이 높이 자동 설정
SumPosition	Boolean	합계행 위치(1: 하단 고정, 0: 상단 고정)
SumZeroValue	String	합계 행의 합계값이 0 인경우 치환값 설정
ToolTip	Boolean	셀의 풍선도움말을 표시한다.

※ 세부 정보는 [SetConfig 메소드](#)의 설명 참조

4.3.2. 헤더 정보 설정

IBSheet객체.InitHeaders(headers, [info]);

전달되는 인자의 형식은 JSON 타입으로 설정 하고자 하는 정보를 JSON 형태로 구성하여 전달한다.

예) var headers = [{Text:"사원정보|사원정보|사원정보", Align:" Center"},
 {Text:"사원명|사원번호|입사일", Align:" Center"}];

```
var info = {Sort:1, ColMove:1, ColResize:0, HeaderCheck:0};
```

이 함수를 통하여 헤더의 텍스트와 헤더 텍스트의 정렬을 설정 한다.

설정변수		타입	설명
headers	Text	String	헤더에 표시할 Text를 " "로 연결한 문자열
	Align	String	헤더 텍스트의 정렬방법
info	Sort	Boolean	헤더 클릭 시 소트 가능 여부 (Default=1)
	ColMove	Boolean	헤더 컬럼 이동 가능 여부 (Default=1)
	ColResize	Boolean	컬럼 너비 ReSize 여부 (Default=1)
	HeaderCheck	Boolean	헤더에 전체 체크 표시 여부 (Default=1)

※ 세부 정보는 [InitHeaders 메소드](#)의 설명 참조

4.3.3. 컬럼 정보 설정

IBSheet객체.InitColumns(cols)

전달되는 인자의 형식은 JSON 타입으로 설정 하고자 하는 정보를 JSON 형태로 구성하여 전달한다.

```
예) var cols = [
    {Type:"Text", Width:100 ,SaveName:"sText", Align:"Left"},
    {Type:"Int", Width:80 ,SaveName:"sNumber", Format:"NullInteger"}
];
```

이 함수를 통하여 컬럼에 대한 기본 속성을 설정한다.

컬럼별로 설정 할 수 있는 속성은 다음과 같다.

속성	타입	설명
Type	String	컬럼의 데이터 타입
Width	Integer	컬럼의 너비
Align	String	데이터의 정렬
SaveName	String	데이터 저장 또는 조회시 사용하는 변수명
Edit	Boolean	편집가능 여부
EditLen	Integer	데이터의 입력가능한 글자수
Format	String	데이터의 Mask 적용 형태
ComboText	String	콤보 리스트의 문자열 집합
ComboCode	String	콤보 리스트의 코드 집합



MultiLineText	Boolean	다중라인 입력여부
Wrap	Boolean	자동 줄바꿈 여부
TreeCol	Boolean	트리 기준컬럼
KeyField	Boolean	필수 입력 항목
CalcLogic	String	컬럼별 계산식
ColMerge	Boolean	컬럼 머지 허용여부
Hidden	Boolean	컬럼 숨김 여부
ImgWidth	Integer	이미지 너비
ImgHeight	Integer	이미지 높이
PopupText	String	팝업메뉴의 문자열 집합
PopupCode	String	팝업메뉴의 코드 집합
PopupCheckEdit	Boolean	팝업메뉴 설정시 편집가능 여부
UpdateEdit	Boolean	트랜잭션이 "조회"인 상태에서 데이터의 Edit 가능 여부
InsertEdit	Boolean	트랜잭션이 "입력"인 상태에서 데이터의 Edit 가능 여부
LevelSaveName	String	트리레벨의 데이터 저장 또는 조회시 사용하는 변수명
ButtonUrl	String	버튼이미지파일명(Popup 타입일경우)
TrueValue	String	1 이외의 CheckBox 형태 컬럼의 True 값 지정. "M" 으로 지정한 경우 1 대신 "M"을 True 값으로 사용 가능.
FalseValue	String	1 이외의 CheckBox 형태 컬럼의 False 값 지정. "F" 으로 지정한 경우 0 대신 "F"를 False 값으로 사용 가능.
MaximumValue	Number	숫자포맷일 때 편집시 입력할수 있는 최대값
MinimumValue	Number	숫자포맷일 때 편집시 입력할수 있는 최소값
PointCount	Integer	컬럼타입이 Float 인 경우 소수점 이하의 자리수
FullInput	Boolean	컬럼타입이 단일행 문자열인 경우 전체길이 (EditLen)만큼의 입력 여부
ToolTipText	String	풍선 도움말에 설정할 문자열

Ellipsis	Boolean	말줄임 여부
BackColor	String	배경 색상
FontColor	String	폰트 색상
ApproximateType	Integer	근사값처리방식 설정 (1:반올림, 2:내림, 3:올림)
ShowCol	Number	멀티콤보 설정 시 컬럼 값으로 출력할 컬럼 설정

Type은 해당 컬럼에 대한 Type 설정으로 다음과 같은 값을 설정 할 수 있다.

Type	설명
Text	기본 문자열 데이터
Status	트랜잭션 상태를 표시하고, 담고 있는 데이터
DelCheck	삭제 처리만 담당하는 CheckBox 형태 데이터
CheckBox	CheckBox 형태 데이터
DummyCheck	CheckBox 형태이나 체크/언체크시 상태를 변화시키지 않음.
Radio	데이터 행 중 하나의 데이터만 체크되는 형태
Combo	Edit 불가능 Combo 데이터
ComboEdit	자동완성형태의 Combo 데이터
AutoSum	자동 합계 계산을 위한 데이터, 기본포맷은 NullFloat
Image	Edit 불가능한 단순 이미지 표현 형태 데이터
Int	정수형 타입
Float	실수형 타입
Date	날짜형 타입
Popup	팝업을 사용한 데이터
Pass	패스워드 형태 데이터
Seq	DB의 시퀀스처럼 무조건 값이 증가하는 값이며, Edit가 불가능한 형태의 데이터

Width는 컬럼에 대한 픽셀 단위 너비 설정으로 설정하지 않을 경우 헤더 Text의 너비에 맞게 자동 설정된다.

(참고. 너비설정과 관련하여 설정한 너비보다 줄어들지 않게하는 **MinWidth**도 있다.)

Align은 데이터에 대한 정렬 설정 Left, Center, Right 값을 설정한다.

DataAlign	설명
Left	좌측 정렬 데이터(Default)
Center	가운데 정렬 데이터
Right	우측 정렬 데이터

SaveName은 데이터를 저장할 때 사용하는 변수 명을 설정한다. 변수명을 설정하지 않을 경우 기본적으로 컬럼 순서대로 C1, C2, ... 로 설정된다.

Edit는 데이터의 편집가능 여부를 설정한다. 기본적으로 1이 설정된다.

EditLen은 데이터의 입력 가능한 글자 수를 설정한다.

Format은 데이터에 대한 Mask 적용 형태를 설정하는 것으로 다음과 같은 값을 설정한다. 포맷 설정은 기본제공 외에 사용자가 직접 정의 해서 설정 할 수 있으며 ((ex) Format="#,###.#0" or "#####-*****" or "MM.dd.yyyy") Type 값에 따라 설정한 Format이 적용 된다.

Format	설명	사용가능타입
Ymd	"년월일" 형태. lmsg의 SYS_d 포맷을 따른다	Date,Text,Popup
Ym	"년월" 형태. lmsg의 SYS_m 포맷을 따른다.	Date,Text,Popup
Md	"월일" 형태. lmsg의 SYS_M 포맷을 따른다.	Date,Text,Popup
Hms	"시분초" 형태. lmsg의 SYS_T 포맷을 따른다.	Date,Text,Popup
Hm	"시분" 형태. lmsg의 SYS_t 포맷을 따른다.	Date,Text,Popup
YmdHms	"년월일시분초" 형태. lmsg의 SYS_G 포맷을 따른다	Date,Text,Popup
YmdHm	"년월일시분" 형태. lmsg의 SYS_g 포맷을 따른다	Date,Text,Popup
Integer	정수 형태, 기본 0	Int,AutoSum,AutoAvg
NullInteger	널 정수 형태, 기본 널	Int,AutoSum,AutoAvg
Float	실수 형태, 기본 0	Float,AutoSum,AutoAvg
NullFloat	널 실수 형태, 기본 널	Float,AutoSum,AutoAvg
IdNo	주민등록번호 형태	Text
SaupNo	사업자번호 형태	Text

PostNo	우편번호 형태	Text
CardNo	카드번호 형태	Text

ComboText는 Type이 "Combo" 또는 "ComboEdit"인 경우 화면에 보여질 문자열 항목을 "|"로 연결하여 구성한다.

콤보리스트를 다중 컬럼 형태로 출력하려면 문자열을 "wt"로 연결하여 구성한다.

예) ComboText:"가wt나wt다|라wt마wt바"

ComboCode는 Type이 "Combo" 또는 "ComboEdit"인 경우 저장될 코드 항목을 "|"로 연결하여 구성하며 위의 ComboText 항목의 개수와 동일 하여야 한다.

ShowCol 속성은 ComboText에서 다중컬럼 형태의 콤보를 사용할때 콤보 선택시 보여지는 기준 컬럼을 설정 한다. 설정하지 않은 경우 ComboText에 처음 문자열이 보여진다.

MultiLineText는 Type이 "Text"인 경우 다중라인 입력여부를 설정한다. 기본적으로 0값이 설정된다.

Wrap는 컬럼너비에 따라 자동 줄바꿈 여부를 설정한다. 기본적으로 0값이 설정된다.

TreeCol은 트리형태 조회의 경우 기준컬럼 여부를 설정한다.

KeyField는 데이터의 필수입력 항목 여부를 설정하는 것으로 1인 경우 저장 함수 호출 시 셀에 데이터가 없는 경우 경고 메시지를 표시하고, Edit 하도록 유도한다. 기본적으로 0값이 설정된다.

CalcLogic은 해당 데이터에 대한 계산 공식을 설정한다. 컬럼 데이터의 값이 공식에 사용 되는 경우 "|"로 감싸서 공식에 설정한다. 기본적으로 공식없는 ""으로 설정된다.



예를 들어, 5컬럼 값에 2를 곱해서 3컬럼 값을 더해야 한다면 공식은 " $|5| * 2 + |3|$ "와 같이 설정된다. 공식이 설정 된 경우 다른 컬럼의 값이 바뀌었거나 조회했을 때 자동으로 계산 처리한다.

컬럼번호가 아니라 다른컬럼의 SaveName 을 사용하여 공식을 작성할 수도 있다. 예를 들어 위 공식에서 5컬럼의 SaveName이 "pay"였다고 하면 공식은 " $|pay|*2+|3|$ "이라고 설정해도 같은 효과를 볼 수 있다.

ColMerge는 데이터 컬럼에 대한 세로 머지 가능 여부를 설정한다. 기본적으로 1값이 설정된다.

Hidden는 데이터 컬럼 숨김 여부를 설정한다.

ImgWidth는 Type이 "Image" 인 경우 표시되는 이미지의 너비값을 설정한다. 기본적으로 0값이 설정되며 0으로 설정하는 경우 표시되는 이미지는 원본 사이즈로 표시된다.

ImgHeight는 Type이 "Image" 인 경우 표시되는 이미지의 높이값을 설정한다. 기본적으로 0값이 설정되며 0으로 설정하는 경우 표시되는 이미지는 원본 사이즈로 표시된다

(Image 타입을 사용하는 경우, 컬럼의 이미지 사이즈가 동일하다면 **ImgWidth,ImgHeight** 속성을 설정하는 것으로 조회 속도를 향상시킬 수 있다)

PopupCode는 PopupText에 매핑되는 코드값을 "|"로 연결하여 구성하며 PopupText 항목의 개수와 동일하여야 한다. 이 속성을 설정하는 경우에는 팝업 메뉴에서 항목을 선택 시 해당 컬럼의 옆 컬럼에 매핑되는 코드 값이 설정된다.

PopupText는 마우스 오른쪽 버튼 클릭시 보여지는 팝업 메뉴 목록을 설정할 항목을 "|"로 연결하여 구성한다.

UpdateEdit는 트랜잭션 상태가 조회인 데이터에 대해 Edit 가능 여부를 설정하며, 기본적으로 1값이 설정된다.

InsertEdit는 트랜잭션 상태가 입력인 데이터에 해서 Edit 가능 여부를 설정하며, 기본적으로 1값이 설정된다.

LevelSaveName은 트리 형태의 조회인 경우 데이터를 저장할 때 사용하는 트리 레벨의 변수 명을 설정한다. 변수명을 설정하지 않을 경우 해당 행의 트리 레벨은 서버로 전달되지 않는다.

ButtonUrl는 컬럼타입이 Popup 일때 Format이 날짜포맷일 경우 해당 버튼이미지를 변경하는 속성이다. 기본적으로 날짜버튼이미지로 설정된다.
(변경할 이미지는 테마폴더 안에 넣는다. 이미지 사이즈는 12*12)

TrueValue, FalseValue는 CheckBox 형태인 컬럼에서 기존에 지원하던 0,1 이외의 지정한 값을 사용할 수 있도록 사용자가 지정할 수 있게 하는 기능이다. 예를 들어 TrueValue:"T", FalseValue:"F" 로 지정하게 되면 1 대신 "T", 0 대신 "F" 를 사용하여 값을 나타낼 수 있다.

MaximumValue는 Format이 숫자와 관련된 포맷인 Integer, Float, NullInteger, NullFloat 일때, 편집시 입력할 수 있는 최대값을 설정한다.

MinimumValue는 Format이 숫자와 관련된 포맷인 Integer, Float, NullInteger, NullFloat 일때, 편집시 입력할 수 있는 최소값을 설정한다.

PointCount는 컬럼타입이 Float 인 경우 소수점 이하의 자리수를 설정한다.
설정하지 않은 경우 Format 설정값을 따르며 설정한 경우 Format 설정값은 무시되고 재설정 된다.

FullInput은 컬럼타입이 단일행 문자열인 경우 EditLen 만큼 모두 입력해야 하는 경우 설정한다. 기본적으로 0값이 설정된다.

ToolTipText는 풍선도움말에 표시할 문자열을 설정한다.

Ellipsis는 컬럼 텍스트의 표시글자가 잘릴경우 말줄임 여부를 설정한다. 기본적으



로 0값이 설정된다

BackColor는 컬럼의 배경색상을 설정한다.

FontColor는 컬럼의 폰트색상을 설정한다.

※ 세부 정보는 [InitColumns 메소드](#)의 설명 참조

4.3.4. 시트초기화 단순화

시트 초기화시 헤더에 타이틀과 각 컬럼의 속성이 각각 InitHeader와 InitColumns 메서드를 통해 설정되어 컬럼이 많아지는 경우 타이틀에 해당하는 컬럼을 찾기가 어려운 점이 있다. 따라서 이러한 부분을 하나의 함수를 통해 초기화 하는 방법도 제공한다.

```
//배열 선언
var initdata = {};
//SetConfig
initdata.Cfg = {SearchMode:smLazyLoad,Page:50,MergeSheet:msHeaderOnly};
//InitHeaders의 두번째 인자
initdata.HeaderMode = {Sort:1,ColMove:1,ColResize:1,HeaderCheck:0};
//InitColumns + Header Title
initdata.Cols = [
{Header:"결재여부|결재여부",Type:"CheckBox",Width:80,SaveName:"A",ColMerge:0},
{Header:"신청인|신청인",Type:"Text",Width:80,ColMerge:1},
{Header:"신청일자|신청일자",Type:"Date",Width:100,Format:"Ymd"},
{Header:"신청금액|신청금액",Type:"AutoSum",Width:85,SaveName:"B",Format:"Integer"},
{Header:"기간|시작일",Type:"Date",Width:100,SaveName:"C",Format:"Ymd",ColMerge:0},
{Header:"기간|종료일",Type:"Date",Width:100,SaveName:"D",Format:"Ymd",ColMerge:0},
{Header:"시간|시작",Type:"Date",Width:100,Format:"Hm",ColMerge:1,Align:"Center"},
{Header:"시간|종료",Type:"Date",Width:100,Format:"Hm",ColMerge:1,Align:"Center"}
];
//초기화
IBS_InitSheet(mySheet, initdata);
```

4.4 기능 구현하기

4.4.1 조회

```
IBSheet객체.DoSearch(PageUrl, [CondParam])
```



Parameter	Type	필수여부	Remark
PageUrl	String	필수	조회 처리할 페이지 파일 이름
CondParam	String	선택	조회조건 QueryString, Default=""

CondParam 인자는 조회조건을 "조건명1=값1&조건명2=값2" 식으로 QueryString 을 구성하여 설정한다.

문자열 조합으로 CondParam 인자를 구성하기 어려울 때 ibsheetinfo.js에서 제공하는 자바스크립트 함수인

FormQueryStringEnc()을 이용할 경우 좀더 쉽게 조회조건을 구성할 수 있다.

```
mySheet.DoSearch("list.jsp", "stockNo=1234&stockNm=한국");
```

```
mySheet.DoSearch("list.jsp", FormQueryStringEnc(document.forms[0]));
```

4.4.2 입력

IBSheet객체.DataInsert([Row],[Level])

데이터 행을 신규 생성하고자 할 때 이 함수를 이용한다.

호출되면 단위 데이터 행 개수 만큼 신규입력 행이 생성되고, Type이 Status인 컬럼에 "입력"이라는 글자를 표시하고, 내부적으로 "I"라는 코드값을 갖는다.

Row인자에 따라 생성되는 행의 위치는 다음과 같이 결정된다.

Row	생성 행	Level
Row < 0	마지막 행에 생성	0레벨
Row >= 전체 행	마지막 행에 생성	0레벨
Row < 데이터첫행	첫 행에 생성	0레벨
그 외	해당 행 아래에 생성	설정 레벨
기 본	선택 된 행 아래에 생성	설정 값이 없는 경우 선택 행의 자식 레벨

Level인자는 값을 설정하지 않으면 기본적으로 선택 행의 자식 레벨로 생성한다. 만약 선택 행이 삭제 상태라면 에러 메시지를 표시하고 신규 생성 처리를 중단한다.

4.4.3 수정

데이터를 수정하면 Type이 Status인 컬럼에 "수정"이라고 표시하고 내부적으로 "U"라는 코드 값을 갖는다. 기존 상태가 "입력" 이라면 기존 상태 그대로 유지 하며, 기존 상태가 "삭제" 인 경우 모든 데이터는 수정할 수 없다.

수정 상태를 기록하기 위해 반드시 Type이 Status인 컬럼이 존재해야 한다.

4.4.4 삭제

Type이 DelCheck 인 컬럼을 CheckBox를 Check 할때 Status인 컬럼에 "삭제"라고 표시하고 내부적으로 "D"라는 코드값을 갖는다.

그리고, 모든 Edit 가능한 셀이 Edit 불가능으로 설정된다. 다시 UnCheck 하면 기존에 상태값을 그대로 표시되고, Edit 가능 설정된다.

기존 상태가 "입력" 인 행을 Check하면 DB에 없는 데이터이므로 바로 행을 제거 한다.

Type이 Status인 컬럼과 DelCheck 인 컬럼이 존재 할 경우 처리 가능하다.

4.4.5 저장

```
IBSheet객체.DoSave(PageUrl, [SubParam], [Col] )
```

데이터의 트랜잭션 상태 또는 특정 컬럼 데이터에 따라 저장 처리한다.

Col 인자에 아무것도 설정하지 않는 경우 트랜잭션 상태가 조회가 아닌 데이터 행만 저장 처리하고, Col에 특정 인자가 있는 경우 해당 컬럼에 값이 있는 데이터만 저장 처리한다.

해당 컬럼이 CheckBox 형태 인 경우 CheckBox에 체크된 것만 저장 처리한다.

```
IBSheet객체.DoSave("save.jsp", "txtid=khlee&seq=1");
```

4.4.6 소계

```
IBSheet객체.ShowSubSum(info)
```

특정 컬럼을 기준으로 조회 데이터 사이에 소계 행을 삽입한다.

소계행을 만들기 위해서는 기준컬럼(1개)와 계산 컬럼(1개이상)가 필요하다. StdCol로 기준컬럼을 설정하고, SumCols로 계산값이 보여져야 할 컬럼을 구분자 "|"로 연결하여 넣으면, **메서드 호출시 바로 소계행이 삽입 되는 게 아니라, 설정 후 다음 조회 작업시 데이터 사이에 소계값이 함께 표현된다.**

이 외에도 누계를 포함시키거나, 소계타이틀을 변경하는 등의 기능도 있다.

```
var info = [{StdCol:1 , SumCols:"3|6|7"}];
IBSheet객체.ShowSubSum (info);
```

4.4.7 엑셀다운

IBSheet객체.Down2Excel (params)

시트의 내용을 엑셀파일로 다운로드 한다.

이 메서드의 동작방식은 다음과 같다.

1. Down2Excel()메서드를 호출하면 시트의 내용이 Multipart 형태로 서버의 Down2Excel.jsp 파일에 전달된다.
2. Down2Excel.jsp 파일에서는 넘어온 내용을 POI라이브러리를 이용하여 xls 포맷의 파일을 생성하고, 다시 화면쪽으로 다운로드 시킨다.

따라서 이 메서드를 사용하려면 다음과 같은 준비과정이 필요하다.

1. Down2Excel.jsp 파일이 서버에 존재하며, ibsheet.cfg 파일에 jsp파일에 대한 경로가 정확하게 기재 되어 있어야 한다.
2. 서버에 POI 3.9 라이브러리가 올라와 있어야 한다.
3. **Down2Excel.jsp 파일을 편집기로 열어, 사용하는 서버의 Root 경로와 Encoding을 설정해 준다.**

메서드를 사용하면 단순히 시트에 있는 내용뿐 아니라 특정 위치에 특정글자를 같이 다운로드 받을 수도 있으며, 특정 행이나 열을 제외하고 다운로드 받는 등 여러가지 기능을 제공한다.

```
var params = {FileName:"myFile.xls",SheetName:"sub Total",Merge:1,SheetDesign:1 };
IBSheet객체.Down2Excel (params);
```

4.4.8 이벤트 사용

```
function IBSheet객체_이벤트명(인자)
```

아이비시트의 이벤트는 사용하는 **객체명_이벤트 명** 형태의 스크립트 함수를 만들어 두고, 이벤트 발생시 행해야 하는 액션을 함수 안에 기술하면 된다.

```
function mySheet_OnChange(row,col,VALUE){
    if(mySheet.ColSaveName(col) == "DT_price"  && VALUE > 5000 ){
        alert("가격 제한 범위를 넘어셨습니다.");
        mySheet.ReturnCellData( row , col);
    }
}
```

4.5 조회 XML 구현하기

조회 함수를 호출하면 인자로 넘긴 URL로 페이지에 연결한다. XML 형태로 구성된 페이지의 내용을 읽어 들여 데이터로 표현한다.

➤ 조회된 내역이 있는 경우

조회 데이터를 담고 있는 페이지는 다음과 같은 구조의 XML로 구성된다.

설명	XML 구성
대분류 시작	<SHEET>
데이터 시작	<DATA>
1행 시작	<TR>
1행의 0컬럼	<TD>만도</TD>
1행의 1컬럼	<TD>홍길동</TD>



1행 종료	</TR>
2행 시작	<TR>
2행의 0컬럼	<TD>두산</TD>
2행의 1컬럼	<TD>택트인</TD>
2행 종료	</TR>
데이터 종료	</DATA>
대분류 종료	</SHEET>

➤ XML 생성 시 주의 사항

- XML의 인코딩 타입은 반드시 **UTF-8**로 해야하며, 그렇지 않으면 한글등을 데이터로 사용할 때 글자가 깨지게 된다.
- XML의 내용 중 HTML 관련 태그 (html, body, input 등)나 Script 관련 태그 (JavaScript, alert 등)은 사용 불가능하며 만약 사용 시 XML오류가 발생한다.
- 다음과 같은 특수 문자는 사용 불가능 하므로 변환 문자로 변환하여 XML을 생성한다. 그대로 사용할 경우 XML오류가 발생한다.

특수문자	&	<	>
변환문자	&	<	>

Mxml 버전이 2.0 이전 버전에서는 다음의 특수 문자도 변환문자로 변경해야 한다.

특수문자	'	"
변환문자	'	"

특수문자를 변환문자로 변환하지 않고 CDATA Section를 사용하면 XML 오류가 발생하지 않는다.

오류 예	<TD>사장 & 사원</TD>	XML오류
변환문자 예	<TD>사장 & 사원</TD>	사장 & 사원
CDATA 예	<TD><![CDATA[사장 & 사원]]></TD>	사장 & 사원
	<TD><![CDATA[사장 & 사원]]></TD>	사장 & 사원

4.6 저장 XML 구현하기

저장 함수를 호출하면 인자로 넘긴 URL로 페이지에 연결한다.

저장은 <XML> 형식을 사용하지 않고 기존 param 형식을 사용한다.

저장 처리 후 처리 결과는 XML 형태로 구성되며 페이지의 내용을 읽어 들여 결과를 표현한다.

➤ 저장 처리 결과 표시 형태

저장 처리를 완료하고 전체 데이터에 대해서 처리 완료 여부를 XML로 반환하는 형태이다

설명	XML 구성
대분류 시작	<SHEET>
처리결과 메시지	<Message>저장되었습니다.</Message>
XML 포맷에 대한 param 정보	<Result
결과값	Code="0 or -1"
	/>
대분류 종료	</SHEET>

➤ 처리 결과 설정

저장 처리 결과로 설정할 수 있는 값은 다음과 같다.

트랜잭션 결과	처리 결과		
	트랜잭션 상태	메세지 표시 결과	
Code	양수 : 성공	Message	"저장되었습니다" 표시
	음수 : 실패		"저장이 실패 되었습니다." 표시

4.7 조회 JSON 구현하기

javascript언어의 특성상 json 유형은 대소문자를 정확하게 지켜야 한다.

조회 함수를 호출하면 인자로 넘긴 URL로 페이지에 연결한다. JSON 형태로 구성



된 페이지의 내용을 읽어 들여 데이터로 표현한다.

➤ **조회된 내역이 있는 경우**

컬럼 초기화시 설정한 SaveName 으로 Key값을 설정한 경우 해당 SavName의 컬럼과 매핑되어 조회된다.

[savename을 사용하는 예]

```
{
  Etc:{"USER":"황상구","CRTDATE":"20130120"}
  Message:"조회가 정상 처리되었습니다.",
  Data: [
    {SANO: 070712, SANM:'홍길동', CLSCD:'5',DEPTCD:'054'},
    { SANO: 020457, SANM:'김대한', CLSCD:'4',DEPTCD:'040'}
  ]
}
```

4.8 저장 JSON 구현하기

저장 함수를 호출하면 인자로 넘긴 URL로 페이지에 연결한다.

저장은 JSON 형식을 사용하지 않고 기존 param형식을 사용한다.

저장 처리 후 처리 결과는 JSON 형태로 구성되며 페이지의 내용을 읽어 들여 결과를 표현한다.

➤ **저장 처리 결과 표시 형태**

저장 처리를 완료하고 전체 데이터에 대해서 처리 완료 여부를 JSON으로 반환하는 형태이다

```
{
  Result : {Code:0, Message:'저장되었습니다.'}
}
```

➤ 처리 결과 설정

저장 처리 결과로 설정할 수 있는 값은 다음과 같다.

트랜잭션 결과	처리 결과		
	트랜잭션 상태	메세지 표시 결과	
Code	양수:성공	Message	"저장되었습니다" 표시
	음수:실패		"저장이 실패 되었습니다." 표시

조회/저장 json 데이터를 구성할 때 내용 중에 "줄 넘김" 문자가 포함되어 있을 경우 json유형이 깨져 조회가 안되는 경우가 있다.

가령 데이터가 아래와 같은 경우이다.

```
out.println("{}");
out.println("Result:{Code:0, Message:'저장되었습니다\n다음 화면으로 이동합니다.'}");
out.println("{}");
```

위와 같은 경우 실제 시트가 받아들이는 json 데이터는 아래와 같다.

```
{
  Result:{Code:0, Message:'저장되었습니다
다음 화면으로 이동합니다.'}
}
```

이런 경우, json문자는 파싱 중에 오류가 발생하게 된다. 따라서 줄넘김 문자를 \r\n 으로 치환하여 주어야 한다.

치환된 문자

```
out.println("{}");
out.println("Result:{Code:0, Message:'저장되었습니다\r\n다음 화면으로 이동합니다.'}");
out.println("{}");
```

만약 데이터가 위와같이 jsp안에 하드코딩 된 것이 아니라 DB로부터 가져오는



문자열이라면 치환하는 방법은 아래와 같이 두가지 이다.

1. 쿼리에서 수정하기.

쿼리에서 줄넘김 문자를 찾아 'WrWn' 문자열로 변경해 준다.

oracle db 예)

SELECT replace(contents , chr(10) , 'WrWn') from TABLE

2. 조회 json데이터를 만드는 과정에서 치환

예)

"CONTENTS:"+(rs.getString("CONTENTS")).replace("Wn", "WrWn")



(주) 아이비 리더스

Inno Biz Leaders Co., Ltd.



Chapter 5. 개발 실습



(주) 아이비 리더스

Inno Biz Leaders Co., Ltd.



5. 개발 실습

Oracle DB 기본적으로 갖고 있는 EMP 테이블을 기준으로 예제화면을 만들어 본다.

5.1 IBSheet 객체 생성 및 기본 속성 설정하기

lbsheetinfo.js 내의 createIBSheet 함수를 사용하여 IBSheet 객체를 생성하고 Body의 OnLoad 이벤트에서 생성된 객체에 대한 기본 속성을 설정한다.

➤ 객체 생성하기

```
<body onload="LoadPage()">
  <script type="text/javascript">
    createIBSheet("mySheet", "100%", "100%");
  </script>
</body>
```

➤ 기본 속성 설정하기

```
<script type="text/javascript" src="/js/ibsheet/ibsheet.js"> </script>
<script type="text/javascript" src="/js/ibsheet/ibsheetinfo.js"> </script>
<script type="text/javascript">
function LoadPage() {
  var cfg = {FrozenCol:3,SearchMode:2,Page:100};
  mySheet.SetConfig(cfg);
  var header = [{Text:"상태|삭제|직책|부서코드|ID|이름|입사일|책임자|급여|COMM",Align:"Center"}];
  var info = {Sort:0,ColMove:1,ColResize:1};
  mySheet.InitHeaders(header,info);
  var cols = [
    {Type:"Status",Width:60,SaveName:"sStatus",Align:"Center"},
    {Type:"DelCheck",Width:60,SaveName:"sDelete",Align:"Center"},
    {Type:"Text",Width:100,SaveName:"JOB",Align:"Center"},
    {Type:"Text",Width:100,SaveName:"DEPTNO",Align:"Center"},
  ]
}
```

```

        {Type:"Text",Width:60,SaveName:"EMPNO",Align:"Center"},
        {Type:"Text",Width:150,SaveName:"ENAME",Align:""},

        {Type:"Date",Width:120,SaveName:"HIREDATE",Format:"Ymd",Align:"Center",EditLen:8},
        {Type:"Text",Width:120,SaveName:"MGR",Align:"Center"},

        {Type:"Int",Width:120,SaveName:"SAL",Align:"Right",Format:"NullInteger"},

        {Type:"Int",Width:60,SaveName:"COMM",Align:"Right",Format:"Integer"}
    ];
    mySheet.InitColumns(cols);
}
</script>

```

위와 같이 소스를 생성하고 실행하면 다음과 같은 모습의 오브젝트가 생성된 것을 볼 수 있다.

상태	직책	직책	부서 코드	ID	이름	입사일	퇴직자	급여	COMM

5.2 각종 기능 구현 하기

트랜잭션과 관련된 각종 기능 버튼을 생성하고 기능버튼이 처리해야할 작업을 자바스크립트 함수로 구현한다.

입력, 행복사 기능을 다음과 같이 구현한다.

```

function doAction(sAction) {
    switch(sAction) {
        case "Insert":           //입력
            mySheet.DataInsert();
            break;
        case "Copy":             //행복사
            mySheet.DataCopy();

```



```
        break;
    }
}

<!-- 입력 버튼 -->
<input type="button" value="입력" onClick="doAction('Insert')">
<!-- 행복사 버튼 -->
<input type="button" value="행복사" onClick="doAction('Copy')">
<!-- 조회 조건 -->
<table cellpadding="0" cellspacing="1" border="0" width="100%">
<tr>
    <td bgcolor="#FFFFFF">Employee Name</td>
    <td bgcolor="#FFFFFF">
        <input type="text" name="sa_name">
    </td>
    <td bgcolor="#FFFFFF">Salary</td>
    <td bgcolor="#FFFFFF">
        <input type="text" class="on_curr1" name="sal" value="0">
    </td>
    <td bgcolor="#FFFFFF">HireDate</td>
    <td bgcolor="#FFFFFF">
        <input type="text" class="on_date" name="hdate" >
    </td>
</tr>
</table>
```

5.3 조회하기

조회 기능을 처리 하기 위해서는 다음의 4가지 과정을 처리한다.

- 조회 함수 생성하기(필수)
- 다음 페이지 조회 이벤트 처리하기(선택)
- 조회 XML 생성하기(필수)
- 조회 완료 이벤트 처리하기(선택)



➤ **조회 함수 생성하기(필수)**

조회 처리할 조회 함수를 다음과 같이 자바스크립트로 생성한다.

```
function doSearch() { //조회조건
    var param = "sa_name="+document.frm.sa_name.value+"&sal="+
        document.frm.sal.value+"&hdate="+document.frm.hdate.value;
    //조회하기 (페이지명, 조회조건);
    mySheet.DoSearch("dataList.jsp",param);
}
```

조회 조건 인자는 여러 개 조회 조건을 조건명과 조건값을 "="로 연결하고, 여러 개 조건이 사용될 경우 "&"로 연결하여 다음과 같이 구성한다.

조건1=값1&조건2=값2&조건3=값3

➤ **조회 XML 생성하기(필수)**

조회 함수에서 설정한 조회XML 페이지를 구성한다.

다음은 dataList.jsp를 구현한 샘플 페이지 이다.

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"
    import="java.sql.*"
    %>
<%
Connection conn = null;
PreparedStatement pstmt = null;
ResultSet rs = null;
String url = "jdbc:oracle:thin:@192.168.0.99:1521:hubdb";
String id = "scott";
String pwd = "tiger";
String driver = "oracle.jdbc.driver.OracleDriver";
String query = "select JOB,DEPTNO,EMPNO,ENAME,"
                + "substr(to_char(HIREDATE,'YYYYMMDD'),0,8) HIREDATE,"
                + "MGR,SAL,COMM from emp where ename like '%||?'";
try{
```



```
Class.forName(driver);
conn = DriverManager.getConnection(url,id,pwd);
pstmt = conn.prepareStatement(query);
pstmt.setString(1, request.getParameter("sa_name"));
rs = pstmt.executeQuery();

//JSON 문자열 생성
out.println("{}");
out.println(" DATA:[");
while(rs.next()){
    //둘중에 하나를 사용할 것.
    //1. 컬럼 순서에 맞춰 데이터 생성
    out.println("{C1:'"+rs.getString("JOB")+"',"
        + "C4:'"+rs.getString("DEPTNO")+"',"
    out.println(",C5:'"+rs.getString("EMPNO")+"',"
        + "C6:'"+rs.getString("ENAME")+"',"
        + "C7:'"+rs.getString("HIREDATE")+"',"
    out.println(",C8:'"+rs.getString("MGR")+"',"
        + "C9:'"+rs.getString("SAL")+"',"
        + "C10:'"+rs.getString("COMM")+""},");
//2. SaveName을 이용한 방법
    out.println("{JOB:'"+rs.getString("JOB")+"',"
        + "DEPTNO:'"+rs.getString("DEPTNO")+"',"
        + "EMPNO:'"+rs.getString("EMPNO")+"',"
    out.println(",ENAME:'"+rs.getString("ENAME")+"',"
        + "HIREDATE:'"+rs.getString("HIREDATE")+"',"
    out.println(",MGR:'"+rs.getString("MGR")+"',"
        + "SAL:'"+rs.getString("SAL")+"',"
        + "COMM:'"+rs.getString("COMM")+""},");
}
    out.println("]]");
```




```
}catch(Exception ex){
    String msg = ex.getMessage();
    //오류 발생
    out.println("{}");
    out.println("Result : {Code:-1, Message:'오류가 발생하였습니다.\n'+msg+'}");
    out.println("{}");

}finally{
    rs.close();
    pstmt.close();
    conn.close();
}
%>
```

위와 같이 작성된 페이지는 웹서버에서 컴파일이 완료되어 다음과 같은 형태로 나타나게 된다.

■ 조회된 내역이 없는 경우

```
{ data:[ ] }
```

■ 조회된 내역이 있는 경우

```
{ DATA:[ { JOB:'ANALYST',DEPTNO:'10',EMPNO:'8888' ,ENAME:'black',HIREDATE:'19991111' ,MGR:'8888',SAL:'8888',COMM:'888'},
{ JOB:'CLERK',DEPTNO:'20',EMPNO:'7566' ,ENAME:'JONES',HIREDATE:'19810402' ,MGR:'7839',SAL:'2975',COMM:'200'},
{ JOB:'ANALYST',DEPTNO:'30',EMPNO:'7654' ,ENAME:'MARTIN',HIREDATE:'19810928' ,MGR:'7698',SAL:'1250',COMM:'1400'},
{ JOB:'MANAGER',DEPTNO:'30',EMPNO:'7698' ,ENAME:'BLAKE',HIREDATE:'19810501' ,MGR:'7839',SAL:'2850',COMM:'470'},
{ JOB:'MANAGER',DEPTNO:'10',EMPNO:'7782' ,ENAME:'CLARK',HIREDATE:'19810609' ,MGR:'783
```

```
9,SAL:'2450',COMM:'800'},
{JOB:'ANALYST',DEPTNO:'20',EMPNO:'7788' ,ENAME:'SCOTT',HIREDATE:'19870419' ,MGR:'7566'
,SAL:'3000',COMM:'555'},
{JOB:'PRESIDENT',DEPTNO:'10',EMPNO:'7839' ,ENAME:'KING',HIREDATE:'19811117' ,MGR:'null',
SAL:'5000',COMM:'3000'},
{JOB:'SALESMAN',DEPTNO:'30',EMPNO:'7844' ,ENAME:'TURNER',HIREDATE:'19810908' ,MGR:'7
698',SAL:'1500',COMM:'1350'},
{JOB:'CLERK',DEPTNO:'20',EMPNO:'7876' ,ENAME:'ADAMS',HIREDATE:'19870523' ,MGR:'7788',S
AL:'1100',COMM:'900'},
{JOB:'CLERK',DEPTNO:'30',EMPNO:'7900' ,ENAME:'JAMES',HIREDATE:'19811203' ,MGR:'7698',S
AL:'950',COMM:'700'},
{JOB:'ANALYST',DEPTNO:'20',EMPNO:'7902' ,ENAME:'FORD',HIREDATE:'19811203' ,MGR:'7566',
SAL:'3000',COMM:'1200'},
{JOB:'CLERK',DEPTNO:'10',EMPNO:'7934' ,ENAME:'MILLER',HIREDATE:'19820123' ,MGR:'7782',S
AL:'1300',COMM:'1000'}, ]}]
```

➤ 조회 완료 이벤트 처리하기(선택)

조회가 완료되고 그뒤에 어떤 처리가 필요한 경우 완료 이벤트에 대한 처리를 구현한다.

```
function mySheet_OnSearchEnd() {
    alert("조회가 완료되었습니다.");
}
```

위와 같이 조회 기능을 완료 하여 처리하면 다음과 같은 결과 화면이 생성된다.

일련	직책	부서코드	ID	이름	입사일	백분율	금액	COMM
	ANALYST	10	8888	black	1999/11/11	8888	8,888	888
	CLERK	20	7566	JONES	1981/04/02	7839	2,975	200
	ANALYST	30	7654	MARTIN	1981/03/28	7698	1,250	1,400
	MANAGER	30	7698	BLAKE	1981/05/01	7839	2,850	470
	MANAGER	10	7782	CLARK	1981/06/06	7839	2,450	800
	ANALYST	20	7788	SCOTT	1987/04/19	7566	3,000	555
	PRESIDENT	10	7839	KING	1981/11/17	null	5,000	3,000
	SALESMAN	30	7844	TURNER	1981/09/08	7698	1,500	1,350
	CLERK	20	7876	ADAMS	1987/05/23	7788	1,100	900
	CLERK	30	7900	JAMES	1981/12/03	7698	950	700
	ANALYST	20	7902	FORD	1981/12/03	7566	3,000	1,200
	CLERK	10	7934	MILLER	1982/01/23	7782	1,300	1,000

5.4 저장하기

저장 기능을 처리 하기 위해서는 다음의 4가지 과정을 처리한다.

- 저장 함수 생성하기(필수)
- Validation 확인하기(선택)
- 저장 XML 생성하기(필수)
- 저장 완료 이벤트 처리하기(선택)

➤ 저장 함수 생성하기(필수)

저장 처리할 저장 함수를 다음과 같이 자바스크립트로 생성한다.

```
function doSave()
{
    //저장하기 (페이지명, 저장추가정보);
    var param = "sa_name="+document.frm.sa_name.value+"&sal="+
    document.frm.sal.value+"&hdate="+document.frm.hdate.value;
    mySheet.DoSave("save.jsp");}
```

저장 처리할 페이지명을 설정하고 추가 저장할 정보를 설정하여 조회 함수를 호출하면 내부적으로 트랜잭션 상태가 "조회"가 아닌 행의 값을 해당 데이터의 SaveName 으로 조합하여 저장 페이지로 넘긴다. SaveName은 InitColumns 함수를 통해서 설정된다.

예를 들어 위의 예제와 같은 데이터의 경우 다음과 같이 트랜잭션 데이터를 조합하여 저장 처리한다.

```
sStatus=U&sDelete=0&JOB=CLERK&DEPTNO=20&EMPNO=1111&ENAME=SMITH1&HIRED
ATE=19991231&MGR=8888&SAL=550&COMM=100&sStatus=U&sDelete=0&JOB=CLERK&
DEPTNO=20&EMPNO=2222&ENAME=CHRIS2&HIREDATE=20010222&MGR=8888&SAL=57
0&COMM=120
```

➤ Validation 확인하기(선택)

저장 함수를 이용하여 저장 하기 전에 각 데이터에 대한 Validation 확인을 OnValidation 이벤트에 정의한다.

업무에 따라 발생할 수 있는 각종 Validation은 이 이벤트에 구현한다.

```
function mySheet_OnValidation(Row, Col, Value) {
    switch(Col) {
```

```
case 8:
    if (Value > 1000 ) {
        alert("1000 이상을 설정할 수 없습니다.");
        mySheet.ValidateFail(1);
        mySheet.SetSelectCell(Row, 8);
    }
    break;
}
```

OnValidation 이벤트에서 확인하고 유효하지 않은 데이터가 있어서 처리를 중단해야 한다면 ValidateFail 함수 인자에 1을 설정하고 SetSelectCell 함수를 이용하여 포커스를 옮긴다.

이후 저장 처리는 중단된다.

➤ 저장 XML 생성하기(필수)

저장할 데이터가 모두 유효하다면 다음과 같은 저장 처리 페이지를 구성한다.

결과 컬럼에 저장 처리 결과를 설정하는 방식의 저장 XML을 다음과 같이 구성한다.

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"
    import="java.sql.*"

    %>
<%

//디비 접속 관련
Connection conn = null;
PreparedStatement pstmt = null;
ResultSet rs = null;
String url = "jdbc:oracle:thin:@192.168.0.99:1521:hubdb";
```



```
String id = "scott";
String pwd = "tiger";
String driver = "oracle.jdbc.driver.OracleDriver";

//쿼리들
String insertQ = "insert into
emp(job,deptno,empno,ename,hiredate,mgr,sal,comm) values
(?,?,?,to_date(?, 'YYYYMMDD'),?,?,?)";
String updateQ = "update emp set hiredate =
to_date(?, 'YYYYMMDD'),sal=?,comm=? where empno = ?";
String deleteQ = "delete from emp where empno = ?";

//시트로부터 넘어온 내용들
String[] sStatus    = request.getParameterValues("sStatus");
String[] JOB        = request.getParameterValues("JOB");
String[] DEPTNO     = request.getParameterValues("DEPTNO");
String[] EMPNO      = request.getParameterValues("EMPNO");
String[] ENAME      = request.getParameterValues("ENAME");
String[] HIREDATE   = request.getParameterValues("HIREDATE");
String[] MGR        = request.getParameterValues("MGR");
String[] SAL        = request.getParameterValues("SAL");
String[] COMM       = request.getParameterValues("COMM");
try{
    Class.forName(driver);
    conn = DriverManager.getConnection(url,id,pwd);
    conn.setAutoCommit(false);

    for(int row=0;row<sStatus.length;row++){
        if("I".equals(sStatus[row])){
            pstmt = conn.prepareStatement(insertQ);
            pstmt.setString(1, JOB[row]);
```



```
pstmt.setString(2, DEPTNO[row]);
pstmt.setString(3, EMPNO[row]);
pstmt.setString(4, ENAME[row]);
pstmt.setString(5, HIREDATE[row]);
pstmt.setString(6, MGR[row]);
pstmt.setString(7, SAL[row]);
pstmt.setString(8, COMM[row]);
}else if("U".equals(sStatus[row])){
    pstmt = conn.prepareStatement(updateQ);
    pstmt.setString(1, HIREDATE[row]);
    pstmt.setString(2, SAL[row]);
    pstmt.setString(3, COMM[row]);
    pstmt.setString(4, EMPNO[row]);
}else{
    pstmt = conn.prepareStatement(deleteQ);
    pstmt.setString(1, EMPNO[row]);
}
pstmt.executeUpdate();
pstmt.clearParameters();
}
conn.commit();
%>
{
    Result : {Code:0, Message:'저장 되었습니다.'}
}
<%
}catch(Exception ex){
    String msg = ex.getMessage();
%>
{
    Result : {Code:-1, Message:'저장이 실패하였습니다.₩n<%=msg%>'}
}
```



```
<%  
}finally{  
    pstmt.close();  
    conn.close();  
}  
%>
```

데이터에 대해 저장 처리 완료 되면 다음과 같은 데이터가 최종적으로 생성된다.

저장 성공 시
{ Result : {Code:0, Message:'저장 되었습니다.'} }
저장 오류 시
{ Result : {Code:-1, Message:'오류가 발생하였습니다.'} }

➤ **저장 완료 이벤트 처리하기(선택)**

저장 처리가 완료되면 onSaveEnd 이벤트가 발생한다.

이 이벤트는 저장이 성공하든 실패하든 저장처리가 끝나면 발생하는 이벤트로 에러 메시지가 있는 경우 인자로 설정되어 이벤트가 발생한다.

onSaveEnd 완료 이벤트에 대한 처리를 구현한다.

```
function mySheet_onSaveEnd(code) {  
    if (code == 0) {  
        alert("저장에 성공하였습니다.");  
    } else {  
        alert("저장에 실패하였습니다.");  
    }  
}
```





(주) 아이비 리더스

Inno Biz Leaders Co., Ltd.



Chapter 6. 데이터 구조



(주) 아이비 리더스

Inno Biz Leaders Co., Ltd.

6. 데이터 구조

IB Sheet에서는 조회 및 저장 처리시 XML 과 JSON 포맷 형태를 지원한다.

6.1 XML

6.1.1 조회 XML

조회 함수를 통해 읽어들이는 XML 전체 구조는 다음과 같다.

```
<?xml version='1.0' ?>
<SHEET>
  <!-- 선택항목. ETC-DATA 요소 -->
  <ETC-DATA>
    <ETC KEY="key_name">key_value</ETC>
  </ETC-DATA>
  <!-- 선택항목. Message 요소 -->
  <MESSAGE>메시지</MESSAGE>
  <!-- 필수항목. DATA 요소 -->
  <DATA TOTAL="50"
    COLORORDER="prod_nm|pay|enter_date|book_chk"
    COLSEPARATOR="구분자">
    <!-- 필수항목. TR 요소 : 행단위 설정 -->
    <TR LEVEL="" HAVECHILD="">
      <!-- 필수항목. TD 요소 : 데이터단위 설정 -->
      <TD> 조회된 데이터 </TD>
    </TR>
  </DATA>
  <!-- 필수항목. Result 요소 -->
  <RESULT CODE="" MESSAGE=""></RESULT>
</SHEET>
```

위의 조회 XML에서 사용하는 XML요소는 다음과 같이 필수 항목과 선택 항목으로 나누어 진다.

요소명	필수여부	설 명
SHEET	필수	조회 데이터의 최상위 요소



DATA	필수	조회 데이터의 시작 요소
TR	필수	조회 데이터의 각 행 단위 설정 요소
TD	필수	조회 데이터의 셀 단위 설정 요소
ETC-DATA	선택	기타 데이터 설정 요소
MESSAGE	선택	서버 메시지 설정 요소
RESULT	선택	서버 처리 결과 설정 요소

6.1.1.1 SHEET 요소

조회된 데이터의 최상위를 의미 한다.

이 요소는 조회 XML의 최 상위 요소로 XML 기본 정의 아래 반드시 하나만 사용해야 한다. 이 요소 안에 사용되는 <DATA> 요소는 조회된 데이터를 의미하며 단 하나만 사용해야 한다. <ETC-DATA> 요소는 조회된 데이터 이외의 기타 데이터를 가져오는 요소로 여러 개 사용할 수 있다.

➤ 속성

속성	설 명
	없음

➤ Example

```

<!-- 조회된 내역이 없는 경우 -->
<?xml version="1.0" ?>
<SHEET>
  <DATA>
  </DATA>
</SHEET>

<!--일반 조회의 경우 -->
<?xml version="1.0" ?>
<SHEET>
  <DATA>
    <TR>
      <TD>R</TD>
      <TD>1234</TD>
      <TD>가나다라</TD>
    </TR>
  </DATA>
</SHEET>

```



- 제공 버전
7.0.0.0

6.1.1.2 DATA 요소

조회된 데이터 영역을 의미하는 요소로 일반적으로 <SHEET> 요소 안에 사용하여 이 요소 안의 데이터를 조회된 데이터로 표현한다.

이 요소 안에 사용되는 <TR> 요소는 조회된 각 행의 데이터를 의미하며, 하나도 사용되지 않은 경우 조회된 데이터가 없는 것으로 처리된다.

TOTAL 속성은 서버 페이징 조회 (실시간 조회)를 하는 경우에만 사용하며 전체 데이터 건수를 설정 한다. 이 속성 값에 따라 스크롤 사이즈가 설정된다.

COLORDER 속성은 <TR>요소 하단의 <TD>요소 순서가 컬럼의 순서와 다를 때 사용 한다. 이 속성을 사용할 경우 반드시 각 데이터 별로 SaveName을 설정해야 하며 조회한 데이터의 컬럼 순서를 SaveName 과 "|"를 연결한 문자열로 이 속성에 설정한다. 예를 들어 실제 컬럼 순서는 prod_nm -> count -> amt 라고 할때 ColOrder가 설정되지 않은 경우 <TD>는 반드시 prod_nm -> count -> amt 순서로 조회해야 한다. 조회 시 Data 요소를 <DATA COLORDER="amt | prod_nm | count"> 라고 설정하면 <TD>는 amt -> prod_nm -> count 순서로 조회해야 한다.

COLSEPARATOR 속성은 <TR> 요소 하단에 <TD>요소를 사용하지 않고 특정 구분자로 구분하여 조회할 때 설정 한다.

➤ 속성

속성	설 명
TOTAL	조회 할 전체 건수 실시간 서버 페이징 조회시 이 속성을 반드시 설정해야 함.
COLORDER	컬럼 순서를 SaveName 과 " "를 연결한 문자열
COLSEPARATOR	컬럼을 구분하는 문자열

➤ Example



```
<!-- 전체건수가 1000건이고, 조회건수가 100건인경우 -->
```

```
<?xml version="1.0" ?>
```

```
<SHEET>
```

```
  <DATA TOTAL="1000">
```

```
    <TR>(조회된 데이터 중략)</TR>
```

```
  </DATA>
```

```
</SHEET>
```

```
<!-- COLORDER 사용 조회 -->
```

```
<?xml version="1.0" ?>
```

```
<SHEET>
```

```
  <DATA COLORDER=" amt | prod_nm | count ">
```

```
    <TR>(조회된 데이터 중략)</TR>
```

```
  </DATA>
```

```
</SHEET>
```

```
<!-- COLSEPARATOR 사용 조회 -->
```

```
<?xml version="1.0" ?>
```

```
<SHEET>
```

```
  <DATA COLSEPARATOR="|">
```

```
    <TR> |001|10000|아이비리더스|1 </TR>
```

```
    <TR> |001|10000|아이비리더스|1 </TR>
```

```
  </DATA>
```

```
</SHEET>
```

➤ 제공 버전

7.0.0.0

6.1.1.3 TR 요소

조회된 데이터의 각 행 단위 데이터를 의미 한다.

이 요소는 <DATA>요소 하위에 사용되며 조회된 행의 개수만큼 반복 사용한다.

<DATA>요소 안에 이 요소가 없는 경우 조회된 데이터가 없는 것으로 처리 된다.

LEVEL 속성은 트리 구조 조회시 해당 행의 트리 레벨 정보를 설정한다.

최상위 레벨은 0으로 설정한다.

➤ 속성

속성	Type	설 명
BackColor	String	해당 행의 배경색상
Drag	Boolean	해당 행의 Drag 허용 여부 (Default = 1)
Edit	Boolean	해당 행의 Edit 허용 여부
Expand	Boolean	해당 행의 자식 레벨 데이터를 펼칠것인지 여부
FontColor	String	해당 행의 폰트색상
HaveChild	Boolean	트리형태일때 해당행의 Child 데이터가 있으면서 조회하지 않는경우 사용, 나중에 OnTreeChild Event가 발생할 수 있는 행 설정
Hidden	Boolean	해당 행의 숨김 여부
Level	Number	해당 행의 트리 레벨
Merge	Boolean	해당 행의 RowMerge 허용 여부
Sum	Boolean	해당 행의 합계 계산 포함 여부
TreeCheck	Boolean	트리 컬럼의 체크박스 값

➤ Example

```
<!-- 트리 형태의 데이터 조회하기 1-->
<?xml version="1.0" ?>
<SHEET>
  <DATA>
```



```
<TR LEVEL="0">(조회된 데이터 중략)</TR>
<TR LEVEL="1">(조회된 데이터 중략)</TR>
<TR LEVEL="2">(조회된 데이터 중략)</TR>
<TR LEVEL="2">(조회된 데이터 중략)</TR>
<TR LEVEL="1">(조회된 데이터 중략)</TR>
<TR LEVEL="2">(조회된 데이터 중략)</TR>
<TR LEVEL="2">(조회된 데이터 중략)</TR>
<TR LEVEL="2">(조회된 데이터 중략)</TR>
</DATA>
</SHEET>
```

<!-- 트리 형태의 데이터 조회하기 2. Child 데이터 조회하지 않고 나중에
조회하기-->

```
<?xml version="1.0" ?>
<SHEET>
  <DATA>
    <TR LEVEL="0">(조회된 데이터 중략)</TR>
    <TR LEVEL="1" HAVECHILD="1">(조회된 데이터 중략)</TR>
    <TR LEVEL="1" HAVECHILD="1">(조회된 데이터 중략)</TR>
  </DATA>
</SHEET>
```

➤ **제공 버전**

7.0.0.0

6.1.1.4 TD 요소

해당 행의 각 컬럼별 조회 데이터를 의미 한다.

이 요소는 <TR>요소 안에 사용되며 해당 행의 컬럼개수 만큼 반복 사용한다.

➤ 속성

속성	설 명
Align	정렬 설정
BackColor	해당 셀의 배경색상
ComboCode	콤보타입일경우 해당셀의 콤보 코드값을 설정
ComboText	콤보타입일경우 해당셀의 콤보 텍스트값을 설정
Edit	해당 셀의 Edit 허용 여부
FontBold	Bold 설정 여부
FontColor	해당 셀의 폰트색상
FontItalic	기울이기 설정 여부
FontStrike	취소선 설정 여부
FontUserline	밑줄 설정 여부
Image	해당 셀의 이미지 경로 (ImageList 인덱스 사용가능)
ToolTip	표시할 푼선도움말 문자열

➤ Example

```
<!--조회된 내역-->
<?xml version="1.0" ?>
<SHEET>
  <DATA>
    <TR BackColor="#FFFF00" FontColor="#00FF00" EDIT="0">
      <TD BackColor="#FF0000" Edit="0"> </TD>
      <TD FontColor="#00FF00" Edit="0"> </TD>
      <TD Image="0">006</TD>
      <TD>홍길동</TD>
      <TD>10000</TD>
```



```
<TD></TD>
<TD>20030201</TD>
<TD>200302</TD>
<TD>0201</TD>
<TD>770101-1234567</TD>
<TD>123-45-67890</TD>
<TD>1111-2222-3333-4444</TD>
<TD>가</TD>
<TD>abc</TD>
<TD>150-080</TD>
<TD>0</TD>
<TD>180101</TD>
<TD ComboText="대기|진행|완료" ComboCode="1|2|3">0101</TD>
<TD></TD>
</TR>
</DATA>
</SHEET>
```

➤ **제공 버전**

7.0.0.0

6.1.1.5 ETC-DATA 요소

조회 데이터 이외의 여분의 기타 데이터를 의미 한다.

이 요소는 IB Sheet 내의 데이터 이외의 데이터를 가져와 다른 용도로 사용하는 경우에 설정하여 사용한다. 이 요소를 통해 가져온 데이터는 EtcData Method를 통해 확인 할 수 있고 다른 값으로도 변경이 가능 하다.

➤ 속성

속성	설 명
KEY	대표 키 이름

➤ Example

```
<!--조회 시 사용하기-->
<?xml version="1.0" ?>
<SHEET>
  <ETC-DATA>
    <ETC KEY="name">홍길동</ETC>
    <ETC KEY="age">30</ETC>
  </ETC-DATA>
  <DATA>
    (조회된 데이터 중략)
  </DATA>
</SHEET>

<!--저장 시 사용하기-->
<?xml version="1.0" ?>
<SHEET>
  <ETC-DATA>
    <ETC KEY="name">홍길동</ETC>
    <ETC KEY="age">30</ETC>
  </ETC-DATA>
  <RESULT CODE="0" Message="결과 메시지"/>
```



</SHEET>

➤ 제공 버전

7.0.0.0

6.1.1.6 MESSAGE 요소

서버로부터 조회 처리 중 발생하는 메시지를 처리하기 위한 요소이다.

이 요소가 설정된 경우 조회 완료 후 발생하는 OnSearchEnd 이벤트에서 message 인자에 값이 설정되어 이벤트가 발생한다.

➤ 속성

속성	설 명
	없음

➤ Example

```
<!--조회 시 사용하기-->
<?xml version="1.0" ?>
<SHEET>
  <MESSAGE>조회가 완료되었습니다.</MESSAGE>
  <DATA>
    (조회된 데이터 중략)
  </DATA>
</SHEET>
```

➤ 제공 버전

7.0.0.0



6.1.1.7 RESULT 요소

조회 데이터의 결과를 의미 한다.

CODE 속성은 조회 처리 결과에 대한 코드이며 정수값을 설정 한다.

설정 값이 0 이상인 경우 정상 처리로 0보다 작은 경우는 실패 처리한다..

MESSAGE 속성은 처리 결과에 대한 메시지를 설정한다.

<MESSAGE> 요소와 같이 사용하는 경우는 <MESSAGE> 요소가 우선적으로 처리 되어 <RESULT> 요소에서 설정한 MESSAGE 속성은 무시 된다.

➤ 속성

속성	설 명
CODE	처리 결과 코드
MESSAGE	처리 결과 메시지

➤ Example

```
<!--조회 시 사용하기-->
<?xml version="1.0" ?>
<SHEET>
  <DATA>
    (조회된 데이터 중략)
  </DATA>
  <RESULT CODE="0" MESSAGE="조회가 완료되었습니다."></RESULT>
</SHEET>
```

➤ 제공 버전

7.0.0.0



6.1.2 저장 XML

저장 함수를 통해 읽어들이는 XML 전체 구조는 다음과 같다.

```
<!--전체 처리 결과를 가져오는 경우 -->
<?xml version='1.0' ?>
<SHEET>
  <!--선택항목. ETC-DATA 요소 -->
  <ETC-DATA>
    <ETC KEY="key_name">key_value</ETC>
  </ETC-DATA>
  <!--필수항목. RESULT 요소 -->
  <RESULT CODE="0"
    MESSAGE="완료 후 메시지">
  </RESULT>
</SHEET>

<!--각 행단위 트랜잭션 결과의 경우 -->
<?xml version='1.0' ?>
<SHEET>
  <!--선택항목. ETC-DATA 요소 -->
  <ETC-DATA>
    <ETC KEY="key_name">key_value</ETC>
  </ETC-DATA>
  <!--필수항목. RESULT 요소 -->
  <RESULT>
    <TR>
      <TD>OK</TD>
      <TD>DUP</TD>
      <TD>MISS</TD>
      <TD>NO</TD>
    </TR>
  </RESULT>
```



</SHEET>

위의 저장 XML에서 사용하는 XML요소는 다음과 같이 필수 항목과 선택 항목으로 나누어 진다.

요소명	필수여부	설 명
SHEET	필수	저장 데이터의 최상위 요소
RESULT	필수	저장 데이터의 결과 요소
ETC-DATA	선택	기타 데이터 설정 요소

6.1.2.1 SHEET 요소

저장 데이터의 최상위를 의미 한다.

이 요소는 저장 XML의 최 상위 요소로 XML 기본 정의 아래 반드시 하나만 사용해야 한다. 이 요소 안에 사용되는 <RESULT> 요소는 저장 결과 데이터를 의미하며 단 하나만 사용해야 한다. <ETC-DATA> 요소는 조회된 데이터 이외의 기타 데이터를 가져오는 요소로 여러 개 사용 할 수 있다.

➤ 속성

속성	설 명
	없음

➤ Example

```

<!--저장 성공의 경우 -->
<?xml version="1.0" ?>
<SHEET>
  <RESULT CODE="0" MESSAGE="정상적으로 처리되었습니다.">
</SHEET>

<!--저장 실패의 경우 -->
<?xml version="1.0" ?>
<SHEET>
  <RESULT CODE="-1" MESSAGE="저장에 실패하였습니다.">
</SHEET>

```

➤ 제공 버전

7.0.0.0



6.1.2.2 RESULT 요소

저장 데이터의 결과를 의미 한다.

CODE 속성은 저장 처리 결과에 대한 코드이며 정수값을 설정 한다.

설정 값이 0 이상인 경우 정상 처리로 모든 트랜잭션 상태의 행을 업데이트 처리 한다. 설정 값이 0보다 작은 경우는 실패 처리로 결과에 대한 업데이트는 처리 하지 않고 현재 트랜잭션 상태를 그대로 유지한다.

MESSAGE 속성은 처리 결과에 대한 메시지를 설정한다.

이 속성을 설정 한 경우 onSaveEnd 이벤트의 인자로 전달된다.

➤ 속성

속성	설 명
CODE	처리 결과 코드
MESSAGE	처리 결과 메시지

➤ Example

```
<!--저장 성공의 경우 -->
<?xml version="1.0" ?>
<SHEET>
  <RESULT CODE="0" MESSAGE="정상적으로 처리되었습니다.">
</SHEET>

<!--저장 실패의 경우 -->
<?xml version="1.0" ?>
<SHEET>
  <RESULT CODE="-1" MESSAGE="저장에 실패하였습니다.">
</SHEET>
```

➤ 제공 버전

7.0.0.0

6.1.2.3 TR 요소

저장 완료된 데이터의 트랜잭션 결과를 의미한다.

이 요소는 RESULT 요소 안에 사용되며, 각 행단위 트랜잭션 결과를 묶어 놓은 요소로 이 요소 안에 사용되는 TD 요소가 각 행단위의 결과를 의미한다.

➤ 속성

속성	설 명
	없음

➤ Example

```
<?xml version='1.0' ?>
<SHEET>
  <RESULT>
    <TR>
      <TD>OK</TD>
      <TD>DUP</TD>
      <TD>MISS</TD>
      <TD>NO</TD>
    </TR>
  </RESULT>
</SHEET>
```

➤ 제공 버전

7.0.0.0



6.1.2.4 TD 요소

각 행의 트랜잭션 결과를 의미한다.

이 요소는 TR 요소안에 사용되며 각 행의 트랜잭션 결과를 의미한다. 따라서 트랜잭션 개수가 3개라면 이 요소도 3번 반복 사용해야 한다.

트랜잭션 결과는 다음과 같이 처리된다.

결과	사용예	처리결과		
		상태	상태값 변경	Result 컬럼 표시값
OK	<TD> OK </TD>	"입력"일 때	조회상태로 변경	"입력완료" 표시
		"수정"일 때	조회상태로 변경	"수정완료" 표시
		"삭제"일 때	행이 사라짐	없음
		그외 일 때	조회상태로 변경	"저장완료" 표시
DUP	<TD> DUP </TD>	무관	변경 없음	"저장중복" 표시
MISS	<TD> MISS </TD>	무관	변경 없음	"미처리" 표시
NO	<TD> NO </TD>	무관	변경 없음	"저장오류" 표시

※ Result 컬럼 표시 값은 ibmsg 에서 변경하여 사용할 수 있다.

➤ 속성

속성	설 명
	없음

➤ Example

```
<?xml version='1.0' ?>
<SHEET>
  <RESULT>
    <TR>
      <TD>OK</TD>
      <TD>DUP</TD>
      <TD>MISS</TD>
```




<pre><TD>NO</TD> </TR> </RESULT> </SHEET></pre>

➤ **제공 버전**

7.0.0.0



6.1.2.5 ETC-DATA 요소

6.1.1.5 ETC-DATA 요소 참조

6.2 JSON

6.2.1 조회 JSON

조회 함수를 통해 읽어들이는 JSON 전체 구조는 다음과 같다.

```
{
  // 선택항목. TOTAL 요소
  Total:0,
  // 선택항목. ETC 요소
  Etc {"key_name":"key_value", "key_name2":"key_value2"},
  // 선택항목 MESSAGE 요소
  Message : "메시지",
  // 필수항목. DATA 요소
  Data:[
    // 필수항목. 데이터 단위 설정
    {C1:"데이터1", C2: "데이터2", ...}
  ],
  // 선택항목 RESULT 요소
  Result : {Code:0, Message:"정상적으로 처리되었습니다."}
}
```

json 생성시 요소명은 대소문자를 정확하게 지정해 주어야 한다. (매우 중요)

위의 조회 JSON에서 사용하는 JSON 요소는 다음과 같이 필수 항목과 선택 항목으로 나누어 진다.

요소명	필수여부	설 명
Data	필수	조회 데이터의 시작 요소
Total	선택	전체 데이터 개수 설정 요소
Etc	선택	기타 데이터 설정 요소
Message	선택	서버 메시지 설정 요소
Result	선택	서버 처리 결과 설정 요소

6.2.1.1 DATA 요소

조회된 데이터 영역을 의미하는 요소로 이 요소 안의 데이터를 조회된 데이터로 표현한다.

이 요소의 Value 값은 배열로 구성되며 하나의 배열 요소가 행 단위 데이터를 의미 한다.

행 단위 데이터는 Key:Value 형태의 구조로 행의 컬럼 개수만큼 반복되어 구성된다. 컬럼의 Key 는 컬럼 순차적으로 조회 하는 경우는 C1, C2, ... 로 Key를 구성하며 Key 를 SaveName 으로 구성하는 경우는 InitColumns Method 에서 컬럼 초기화 구성시 설정한 SaveName 과 매핑하여 조회 처리 된다.

BackColor, FontColor, Edit, Image, ComboText, ComboCode Attribute를 사용하고자 하는 경우 ibsheet.cfg 에서 UseJsonAttribute = "1" 로 설정 하여야 한다.

내부적으로 사용하는 키값은 다음과 같다.

Key	설 명
BackColor	해당 행의 배경색상
Drag	해당행의 Drag 허용 여부 (Default = 1)
Edit	해당 행의 Edit 허용 여부
Expand	해당 행의 자식 레벨 데이터를 펼칠것인지 여부
FontColor	해당 행의 폰트색상
HaveChild	트리 형태일 때 해당 행의 Child 데이터가 있으면서 조회하지 않는 경우 사용, 나중에 OnTreeChild Event가 발생할 수 있는 행 설정
Hidden	해당 행의 숨김 여부
Level	트리 형태일 때 행의 트리레벨 설정 (UseJsonTreeLevel 속성을 사용하는 경우)
Merge	해당 행의 RowMerge 허용 여부
Sum	해당 행의 합계 계산 허용 여부

셀별 속성 설정은 해당 키값 과 속성명을 "#"으로 구분하여 사용하며 사용 가능한 속성은 다음과 같다 (예, SaveName#BackColor)

속성명	설 명
Align	정렬 설정
BackColor	해당 셀의 배경색상
ComboCode	콤보타입일 경우 해당셀의 콤보 코드값을 설정
ComboText	콤보타입일 경우 해당셀의 콤보 텍스트값을 설정
Edit	해당 셀의 Edit 허용 여부
FontBold	Bold 설정 여부
FontColor	해당 셀의 폰트색상
FontItalic	기울이기 설정 여부
FontStrike	취소선 설정 여부
FontUnserline	밑줄 설정 여부
Image	해당 셀의 이미지 경로 (ImageList 인덱스 사용가능)
ToolTip	표시할 툴팁도움말 문자열
TreeCheck	트리 컬럼의 체크박스 값

➤ Example

```
// 일반 순차 조회의 경우
{
    Data:[
        {BackColor:"#FF0000", FontColor:"#FFFF00", Edit:"0", C1: "", C2: "006", C3:
"홍길동", C4: "10000", C5: "20030201"},
        {C1:"",C2:"006",C3:"홍길동", "C3#BackColor":"#FF0000",
"C3#FontColor":"#00FF00", "C3#Edit":"0", C4: "10000", C5: "20030201"},
        ...
    ]
}

// SaveName 매핑 조회
{
```



```
Data:[
  { BackColor:"#FF0000", FontColor:"#FFFF00", Edit:"0", "sStatus": "", "sCode":
"006", "sName": "홍길동", "sRank": "10000", "sDate": "20030201"},
    {"sStatus": "", "sCode": "006", "sName": "홍길동",
"sName#BackColor":"#FF0000", "sName#FontColor":"#00FF00",
"sName#Edit":"0", "sRank": "10000", "sDate": "20030201"},
    ...
  ]
}
```

➤ **제공 버전**

7.0.0.0

6.2.1.2 TOTAL 요소

실시간 서버 조회 처리시 사용되는 요소로 전체 데이터의 총 개수를 의미 한다.

이 요소의 Value 값은 정수값으로 설정하며, 실시간 서버 조회 처리시 이 요소는 반드시 설정 하여야 하며, 설정하지 않거나 0으로 설정한 경우는 조회 데이터가 없는 것으로 처리 된다.

➤ Example

```
// 전체 데이터가 1000건인 실시간 서버 조회의 경우
{
  Total:1000
  Data:[
    {C1: "", C2: "006", C3: "홍길동", C4: "10000", C5: "20030201"},
    {C1: "", C2: "006", C3: "홍길동", C4: "10000", C5: "20030201"},
    ...
  ]
}
```

➤ 제공 버전

7.0.0.0



6.2.1.3 ETC 요소

조회 또는 저장 데이터 이외의 여분의 기타 데이터를 의미 한다.

이 요소의 Value 값은 Key:Value 형태의 반복으로 구성되며 IB Sheet 내의 데이터 이외의 데이터를 가져와 다른 용도로 사용하는 경우에 설정하여 사용한다. 이 요소를 통해 가져온 데이터는 EtcData Method를 통해 확인 할 수 있고 다른 값으로도 변경이 가능 하다.

➤ Example

```
// 조회 시 사용하기
{
  Etc: {"name":"홍길동", "age":"30"},
  Data:[
    (조회된 데이터 중략)
  ]
}

// 저장 시 사용하기
{
  Etc: {"name":"홍길동", "age":"30"},
  Result : {Code:0, Message:"결과 메시지."}
}
```

➤ 제공 버전

7.0.0.0



6.2.1.4 MESSAGE 요소

서버로부터 조회 처리 중 발생하는 메시지를 처리하기 위한 요소이다.

이 요소가 설정된 경우 조회 완료 후 발생하는 OnSearchEnd 이벤트에서 message 인자에 값이 설정되어 이벤트가 발생한다.

➤ Example

```
// 조회 시 사용하기
{
  Message: "조회가 완료되었습니다.",
  Data:[
    (조회된 데이터 중략)
  ]
}
```

➤ 제공 버전

7.0.0.0



6.2.1.5 RESULT 요소

조회 데이터의 결과를 의미 한다.

CODE 속성은 조회 처리 결과에 대한 코드이며 정수값을 설정 한다.

설정 값이 0 이상인 경우 정상 처리로 0보다 작은 경우는 실패 처리한다..

MESSAGE 속성은 처리 결과에 대한 메시지를 설정한다.

<MESSAGE> 요소와 같이 사용하는 경우는 <MESSAGE> 요소가 우선적으로 처리 되어 <RESULT> 요소에서 설정한 MESSAGE 속성은 무시 된다.

➤ 속성

속성	설 명
Code	처리 결과 코드
Message	처리 결과 메시지

➤ Example

```
// 조회 시 사용하기
{
  Data:[
    (조회된 데이터 중략)
  ],
  Result:{Code:0, Message:"조회가 완료되었습니다."}
}
</SHEET>
```

➤ 제공 버전

7.0.0.0

6.2.2 저장 JSON

저장 함수를 통해 읽어들이는 JSON 전체 구조는 다음과 같다.

```
{
  // 선택항목. ETC 요소
  Etc: {"key_name":"key_value", "key_name2":"key_value2"},
  // 필수항목. DATA 요소
  Result:{Code: "0", Message: "완료 후 메시지", Result: "행별 트랜잭션 결과"}
}
```

위의 조회 JSON에서 사용하는 JSON 요소는 다음과 같이 필수 항목과 선택 항목으로 나누어 진다.

요소명	필수여부	설 명
RESULT	필수	저장 데이터의 결과 요소
ETC-DATA	선택	기타 데이터 설정 요소



6.2.2.1 RESULT 요소

저장 데이터의 결과를 의미 한다.

이 요소의 Value 값은 Key:Value 형태로 Key 값으로는 Code, Message, Result 를 설정 할 수 있다.

Code Key는 저장 처리 결과에 대한 코드이며 정수값을 설정 한다.

설정 값이 0 이상인 경우 정상 처리로 모든 트랜잭션 상태의 행을 업데이트 처리 한다. 설정 값이 0보다 작은 경우는 실패 처리로 결과에 대한 업데이트는 처리 하지 않고 현재 트랜잭션 상태를 그대로 유지한다.

Message Key는 처리 결과에 대한 메시지를 설정한다.

이 속성을 설정 한 경우 OnSaveEnd 이벤트의 인자로 전달된다.

Result Key는 각 행단위 트랜잭션 결과를 처리하는 경우에 설정한다.

각 행단위 트랜잭션 결과를 "|"로 연결하여 설정한다.

이 속성은 행 단위 트랜잭션을 처리하는 경우에만 사용한다.

트랜잭션 결과는 다음과 같이 처리된다.

결과	사용예	처리결과		
		상태	상태값 변경	Result 컬럼 표시값
OK	OK	"입력"일 때	조회상태로 변경	"입력완료" 표시
		"수정"일 때	조회상태로 변경	"수정완료" 표시
		"삭제"일 때	행이 사라짐	없음
		그외 일 때	조회상태로 변경	"저장완료" 표시
DUP	DUP	무관	변경 없음	"저장중복" 표시
MISS	MISS	무관	변경 없음	"미처리" 표시
NO	NO	무관	변경 없음	"저장오류" 표시

※ Result 컬럼 표시 값은 ibmsg 에서 변경하여 사용할 수 있다.



➤ **Example**

```
// 전체데이터 처리 결과 : 저장 성공의 경우
{
  Result:{ Code:0, Message: "정상적으로 처리되었습니다." }
}

// 전제데이터 처리 결과 : 저장 실패의 경우
{
  Result:{ Code:-1, Message: "저장에 실패하였습니다." }
}

// 각 행단위 트랜잭션 처리 결과의 경우
{
  Result:{ Result: "OK|DUP|MISS|NO" }
}
```

➤ **제공 버전**

7.0.0.0



6.2.2.2 ETC 요소

6.2.1.3 ETC 요소 참조



6.2.1.4 MESSAGE 요소

서버로부터 조회 처리 중 발생하는 메시지를 처리하기 위한 요소이다.

이 요소가 설정된 경우 조회 완료 후 발생하는 OnSearchEnd 이벤트에서 message 인자에 값이 설정되어 이벤트가 발생한다.

➤ Example

```
// 조회 시 사용하기
{
  Message: "조회가 완료되었습니다.",
  Data:[
    (조회된 데이터 중략)
  ]
}
```

➤ 제공 버전

7.0.0.0

6.3 Tree 기능 사용시 JSON

특정 컬럼을 Tree 형태로 사용시 데이터 JSON의 기본 구조는 다음과 같다.

➤ Example

```
{data:[  
  
  {sa_nm:"김사장",job:" PRESIDENT",Items:[           //0 LEVEL  
  
    {sa_nm:"안전무",job:"GMANAGER"},                //1 LEVEL  
  
    {sa_nm:"이상무",job:"GMANAGER",Items:[           //1LEVEL  
  
      {sa_nm:"박대리",job:"ASSISTANT"}                //2LEVEL  
  
    ]}  
  
  ]}  
  
]}
```

단 위와 같은 json 구조를 구성하기 어려울 경우, xml와 같이 Level 프로퍼티를 이용한 방법도 기능으로 제공하고 있다.

Level 인자를 이용한 방법

먼저 ibsheet.cfg 파일에 UseJsonTreeLevel = "1"로 설정

➤ Example

```
{data:[  
  
  {sa_nm:"김사장",job:" PRESIDENT",Level:0},          //0 LEVEL
```




```
{sa_nm:"안전무",job:"GMANAGER",Level:1},    //1 LEVEL
{sa_nm:"이상무",job:"GMANAGER",Level:1},    //1LEVEL
{sa_nm:"박대리",job:"ASSISTANT",Level:2}      //2LEVEL
}}
```

➤ 제공 버전

7.0.0.0



6.4 엑셀리포트 XML

엑셀 다운로드 함수(Down2Excel Method, DirectDown2Excel Method)를 통해 엑셀로 내릴 때 엑셀의 요지나 머리글 바닥글과 같이 엑셀에 대한 속성을 설정한다.

```
<?xml version='1.0' ?>
<Excel>
  <!-- IBSheetSet 요소 -->
  <IBSheetSet>
    <StartRow>6</StartRow>
    <ViewCols>1|2|3|4</ViewCols>
    <Data>조회한 데이터</Data>
  </IBSheetSet>

  <!-- PageSet 요소 -->
  <PageSet>
    <PaperSize>A4</PaperSize>
    <Orientation>Portrait</Orientation>
    <CenterOnPage Horizontal="True" Vertical="False"/>
    <RowsRepeat Row1="1" Row2="1" />
    <ColsRepeat Col1="1" Col2="1" />
    <DefaultRowHeight>18</DefaultRowHeight>
    (중략)
  </PageSet>

  <!-- ColumnWidth 요소 -->
  <ColumnWidth Col1="1" Col2="End">20</ColumnWidth>
  <!-- RowHeight 요소 -->
  <RowHeight Row1="5" Row2="6">30</RowHeight>

  <!-- CustomHeader 요소 -->
  <CustomHeader>
```



```

    <LeftSection>(중략)</LeftSection>
    <CenterSection>(중략)</CenterSection>
    <RightSection>(중략)</RightSection>
</CustomHeader>
<!-- CustomFooter 요소 -->
<CustomFooter>
    <LeftSection>(중략)</LeftSection>
    <CenterSection>(중략)</CenterSection>
    <RightSection>(중략)</RightSection>
</CustomFooter>

<!-- Label 요소 -->
<Label>
    <Range Row1="5" Col1="1" Row2="4" Col2="10"> (중략) </Range>
</Label>

<!-- DataPattern 요소 -->
<DataPattern>
    <Standard SheetCol="0" Word="소계*">(중략)
        <Range Row1="5" Col1="1" Row2="4" Col2="10">(중략)</Range>
    </Standard>
</DataPattern>

<!-- BorderSet 요소 -->
<BorderSet>
    <Range Row1="5" Col1="1" Row2="4" Col2="10" >(중략)</Range>
</BorderSet>
</Excel>

```

위의 엑셀리포트XML에서 사용하는 XML 요소는 다음과 같다.

요소명	설명
IBSheetSet 요소	엑셀에 내릴 IBSheet 부분을 설정, 이부분은 필수이고, 반



	드시 맨 처음 설정해야 함
PageSet 요소	엑셀의 페이지 설정
ColumnWidth 요소	엑셀의 컬럼 너비 설정
RowHeight 요소	엑셀의 행의 높이 설정
CustomHeader 요소	엑셀의 용지 설정의 머리글 설정
CustomFooter 요소	엑셀의 용지 설정의 바닥글 설정
Label 요소	엑셀의 셀에 표시할 문자열 설정
DataPattern 요소	엑셀에서 기준 글자를 찾아 일정한 패턴을 설정
BorderSet 요소	엑셀의 영역별 테두리 설정

6.4.1 IBSheetSet 요소

엑셀파일에 설정될 IBSheet 부분을 의미한다.

IBSheetSet 요소 하단에 사용하는 StartRow 요소는 Sheet의 데이터를 엑셀로 내릴 때 엑셀의 시작 행의 위치를 설정하며, 엑셀은 Sheet와 달리 첫 행을 1부터 시작한다.

IBSheetSet 요소 하단에 사용하는 ViewCols 요소는 Sheet위의 컬럼 중 다운받고자 하는 컬럼을 지정해야 하는 경우 컬럼번호를 "|" 문자열로 조합하여 설정한다. ViewCols 요소를 생략한다면 기본적으로 호출된 함수의 인자 설정에 따라 다운받을 컬럼이 결정된다.

IBSheetSet 요소는 Label 요소를 사용할 경우 반드시 설정해야 하고, Excel 요소 바로 아래 최상단에 설정한다.

➤ 하위요소

속성	설 명
<StartRow>	Sheet의 데이터를 표시할 엑셀의 시작 행 위치
<ViewCols>	Sheet에서 내릴 컬럼 번호, 미설정시 함수 인자 설정에 따름

➤ Example

```
<?xml version="1.0" ?>
<Excel>
  <IBSheetSet>
    <StartRow>6</StartRow>
    <ViewCols>3|5|7</ViewCols>
  </IBSheetSet>
</Excel>
```

➤ 제공 버전

7.0.0.0

6.4.2 PageSet 요소

엑셀 파일에 페이지 설정 부분을 의미한다.

PageSet 요소는 엑셀 파일의 메뉴에서 "파일 > 페이지설정" 부분과 거의 흡사하다. 페이지에 대해서 설정해야 하는 각종 속성을 설정한다.

➤ 하위요소

속성	설 명
<PaperSize>	엑셀의 용지 크기 설정
<Orientation>	엑셀의 용지 방향 설정
<Margins>	엑셀의 여백 설정
<Scaling>	엑셀의 용지 배율 설정
<CenterOnPage>	엑셀의 페이지의 가운데에 대한 설정
<RowsRepeat>	엑셀의 반복 행 설정
<ColsRepeat>	엑셀의 반복 컬럼 설정
<DefaultRowHeight>	엑셀의 기본 행 높이 설정
<DefaultFont>	엑셀의 기본 글꼴 설정

PageSet 요소 하단의 PaperSize 요소는 페이지의 용지 크기를 설정하는 것으로 "A3, A4, B4, B5" 중 하나를 선택하여 설정하도록 한다. 기본은 "A4"이다.

PageSet 요소 하단의 Orientation 요소는 용지의 방향을 설정하는 것으로 세로인 경우 "Portrait"로 설정하고, 가로인 경우 "Landscape"로 설정한다. 기본은 세로이다.

PageSet 요소 하단의 Margins 요소는 용지의 여백을 설정하는 것으로 요소의 속성으로 여백을 설정한다. 기본은 엑셀의 기본 설정에 따라 설정된다.

PageSet 요소 하단의 Scaling 요소는 용지의 배율을 설정하는 것으로 엑셀의 "축소/확대 배율"을 사용하는 경우와 "자동맞춤"을 사용하는 경우에 따라 "True", "False"로 설정한다. "자동맞춤"을 사용하는 경우 "용지너비"와 "용지높이"를 엑셀에서 설

정할수 있는 값으로 설정한다. 기본은 "축소/확대 배율"을 사용하는 것이다.

PageSet 요소 하단의 CenterOnPage 요소는 페이지의 가운데에 대한 설정하는 것으로 가로로 가운데로 설정시 Horizontal 속성을 "true"로 설정하고, 세로로 가운데로 설정 시 Vertical 속성을 "true"로 설정한다. 기본은 둘 다 "false"이다.

PageSet 요소 하단의 RowsRepeat 요소는 반복 행에 대한 설정하는 것으로 시작 행을 Row1 속성에 마지막 행은 Row2 속성에 설정한다. 만약 1개 행만 반복하고자 하는 경우 Row2는 설정하지 않아도 된다. 기본은 반복 행이 없다.

PageSet 요소 하단의 ColsRepeat 요소는 반복 컬럼에 대한 설정하는 것으로 시작 컬럼을 Col1 속성에 마지막 컬럼은 Col2 속성에 설정한다. 만약 1개 컬럼만 반복하고자 하는 경우 Col2는 설정하지 않아도 된다. 기본은 반복 컬럼이 없다.

PageSet 요소 하단의 DefaultRowHeight 요소는 엑셀 각 행의 기본 높이를 설정하는 것으로 기본은 13.5이다.

PageSet 요소 하단의 DefaultFont 요소는 페이지의 기본 글꼴을 설정한다.

Name 속성은 글꼴, Bold 속성은 굵기, Italic 속성은 기울임 여부, size 속성은 글자 크기, UnderLine은 밑줄로 "None | Single | Double"중 선택한다. Strikethrough 속성은 취소선 사용 여부, Superscript 속성은 위첨자 사용 여부, Superscript 속성은 아래첨자 사용 여부, Color 글자색상을 설정한다.

➤ Example

```
<?xml version="1.0" ?>
<Excel>
  <PageSet>
    <PaperSize>A4</PaperSize>
    <Orientation>Landscape</Orientation>
    <Margins Top="1.5" Header="1.3" Left="1" Right="1" Bottom="1.5"
Footer="1.3" >True</Margins>
```



```
<Scaling>
  <Adjust Value="100">true</Adjust>
  <Fit Height="1" Width="1">false</Fit>
</Scaling>
<RowsRepeat Row1="6" Row2="6">True</RowsRepeat>
<ColsRepeat Col1="1" Col2="1">False</ColsRepeat>
<DefaultRowHeight>17</DefaultRowHeight>
<DefaultFont Name="돋움체" size="9"/>
</PageSet>
</Excel>
```

➤ **제공 버전**
7.0.0.0

6.4.3 ColumnWidth 요소

엑셀의 각 컬럼에 너비를 설정한다.

이 요소는 1개 컬럼 너비 뿐만 아니라 , 여러개 컬럼의 너비를 영역으로 설정할수 있다. 이 요소의 Col1 속성은 영역으로 컬럼너비를 설정할때 시작 컬럼의 번호이고, Col2 속성은 마지막 컬럼의 번호이다.

컬럼의 너비는 -1로 설정할 경우 엑셀이 모두 내려진 후 해당 컬럼의 가장 넓은 글자의 너비에 맞춰 자동으로 조정되는 설정이고, 일반적으로 엑셀에서 설정하는 컬럼 너비를 숫자로 설정한다.

➤ 속성

속성	설 명
Col1	엑셀의 시작 컬럼 번호(엑셀은 1부터 시작)
Col2	엑셀의 마지막 컬럼 번호, 마지막은 "End"로 설정 가능

➤ Example

```
<?xml version="1.0" ?>
<Excel>
  <ColumnWidth Col1="1" Col2="End"> -1 </ColumnWidth>
  <ColumnWidth Col1="8">10</ColumnWidth>
</Excel>
```

➤ 제공 버전

7.0.0.0



6.4.4 RowHeight 요소

엑셀의 각 행에 높이를 설정한다.

이 요소는 1개 행 높이 뿐만 아니라, 여러개 행의 높이를 영역으로 설정할수 있다. 이 요소의 Row1 속성은 영역으로 행높이를 설정할때 시작 행의 번호이고, Row2 속성은 마지막 행의 번호이다.

➤ 속성

속성	설 명
Row1	엑셀의 시작 행 번호(엑셀은 1부터 시작)
Row2	엑셀의 마지막 행 번호, 마지막은 "End"로 설정 가능

➤ Example

```
<?xml version="1.0" ?>
<Excel>
  <RowHeight Row1="2" Row2="5">15</RowHeight>
  <RowHeight Row1="6">5</RowHeight>
</Excel>
```

➤ 제공 버전

7.0.0.0

6.4.5 CustomHeader 요소

엑셀의 페이지 설정의 머리글을 설정한다.

엑셀 메뉴에서 "파일 > 페이지설정"의 "머리글/바닥글" 설정 탭 중 "머리글 부분을 설정하는것이 이 요소이다.

➤ 하위요소

속성	설 명
<LeftSection>	왼쪽 구역 설정
<CenterSection>	가운데 구역 설정
<RightSection>	오른쪽 구역 설정

➤ Example

```
<?xml version="1.0" ?>
<Excel>
  <!--머리글 설정하기-->
  <CustomHeader>
    <LeftSection>
      <font Name="돋움체" Size="12"/>
      <InputText><![CDATA[IB Leaders]]> </InputText>
    </LeftSection>
    <RightSection>
      <font Name="돋움체" Size="12"/>
      <InputText><![CDATA[IBSheet 엑셀 리포트]]> </InputText>
    </RightSection>
  </CustomHeader>
</Excel>
```

➤ 제공 버전

7.0.0.0

6.4.5.1 LeftSection, CenterSection, RightSection 요소

엑셀의 페이지의 머리글과 바닥글의 각 섹션을 설정한다.

이 요소들은 머리글 또는 바닥글의 각 영역에 표시할 글자나 이미지를 설정한다.

➤ 하위요소

속성	설 명
	글꼴 설정
<InputText>	해당 영역에 표시할 글자 설정

이 요소 하단에 Font 요소는 해당 영역의 글꼴을 설정한다. Font의 속성의 Name 속성은 글꼴, Bold 속성은 굵기, Italic 속성은 기울임 여부, size 속성은 글자 크기, UnderLine은 밑줄로 "None | Single | Double"중 선택한다.

이 요소 하단에 InputText 요소는 해당 영역의 표시할 글자를 설정한다. InputText 요소의 PCDATA 영역에 표시하고 하는 글자를 다음 예약어와 함께 사용할 수 있다.

&P	&N	&D	&T
페이지번호	전체페이지수	날짜	시간
&F	&A	문자열	
파일명	워크시트명	표시문자열	

➤ Example

```
<?xml version="1.0" ?>
<Excel>
  <!--머리글 설정하기-->
  <CustomHeader>
    <LeftSection>
      <font Name="돋움체" Size="12"/>
      <InputText><![CDATA[&D]]> </InputText>
    </LeftSection>
```



```
<RightSection>  
  <font Name="돋움체" Size="12"/>  
  <InputText><![CDATA[IBSheet 엑셀 리포트]]></InputText>  
</RightSection>  
</CustomHeader>  
</Excel>
```

➤ **제공 버전**
7.0.0.0



6.4.6 CustomFooter 요소

엑셀의 페이지 설정의 바닥글을 설정한다.

엑셀 메뉴에서 "파일 > 페이지설정"의 "머리글/바닥글" 설정 탭 중 "바닥글 부분을 설정하는것이 이 요소이다.

➤ 하위요소

속성	설 명
<LeftSection>	왼쪽 구역 설정
<CenterSection>	가운데 구역 설정
<RightSection>	오른쪽 구역 설정

요소의 상세설명은 CustomHeader의 하단 요소 설명을 참조한다.

➤ Example

```
<?xml version="1.0" ?>
<Excel>
  <!--바닥글 설정하기-->
  <CustomFooter>
    <CenterSection>
      <font Name="돋움체" Size="12"/>
      <InputText><![CDATA[&P / &N]]></InputText>
    </CenterSection>
  </CustomFooter>
</Excel>
```

➤ 제공 버전

7.0.0.0

6.4.7 Label 요소

엑셀의 셀에 원하는 글자를 설정한다.

엑셀의 워크시트에 셀이라는 영역에는 다양한 글자를 설정할 수 있다. Sheet에서 자동으로 내리는 데이터는 "Excel 요소 > IBSheetSet 요소"에 의해서 가능하지만 그 외의 엑셀의 특정 영역(Range)에 글자를 쓰기 위해서는 이 요소를 사용한다.

➤ 하위요소

속성	설 명
<Range>	Label을 설정하는 엑셀의 셀 영역에 대한 요소

➤ Example

```
<Label>
  <Range Row1="1" Col1="1" Row2="1" Col2="End" RowHeight="40">
    <CellFormat>
      <Merge>true</Merge>
      <Alignment Horizontal="Center" Vertical="Center"/>
      <Interior      BackColor="RGB(255,255,133)"      Pattern="xlGray50"
PatternColor="RGB(192,192,192)"/>
      <Font Name="굴림" Bold="True" Size="15" />
    </CellFormat>
    <InputText> <![CDATA[모델별 용량 금액 리스트]]> </InputText>
    <BorderStyle>
      <TopEdge Style="Continuous" Weight="Medium"/>
      <BottomEdge Style="Continuous" Weight="Medium"/>
    </BorderStyle>
  </Range>
</Label>
```

➤ 제공 버전

7.0.0.0

6.4.7.1 Range 요소(Label 요소의 하위)

Label 요소로 엑셀의 워크시트의 셀에 원하는 글자를 설정할때 셀 영역을 의미한다.

이 요소의 속성은 Row1, Row2는 엑셀의 행번호를 의미하며 엑셀은 행을 1부터 시작하므로 Sheet의 시작행과 동일하게 처리하면 안된다. Row1, Row2 인자를 행번호를 마지막 행번호 사용할 경우 "end"라는 글자로 설정할 수 있다.

Col1, Col2 인자도 동일하게 1컬럼부터 시작하며, 마지막 컬럼을 사용할 경우 "end"라는 글자를 설정할수 있다.

Label을 표시할 엑셀의 영역이 1개인 경우 Row1과 Col1 속성만 사용하고, Row2와 Col2 속성은 제외하고 사용할 수 있다.

➤ 속성

속성	설 명
Row1	Label을 설정할 엑셀 영역의 시작 행번호 또는 "End"
Col1	Label을 설정할 엑셀 영역의 시작 컬럼번호 또는 "End"
Row2	Label을 설정할 엑셀 영역의 마지막 행번호 또는 "End"
Col2	Label을 설정할 엑셀 영역의 마지막 컬럼번호 또는 "End"
RowHeights	엑셀 영역의 행높이, 픽셀단위

➤ 하위요소

속성	설 명
<CellFormat>	엑셀 영역의 머지, 정렬, 색상이나 글꼴 등을 설정하는 요소
<InputText>	엑셀 영역에 표시할 글자를 설정하는 요소
<BorderStyle>	엑셀 영역에서 테두리를 설정하는 요소

➤ Example

```
<?xml version="1.0" ?>
```




```
<Excel>
  <Label>
    <Range Row1="3" Col1="end" Row2="3" Col2="end" RowHeight="15">
      <CellFormat>
        <Alignment Horizontal="Right" Vertical="Center"/>
        <Font Name="굴림" Bold="True" Size="9" />
      </CellFormat>
      <InputText> <![CDATA[단위 [원화:만원]]> </InputText>
    </Range>
  </Label>
</Excel>
```

➤ 제공 버전

7.0.0.0

6.4.7.1.1 CellFormat 요소

엑셀 영역의 머지나 정렬, 색상, 글꼴 등을 설정한다.

이 요소는 부모인 Range 요소에서 선택한 엑셀의 셀 영역에 대한 각종 속성을 설정한다.

➤ 하위요소

요소	설 명
<Merge>	엑셀의 셀 영역을 머지 할지 여부 설정
<Alignment>	엑셀의 셀 영역의 정렬 설정
<Interior>	엑셀의 셀 영역의 배경 색 및 패턴 설정
	엑셀의 셀 영역의 글꼴 설정

이 요소 하단의 Merge 요소는 엑셀의 셀 영역을 머지할것인가 여부이며, 영역의 머지는 아래와 같이 "TRUE" 또는 "FALSE"로 설정한다. 기본을 "FALSE"이다

이 요소 하단의 Alignment 요소는 엑셀의 셀 영역의 가로 정렬과 세로 정렬을 설정한다. 가로 정렬은 "Horizontal" 이고, 설정할 수 있는 값은 "Left", "Center", "Right"이다. 세로 정렬은 "Vertical"이고, 설정할 수 있는 값은 "Top", "Center", "Bottom" 이다.

이 요소 하단의 Interior 요소는 엑셀의 셀 영역의 색상이나 패턴을 설정한다. 엑셀에서 셀서식의 "무늬" 설정과 동일하며, 설정할수 있는 속성은 다음과 같다.

➤ Interior 속성

속성	설 명
BackColor	영역의 배경색을 설정한다.
PatternColor	영역의 패턴색을 설정한다.
Pattern	영역의 패턴을 설정한다. 패턴종류는 (xlSolid, xlGray75, xlGray50, xlGray25, xlGray16, xlGray8, xlHorizontal, xlVertical,

	xlDown, xlUp, xlChecker, xlSemiGray75, xlLightHorizontal, xlLightVertical, xlLightDown, xlLightUp, xlGrid, xlCrissCross)
--	--

이 요소 하단에 Font 요소는 해당 영역의 글꼴을 설정한다. Font의 속성의 Name 속성은 글꼴, Bold 속성은 굵기, Italic 속성은 기울임 여부, size 속성은 글자 크기, UnderLine은 밑줄로 "None | Single | Double"중 선택한다.

➤ Example

```
<?xml version="1.0" ?>
<Excel>
  <Label>
    <Range Row1="3" Col1="end" Row2="3" Col2="end" RowHeight="15">
      <CellFormat>
        <Alignment Horizontal="Right" Vertical="Center"/>
        <Font Name="굴림" Bold="True" Size="9" />
      </CellFormat>
      <InputText><![CDATA[단위 [원화:만원]]></InputText>
    </Range>
  </Label>
</Excel>
```

➤ 제공 버전

7.0.0.0



6.4.7.1.2 InputText 요소

엑셀 영역에 표시할 글자를 설정한다.

이 요소는 부모인 Range 요소에서 선택한 엑셀의 셀 영역에 글자를 표시한다.

이때 주의할 점은 표시하는 글자 중에 특수문자(&, <, >, ', ") 가 있는 경우 변환문자로 변환하거나 CDATA Section을 사용한다.

줄바꿈 글자를 설정할때 "wn"을 포함하여 설정한다.

➤ Example

```
<?xml version="1.0" ?>
<Excel>
  <Label>
    <Range Row1="3" Col1="end" Row2="3" Col2="end" RowHeight="15">
      <InputText><![CDATA[단위 [원화:만원]]></InputText>
    </Range>
  </Label>
</Excel>
```

➤ 제공 버전

7.0.0.0

6.4.7.1.3 BorderStyle 요소

엑셀 영역에 테두리 선을 설정한다.

이 요소는 Range 요소 하단에 사용되며, Range 요소에서 선택된 영역에 테두리를 설정한다. 이 요소 하단에 사용되는 각 요소인 TopEdge 요소, BottomEdge 요소, LeftEdge 요소, RightEdge 요소 등의 요소 들은 Style, Weight, Color 속성을 각각 갖는다.

➤ Style 속성

선없음	연속선	대쉬	대쉬+점
None	Continuous	Dash	DashDot
대쉬+점+점	점선	이중선	기울어진대쉬+점
DashDotDot	Dot	Double	SlantDashDot

➤ Weight 속성

매우가는선	가는선	중간굵은선	굵은선
Hairline	Thin	Medium	Thick

Color 속성은 원하는 색상을 RGB로 설정한다.

➤ Example

```
<?xml version="1.0" ?>
<Excel>
<Label>
  <Range Row1="3" Col1="1" Row2="3" Col2="end" RowHeight="15">
    <InputText>표시글자</InputText>
    <BorderStyle>
      <TopEdge Style="Dash" Weight="Medium" Color="RGB(255,0,0)" />
      <BottomEdge Style="Dash" Weight="Medium" Color="RGB(0,0,0)" />
      <LeftEdge Style="Continuous" Weight="Hairline" Color="RGB(0,0,255)" />
    </BorderStyle>
  </Range>
</Label>
</Excel>
```



```
<RightEdge Style="Continuous" Weight="Hairline" Color="RGB(0,0,0)"/>
</BorderStyle>
</Range>
</Label>
</Excel>
```

➤ 제공 버전

7.0.0.0

6.4.8 DataPattern 요소

Sheet에서 내려진 데이터의 기준 글자를 찾아 엑셀 영역에 일정 패턴을 설정한다.

Sheet에서 내리지는 데이터는 "Excel 요소 > IBSheetSet 요소"에 의해서 엑셀의 x 번째 행부터 내려진다.

이렇게 Sheet에서 내려진 데이터의 특정 컬럼에 해당하는 데이터가 기준에 맞을 때 이 요소에서 설정된 일정 패턴을 적용한다.

➤ 하위요소

속성	설 명
<Standard>	패턴을 설정할 데이터의 기준을 설정한다.

➤ Example

```
<?xml version="1.0" ?>
<Excel>
  <DataPattern>
    <Standard SheetCol="0" Word="소계*">
      <Range Col1="1" Col2="End" RowHeight="25">
        <CellFormat> (중략) </CellFormat>
        <BorderStyle> (중략) </BorderStyle>
      </Range>
    </Standard>
  </DataPattern>
</Excel>
```

➤ 제공 버전

7.0.0.0



6.4.8.1 Standard 요소

패턴을 설정하는 기준 글자의 기준을 설정한다.

Sheet에서 내려진 데이터에 일정한 패턴을 설정하는데, 이때 이 요소는 기준이 되는 정보를 설정한다.

SheetCol 속성은 엑셀에 내려지는 Sheet 데이터 중 기준을 설정할 컬럼 번호나 SaveName을 설정한다.

Word 속성은 SheetCol 속성에 설정된 컬럼에서 찾을 글자로 "*"을 이용하여 문자열을 like 비교할 수 있다.

➤ 속성

속성	설 명
SheetCol	Sheet에서 내려지는 데이터 컬럼 Index 또는 SaveName
Word	찾을 문자열, 찾을 글자 앞뒤로 "*" 을 사용 가능

➤ 하위요소

속성	설 명
< Range>	패턴을 설정할 데이터의 기준을 설정한다.

➤ Example

```
<?xml version="1.0" ?>
<Excel>
  <DataPattern>
    <Standard SheetCol="0" Word="소계*>
      <Range Col1="1" Col2="End" RowHeight="25">
        <CellFormat>
          <Merge>False</Merge>
          <Alignment Horizontal="Center" Vertical="Center"/>
        </CellFormat>
      </Range>
    </Standard>
  </DataPattern>
</Excel>
```




```
<Interior BackColor="RGB(0,255,0)" />  
<Font Name="굴림" Size="12" Bold="True" Italic="True"/>  
</CellFormat>  
</Range>  
</Standard>  
</DataPattern>  
</Excel>
```

➤ 제공 버전

7.0.0.0

6.4.7.1 Range 요소(DataPattern 요소의 하위)

엑셀의 셀에 원하는 글자를 설정할때 셀 영역을 의미한다.

이 요소는 데이터 패턴을 설정할 엑셀의 영역을 설정하기 위한 요소이다. 이 요소의 속성 Col1, Col2는 엑셀의 컬럼번호를 의미하며 엑셀은 컬럼을 1부터 시작하므로 Sheet의 시작행과 동일하게 처리하면 안된다.

Col1, Col2 인자의 행번호를 마지막 행번호 사용할 경우 "End"라는 글자로 설정할 수 있다.

RowHeight 속성은 영역의 행높이를 설정하는것으로 픽셀로 설정하고, 설정하지 않을 경우 기본적으로 "PageSet 요소 > DefaultRowHeight 요소"에서 설정하는 기본 행 높이로 사용된다.

➤ 속성

속성	설 명
Col1	엑셀 영역의 시작 컬럼번호 또는 "End"
Col2	엑셀 영역의 마지막 컬럼번호 또는 "End"
RowHeights	엑셀 영역의 행높이, 픽셀단위

➤ 하위요소

속성	설 명
<CellFormat>	엑셀 영역의 머리, 정렬, 색상이나 글꼴 등을 설정하는 요소
<BorderStyle>	엑셀 영역에서 테두리를 설정하는 요소

요소의 상세설명은 Label > Range의 하단 요소 설명을 참조한다.

➤ Example

```
<?xml version="1.0" ?>
<Excel>
```



```
<DataPattern>
  <Standard SheetCol="0" Word="소계*">
    <Range Col1="1" Col2="End" RowHeight="25">
      <CellFormat>
        <Merge>False</Merge>
        <Alignment Horizontal="Center" Vertical="Center"/>
        <Interior BackColor="RGB(0,255,0)" />
        <Font Name="굴림" Size="12" Bold="True" Italic="True"/>
      </CellFormat>
      <BorderStyle>
        <TopEdge          Style="Dash"          Weight="Medium"
Color="RGB(255,0,0)"/>
        <BottomEdge       Style="Dash"          Weight="Medium"
Color="RGB(0,0,0)"/>
        <LeftEdge         Style="Continuous"     Weight="Hairline"
Color="RGB(0,0,255)"/>
        <RightEdge        Style="Continuous"     Weight="Hairline"
Color="RGB(0,0,0)"/>
      </BorderStyle>
    </Range>
  </Standard>
</DataPattern>
</Excel>
```

➤ 제공 버전

7.0.0.0



6.4.9 BorderSet 요소

엑셀 워크시트의 셀 테두리를 설정한다.

이 요소는 하단의 Range 요소를 이용하여 영역을 선택하고, 영역에 테두리와 포맷을 설정한다.

"Excel 요소 > Label 요소"는 영역에 글자까지 설정할 수 있지만 이 요소는 글자를 설정할 수 없으며 영역의 테두리 설정을 기본으로 처리한다.

➤ 하위요소

속성	설 명
< Range >	패턴을 설정할 데이터의 기준을 설정한다.

➤ Example

```
<?xml version="1.0" ?>
<Excel>
  <BorderSet>
    <Range Row1="3" Col1="end" Row2="3" Col2="end" RowHeight="15">
      <BorderStyle>
        <TopEdge Style="Double" Weight="thick"/>
        <BottomEdge Style="Double" Weight="thick"
Color="RGB(0,0,250)"/>
        <LeftEdge Style="Double" Weight="thick" Color="RGB(0,0,250)"/>
        <RightEdge Style="Double" Weight="thick" Color="RGB(0,0,250)"/>
        <InVertical Style="Double" Weight="thick" Color="RGB(0,0,250)"/>
        <InHorizontal Style="Double" Weight="thick"
Color="RGB(0,0,250)"/>
      </BorderStyle>
      <CellFormat>
        <Alignment Horizontal="Center" Vertical="Center"/>
        <Interior Pattern="test" PatternColor="RGB(255,255,255)"/>
        <Font Name="굴림" Bold="True" Italic="True" Size="8"
```



```
UnderLine="None" Strikethrough="False"/>
    </CellFormat>
</Range>
</BorderSet>
</Excel>
```

➤ 제공 버전

7.0.0.0

6.4.9.1 Range 요소(BorderSet 요소의 하위)

엑셀의 셀에 원하는 글자를 설정할때 셀 영역을 의미한다.

이 요소는 테두리를 설정할 엑셀의 영역을 설정하기 위한 요소이다. 이 요소의 속성 Row1, Row2는엑셀의 행번호를 의미하며 엑셀은 행을 1부터 시작하므로 Sheet의 시작행과 동일하게 처리하면 안된다. Row1, Row2 인자의 행번호를 마지막 행번호 사용할 경우 "end"라는 글자로 설정할수 있다.

Col1, Col2 인자도 동일하게 1컬럼부터 시작하며, 마지막 컬럼을 사용할 경우 "end"라는 글자를 설정할수 있다.

Label을 표시할 엑셀의 영역이 1개인 경우 Row1과 Col1 속성만 사용하고, Row2와 Col2 속성은 제외하고 사용할 수 있다.

➤ 속성

속성	설 명
Col1	엑셀 영역의 시작 컬럼번호 또는 "End"
Row1	엑셀 영역의 시작 행번호 또는 "End"
Col2	엑셀 영역의 마지막 컬럼번호 또는 "End"
Row2	엑셀 영역의 마지막 행번호 또는 "End"
RowHeights	엑셀 영역의 행높이, 픽셀단위

➤ 하위요소

속성	설 명
<CellFormat>	엑셀 영역의 머지, 정렬, 색상이나 글꼴 등을 설정하는 요소
<BorderStyle>	엑셀 영역에서 테두리를 설정하는 요소

요소의 상세설명은 Label > Range의 하단 요소 설명을 참조한다.

➤ Example



```
<?xml version="1.0" ?>
<Excel>
  <BorderSet>
    <Range Row1="11" Col1="1" Row2="13" Col2="End">
      <BorderStyle>
        <TopEdge Style="Double" Weight="thick"/>
        <BottomEdge Style="Double" Weight="thick"
Color="RGB(0,0,250)"/>
        <LeftEdge Style="Double" Weight="thick" Color="RGB(0,0,250)"/>
        <RightEdge Style="Double" Weight="thick" Color="RGB(0,0,250)"/>
      </BorderStyle>
      <CellFormat>
        <Alignment Horizontal="Center" Vertical="Center"/>
        <Interior Pattern="test" PatternColor="RGB(255,255,255)"/>
        <Font Name="굴림" Bold="True" Italic="True" Size="8"/>
      </CellFormat>
    </Range>
  </BorderSet>
</Excel>
```

➤ **제공 버전**

7.0.0.0





Chapter 7. IBSheet Config





7. IBSheet Config

7.1 Config 설정

모든 IB Sheet 에 전역으로 설정 하고자 할 때 ibsheet.cfg 파일에 해당 속성을 정의 한다. 파일의 포맷은 xml 형태이다.

예를 들어 엑셀 다운로드 Url 을 설정 한다면 다음과 같이 ibsheet.cfg 파일에 설정 한다.

```
<SHEET>
    <Cfg Down2Excel_Url="../jsp/Down2Excel.jsp" />
</SHEET>
```

전역 설정 할 수 있는 속성은 다음과 같다.



7.2 Property List (빈번히 사용되는 속성)

속성명	설명										
Alternate	AlternateBackColor 사용여부										
	<table><tr><th>설정값</th><th>내용</th></tr><tr><td>0</td><td>사용안함</td></tr><tr><td>1</td><td>사용함 (Default)</td></tr></table>	설정값	내용	0	사용안함	1	사용함 (Default)				
	설정값	내용									
	0	사용안함									
1	사용함 (Default)										
AutoClearHeaderCheck	RemoveAll 메소드 호출시 또는 조회시 전체체크를 자동 초기화 여부 설정										
	<table><tr><th>설정값</th><th>내용</th></tr><tr><td>0</td><td>사용안함 (Default)</td></tr><tr><td>1</td><td>사용함</td></tr></table>	설정값	내용	0	사용안함 (Default)	1	사용함				
	설정값	내용									
	0	사용안함 (Default)									
1	사용함										
AutoCloseDialog	콤보목록, 달력팝업, 액션메뉴 팝업의 자동닫기 여부 설정.										
	<table><tr><th>설정값</th><th>내용</th></tr><tr><td>0</td><td>사용안함 (Default)</td></tr><tr><td>1</td><td>사용함</td></tr></table>	설정값	내용	0	사용안함 (Default)	1	사용함				
	설정값	내용									
	0	사용안함 (Default)									
1	사용함										
AutoCloseDialogTimeout	콤보목록, 달력팝업, 액션메뉴 팝업의 자동닫기 지연 시간 설정 (단위 : ms) (Default : 300)										
AutoFitColWidth	특정 시점에서 컬럼의 너비를 자동으로 조정하는 FitColWidth() 함수를 호출한다. 적용하고자 하는 시점을 구분자 " " 로 연결하여 설정한다. (Default : "")										
	<table><tr><th>설정값</th><th>내용</th></tr><tr><td>search</td><td>조회 및 로드 시점</td></tr><tr><td>resize</td><td>시트 Resize 시점</td></tr><tr><td>init</td><td>초기화 및 RemoveAll 호출 시점</td></tr><tr><td>colhidden</td><td>컬럼 숨김/보임 시점</td></tr></table>	설정값	내용	search	조회 및 로드 시점	resize	시트 Resize 시점	init	초기화 및 RemoveAll 호출 시점	colhidden	컬럼 숨김/보임 시점
	설정값	내용									
	search	조회 및 로드 시점									
	resize	시트 Resize 시점									
	init	초기화 및 RemoveAll 호출 시점									
	colhidden	컬럼 숨김/보임 시점									

	rowtransaction	로우 추가/삭제/숨김/보임 이후
AutoRowHeight	조회된 데이터에 따라 각 행의 높이를 자동으로 설정	
	설정값	내용
	0	사용안함
	1	사용함 (Default)
AutoSumCalcMode	병합된 셀에 대한 합계 계산 처리 방법 설정	
	설정값	내용
	0	모든 셀에 대한 합계 계산 (Default)
	1	병합된 셀의 부속셀은 제외 하고 계산
CachePageCount	페이지 캐싱 개수 설정	
CalButtonAlign	달력팝업에서 버튼을 사용하는 경우 정렬방법 설정 이 설정은 시트내의 달력팝업에만 적용된다.	
	설정값	내용
	Left	왼쪽정렬
	Center	가운데정렬 (Default)
	Right	오른쪽정렬
CalButtons	달력팝업의 사용할 버튼 설정 사용하고자 하는 버튼을 " " 로 연결하여 설정한다. 이 설정은 시트내의 달력팝업에만 적용된다.	
	설정값	내용
	Close	취소 버튼
	Today	오늘일자 입력 버튼
	Yesterday	어제일자 입력 버튼
	InputEmpty	빈값 입력 버튼
	Ok	선택 버튼



CalWeekNumber	달력팝업의 주차 표시 설정	
	설정값	내용
	0	주차 표시하지 않음
	1	연내 주차 표시 (Default)
	2	월내 주차 표시
CellItemsSeparator	멀티 아이템에 대한 아이템간 구분자 설정 (Default : " ")	
CellItemsKeyValueSeparator	멀티 아이템에 대한 Key-Value 간 구분자 설정 (Default : ":")	
CheckActionKey	체크박스, 라디오 타입 컬럼에서 Space 또는 Enter 키를 이용하여 체크 동작을 하도록 설정하는 옵션 사용자가 설정한 키 입력 시 선택/해제가 토글로 동작한다.	
	설정값	내용
	Space	Space 키 사용
	Enter	Enter키 사용
	Space Enter	Space, Enter 키 사용
CheckActionMode	체크박스 타입 컬럼의 체크/언체크 동작 발생 시점 설정	
	설정값	내용
	0	OnClick 이벤트 발생 이전에 동작 수행 (Default)
	1	OnClick 이벤트 발생 이후에 동작 수행
ClipPasteMode	클립보드 붙여넣기 방법 설정	
	설정값	내용
	-1	사용안함



	0	선택 셀 (Default)
	1	선택 셀 기준 구분자 처리
	2	선택 셀 기준 구분자 (부족한 경우 행 추가)
	3	신규 행
※ 설정값 세부설명은 ClipPasteMode() 메서드 참조		
ComboMaxHeight	콤보 리스트의 최대 높이	
ComboOpenMode	포커스 이동 간에 콤보목록 열기 방법 설정	
	설정값	내용
	0	콤보 리스트 목록을 표시하지 않음.(Default)
	1	포커스시 콤보 리스트 표시 처리
ComboSettingMode	콤보아이템을 Trim할 것인지 안할 것인지 설정.	
	설정값	내용
	0	Trim 처리 안함 (Default)
	1	Trim 처리 함
ComboEditValidation	콤보에디트에서 직접 입력한 값에 대한 유효성 검사를 사용할지 여부 설정	
	설정값	내용
	0	유효성검사 안함 (Default)
	1	유효성 검사 함
CountFormat	보여지는 건수 정보의 포맷 설정 (세부 사항은 해당 CountFormat() 메서드 참조)	
CountPosition	건수정보 표시 여부 및 위치 설정	
	설정값	내용
	0	표시 하지 않음 (Default)
	1	좌측 상단
	2	우측 상단
	3	좌측 하단
	4	우측 하단



CustomScroll	<p>가로/세로 스크롤바에 대한 디자인 설정. 스크롤바의 디자인 처리와 모바일 출력을 위해 이 미지 스크롤바 기능을 제공한다.</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>1</td><td>18px 크기의 스크롤바 디자인</td></tr> <tr> <td>2</td><td>10px 크기의 스크롤바 디자인</td></tr> <tr> <td>3</td><td>9px 크기의 스크롤바 디자인</td></tr> </table>	설정값	내용	1	18px 크기의 스크롤바 디자인	2	10px 크기의 스크롤바 디자인	3	9px 크기의 스크롤바 디자인
설정값	내용								
1	18px 크기의 스크롤바 디자인								
2	10px 크기의 스크롤바 디자인								
3	9px 크기의 스크롤바 디자인								
DataTypeToInvalidMessage	<p>유효하지 않은 값 입력에 대한 메시지 출력 대상 데이터 타입 설정. 대상 타입을 구분자 ' '로 연결한 문자열로 설정한 다. (Default = "Date")</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>Date</td><td>날짜 형태의 데이터 타입</td></tr> <tr> <td>Number</td><td>숫자 형태의 데이터 타입</td></tr> <tr> <td>Combo</td><td>콤보 형태의 데이터 타입</td></tr> </table>	설정값	내용	Date	날짜 형태의 데이터 타입	Number	숫자 형태의 데이터 타입	Combo	콤보 형태의 데이터 타입
설정값	내용								
Date	날짜 형태의 데이터 타입								
Number	숫자 형태의 데이터 타입								
Combo	콤보 형태의 데이터 타입								
DataRowHeight	<p>데이터 행의 기본 높이 값 (행의 높이는 css 가 아닌 이 속성으로 처리해야 함)</p>								
DeferredHScroll	<p>가로 스크롤에 대한 지연 처리 모드 설정</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>사용 안함 (Default)</td></tr> <tr> <td>1</td><td>사용 함.</td></tr> </table>	설정값	내용	0	사용 안함 (Default)	1	사용 함.		
설정값	내용								
0	사용 안함 (Default)								
1	사용 함.								
DeferredScrollTime	<p>스크롤 지연 모드 사용시 지연 처리 시간 설정 (단위:ms) (Default:200)</p>								
DeferredVScroll	<p>세로 스크롤에 대한 지연 처리 모드 설정</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>사용 안함 (Default)</td></tr> <tr> <td>1</td><td>사용 함.</td></tr> </table>	설정값	내용	0	사용 안함 (Default)	1	사용 함.		
설정값	내용								
0	사용 안함 (Default)								
1	사용 함.								
DirectLoadExcel_Url	<p>(서버로 직접) 엑셀 업로드 처리를 담당하는 jsp 파 일의 경로</p>								
DocIODelimMode	<p>문서 연동시 서버로 전달되는 전문의 구분자 설정 모드</p>								



	<p>※ 신규 방식 모드로 설정하는 경우 각 문서연동 처리를 하는 서버 파일에서도 동일한 값으로 설정하여야 한다.</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>기존 방식 (Default)</td></tr> <tr> <td>1</td><td>신규 방식</td></tr> </table>	설정값	내용	0	기존 방식 (Default)	1	신규 방식		
설정값	내용								
0	기존 방식 (Default)								
1	신규 방식								
Down2Excel_Url	엑셀 다운로드 처리를 담당하는 jsp 파일의 경로								
Down2Pdf_Url	PDF 다운로드 처리를 담당하는 jsp 파일의 경로								
DownWaitImageVisible	<p>파일 다운 중 대기 이미지 출력여부 설정 출력 안함으로 설정 시 OnDownFinish 이벤트도 발생하지 않는다.</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>대기이미지,이벤트 출력 안함</td></tr> <tr> <td>1</td><td>대기이미지,이벤트 출력 함 (Default)</td></tr> </table>	설정값	내용	0	대기이미지,이벤트 출력 안함	1	대기이미지,이벤트 출력 함 (Default)		
설정값	내용								
0	대기이미지,이벤트 출력 안함								
1	대기이미지,이벤트 출력 함 (Default)								
DownloadImage	파일 다운로드중 이미지 경로								
DragCell	<p>드래그시 셀 단위 드래그 여부 설정</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>행 단위 드래깅 (Default)</td></tr> <tr> <td>1</td><td>셀 단위 드래깅</td></tr> </table>	설정값	내용	0	행 단위 드래깅 (Default)	1	셀 단위 드래깅		
설정값	내용								
0	행 단위 드래깅 (Default)								
1	셀 단위 드래깅								
DragMode	<p>드래그 처리 방법 설정</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>-1</td><td>드래그 사용 안함</td></tr> <tr> <td>0</td><td>일반 : 셀 또는 행 범위 선택 Ctrl 키 이용 : 행 드래깅 (Default)</td></tr> <tr> <td>1</td><td>일반 : 행 드래깅 Ctrl 키 이용 : 셀 또는 행 범위 선택</td></tr> </table>	설정값	내용	-1	드래그 사용 안함	0	일반 : 셀 또는 행 범위 선택 Ctrl 키 이용 : 행 드래깅 (Default)	1	일반 : 행 드래깅 Ctrl 키 이용 : 셀 또는 행 범위 선택
설정값	내용								
-1	드래그 사용 안함								
0	일반 : 셀 또는 행 범위 선택 Ctrl 키 이용 : 행 드래깅 (Default)								
1	일반 : 행 드래깅 Ctrl 키 이용 : 셀 또는 행 범위 선택								
DragPopup	<p>시트내 팝업다이얼로그의 드래그기능 사용 여부</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>사용 안함 (Default)</td></tr> <tr> <td>1</td><td>사용함.</td></tr> </table>	설정값	내용	0	사용 안함 (Default)	1	사용함.		
설정값	내용								
0	사용 안함 (Default)								
1	사용함.								



EditEnterBehavior	편집중 엔터키 입력에 대한 동작 설정	
	설정값	내용
	tab	편집 완료 후 오른쪽 셀로 포커스 이동 (Default)
	down	편집 완료 후 아래 셀로 포커스 이동
	newline	줄바꿈되어 개행 처리 됨
	none	편집 완료 후 포커스 유지
	editTab	편집 완료 후 다음 편집 가능한 셀로 이동
EditLenMode	입력문자열 길이 체크 방법 설정	
	설정값	내용
	0	편집모드, 메소드에서 체크함 (Default)
	1	편집모드에서 직접 입력시에만 체크
EditTabBehavior	편집중 탭키 입력에 대한 동작 설정	
	설정값	내용
	0	편집 가능한 다음 셀로 이동 (Default)
	1	편집여부와 상관없이 다음 셀로 이동 (다음셀이 편집 가능한 경우면 편집상태 유지)
	2	편집여부와 상관없이 다음 셀로 이동 (다음셀의 편집 가능여부와 무관하게 편집상태 종료)

EventCacheMode	<p>이벤트 사용여부 체크에 대한 캐싱 사용여부 설정값이 0인 경우 이벤트 발생시점에 사용여부를 항상 체크하게 되고, 설정값이 1인 경우 이벤트 발생시점에 사용여부를 이벤트별 최초 1회만 체크하여 0의 설정보다 높은 성능을 보인다. 캐싱모드 사용시 이벤트를 동적으로 할당하는 경우 최초 이벤트 발생시점 이전에 할당하여 사용하여야 한다.</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>사용 안함</td></tr> <tr> <td>1</td><td>사용함 (Default)</td></tr> </table>	설정값	내용	0	사용 안함	1	사용함 (Default)
설정값	내용						
0	사용 안함						
1	사용함 (Default)						
ExcelTreeMode	<p>엑셀 다운시 트리의 접혀진 행을 다운 받을지 설정</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>트리의 접혀진 행을 다운 안함</td></tr> <tr> <td>1</td><td>트리의 접혀진 행도 모두 다운 SetRowHidden 으로 숨긴 로우도 모두 다운</td></tr> </table>	설정값	내용	0	트리의 접혀진 행을 다운 안함	1	트리의 접혀진 행도 모두 다운 SetRowHidden 으로 숨긴 로우도 모두 다운
설정값	내용						
0	트리의 접혀진 행을 다운 안함						
1	트리의 접혀진 행도 모두 다운 SetRowHidden 으로 숨긴 로우도 모두 다운						
ExtendLastCol	<p>마지막 컬럼 확장 여부</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>사용안함. (Default)</td></tr> <tr> <td>1</td><td>사용함</td></tr> </table>	설정값	내용	0	사용안함. (Default)	1	사용함
설정값	내용						
0	사용안함. (Default)						
1	사용함						
FalseValue	<p>0 이외의 CheckBox 형태 컬럼의 False 값 지정. "F"으로 지정한 경우 0 대신 "F"을 False 값으로 사용 가능.</p>						
FilterCaseSensitive	<p>필터 사용시 영문 대소문자 구분 사용여부</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>사용안함.</td></tr> <tr> <td>1</td><td>사용함 (Default)</td></tr> </table>	설정값	내용	0	사용안함.	1	사용함 (Default)
설정값	내용						
0	사용안함.						
1	사용함 (Default)						



FilterComboSort	필터 사용시 콤보 타입의 목록 정렬 설정	
	설정값	내용
		사용안함.(Default)
	ASC	오름차순 정렬
FilteredCountFormat	DESC	내림차순 정렬
	필터링 상태에서 보여지는 건수 정보의 포맷 설정 (설정방법은 CountFormat 과 동일)	
	필터 셀 편집시 다이얼로구 사용여부 설정	
	설정값	내용
FilterInputPopup	0	사용안함. (Default)
	1	사용함
	컬럼에 대한 FitSize 처리시 컬럼간 셀병합이 되어 있는 헤더를 포함하여 처리할지 여부 설정	
	설정값	내용
FitSizeColHeaderMode	0	포함하여 처리 (Default)
	1	제외하고 처리
	헤더의 컬럼 사이를 더블클릭시 처리되는 FitSize 처리 모드 설정	
	설정값	내용
FitSizeColMode	0	화면에 보여지는 행 범위 내에서 처리.
	1	전체 행 범위 에서 처리
	조회후 포커스 처리방법 설정	
	설정값	내용
FocusAfterProcess	0	포커스 처리 안함
	1	포커스 처리 함 (Default)
	행 추가/삭제/이동/복사 후 포커스 처리방법 설정	
	설정값	내용
FocusAfterRowTransaction	0	포커스 이동 안함
	1	포커스 이동 함 (Default)
	포커스시 편집여부 설정	
	FocusEditMode	



	설정값	내용
	0	(기본값)
	1	포커스시 편집 모드
	2	포커스시 편집 모드 (Combo, ComboEdit 타입 제외)
FocusSumRow	합계행의 셀에 대한 Focus 허용 여부 설정	
GroupSort	그룹핑시 기준컬럼에 대한 Sort 처리여부 설정	
	설정값	내용
	0	Sort 처리 없이 현재 상태로 그룹핑
	1	Sort 처리 후 그룹핑 (기본값)
HeaderEventMode	헤더의 이벤트 발생종류 설정	
	설정값	내용
	0	모든 마우스 이벤트 발생 안함
	1	모든 마우스 이벤트 발생 함 (Default)
	2	MouseDown, MouseMove 이벤트만 발생 함
HeaderRowHeight	헤더행의 높이 (헤더 행의 높이는 css 가 아닌 이 속성을 통해 설정해야 함)	
HeaderMergeMode	헤더의 머지 방식 설정	
	설정값	내용
	0	ColMerge 속성 설정값에 상관없이 머지 처리(Default)
	1	ColMerge 속성에 따라 머지 처리
HeaderSort	헤더 클릭을 통한 컬럼 Sort 사용 옵션 설정	
	설정값	내용
	0	사용 안함
	1	Sort 기능 사용
	2	Sort 아이콘만 표시 기능 사용



HeaderSortActionMode	단일 컬럼 정렬, 다중 컬럼 정렬을 위한 헤더의 사용자 입력 방식 모드 설정	
	설정값	내용
	0 (Default)	Click
		다중 컬럼 정렬
	1	Ctrl + Click
		단일 컬럼 정렬
Holiday	달력에 표시할 사용자 정의 휴일 설정 파이프 연산자로 설정값을 연결한 문자열로 설정할 수 있으며 포맷은 yyyyMMdd 만 허용한다. 매년, 매월에 대한 와일드카드로 * 을 사용한다. 설정 예) Holiday = "20120725 *0703 2012*27 **17" *0703 : 매년 7월 3일 2012*27 : 2012년 매월 27일 **17 : 매년, 매월 17일	
	상태값 표현을 이미지로 할지 여부 설정 (Default:0)	
	시트 초기화 시 랜더링 여부 설정.	
	설정값	내용
	0	초기화 시 랜더링 안함 (Default)
	1	초기화 시 랜더링 함
InvalidArgsReturnValue	메소드 인자가 잘못 설정된 경우의 반환값 설정 (Default = -1)	
InvalidInputBehavior	편집 완료시점에 입력값이 invalid 인 경우에 대한 처리 방법 설정	
	설정값	내용
	0	alert 메시지 후 해당 셀 편집상태 유지 (Default)
	1	alert 메시지 후 해당 셀 편집종료(입력값 취소 처리)



	2	alert 메시지 후 해당 셀 편집상태 유지 (입력값 취소 처리)
InvalidPasteMsgMode	설정값	내용
	1	메시지 출력 함 (Default)
	2	메시지 출력 안함
JsonAttributeDelimiter	Json 조회시 속성 구분자 변경 설정 (Default : '#')	
KeyFieldPosition	필수 입력 표시 위치	
	설정값	내용
	Left	왼쪽에 표기 (Default)
	Right	오른쪽에 표기
LoadExcel_Url	엑셀 업로드 처리를 담당하는 jsp 파일의 경로	
MarkupTagDelimiter	<p>서버모듈 연동시 전송되는 전문의 마크업 태그의 구분문자를 설정한다. 설정값은 해당 구분문자를 구분자 ","로 연결한 4개의 문자(또는 문자열)로 설정한다.</p> <p>기본값과 다르게 설정한 경우 서버단 수행로직에서도 동일한 값으로 각각 설정하여야 한다.</p> <p>(Default="<,>,</,>")</p>	
MaxSort	헤더클릭을 이용한 연계 소팅 최대 개수를 설정한다.	
MessageShowLevel	Alert 또는 Confirm 메시지의 출력 레벨을 설정한다.	
	설정값	내용
	U	사용자 노출용 메시지만 출력 (Default)
	D	사용자용 메시지와 개발자용 메시지를 모두 출력



MouseHoverMode	시트내의 행 또는 셀에 MouseOver에 대한 Hover 방식을 설정		
		설정값	내용
		0	사용안함 (Default)
		1	셀단위 Hover
		2	행단위 Hover
NextPageCall	조회 모드가 서버페이징, LazyLoad 방식인 경우 다음 페이지를 호출하는 시점 설정 %에 해당하는 숫자값으로 60~90 사이만 설정 가능		
NewRowDeleteMode	"입력" 상태인 행의 삭제 컬럼 체크시 confirm 메시지 사용 여부 설정		
		설정값	내용
		0	사용안함 (Default)
		1	사용함
NoImageUrl	이미지 경로가 잘못된 경우 대체할 이미지의 경로를 설정. 설정시 기본경로는 ibsheet.js 파일위치를 기준으로 하며, 해당 속성의 이미지 경로가 잘못된 경우 성능에 심각한 문제를 발생하기 때문에 정확하게 설정 하여야 한다.		
PagingPosition	페이지 네비게이션 표시 여부 및 위치 설정		
		설정값	내용
		0	표시 하지 않음 (Default)
		1	좌측
		2	우측
PasteToNumberFormat	숫자 형태 데이터타입의 셀 범위 선택 영역에 붙여넣기시 Format 에 맞게 데이터 가공 처리 여부		
		설정값	내용
		0	입력값 그대로 사용 (Default)
		1	Format, PointCount 기준 절삭 처리

PopupCheckEditMode	<p>컬럼팝업메뉴를 통한 값 설정시 편집가능여부에 따른 처리 여부.</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>편집가능여부와 무관하게 처리</td></tr> <tr> <td>1</td><td>편집가능한 경우에만 처리 (Default)</td></tr> </table>	설정값	내용	0	편집가능여부와 무관하게 처리	1	편집가능한 경우에만 처리 (Default)
설정값	내용						
0	편집가능여부와 무관하게 처리						
1	편집가능한 경우에만 처리 (Default)						
RefreshHeaderMode	<p>헤더 갱신모드 설정</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>헤더 리프레쉬 안함</td></tr> <tr> <td>1</td><td>컬럼이동시 헤더 리프레쉬함 (Default)</td></tr> </table>	설정값	내용	0	헤더 리프레쉬 안함	1	컬럼이동시 헤더 리프레쉬함 (Default)
설정값	내용						
0	헤더 리프레쉬 안함						
1	컬럼이동시 헤더 리프레쉬함 (Default)						
ReverseSortOrder	<p>헤더 클릭시 소트 처리 방법 설정</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>마지막 선택 컬럼 우선 정렬 (Default)</td></tr> <tr> <td>1</td><td>이전 선택 컬럼 우선 정렬</td></tr> </table>	설정값	내용	0	마지막 선택 컬럼 우선 정렬 (Default)	1	이전 선택 컬럼 우선 정렬
설정값	내용						
0	마지막 선택 컬럼 우선 정렬 (Default)						
1	이전 선택 컬럼 우선 정렬						
RowHeightMax	<p>행의 최대 높이를 설정한다. 데이터에 줄넘김 글자가 많아 여기서 설정한 높이보다 커지는 경우 편집시 스크롤 바가 나타난다.</p>						
SaveImage	<p>저장시 나타나는 대기 이미지를 설정한다.</p>						
SaveValidationMode	<p>저장시 숨김 컬럼에 대한 필수입력, 전체입력에 대한 체크 여부를 설정한다.</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>모든 컬럼에 대한 체크 (Default)</td></tr> <tr> <td>1</td><td>Visible 컬럼에 대한 체크</td></tr> </table>	설정값	내용	0	모든 컬럼에 대한 체크 (Default)	1	Visible 컬럼에 대한 체크
설정값	내용						
0	모든 컬럼에 대한 체크 (Default)						
1	Visible 컬럼에 대한 체크						
SearchImage	<p>조회시 나타나는 대기 이미지를 설정한다.</p>						



SearchMode	조회 모드 설정	
	설정값	내용
	0	smGeneral - 일반 조회 모드
	1	smClientPaging - 페이징 모드
	2	smLazyLoad - Lazy Load 모드 (Default)
	3	smServerPaging - 실시간 서버 처리 모드 (스크롤링 방식)
SearchSync	조회 메소드 전체의 동기조회 사용 여부를 설정	
	설정값	내용
	0	동기조회 사용 안함
	1	동기조회 사용
SelectCellEventMode	조회된 데이터가 없는 경우에도 OnSelectCell 이벤트를 발생시킬지 여부를 설정	
	설정값	내용
	0	OnSelectCell 이벤트 발생 안함 (Default)
	1	OnSelectCell 이벤트 발생
SelectionRowsMode	GetSelectionRows 시 포커스행 포함 여부 설정	
	설정값	내용
	0	포커스행 포함함 (Default)
	1	포커스행 포함하지 않음
SelectionSummary	선택영역 요약 정보 표시 설정 아래의 옵션을 " " 로 연결하여 설정한다.	
	설정값	내용
	EmptyCell	빈값의 셀을 포함
	NotEmptyCell	빈값의 셀을 포함하지 않음
	DelRow	삭제상태의 행을 포함
	NotDelRow	삭제상태의 행을 포함하지 않음



SkipDefaultTheme	<p>시트 생성시 기본테마 적용 생략 여부</p> <p>※ 1설정시 초기화 시점에 SetTheme를 반드시 호출하여야 함</p> <table border="1"> <thead> <tr> <th>설정값</th><th>내용</th></tr> </thead> <tbody> <tr> <td>0</td><td>기본테마 적용함 (Default)</td></tr> <tr> <td>1</td><td>기본테마 적용 생략 함</td></tr> </tbody> </table>	설정값	내용	0	기본테마 적용함 (Default)	1	기본테마 적용 생략 함
설정값	내용						
0	기본테마 적용함 (Default)						
1	기본테마 적용 생략 함						
SmartResize	<p>OnResize를 OnSmartResize 동작처럼 변환 이 값을 1로 설정하면 OnSmartResize이벤트는 발생하지 않고 OnResize이벤트가 OnSmartResize 이벤트 처럼 0.3초단위로 동작한다.</p> <table border="1"> <thead> <tr> <th>설정값</th><th>내용</th></tr> </thead> <tbody> <tr> <td>0</td><td>사용안함 (Default)</td></tr> <tr> <td>1</td><td>사용함</td></tr> </tbody> </table>	설정값	내용	0	사용안함 (Default)	1	사용함
설정값	내용						
0	사용안함 (Default)						
1	사용함						
SortCaseSensitive	<p>컬럼 Sort 처리시 문자열에 대한 대소구분 처리 여부 설정</p> <table border="1"> <thead> <tr> <th>설정값</th><th>내용</th></tr> </thead> <tbody> <tr> <td>0</td><td>사용안함</td></tr> <tr> <td>1</td><td>사용함 (Default)</td></tr> </tbody> </table>	설정값	내용	0	사용안함	1	사용함 (Default)
설정값	내용						
0	사용안함						
1	사용함 (Default)						
SortEventMode	<p>OnSort 이벤트 발생시 정렬된 모든 컬럼의 정보를 반환 할지 여부.</p> <table border="1"> <thead> <tr> <th>설정값</th><th>내용</th></tr> </thead> <tbody> <tr> <td>0</td><td>마지막 정렬 컬럼만 반환 (Default)</td></tr> <tr> <td>1</td><td>정렬된 모든 컬럼 반환</td></tr> </tbody> </table>	설정값	내용	0	마지막 정렬 컬럼만 반환 (Default)	1	정렬된 모든 컬럼 반환
설정값	내용						
0	마지막 정렬 컬럼만 반환 (Default)						
1	정렬된 모든 컬럼 반환						
SparklineColor	<p>스파크라인 차트의 양수 값(영역)에 대한 색상 값 설정</p>						
SparklineNegativeColor	<p>스파크라인 차트의 음수 값(영역)에 대한 색상 값 설정</p>						
SubSumMode	<p>소계행 표시 방법 설정</p> <table border="1"> <thead> <tr> <th>설정값</th><th>내용</th></tr> </thead> <tbody> <tr> <td>0</td><td>모든 대상 그룹에 대한 소계행 표시 (Default)</td></tr> </tbody> </table>	설정값	내용	0	모든 대상 그룹에 대한 소계행 표시 (Default)		
설정값	내용						
0	모든 대상 그룹에 대한 소계행 표시 (Default)						



	1	대상 그룹이 2개 이상인 그룹에 대해서만 소계행 표시
SumBackColor	합계행의 배경 색상	
SumFontBold	합계 행의 Font Bold 여부	
SumFontColor	합계 행의 Font 색상	
SumZeroValue	합계 행의 합계값이 0 인경우 치환값 설정	
SyncPaste	클립보드 붙여넣기를 동기방식으로 처리할지 여부 설정	
	설정값	내용
	0	비동기 방식 처리
	1	동기 방식 처리. (Default)
TapHoldTreshold	모바일에서 TapHold 인지 시간 (단위 : ms) (Default : 800)	
ThemeVersion	사용할 테마 버전 설정	
	설정값	내용
	0	일반 테마 버전 (Default)
	1	IE10 이상 모던 테마 버전
	<p>※ 설정값과 테마의 버전이 상이한 경우 화면의 깨짐현상이 있을수 있음</p> <p>- 모던 테마 : ModernWhite, ModernBlue</p>	
Timestamp	날짜 타입에 대한 timestamp 형식의 데이터 사용여부 설정.	
	설정값	내용
	0	사용함 (Default)
	1	사용 안함.



TreeDragIconMode	트리 구조에서 드래그시 드래그 위치 아이콘 표시 모드	
	설정값	내용
	0	이동이 가능한 위치에 맞게 아이콘 표시 (Default)
	1	이동 위치와 상관없이 모든 아이콘 표시.
TreeNodeIcon	트리 구조에서 트리 노드 아이콘 표시 여부 (Default : 0)	
TreeNodeToggleMode	마우스 입력을 통한 Tree 노드 토글시 처리 방법 설정	
	설정값	내용
	0	토글후 해당 셀에 포커스 이동 처리 방법 (Default)
	1	셀의 포커스 이동 없이 토글 처리 방법
TrueValue	1 이외의 CheckBox 형태 컬럼의 True 값 지정. "M"으로 지정한 경우 1 대신 "M"을 True 값으로 사용 가능.	
ToolTipMode	툴팁 표시 모드 설정	
	설정값	내용
	0	div 레이어를 이용한 표현 (Default)
	1	element 의 title 속성을 이용한 표현
UnicodeByte	UnicodeByte 값을 설정한다. (세부 설정은 UnicodeByte 메소드 참조)	
UploadImage	파일 업로드중 이미지 경로	



UseCache	css, msg 파일의 브라우저 캐쉬 사용여부	
	설정값	내용
	0	브라우저 캐쉬 사용안함 (Default)
	1	브라우저 캐쉬 사용함
UseDefaultTime	편집시 현재시간 표시 여부	
UseEditMask	<p>편집시 사용자포맷, 날짜에 대한 IMaskEdit 사용 여부 (Default:0)</p> <p>(세부 사항은 SetConfig 메소드의 UseEditMask 속성 참고)</p>	
UseFindDialog	찾기 팝업 다이얼로그 사용 여부 설정	
	설정값	내용
	0	사용 안함
	1	사용 함 (Default)
UseHeaderActionMenu	<p>헤더 컨텍스트 메뉴 사용여부 설정</p> <p>헤더메뉴를 사용함으로 설정하고 헤더메뉴를 설정하지 않는경우 자동으로 기본 메뉴가 출력된다.</p> <p>(HeaderActionMenu 참조)</p>	
	설정값	내용
	0	속성 사용 안함 (Default)
	1	속성 사용 함
UseHeaderSortCancel	헤더 클릭을 통한 컬럼 Sort 처리시 취소 모드 사용 여부	
	설정값	내용
	0	속성 사용 안함 (Default) 헤더 클릭시 Asc / Desc 순으로 처리
	1	속성 사용 함 헤더 클릭시 Asc / Desc / 취소 순으로 처리

UseGroupActionMenu	<p>그룹행에서 컨텍스트 메뉴 사용 여부 (GroupActionMenu 참조)</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>속성 사용 안함 (Default)</td></tr> <tr> <td>1</td><td>속성 사용 함</td></tr> </table>	설정값	내용	0	속성 사용 안함 (Default)	1	속성 사용 함
설정값	내용						
0	속성 사용 안함 (Default)						
1	속성 사용 함						
UseJsonAttribute	<p>Json 형식의 데이터 구조에서 Attribute 사용 여부 설정</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>속성 사용 안함 (Default)</td></tr> <tr> <td>1</td><td>속성 사용 함</td></tr> </table>	설정값	내용	0	속성 사용 안함 (Default)	1	속성 사용 함
설정값	내용						
0	속성 사용 안함 (Default)						
1	속성 사용 함						
UseJsonTreeLevel	<p>Json 형식의 데이터 구조에서 트리 형식 구성시 Level 속성을 이용하여 구성할지 여부 설정</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>Level 속성 사용 안함 (Default)</td></tr> <tr> <td>1</td><td>Level 속성을 사용 함</td></tr> </table>	설정값	내용	0	Level 속성 사용 안함 (Default)	1	Level 속성을 사용 함
설정값	내용						
0	Level 속성 사용 안함 (Default)						
1	Level 속성을 사용 함						
UseNoDataRow	<p>조회된 데이터가 없는 경우, '조회된 데이터가 없습니다' 문구 표시 여부 설정</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>문구 사용 안함</td></tr> <tr> <td>1</td><td>문구 사용 함 (Default)</td></tr> </table>	설정값	내용	0	문구 사용 안함	1	문구 사용 함 (Default)
설정값	내용						
0	문구 사용 안함						
1	문구 사용 함 (Default)						
UsePivotDialog	<p>피벗 설정 팝업 다이얼로그 사용 여부 설정</p> <table> <tr> <th>설정값</th><th>내용</th></tr> <tr> <td>0</td><td>사용 안함</td></tr> <tr> <td>1</td><td>사용 함 (Default)</td></tr> </table>	설정값	내용	0	사용 안함	1	사용 함 (Default)
설정값	내용						
0	사용 안함						
1	사용 함 (Default)						
UserAgent	User-Agent 값 설정						
WaitTimeOut	응답 대기시간 설정 (Default:60 , 단위:초)						
WheelScrollSize	휠 스크롤시 이동할 행의 개수 설정 (Default:1)						

➤ 제공 버전

7.0.0.0	
---------	--



7.0.1.0	ComboEditValidation 속성 추가
7.0.4.4	InvalidArgsReturnValue 속성 추가
7.0.4.6	CalWeekNumber 속성 추가
7.0.8.0	AutoFitColWidth, MouseHoverMode 속성 추가
7.0.11.0	PopupCheckEditMode, EventCacheMode, SparklineColor, SparklineNegativeColor 속성 추가 CalButtons 속성의 "InputEmpty" 값 추가
7.0.12.2	UnicodeByte, MarkupTagDelimiter 속성 추가
7.0.12.3	CellItemsSeparator, CellItemsKeyValueSeparator 속성 추가
7.0.13.5	FitSizeColHeaderMode, FilterInputPopup 속성 추가
7.0.13.6	HeaderSortActionMode 속성 추가
7.0.13.10	InvalidInputBehavior 속성 추가 CalButtons 속성의 "Ok" 값 추가
7.0.13.11	Timestamp, NolImageUrl, DeferredHScroll, DeferredVScroll, DeferredScrollTime 속성 추가
7.0.13.14	CachePageCount, HeaderSort 속성 추가
7.0.13.15	DataTypeToInvalidMessage 속성 추가
7.0.13.16	EditEnterBehavior 속성의 "editTab" 모드 추가
7.0.13.17	ImageStatus 속성 추가
7.0.13.21	SyncPaste 속성 추가
7.0.13.22	AutoSumCalcMode 속성 추가
7.0.13.26	GroupSort 속성 추가
7.0.13.28	PasteToNumberFormat 속성 추가
7.0.13.32	DocIODelimMode 속성 추가
7.0.13.33	UseEditMask 속성 추가
7.0.13.34	UseFindDialog, UsePivotDialog 속성 추가
7.0.13.39	LicenseCheckMode, LicenseFileName 속성 삭제
7.0.13.42	AutoCloseDialogTimeout, FilteredCountFormat 속성 추가
7.0.13.48	UseHeaderSortCancel 속성 추가
7.0.13.50	ThemeVersion 속성 추가
7.0.13.51	SubSumMode, FocusSumRow 속성 추가



7.0.13.52	TreeNodeToggleMode 속성 추가
7.0.13.54	TapHoldTreshold 속성 추가
7.0.13.56	TreeNodeIcon 속성 추가
7.0.13.60	InvalidPasteMsgMode 속성 추가





Chapter 8. IBSheet Events



8. IBSheet Events

8.1 Event 사용

IBSheet에서의 사용자가 원하는 기능을 추가 또는 변경 처리할 수 있도록 다양한 이벤트를 제공한다.

```
Function 오브젝트ID_이벤트명(파라미터, ...) { }
```

예를 들어 IBSheet ObjectID를 "mySheet"라 하고, OnChange 이벤트에 대해 기능 처리를 해야 한다면 다음과 같이 기술할 수 있다.

```
Function mySheet_OnChange(Row, Col, Value) {  
    alert(Row + "," + Col + "의 값이 변경되었습니다.");  
}
```

IBSheet가 제공하는 이벤트는 다음과 같다.



8.2 Event List

OnAfterColumnMove Event

➤ 기능

사용자가 마우스드래그를 이용하여 컬럼을 이동하거나 MoveColumnPos 함수를 이용하여 컬럼을 이동하고 이동이 성공적으로 완료되었을 때 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnAfterColumnMove(Col, NewPos) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Col	Long	이동한 컬럼의 Index
NewPos	Long	이동된 위치의 컬럼 Index

➤ Example

```
//컬럼이 이동된 후 이벤트가 발생한다.  
Function mySheet_OnAfterColumnMove(Col, NewPos) {  
    alert(Col + " 위치가 " + NewPos + " 위치로 이동되었습니다..");  
}
```

➤ 제공 버전

7.0.0.0

➤ See Also

[OnBeforeColumnMove](#) , [MoveColumnPos](#) , [MoveColumnFail](#)



OnAfterEdit Event

➤ 기능

셀의 값을 편집한 직후에 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnAfterEdit(Row, Col) { }
--------	--

➤ Parameters

Parameter	Type	설 명
Row	Long	해당 셀의 Row Index
Col	Long	해당 셀의 Column Index

➤ Example

<pre>//값을 편집한 직후 이벤트가 발생한다. Function mySheet_OnAfterEdit(Row, Col) { alert("입력을 마칩니다."); }</pre>
--

➤ 제공 버전

7.0.0.0

➤ See Also

[OnBeforeEdit](#)



OnAfterPaste Event

➤ 기능

클립보드 붙여넣기 완료 시점에 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnAfterPaste() { }
--------	---

➤ Parameters

Parameter	Type	설 명
없음		

➤ Example

<pre>function mySheet_OnAfterPaste() { console.log("클립보드 붙여넣기 완료"); }</pre>

➤ 제공 버전

7.0.13.21	
-----------	--



OnAfterExpand Event

➤ 기능

트리 형태 시트를 사용할 때 + 기호나 - 기호를 클릭하여 트리를 접거나 펼친 후에 이 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnAfterExpand(Row, Expand) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Row	Long	해당 셀의 Row Index
Expand	Long	7 : 펼치는 경우 2 : 접는 경우

➤ Example

<pre>function mySheet_OnAfterExpand(Row, Expand) { alert(Row + "행, " + Expand + "의 상태"); }</pre>

➤ 제공 버전

7.0.0.0

➤ See Also

[OnBeforeExpand](#), [RowExpand](#), [RowLevel](#), [ShowTreeLevel](#)



OnBeforeCheck Event

➤ 기능

사용자가 체크박스의 값 변경시 (마우스 클릭 또는 Space 키를 누름으로 변경) 편집하기 직전에 이벤트가 발생한다.

체크박스의 값 변경을 하기 직전에 이벤트가 발생하므로 체크시점에 조건에 의한 처리가 가능하다.

지원되는 데이터타입은 아래와 같다.

- DelCheck
- CheckBox
- Radio
- DummyCheck

➤ Syntax

Syntax	function 오브젝트ID_OnBeforeCheck(Row, Col) { }
--------	--

➤ Parameters

Parameter	Type	설 명
Row	Long	해당 셀의 Row Index
Col	Long	해당 셀의 Column Index

➤ Example

<pre>// 체크 되기 직전 이벤트가 발생한다. //체크하려는 행이 입력행이면 경고창 띄우기. Function mySheet_OnBeforeCheck(Row, Col) { if(mySheet.GetCellValue(Row,"sStatus") == "I") { alert("상태가 '입력'인 행은 선택이 불가능 합니다."); mySheet.SetAllowCheck(0);//체크를 막는다. }else{</pre>



```
mySheet.SetAllowCheck(1);//반드시 else인 경우에는 체크를 열어줘야 한  
다.  
}  
}
```

➤ **제공 버전**

7.0.0.0



OnBeforeCheckAll Event

➤ 기능

사용자의 마우스 클릭을 통한 의한 헤더의 전체체크박스의 값 변경시 전체체크 처리 이전에 이벤트가 발생한다.

이벤트의 리턴값을 false로 설정하는 경우 전체체크 처리는 취소 된다.

➤ Syntax

Syntax	function 오브젝트ID_OnBeforeCheckAll(Row, Col) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Row	Number	해당 셀의 Row Index
Col	Number	해당 셀의 Column Index

➤ Example

```
// confirm 을 통한 전체체크 제어
function mySheet_OnBeforeCheckAll(Row, Col) {
    if (!confirm("전체체크를 진행하시겠습니까?")) {
        // 진행 취소 처리
        return false;
    }
}
```

➤ 제공 버전

7.0.1.0	
----------------	--

OnBeforeColumnMove Event

➤ 기능

사용자가 마우스를 이용하여 컬럼 이동을 시작하거나 MoveColumnPos 함수를 이용하여 컬럼 이동을 시작할 때 이벤트가 발생한다. MoveColumnFail(1) 함수를 이용하여 컬럼이동을 방지할 수 있다.

➤ Syntax

Syntax	function 오브젝트ID_OnBeforeColumnMove(Col, NewPos) { }
--------	--

➤ Parameters

Parameter	Type	설 명
Col	Long	이동할 컬럼의 Index
NewPos	Long	이동될 위치의 컬럼 Index

➤ Example

```
//0컬럼이 3컬럼 이후의 위치로 이동하고자 할때 컬럼 이동을 취소한다
function mySheet_OnBeforeColumnMove(Col, NewPos) {
    if(Col==0 && NewPos > 3) {
        mySheet.MoveColumnFail(1);
    }
}
```

➤ 제공 버전

7.0.0.0



OnBeforeDownload Event

➤ 기능

엑셀, 텍스트, Pdf 다운전에 데이터가 만들어지고 폼이 서버밋되기 전에 발생한다. 해당 이벤트 안에서 폼을 서버밋해야 정상적으로 다운이 된다.

➤ Syntax

Syntax	function 오브젝트ID_OnBeforeDownload(downloadType, frm) { }
--------	--

➤ Parameters

Parameter	Type	설 명
downloadType	String	다운로드 종류. "EXCEL" : 엑셀 다운로드 "TEXT" : 텍스트 다운로드 "PDF" : PDF 다운로드
frm	Object	Html Form 객체

➤ Example

```
function mySheet_OnBeforeDownload(type, frm) {  
    //서버측으로 전달될 문자열을 확인  
    var data = frm.Data.value;  
    //다운로드 폼을 서버밋  
    frm.submit();  
}
```

➤ 제공 버전

7.0.13.43	
-----------	--



OnBeforeEdit Event

➤ 기능

셀의 값을 편집하기 직전에 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnBeforeEdit(Row, Col) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Row	Long	해당 셀의 Row Index
Col	Long	해당 셀의 Column Index

➤ Example

```
//컬럼이 이동된 후 이벤트가 발생한다.  
Function mySheet_OnBeforeEdit(Row, Col) {  
    alert("입력을 시작합니다.");  
}
```

➤ 제공 버전

7.0.0.0



OnBeforeExpand Event

➤ 기능

트리 형태일 때 클릭시 트리가 펼쳐지기 전 또는 접히기 전에 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnBeforeExpand (Row, Expand) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Row	Number	해당 셀의 Row Index
Expand	Number	0 : 펼치는 경우 2 : 접는 경우

➤ Example

<pre>function mySheet_OnBeforeExpand(Row, Expand) { if(Row == 1 && Expand == 2) { mySheet.SetAllowExpand(0); alert("1 행에서는 펼칠 수 있지만 접을 수 없습니다."); } }</pre>

➤ 제공 버전

7.0.0.0

OnBeforeMovePage Event

➤ 기능

페이지 인덱스 방식의 조회모드인 경우 페이지 네비게이션 또는 페이지 이동 메소드를 통해 페이지를 이동하는 경우 이동 전에 이벤트가 발생 한다.
해당 이벤트의 반환값을 false로 설정한 경우 페이지 이동이 취소 된다.

➤ Syntax

Syntax	function 오브젝트ID_OnBeforeMovePage(Current, Move) { }
--------	--

➤ Parameters

Parameter	Type	설 명
Current	Number	페이지 이동전의 현재 페이지 번호
Move	Number	이동할 페이지 번호

➤ Example

```
function mySheet_OnBeforeMovePage(current, move) {
    // 트랜잭션이 발생한 경우 페이지 이동을 취소 한다.
    If (mySheet.IsDataModified()) {
        return false;
    }
}
```

➤ 제공 버전

7.0.0.0



OnBeforePaste Event

➤ 기능

시트에 데이터를 붙여넣기 직전에 발생한다

➤ Syntax

Syntax	function 오브젝트ID_OnBeforePaste(text) { }
--------	--

➤ Parameters

Parameter	Type	설 명
text	String	시트에 붙여넣을 텍스트

➤ Remarks

데이터를 붙여넣기 전에 발생하며 필요에 따라 붙여넣기를 취소하거나 붙여넣을 값을 수정 할 수 있다. False 리턴시 붙여넣기는 취소되며, 문자열 리턴시 붙여넣을 텍스트가 대체된다.

➤ Example

```
function mySheet_OnBeforePaste(text) {  
    if(text == "not allow text") {  
        alert("붙여넣기가 취소됩니다.");  
        return false;  
    }  
}  
  
function mySheet_OnBeforePaste(text) {  
    if(text == "not allow text") {  
        return "allow text";  
    }  
}
```

➤ 제공 버전



7.0.0.0



OnBeforeSave Event

➤ 기능

저장 메소드 호출 시 Ajax 통신 직전에 발생한다

➤ Syntax

Syntax	function 오브젝트ID_OnBeforeSave() { }
--------	---

➤ Parameters

Parameter	Type	설 명
없음		

➤ Remarks

DoSave, DoAllSave 메소드 호출 시 Ajax 통신 전에 발생한다.

저장 중 이미지 등을 사용자가 원하는 이미지로 변경하고자 할때 사용한다.

➤ Example

```
function mySheet_OnBeforeSave() {  
    alert("저장중입니다.");  
}
```

➤ 제공 버전

7.0.0.0

OnBeforeSearch Event

➤ 기능

조회 메소드 호출 시 Ajax 통신 직전에 발생한다

➤ Syntax

Syntax	function 오브젝트ID_OnBeforeSearch() { }
--------	---

➤ Parameters

Parameter	Type	설 명
없음		

➤ Remarks

DoSearch, DoSearchChild, DoSearchPaging, DoRowSearch 메소드 호출 시 Ajax 통신 전에 발생한다.

조회 중 이미지 등을 사용자가 원하는 이미지로 변경하고자 할때 사용한다.

➤ Example

<pre>function mySheet_OnBeforeSearch() { alert("조회중입니다."); }</pre>
--

➤ 제공 버전

7.0.0.0



OnButtonClick Event

➤ 기능

"Button" 컬럼타입의 버튼 클릭시 발생하는 이벤트

➤ Syntax

Syntax	function 오브젝트ID_OnButtonClick(Row, Col) { }
--------	--

➤ Parameters

Parameter	Type	설 명
Row	Number	대상 행의 Index
Col	Number	대상 컬럼의 Index

➤ Example

<pre>function mySheet_OnButtonClick(Row, Col) { console.log "[" + Row + "," + Col + "] 셀의 버튼 클릭"); }</pre>
--

➤ 제공 버전

7.0.13.9	
----------	--



OnCellDropEnd Event

➤ 기능

셀 단위 Drag를 시작 후 Drop 시점에 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnCellDropEnd (FromSheet, FromRow, FromCol, ToSheet, ToRow, ToCol, X, Y) { }
--------	--

➤ Parameters

Parameter	Type	설 명
FromSheet	Object	Drag 위치 Sheet 객체
FromRow	Long	Drag 위치 Sheet 객체의 Row Index
FromCol	Long	Drag 위치 Sheet 객체의 Column Index
ToSheet	Object	Drop 위치 Sheet 객체
ToRow	Long	Drop 위치 Sheet 객체의 Row Index
ToCol	Long	Drop 위치 Sheet 객체의 Column Index
X	Long	Drop 위치의 X 좌표
Y	Long	Drop 위치의 Y 좌표

➤ Example

```
// 드래그 셀의 Value를 드랍위치 셀에 설정한다
function mySheet_OnCellDropEnd(FromSheet, FromRow, FromCol, ToSheet, ToRow, ToCol) {
    var bValue = ToObj.GetCellValue(ToRow, ToCol);
    var aValue = Obj.GetCellValue(Row, Col);
    if (ToRow < 0) {
        ToRow = ToObj.DataInsert(ToRow);
        ToCol = ToObj.MouseCol();
    }
    if (ToObj && ToRow > 0 && ToCol >= 0) {
```



```
ToObj.SetCellValue(ToRow, ToCol, aValue);  
if (bValue) {  
    Obj.SetCellValue(Row, Col, bValue);  
} else {  
    Obj.SetCellValue(Row, Col, "");  
}  
}  
}
```

➤ **제공 버전**

7.0.0.0

OnChange Event

➤ 기능

셀의 값을 편집 완료하고, 기존 값이 변경되었을 때 이벤트가 발생한다.
사용자의 값 변경이외에도 CellValue 함수를 이용하거나 헤더 영역의 전체 선택 CheckBox가 변경되었을 때 또는 CheckAll 함수를 이용한 경우 각 데이터 행 단위로 변경된 CheckBox에 대해서 이벤트가 발생한다. CellValue 함수의 flag 인자의 값이 0 일 경우는 이 이벤트가 발생하지 않는다.

➤ Syntax

Syntax	function 오브젝트ID_OnChange(Row, Col, Value, OldValue, RaiseFlag) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Row	Long	해당 셀의 Row Index
Col	Long	해당 셀의 Column Index
Value	String	변경된 값, Format이 적용되지 않은 저장 시 사용되는 값
OldValue	String	변경 이전의 값
RaiseFlag	Integer	이벤트 발생 옵션 (0:사용자 편집, 1:메소드, 2:Paste(ctrl+v 붙여넣기))

➤ Example

```
function mySheet_OnChange(Row, Col, Value, OldValue, RaiseFlag) {
    if(Col == 3 && mySheet.GetCellValue(Row, 2) == '원화' && Value == '9') {
        alert("원화 일 때 코드는 10을 선택하여야 합니다.");
    }
}
```



- 제공 버전
7.0.0.0

OnChangeFilter Event

➤ 기능

필터행의 셀의 값을 바꾸거나 옵션이 변경되었을 때 이벤트가 발생한다.

사용자의 값 변경외에도, SetFilterOption 함수를 이용했을때도 이벤트가 발생하게 된다.

➤ Syntax

Syntax	function 오브젝트ID_OnChangeFilter() { }
--------	---

➤ Parameters

Parameter	Type	설 명
		None

➤ Example

```
//SearchMode:3 일때 필터링 조건이 바뀌게 되면 조회
function mySheet_OnChangeFilter() {
    // 필터행 QueryString 변환
    var fp = mySheet.GetFilterParam(0,1);
    var info = {PageParam: "page", Param: "id=ibleaders&seq=1&" + fp};
    mySheet.DoSearchPaging("list.jsp",info);
}
```

➤ 제공 버전

7.0.0.0



OnChangeSum Event

➤ 기능

합계행에 값이 바뀌었을 때 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnChangeSum(Row,Col) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Row	Long	합계행의 상단 Row Index
Col	Long	변경 셀의 Col Index

➤ Example

```
function mySheet_OnChangeSum(Row, Col) {  
    //합계 행에 값이 바뀌었을 때, 같은 행의 다른셀에 계산 정보 표시  
    mySheet.SetSumText(0,2, mySheet.GetSumValue(Col) / 100 + " %");  
}
```

➤ 제공 버전

7.0.0.0

OnCheckAllEnd Event

➤ 기능

체크박스 타입 컬럼이 전체체크가 수행된 후 완료시점에 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnCheckAllEnd(Col, Value) { }
--------	--

➤ Parameters

Parameter	Type	설 명
Col	Integer	해당 컬럼의 Index
Value	Boolean	체크여부

➤ Example

<pre>function mySheet_OnCheckAllEnd(Col, Value) { alert(Col + "번째 컬럼의 전체체크 수행이 완료되었습니다."); }</pre>
--

➤ 제공 버전

7.0.0.0



OnClick Event

➤ 기능

데이터 영역의 셀을 마우스로 클릭했을 때 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnClick(Row, Col, Value, CellX, CellY, CellW, CellH) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Row	Long	해당 셀의 Row Index
Col	Long	해당 셀의 Column Index
Value	String	이벤트가 발생한 셀의 값
CellX	Long	해당셀의 X좌표
CellY	Long	해당셀의 Y좌표
CellW	Long	해당셀의 가로 넓이값
CellH	Long	해당셀의 세로 높이값

➤ Example

```
function mySheet_OnClick(Row, Col, Value, CellX, CellY, CellW, CellH) {  
    //특정 열을 클릭했을 때 다른 페이지로 이동하도록 처리  
    if( mySheet.ColSaveName(Col) == "LINK_BUFF"){  
        location.href = "link.jsp?key=" + mySheet.GetCellValue(Row,"KFELD");  
    }  
}
```

➤ 제공 버전

7.0.0.0

OnDbClick Event

➤ 기능

데이터 영역의 셀을 마우스로 더블 클릭했을 때 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnDbClick(Row, Col, Value, CellX, CellY, CellW, CellH) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Row	Long	해당 셀의 Row Index
Col	Long	해당 셀의 Column Index
Value	String	해당 셀의 Value
CellX	Long	해당셀의 X좌표
CellY	Long	해당셀의 Y좌표
CellW	Long	해당셀의 가로 넓이값
CellH	Long	해당셀의 세로 높이값

➤ Example

<pre>function mySheet_OnDbClick(Row, Col, Value, CellX, CellY, CellW, CellH) { //더블 클릭했을 때, 선택행의 값을 전달 어미 창으로 전달. var selectRowJson = mySheet.GetRowData(Row); parent.GetData(selectRowJson); self.close(); return false; //창을 닫는 경우에는 false를 리턴해 줘야 함. }</pre>

➤ 제공 버전

7.0.0.0



OnDebugMsg Event

➤ 기능

모든 기능 처리 과정 중 발생하는 디버깅용 메시지가 있을 때 이벤트가 발생한다.

ShowDebugMsg 속성을 0으로 설정하는 경우 디버깅용 메시지가 이 이벤트를 통해서 호출되고, -1로 설정하는 경우 사용자가 볼 수 있도록 메시지 팝업 형태로 표시된다.

➤ Syntax

Syntax	function 오브젝트ID_OnDebugMsg(Msg) { }
--------	--

➤ Parameters

Parameter	Type	설 명
Msg	String	디버그 메시지

➤ Example

```
function mySheet_OnDebugMsg(Msg) {  
    txtErr.value = txtErr.value + « \n>>> » + Msg ;  
}  
  
//디버그 메시지 표시를 위한 TextArea 생성  
<textarea name="txtErr" rows=10 cols=70></textarea>
```

➤ 제공 버전

7.0.0.0

OnDecryption Event

➤ 기능

조회를 하는 과정에서 각 셀에 들어가야 할 데이터를 수정하거나 암호화 모듈과 연동하여 사용하기 위해 각 셀과 EtcData 요소에 대해 발생하는 Event

➤ Syntax

Syntax	function 오브젝트ID_OnDecryption(Row,Col,SaveName,Value){}
--------	---

➤ Parameters

Parameter	Type	설 명
Row	Long	셀의 Row Index(EtcData에 대해서는 -1)
Col	Long	셀의 Col Index(EtcData에 대해서는 -1)
SaveName	String	컬럼의 SaveName이나 EtcData 의 KEY
Value	String	셀에 들어갈 데이터 나 EtcData의 Value

➤ Remarks

데이터를 조회하는 과정에서 값이 A 인 경우, B라고 표시해야 된다면, 혹은 데이터가 암호화 되어 내려와 이를 표현하기 전에 복호화 해야 하는 경우 이 이벤트를 통해 작업할 수 있다.

자세한 사용 예는 아래 소스를 참조

➤ Example

```
function mySheet_OnDecryption(Row, Col, SaveName, Value){
/* 값을 매개변수로 복호화를 수행하는 Dec 사용자 함수를 호출하여 리턴한다*/.
    Return Dec(Value);
}
```

➤ 제공 버전

7.0.0.0



OnDownFinish Event

➤ 기능

엑셀 및 텍스트 파일이 다운로드 완료 되었을때 이벤트가 발생한다.

.

➤ Syntax

Syntax	function 오브젝트ID_OnDownFinish(downloadType, result){
--------	--

➤ Parameters

Parameter	Type	설 명
downloadType	String	엑셀/텍스트 종류. "EXCEL", "TEXT"
result	Boolean	다운로드 오류 여부. 성공 : true, 실패 : false

➤ Remarks

➤ Example

```
function mySheet_OnDownFinish(downloadType, result) {  
    alert(downloadType + ' 다운이 완료되었습니다. 다운로드 결과 : ' +  
    result);  
}
```

➤ 제공 버전

7.0.0.0



OnDragStart Event

➤ 기능

행 또는 셀 단위 Drag를 시작할 때 이벤트가 발생 한다.

➤ Syntax

Syntax	function 오브젝트ID_OnDragStart (Row, Col) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Row	Long	Drag 위치의 Row Index
Col	Long	Drag 위치의 Column Index

➤ Example

```
var dragValue = "";  
function mySheet_OnDragStart(Row, Col) {  
    // Drag 시작위치의 CellValue를 저장  
    dragValue = mySheet.GetCellValue(Row, Col) ;  
}
```

➤ 제공 버전

7.0.0.0



OnDropEnd Event

➤ 기능

행 단위 Drag를 시작 후 Drop 시점에 이벤트가 발생한다.

이벤트 인자중 Type 인자는 트리구조에서 Drop 위치의 상세 값을 전달 한다.

Type 인자의 값은 아래와 같다

Type	설 명
1	Drop 위치 행의 상단
2	Drop 위치 행의 중앙
3	Drop 위치 행의 하단

➤ Syntax

Syntax	function 오브젝트ID_OnDropEnd (FromSheet, FromRow, ToSheet, ToRow, X, Y, Type) { }
--------	---

➤ Parameters

Parameter	Type	설 명
FromSheet	Object	Drag 위치 Sheet 객체
Row	Long	Drag 위치 Sheet 객체의 Row Index
ToSheet	Object	Drop 위치 Sheet 객체
ToRow	Long	Drop 위치 Sheet 객체의 Row Index
X	Integer	Drop 위치의 X 좌표
Y	Integer	Drop 위치의 Y 좌표
Type	Integer	트리 구조인 경우 Drop 위치의 타입

➤ Example

```
// 드래그 행을 드랍위치에 추가하고 드래그 시트에서 삭제한다.
Function mySheet_OnDropEnd(FromSheet, FromRow, ToSheet, ToRow, X, Y, Type) {
    var NewRow = ToObj.DataInsert(ToRow);
```



```
for (var c = 0; c <= Obj.LastCol(); c++) {  
    ToObj.SetCellValue(NewRow, c, Obj.GetCellValue(Row, c));  
}  
Obj.RowDelete(Row);  
}
```

➤ **제공 버전**

7.0.0.0



OnEditValidation Event

➤ 기능

셀의 편집완료 시점에 수정된 값에 대해 Validation Check할수 있도록 이벤트가 발생한다.

Validation에 맞지 않다면 ValidateFail(1)로 설정하여 셀 편집이전의 값으로 되돌린다.

➤ Syntax

Syntax	function 오브젝트ID_OnEditValidation(Row, Col, Value) { }
--------	--

➤ Parameters

Parameter	Type	설 명
Row	Long	해당 셀의 Row Index
Col	Long	해당 셀의 Column Index
Value	variant	Format이 적용되지 않은 저장 시 사용되는 값

➤ Example

```
function mySheet_OnEditValidation(Row, Col, Value) {  
    if(mySheet.ColSaveName(Col)=="ITMS_CD"){  
        if(Value.indexOf("현행_")>-1){  
            alert("현행 정보값은 4/4분기 이후에 설정 가능합니다.");  
            //값을 이전 값으로 되돌림  
            mySheet.ValidateFail(1);  
        }  
    }  
}
```

➤ 제공 버전

7.0.0.0

OnEncryption Event

➤ 기능

시트의 데이터를 저장할 때 변경된 내용들을 QueryString으로 묶게 되는데, 이 때 암호화가 필요하거나 값을 변경하여 저장해야 하는 경우 사용하는 Event.

➤ Syntax

Syntax	function 오브젝트ID_OnEncryption(Row, Col, Value) { }
--------	--

➤ Parameters

Parameter	Type	Remark
Row	Long	셀의 Row Index
Col	Long	셀의 Col Index
SaveName	String	컬럼의 SaveName
Value	String	셀의 현재 데이터

➤ Remarks

데이터를 저장할때, 유효성의 검증이 필요하다면 OnValidation 이벤트를 통해 저장할지 여부를 결정할 수 있다. 하지만 무조건 저장을 한다는 가정하에 만약 특정 값이 들어있으면 값을 바꾸어 저장해야 한다면 이 이벤트를 통하여 가능하다. 자세한 사용 예는 아래 소스를 참조

➤ Example

```
function mySheet_OnEncryption(Row, Col, SaveName, Value){
/* 저장할 값을 매개변수로 암호화를 수행하는 Enc 사용자 함수를 호출하여
리턴한다*/.
    Return Enc(Value);
}
```

➤ 제공 버전

7.0.0.0



OnExportEncryption Event

➤ 기능

시트의 데이터를 엑셀 또는 텍스트 파일로 변환시 각 셀에 대한 데이터 수집 시 발생하는 이벤트 이다. 해당 셀의 데이터에 대한 암호화 또는 마스킹 처리를 하는 경우 이 이벤트를 이용하여 처리 한다. 이 이벤트에서 반환값이 있는 경우 해당 셀의 데이터는 반환한 값으로 치환되어 처리 된다.

** 해당 이벤트는 PDF 문서변환은 지원하지 않는다.*

➤ Syntax

Syntax	function 오브젝트ID_OnExportEncryption(Row, Col, SaveName, Value) { }
--------	--

➤ Parameters

Parameter	Type	Remark
Row	Long	셀의 Row Index
Col	Long	셀의 Col Index
SaveName	String	컬럼의 SaveName
Value	String	셀의 현재 데이터

➤ Example

```
function mySheet_OnExportEncryption(Row, Col, SaveName, Value){  
    var val = Value;  
    // SaveName 이 sName 인 경우 마지막 문자열을 *로 치환한다.  
    If (SaveName == "sName") {  
        val = val.substring(0, val.length - 1) + "*";  
        return val;  
    }  
}
```




➤ 제공 버전

7.0.13.2	
----------	--



OnFilterEnd Event

➤ 기능

필터링이 완료된 후에 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_ OnFilterEnd (RowCnt, FirstRow) { }
--------	---

➤ Parameters

Parameter	Type	설 명
RowCnt	Long	필터링된 이후 행의 개수
FirstRow	Long	필터링된 이후 첫번째 행의 Index

➤ Example

<pre>function mySheet_ OnFilterEnd(RowCnt, FirstRow) { } }</pre>
--

➤ 제공 버전

7.0.0.0



OnGroupFinish Event

➤ 기능

그룹핑 처리가 완료된 시점에 이 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_ OnGroupFinish(Group) { }
--------	--

➤ Parameters

Parameter	Type	설 명
Group	String	그룹 기준 컬럼의 SaveName을 구분자 " "로 연결한 문자열

➤ Example

<pre>function mySheet_OnGroupFinish(group) { console.log("그룹기준 컬럼 : ", group.split(" ").join(", ")); }</pre>
--

➤ 제공 버전

7.0.12.1	
----------	--



OnHScroll Event

➤ 기능

가로스크롤 시 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnHScroll(hpos, oldhpos, isLeft, isRight, section) { }
--------	---

➤ Parameters

Parameter	Type	설 명
hpos	Long	가로 스크롤 값
oldhpos	Long	이전 가로 스크롤 값
isLeft	Boolean	가로 스크롤이 좌측 끝에 위치했는지 여부
isRight	Boolean	가로 스크롤이 우측 끝에 위치했는지 여부
section	Int	스크롤 섹션값 (항상 1이 반환됨)

➤ Example

<pre>function mySheet_OnHScroll(hpos, oldhpos, isLeft, isRight, section) { }</pre>
--

➤ 제공 버전

7.0.0.0

OnKeyDown Event

➤ 기능

셀의 값을 수정 중이거나 선택된 셀에 키보드를 누른 경우 이벤트가 발생한다.
KeyCode는 아스키 값이므로 변환하여 사용한다.

➤ Syntax

Syntax	function 오브젝트ID_OnKeyDown (Row, Col, KeyCode, Shift) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Row	Long	해당 셀의 Row Index
Col	Long	해당 셀의 Column Index
KeyCode	Integer	키보드의 아스키 값
Shift	Integer	1 : Shift키가 눌린 경우 2 : Ctrl키가 눌린 경우 0 : 그외

➤ Example

```
function mySheet_OnKeyDown(Row, Col, KeyCode, Shift) {
    //마지막 컬럼에서 엔터키가 눌린 경우 다음행의 처음에 포커스 두기
    if(KeyCode == 13 && Col == mySheet.LastCol()) {
        mySheet.SelectCell(Row + 1, 2);
    }
}
```

➤ 제공 버전

7.0.0.0



OnKeyUp Event

➤ 기능

셀의 값을 수정 중이거나 선택된 셀에 눌린 키보드가 올라오는 경우 이벤트가 발생한다. 이 이벤트는 OnKeyDown 이벤트 발생 후에 바로 발생한다.
KeyCode는 아스키 값이므로 변환하여 사용한다.

➤ Syntax

Syntax	function 오브젝트ID_OnKeyUp (Row, Col, KeyCode, Shift) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Row	Long	해당 셀의 Row Index
Col	Long	해당 셀의 Column Index
KeyCode	Integer	키보드의 아스키 값
Shift	Integer	1 : Shift키가 눌린 경우 2 : Ctrl키가 눌린 경우 0 : 그외

➤ Example

```
function mySheet_OnKeyUp(Row,Col,KeyCode,Shift) {  
    //마지막 컬럼에서 엔터키가 눌린 경우 다음 행의 처음에 포커스 두기  
    if(KeyCode ==13 && Col == mySheet.LastCol()  
        && Row < mySheet.RowCount()) {  
        mySheet.SelectCell(Row + 1, 2);  
    }  
}
```

➤ 제공 버전

7.0.0.0

OnLoad Event

➤ 기능

createIBSheet 또는 Reset 메소드 호출시 시트의 객체가 생성 완료시점에 이 이벤트가 발생 한다.

이 이벤트는 body의 onload 이전에 발생하며 DOM 엘리먼트 생성이전 시점이기 때문에 시트 객체의 메소드 확장 또는 재정의 목적으로 사용하여야 하며 초기화는 이 이벤트내에서 처리가 불가능 하다. (body 의 onload 이후 시점에 서만 가능)

➤ Syntax

Syntax	function 오브젝트ID_OnLoad() { }
--------	-------------------------------------

➤ Parameters

Parameter	Type	설 명
		없음

➤ Example

```
function mySheet_OnLoad() {
    // 시트 객체에 CusomMethod 메소드를 추가
    mySheet.CustomMethod = function(arg1, arg2) {
    }
}
```

➤ 제공 버전

7.0.3.0	
----------------	--



OnLoadData Event

➤ 기능

데이터 조회 관련 메소드 및 저장 메소드 호출시 서버로부터 데이터를 받은 후 시트로 로드할때 발생한다.

이 이벤트는 서버로부터 데이터를 받은 후 데이터를 수정 하거나 암호화 모듈과 연동하는 경우에 사용한다.

➤ Syntax

Syntax	function 오브젝트ID_OnLoadData(Data) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Data	String	조회 XML 또는 JSON 문자열/객체

➤ Example

```
function mySheet_OnLoadData(data) {  
    // 복호화 처리  
    var decrypt_data = fnDecryption(data);  
  
    // 복호화된 데이터 리턴  
    return decrypt_data;  
}
```

➤ 제공 버전

7.0.0.0

OnLoadExcel Event

➤ 기능

LoadExcel 처리가 완료된 후에 이벤트가 발생한다.

Code, Msg 인자는 실패의 경우에만 해당 값이 전달 된다.

➤ Syntax

Syntax	function 오브젝트ID_OnLoadExcel(result, code, msg) { }
--------	---

➤ Parameters

Parameter	Type	설 명
result	Boolean	로드한 결과. 성공 : true, 실패 : false
Code	Number	결과 코드
Msg	String	결과 메시지

➤ Example

<pre>function mySheet_OnLoadExcel(result) { if(result) { alert('엑셀 로딩이 완료되었습니다.');</pre>
<pre> } else { alert('엑셀 로딩중 오류가 발생하였습니다.');</pre>
<pre> } }</pre>

➤ 제공 버전

7.0.0.0	
7.0.13.13	Code, Msg 인자 추가



OnLoadText Event

➤ 기능

LoadText 처리가 완료된 후에 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnLoadText(result) { }
--------	---

➤ Parameters

Parameter	Type	설 명
result	Boolean	로드한 결과. 성공 : true, 실패 : false

➤ Example

```
function mySheet_OnLoadText(result) {  
    if(result) {  
        alert('텍스트 파일 로딩이 완료되었습니다.');    } else {  
        alert('텍스트 파일 로딩중 오류가 발생하였습니다.');    }  
}
```

➤ 제공 버전

7.0.0.0

OnMessage Event

➤ 기능

SetShowMsgMode(0)인 경우 각종 확인 메시지 또는 경고 메시지가 발생했을 때 시스템 팝업이 표시되는 것이 아니라 이 이벤트가 발생한다. 확인 메시지인 경우는 IsConfirm = 1이므로, 반드시 ConfirmOK 함수를 이용하여 응답 결과를 Sheet로 되돌려 준다.

➤ Syntax

Syntax	function 오브젝트ID_OnMessage (Msg, Level, IsConfirm) { }
--------	--

➤ Parameters

Parameter	Type	설 명
Msg	String	메시지
Level	String	메시지 레벨 코드 "U" – EndUser를 위한 메시지 "E" – 개발자를 위한 메시지 "D" – 서버 연결 함수의 페이지 관련 메시지 "X" – 조회, 저장XML의 XML파서 메시지
IsConfirm	Boolean	확인 메시지 여부

➤ Example

```
//메시지 모드를 설정한다.
mySheet.SetShowMsgMode(0);

//OnMessage 이벤트를 처리한다.
Function mySheet_OnMessage(Msg, Level, IsConfirm) {
    //메시지 표시하기
    var win_result = window.showModalDialog(
        "sheet_message.jsp?Msg=" + Msg + "&IsConfirm=" + IsConfirm,
        'modalResult',
```



```
'dialogWidth:200px;dialogHeight:200px;center:yes;help:no;status:no;');  
//Sheet로 메시지 결과를 반환한다.  
If(IsConfirm) mySheet.ConfirmOK(win_result);  
}
```

➤ **제공 버전**

7.0.0.0



OnMouseDown Event

➤ 기능

마우스가 눌러졌을 때 이벤트가 발생한다.

마우스가 눌린 셀의 위치를 알고자 한다면 MouseRow 와 MouseCol 함수를 이용한다.

➤ Syntax

Syntax	function 오브젝트ID_OnMouseDown(Button, Shift, X, Y) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Button	Integer	0 : 왼쪽, 2 : 오른쪽 (마우스버튼 방향)
Shift	Integer	1 : Shift키가 눌린 경우 2 : Ctrl키가 눌린 경우 0 : 그외
X	Long	X좌표
Y	Long	Y좌표

➤ Example

<pre>function mySheet_OnMouseDown(Button, Shift, X, Y) { //눌린 컬럼 확인 alert(mySheet.MouseRow() + "행 " + mySheet.MouseCol() + "컬 럼이 눌림"); }</pre>
--

➤ 제공 버전

7.0.0.0



OnMouseMove Event

➤ 기능

마우스가 Sheet 위에서 이동되었을 때 이벤트가 발생한다.

마우스가 이동하고있는 셀의 위치를 알고자 한다면 MouseRow와 MouseCol 함수를 이용한다.

➤ Syntax

Syntax	function 오브젝트ID_OnMouseMove (Button, Shift, X, Y) { }
--------	--

➤ Parameters

Parameter	Type	설 명
Button	Integer	0 : 왼쪽, 2 : 오른쪽 (마우스버튼 방향)
Shift	Integer	1 : Shift키가 눌린 경우 2 : Ctrl키가 눌린 경우 0 : 그외
X	Long	X좌표
Y	Long	Y좌표

➤ Example

<pre>function mySheet_OnMouseMove(Button, Shift, X, Y) { //마우스 위치를 행과 컬럼과 값 가져오기 var Row = mySheet.MouseRow(); var Col = mySheet.MouseCol(); var sText = mySheet.GetCellText(Row, Col); //2컬럼이고 내용이 2011-07-14일때만 div popup을 보여줌 if(Col == 2 && sText == "2011-07-14") { \$("#div_pop_summary").show(); } else { \$("#div_pop_summary").hide(); } }</pre>
--



```
}  
}
```

- 제공 버전
7.0.0.0



OnMouseUp Event

➤ 기능

눌려진 마우스 버튼이 올라올 때 이벤트가 발생한다.

버튼이 올라온 셀의 위치를 알고자 한다면 MouseRow와 MouseCol 함수를 이용한다.

➤ Syntax

Syntax	function 오브젝트ID_OnMouseUp(Button, Shift, X, Y) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Button	Integer	0 : 왼쪽, 2 : 오른쪽 (마우스버튼 방향)
Shift	Integer	1 : Shift키가 눌린 경우 2 : Ctrl키가 눌린 경우 0 : 그외
X	Long	X좌표
Y	Long	Y좌표

➤ Example

```
function mySheet_OnMouseUp(Button, Shift, X, Y) {  
    //마우스 버튼이 올라온 컬럼 확인  
    alert(mySheet.MouseRow() + "행 " + mySheet.MouseCol() + "컬럼이 눌  
림");  
}
```

➤ 제공 버전

7.0.0.0



OnPageRequest Event

➤ 기능

서버페이징 조회시 스크롤이 이동하여 해당 페이지를 새로 로딩하기 위하여 서버에 해당 페이지를 요청할때 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnPageRequest(page) { }
--------	--

➤ Parameters

Parameter	Type	설 명
page	Integer	요청할 페이지 번호

➤ Example

```
function mySheet_OnPageRequest(page) {  
    //서버에 해당 페이지를 요청하기 직전에 경고창을 띄운다.  
    Alert(page + " 페이지를 서버에 요청합니다.");  
}
```

➤ 제공 버전

7.0.0.0



OnPopupClick Event

➤ 기능

셀의 Type이 Popup이거나 PopupButton 을 사용하는 셀인 경우 해당 셀에 포커스가 갔을 때 보이는 팝업 버튼을 마우스로 클릭하거나 Edit하려고 할 때 이 이벤트가 발생한다.

사용자는 이 이벤트에서 팝업 호출 로직을 기재하여 팝업 처리하도록 한다.

➤ Syntax

Syntax	function 오브젝트ID_OnPopupClick(Row, Col) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Row	Long	해당 셀의 Row Index
Col	Long	해당 셀의 Column Index

➤ Example

```
function mySheet_OnPopupClick(Row,Col) {  
    //팝업을 연다.  
    Window.open("Sheet_Popup.jsp?row="+Row+"&col="+Col, "list",  
        "scrollbars=no,fullscreen=no,width=250,height=350");  
}
```

➤ 제공 버전

7.0.0.0

OnResize Event

➤ 기능

너비를 %로 설정시, IBSheet의 너비 또는 높이가 변경된 경우 이벤트가 발생한다.

이 이벤트는 사용자가 드래그를 통해 창의 사이즈를 조절하는 경우 굉장히 빈번하게 발생하여 브라우저에 부담을 줄수 있다. 따라서 가급적이면 이 이벤트 보다는 OnSmartEvent를 사용할 것을 권장한다.

➤ Syntax

Syntax	function 오브젝트ID_OnResize (Width, Height) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Width	Integer	전체 너비
Height	Integer	전체 높이

➤ Example

```
function mySheet_OnResize(Width, Height) {
    //변경된 정보에 따라 컬럼들의 너비를 재조정한다.
    mySheet.FitColWidth();
}
```

➤ 제공 버전

7.0.0.0



OnRowSearchEnd Event

➤ 기능

DoSearch나 LoadSearchData 메서드를 통해 데이터가 조회되는 과정에서 행단위로 발생한다.

특정 행의 데이터를 기준으로 판단하여 Cell에 색상이나 배경색등을 표현하고자 하는 경우 유용하게 쓰일수 있다.

이 기능은 조회 중에 행단위로 한번씩 발생함으로 이벤트 안에 로직이 복잡하거나 loop문이 들어가는 경우 조회속도가 저하될 수 있다.

➤ Syntax

Syntax	function 오브젝트ID_OnRowSearchEnd (row) { }
--------	---

➤ Parameters

Parameter	Type	설 명
row	Integer	행의 인덱스

➤ Example

```
function mySheet_OnRowSearchEnd(row) {  
    //3번컬럼이 check되어있고, 4번 컬럼의 값이 100보다 클때, 6번 컬럼의  
    글자 색상을 붉은색으로 변경한다.  
    If( mySheet.GetCellValue(row,3) == 1 && mySheet.GetCellValue(row,4) >  
    100){  
        mySheet.SetCellFontColor(row ,6 ,"#FF0000");  
    }  
}
```

➤ 제공 버전

7.0.0.0

OnSaveEnd Event

➤ 기능

저장 함수를 이용하여 저장 처리를 완료하고, 내부적인 처리를 마친 후 이벤트가 발생한다.

저장 중 오류 메시지가 발생한 경우 이벤트 인자인 code로 설정되므로, code가 0보다 작은 경우 오류 처리 로직을 작성하도록 한다.

DoSave 함수 또는 DoAllSave 함수를 호출할 때 이 이벤트가 발생한다.

Response 인자는 서버에서 정상적으로 처리 되지 않은 경우에만 전달된다.
(HTTP 응답코드가 100 보다 작거나 400보다 큰 경우에만 전달)

➤ Syntax

Syntax	function 오브젝트ID_OnSaveEnd(Code, Msg, StCode, StMsg, [Response]) { }
--------	--

➤ Parameters

Parameter	Type	설 명
Code	Long	처리결과 코드(0이상이면 정상, 그외는 오류 처리 Sheet ErrCode -3 : 요청 Url이 잘못된 경우나 네트워크 오류 등으로 결과를 받지 못한 경우 -4 : 정상적인 응답은 받았으나 데이터 파싱처리중 오류가 발생한 경우(대부분 데이터 이상) -5 : 응답 결과가 빈값인 경우 -6 : 서버에 연결하여 응답을 대기하는 시간이 초과된 경우(WaitTimeOut 초과) 그외 : 사용자 정의 코드)
Msg	String	처리결과 메시지
StCode	Integer	HTTP 응답 코드



StMsg	String	HTTP 응답 메시지
Response	String	서버 응답 결과

➤ **Example**

```
function mySheet_OnSaveEnd(code, msg) {  
    if(code >= 0) {  
        alert(msg); // 저장 성공 메시지  
        mySheet.DoSearch("list1.jsp");  
    } else {  
        alert(msg); // 저장 실패 메시지  
    }  
}
```

```
<?xml version='1.0' ?>
```

```
<SHEET><Result Code="-1" Message= „전표 처리 완료 되어 수정할 수 없  
습니다.“/></SHEET>
```

```
<?xml version='1.0' ?>
```

```
<SHEET><Result Code = "0" Message = "" / ></SHEET>
```

➤ **제공 버전**

7.0.0.0	
7.0.13.30	Response 인자 추가

OnSearchEnd Event

➤ 기능

조회 함수를 이용하여 조회 처리를 완료하고, 내부적인 데이터 표시 처리를 마친 후 이벤트가 발생한다.

조회중 오류 메시지가 발생한 경우 이벤트 인자인 code로 설정되므로, code가 0보다 작은 경우 오류 처리로직을 작성하도록 한다.

Response 인자는 서버에서 정상적으로 처리 되지 않은 경우에만 전달된다.
(HTTP 응답코드가 100 보다 작거나 400보다 큰 경우에만 전달)

➤ Syntax

Syntax	function 오브젝트ID_OnSearchEnd(Code, Msg, StCode, StMsg, [Response]) { }
--------	--

➤ Parameters

Parameter	Type	설 명
Code	Long	처리결과 코드(0이면 정상, 그외는 오류 처리 Sheet ErrCode -3 : 요청 Url이 잘못된 경우나 네트워크 오류 등으로 결과를 받지 못한 경우 -4 : 정상적인 응답은 받았으나 데이터 파싱처리중 오류가 발생한 경우(대부분 데이터 이상) -5 : 응답 결과가 빈값인 경우 -6 : 서버에 연결하여 응답을 대기하는 시간이 초과된 경우(WaitTimeOut 초과) 그외 : 사용자 정의 코드)
Msg	String	처리결과 메시지
StCode	Integer	HTTP 응답 코드
StMsg	String	HTTP 응답 메시지
Response	String	서버 응답 결과



➤ **Example**

```
function mySheet_OnSearchEnd(code, message) {  
    if(code == 0) {  
        alert(message);  
        //조회 후 작업 수행  
    } else {  
        alert("조회 중에 오류가 발생하였습니다..");  
    }  
}
```

➤ **제공 버전**

7.0.0.0	
7.0.13.30	Response 인자 추가



OnSelectMenu Event

➤ 기능

컬럼 팝업 메뉴가 아닌 ActionMenu에 설정된 메뉴가 표시되고, 사용자가 특정한 메뉴 항목을 선택할 때 이벤트가 발생한다.

이벤트에서 반환하는 Text는 메뉴항목에 표시된 글자와 동일하므로 각각 해당하는 기능을 처리한다.

Col 인자는 컬럼의 ActionMenu 선택시에만 사용 할 수 있다.

➤ Syntax

Syntax	function 오브젝트ID_OnSelectMenu(Text, Code, Col) { }
--------	--

➤ Parameters

Parameter	Type	설 명
Text	String	선택된 메뉴 문자열
Code	String	선택된 메뉴 코드 문자열
Col	Number	대상 컬럼의 Index

➤ Example

```
// 메뉴설정
mySheet.SetActionMenu("입력|행복사|-|행삭제|Clear|엑셀다운로드");

function mySheet_ OnSelectMenu(Text, Code) {
    // text 또는 code값으로 Action수행
    switch(text) {
        case "첫행입력" :
            mySheet.DataInsert(0);
            break;

        case "마지막행입력" :
            mySheet.DataInsert(-1);
```



```
        break;

        case "입력" :
            mySheet.DataInsert();
            break;

        case "행복사":
            mySheet.DataCopy();
            break;

        case "행삭제":
            mySheet.RowDelete();
            break;

        case "Clear": //RemoveAll
            mySheet.RemoveAll();
            break;

        case "엑셀다운로드":
            mySheet.Down2Excel();
            break;

        Default=
            break;

    }
}
```

➤ 제공 버전

7.0.0.0	
7.0.13.16	Col 인자 추가

OnSelectCell Event

➤ 기능

특정 셀이 선택되었을 때 이벤트가 발생한다.

기존에 선택되었던 셀 정보와 새로 선택된 셀 정보를 모두 반환하므로 선택된 셀에 대한 처리 로직을 기재할 수 있다.

➤ Syntax

Syntax	function 오브젝트ID_OnSelectCell (OldRow, OldCol, NewRow, NewCol,isDelete) { }
--------	---

➤ Parameters

Parameter	Type	설 명
OldRow	Long	이전 선택 셀의 Row Index
OldCol	Long	이전 선택 셀의 Column Index
NewRow	Long	현재 선택 셀의 Row Index
NewCol	Long	현재 선택 셀의 Column Index
isDelete	Boolean	추가된 행을 DelCheck를 통해 삭제하는 경우, 같은 위치를 기준으로 OnSelectCell 이벤트가 발생하는데, 이때 행삭제됨을 구분해 주기 위한 값

➤ Example

<pre>function mySheet_OnSelectCell(OldRow, OldCol, NewRow, NewCol,isDelete) { alert("셀(" + OldRow + "," + OldCol + ")을 선택하였다가 셀(" + NewRow + "," + NewCol + ")을 선택하였습니다."); }</pre>
--

➤ 제공 버전

7.0.0.0



OnSelectEnd Event

➤ 기능

마우스 드래그 또는 키 입력으로 셀 영역 선택 완료 시점에 이벤트가 발생 한다.

➤ Syntax

Syntax	function 오브젝트ID_OnSelectEnd(Rows, Cols) { }
--------	--

➤ Parameters

Parameter	Type	설 명
Rows	String	선택된 행의 Index를 구분자 ' '로 연결한 문자열
Cols	String	선택된 컬럼의 Index를 구분자 ' '로 연결한 문자열

➤ Example

<pre>function mySheet_OnSelectEnd(Rows, Cols) { console.log("SelectedRows:", Rows, "SelectedCols", Cols); }</pre>

➤ 제공 버전

7.0.13.26	
------------------	--

OnSmartResize Event

➤ 기능

시트의 너비 또는 높이가 최초 변경 시점부터 일정간격(300ms)동안 재발생이 없는 시점에 이 이벤트가 발생한다. Resize시점에 FitColWidth등의 처리를 하는 경우 OnResize 이벤트가 아닌 이 이벤트에서 처리하는 것이 성능 개선에 도움이 된다.

이 이벤트는 전역 또는 SetConfig 메소드에서 "SmartResize" 속성을 0(false)로 설정한 경우에만 발생하며, "SmartResize" 속성을 1(true)로 설정한 경우에는 발생하지 않고, OnResize 이벤트가 위의 설명처럼 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnSmartResize(Width, Height) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Width	Integer	전체 너비
Height	Integer	전체 높이

➤ Example

```
function mySheet_OnSmartResize(Width, Height) {
    //변경된 정보에 따라 컬럼들의 너비를 재조정한다.
    mySheet.FitColWidth();
}
```

➤ 제공 버전

7.0.0.0



OnSort Event

➤ 기능

헤더를 마우스로 눌러 데이터가 소트 완료 되었을 때 이벤트가 발생한다. Col 인자는 컬럼 Index 번호이고, SortOrder는 소트 방향으로 아래와 같이 표현된다. SortEventMode 가 1로 설정되어 있으면 소트되어 있는 모든 컬럼의 정보가 "|"로 구분하여 표현된다.

시트가 단위데이터행 구조인 경우 Col 인자는 Index 값이 아닌 해당 컬럼의 SaveName으로 설정되며 SaveName 이 설정되지 않은 경우 -1로 설정되어 이 이벤트가 발생 한다.

SortOrder	설명
"ASC"	오름차순
"DESC"	내림차순

➤ Syntax

Syntax	function 오브젝트ID_OnSort(Col, SortOrder) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Col	Number/ String	소트가 처리된 컬럼 인덱스 SortEventMode 가 1인 경우 " "로 구분
SortOrder	String	소트 방향 문자열 SortEventMode 가 1인 경우 " "로 구분

➤ Example

<pre>function mySheet_OnSort(col, order) { if(order == "ASC") alert(col + "번째 컬럼이 오름차순으로 정렬되었습니다."); else alert(col + "번째 컬럼이 내림차순으로 정렬되었습니다."); }</pre>
--



➤ 제공 버전

7.0.0.0	
7.0.6.0	단위데이터행 구조에 대한 지원



OnTab Event

➤ 기능

시트에서 탭 키를 누르는 경우 OnTab 이벤트가 발생한다.

isLast 인자의 경우 탭으로 이동할 수 있는 마지막 셀인 경우 반환되는 값으로 Shift + Tab을 누르는 경우 첫셀, Tab을 누르는 경우 마지막 셀에서 1값을 반환한다.

TabBehavior, EditTabBehavior를 설정한 경우 설정값에 따라서 이동할 수 있는 마지막 셀에서 1값이 반환된다.

(이전행/컬럼과 현재행/컬럼이 동일한 경우에 1로 반환됨)

➤ Syntax

Syntax	function 오브젝트ID_OnTab (Row, Col, Old_Row, Old_Col, isShift, isLast) { }
--------	--

➤ Parameters

Parameter	Type	설 명
Row	Long	선택된 행 인덱스
Col	Long	선택된 컬럼 인덱스
Old_Row	Long	이전 행 인덱스
Old_Col	Long	이전 컬럼 인덱스
isShift	Boolean	Shift 키 입력 여부
isLast	Boolean	마지막 셀 여부

➤ Example

```
function mySheet_OnTab(Row, Col, Orow, Ocol, isShift, isLast) {  
    alert("[OnTab] New : " + Row + "," + Col + " Old : " + Orow + "," + Ocol + "  
Shift : " + isShift + " isLast : " + isLast);  
}
```




- 제공 버전
7.0.0.0



OnTreeCheckChange Event

➤ 기능

트리구조의 시트에서 트리 기준 컬럼에 대한 체크박스 사용 모드인 경우 해당 체크박스의 값이 변경시 이 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnTreeCheckChange(Row, Col, Value, Level, HasChild) {}
--------	---

➤ Parameters

Parameter	Type	설 명
Row	Number	대상 행의 Index
Col	Number	대상 컬럼의 Index
Value	Boolean	변경된 체크박스의 값
Level	Number	대상 행의 트리 레벨
HasChild	Boolean	대상 행의 자식노드 존재 여부

➤ Example

```
// 해당 행의 자식노드가 있는 경우 자식노드에 동일한값 처리
function mySheet_OnTreeCheckChange(row, col, value, level, hasChild) {
    var childRows = [];
    // 자식행이 있는 경우
    if (hasChild) {
        // 해당 행의 바로 아래의 자식행 정보 수집
        childRows = mySheet.GetChildRows(row, level + 1).split("|");
        for (var i = 0, len = childRows.length; i < len; i++) {
            // 자식행에 동일한 체크값 설정
            mySheet.SetTreeCheckValue(childRows[i], value);
        }
    }
}
```



}

➤ 제공 버전

7.0.9.0	
---------	--



OnTreeChild Event

➤ 기능

자식을 조회하지 않은 상태에서 부모 노드에서 트리 확장기능을 선택했을 때 이벤트가 발생한다.

이벤트가 발생하는 자식노드를 DoSearchChild 조회함수로 조회하는 기능을 처리할 수 있다.

이 이벤트는 조회시 XML 파일에는 <TR HAVECHILD="1">, JSON파일에는 TR 에 HaveChild 인자값을 먼저 설정하도록 한다.

HaveChild 속성값을 설정한 데이터에서만 발생하므로 반드시 조회XML,JSON 생성시 자식 레벨이 있는지 설정하여 처리하도록 한다.

이 이벤트는 트리 형태 데이터를 한꺼번에 조회하지 않고, 트리 확장 버튼을 눌렀을 때 자식 레벨의 데이터를 조회하기 위한 용도로 사용된다.

➤ Syntax

Syntax	function 오브젝트ID_OnTreeChild(Row)
--------	---

➤ Parameters

Parameter	Type	설 명
Row	Long	확장할 부모 행의 Index

➤ Example

<pre>// 자식데이터 조회하기 <script type="text/Javascript"> function mySheet_OnTreeChild(Row){ var url = ""; // 4컬럼 : 트리컬럼 switch(mySheet.GetCellValue(Row, 4)){ case "서울" : url = " type15_dat(1).xml";</pre>



```
        break;
    case "인천":
        url = "type15_data(2).xml";
        break;
    }
    mySheet.DoSearchChild(Row, url, "", 1);
}
</script>
```

➤ **제공 버전**

7.0.0.0



OnUserResize Event

➤ 기능

End-User가 마우스를 이용하여 상단 헤더의 컬럼 너비를 변경할때 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnUserResize(Col, Width) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Col	Long	해당 Column Index
Width	Long	해당 컬럼 너비

➤ Example

```
function mySheet_OnUserResize(Col, Width) {  
    alert(Col + "컬럼의 너비가 " + Width + "로 변경되었습니다.")  
}
```

➤ 제공 버전

7.0.0.0

OnValidation Event

➤ 기능

저장함수를 호출하여 저장 처리 하기 전에 사용자들이 저장할 값들에 대해 Validation Check할수 있도록 이벤트가 발생한다.

기본적인 필수입력 항목에 대한 확인이나 전체 입력 여부에 대한 확인은 저장 전에 IBSheet가 내부적으로 확인하나 업무 로직에 따라 발생할 수 있는 각종 Validation을 이 이벤트에서 처리한다.

Validation에 맞지 않다면 ValidateFail(1)로 설정하여 저장 처리를 중단한다.

이 이벤트는 저장할 데이터 각 셀단위로 발생하므로 여기서 For를 사용하여 전체 Validation을 확인할 경우 시간이 오래 걸릴수 있으므로 전체에 대한 확인은 저장 함수 호출 전에 처리하고, 각 셀 단위로 확인할 로직만 기재하도록 한다.

➤ Syntax

Syntax	function 오브젝트ID_OnValidation (Row, Col, Value) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Row	Long	해당 셀의 Row Index
Col	Long	해당 셀의 Column Index
Value	variant	Format이 적용되지 않은 저장 시 사용되는 값

➤ Example

```
function mySheet_OnValidation(Row, Col, Value) {
    switch(Col) {
        case 2:
            if(Value=="원화" && mySheet.GetCellValue(Row,Col+1) >= 10000000) {
                alert("통화가 원화일 때 금액은 천만원 이상일수 없습니다.");
            }
        }
    }
```



```
mySheet.ValidateFail(1);  
mySheet.SelectCell(Row, Col+1);  
} else if(Value=="외화" && mySheet.GetCellValue(Row,Col+1) < 10000000)  
{  
    alert("통화가 외화일 때 금액은 천만원 이상이어야 합니다.");  
    mySheet.ValidateFail(1);  
    mySheet.SelectCell(Row, Col+1);  
}  
}  
}
```

➤ **제공 버전**

7.0.0.0



OnVScroll Event

➤ 기능

세로스크롤 시 이벤트가 발생한다.

➤ Syntax

Syntax	function 오브젝트ID_OnVScroll(vpos, oldvpos, isTop, isBottom) { }
--------	---

➤ Parameters

Parameter	Type	설 명
vpos	Long	세로 스크롤 값
oldvpos	Long	이전 세로 스크롤 값
isTop	Boolean	세로 스크롤이 최상단에 위치했는지 여부
isBottom	Boolean	세로 스크롤이 최하단에 위치했는지 여부

➤ Example

<pre>function mySheet_OnVScroll(vpos, oldvpos, isTop, isBottom) { }</pre>

➤ 제공 버전

7.0.0.0



OnWaitTimeOut Event

➤ 기능

서버에 연결하여 처리 중 일정 시간을 초과하여 처리가 중단 된 경우 이벤트가 발생한다.

아이비시트는 조회,저장과 같은 ajax 통신시 기본적으로 60초간 대기 후 커넥션을 끊으며, 이 시간은 WaitTimeOut() 메서드를 통해 조절이 가능하다.

➤ Syntax

Syntax	function 오브젝트ID_OnWaitTimeOut(Sec) { }
--------	---

➤ Parameters

Parameter	Type	설 명
Sec	Long	타임아웃에 걸린 시간 (단위:초)

➤ Example

<pre>function mySheet_OnWaitTimeOut(Sec) { // 시간 초과가 발생한 경우 alert("대기 시간을 초과 하였습니다. "); }</pre>

➤ 제공 버전

7.0.0.0



(주) 아이비 리더스

Inno Biz Leaders Co., Ltd.



Chapter 9. IBSheet Methods



(주) 아이비 리더스

Inno Biz Leaders Co., Ltd.



9. IBSheet Methods

9.1 Method 사용

IBSheet에서 제공하는 Method를 사용하기 위해서는 다음과 같은 Syntax를 사용할 수 있다.

Syntax	오브젝트ID.메소드명()
--------	---------------

Method의 인자는 [~] 사이에 설정된 경우 선택인자 이므로 설정 하지 않고 호출할 경우 Default 값 적용을 받아 Method가 처리된다.

IBSheet가 제공하는 함수는 다음과 같다.

9.2 Method List

ActionMenu Method

➤ 기능

컬럼 팝업과 같이 특정 컬럼에 값을 바꾸기 위해 사용하는 메뉴팝업이 아니라 전반적으로 어떤 Action을 처리하기 위해 사용하는 메뉴를 설정하거나 확인한다. 마우스 오른쪽 버튼을 눌렀을 때 팝업 메뉴 형태로 표시되며, 해당하는 위치 컬럼에 이미 컬럼 팝업이 설정되어 있다면 컬럼 팝업이 우선적으로 표시된다. 표시할 메뉴글자를 "|"로 연결하여 설정하고 메뉴글자가 "*-"가 설정되면 메뉴가 아닌 중간 구분자로 표시된다.

표시된 메뉴를 선택하면 OnSelectMenu 이벤트가 발생한다.

텍스트와 코드를 모두 설정하거나 설정인자를 JSON 객체로 설정하는 경우 GetActionMenu의 리턴값은 JSON객체로 리턴된다.

JSON 객체로 설정하는 경우에 Icon 속성에 이미지 url 을 설정하여 메뉴앞에 아이콘을 출력할 수 있다. 아이콘을 설정하는경우 기본으로 메뉴 앞에 출력되는 Cursor 아이콘의 출력 여부를 두번째 인자에 설정할 수 있으며 이미지 아이콘을 설정하는 경우 기본값은 0으로 커서 아이콘을 출력하지 않는다.

이미지리스트를 설정해 둔 경우 Icon 속성에 이미지 인덱스를 설정할 수 있다.

➤ Syntax

Syntax	Get	ObjId. GetActionMenu ()
	Set	ObjId. SetActionMenu (Text, Code)

➤ Info

Return	String(Object), 설정된 팝업메뉴 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Text	String	필수	설정할 팝업메뉴 문자열
Code	String	선택	설정할 팝업메뉴 코드 문자열 Default="텍스트 문자열"



➤ **Example**

```
// 메뉴만 설정
mySheet.SetActionMenu("입력|행복사|*-*|행삭제|Clear|엑셀다운로드");

// 메뉴와 코드 설정
var Text = "입력|행복사|*-*|행삭제|Clear|엑셀다운로드";
var Code = "Ins|Copy||Del|Clear|Download";
mySheet.SetActionMenu(Text, Code);

// JSON 객체 설정
var Menu = [
    {Text: "입력", Code: "Ins"},
    {Text: "행복사", Code: "Copy"},
    {Text: "*-*"},
    {Text: "행삭제", Code: "Delete"},
    {Text: "Clear", Code: "Clear"},
    {Text: "엑셀다운로드", Code: "Download"}
];
mySheet.SetActionMenu(Menu);

// JSON 객체 계층구조설정
var Menu = [
    {Text: "입력", Code: "Ins",
        Items : [
            { Text: "첫행입력", Code: "Fins"},
            { Text: "마지막행입력", Code: "Lins"}
        ]
    },
    {Text: "행복사", Code: "Copy"},
    {Text: "*-*"},
    {Text: "행삭제", Code: "Delete"},

```




```
{Text: "Clear", Code: "Clear"},
{Text: "엑셀다운로드", Code: "Download"}
];
mySheet.SetActionMenu(Menu);

// 이미지 Icon 설정
var Menu = [
{Text: "입력", Code: "Ins", Icon: "../image/ess1.gif",
  Items : [
    { Text: "첫행입력", Code: "Fins", Icon: "../image/ess2.gif"},
    { Text: "마지막행입력", Code: "Lins"}
  ]
},
{Text: "행복사", Code: "Copy"},
{Text: "*-"},
{Text: "행삭제", Code: "Delete"},
{Text: "Clear", Code: "Clear"},
{Text: "엑셀다운로드", Code: "Download"}
];
mySheet.SetActionMenu(Menu);    // 커서 아이콘 출력 안함
mySheet.SetActionMenu(Menu, 1); // 커서 아이콘 출력함

// 설정된 메뉴 확인
var menu = mySheet.GetActionMenu();
```

➤ 제공 버전

7.0.0.0



AllowCheck Method

➤ 기능

사용자가 체크박스의 값 변경시, 값의 변경을 수용할 지의 여부를 설정하거나 확인한다.

OnBeforeCheck Event 발생시 AllowCheck 메소드를 통하여 체크박스의 체크 허용 / 불허의 제어가 가능하다.

➤ Syntax

Syntax	Get	ObjId. GetAllowCheck()
--------	-----	-------------------------------

➤ Info

Return	Boolean, 설정된 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// 체크박스의 값을 확인한다.
mySheet.GetAllowCheck();
```

➤ Syntax

Syntax	Set	ObjId. SetAllowCheck(Val)
--------	-----	----------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Val	Boolean	필수	값의 변경 수용여부. Default=true

➤ Example

```
// 체크박스의 값의 변경을 하지 않는다.
mySheet.SetAllowCheck(false);
```



- 제공 버전
7.0.0.0



AllowEvent4CheckAll Method

➤ 기능

전체선택 기능을 실행할 때 OnChange Event를 실행할지 여부를 지정한다. Data의 Row가 많은 경우 전체선택 기능을 실행하면 각 체크박스에서 OnChange Event가 발생하게 되어 속도가 저하되는 현상이 발생한다. 각 체크박스에서 OnChange 이벤트가 필요 없을 때 AllowEvent4CheckAll(0)으로 설정하면 전체선택 속도를 증가시킬 수 있다. 기본값은 1이다.

➤ Syntax

Syntax	ObjId.AllowEvent4CheckAll(Val)
--------	--------------------------------

➤ Info

Return	None.		
Parameter	Type	필수여부	설 명
Val	Boolean	필수	OnChange Event 발생여부

➤ Example

```
// 전체선택시 이벤트 발생안함  
sheetObj.AllowEvent4CheckAll(0);
```

➤ 제공 버전

7.0.0.0

AllowExpand Method

➤ 기능

OnBeforeExpand 이벤트가 발생한 직후에 실제로 트리를 펼치거나 접혀지는 것을 막고싶은 경우 사용한다.

AllowExpand 값은 OnBeforeExpand 이벤트가 발생할 때마다 매번 1 로 리셋 된다.

OnBeforeExpand 이벤트 안에서 이 속성을 0으로 지정시에만 사용자의 트리 동작을 막을 수 있다.

➤ Syntax

Syntax	Get	ObjId. GetAllowExpand()
--------	-----	--------------------------------

➤ Info

Return	Boolean, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
mySheet.GetAllowExpand();
```

➤ Syntax

Syntax	Set	ObjId. SetAllowExpand (Expand)
--------	-----	---------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Expand	Boolean	필수	트리의 펼쳐짐 여부

➤ Example



```
function mySheet_OnBeforeExpand(Row,Expand) {  
    if(Row == 1 && Expand == 2) {  
        mySheet.SetAllowExpand(0);  
        alert("1 행에서는 펼칠 수 있지만 접을 수 없습니다.");  
    }  
}
```

➤ **제공 버전**

7.0.0.0

AutoRowHeight Method

➤ 기능

데이터 행의 높이를 자동으로 조정할지 여부를 확인하거나 설정한다.

이 속성이 1이면, 행의 글자를 모두 표시할수 있도록 행의 높이를 자동으로 높이거나 줄인다. 행의 높이가 변경되는 경우는 다음과 같다.

- 1) InitColumns에 컬럼속성을 MultiLineText:1로 설정하여 키보드로 편집도중 Shift+Enter 키를 눌러 줄바꿈된 글자를 설정하는 경우
- 2) InitColumns에 컬럼속성을 Wrap:1로 설정하여 컬럼너비조정에 따라 글자가 자동으로 줄바꿈된 경우
- 3) 줄바꿈된 글자를 삭제하여 행의 높이가 줄어드는 경우
- 4) SetCellText, SetCellValue 로 줄바꿈된 글자를 설정한 경우
- 5) 조회, 엑셀업로드로 줄바꿈된 글자를 로드하는 경우

이 속성이 0이면, 행의 높이는 SetDataRowHeight 속성으로 설정된 높이인 한 줄 높이로 고정된다.

➤ Syntax

Syntax	Get	ObjId. GetAutoRowHeight()
--------	-----	----------------------------------

➤ Info

Return	Boolean, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

// 설정값 확인 mySheet.GetAutoRowHeight();
--

➤ Syntax

Syntax	Set	ObjId. SetAutoRowHeight (Flag)
--------	-----	---------------------------------------



➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Flag	Boolean	필수	데이터행의 높이 자동조정 여부 Default=1

➤ **Example**

```
// 높이를 자동으로 조정하지 않고, 한줄 높이로 고정한다.  
mySheet.SetAutoRowHeight(0);
```

➤ **제공 버전**

7.0.0.0

AutoSumPosition Method

➤ 기능

전체 합계/평균의 표시 위치를 설정하거나 확인한다.

컬럼 타입이 "AutoSum", "AutoAvg"인 경우 전체 합계 또는 전체 평균을 계산하여 표시한다. 기본값은 1이다.

표시하는 위치는 다음과 같다

설정 값	설 명
0	헤더 바로 아래인 최상단에 고정하여 표시
1	화면에 보이는 영역의 최하단에 고정되어 표시

➤ Syntax

Syntax	Get	ObjId. GetAutoSumPosition ()
--------	-----	-------------------------------------

➤ Info

Return	Integer, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

//합계 행을 상단에 표시 mySheet.GetAutoSumPosition();

➤ Syntax

Syntax	Set	ObjId. SetAutoSumPosition (Position)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Position	Integer	필수	합계행의 위치 값



➤ **Example**

```
//합계 행을 상단에 표시  
mySheet.SetAutoSumPosition(0);
```

➤ **제공 버전**

7.0.0.0

BasicImeMode Method

➤ 기능

Edit 가능한 셀에 포커스가 들어갔을 때 한/영 키보드 상태를 설정하거나 확인한다. 설정할수 있는 값은 아래와 같다.

설정 값	Value	설 명
imeAuto	0	마지막 상태 그대로 사용
imeHan	1	기본 상태를 한글 입력 상태로 함
imeEng	2	기본 상태를 영문 입력 상태로 함

단 SetBasicImeMode() 메서드의 사용은 조회 전이나 RemoveAll()메서드를 통해 데이터를 초기화 한 상태에서만 사용이 가능하며 해당 기능은 IE 브라우저에서만 사용이 가능 하다.

➤ Syntax

Syntax	Get	ObjId. GetBasicImeMode()
--------	-----	---------------------------------

➤ Info

Return	Integer, 설정값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

//기본 한글 입력 키보드 상태 사용 mySheet.GetBasicImeMode();
--

➤ Syntax

Syntax	Set	ObjId. SetBasicImeMode(Val)
--------	-----	------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명



Val	Integer	필수	ImeMode 값
-----	---------	----	-----------

➤ **Example**

```
//기본 한글 입력 키보드 상태 사용  
mySheet.SetBasicImeMode(1);
```

➤ **제공 버전**

7.0.0.0

CellAlign Method

➤ 기능

셀에 대한 정렬을 설정하거나 확인한다. 이 속성은 InitColumns 함수에서 설정한 정렬과 달리 화면이 보이는 상태에서 정렬을 단순히 바꾸는 처리를 한다. 설정할 수 있는 값을 아래와 같다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

좌측 정렬	가운데 정렬	우측 정렬
"Left"	"Center"	"Right"

➤ Syntax

Syntax	Get	ObjId. GetCellAlign (Row, Col)
--------	-----	---------------------------------------

➤ Info

Return	String, 현재 정렬 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName

➤ Example

```
// 해당 셀의 정렬 값을 확인한다.
var align = mySheet.GetCellAlign(1, 1);
alert("셀의 정렬 값은 " + align + "입니다.");
```

➤ Syntax

Syntax	Set	ObjId. SetCellAlign (Row, Col, Align)
--------	-----	--



➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName
Align	String	필수	정렬 값

➤ **Example**

```
// 해당 셀을 가운데 정렬로 변경한다.  
mySheet.SetCellAlign(1, 1, "Center");
```

➤ **제공 버전**

7.0.0.0

CellBackColor Method

➤ 기능

셀의 배경색을 설정하거나 확인한다.

그 셀이 데이터 행이든 헤더 행이든 셀의 배경색을 설정하거나 확인할 수 있으며, 셀이 존재하지 않는 경우 에러메시지 없이 배경색 설정은 취소된다. 색상 설정은 WebColor 또는 Basic 16 Color 문자열로 설정한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

➤ Syntax

Syntax	Get	ObjId. GetCellBackColor (Row, Col)
--------	-----	---

➤ Info

Return	String, 설정된 색상값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName

➤ Example

```
// 설정된 색상 값을 확인한다.
var color = mySheet.GetCellBackColor(1,1);
alert("셀의 색상 값은 " + color + "입니다.");
```

➤ Syntax

Syntax	Set	ObjId. SetCellBackColor (Row, Col, Color)
--------	-----	--

➤ Info



Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName
Color	String	필수	색상 값

➤ **Example**

```
//헤더행의 첫번째 셀의 배경색을 빨간색으로 변경
mySheet.SetCellBackColor(0, 0, "#FF0000");           // WebColor

//데이터행의 셀의 배경색을 헤더행의 첫번째 셀의 배경색과 동일하게 설정
mySheet.SetCellBackColor(1,0, mySheet.GetCellBackColor(0, 0));
//헤더행의 SaveName이 "amt"인 셀의 배경색을 빨간색으로 변경
mySheet.SetCellBackColor(0,"amt", "#FF0000");
```

➤ **제공 버전**

7.0.0.0

CellComboItem Method

➤ 기능

특정 셀의 Combo 항목만 다른 경우 Combo항목을 개별 설정한다.

기본적인 컬럼 전체의 콤보 항목은 InitColumns함수를 이용하여 설정하고, 특정 셀만 항목이 다른 경우 이 함수를 이용한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

➤ Syntax

Syntax	ObjId.CellComboItem(Row, Col,info)
--------	------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	특정 셀의 Row Index
Col	Long / String	필수	특정 셀의 Column Index 또는 SaveName
info	Object	필수	변경할 Combo item 내용을 구분자 " "로 연결된 문자열을 만들어 ComboCode와 ComboText로 값을 설정

➤ Example

```
//특정 셀의 콤보 항목 바꾸기
var info = {"ComboCode":"사장|과장|대리","ComboText":"A|B|C"};
mySheet.CellComboItem(1,2,info);
```

➤ 제공 버전



7.0.0.0

CellCursor Method

➤ 기능

특정 셀의 마우스 오버시 커서의 모양을 설정한다.

"Default","Pointer" 로 설정이 가능하며 기본적으로 "Default"값이 설정된다.

➤ Syntax

Syntax	ObjId. GetCellCursor (Row, Col)
--------	--

➤ Info

Return	String		
Parameter	Type	필수여부	설 명
Row	Long	필수	특정 셀의 Row Index
Col	Long / String	필수	특정 셀의 Column Index 또는 SaveName

➤ Example

// (2, 3) 셀의 마우스 커서포인터를 확인한다. mySheet.GetCellCursor(2, 3);

➤ Syntax

Syntax	ObjId. SetCellCursor (Row, Col, Cursor)
--------	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	특정 셀의 Row Index
Col	Long / String	필수	특정 셀의 Column Index 또는 SaveName
Cursor	String	필수	마우스 모양 설정



➤ **Example**

```
// (2, 3) 셀의 마우스 커서포인트를 손가락 모양으로 변경한다.  
mySheet.SetCellCursor(2, 3, 'Pointer');
```

➤ **제공 버전**

7.0.13.60

CellEditable Method

➤ 기능

셀의 Edit 가능 여부를 확인하거나 설정한다.

해당 셀의 Edit 가능 여부는 Editable 속성에서 설정한 전체 Edit 가능 여부와 InitColumns 함수에서 설정한 Edit설정 및 RowEditable 함수의 설정, 트랜잭션 상태, 자동 계산 여부에 따라 결정된다.

자동 계산 셀은 CellEditable 설정에 관계 없이 Edit 불가능하다.

트랜잭션 상태가 "삭제"인 경우 컬럼타입이 DelCheck를 제외한 모든 데이터는 Edit 불가능하게 된다. 여타 설정에 관계없이 Edit 불가능한 데이터 타입은 Status, Image, Seq 이다

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

위의 경우를 제외한 이속성에 영향을 받는 경우 데이터의 Edit 가능여부는 다음과 같다 .

Editable	ColEditable	RowEditable	CellEditable	셀의 Edit 가능 여부
불가	무관	무관	무관	불가
가능	불가	불가	가능/불가	가능/불가
가능	가능	가능/불가	가능/불가	가능/불가

➤ Syntax

Syntax	Get	ObjId. GetCellEditable (Row, Col)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명



Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName

➤ **Example**

```
//1행 2컬럼이 Edit 가능이면 3컬럼도 Edit 가능으로 변경한다.
If(mySheet.GetCellEditable(1, 2) == 1) {
    mySheet.SetCellEditable(1, 3, 1);
}
```

➤ **Syntax**

Syntax	Set	ObjId. SetCellEditable (Row, Col, Edit)
--------	-----	--

➤ **Info**

Return	Boolean, 편집 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName
Editable	Boolean	필수	편집 가능 여부

➤ **Example**

```
//1행 2컬럼이 Edit 가능이면 3컬럼도 Edit 가능으로 변경한다.
If(mySheet.GetCellEditable(1, 2) == 1) {
    mySheet.SetCellEditable(1, 3, 1);
}
```

➤ **제공 버전**

7.0.0.0

CellFont Method

➤ 기능

특정 셀 또는 영역의 글꼴, 크기, 글자색, 기울임, 볼드, 밑줄 등을 확인하거나 설정한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

Flag에 사용할 수 있는 Font 구분값은 다음과 같다.

FLAG 속성	설명
"FontName"	글꼴, String
"FontSize"	글자크기, Integer
"FontColor"	글자색상, String
"FontBold"	글자 볼드 여부, Boolean
"FontItalic"	글자 기울임 여부, Boolean
"FontUnderline"	글자 밑줄 표시 여부, Boolean
"FontStrike"	글자 취소선 표시 여부, Boolean

➤ Syntax

Syntax	Get	ObjId. GetCellFont (Flag, Row1, Col1, Row2, Col2)
--------	-----	--

➤ Info

Return	Boolean / String, 설정한 속성 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Flag	String	필수	폰트 속성
Row1	Long	필수	영역 시작 셀의 Row Index
Col1	Long / String	필수	영역 시작 셀의 Column Index 또는 SaveName
Row2	Long	필수	영역 마지막 셀의 Row Index
Col2	Long / String	필수	영역 마지막 셀의 Column Index



			또는 SaveName
--	--	--	-------------

➤ **Example**

```
//글자크기가 10보다 큰 경우 9사이즈로 변경한다.
If (mySheet.GetCellFont("FontSize", 2,1,2,1) >= 10) {
    mySheet.SetCellFont("FontSize", 2,1,2,1,9);
}
```

➤ **Syntax**

Syntax	Set	ObjId. SetCellFont (Flag, Row1, Col1, Row2, Col2, Value)
--------	-----	---

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Flag	String	필수	폰트 속성
Row1	Long	필수	영역 시작 셀의 Row Index
Col1	Long / String	필수	영역 시작 셀의 Column Index 또는 SaveName
Row2	Long	필수	영역 마지막 셀의 Row Index
Col2	Long / String	필수	영역 마지막 셀의 Column Index 또는 SaveName
Value	Boolean / String	필수	폰트속성에 설정할 값

➤ **Example**

```
//영역에 글자 Bold 설정
mySheet.SetCellFont(« FontBold », 1,1,2,3,1) ;

//영역에 글자 취소선 설정
mySheet.SetCellFont(« FontStrike », 1,1,2,3,1) ;
```

➤ **제공 버전**



7.0.0.0



CellFontBold Method

➤ 기능

특정 셀의 글자 Bold(굵은 글자체) 여부를 확인하거나 설정한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

➤ Syntax

Syntax	Get	ObjId. GetCellFontBold (Row, Col)
--------	-----	--

➤ Info

Return	Boolean, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName

➤ Example

//특정 셀의 Bold 처리 여부를 확인한다. Alert(mySheet.GetCellFontBold(1, 1));
--

➤ Syntax

Syntax	Set	ObjId. SetCellFontBold (Row, Col, Bold)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long	필수	해당 셀의 Column Index



	/String		또는 SaveName
Bold	Boolean	필수	Bold 여부

➤ **Example**

//특정 셀에 글자를 Bold 처리한다. mySheet.SetCellFontBold(1, 1,1)

➤ **제공 버전**

7.0.0.0



CellFontColor Method

➤ 기능

셀의 글자 색을 확인하거나 설정한다.

셀이 헤더 영역이든 데이터 영역이든 글자색을 설정할 수 있고, 셀이 존재하지 않는 경우 에러메시지를 표시하지는 않지만 설정은 취소된다.

색상 설정은 WebColor 또는 Basic 16 Color 문자열로 설정한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

➤ Syntax

Syntax	Get	ObjId. GetCellFontColor (Row, Col)
--------	-----	---

➤ Info

Return	String, 설정된 색상값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName

➤ Example

```
//금액이 0보다 작으면 숫자는 붉은색으로 표시한다.  
If(mySheet.GetCellValue(1, 2) < 0 ) {  
    mySheet.SetCellFontColor(1,2,"#FF0000") ;  
//금액이 0보다 크면 숫자는 검은색으로 표시한다.  
} else {  
    mySheet.SetCellFontColor(1,2, "#000000");  
}
```



➤ **Syntax**

Syntax	Set	ObjId. SetCellFontColor (Row, Col, Color)
--------	-----	--

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName
Color	String	필수	색상 값

➤ **Example**

```
//금액이 0보다 작으면 숫자는 붉은색으로 표시한다.  
If(mySheet.GetCellValue(1, 2) < 0 ) {  
    mySheet.SetCellFontColor(1,2,"#FF0000") ;  
//금액이 0보다 크면 숫자는 검은색으로 표시한다.  
} else {  
    mySheet.SetCellFontColor(1,2, "#000000");  
}
```

➤ **제공 버전**

7.0.0.0



CellFontItalic Method

➤ 기능

특정 셀의 글자 이탤릭(기울임 글자체) 여부를 확인하거나 설정한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

➤ Syntax

Syntax	Get	ObjId. GetCellFontItalic (Row, Col)
--------	-----	--

➤ Info

Return	Boolean, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName

➤ Example

//특정 셀의 이탤릭체 여부를 확인한다. Alert(mySheet.GetCellFontItalic(1, 1));

➤ Syntax

Syntax	Set	ObjId. SetCellFontItalic (Row, Col, Italic)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long	필수	해당 셀의 Column Index



	/String		또는 SaveName
Italic	Boolean	필수	이탤릭 여부

➤ **Example**

//특정 셀에 글자를 이탤릭체로 설정한다. mySheet.SetCellFontItalic(1, 1,1)
--

➤ **제공 버전**

7.0.0.0



CellFontName Method

➤ 기능

특정 셀에 대해 글자체를 확인하거나 설정한다.

확인 리턴값 과 설정 인자는 표준 태그의 style 속성에서 사용하는 폰트명을 사용한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

➤ Syntax

Syntax	Get	ObjId. GetCellFontName (Row, Col)
--------	-----	--

➤ Info

Return	String, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName

➤ Example

//글자체를 확인한다 alert(mySheet.GetCellFontName(1, 1));
--

➤ Syntax

Syntax	Set	ObjId. SetCellFontName (Row, Col, FontName)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명



Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName
FontName	String	필수	글자체

➤ **Example**

```
//글자체를 궁서체로 바꾼다.  
mySheet.SetCellFontName(1, 1, "Gungsuh")
```

➤ **제공 버전**

7.0.0.0



CellFontSize Method

➤ 기능

특정 셀에 대해 글자 크기를 확인하거나 설정한다.

설정 및 확인 값은 픽셀 단위의 값이다..

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

➤ Syntax

Syntax	Get	ObjId. GetCellFontSize (Row, Col)
--------	-----	--

➤ Info

Return	Integer, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName

➤ Example

// 셀의 글자 크기를 확인한다. Alert(mySheet.GetCellFontSize(1,1));
--

➤ Syntax

Syntax	Set	ObjId. SetCellFontSize (Row, Col, FontSize)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index



Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName
FontSize	Integer	필수	설정하고자 하는 글자크기

➤ **Example**

```
// 글자크기를 20px로 바꾼다.  
mySheet.SetCellFontSize(1, 1, 20)  
  
// SaveName이 "sName" 인 셀의 글자크기를 20으로 바꾼다.  
mySheet.SetCellFontSize(1,"sName", 20);
```

➤ **제공 버전**

7.0.0.0



CellFontStrike Method

➤ 기능

특정 셀 글자에 취소선을 설정한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

출력 예) 텍스트

➤ Syntax

Syntax	Get	ObjId. GetCellFontStrike (Row, Col)
--------	-----	--

➤ Info

Return	Integer, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Number	필수	대상 셀의 행 Index
Col	Number /String	필수	대상 셀의 컬럼 Index 또는 SaveName

➤ Example

```
// 셀의 글자 취소선 설정여부를 확인한다.
Console.log("GetCellFontStrike : ", mySheet.GetCellFontStrike(1,1));
```

➤ Syntax

Syntax	Set	ObjId. SetCellFontStrike (Row, Col, FontStrike)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명



Row	Number	필수	대상 셀의 행 Index
Col	Number /String	필수	대상 셀의 컬럼 Index 또는 SaveName
FontStrike	Boolean	필수	취소선 설정 여부

➤ **Example**

```
// 글자 취소선을 설정한다.  
mySheet.SetCellFontStrike(1, 1, 1)  
// SaveName이 "sName" 인 셀의 글자에 취소선을 설정한다.  
mySheet.SetCellFontStrike(1,"sName", 1);
```

➤ **제공 버전**

7.0.0.0



CellFontUnderline Method

➤ 기능

셀의 글자의 밑줄을 확인하거나 설정한다.

셀이 헤더 영역이든 데이터 영역이든 언더라인을 설정할 수 있고, 셀이 존재하지 않는 경우 에러메시지를 표시하지는 않지만 설정은 취소된다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

➤ Syntax

Syntax	Get	ObjId. GetCellFontUnderline (Row, Col)
--------	-----	---

➤ Info

Return	Boolean, 현재 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName

➤ Example

```
//금액이 0보다 작으면 언더라인 표시한다.  
If(mySheet.GetCellValue(1, 2) < 0 ) {  
    mySheet.SetCellFontUnderline(1, 2, 1);  
  
//금액이 0보다 크면 언더라인을 표시하지 않는다.  
} else {  
    mySheet.SetCellFontUnderline(1, 2, 0);  
}
```



➤ **Syntax**

Syntax	Set	ObjId. SetCellFontUnderline (Row, Col, Underline)
--------	-----	--

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName
Underline	Boolean	필수	밑줄 여부

➤ **Example**

```
//금액이 0보다 작으면 언더라인 표시한다.  
If(mySheet.GetCellValue(1, 2) < 0 ) {  
    mySheet.SetCellFontUnderline(1, 2, 1);  
  
//금액이 0보다 크면 언더라인을 표시하지 않는다.  
} else {  
    mySheet.SetCellFontUnderline(1, 2, 0);  
}
```

➤ **제공 버전**

7.0.0.0



CellImage Method

➤ 기능

Type이 Image이거나 Image 속성을 사용한 경우 해당 셀의 이미지를 확인하거나 설정한다.

이 속성에 설정되는 값은 이미지파일의 실제 경로를 설정하고, Type이 맞지 않거나 셀에 이미지가 설정되지 않은 경우 설정이 취소된다.

또한, CellValue 함수로 값을 확인하고 설정할수 있다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

➤ Syntax

Syntax	Get	ObjId. GetCellImage (Row, Col)
--------	-----	---------------------------------------

➤ Info

Return	String, 설정된 이미지 경로 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName

➤ Example

// 데이터 행을 신규 생성 하고 셀의 이미지와 문자열을 변경한다. var Row=mySheet.DataInsert(); mySheet.GetCellImage(Row, 1);

➤ Syntax

Syntax	Set	ObjId. SetCellImage (Row, Col, Image)
--------	-----	--



➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName
Image	String	필수	실제 이미지 경로

➤ **Example**

```
// 데이터 행을 신규 생성 하고 셀의 이미지와 문자열을 변경한다.  
var Row=mySheet.DataInsert();  
mySheet.SetCellImage(Row, 1, "../image/myImage1.jpg");  
  
// 1컬럼의 SaveName이 "btnAction"인 경우  
mySheet.SetCellImage(Row ,"btnAction", "../image/myImage1.jpg");
```

➤ **제공 버전**

7.0.0.0



CellSaveName Method

➤ 기능

Row, Col 인자의 데이터 셀에 대한 SaveName을 확인 한다.

➤ Syntax

Syntax	ObjId. CellSaveName (Row, Col)
--------	---------------------------------------

➤ Info

Return	String, 해당 셀의 SaveName		
Parameter	Type	필수여부	설 명
Row	Number	필수	해당 셀의 Row Index
Col	Number	필수	해당 셀의 Column Index

➤ Example

```
function mySheet_OnClick(Row, Col, Value, CellX, CellY, CellW, CellH) {  
    // 클릭한 셀의 SaveName을 확인한다.  
    var saveName = mySheet.CellSaveName(Row, Col);  
}
```

➤ 제공 버전

7.0.0.0

CellSearchValue Method

➤ 기능

셀이 조회되었던 당시의 값을 확인한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

해당 행이 입력된 행이거나 다음과 같은 Type일때, 값은 공백이다.

Status, DelCheck, Seq, Image

➤ Syntax

Syntax	ObjId. CellSearchValue (Row, Col)
--------	--

➤ Info

Return	String, 조회당시 셀 값		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName

➤ Example

```
// 조회되었던 셀 값을 확인하고, 해당 셀 값을 원래 값으로 되돌린다.
var OrgValue = mySheet.CellSearchValue(Row, Col);

if(OrgValue != mySheet.GetCellValue(Row, Col)) {
    mySheet.SetCellValue(Row, Col, OrgValue);
}
```

➤ 제공 버전



7.0.0.0

CellVAlign Method

➤ 기능

셀 세로 정렬을 설정하거나 확인한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

➤ Syntax

Syntax	Get	ObjId. GetCellVAlign (Row, Col)
--------	-----	--

➤ Info

Return	String, 세로 정렬 설정값		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName

➤ Example

```
// 특정 셀의 세로 정렬 설정을 확인한다.
Alert(mySheet.GetCellVAlign(1,1));
```

➤ Syntax

Syntax	Set	ObjId. SetCellVAlign (Row, Col, valign)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long	필수	해당 셀의 Column Index



	/String		또는 SaveName
valign	String	필수	셀 세로 정렬 설정값 (top / middle / bottom)

➤ **Example**

```
// 특정 셀에 세로 정렬을 top으로 설정한다.  
mySheet.SetCellVAlign(1, 1, "top");  
// 특정 셀에 세로 정렬을 bottom 으로 설정한다.  
mySheet.SetCellVAlign(1, 1, "bottom");
```

➤ **제공 버전**

7.0.0.0

CellText Method

➤ 기능

셀의 값을 Format이 적용된 형태로 화면에 보여지는 값 그대로 확인하거나 설정한다.

Type이 Status이고 화면에 보여지는 상태가 "입력"인 경우 이 속성을 호출하면 입력이라는 값 자체를 읽어오고, CellValue 속성을 호출하면 "I"라는 코드값을 반환한다.

Status처럼 모든 Type의 값은 화면에 보여지는 값 그대로 읽어오거나 설정하고, 체크박스가 사용되는 DelCheck, CheckBox와 같은 형태는 체크 여부 이외에 설정된 Text 자체를 확인할 수 있고, 체크여부는 CellValue속성을 이용하여 확인한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

➤ Syntax

Syntax	Get	ObjId. GetCellText (Row, Col)
--------	-----	--------------------------------------

➤ Info

Return	String, 셀에 설정된 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName

➤ Example

```
//날짜 셀, 확인 값은 "2011/07/15"
alert(mySheet.GetCellText(1,1));
```



➤ **Syntax**

Syntax	Set	ObjId. SetCellText (Row, Col, Text)
--------	-----	--

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName
Text	String	필수	셀에 설정할 값

➤ **Example**

```
//공백 설정
mySheet.SetCellText(1, 0, "" );

//날짜 셀에 값 설정
mySheet.SetCellText(1, 1, "2011/07/15");

//숫자 셀에 값 설정
mySheet.SetCellText(1, 2, 1,234,567);

//콤보 셀에 콤보코드가 아닌 콤보텍스트 설정
mySheet.SetCellText(1, 3, "원화");
```

➤ **제공 버전**

7.0.0.0

CellValue Method

➤ 기능

셀에 값을 설정하거나 확인한다.

셀의 값을 Format이 적용되지 않은 실제로 저장 시 사용하는 값으로 확인하고, 설정한다. 이 속성을 이용하여 셀에 값을 설정하면 기본적으로 OnChange 이벤트가 발생하며, CellValue 함수의 Flag 인자의 값이 0이면 OnChangeEvent는 발생하지 않고 값이 변경된다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

Type과 Format 별로 셀의 값은 다음과 같다.

Type	설명	
Seq Pass	설정이 불가능한 타입	
Status	입력 = "I", 수정 = "U", 삭제 = "D", 조회 = ""	
CheckBox, DummyCheck, RadioCheck	1 또는 0	
Combo ComboEdit	콤보 코드	
Image	해당 이미지의 실제경로	
그외 Type	Format	설명
	Ymd	숫자8자리일 때 "yyyy.mm.dd"으로 표현하며 확인 시 숫자 8자리 반환
	Ym	숫자6자리일 때 "yyyy.mm"으로 표현하며 확인 시 숫자 6자리 반환
	Md	숫자4자리일 때 "mm.dd"로 표현하며 확인 시 숫자 4자리 반환



Hms	숫자6자리일 때 "hh:mm:ss"로 표현하며 확인 시 숫자 6자리 반환
Hm	숫자4자리일 때 "hh:mm"로 표현하며 확인 시 숫자 4자리 반환
YmdHms	숫자14자리일 때 "yyyy.mm.dd hh:mm:ss"으로 표현하며 확인 시 숫자 14자리 반환
YmdHm,	숫자12자리일 때 "yyyy.mm.dd hh:mm"으로 표현하며 확인 시 숫자 12자리 반환
IdNo	숫자13자리일 때 "#####-#####"로 표현하며 확인 시 숫자13자리
SaupNo	숫자10자리일 때 "###-##-#####"로 표현하며 확인 시 숫자10자리
CardNo	숫자14자리일 때 "####-####-####-####"로 표현하며 확인 시 숫자14자리 반환
PostNo	숫자6자리일 때 "###-###"로 표현하며 확인 시 숫자6자리 반환
Integer	"#,##0" 로 표현하며 확인 시 숫자로 반환
NullInteger	"#,###" 로 표현하며 확인 시 숫자로 반환
Float	"#,##0."+Point개수로 표현하며 확인 시 숫자로 반환
NullFloat	"#,##.#."+Point개수로 표현하며 확인 시 숫자로 반환

➤ **Syntax**

Syntax	Get	ObjId. GetCellValue (Row, Col)
--------	-----	---------------------------------------

➤ **Info**

Return	String, 셀에 설정된 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index



Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName
-----	-----------------	----	-----------------------------------

➤ **Example**

```
//날짜 셀에 값 확인, 결과는 2011/07/15임
alert(mySheet.GetCellValue(1, 4));
```

➤ **Syntax**

Syntax	Set	ObjId. SetCellValue (Row, Col, Value)
--------	-----	--

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long /String	필수	해당 셀의 Column Index 또는 SaveName
Value	String	필수	셀에 설정할 값
Flag	Boolean	선택	OnChange 이벤트 발생 여부 (Default=1)

➤ **Example**

```
//상태 셀을 "삭제" 상태로 설정
mySheet.SetCellValue(1, 0, "D");

//CheckBox에 체크되도록 설정
mySheet.SetCellValue(1, 1, 1);

//숫자 셀에 값 설정, 표시되는 값은 12,345임
mySheet.SetCellValue(1, 2, 12345);

//콤보 셀에 값 설정, 표시되는 값은 콤보 텍스트 임
mySheet.SetCellValue(1, 3, "01");
```



```
//날짜 셀에 값 설정, 표시되는 값은 "2011/07/15" 임  
mySheet.SetCellValue(1, 4, "2011/07/15");
```

```
//소수점 숫자 셀에 값 설정, 소수점 3자리이면 표시되는 값은 123.450 임  
// OnChange 이벤트 발생  
mySheet.SetCellValue(1,5, 123.450);
```

```
// OnChange 이벤트 발생하지 않음  
mySheet.SetCellValue(1,5, 123.450, 0);
```

➤ **제공 버전**
7.0.0.0

CheckAll Method

➤ 기능

CheckBox가 존재하는 컬럼의 모든 CheckBox를 전체 Check하거나 UnCheck 처리한다.

이 속성은 사용자가 헤더의 전체 CheckBox를 누른 것과 같은 처리를 하며, 설정 값을 0으로 하면 전체 UnCheck 처리하고, 1로 하면 전체 Check 처리한다. 이외의 값을 설정하면 이전 전체 Check 설정의 반대로 처리된다.

이 속성은 Type이 DelCheck와 CheckBox인 경우 처리하며, Edit 가능한 셀의 CheckBox만 처리한다. 처리를 완료 하면 각 데이터 행 단위로 OnChange 이벤트가 발생한다.

2개행 이상의 단위데이터행 구조인 경우 Col 인자에 Index를 설정할 경우 모든행에 대해서 처리를 하고 SaveName으로 설정할 경우 해당 SaveName인 행에 대해서만 처리 한다.

이 속성에 설정하는 값에 따라 CheckBox는 아래와 같이 처리된다.

설정값	설명
0	전체 UnCheck하기
1	전체 Check 하기
그외값	이전 설정에 반대로 처리하기

➤ Syntax

Syntax	ObjId. CheckAll (Col, Value, [OnChangeEvent])
--------	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Col	Long / String	필수	특정 컬럼의 Column Index 또는 SaveName
Value	Integer	필수	0 : 전체체크해제,



			1 : 전체체크, 그외 : 이전값 반대
OnChangeEvent	Boolean	선택	OnChange 이벤트 발생 여부 Default=1

➤ **Example**

```
// 모두 선택하기  
mySheet.CheckAll(1, 1);  
  
// 모두 선택 취소하기  
mySheet.CheckAll(1, 0);
```

➤ **제공 버전**

7.0.0.0

CheckedRows Method

➤ 기능

해당 컬럼에 존재하는 체크박스 중 체크된 행의 개수를 반환한다.

Opt 인자의 RowLevel, ParentRow, Recursive 인자는 트리구조의 시트에서 트리 기준 컬럼에 대한 체크박스 사용 모드에서만 유효한 인자 속성이다.

RowLevel, ParentRow 속성은 동시에 같이 사용할 수 없으며 두 속성 모두 설정이 있는 경우 ParentRow 속성을 우선 처리 한다.

2개행 이상의 단위데이터행 구조인 경우 Col 인자에 Index를 설정할 경우 모든행에 대해서 처리를 하고 SaveName으로 설정할 경우 해당 SaveName인 행에 대해서만 처리 한다.

➤ Syntax

Syntax	ObjId.CheckedRows(Col, [Opt])
--------	-------------------------------

➤ Info

Return	Long, 특정 컬럼에 체크된 행의 개수		
Parameter	Type	필수여부	설 명
Col	Long / String	필수	특정 컬럼의 Column Index 또는 SaveName
Opt.RowLevel	Number	선택	처리 대상 트리 레벨 (Default:0)
Opt.ParentRow	Index	선택	처리 대상 부모 행의 Index
Opt.Recursive	Boolean	선택	하위 노드에 대한 재귀 처리 여부 (Default:1)

➤ Example

```
var cnt1 = mySheet.CheckedRows(1);
console.log("체크된 행의 개수는 ", cnt1, "개 입니다.");
```



```
var cnt2 = mySheet.CheckedRows("chkData");  
console.log("체크된 행의 개수는 ", cnt2, "개 입니다.");  
  
// 트리기준 컬럼의 체크박스에 대한 사용  
// 트리 레벨이 2인 행 가운데 체크된 개수 확인  
var res = mySheet.CheckedRows(1, {RowLevel:2, Recursive:0});  
  
// Index가 5인 행의 자식행 가운데 체크된 개수 확인  
var arRes = mySheet.CheckedRows (1, {ParentRow:5, Recursive:1});
```

➤ **제공 버전**

7.0.0.0	
7.0.9.0	트리 구조의 트리체크에 대한 처리 지원 (RowLevel, ParaenRow 속성 인자 추가)

CheckReverse Method

➤ 기능

특정 컬럼에 존재하는 체크박스에 체크된 것은 체크를 풀어주고, 체크되지 않은것은 체크한다.

2개행 이상의 단위데이터행 구조인 경우 Col 인자에 Index를 설정할 경우 모든행에 대해서 처리를 하고 SaveName으로 설정할 경우 해당 SaveName인 행에 대해서만 처리 한다.

➤ Syntax

Syntax	ObjId. CheckReverse (Col, [Editable], [Event])
--------	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Col	Long / String	필수	특정 컬럼의 Column Index 또는 SaveName
Editable	Boolean	선택	편집가능 불가능을 확인하여 편집가능일 때만 변경한다. Default=0
Event	Boolean	선택	체크가 바뀔때 OnChange 이벤트를 발생시킬것인가 여부, Default=0

➤ Example

<pre>// 1컬럼의 체크를 반전한다. mySheet.CheckReverse(1); // Edit를 체크하고, Event를 발생시킨다. mySheet.CheckReverse(1, 1, 1);</pre>
--

➤ 제공 버전



7.0.0.0



ClearHeaderCheck Method

➤ 기능

헤더의 모든 전체체크박스의 값을 언체크로 초기화 한다.

HeaderCheck 메소드와 마찬가지로 전체체크의 기능 처리는 하지 않고 단순 언체크 처리만 한다.

➤ Syntax

Syntax	ObjId.ClearHeaderCheck()
--------	--------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// RemoveAll 호출전에 전체체크박스를 초기화 한다..  
mySheet.ClearHeaderCheck();  
mySheet.RemoveAll();
```

➤ 제공 버전

7.0.0.0



ClearSelection Method

➤ 기능

마우스 드래그를 이용한 선택 영역을 초기화 한다.

➤ Syntax

Syntax	ObjId. ClearSelection()
--------	--------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// 선택 영역을 초기화 한다.  
mySheet.ClearSelection()
```

➤ 제공 버전

7.0.0.0

ClipCopyMode Method

➤ 기능

문자열을 선택하고 Ctrl + C 를 하면 선택된 내용이 클립보드에 복사된다. 이렇게 내용을 복사할 때, Ctrl + C 키보드를 사용하여 Clipboard에 복사하는 방식을 확인하거나 설정한다.

설정값에 따른 처리 방식은 다음과 같다.

설정값	설명
-1	시트내에서 클립보드 복사 허용 안함
0	포커스만 있는 경우 포커스 셀 복사, 다수의 셀 선택시 다수의 셀 값 복사. Default
1	포커스만 있는 경우 포커스 행 복사, 다수의 셀 선택시 다수의 셀 값 복사.

➤ Syntax

Syntax	Get	ObjId.GetClipCopyMode()
--------	-----	-------------------------

➤ Info

Return	Integer, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

mySheet.GetClipCopyMode();

➤ Syntax

Syntax	Set	ObjId.SetClipCopyMode(Mode)
--------	-----	-----------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명



Mode	Integer	필수	복사 방법에 따른 설정값
------	---------	----	---------------

➤ **Example**

```
// 포커스셀 값이나 선택셀 값 모두 복사  
mySheet.SetClipCopyMode(0);  
  
// 포커스행 복사나 선택셀 값 모두 복사  
mySheet.SetClipPasteMode(1);
```

➤ **제공 버전**

7.0.0.0	
7.0.2.0	클립보드 복사를 허용하지 않는 모드 추가 (Mode:-1)

ClipPasteMode Method

➤ 기능

문자열을 선택하고 Ctrl + C 를 하면 선택된 내용이 클립보드에 복사된다. 이렇게 복사된 내용을 Ctrl + V 키보드를 사용하여 Sheet위에 붙여넣을 때 처리 방식을 확인하거나 설정한다. 붙여넣기 하는 셀에 데이터의 유형이 맞지 않거나, 그 외의 오류가 발생할 수 있는 셀에는 붙여넣기가 되지 않는다.

설정값에 따른 처리 방식은 다음과 같다.

설정값	설명
-1	붙여넣기 기능 사용안함
0	선택된 1개 셀에만 붙여넣는다, Default
1	선택된 셀을 기준으로 여러개 셀에 붙여넣는다. 주로 엑셀에서 복사된 글자를 붙이는 경우와 같이 개행문자("\r\n")는 행을 구분하고, 탭문자("\t")는 컬럼을 구분한다.
2	설정값1과 동일하게 여러개 셀을 붙여넣으며, 붙여넣을 데이터가 남아 있는데 데이터 행이 모자랄 경우 행을 추가하며 붙여넣는다
3	붙여넣을 데이터의 행만큼 행을 추가하면서 붙여넣기 한다..

➤ Syntax

Syntax	Get	ObjId. GetClipPasteMode()
--------	-----	----------------------------------

➤ Info

Return	Integer, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

mySheet.GetClipPasteMode();

➤ Syntax

Syntax	Set	ObjId. SetClipPasteMode (Mode)
--------	-----	---------------------------------------



➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Mode	Integer	필수	붙여넣기 방법에 따른 설정값

➤ **Example**

```
// 1개 셀에 붙이기 모드  
mySheet.SetClipPasteMode(0);  
  
// 여러개 셀에 붙이기 모드  
mySheet.SetClipPasteMode(1);
```

➤ **제공 버전**

7.0.0.0

ColBackColor Method

➤ 기능

대상 컬럼 전체의 배경색을 설정하거나 확인한다. 헤더 행을 제외한 데이터 행의 배경색만 처리한다.

컬럼이 존재하지 않는 경우 에러메시지 없이 글자색 설정은 취소된다.

색상 설정은 WebColor 또는 Basic 16 Color 문자열로 설정한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 모든행에 대해서 처리를 하고 SaveName으로 설정할 경우 해당 SaveName이 포함된 행에 대해서만 처리 한다.

➤ Syntax

Syntax	Get	ObjId. GetColBackColor (Col)
--------	-----	-------------------------------------

➤ Info

Return	String, 설정된 색상값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Col	Long /String	필수	특정 컬럼의 Column Index 또는 SaveName

➤ Example

```
// 2컬럼 배경색을 확인한다.
mySheet.GetColBackColor(2);
//SaveName이 sa_Name 인 컬럼의 배경색을 확인한다.
mySheet.GetColBackColor("sa_Name");
```

➤ Syntax

Syntax	Set	ObjId. SetColBackColor (Col, Color)
--------	-----	--

➤ Info



Return	None		
Parameter	Type	필수여부	설 명
Col	Long /String	필수	특정 컬럼의 Column Index 또는 SaveName
Color	String	필수	WebColor 색상 값

➤ **Example**

```
//컬럼 배경색을 회색으로 설정한다.  
mySheet.SetColBackColor(1, "#ADADAD");  
//SaveName이 "sa_DistQty"인컬럼 배경색을 설정한다.  
mySheet.SetColBackColor("sa_DistQty", "#FFDDDD");
```

➤ **제공 버전**

7.0.0.0

ColCondProperty Method

➤ 기능

숫자형 타입에서 특정 컬럼의 각 셀의 데이터값이 조건에 맞을 경우 속성을 설정, 설정한 조건을 확인한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 모든행에 대해서 처리를 하고 SaveName으로 설정할 경우 해당 SaveName이 포함된 행에 대해서만 처리 한다.

이 함수를 통해 설정이 가능한 속성은 다음과 같다.

속성	타입	설명
BackColorT	String	조건이 True 인경우 배경색상
BackColorF	String	조건이 False 인경우 배경색상
FontColorT	String	조건이 True 인경우 폰트색상
FontColorF	String	조건이 False 인경우 폰트색상
EditT	Boolean	조건이 True 인경우 편집여부
EditF	Boolean	조건이 False 인경우 편집여부
CursorT	String	조건이 True 인경우 마우스 포인터 설정
CursorF	String	조건이 False 인경우 마우스 포인터 설정

설정할수 있는 마우스 모양은 아래와 같다.

설정값	마우스 모양
Default	기본 화살표 모양
Pointer	손가락 모양

➤ Syntax

Syntax	Get	ObjId. GetColCondProperty (Col)
--------	-----	--

➤ Info

Return	String, 설정한 조건 (Get Method 인 경우)
--------	----------------------------------



Parameter	Type	필수여부	설 명
Col	Long / String	필수	대상이 되는 컬럼 index 또는 SaveName

➤ **Example**

```
// 설정한 조건을 확인한다.
mySheet.GetColCondProperty(13)
```

➤ **Syntax**

Syntax	Set	ObjId. SetColCondProperty (Col, Cond, Prop)
--------	-----	--

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Col	Long / String	필수	대상이 되는 컬럼 index 또는 SaveName
Cond	String	필수	설정할 조건 (예: "%d > 1000")
Prop	Object	선택	설정할 컬럼 속성

➤ **Example**

```
// 13번째 컬럼의 각 셀 데이터 값이 1000 보다 클경우
mySheet.SetColCondProperty(13,"%d>1000",{BackColorT:"#00ff00",FontColorT:
:"#ffff00", EditT : false, BackColorF : "#0000ff",FontColorF:"#ffffff", EditF : true,
CursorT:"Pointer", CursorF:"Default"})
```

➤ **제공 버전**

7.0.0.0

ColEditable Method

➤ 기능

특정 컬럼의 Edit 가능 여부를 확인하거나 설정한다.

특정 컬럼의 Edit 가능 여부는 전체 Edit 가능 여부가 가능일 때 변경 가능하다. 단, ColEditable 설정값이 불가능인 경우는 RowEditable 설정이 무시된다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 모든행에 대해서 처리를 하고 SaveName으로 설정할 경우 해당 SaveName이 포함된 행에 대해서만 처리 한다.

특정 컬럼의 Edit 가능 여부는 다음과 같이 결정된다.

Editable	ColEditable	RowEditable	CellEditable	셀의 Edit 가능 여부
불가	무관	무관	무관	불가
가능	불가	불가	가능/불가	가능/불가
가능	가능	가능/불가	가능/불가	가능/불가

➤ Syntax

Syntax	Get	ObjId. GetColEditable (Col)
--------	-----	------------------------------------

➤ Info

Return	Boolean, Edit 가능 여부 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Col	Long /String	필수	특정 컬럼의 Column Index 또는 SaveName

➤ Example

//5컬럼의 Edit 가능 여부를 확인한다. mySheet. GetColEditable (5);
--



➤ **Syntax**

Syntax	Set	ObjId. SetColEditable (Col, Editable)
--------	-----	--

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Col	Long /String	필수	특정 컬럼의 Column Index 또는 SaveName
Editable	Boolean	필수	특정컬럼의 Edit 가능 여부 (Set일때만 사용)

➤ **Example**

<pre>//5컬럼의 Edit 가능 여부를 불가로 설정한다. mySheet.SetColEditable(5,0); //5컬럼의 Edit 가능 여부를 가능으로 설정한다. mySheet. SetColEditable (5,1);</pre>
--

➤ **제공 버전**

7.0.0.0

ColDelete Method

➤ 기능

특정 컬럼을 삭제 한다.

Col 인자를 설정하지 않는 경우 마지막 컬럼을 삭제 한다.

아래과 같은 조건에 해당하는 컬럼을 삭제가 불가능 하다.

- ✓ 컬럼 타입이 Seq, Status, DelCheck 인 경우
- ✓ 타 컬럼의 계산식(CalcLogic)에 사용되는 컬럼인 경우
- ✓ 트리 기준 컬럼(TreeCol 속성 설정이 있는 컬럼)인 경우
- ✓ 단위데이터행 구조의 FixedMerge에 포함된 컬럼인 경우

➤ Syntax

Syntax	ObjId.ColDelete([Col])
--------	------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Col	Number /String	선택	대상 컬럼의 Index 또는 SaveName (Default : 마지막 컬럼)

➤ Example

```
// Index값이 3인 컬럼을 삭제 한다.
mySheet.ColDelete(3);

// SaveName이 "sText"인 컬럼을 삭제 한다.
mySheet.ColDelete("sText");

// 마지막 컬럼을 삭제 한다.
mySheet.ColDelete();
```

➤ 제공 버전



7.0.0.0

ColFontBold Method

➤ 기능

대상 컬럼 전체의 Bold(굵은 글자체)를 설정하거나 확인한다. 헤더 행을 제외한 데이터 행만 처리한다.

컬럼이 존재하지 않는 경우 에러메시지 없이 설정은 취소된다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 모든행에 대해서 처리를 하고 SaveName으로 설정할 경우 해당 SaveName이 포함된 행에 대해서만 처리 한다.

➤ Syntax

Syntax	Get	ObjId. GetColFontBold (Col)
--------	-----	------------------------------------

➤ Info

Return	Boolean, Bold 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Col	Number /String	필수	특정 컬럼의 Index 또는 SaveName

➤ Example

```
//인덱스가 1인 컬럼에 Bold 설정여부를 확인한다.
Alert(mySheet.GetColFontBold(1));
```

➤ Syntax

Syntax	Set	ObjId. SetColFontBold (Col, Bold)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Col	Number	필수	특정 컬럼의 Index 또는 SaveName



	/String		
Bold	Boolean	필수	Bold 여부

➤ **Example**

<pre>//컬럼 전체 글자에 Bold를 설정한다. mySheet.SetColFontBold(1, 1);</pre>
--

➤ **제공 버전**

7.0.0.0

ColFontColor Method

➤ 기능

대상 컬럼 전체의 글자색을 설정하거나 확인한다. 헤더 행을 제외한 데이터 행의 글자색만 처리한다.

컬럼이 존재하지 않는 경우 에러메시지 없이 글자색 설정은 취소된다.

색상 설정은 WebColor로 설정한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 모든행에 대해서 처리를 하고 SaveName으로 설정할 경우 해당 SaveName이 포함된 행에 대해서만 처리 한다.

➤ Syntax

Syntax	Get	ObjId. GetColFontColor (Col)
--------	-----	-------------------------------------

➤ Info

Return	String, 설정된 색상값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Col	Long /String	필수	특정 컬럼의 Column Index 또는 SaveName

➤ Example

```
//2컬럼 글자색을 확인한다..
alert(mySheet.GetColFontColor(2));
```

➤ Syntax

Syntax	Set	ObjId. SetColFontColor (Col, Color)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명



Col	Long /String	필수	특정 컬럼의 Column Index 또는 SaveName
Color	String	필수	색상 값

➤ **Example**

```
//컬럼 글자색을 회색으로 설정한다.  
mySheet.SetColFontColor(1, "#FF0000");  
//2컬럼 글자색을 1컬럼 글자색으로 설정한다.  
mySheet. SetColFontColor(2, mySheet.GetColFontColor(1));
```

➤ **제공 버전**

7.0.0.0

ColFontUnderline Method

➤ 기능

대상 컬럼 전체의 글자 언더라인을 설정하거나 확인한다. 헤더 행을 제외한 데이터 행만 처리한다.

컬럼이 존재하지 않는 경우 에러메시지 없이 언더라인 설정은 취소된다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 모든행에 대해서 처리를 하고 SaveName으로 설정할 경우 해당 SaveName이 포함된 행에 대해서만 처리 한다.

➤ Syntax

Syntax	Get	ObjId. GetColFontUnderline (Col)
--------	-----	---

➤ Info

Return	Boolean, 밑줄 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Col	Number /String	필수	특정 컬럼의 Index 또는 SaveName

➤ Example

```
//컬럼 전체 글자에 언더라인 설정여부를 확인한다.
Alert(mySheet.GetColFontUnderline(1));
```

➤ Syntax

Syntax	Set	ObjId. SetColFontUnderline (Col, Underline)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Col	Number	필수	특정 컬럼의 Index 또는 SaveName



	/String		
Underline	Boolean	필수	밀줄 여부

➤ **Example**

<pre>//컬럼 전체 글자에 언더라인을 설정한다. mySheet.SetColFontUnderline(1, 1);</pre>

➤ **제공 버전**

7.0.0.0



ColHidden Method

➤ 기능

컬럼의 숨김여부를 설정하거나 확인한다.

➤ Syntax

Syntax	Get	ObjId. GetColHidden (Col)
--------	-----	----------------------------------

➤ Info

Return	Boolean, 숨김 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Col	Long /String	필수	특정 컬럼의 Column Index 또는 SaveName

➤ Example

```
//해당 컬럼이 숨겨졌는지 확인한다.
var isHide = mySheet.GetColHidden(12);
var isHide2 = mySheet.GetColHidden("sa_Personal_Id");
```

➤ Syntax

Syntax	Set	ObjId. SetColHidden (Col, Hidden)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Col	Long /String	필수	특정 컬럼의 Column Index 또는 SaveName
Hidden	Boolean	필수	숨김여부

➤ Example



// 컬럼이 숨겨진 여부를 확인하여 숨겨진 경우 표시되도록 설정한다.

```
If(mySheet.GetColHidden(1) == 1) {  
    mySheet.SetColHidden(1, 0);  
}
```

➤ **제공 버전**

7.0.0.0

CollInsert Method

➤ 기능

특정 위치에 컬럼을 생성 한다.

생성위치(Pos)를 설정한 경우 해당 위치에 생성하고 설정하지 않은 경우 마지막 위치에 생성 한다.

컬럼타입이 Seq, Status, DelCheck인 컬럼은 생성이 불가능 하며 컬럼생성시 CalcLogic 속성은 사용이 불가능 하다.

생성할 컬럼의 설정 속성 집합 객체는 [InitColumns Method의 내용을 참조](#) 한다. 컬럼의 설정 속성을 설정하지 않은 경우 "Text" 타입의 컬럼으로 생성 한다.

➤ Syntax

Syntax	ObjId.CollInsert([Info])
--------	--------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Info.Pos	Number /String	선택	생성할 위치의 컬럼 Index 또는 SaveName (Default : 마지막 컬럼)
Info.Header.Text	String	선택	헤더에 설정할 문자열 (헤더가 2줄 이상인 경우 구분자 " "로 연결한 문자열로 설정) (Default : "")
Info.Header.Align	String	선택	헤더 문자열의 정렬값 (Default : "Center")
Info.Col	Object	선택	생성할 컬럼의 설정 속성 집합 객체

➤ Example



```
// 마지막 위치에 기본("Text" 컬럼타입) 컬럼을 생성 한다.  
var info = {  
    Header : {Text : "기본타입"}  
};  
mySheet.ColInsert(info);  
  
// Index 3의 위치에 "CheckBox" 타입의 컬럼을 생성 한다.  
var info = {  
    Pos:3,  
    Header:{Text: "체크박스",Align: "Left"},  
    Col:[{  
        Type: "CheckBox",  
        Width:60,  
        SaveName: "sCheckBox"  
    }]  
};  
mySheet.ColInsert(info);
```

➤ **제공 버전**

7.0.0.0



ColLeft Method

➤ 기능

컬럼의 왼쪽 위치를 확인한다.

➤ Syntax

Syntax	ObjId.ColLeft(Col)
--------	--------------------

➤ Info

Return	Long, 특정컬럼 왼쪽 위치 값		
Parameter	Type	필수여부	설 명
Col	Long /String	필수	특정 컬럼의 Column Index 또는 SaveName

➤ Example

```
//컬럼의 왼쪽 위치를 파악한다.  
var iLeft = mySheet.ColLeft(1);
```

➤ 제공 버전

7.0.0.0



ColSaveName Method

➤ 기능

특정 컬럼 Index에 해당하는 InitColumns 함수에서 설정한 SaveName을 확인한다.

➤ Syntax

Syntax	ObjId.ColSaveName(DataRow, Col)
--------	---------------------------------

➤ Info

Return	String, 특정컬럼의 SaveName		
Parameter	Type	필수여부	설 명
DataRow	Number	필수	단위 데이터행의 Index
Col	Number	필수	특정 컬럼의 Index

➤ Example

```
//컬럼의 SaveName을 가져온다.  
var sSaveName = mySheet.ColSaveName(0, 1);
```

➤ 제공 버전

7.0.0.0

ColumnSort Method

➤ 기능

단일 또는 다중 컬럼의 데이터를 정렬한다.

여러 개 컬럼을 정렬하는 경우 컬럼을 정렬 순서에 맞게 "|"로 연결하여 설정하면 설정한 앞컬럼 부터 순서에 맞게 정렬한다. 기본적으로 Column Index가 작은 컬럼부터 큰 컬럼 순서로 정렬한다.

단위데이터행 정렬 시 인덱스를 설정하면 단위데이터행 첫번째 행에 대해서만 처리되며, 두번째 행 이후부터는 SaveName으로 설정하여야 한다.

ColSort 인자는 값을 공백으로 설정하면 모든 컬럼이 Sort 인자에 설정된 소트 방향 순서로 소트되고, 각 컬럼별로 소트 방향을 달리 설정할 때 사용한다.

KeepColOrder 인자는 Col 인자에 설정된 컬럼이 소트되는 순서를 의미하는것으로 인자값이 0 인 경우 컬럼 인덱스 순서로 소트되고, 인자값이 1 인 경우 설정된 컬럼 순서 그대로 소트된다. 이 인자의 기본값을 0 이다.

➤ Syntax

Syntax	ObjId.ColumnSort(Col, [Sort], [ColSort],[KeepColOrder])
--------	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Col	Long / String	필수	소트할 컬럼 Index 또는 SaveName을 " "로 조합한 문자열
Sort	String	선택	"ASC" 또는 "DESC", Default ="ASC"
ColSort	String	선택	각 컬럼의 소트 방향을 " "로 연결한 문자열, Default=""
KeepColOrder	Boolean	선택	Cols파라미터에서 지정된 순서대로 소트할 것인지 선택, Default=0



➤ **Example**

```
//6컬럼만 내림차순으로 정렬
```

```
mySheet.ColumnSort("6", "DESC")
```

```
//4컬럼을 중심으로 먼저 오름차순 정렬하고, 그 안에서 5컬럼 오름차순 정렬
```

```
mySheet.ColumnSort("4|5")
```

```
//2,3,4컬럼 순서로 정렬하고, 내림차순으로 정렬한다.
```

```
mySheet.ColumnSort("2|3|4", "DESC");
```

```
//3,2,4컬럼을 2,3,4컬럼인덱스 순서대로 내림차순, 오름차순, 오름차순으로 정렬한다.
```

```
mySheet.ColumnSort("3|2|4", "DESC","ASC|DESC|ASC");
```

```
//3,2,4컬럼을 3,2,4컬럼 지정된 순서대로 오름차순, 내림차순, 오름차순으로 정렬한다.
```

```
mySheet.ColumnSort("3|2|4", " DESC ","ASC|DESC|ASC", 1);
```

➤ **제공 버전**

7.0.0.0

ColValueDup Method

➤ 기능

특정 컬럼 내에 중복된 값이 존재하는지 여부를 확인한다.

중복된 값이 존재하는 경우 두 번째 중복된 값의 Row Index를 반환한다.

컬럼을 하나만 사용할 수도 있고, 여러 개를 사용할 경우 "|" 문자로 조합하여 설정하면, 여러 컬럼의 값이 모두 같은 경우 해당 행의 Index를 반환한다.

IncludeDelRow 인자가 1인 경우 트랜잭션 상태가 삭제인 행을 포함하여 중복 행을 체크하고, IncludeDelRow인자가 0인 경우 트랜잭션 상태가 삭제인 행을 제외하고 중복행을 체크한다.

IncludeEmpty 인자가 0인 경우 비교 대상 셀의 모든 값이 빈값인 경우 해당 행은 비교 대상에서 제외 된다.

2개행 이상의 단위데이터행 구조에서 반환되는 행의 Index는 단위데이터행 구조의 기준이 되는 첫번째 행의 Index를 반환하게 된다. 또한, Col 인자에 Index를 설정할 경우 단위데이터 행의 컬럼 Index 위치의 모든 컬럼에 대해서 처리를 하고 SaveName으로 설정할 경우 해당 SaveName의 컬럼 대해서만 처리를 한다.

중복된 값이 존재하지 않는 경우 -1을 반환한다.

➤ Syntax

Syntax	ObjId.ColValueDup(Cols, [Opt])
--------	--------------------------------

➤ Info

Return	Number, 중복 행의 Index		
Parameter	Type	필수여부	설 명
Cols	Number / String	필수	컬럼의 Index 또는 SaveName을 구분자 " "로 연결한 문자열



Opt.CaseSensitive	Boolean	선택	대/소문자 구분 비교 여부 (Default=1)
Opt.IncludeDelRow	Boolean	선택	트랜잭션 상태가 삭제인 행을 포함 여부. (Default=1)
Opt.IncludeSumRow	Boolean	선택	합계/소계/누계 행 포함 여부. (Default=1)
Opt.IncludeEmpty	Boolean	선택	빈값 포함 여부 (Default=1)

➤ **Example**

```
//1컬럼에 중복된 값이 존재하는 행의 Index 확인
var Row = mySheet.ColValueDup("1");

//2,3,7 컬럼에 중복된 값이 존재하는 행의 Index 확인
var Row = mySheet.ColValueDup("2|3|7");

//삭제된 행과 빈값을 제외하고 중복 체크하기
var Row = mySheet.ColValueDup("2|3|7",{
    "IncludeDelRow" : 0,
    "IncludeEmpty" : 0
});
```

➤ **제공 버전**

7.0.0.0	
7.0.12.3	인자 구조 변경, IncludeEmpty 인자 속성 추가

ColValueDupRows Method

➤ 기능

중복된 모든 행을 ","로 조합하여 문자열로 반환 한다.

ColValueDup 메소드는 처음으로 중복된 행의 Index만 확인할수 있지만 이 속성은 중복이 시작된행과 모든 중복된 행번호를 확인할 수 있는 기능이다.

예를 들어 Sheet에 다음과 같이 "기간구분"과 "콘도종류" 2개의 컬럼으로 구성되어 있는 구조라고 할때, ColValueDup=4이다. 1행으로 검사 후 첫 발견된 데이터이므로 결과는 4행이다. 동일한 조건에서 이 속성의 결과는 "3,4,7,8" 이다.

NO	기간구분	콘도종류
1	평일	한화콘도
2	주말	한국콘도
3	주말	한국콘도
4	평일	한화콘도
5	여름성수기	글로리콘도
6	겨울성수기	삼립하일라
7	여름성수기	글로리콘도
8	여름성수기	글로리콘도
총8건		

IncludeDelRow 인자는 트랜잭션 상태가 삭제인 행을 중복 검사 대상에 포함할 지 여부를 설정한다. 이 인자값이 true인 경우 삭제인 행을 포함하고, false인 경우 삭제인 행을 포함하지 않는다. 위의 경우 1행이 삭제 상태이고, IncludeDelRow 인자를 false로 설정하면 ColValueDup=3이고, 이 속성의 결과는 "3,7,8"이다.

IncludeFirstRow 인자는 중복이 발견된 행의 최초의 기준행을 포함할지 여부를 설정한다. 이 인자 값이 false이면 최초의 기준행을 포함하지 않고, 이 인자 값이 true이면 최초의 기준행들을 ","로 구분하여 문자열로 만들고, 기준행으로 중복된 행들도 ","로 구분하여 문자열로 만든 후 두개 문자열을 "|"로 조합하여 반환한다.

위의 경우 최초의 기준행들은 "1,2,5"이고, 중복된 행들은 "3,4,7,8"이다. 따라서



결과는 "1,2,5|3,4,7,8"이다.

StartRow 인자와 **EndRow** 인자는 중복을 검사할 행의 영역을 행Index로 설정하며, 기본적으로 모든 데이터 행을 중복 검사 대상으로 처리한다. 위의 예에서 StartRow=1, EndRow=4로 제한한다면 결과는 "3,4"이다.
중복된 값이 존재하지 않는 경우 ""을 반환한다.

IncludeEmpty 인자가 0인 경우 비교 대상 셀의 모든 값이 빈값인 경우 해당 행은 비교 대상에서 제외 된다.

2개행 이상의 단위데이터행 구조에서 반환되는 행의 Index는 단위데이터행 구조의 기준이 되는 첫번째 행의 Index를 반환하게 된다. 또한, Col 인자에 Index를 설정할 경우 단위데이터 행의 컬럼 Index 위치의 모든 컬럼에 대해서 처리를 하고 SaveName으로 설정할 경우 해당 SaveName의 컬럼 대해서만 처리를 한다.

➤ Syntax

Syntax	ObjId.ColValueDupRows(Cols, [Opt])
--------	------------------------------------

➤ Info

Return	String, 중복된 모든 행을 ","로 조합한 문자열		
Parameter	Type	필수여부	설 명
Cols	Number / String	필수	컬럼의 Index 또는 SaveName을 구분자 " "로 연결한 문자열
Opt.CaseSensitive	Boolean	선택	대/소문자 구분 비교 여부 (Default=1)
Opt.IncludeDelRow	Boolean	선택	트랜잭션 상태가 삭제인 행을 포함 여부. (Default=1)
Opt.IncludeSumRow	Boolean	선택	합계/소계/누계 행 포함 여부. (Default=1)
Opt.IncludeEmpty	Boolean	선택	빈값 포함 여부 (Default=1)

Opt.IncludeFirstRow	Boolean	선택	중복된 행의 최초의 행을 포함할지 여부. (Default=0)
Opt.StartRow	Number	선택	중복 검사를 수행 하고자 하는 부분의 첫행의 Index. (Default="첫 행")
Opt.EndRow	Number	선택	중복 검사를 수행 하고자 하는 부분의 마지막행 Index. (Default="마지막 행")

➤ **Example**

//6컬럼,7컬럼의 글자가 중복된 행들(삭제행포함, 최초행미포함, 전체 데이터 영역 검사)

```
var duprows1 = mySheet.ColValueDupRows("6|7");
```

//4컬럼,5컬럼에 대해 첫번째 행부터 50번째 행까지의 중복검사(삭제행제외, 최초행포함, 1행~50행)

```
var duprows2 = mySheet.ColValueDupRows("4|5",{
    "IncludeDelRow" : 0,
    "IncludeFirstRow" : 1,
    "StartRow" : 1,
    "EndRow" : 50
});
```

//가져온 행을 배열로 만든다.

```
var arrRow = duprows1.split(",");
for (idx=0; idx<arrRow.length-1; idx++){ alert(arrRow[idx] + "행"); }
```

➤ **제공 버전**

7.0.0.0	
7.0.12.3	인자 구조 변경, CaseSensitive, IncludeEmpty 인자 속성 추가



ColWidth Method

➤ 기능

특정 컬럼의 너비를 설정하거나 확인한다.

픽셀 단위의 너비를 설정할 수 있고, 값을 0으로 설정하면 해당 컬럼 내의 Text 중 가장 긴 값에 맞게 너비를 자동 조정한다.

컬럼이 존재하지 않는 경우 에러메시지는 표시하지 않지만 처리는 취소된다.

➤ Syntax

Syntax	Get	ObjId. GetColWidth (Col)
--------	-----	---------------------------------

➤ Info

Return	Integer, 특정컬럼의 너비 픽셀 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Col	Long/ String	필수	특정 컬럼의 Column Index 또는 SaveName

➤ Example

// 1컬럼의 너비값을 확인 mySheet.GetColWidth(1);
--

➤ Syntax

Syntax	Set	ObjId. SetColWidth (Col, Width)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Col	Long/ String	필수	특정 컬럼의 Column Index 또는 SaveName
Width	Integer	필수	너비 픽셀 값



➤ **Example**

```
//50 픽셀로 너비 수정  
mySheet.SetColWidth(1, 50);  
  
//컬럼 내의 최대 글자 너비로 자동 변경  
mySheet.SetColWidth(2, 0);  
  
//3컬럼의 너비를 2컬럼의 너비와 동일하게 변경  
mySheet.SetColWidth(3, mySheet.GetColWidth(2));
```

➤ **제공 버전**

7.0.0.0



ComboOpenMode Method

➤ 기능

Combo, ComboEdit 컬럼을 한번 클릭으로 펼쳐줄지에 대해 확인하거나 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetComboOpenMode()
--------	-----	----------------------------------

➤ Info

Return	Boolean, 현재 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//ComboOpenMode 설정되어 있는 값을 확인한다.  
mySheet.GetComboOpenMode();
```

➤ Syntax

Syntax	Set	ObjId. SetComboOpenMode(mode)
--------	-----	--------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
mode	Boolean	필수	0 (Default) 콤보를 한번 클릭하면 펼쳐지지 않고 포커스만 얻는다.(Default)
			1 콤보를 한번 클릭으로 펼친다.

➤ Example



//콤보를 한번 클릭으로 연다.

Mysheet.SetComboOpenMode(1);

➤ **제공 버전**

7.0.0.0



ComputeSum Method

➤ 기능

특정 영역의 합계를 계산하여 반환한다.

이 속성은 특정 컬럼의 합계만 계산할 수도 있고, 공식을 사용하여 계산 되어진 값의 합계를 구할수도 있다. 특정 영역을 지정하지 않으면 모든 데이터의 합계를 계산하여 반환한다.

➤ Syntax

Syntax	ObjId. ComputeSum (CalcuLogic,[FirstRow],[LastRow],[isFullSum])
--------	--

➤ Info

Return	Double, 계산되어진 특정영역의 합계		
Parameter	Type	필수여부	설 명
CalcuLogic	String	필수	계산 공식, 공식에 다른 컬럼값이 사용되는 경우 반드시 " "로 감싸서 사용한다.
FirstRow	Long	선택	계산 대상행의 시작 index, Default=-1
LastRow	Long	선택	계산 대상행의 마지막 index, Default=-1
isFullSum	Boolean	선택	해당 계산식을 소계행도 포함할지의 여부, Default=1 1 : 소계행을 계산식에 포함 0 : 소계행은 계산식에서 제외

➤ Example

<pre>//3컬럼의 합계를 구한다. var Sum3 = mySheet.ComputeSum(" 3 "); //1행부터 10행까지 3컬럼 * 4컬럼 / 100 한 값들의 합계를 구한다. var Sum4 = mySheet.ComputeSum(" 3 * 4 / 100", 1, 10);</pre>

➤ 제공 버전



7.0.0.0



ConfirmOK Method

➤ 기능

OnMessage Event에서 IsConfirm = 1인 경우 확인창을 표시하고 응답 결과를 Sheet로 되돌려 준다.

ShowMsgMode(0)이면 Sheet에서 메시지가 발생할 때 시스템 팝업이 아닌 OnMessage Event가 발생한다. 발생한 메시지의 종류가 확인용 메시지일때, OnMessage Event의 IsConfirm 인자값이 1이 된다.

경고용 메시지의 경우 메시지 창이 닫히면 완료되지만, 확인용 메시지의 경우 사용자의 응답을 Sheet로 다시 넘겨주어야 한다. 이 속성은 OnMessage Event 안에서만 사용하며, 응답 결과를 Sheet로 돌려준다.

➤ Syntax

Syntax	ObjId. ConfirmOK (Val)
--------	-------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Val	Boolean	필수	확인창 선택 결과 값

➤ Example

```
//메시지 모드를 설정한다.
mySheet.ShowMsgMode=0;

//OnMessage 이벤트를 처리한다.
Function mySheet_OnMessage(grid,msg, level, isconfirm)
    //메시지 표시하기
    var win_result = window.showModalDialog(
        "sheet_message.jsp?Msg=" + msg + "&IsConfirm=" + isconfirm,
        'modalResult',
        'dialogWidth:200px;dialogHeight:200px;center:yes;help:no;status:no;');
```



```
//Sheet로 메시지 결과를 반환한다.  
If(IsConfirm) mySheet.ConfirmOK(win_result);  
</script>
```

- **제공 버전**
7.0.0.0



CountFormat Method

➤ 기능

보여지는 건수 정보의 포맷을 확인하거나 설정한다. 건수 정보는 사용 가능한 예약어의 조합으로 구성되는데 가능한 예약어의 아래와 같다.

예 약 어	설 명
"BOTTOMDATA"	바닥 데이터 행번
"TOTALROWS"	DB 전체 데이터 건수 1) 조회된 전체데이터 건수 2) 사용자가 세팅 위 2단계중 맨 마지막에 세팅된 값이 TOTALROWS로 보여지게 된다.
"SEARCHROWS"	조회된 데이터 건수 (상태값이 ""만 카운트)
"INSERTROWS"	입력 건수 (상태값이 "I"만 카운트)
"UPDATEROWS"	수정 건수 (상태값이 "U"만 카운트)
"DELETEROWS"	삭제 건수 (상태값이 "D"만 카운트)
"ROWCOUNT"	전체(입력+수정+삭제) 건수
"SELECTDATAROW"	현재 포커스 선택된 행의 레코드 순서 1) 숨겨진 행은 개수에 포함한다. 2) 합계와 소계는 레코드 개수에서 제외한다. 3) 2줄 이상이 한 레코드인 경우 행의 순서가 아니라 레코드 순서로 표시한다.
" FILTEREDCOUNT"	필터링 된 행의 개수

기본 포맷은 "[BOTTOMDATA / TOTALROWS]"으로 구성되어 있다.

➤ Syntax

Syntax	Get	ObjId. GetCountFormat()
--------	-----	--------------------------------



➤ **Info**

Return	String, 설정된 출력 포맷 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ **Example**

```
//건수 포맷을 확인한다.  
mySheet.GetCountFormat();
```

➤ **Syntax**

Syntax	Set	ObjId. SetCountFormat (Format)
--------	-----	---------------------------------------

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Format	String	필수	설정할 출력포맷

➤ **Example**

```
//건수 포맷을 설정한다.  
mySheet.SetCountFormat("현재 BOTTOMDATA건 / 전체 TOTALROWS건");  
  
//포커스로 표시한다.  
mySheet.SetCountFormat("선택 SELECTDATAROW행 / 총 ROWCOUNT건");
```

➤ **제공 버전**

7.0.0.0



CountInfoElement Method

➤ 기능

건수 정보를 시트 외부의 DOM 엘리먼트에 표현하기 위한 설정을 하거나 설정된 엘리먼트를 확인 한다.

설정이 가능한 DOM 엘리먼트는 input 과 div, span 등의 innerText 를 지원하는 DOM 엘리먼트이다.

➤ Syntax

Syntax	Get	ObjId. GetCountInfoElement ()
--------	-----	--------------------------------------

➤ Info

Return	object, 설정된 DOM 엘리먼트 객체		
Parameter	Type	필수여부	설 명

➤ Example

```
// 건수정보를 표시할 DOM 엘리먼트 객체를 확인 한다.
var elem = mySheet.GetCountInfoElement();
```

➤ Syntax

Syntax	Set	ObjId. SetCountInfoElement (Element)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Element	Object/ String	필수	설정할 DOM 엘리먼트 객체 또는 Id

➤ Example



```
// 건수정보를 표시할 DOM 엘리먼트를 설정한다.  
// ("countElem" 아이디의 DOM 엘리먼트가 생성되어 있어야 함)  
mySheet.SetCountInfoElement("countElem");  
}
```

➤ **제공 버전**

7.0.0.0

CountPosition Method

➤ 기능

건수 정보를 IBSheet의 특정 영역에 보이도록 설정한다.

건수 정보는 CountFormat 메소드에 설정된 건수정보가 표시된다. 기본적으로 현재 위치 정보와 최종 건수 정보를 표시하며 스크롤 바를 움직이거나 행이 추가되었을 때 변경되어 표시된다.

설정 값 별로 개수 정보 표시 영역 정보는 다음과 같다.

설정값	0	1	2	3	4
표시위치	표시하지 않음	좌측상단	우측상단	좌측하단	우측하단

➤ Syntax

Syntax	Get	ObjId. GetCountPosition()
--------	-----	----------------------------------

➤ Info

Return	Integer, 설정된 출력정보 위치 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// 개수 정보가 표시되지 않고 있다면 개수 정보를 좌측 상단에 표시한다.
If(mySheet.GetCountPosition() == 0) {
    mySheet.SetCountPosition(1);
}
```

➤ Syntax

Syntax	Set	ObjId. SetCountPosition (Position)
--------	-----	---

➤ Info

Return	None
--------	------



Parameter	Type	필수여부	설 명
Position	Integer	필수	설정할 출력정보 위치 값

➤ **Example**

```
// 개수 정보가 표시되지 않고 있다면 개수 정보를 좌측 상단에 표시한다.  
If(mySheet.GetCountPosition() == 0) {  
    mySheet.SetCountPosition(1);  
}
```

➤ **제공 버전**

7.0.0.0



CreatePivotTable Method

➤ 기능

피벗 테이블은 대화형 테이블의 일종으로, 데이터의 나열 형태에 따라서 합계나 카운트 등의 계산을 하는 기능이다.

데이터가 있는 IBSheet의 값이 변경되더라도 피벗 테이블에 자동 반영되지 않으므로 원하는 시점에 이 메소드를 다시 호출 하여야 한다.

➤ Syntax

Syntax	ObjId. CreatePivotTable (Info, DataSheet)
--------	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Info	json	필수	피벗 테이블 설정 객체 집합 (세부내용참고)
DataSheet	Object	필수	원본 데이터가 있는 IBSheet의 객체

세부내용

Info에 설정 할 수 있는 속성은 다음과 같다.

속성	타입	설명
Cols	String	열 레이블 필드에 설정할 ' '로 구분된 Column Index 또는 SaveName.
DefaultView	String	값이 없는 셀에 보여질 문자열
Rows	String	행 레이블 필드에 설정할 ' '로 구분된 Column Index 또는 SaveName.
SortRow	Boolean	행 레이블 집계시 정렬후 집계할지 여부. 0으로 설정시 소트를 하지 않고 조회된 데이터 기준으로 행 레이블이 생성된다. 1로 설정시 조회된 데이터를 정렬후에 행 레이블을 생성한다.(기본값 : 1)



Value	String	합계 또는 카운트할 ' '로 구분된 Column Index 또는 SaveName.										
ValueType	String	' '로 구분된 합계행 타입. ('Sum' 합계, 'Count' 개수) Value 의 개수와 동일해야 한다.										
AutoFitColWidth	String	FitColWidth에 대한 적용 시점 설정 적용하고자 하는 설정값을 구분자 " " 로 연결하여 설정한다. (Default : "") 세부 설명은 SetConfig Method 참고										
CalcSumRatio	Number	합계 비율 모드 (Default : 0)										
		<table><tr><th>설정 값</th><th>설명</th></tr><tr><td>0</td><td>기본 합계</td></tr><tr><td>1</td><td>총 합계 비율</td></tr><tr><td>2</td><td>열 합계 비율</td></tr><tr><td>3</td><td>행 합계 비율</td></tr></table>	설정 값	설명	0	기본 합계	1	총 합계 비율	2	열 합계 비율	3	행 합계 비율
		설정 값	설명									
		0	기본 합계									
		1	총 합계 비율									
		2	열 합계 비율									
3	행 합계 비율											

➤ Example

//5,6,7 열을 행 레이블 필드로 설정하고, 8,9,10 열을 열 레이블의 필드로 설정하여, 13 컬럼을 요약정보(합계)에 표시한다.

```
mySheet2.CreatePivotTable({Rows:'5|6|7',          Cols:'8|9|10',          Value:'13',
Value:'Sum'}, mySheet);
```

// 5,6,7 열을 행 레이블 필드로 설정하고, 2 열을 열 레이블의 필드로 설정하여, 13 컬럼을 요약정보(개수)에 표시한다.

```
mySheet2.CreatePivotTable({Rows:'5|6|7', Cols:'2', Value:'13', ValueType:'Count'},
mySheet);
```

➤ 제공 버전

7.0.0.0	
7.0.13.0	AutoFitColWidth, CalcSumRatio 인자 속성 추가



CumulateBackColor Method

➤ 기능

누계 행의 배경 색상을 확인하거나 설정한다.

색상 설정은 WebColor 값을 사용하여 설정한다..

➤ Syntax

Syntax	Get	ObjId. GetCumulateBackColor()
--------	-----	--------------------------------------

➤ Info

Return	String, 현재 누계 색상 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

//누계 행 배경색상을 확인 mySheet.GetCumulateBackColor();
--

➤ Syntax

Syntax	Set	ObjId. SetCumulateBackColor (Color)
--------	-----	--

➤ Info

Return	String, 현재 누계 색상 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Color	String	필수	WebColor 값

➤ Example

//누계 행 배경색상을 녹색으로 설정 mySheet.SetCumulateBackColor("#00FF00");
--

➤ 제공 버전



7.0.0.0

CurrentCollInfo Method

➤ 기능

현재 IBSheet 컬럼들의 위치, 숨김, 너비 정보를 확인하거나 설정 한다.
확인시 반환되는 값은 저장용도로만 사용이 가능 하다.

헤더 메뉴의 "컬럼 정보 저장"을 통해 컬럼들의 정보를 localStorage에 저장하는 기능을 제공하는데, 이 기능을 localStorage 이외의 다른 저장 매체를 이용 할때 이 기능을 사용 한다.

➤ Syntax

Syntax	Get	ObjId. GetCurrentCollInfo ()
--------	-----	-------------------------------------

➤ Info

Return	String, 현재 컬럼들의 정보		
Parameter	Type	필수여부	설 명
			없음

➤ Example

```
// 현재 컬럼들의 정보를 확인 한다.
var collInfo = mySheet.GetCurrentCollInfo()
```

➤ Syntax

Syntax	Set	ObjId. SetCurrentCollInfo (Info)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Info	String	필수	설정할 컬럼 정보 문자열

➤ Example



```
// 이전 컬럼 정보를 설정 한다.  
mySheet.SetCurrentColInfo(prevColinfo)
```

➤ **제공 버전**

7.0.13.30	
------------------	--

Data2Clipboard Method

➤ 기능

IBSheet위의 있는 모든 데이터를 클립보드에 복사한다. 이 함수를 호출하면 헤더를 포함하여 데이터 부분과 합계 부분까지 모든 데이터를 복사하며, 컬럼과 컬럼사이에는 탭으로 구분하고, 행과 행사이에는 줄바꿈으로 구분하여 클립보드에 복사한다.

해당 기능은 브라우저별 Clipboard 보안 상 IE 브라우저에서만 사용이 가능하다.

➤ Syntax

Syntax	ObjId.Data2Clipboard()
--------	------------------------

➤ Info

Return	String, 클립보드에 복사된 문자열		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
//클립보드에 복사한다.
mySheet.Data2Clipboard();
```

➤ 제공 버전

7.0.0.0



DataAlternateBackColor Method

➤ 기능

짝수번째 데이터 행의 기본 배경색을 설정하거나 확인한다.

데이터 행의 짝수번째와 홀수번째를 번갈아 가며 색상을 다르게 표시할 때 사용하는 속성이다. 데이터 행의 배경색은 이 속성과 DataBackColor 함수에 의해 설정된다.

색상 설정은 WebColor값을 이용하여 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetDataAlternateBackColor()
--------	-----	---

➤ Info

Return	String, 현재 짝수행 색상값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// 현재 색상 값을 확인한다.
var color = mySheet.GetDataAlternateBackColor(1,1);
alert("짝수번째행의 색상 값은 " + color + "입니다.");
```

➤ Syntax

Syntax	Set	ObjId. SetDataAlternateBackColor (Color)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Color	String	필수	설정할 색상 값

➤ Example



// 짝수번째행의 배경색을 빨간색으로 변경

mySheet.SetDataAlternateBackColor("#FF0000");

// WebColor

➤ **제공 버전**

7.0.0.0



DataAutoTrim Method

➤ 기능

조회 또는 저장 시 데이터의 공백 트림 여부를 확인하거나 설정한다.

기본적으로 조회하거나 저장할 때 데이터 앞뒤에 있는 공백은 자동으로 트림되어 사용되지만 이 속성을 0로 설정한다면 조회시 데이터에 있는 공백을 그대로 사용하고, 저장 시도 공백을 그대로 사용한다. 이 속성 설정은 조회나 저장 뿐만 아니라 CellValue 와 같은 속성에 모두 적용된다. 기본으로 데이터의 공백 트림 속성은 설정되어있다.

➤ Syntax

Syntax	Get	ObjId. GetDataAutoTrim ()
--------	-----	----------------------------------

➤ Info

Return	Boolean, 데이터의 공백Trim 설정값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

//자동 트림여부 설정 확인하기 mySheet.GetDataAutoTrim();

➤ Syntax

Syntax	Set	ObjId. SetDataAutoTrim (Trim)
--------	-----	--------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Trim	Boolean	필수	설정할 데이터의 공백Trim값

➤ Example



```
//자동 트림하여 조회하기  
mySheet.SetDataAutoTrim(1);  
mySheet.DoSearch("list.jsp");  
//트림하지 않고 데이터 그대로 저장하기  
mySheet.SetDataAutoTrim(0);  
mySheet.DoSave("save.jsp");
```

➤ **제공 버전**

7.0.0.0



DataBackColor Method

➤ 기능

홀수번째 데이터 행의 기본 배경색을 설정하거나 확인한다.

데이터 행의 짝수번째와 홀수번째를 번갈아 가며 색상을 다르게 표시할 때 사용하는 속성이다. 데이터 행의 배경색은 이 속성과 DataBackColor 함수에 의해 설정된다.

색상 설정은 WebColor값을 이용하여 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetDataBackColor()
--------	-----	----------------------------------

➤ Info

Return	String, 현재 홀수행 색상값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

//홀수번째 데이터 행의 배경색을 확인한다. mySheet.GetDataBackColor();

➤ Syntax

Syntax	Set	ObjId. SetDataBackColor (Color)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Color	String	필수	설정할 색상 값

➤ Example



```
//홀수번째 데이터 행의 배경색을 흰색으로 설정한다.  
mySheet.SetDataBackColor("#FFFFFF");
```

- **제공 버전**
7.0.0.0



DataCopy Method

➤ 기능

마지막으로 선택된 데이터 행의 내용을 복사하여 신규 입력 행으로 생성하고, 생성된 행의 Row Index를 반환한다. 트리 형태로 구성된 경우 복사할 대상 행에 자식레벨이 존재할 경우 IncludeChild 인자가 1이면 자식 레벨의 행까지 모두 복사한다.

신규 생성된 행의 트랜잭션 상태는 "입력"으로 표시된다.

➤ Syntax

Syntax	ObjId. DataCopy ([IncludeChild])
--------	---

➤ Info

Return	Long, 복사하여 생성된 행의 Row Index		
Parameter	Type	필수여부	설 명
IncludeChild	Boolean	선택	자식 레벨의 행까지 복사하는지 여부 Default=0

➤ Example

```
//행을 복사하고, 복사된 행의 트랜잭션 상태를 "조회"로 변경한다.  
//'Status' 는 상태 컬럼의 SaveName  
var Row = mySheet.DataCopy();  
mySheet.SetCellValue(Row, "Status", "R");  
//자식 레벨까지 모두 복사하기  
mySheet.DataCopy(1);
```

➤ 제공 버전

7.0.0.0

DataFontColor Method

➤ 기능

모든 데이터 행의 글자색을 설정하거나 확인한다.
색상 설정은 WebColor값을 이용하여 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetDataFontColor()
--------	-----	----------------------------------

➤ Info

Return	String, 설정된 글자 색상값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//데이터 행의 글자색을 확인한다.
mySheet.GetDataFontColor();
```

➤ Syntax

Syntax	Set	ObjId. SetDataFontColor (Color)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Color	String	필수	설정할 색상 값

➤ Example

```
//데이터 행의 글자색을 빨간색으로 설정한다.
mySheet.SetDataFontColor("#FF0000");
```

➤ 제공 버전



7.0.0.0

DataInsert Method

➤ 기능

데이터 행을 신규 생성하고, 생성된 행의 Row Index를 반환한다.

Row인자와 Level인자는 다음과 같이 설정된다.

Row 설정값	생성 행 위치	Level
Row < 0	마지막 행에 생성	0레벨
Row >= 전체 행	마지막 행에 생성	0레벨
Row < 데이터 첫 행	첫 행에 생성	0레벨
그 외	해당 행에 생성	설정 레벨
기 본	선택된 행 아래 생성	설정 값이 없는 경우 선택 행의 자식 레벨

트리 구조시 Level인자는 값을 설정하지 않으면 기본적으로 선택 행의 자식 레벨로 생성한다. 만약 선택 행이 삭제 상태라면 에러 메시지를 표시하고 신규 생성 처리를 중단한다.

➤ Syntax

Syntax	ObjId. DataInsert ([Row], [Level])
--------	---

➤ Info

Return	Long, 신규 생성된 행의 Row Index		
Parameter	Type	필수여부	설 명
Row	Long	선택	생성될 행 위치, Default="마지막 선택 행 바로 아래"
Level	Long	선택	생성될 행의 트리 레벨 수준, Default="마지막 선택 행의 레벨보다 1큰 레벨로, 자식으로 생성"

➤ Example



```
//첫 행에 생성하기  
mySheet.DataInsert(0);  
  
//마지막 행에 생성하기  
mySheet.DataInsert(-1);  
  
//현재 선택된 행의 바로 아래에 생성  
mySheet.DataInsert();  
  
//7 행에 생성  
mySheet.DataInsert(7);
```

➤ **제공 버전**
7.0.0.0

DataLinkMouse Method

➤ 기능

데이터 행의 페이지 링크 여부를 확인하거나 설정한다.

데이터 행에 Click 또는 DbClick 시 링크 페이지가 연결되어 있다면 Mouse Cursor가 손가락 모양으로 변경되어 링크가 연결되어 있음을 알리기 위해 사용한다.

Link 인자에 1값을 설정할 경우 해당 컬럼에서 Mouse Cursor가 손가락 모양으로 변경된다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 모든행에 대해서 처리를 하고 SaveName으로 설정할 경우 해당 SaveName이 포함된 행에 대해서만 처리 한다.

➤ Syntax

Syntax	Get	ObjId. GetDataLinkMouse (Col)
--------	-----	--------------------------------------

➤ Info

Return	Boolean, 설정된 링크 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Col	Number / String	필수	특정 컬럼의 Index 또는 SaveName

➤ Example

//데이터행의 페이지 링크 여부 확인한다. mySheet.GetDataLinkMouse(1);

➤ Syntax

Syntax	Set	ObjId. SetDataLinkMouse (Col, Link)
--------	-----	--

➤ Info



Return	None		
Parameter	Type	필수여부	설 명
Col	Number / String	필수	특정 컬럼의 Index 또는 SaveName
Link	Boolean	필수	링크여부 설정

➤ **Example**

```
//1,2 컬럼에서만 가능하도록 설정한다.  
mySheet.SetDataLinkMouse(1, 1);  
mySheet.SetDataLinkMouse(2, 1);
```

➤ **제공 버전**

7.0.0.0

DataMove Method

➤ 기능

데이터 행을 원하는 다른 위치로 이동한다. 트리 형태로 설정 될 경우 ChildLevel이 존재하는 경우 선택된 행의 ChildLevel 행도 모두 이동처리한다.

이 함수는 DataInsert 함수나 DataCopy 함수처럼 신규 행이 생성되는 것이 아니라 기존의 행이 다른 위치로 이동할 뿐이다. 하지만 내부적으로는 복사와 삭제의 작업을 하게된다.

따라서 ToRow가 FromRow보다 클때는 복사되는 위치에 주의를 기울여야 한다. 예를 들어 ToRow가 FromRow보다 클때는 다음과 같이 동작하게 된다.

DataMove(4,1):



➤ Syntax

Syntax	ObjId. DataMove (ToRow, [FromRow], [RowLevel])
--------	---

➤ Info

Return	Long, 이동된 행의 Top Row Index		
Parameter	Type	필수여부	설 명
ToRow	Long	필수	이동할 위치의 Row Index
FromRow	Long	선택	선택된 데이터의 Row Index, Default=-1
RowLevel	Integer	선택	이동 후 선택된 데이터의 트리 레벨 Default="원래 레벨 수준"

➤ Example



```
//12행을 10행으로 이동한다.  
mySheet.DataMove(10, 12);
```

- **제공 버전**
7.0.0.0

DataRowHeight Method

➤ 기능

모든 데이터 행의 행높이를 설정하거나 확인한다.

이 속성은 픽셀 단위로 설정하는 값이며, 기본 설정값은 21픽셀이다.

➤ Syntax

Syntax	Get	ObjId. GetDataRowHeight()
--------	-----	----------------------------------

➤ Info

Return	Integer, 설정된 높이 값(Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

//모든 데이터 행의 높이를 확인한다. mySheet.GetDataRowHeight();
--

➤ Syntax

Syntax	Set	ObjId. SetDataRowHeight (Height)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Height	Integer	필수	설정할 데이터행 높이 값

➤ Example

//모든 데이터 행의 높이를 22 픽셀로 설정한다. mySheet.SetDataRowHeight(22);

➤ 제공 버전



7.0.0.0

DataRowMerge Method

➤ 기능

모든 데이터 행의 가로 머지 여부를 설정하거나 확인한다.
기본값은 0으로 설정 하지 않은 경우는 가로 머지가 불가능 하다.

➤ Syntax

Syntax	Get	ObjId. GetDataRowMerge ()
--------	-----	----------------------------------

➤ Info

Return	Boolean, 설정된 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//모든 데이터 행의 가로머지를 여부
mySheet.GetDataRowMerge();
```

➤ Syntax

Syntax	Set	ObjId. SetDataRowMerge (Merge)
--------	-----	---------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Merge	Boolean	필수	머지 허용 여부

➤ Example

```
//모든 데이터 행의 가로머지를 허용
mySheet.SetDataRowMerge(1);
```

➤ 제공 버전



7.0.0.0

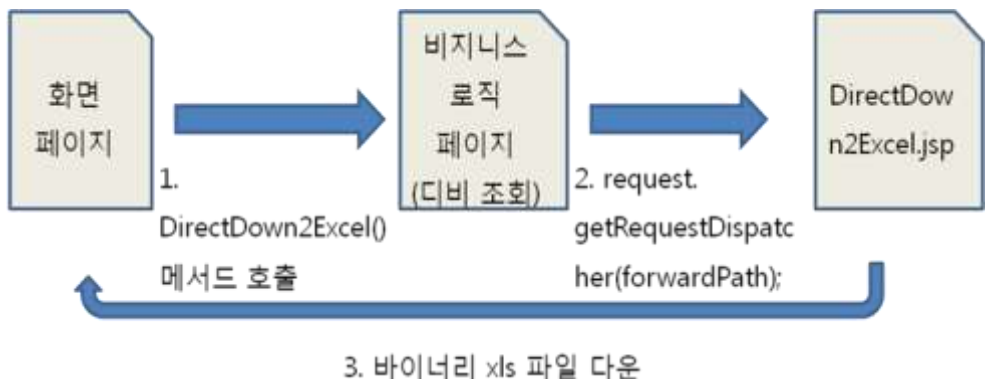
DirectDown2Excel Method

➤ 기능

조회된 데이터가 많은 경우 Down2Excel()메서드를 이용하여 엑셀 파일을 다운로드 하면 client에서 조회된 데이터를 서버로 보내는 과정에서 상당한 부하가 걸릴수 있다.

따라서 client의 아이비시트로부터는 헤더타이틀이나 각 컬럼의 서식(포맷)정보만 서버로 전달하고, 데이터부분은 별도의 페이지(혹은 서블릿)을 통해 데이터를 java.util.List (java.util.Map) 구조로 생성하여 "SHEETDATA"라는 이름으로 request객체에 담아 DirectDown2Excel.jsp파일에 전달하면 client로 부터온 헤더정보와 디비의 데이터를 조합하여 엑셀 파일을 생성 다운로드 하는 메서드이다.

처리되는 과정은 다음과 같다.



// 화면 페이지

```

var cols = [
    {Type:"Text",Width:85,SaveName:"POSTNO",Format:"PostNo",Align:"center"},
    {Type:"Text",Width:70,SaveName:"SIDO"},
    {Type:"Text",Width:80,SaveName:"SIGUNGU"},
    {Type:"Text",Width:80,SaveName:"LEE"},
    {Type:"Text",Width:300,SaveName:"ADDRESS"}
];
mySheet.InitColumns(cols);
  
```



```
var param = {
    URL:"/bus/bussinessList.jsp" //비즈니스 로직 페이지
    ,ExtendParam:"sa_nm=양진열&sa_no=980123"
    ,FileName:"PersonList.xls"
};
//시트에 조회된 데이터는 전달되지 않고, 헤더타이틀이나 컬럼의 포맷 만 서
버로 전달됨.
Sheet.DirectDown2Excel(param);
```

// bussinessList.jsp 페이지

```
// 1. 화면으로 부터 조회 조건을 받음.
String sa_name = request.getParameter("sa_nm");
String sa_no = request.getParameter("sa_no");
// 2. 엑셀로 내려질 데이터를 DB에서 조회
String query
= "SELECT POSTNO, SIDO, SIGUNGU, LEE, ADDRESS FROM POSTNO";
Class.forName(driver);
conn = DriverManager.getConnection(url,id,pwd);
pstmt = conn.prepareStatement(query);
rs = pstmt.executeQuery();
// 3. 데이터를 List(Map)형태로 전환.
Java.util.List li = new java.util.List();
java.util.Map mp = null;
while(rs.next()){
    mp = new Java.util.Map();
    //map 데이터 생성시 이름은 반드시 시트의 SaveName과 동일해야 한다.
    Mp.put("POSTNO",rs.getString("POSTNO"));
    mp.put("SIDO",rs.getString("SIDO"));
    ..
    ..
    li.add(mp);
```

```
}  
// 데이터를 반드시 SHEETDATA라는 이름으로 담는다.  
Request.setAttribute("SHEETDATA", li);  
  
// 4. DirectDown2Excel.jsp 페이지로 forwarding  
System.out.println("전체 건수:"+li.size());  
String forwardPath = "./DirectDown2Excel.jsp";  
if(!"".equals(forwardPath)){  
RequestDispatcher rd = request.getRequestDispatcher(forwardPath);  
rd.forward(request,response);  
}  
}
```

전달되는 인자의 형식은 JSON 타입으로 설정 하고자 하는 정보를 JSON 형태로 구성하여 전달한다.

예)

```
var params = { FileName : "myFile.xls", SheetName : "Sheet" } ;
```

AllTypeToText 인자는 모든 컬럼 타입을 Text타입으로 다운로드 받고자 하는 경우에 설정한다. NumberTypeToText 인자를 설정하지 않거나 0으로 설정하는 경우 숫자타입 컬럼을 제외한 모든 컬럼이 Text 타입으로 다운로드 되며 NumberTypeToText를 1로 설정한 경우 모든 컬럼이 Text타입으로 다운로드 처리된다.

AllTypeToText 인자를 1로 설정하는 경우 컬럼에 설정한 모든 포맷은 무시된다. 또한 TextToGeneral 을 설정한것과 동일하게 다운되므로 TextToGeneral 인자도 무시된다.

AppendPrevSheet 속성은 [Down2ExcelBuffer](#) 메소드를 이용하여 2개 이상의 시트를 엑셀 파일로 다운로드 할 때 이전의 시트 내용을 작성한 워크시트에 현재의 시트 내용을 Append 하여 작성할지 여부를 설정한다. 설정값이 0(false)인 경우 워크시트를 새로 생성하여 작성되고, 1(true)인 경우 마지막 작성한 워크시트에 Append 하여 작성 된다. Buffer 모드의 첫번째의 시트인 경우와 Buffer 모드설정이 아닌 경우는 해당 속성의 설정값은 무시된다.

AutoSizeColumn 인자는 컬럼 넓이를 내용에 맞게 자동으로 조절한다. 단, 자동조절 결과가 정확하게 맞지 않을수 있다.

ComboValidation 인자는 콤보 타입의 컬럼에 대하여 데이터영역(헤더, 소계, 누계, 합계행 제외)에 드롭다운 리스트 형식(데이터 유효성 검사)으로 생성할지를 설정한다. 기본값은 0으로 데이터 유효성 검사를 하지 않는다.

1로 설정시 리스트의 항목은 DownCombo 가 "Code"이면 컬럼에 설정된 ComboCode가 사용되며, "Text"일 경우 ComboText 값을 사용한다.

셀별로 InitCellProperty 로 변경한 Combo 목록은 엑셀의 드롭다운 리스트 목록에 별개로 적용되지 않으나 엑셀다운시 정상적인 값이 보여진다.

DownCols 인자는 다운로드 받을 컬럼들을 "|" 문자로 연이은 문자열이다.

SaveName과 컬럼인덱스가 모두 가능하다. 널인 경우 모든 컬럼의 자료를 다운로드 받는다.

DownHeader 인자는 헤더를 포함하여 다운로드 받을지 설정한다. 기본값은 1(헤더 포함) 이다.

ExcelFontSize 인자는 SheetFontSize 와는 별개로 엑셀문서에서만 특정 폰트 크기로 지정할 수 있다.

ExcelRowHeight 인자는 엑셀문서에서만 모든 행을 특정 픽셀 크기로 지정하기 위하여 사용한다. "auto"로 설정할 경우 자동으로 행 높이를 설정한다.

ExtendParam 인자는 서버측으로 전달할 조회조건등의 내용을 Get 방식의 QueryString을 만들어 넣어주면 URL인자로 지정한 페이지에서 request.getParameter() 메서드를 통해 받아 볼 수 있다.

Ex) ExtendParam:"name=shkim&sa_no=980123&enter_date=19980222";

ExtendParamMethod 인자는 ExtendParam 의 내용을 GET 또는 POST 로 전달할지를 설정한다.

FileName 인자는 엑셀 다운로드시의 파일명을 설정한다. 파일 확장자를 xls 로 지정시 엑셀 2003 포맷으로 다운로드하고,xlsx 로 지정시 엑셀 2007 포맷으로 다운로드하며, 확장자를 지정하지 않을 경우에는 xls 포맷으로 다운로드 받는다.

Ex) FileName:"NameCard.xls"

HiddenColumn 인자는 숨은 컬럼들을 엑셀로 다운로드 받은 경우, 해당 컬럼이 눈에 보이지는 않지만 엑셀 메뉴중 "숨기기 취소"를 선택한 경우 해당 컬럼이 다시 보일 수 있도록 엑셀 문서에 다운로드 받는다.

KeyFieldMark 인자는 KeyField 컬럼의 KeyField 마크(*)를 다운 받을 때 사용한다.

Merge 인자는 헤더 데이터에 인접한 셀간에 같은 글자가 있는 경우, 병합할지 여부를 결정한다. 기본값 0 이다.

Multipart 인자는 시트의 헤더 정보를 서버로 전달할 경우 Multipart 형식인지 아니면 일반 형식으로 전달할 것인지를 결정한다. (Spring,struts2 등의 프레임워크 사용시 반드시 사용할 것)

IBSheet는 excel export 기능과 관련하여 서버로 데이터를 전달시 기본적으로 Multipart 형식으로 전달한다. 이는 시트위에 데이터가 많아서 일반적인 post 형식으로 전달할 경우 서버에서 받지 못하는 것을 방지하기 위함이다. 하지만 DirectDown2Excel메서드와 spring 프레임워크를 사용하는 경우 먼저 데이터를 스프링 프레임워크로 부터 얻게 되는데, 이때 프레임워크에서 Multipart 데이터를 모두 가져가 실제 forwarding을 받는 DirectDown2Excel.jsp 에서는 시트의 헤더정보를 얻지 못하게 된다. 따라서 spring 프레임워크를 사용시에는 이 인자를 0(false)로 설정하고, DirectDown2Excel.jsp 파일에서 헤더 정보를 얻는 아래 부분을 다음과 같이 수정하여 야 한다.

[DirectDown2Excel.jsp 내용]

```
// 파라미터 정보를 얻음  
String data = ibExcel.getData();
```



위 내용을 아래와 같이 수정

```
String data = request.getParameter("data");
```

NumberFormatMode 속성은 실수 형태의 데이터타입에 대한 셀서식 설정 방식을 설정할 때 사용 한다. 설정값에 따른 처리 방식은 아래와 같다.

설정값	내용
0	컬럼 단위 Format 기준 셀서식 설정
1	셀의 값 기준에 따라 정수 또는 실수 포맷형태의 셀서식 설정
2	일반 서식으로 설정 (별도의 셀 서식포맷을 설정하지 않음)

NumberTypeToText 인자는 숫자타입 셀 데이터를 문자 타입으로 다운로드 받을 때 사용한다. 기본값은 0 이다.

ReportXMLURL 특정 xml로 부터 엑셀파일에 설정할 머릿글,바닥글 등을 설정 한다. Xml 파일에 대한 자세한 내용은 6.4장의 "**엑셀리포트 XML**"부분을 참고

SheetDesign 인자는 헤더의 색상을 반영할지 여부를 결정한다. 기본값은 0이다.

SheetName 인자는 엑셀에 WorkSheet 이름을 설정하는것으로 데이터 양이 많아서 1개 WorkSheet에 내릴수 있는 최대 행의 개수인 65536개를 초과할 경우 원래 이름과 중복 사용할수 없으므로 이름 뒤에 괄호와 인덱스가 붙는다. 예를 들어 "1월"이라고 설정했다면 초과된 데이터는 새로운 WorkSheet에 "1월(1)", "1월(2)"라고 이름을 표시한다
ex) SheetName:"SupportTeam"

TextToGeneral 인자는 IBSheet 의 컬럼이 Text Type 일 경우 엑셀로 다운되었을 때 일반 또는 텍스트 형식으로 변경할 때 사용한다.

TitleAlign 인자는 TitleText로 설정한 제목이나 기타 내용에 대한 정렬을 설정

한다. "left", "center", "right" 중 원하는 값으로 설정 가능하며 별도 설정하지 않는 경우 기본값인 "center"로 처리된다.

TitleText 인자는 그리드의 위쪽에 제목이나 기타 자료들을 사용자의 임의대로 넣을 수 있는 기능이다. A|B|CWrWnD|E|F 와 같이 입력한 경우 첫 행에 3개의 셀에 각각 A,B,C 값이 들어가고 두번째 행의 3개의 셀에 각각 D,E,F 값이 들어간다. 값 안에서 엔터를 포함하려면 wr 이나 wn 을 삽입하도록 한다. WrWn 이 10개가 포함되면 11줄을 차지하게 되고 12번째 행부터 IBSheet 자료가 이어진다.

이 인자에 넣은 셀자료들은 엑셀에 표시된 컬럼 수보다 더 많은 컬럼의 자료를 넣을 경우 그 값들은 무시된다. 즉 엑셀다운에 표시되는 컬럼이 모두 10개의 컬럼이면 타이틀이나 기타 자료들을 최대 10개까지만 가능하므로 11개 이상의 자료를 넣을 수는 없다. (타이틀 행의 제한은 별도로 없음)

URL 인자는 엑셀보여질 데이터를 생성할 페이지에 대한 경로를 넣는다.(필수)

ex) [URL:"/bus/displayList.do"](#)

UserMerge 인자는 TitleText 로 입력한 셀별 자료들이나 IBSheet에 해당되는 셀 자료들을 임의로 머지 적용하기 위하여 설정한다. 1개의 머지문은 4개의 숫자와 콤마로 조합되며 여러 개의 머지는 공백(스페이스) 문자로 조합된다. "0,0,2,2 0,2,1,8" 과 같이 설정한 경우 엑셀의 첫행, 첫열인 (0,0)에서 2x2 크기의 머지가 하나 적용되고 첫행, 2열(=세번째컬럼)의 셀인(0,2) 에서 1x8 크기의 머지가 하나 더 적용된다.

➤ Syntax

Syntax	ObjId. DirectDown2Excel ([parameters])
--------	---

➤ Example

```
// 서버에서 데이터를 직접 준비하여다운로드 한다.
var param = {URL:"/sub/ex/bussDeptList.jsp"
,ExtendParam:"DECNO=3422&PartMngNO=982211"
,FileName:"OrgList.xls"};
```



```
mySheet. DirectDown2Excel(param);
```

➤ **제공 버전**

7.0.0.0	
7.0.2.0	AllTypeToText 속성 추가
7.0.13.26	NumberFormatMode 속성 추가

DirectLoadExcel Method

➤ 기능

LoadExcel 처럼 엑셀 문서를 읽어들이는 것은 동일하지만, 엑셀 문서의 내용을 IBSheet 에 담지 않고 서버쪽에서 지정한 페이지로 전달한다.

따라서 엑셀로부터 넘어온 데이터를 받아 줄 페이지를 별도로 개발하여야 하고, DirectLoadExcel의 ExtendParam 을 통해 엑셀의 내용을 받아 줄 포워딩 페이지(**FP=/jsp/excelsave.jsp**) 경로를 넣어주면 된다.

포워딩 페이지에서 엑셀의 내용을 받는 방법은 request 객체 안에 "SHEETDATA"라는 이름으로 값을 뽑아 낼수 있다. SHEETDATA안의 내용은 List(Map)으로 구성되어있고, Map의 키는 IBSheet 각 컬럼의 SaveName이 된다.

Ex) //포워딩 페이지..

```
List sheet = (List)request.getAttribute("SHEETDATA");
for(int i=0;i<sheet.size();i++){
    Map mp =(Map)sheet.get(i);
}
```

또한 화면으로부터 ExtendParam으로 전달했던 내용들은 포워딩 페이지에서 모두 getAttribute로 얻을 수 있다.

p.s : ibsheet.cfg 안에서 DirectLoadExcel 경로를 설정해야 바르게 동작한다.

➤ Syntax

Syntax	ObjId. DirectLoadExcel ([parameters])
--------	--

➤ Info

Parameter	Type	필수여부	설 명
ColumnMapping	String	선택	엑셀 컬럼 번호 Default=""
EndRow	String	선택	엑셀 로딩완료 행번호 Default="0"



ExtendParam	String	필수	서버로 전달될 파라미터를 쿼리스트링 형태로 넣음. 특히 저장작업을 수행할 FP는 필수
FileExt	String	선택	업로드 가능한 파일 확장자 Default=""
Mode	String	선택	로딩 방식 (헤더매칭등) Default="HeaderMatch"
StartRow	String	선택	엑셀 로딩 행번호 Default="1"
WorkSheetNo	String	선택	엑셀WorkSheet번호, Default="1"
WorkSheetName	String	선택	엑셀WorkSheet이름 Default=""

➤ Example

```
// 참고 함수
function makeExParam(key,data){
    return "&"+encodeURIComponent(key)+"="+encodeURIComponent(data);
}

// 엑셀로 넘어온 내용을 전달할 포워딩 페이지(ContextRoot명은 빼고 적는다.)
var param =makeExParam( "FP" ,"/bu/MassSave.jsp");
param += makeExParam("sname","chris");
param += makeExParam("date","20091221");
var parameters = { Mode : mch, StartRow: "1", ExtendParam:param}
// 엑셀로 로딩하여 서버에서 즉시 처리하기
mySheet. DirectLoadExcel(parameters);

// MassSave.jsp 페이지 내용
// 1. 엑셀의 내용을 찍어본다. (데이터확인)
```



```
String PRINT_STR = "";
ArrayList keys = new ArrayList();
List li = (List)request.getAttribute("SHEETDATA");
for(int i=0;i<li.size();i++){
    Map mp = (Map)li.get(i);
    // 헤더 한줄만 뿌려주자.
    If(i==0){
        Iterator it = mp.keySet().iterator();
        while(it.hasNext()){
            String key = (String)it.next();
            PRINT_STR += key+"Wt";
            keys.add(key);
        }
        PRINT_STR += "WWn";
    }
    // 데이터를 뿌리자
    for(int c=0;c<keys.size();c++){
        PRINT_STR += mp.get(keys.get(c))+"Wt";
    }
    PRINT_STR += "WWn";
}
// 서버 콘솔에서 확인
System.out.println(PRINT_STR);

// 2. ExtendParam으로 넘긴 내용도 확인해 본다.
System.out.println( request.getAttribute("sname"));
System.out.println( request.getAttribute("date"));

// 3. 최종적으로 화면에 결과를 리턴한다.
Out.println("<script>alert('총 데이터 건수 :"+li.size()+" 건이Wn저장 되었습니다.');
```



- 제공 버전
7.0.0.0



DisposeSheet Method

➤ 기능

IBSheet 객체를 삭제 한다. 삭제시 사용하던 메모리도 강제로 해제 처리 한다.

➤ Syntax

Syntax	ObjId.DisposeSheet()
--------	----------------------

➤ Info

Parameter	Type	필수여부	설 명

➤ Example

```
// 생성된 mySheet 를 제거한다  
mySheet.DisposeSheet();
```

➤ 제공 버전

7.0.0.0



DoAllSave Method

➤ 기능

데이터의 트랜잭션 상태에 관계 없이 모든 데이터를 저장 하도록 페이지를 호출한다.

데이터 건수가 한건도 없으면 경고 메시지를 표시하고, 처리는 중단한다.
저장 준비를 위한 저장 데이터를 모으는 과정에서 OnValidataion 이벤트가 발생하며 사용자 정의 로직에 따라 OnValidataion에서 실패한 경우 저장 처리는 중단한다.

Url로 저장 페이지를 호출하고 저장 처리를 완료하여 저장 XML을 읽어 들이고 난 후 OnSaveEnd 이벤트를 발생하고 처리는 종료된다.

Mode인자는 Query String 문자열 조합방법을 설정한다.
설정값에 따른 Query String 결과는 다음과 같다.

Mode	설명	예
1	셀기준 조합 (SaveName 기준 Array 형식)	sSeq=1&sStatus=R..
2	컬럼기준 조합 (SaveName 기준 구분자 조합 형식)	sSeq=1 2&sStatus=R U..

"선택" 인자는 json 형식으로 속성 설정이 가능하다.(Example 참고).

➤ Syntax

Syntax	ObjId.DoAllSave(Url, [Opt])
--------	-----------------------------

➤ Info

Return	Boolean, 저장 처리여부		
Parameter	Type	필수여부	설 명

Url	String	필수	저장 처리할 페이지 파일 이름
Opt.Param	String	선택	저장을 위한 Parameter (Default="")
Opt.UrlEncode	Boolean	선택	IBSheet 위의 데이터를 인코딩할 지 여부를 설정 (Default=1)
Opt.Mode	Number	선택	Query String 문자열 조합방법을 설정 Mode=1, Mode=2 (Default =1)
Opt.Delim	String	선택	Mode=2일때, 연결될 구분자 설 정 (Default =" ")
Opt.RegHeader	Object	선택	요청헤더 설정 값 객체, (Default={})
Opt.ValidKeyField	Boolean	선택	KeyField 체크 여부 (Default=1)
Opt.ValidFullInput	Boolean	선택	FullInput 체크 여부 (Default=1)

➤ **Example**

```
// 모든 내역 저장하기
var Result = mySheet.DoAllSave("save.jsp", { Param : "id=khlee&seq=1" });

//저장 실패한 경우 에러 메시지를 표시하고, 성공한 경우 조회처리
if(!Result){
    alert("저장 실패했습니다. 다시 시도하세요");
} else {
    mySheet.DoSearch("list.jsp");
}

//설정할 "선택" 인자 json형식으로 속성 설정하기
mySheet.DoAllSave(PageUrl, { UrlEncode:0, Mode:2, Delim:"$"});
```

➤ **제공 버전**

7.0.0.0	
7.0.1.0	저장 처리 여부에 따른 리턴값 추가



7.0.12.4

ValidKeyField, ValidFullInput 인자 속성 추가



DoPrint Method

➤ 기능

보여지고 있는 모든 데이터를 인쇄한다.

인쇄 처리는 브라우저의 인쇄 기능을 따르며 배경색 및 이미지를 인쇄 하기 위해서는 브라우저의 인쇄설정을 변경 하여야 한다.

➤ Syntax

Syntax	ObjId.DoPrint()
--------	-----------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
없음			

➤ Example

// 인쇄 하기 mySheet.DoPrint();

➤ 제공 버전

7.0.0.0



DoRowSearch Method

➤ 기능

특정 행의 셀 데이터를 조회 한다.

➤ Syntax

Syntax	ObjId. DoRowSearch (Row, PageUrl, [Param], [Opt])
--------	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	대상 행의 Index
PageUrl	String	필수	처리할 페이지 Url
Param	String	선택	조회조건 Query String, Default ""
Opt.ReqHeader	Object	선택	요청헤더 설정 값 객체, Default={}
Opt.Sync	Boolean	선택	동기 조회 여부 Default=0 (SearchSync 설정 시 SearchSync 값)
Opt.Wait	Boolean	선택	대기 이미지 표시 여부, Default = 1

➤ Example

```
//3컬럼의 데이터가 바뀌었을 때 해당 행의 데이터를 DB에서 읽어옴
function mySheet_OnChange(Row, Col, Value) {
    if (Col == 3) {
        var opt = { Wait : 1, Sync : 1 };
        mySheet.DoRowSearch(Row, "grid_rowdata.html", "", opt);
    }
}
// 1. 이미지 미표시, 비동기 조회
var opt = { Wait : 0, Sync : 0 };
```



```
mySheet.DoRowSearch(Row, "grid_rowdata.html", "", opt);
```

```
// 2. 이미지 표시, 동기 조회
```

```
var opt = { Wait : 1, Sync : 1 };
```

```
mySheet.DoRowSearch(Row, "grid_rowdata.html", "", opt);
```

➤ **제공 버전**

7.0.0.0



DoSave Method

➤ 기능

데이터의 트랜잭션 상태 또는 특정 컬럼 데이터에 따라 저장 처리한다.

Col 인자에 아무것도 설정하지 않는 경우 트랜잭션 상태가 조회가 아닌 데이터 행만 저장 처리하고, Col에 특정 인자가 있는 경우 해당 컬럼에 값이 있는 데이터만 저장 처리한다.

해당 컬럼이 CheckBox 형태 인 경우 CheckBox에 체크된 것만 저장 처리한다.

데이터 건수가 한건도 없으면 경고 메시지를 표시하고, 처리는 중단한다.

저장 준비를 위한 저장 데이터를 모으는 과정에서 OnValidataion 이벤트가 발생하며 사용자 정의 로직에 따라 OnValidataion에서 실패한 경우 저장 처리는 중단한다.

Url로 저장 페이지를 호출하고 저장 처리를 완료하여 저장 XML을 읽어 들이고 난 후 OnSaveEnd 이벤트를 발생하고 처리는 종료된다.

Mode인자는 Query String 문자열 조합방법을 설정한다.

설정값에 따른 Query String 결과는 다음과 같다.

Mode	설명	예
1	셀기준 조합 (SaveName 기준 Array 형식)	sSeq=1&sStatus=R..
2	컬럼기준 조합 (SaveName 기준 구분자 조합 형식)	sSeq=1 2&sStatus=R U..

"선택" 인자는 json 형식으로 속성 설정이 가능하다.(Example 참고)

➤ Syntax

Syntax	ObjId.DoSave(Url, [Opt])
--------	--------------------------

➤ Info

Return	Boolean, 저장 처리여부		
Parameter	Type	필수여부	설 명
Url	String	필수	저장 처리할 페이지 파일 이름
Opt.Param	String	선택	저장을 위한 Parameter (Default="")
Opt.Col	Number / String	선택	저장 대상이 되는 기준 컬럼의 Index 또는 SaveName Default=상태컬럼(-1)
Opt.Quest	Boolean	선택	저장 시 확인 메시지 표시 여부 설정, Default=1
Opt.UrlEncode	Boolean	선택	IBSheet 위의 데이터를 인코딩할 지 여부를 설정, Default=1
Opt.Mode	Integer	선택	Query String 문자열 조합방법을 설정Mode=1, Mode=2 (Default=1)
Opt.Delim	String	선택	Mode=2일때, 연결될 구분자 설 정 (Default=" ")
Opt.RegHeader	Object	선택	요청헤더 설정 값 객체, (Default={})
Opt.ValidKeyField	Boolean	선택	KeyField 체크 여부 (Default=1)
Opt.ValidFullInput	Boolean	선택	FullInput 체크 여부 (Default=1)

➤ **Example**

```
//트랜잭션이 일어난 데이터만 저장하기
mySheet.DoSave("Save.jsp" ,{ Param : "id=khlee&seq=1" });

//2컬럼의 CheckBox가 체크된것만 저장하기
mySheet.DoSave("Save.jsp" ,{ Param : "id=khlee&seq=1", Col : 2 });

//설정할 "선택" 인자 json형식으로 속성 설정하기
mySheet.DoSave(PageUrl, { UrlEncode:0, Mode:2, Delim:"$" });
```



➤ 제공 버전

7.0.0.0	
7.0.1.0	저장 처리 여부에 따른 리턴값 추가
7.0.12.4	ValidKeyField, ValidFullInput 인자 속성 추가



(주) 아이비 리더스

Inno Biz Leaders Co., Ltd.

DoSearch Method

➤ 기능

조회 페이지를 연결하여 조회 XML을 읽어 들이고, XML의 데이터를 IBSheet 내부에 표현한다.

Param 인자는 조회조건을 "조건명1=값1&조건명2=값2" 식으로 "="와 "&"로 조건을 연결하여 설정한다.

Opt 인자는 객체형식의 인자로 동기조회여부(Sync) 와 Append조회여부(Append)를 설정 할 수 있다.

Sync 인자는 동기/비동기 조회 모드이다. 비동기일 경우 연속으로 호출시 이전 조회가 종료되지 않으면 이후의 조회는 모두 무시된다. 연속으로 호출해야 되고, 반드시 모든 조회가 완료되어야 한다면 동기 조회 모드를 사용해야 한다.

Append 인자를 1로 설정하는 경우 기존 데이터에 현재 조회 데이터를 첨부하여 조회 할 수 있다.

Url로 조회 페이지를 호출하고 조회 데이터를 읽어 들여 데이터 표현을 완료한 후 OnSearchEnd 이벤트를 발생하고 처리는 종료된다.

➤ Syntax

Syntax	ObjId.DoSearch(PageUrl, [Param], [Opt])
--------	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
PageUrl	String	필수	조회 XML 페이지 파일 이름
Param	String	선택	조회 조건 Query String, Default=""
Opt.Append	Boolean	선택	Append 조회 여부, Default=0
Opt.AppendRow	Number	선택	Append 조회 시 데이터를 붙여넣을 위치 설정
Opt.Fx	Boolean	선택	포매팅된 데이터 조회 처리 여부, Default=0

Opt.ReqHeader	Object	선택	요청헤더 설정 값 객체, Default={}
Opt.Sync	Boolean	선택	동기 조회 여부 Default=0 (SearchSync 설정 시 SearchSync 값)

➤ **Example**

```
// 1. 일반조회
mySheet.DoSearch("list.jsp", "p1=aa&p2=bb");

// 2. Sync 조회
var opt = { Sync : 1 };
mySheet.DoSearch("list.jsp", "p1=aa&p2=bb", opt);

// 3. Append 조회
var opt = { Append : 1 };
mySheet.DoSearch("list.jsp", "p1=aa&p2=bb", opt);

// 4. Sync && Append 조회
var opt = { Sync : 1, Append : 1 };
mySheet.DoSearch("list.jsp", "p1=aa&p2=bb", opt);

// 5. 요청헤더 설정
var opt = { ReqHeader : { "Prop" : "Value" } };
mySheet.DoSearch("list.jsp", "p1=aa&p2=bb", opt);
```

➤ **제공 버전**

7.0.0.0



DoSearchChild Method

➤ 기능

트리 형태의 데이터 구조에서 해당행의 자식을 조회하지 않은 상태에서 OnTreeChild 이벤트 안에서 자식데이터를 조회 페이지를 연결하여 조회 XML, JSON을 읽어 들이고, XML, JSON의 데이터를 자식으로 Append 하여 표현한다. **Row** 인자는 자식데이터를 Append할 부모 인자를 OnTreeChild 이벤트를 통해서 넘겨준다.

Url로 자식데이터로 조회할 페이지를 호출하고 조회 XML, JSON을 읽어 들여 데이터 표현을 완료한 후 OnSearchEnd 이벤트를 발생하고 처리는 종료된다.

Param 인자는 조회조건을 "조건명1=값1&조건명2=값2" 식으로 "="와 "&"로 조건을 연결하여 설정한다.

Wait 인자는 조회중 대기 이미지 표시 여부를 설정한다.

➤ Syntax

Syntax	ObjId.DoSearchChild(Row, PageUrl, [Param], [Opt])
--------	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	OnTreeChild 이벤트에서 받은 행의 Index
PageUrl	String	필수	조회 XML,JSON 페이지 파일 이름
Param	String	선택	조회 조건 Query String, Default=""
Opt.RegHeader	Object	선택	요청헤더 설정 값 객체, Default={}
Opt.Sync	Boolean	선택	동기 조회 여부 (동기상태에서는 대기 이미지가 표시되지 않음)



			Default=0 (SearchSync 설정 시 SearchSync 값)
Opt.Wait	Boolean	선택	대기 이미지 표시 여부, Default=1

➤ **Example**

```
// 자식데이터 조회하기
<script type="text/Javascript">
function mySheet_OnTreeChild(Row){
    var url = "";
    // 4컬럼 : 트리컬럼
    switch(mySheet.GetCellValue(Row, 4)){
        case "서울" :
            url = " type15_dat(1).xml";
            break;
        case "인천":
            url = "type15_data(2).xml";
            break;
    }
    var opt = { Wait : 1, Sync : 0 };
    mySheet.DoSearchChild(Row, url, "", opt);
}
</script>

// 1. 이미지 표시, 비동기 조회
var opt = { Wait : 1 Sync : 0 };
mySheet.DoSearchChild(Row, url, "", opt);

// 2. 이미지 미표시, 동기 조회 (동기조회 선택시 대기이미지는 보여지지 않음)
var opt = { Wait : 0, Sync : 1 };
mySheet.DoSearchChild(Row, url, "", opt);
```



- 제공 버전
7.0.0.0

DoSearchPaging Method

➤ 기능

대량 데이터를 조회 하고자 하는 경우 IBSheet 의 스크롤 위치의 데이터 일부만을 조회 하여 화면에 표현한다.

스크롤의 위치를 변경할 때마다 인자로 설정된 Url이 호출되며 데이터를 해당 위치에 표현한다.

해당 기능 사용시 초기화 함수인 SetConfig 에서 SearchMode를 3으로 설정하고 Page 속성을 반드시 설정하여야 한다.

전체 스크롤 사이즈는 조회 데이터의 Total 속성 설정값을 따르며, Total 속성 값을 반드시 설정 하여야 한다.

해당 기능으로 조회시 Row 및 Cell의 속성, 멀티 트랜잭션 기능 사용에 제약이 따른다.

➤ Syntax

Syntax	ObjId.DoSearchPaging(Url, [Opt])
--------	----------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Url	String	필수	조회 페이지 Url
Opt.PageParam	String	선택	페이지 인덱스를 받을 변수명, Default="ibpage"
Opt.Param	String	선택	조회 조건 Query String, Default=""
Opt.OrderByParam	String	선택	헤더 정렬 정보를 받을 변수명, Default="iborderby" 값은 "SIDO SIGUNGU ^ASC DESC"와 같이 savename 과 정렬 방향이 "^"로 구분되며 각 이름은 " "로 구분되어 있다.
Opt.	Boolean	선택	WaitImageVisible 설정이 true 일때



UseWaitImage			2페이지 이상 조회시 대기 이미지 표시 여부. Default=0
Opt ReqHeader	Object	선택	요청헤더 설정 값 객체, Default={}
Opt.Sync	Boolean	선택	동기 조회 여부 Default=0 (SearchSync 설정 시 SearchSync 값)

➤ **Example**

```
//초기화 설정 (페이지 사이즈를 100으로 설정)
var cfg = {SearchMode:3, Page:100};
mySheet.SetConfig(cfg);

// 실시간 조회
var info = {PageParam: "page", OrderbyParam:"orderbyParam", Param:
"id=ibleaders&seq=1"};
mySheet.DoSearchPaging("list.jsp",info);

// 동기 조회
var info = {PageParam: "page", OrderbyParam:"orderbyParam", Param:
"id=ibleaders&seq=1", Sync : 1};
mySheet.DoSearchPaging("list.jsp",info);
```

➤ **제공 버전**

7.0.0.0

Down2Excel Method

➤ 기능

조회된 데이터가 있는 경우 IBSheet의 내용을 엑셀 파일로 변환하여 다운로드 한다.

전달되는 인자의 형식은 JSON 타입으로 설정 하고자 하는 정보를 JSON 형태로 구성하여 전달한다.

예)

```
var params = { FileName : "myFile.xls", SheetName : "Sheet"} ;  
mySheet.Down2Excel(params);
```

AppendPrevSheet 속성은 [Down2ExcelBuffer](#) 메소드를 이용하여 2개 이상의 시트를 엑셀 파일로 다운로드 할 때 이전의 시트 내용을 작성한 워크시트에 현재의 시트 내용을 Append 하여 작성할지 여부를 설정한다. 설정값이 0(false)인 경우 워크시트를 새로 생성하여 작성되고, 1(true)인 경우 마지막 작성한 워크시트에 Append 하여 작성 된다. Buffer 모드의 첫번째의 시트인 경우와 Buffer 모드설정이 아닌 경우는 해당 속성의 설정값은 무시된다.

FileName 인자는 엑셀 다운로드시의 파일명을 설정한다. 파일 확장자를 xls 로 지정시 엑셀 2003 포맷으로 다운로드하고,xlsx 로 지정시 엑셀 2007 포맷으로 다운로드하며, 확장자를 지정하지 않을 경우에는 xls 포맷으로 다운로드 받는다.

SheetName 인자는 엑셀에 WorkSheet 이름을 설정하는것으로 데이터 양이 많아서 1개 WorkSheet에 내릴수 있는 최대 행의 개수인 65536개를 초과할 경우 원래 이름과 중복 사용할수 없으므로 이름 뒤에 괄호와 인덱스가 붙는다. 예를 들어 "1월"이라고 설정했다면 초과된 데이터는 새로운 WorkSheet에 "1월(1)", "1월(2)"라고 이름을 표시한다.

DownRows 인자는 다운로드 받을 행들을 "|" 문자로 연이은 문자열이다. 널인



경우 모든 행의 자료를 다운로드 받는다. "Visible" 일 경우 숨겨진 행을 제외한 현재 보여지는 행만 다운 받는다. **널이 아닐경우 DownTreeHide 인자는 false로 설정됩니다..**

DownCols 인자는 다운로드 받을 컬럼들을 "|" 문자로 연이은 문자열이다. SaveName과 컬럼인덱스가 모두 가능하다. 널인 경우 모든 컬럼의 자료를 다운로드 받는다.

HiddenColumn 인자는 숨은 컬럼들을 엑셀로 다운로드 받은 경우, 해당 컬럼이 눈에 보이지는 않지만 엑셀 메뉴중 "숨기기 취소"를 선택한 경우 해당 컬럼이 다시 보일 수 있도록 엑셀 문서에 다운로드 받는다.

DownHeader 인자는 헤더를 포함하여 다운로드 받을지 설정한다. 기본값은 1(헤더 포함) 이다.

DownSum 인자는 합계를 포함하여 다운로드 받을지 설정한다. 기본값은 1(합계 포함) 이다.

Merge 인자는 IBSheet의 머지 상태를 엑셀문서에 그대로 반영할지를 설정한다. 이 인자를 사용하는 경우 처리 속도가 더 느릴 수 있다. 기본값은 0(머지 반영 안함) 이다. 설정이 가능한 값은 아래와 같다

설정값	설명
0	사용 안함
1	사용함 (셀 병합시 부속셀의 값 설정 함)
2	사용함 (셀 병합시 부속셀의 값은 설정하지 않음)

※ 머지 적용 처리는 [Appendix#4. 엑셀 다운시 머지 적용 처리](#) 참조

SheetDesign 인자는 IBSheet 의 디자인을 반영한다. 폰트명, 폰트크기, 배경색 등을 지원하고 있다. 이 인자를 사용하는 경우 처리 속도가 더 느릴 수 있다. 기본값은 0(디자인 반영 안함)이다. 폰트색은 항상 검은색만 표시되며 다양한 폰트 색상의 반영을 지원하지 않는다. 셀 배경색은 1개의 엑셀파일 문서 내에서 최대 48가지의 동시 색상만을 지원한다. 따라서, 그 이상의 더 다양한 색상

이 1개의 IBSheet 내에 포함된 경우에는 기존 48가지 셀색상들중 가장 유사한 색상으로 대체 표현됨을 유의하여야 한다. 2로 설정한 경우 1 설정과 같이 디자인은 반영하되 셀 테두리는 그리지 않는다. 3으로 설정한 경우 셀 테두리 및 디자인을 모두 반영하지 않는다.

CheckBoxOnValue 인자는 체크박스과 라디오 박스에서 체크를 한 경우 1 값 대신 다른 값을 지정할 수 있다. 기본값 1.

CheckBoxOffValue 인자는 체크박스과 라디오 박스에서 체크해제를 한 경우 0 값 대신 다른 값을 지정할 수 있다. 기본값 0.

DownCombo 인자는 콤보와 콤보에디트의 선택항목을 TEXT 형태와 CODE 형태로 다운로드 받을 수 있다. 기본값은 "TEXT" 이며 "CODE"를 입력하면 TEXT 대신 해당 CODE 로 다운로드 받는다.

TitleAlign 인자는 TitleText로 설정한 제목이나 기타 내용에 대한 정렬을 설정한다. "left", "center", "right" 중 원하는 값으로 설정 가능하며 별도 설정하지 않는 경우 기본값인 "center"로 처리된다.

TitleText 인자는 그리드의 위쪽에 제목이나 기타 자료들을 사용자의 임의대로 넣을 수 있는 기능이다. A|B|CWrWnD|E|F 와 같이 입력한 경우 첫 행에 3개의 셀에 각각 A,B,C 값이 들어가고 두번째 행의 3개의 셀에 각각 D,E,F 값이 들어간다. 값 안에서 엔터를 포함하려면 Wr 이나 Wn 을 삽입하도록 한다. WrWn 이 10개가 포함되면 11줄을 차지하게 되고 12번째 행부터 IBSheet 자료가 이어진다.

이 인자에 넣은 셀자료들은 엑셀에 표시된 컬럼 수보다 더 많은 컬럼의 자료를 넣을 경우 그 값들은 무시된다. 즉 엑셀다운에 표시되는 컬럼이 모두 10개의 컬럼이면 타이틀이나 기타 자료들을 최대 10개까지만 가능하므로 11개 이상의 자료를 넣을 수는 없다. (타이틀 행의 제한은 별도로 없음)

UserMerge 인자는 TitleText 로 입력한 셀별 자료들이나 IBSheet에 해당되는 셀 자료들을 임의로 머지 적용하기 위하여 설정한다. 1개의 머지문은 4개의 숫



자와 콤마로 조합되며 여러 개의 머지는 공백(스페이스) 문자로 조합된다. "0,0,2,2 0,2,1,8" 과 같이 설정한 경우 엑셀의 첫행, 첫열인 (0,0)에서 2x2 크기의 머지가 하나 적용되고 첫행, 2열(=세번째컬럼)의 셀인(0,2) 에서 1x8 크기의 머지가 하나 더 적용된다.

OnlyHeaderMerge 인자는 데이터영역의 머지를 강제로 제한하여 속도를 개선할 때 사용한다.

ExcelFontSize 인자는 SheetFontSize 와는 별개로 엑셀문서에서만 특정 폰트 크기로 지정할 수 있다.

ExcelRowHeight 인자는 엑셀문서에서만 모든 행을 특정 픽셀 크기로 지정하기 위하여 사용한다. "auto"로 설정할 경우 자동으로 행 높이를 설정한다.

URL 인자는 Down2Excel과 더불어 서버에서 처리해야 하는 내용이 있는 경우,(가령 로그를 남긴다거나) 로직을 처리할 URL을 넣어주면 Down2Excel.jsp 페이지를 호출하기 전에 먼저 URL인자에서 설정한 페이지를 호출한다. 따라서 설정 페이지에서는 작업이 끝난 후, request를 Down2Excel.jsp 페이지로 전달하여야 한다.

예제)

```
var param = { URL:"/ibsheet7_down2excel_extendparam/fp.jsp"};  
mySheet.Down2Excel(param);
```

서버쪽 페이지)

```
RequestDispatcher rd=  
request.getRequestDispatcher("/ibsheet7_down2excel_extendparam/IBSheet/Down2Excel.jsp");  
rd.forward(request,response);
```

ExtendParam 인자는 서버로 전달해야 하는 내용이 있는 경우 Get방식의 QueryString으로 연결하여 설정하면 URL에 지정한 페이지로 같이 전달 된다.

예제)

```
param = { ExtendParam:"sawon_name=shkim&sawon_no=12345",  
URL:"/ibsheet7_down2excel_extendparam/fp.jsp };
```

ExtendParamMethod 인자는 ExtendParam 의 내용을 GET 또는 POST 로 전달할지를 설정한다.

TextToGeneral 인자는 IBSheet 의 컬럼이 Text Type 일 경우 엑셀로 다운되었을 때 일반 또는 텍스트 형식으로 변경할 때 사용한다.

AllTypeToText 인자는 모든 컬럼 타입을 Text타입으로 다운로드 받고자 하는 경우에 설정한다. NumberTypeToText 인자를 설정하지 않거나 0으로 설정하는 경우 숫자타입 컬럼을 제외한 모든 컬럼이 Text 타입으로 다운로드 되며 NumberTypeToText를 1로 설정한 경우 모든 컬럼이 Text타입으로 다운로드 처리된다.

AllTypeToText 인자를 1로 설정하는 경우 컬럼에 설정한 모든 포맷은 무시된다. 또한 TextToGeneral 을 설정한것과 동일하게 다운되므로 TextToGeneral 인자도 무시된다.

DownTreeHide 인자는 트리의 접혀진 행을 다운 받을 때 사용한다. True 일경우 SetRowHidden 으로 숨긴 로우도 모두 다운 받는다.

KeyFieldMark 인자는 KeyField 컬럼의 KeyField 마크(*)를 다운 받을 때 사용한다.

TreeLevel 인자는 트리구조 컬럼의 트리 레벨을 다운 받을 때 사용한다.

WordWrap 인자는 텍스트 셀의 줄바꿈을 허용할지를 설정한다.

AutoSizeColumn 인자는 컬럼 넓이를 내용에 맞게 자동으로 조절한다. 단, 자동조절 결과가 정확하게 맞지 않을수 있다.

ExcludeSubSum 인자는 다운 받는 행중에서 소계/누계 행을 제외하고 싶을

때 설정한다.

ReportXMLURL 특정 xml로 부터 엑셀파일에 설정할 머릿글,바닥글 등을 설정한다. Xml 파일에 대한 자세한 내용은 6.4장의 "**엑셀리포트 XML**"부분을 참고

ComboValidation 인자는 콤보 타입의 컬럼에 대하여 데이터영역(헤더, 소계, 누계, 합계행 제외)에 드롭다운 리스트 형식(데이터 유효성 검사)으로 생성할지를 설정한다. 기본값은 0으로 데이터 유효성 검사를 하지 않는다.

1로 설정시 리스트의 항목은 DownCombo 가 "Code"이면 컬럼에 설정된 ComboCode가 사용되며, "Text"일 경우 ComboText 값을 사용한다.

셀별로 InitCellProperty 로 변경한 Combo 목록은 엑셀의 드롭다운 리스트 목록에 별개로 적용되지 않으나 엑셀다운시 정상적인 값이 보여진다.

TempFile 인자는 템플릿으로 사용할 엑셀 파일을 서버에 두고 해당 파일에 시트 데이터만 삽입하여 다운로드할때 사용하는 속성으로 템플릿으로 사용할 엑셀 파일명을 지정한다. 템플릿 기능을 사용하는 경우 Down2Excel.jsp 파일 내에 TempRoot설정(사용자 환경 설정 #9) 이 되어 있어야 하며 템플릿 파일의 경로가 맞지 않는 경우 SheetDesign을 0으로 설정한 상태로 일반 다운로드 처리된다. StartRow, StartCol 인자를 통해 템플릿 파일에서 데이터를 쓰기 시작할 위치를 설정 할 수 있으며 템플릿 파일에서 데이터를 쓸 WorkSheet를 SheetName 또는 SheetNo로 설정할 수 있다.

SheetName을 설정한 경우 SheetNo보다 우선하며 유효하지 않은 값인 경우 SheetNo 설정값으로 처리된다.

템플릿 다운로드를 사용하는 경우 모든 서식/디자인은 템플릿 파일에 설정된 것을 사용하므로 ExcelFontSize, ExcelRowHeight, SheetDesign,ReportXMLURL 속성이 무시된다.

주의) 템플릿 파일에 날짜타입, 숫자 타입등의 서식 설정이 다운하고자 하는 시트 데이터 타입과 다르게 설정된 경우 데이터 값이 다르게 출력될 수 있다.

NumberFormatMode 속성은 실수 형태의 데이터타입에 대한 셀서식 설정 방식을 설정할 때 사용 한다. 설정값에 따른 처리 방식은 아래와 같다.

설정값	내용
-----	----

0	컬럼 단위 Format 기준 셀서식 설정
1	셀의 값 기준에 따라 정수 또는 실수 포맷형태의 셀서식 설정
2	일반 서식으로 설정 (별도의 셀 서식포맷을 설정하지 않음)

➤ **Syntax**

Syntax	ObjId. Down2Excel ([parameters])
--------	---

➤ **Info**

Parameter	Type	필수여부	설 명
AllTypeToText	Boolean	선택	모든 컬럼을 Text타입으로 다운로드 받을지 여부 Default=0
AppendPrevSheet	Boolean	선택	Buffer 모드 사용시 이전 워크시트에 Append 적용 여부 (Default=0)
AutoSizeColumn	Boolean	선택	Default=0(컬럼 넓이 자동 조절 안함)
CheckBoxOffValue	String	선택	체크박스의 체크 해제시의 값 Default="0"
CheckBoxOnValue	String	선택	체크박스의 체크시의 값 Default="1"
ComboValidation	Boolean	선택	콤보의 드롭다운 형식 다운로드 여부 Default=0
DownCols	String	선택	다운로드 받을 열들을 로 연결. Default=""(모두 받음)
DownCombo	String	선택	콤보의 TEXT/CODE 형태의 다운로드 여부 Default="TEXT"
DownHeader	Boolean	선택	헤더의 다운로드 여부 Default=1



DownRows	String	선택	다운로드 받을 행들을 로 연결. Default=""(모두 받음)
DownSum	Boolean	선택	합계의 다운로드 여부 Default=1
DownTreeHide	Boolean	선택	트리의 접혀진 행 다운 여부 Default=0(다운 안받음)
ExcelFontSize	Number	선택	폰트크기 설정 Default=0
ExcelRowHeight	String	선택	Default=""(사용안함)
ExcludeSubSum	Number	선택	소계/누계 행 제외 여부 1: 소계만 제외 2:누계만 제외 3:소계/누계 모두 제외 Default=0(소계/누계 모두 포함)
ExtendParam	String	선택	Default=""(사용안함)
ExtendParamMethod	String	선택	Default="GET"
FileName	String	선택	저장할 파일 명 Default="Excel.xls"
HiddenColumn	Boolean	선택	열숨김 반영 여부. Default=0
KeyFieldMark	Boolean	선택	KeyField 마크(*)를 다운 여부 Default=1(다운 받음)
Merge	Number	선택	머지의 다운로드 적용 방식 설정 (Default=0)
NumberFormatMode	Number	선택	실수 형태의 데이터 타입에 대한 셀서식 설정 방식 설정 (Default=0)
NumberTypeToText	Boolean	선택	숫자타입 셀 데이터를 문자 타입으로 다운로드 받을지 여부. 숫자타입 컬럼의 일부 셀에 포맷을 변경하여 문자열을 삽입하는 경우 사용

			Default=0
OnlyHeaderMerge	Boolean	선택	헤더만 머지할지의 여부 Default=0
ReportXMLURL	String	선택	엑셀파일에 타이틀이나 패턴등을 별도의 xml파일을 통해 설정
SheetDesign	Number	선택	디자인 다운로드 적용 여부 Default=0
SheetName	String	선택	엑셀WorkSheet이름, Default="Sheet"
SheetNo	Number	선택	템플릿으로 사용할 엑셀 WorkSheet 번호. Default=0
StartCol	Number	선택	템플릿 적용 다운로드 시 엑셀 파일내 데이터를 쓰기 시작할 컬럼 번호 Default=0
StartRow	Number	선택	템플릿 적용 다운로드 시 엑셀 파일내 데이터를 쓰기 시작할 행 번호 Default=0
TempFile	String	선택	템플릿으로 사용할 엑셀 파일명 파일명
TextToGeneral	Boolean	선택	Text 타입의 엑셀 서식 형식 Default=1(일반)
TitleAlign	String	선택	Default="center"(중앙정렬)
TitleText	String	선택	Default=""(사용안함)
TreeLevel	Boolean	선택	Default=0(다운 안받음)
URL	String	선택	Default=""(사용안함)
UserMerge	String	선택	Default=""(사용안함)
WordWrap	Boolean	선택	Default=1(줄바꿈 허용)



➤ **Example**

```
// 엑셀로 내려 받기
mySheet.Down2Excel();

// 다운로드할 파일명을 excel2로 하고 워크시트명은 sheet-test 로 정의하여
// 다운로드 받는다.
mySheet.Down2Excel({FileName:'excel2',SheetName:' sheet-test'});

//시트 색상과 머지를 모두 반영하며, 콤보는 코드로, 체크는 Y/N으로 다운로드
// 받되,헤더와 합계는 제외하고 컬럼은 왼쪽에서 3개의 컬럼만 다운로드 한
// 다.
mySheet.Down2Excel({SheetDesign:1, Merge:1, DownCombo:'CODE',
  CheckBoxOnValue:'Y', CheckBoxOffValue:'N', DownRows:'', DownCols:'0|1|2',
  DownHeader:0, DownSum:0});
```

➤ **제공 버전**

7.0.0.0	
7.0.2.0	AllTypeToText 속성 추가
7.0.13.11	AppendPrevSheet 속성 추가
7.0.13.12	Merge 속성 타입 변경
7.0.13.26	NumberFormatMode 속성 추가

Down2ExcelUrl Method

➤ 기능

엑셀 다운로드 기능을 처리할 서버 페이지 경로를 확인 및 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetDown2ExcelUrl()
--------	-----	----------------------------------

➤ Info

Return	String, 설정된 경로값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// 엑셀 다운로드 설정 경로를 확인한다.
var url = mySheet.GetDown2ExcelUrl();
```

➤ Syntax

Syntax	Set	ObjId. SetDown2ExcelUrl (Url)
--------	-----	--------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Url	String	필수	설정할 서버 페이지 Url

➤ Example

```
// 엑셀 다운로드 경로를 설정한다.
mySheet.SetDown2ExcelUrl("/jsp/Down2Excel.jsp");
```

➤ 제공 버전



7.0.0.0

Down2ExcelBuffer Method

➤ 기능

여러 개의 시트로부터 1개의 엑셀 문서로 다운로드 받는다.

Down2ExcelBuffer 의 buffer 인자를 true 로 설정하면 이후로 실행되는 Down2Excel은 실제로 동작하지 않으며 모두 내부 메모리에 버퍼링된다. 이후에 Down2ExcelBuffer 의 buffer 인자를 false 로 설정하는 순간 버퍼링된 모든 시트들을 하나의 엑셀 파일안에 시트별로 각각의 워크시트에 저장되어 다운로드 된다.

버퍼링되는 동안 지정되는 엑셀 파일명칭은 최초에 설정된 파일명 및 엑셀 파일 포맷이 유효하며, 워크시트명이 고유하지 않고 중복될 경우 자동적으로 괄호가 부여되어 다운로드 된다.

Down2ExcelBuffer 를 true로 설정하고 DirectDown2Excel을 호출하여 여러개의 시트를 하나의 엑셀 파일에 DirectDown2Excel 로 다운로드 할 수 있다.

시트에서의 호출방법은 Down2Excel과 동일하며 URL 인자는 모두 동일 페이지를 설정해야 한다.

데이터를 처리하는 jsp 파일에서는 DirectDown2Excel 을 호출한 시트 순서에 맞게 데이터를 생성하여 "SHEETDATA" 뒤에 1부터 일련번호를 붙여서 setAttribute 처리한다. 이때 첫번째 시트는 일련번호 없이 "SHEETDATA" 로 설정하고, 두번째 시트 이후부터 일련번호를 붙여서 설정해주면 된다.

➤ Syntax

Syntax	ObjId. Down2ExcelBuffer (IsBuffer)
--------	---

➤ Info

Parameter	Type	필수여부	설 명
IsBuffer	bool	필수	버퍼링 여부

➤ Example



```
//이후로는 버퍼링한다. 아무 동작 안함.  
firstSheet.Down2ExcelBuffer(true);  
  
// 첫번째 워크시트에 담아두기를 예약함.  
firstSheet.Down2Excel({FileName:'excel2',SheetName:'sheet1'});  
  
// 두번째 워크시트에 담아두기를 예약함.  
secondSheet.Down2Excel({FileName:'excel2',SheetName:' sheet2'});  
  
// 버퍼링된 모든 엑셀 자료를 1개의 엑셀문서에 모두 모아서 즉시 다운로드  
한다.  
firstSheet.Down2ExcelBuffer(false);
```

DirectDown2Excel 로 여러개의 시트를 다운로드 하는 경우

```
// 시트에서 Down2ExcelBuffer 통해 DirectDown2Excel 호출  
firstSheet.Down2ExcelBuffer(true);  
firstSheet.DirectDown2Excel({URL:"../jsp/fp.jsp",FileName:"IBLeaders.xls"});  
secondSheet.DirectDown2Excel({URL:"../jsp/fp.jsp",FileName:"IBLeaders.xls",  
SheetDesign:1});  
firstSheet.Down2ExcelBuffer(false);  
  
// 데이터 생성 파일 (fp.jsp)  
request.setAttribute("SHEETDATA", li);  
request.setAttribute("SHEETDATA1", li2);
```

➤ **제공 버전**
7.0.0.0

Down2Pdf Method

➤ 기능

조회된 데이터가 있는 경우 IBSheet의 내용을 PDF 파일로 변환하여 다운로드 한다.

이 메서드의 동작방식은 다음과 같다.

- 1) Down2Pdf()메서드를 호출하면 시트의 내용이 Multipart 형태로 서버의 Down2Pdf.jsp/Down2Pdf.aspx 파일에 전달된다.
- 2) Down2Pdf.jsp/Down2Pdf.aspx 파일에서는 넘어온 내용을 iText_2.1.7 라이브러리를 이용하여 pdf 포맷의 파일로 변경하여 다시 화면쪽으로 다운로드 시킨다.

따라서 이 메서드를 사용하려면 다음과 같은 준비과정이 필요하다.

* Java 환경

1. Down2Pdf.jsp 파일이 서버에 존재하며, ibsheet.cfg 파일에 jsp파일에 대한 경로가 정확하게 기재 되어 있어야 한다.
2. 서버에 iText 라이브러리가 올라와 있어야 한다.
3. 서버에 한글폰트 지원을 위한 ttf 파일이 있어야 하며 jsp 파일에 폰트 폴더에 대한 경로가 정확하게 기재되어 있어야 한다.
4. **Down2Pdf.jsp 파일을 편집기로 열어, 사용하는 서버의 폰트파일 경로와 Encoding을 설정해 준다.**

* .Net 환경

1. Down2Pdf.aspx 파일이 서버에 존재하며, ibsheet.cfg 파일에 aspx파일에 대한 경로가 정확하게 기재 되어 있어야 한다.
2. .Net모듈의 경우 자바 모듈을 통해 html 파일을 pdf 파일로 변환처리 하여 다운로드 하기 때문에 서버에 JVM, iText 라이브러리 및 자바버전 모듈이 모두 올라와 있어야 한다.
3. 서버에 한글폰트 지원을 위한 ttf 파일이 있어야 하며 ttf 파일은 자바버전 모



들과 같은 위치에 두고 aspx 파일의 ModulePath 에 해당 위치를 정확히 설정해 준다.

4. Down2Pdf.aspx 파일을 편집기로 열어, 사용하는 서버에서 임시 html파일, pdf 파일을 생성하기 위한 TempFolder(쓰기 권한이 있어야 한다.)와 ttf/jar파일이 위치한 ModulePath, Encoding을 설정해 준다. 자바 모듈인 ibsheet7-1.1.4.jar 파일은 버전에 따라 파일명이 변경되므로 서버에 적용한 자바모듈 파일명을 ModuleFile 정확히 설정해준다.

전달되는 인자의 형식은 JSON 타입으로 설정 하고자 하는 정보를 JSON 형태로 구성하여 전달한다.

예)

```
var params = { FileName:"myPDF.pdf"};  
mySheet.Down2Pdf(params);
```

FileName 인자는 PDF 다운로드시의 파일명을 설정한다. 파일 확장자는 반드시 pdf 여야 한다. 확장자를 생략할시 자동으로 .pdf 를 붙여 다운로드 한다.

DownCols 인자는 다운로드 받을 컬럼들을 "|" 문자로 연이은 문자열이다. SaveName과 컬럼인덱스가 모두 가능하다. 널인 경우 모든 컬럼의 자료를 다운로드 받는다.

예)

```
var params = { DownCols:"4|5|6|7|8|9|10"};  
mySheet.Down2Pdf(params);
```

Paper 인자는 PDF 변환시 용지 방향을 설정한다. 가로 : landscape 세로 : portrait

예)



```
var params = { Paper:" landscape"};  
mySheet.Down2Pdf(params);
```

Dpi 인자는 PDF 변환시 확대 비율을 설정한다. 50 ~ 32800까지 설정 가능하며 값이 클수록 작게 출력된다.

예)

```
var params = { Dpi:1800};  
mySheet.Down2Pdf(params);
```

Title 인자는 PDF 파일에 출력할 제목을 설정한다.

TitleStyle 인자는 PDF 파일에 출력할 제목에 적용할 css style을 설정한다.

예)

```
var params = {Title:"IB Sheet PDF 파일", TitleStyle:"color:red;size:12pt;" };  
mySheet.Down2Pdf(params);
```

URL 인자는 Down2Pdf 와 더불어 서버에서 처리해야 하는 내용이 있는 경우,(가령 로그를 남긴다거나) 로직을 처리할 URL을 넣어주면 Down2Pdf.jsp 페이지를 호출하기 전에 먼저 URL인자에서 설정한 페이지를 호출한다. 따라서 설정 페이지에서는 작업이 끝난 후, request를 Down2Pdf.jsp 페이지로 전달하여야 한다.

예)

```
var param = { URL:"/ibsheet7_down2pdf/fp.jsp"};  
mySheet.Down2Pdf(param);
```

서버쪽 페이지)

```
RequestDispatcher rd= request.getRequestDispatcher("/ibsheet7_down2pdf/  
Down2Pdf.jsp");  
rd.forward(request,response);
```



ExtendParam 인자는 서버로 전달해야 하는 내용이 있는 경우 Get방식의 QueryString으로 연결하여 설정하면 URL에 지정한 페이지로 같이 전달 된다.

예)

```
param = { ExtendParam:"sawon_name=shkim&sawon_no=12345",  
URL: "/ibsheets7_down2pdf/fp.jsp"};  
mySheet.Down2Pdf(param);
```

ExtendParamMethod 인자는 ExtendParam 의 내용을 GET 또는 POST 로 전달할지를 설정한다.

FontTo 인자는 PDF에 사용한 한글 폰트를 설정한다. 시트에 한글을 사용한 경우 PDF 변환 모듈에서 적용 가능한 한글 폰트인 굴림 또는 맑은 고딕중 선택하여 영문으로 설정한다. (Gulim 또는 Gothic)

예)

```
var params = { FontTo:" Gulim"};  
mySheet.Down2Pdf(params);
```

➤ Syntax

Syntax	ObjId. Down2Pdf ([parameters])
--------	---------------------------------------

➤ Info

Parameter	Type	필수여부	설 명
DownCols	String	선택	다운로드 받을 열들을 로 연결. Default=""(모두 받음)
Dpi	Integer	선택	축소/확대 비율. 값이 작을 수록 크게 출력된다. 50~32840 사이 값으로 설정 가능하다. Default = 2000

ExtendParam	String	선택	서버로 전달해야 하는 내용이 있는 경우 Get 방식의 QueryString으로 연결하여 설정Default=""
ExtendParam	String	선택	Default=""(사용안함)
ExtendParamMethod	String	선택	Default="GET"
FileName	String	선택	저장할 파일 명 Default="IBSheet.pdf"
FontTo	String	선택	Default = "Gothic"
Paper	String	선택	용지 방향 설정 landscape 또는 portrait Default = "landscape"
Title	String	선택	Default = ""
TitleStyle	String	선택	Default = ""
URL	String	선택	Default=""(사용안함)

➤ **Example**

```
// PDF 파일로 내려 받기
mySheet.Down2Pdf();

// 다운로드할 파일명을 text로 정의하여다운로드 받는다.
mySheet.Down2Pdf({FileName:'text' });

// 다운로드할 컬럼을 지정하여 myPDF.pdf로 다운받는다.
mySheet.Down2Pdf({FileName:"myPDF", DownCols:"7|8|9|4|5|6|10"});
```

➤ **제공 버전**

7.0.0.0



Down2PdfUrl Method

➤ 기능

PDF 다운로드 기능을 처리할 서버 페이지 경로를 확인 및 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetDown2PdfUrl()
--------	-----	--------------------------------

➤ Info

Return	String, 설정된 경로값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// PDF 다운로드 설정 경로를 확인한다.  
var url = mySheet.GetDown2PdfUrl();
```

➤ Syntax

Syntax	Set	ObjId. SetDown2PdfUrl (Url)
--------	-----	------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Url	String	필수	설정할 서버 페이지 Url

➤ Example

```
// PDF 다운로드 경로를 설정한다.  
mySheet.SetDown2PdfUrl("/jsp/Down2Pdf.jsp");
```

➤ 제공 버전

7.0.0.0

Down2Text Method

➤ 기능

조회된 데이터가 있는 경우 IBSheet의 내용을 텍스트 파일로 변환하여 다운로드 한다.

전달되는 인자의 형식은 JSON 타입으로 설정 하고자 하는 정보를 JSON 형태로 구성하여 전달한다.

예)

```
var params = { FileName : "myFile.txt "};  
mySheet.Down2Text(params);
```

FileName 인자는 텍스트 다운로드시의 파일명을 설정한다. 파일 확장자는 반드시 txt 여야 한다. 확장자를 생략할시 자동으로 .txt 를 붙여 다운로드 한다.

DownRows 인자는 다운로드 받을 행들을 "|" 문자로 연이은 문자열이다. 널인 경우 모든 행의 자료를 다운로드 받는다.

DownCols 인자는 다운로드 받을 컬럼들을 "|" 문자로 연이은 문자열이다. SaveName과 컬럼인덱스가 모두 가능하다. 널인 경우 모든 컬럼의 자료를 다운로드 받는다.

DownHeader 인자는 헤더를 포함하여 다운로드 받을지 설정한다. 기본값은 1(헤더 포함) 이다.

DownSum 인자는 합계를 포함하여 다운로드 받을지 설정한다. 기본값은 1(합계 포함) 이다.

DownCombo 인자는 콤보와 콤보에디트의 선택항목을 TEXT 형태와 CODE 형태로 다운로드 받을 수 있다. 기본값은 "TEXT" 이며 "CODE"를 입력하면 TEXT 대신 해당 CODE 로 다운로드 받는다.

ExtendParam 인자는 서버로 전달해야 하는 내용이 있는 경우 Get 방식의 QueryString으로 연결하여 설정하면 URL에 지정한 페이지로 같이 전달 된다.
예제)

```
param = {ExtendParam:"sawon_name=shkim&sawon_no=12345",  
URL:"/ibsheet7_down2text_extendparam/fp.jsp };
```

DownTreeHide 인자는 트리의 접혀진 행을 다운 받을 때 사용한다.

➤ **Syntax**

Syntax	ObjId. Down2Text ([parameters])
--------	--

➤ **Info**

Parameter	Type	필수여부	설 명
FileName	String	선택	저장할 파일 명 Default="Test.txt"
RowDelim	String	선택	행 자료 사이에 표시될 레코드 구분자 Default="\n" (엔터 형식)
ColDelim	String	선택	셀 자료 사이에 표시될 컬럼 구분자 Default=" " (공백 형식)
DownRows	String	선택	다운로드 받을 행들을 로 연결. Default=""(모두 받음)
DownCols	String	선택	다운로드 받을 열들을 로 연결. Default=""(모두 받음)
DownHeader	bool	선택	헤더의 다운로드 여부 Default=1
DownSum	bool	선택	합계의 다운로드 여부 Default=1

DownCombo	String	선택	콤보의 TEXT / CODE 형태의 다운로드 여부 Default="TEXT"
ExtendParam	String	선택	서버로 전달해야 하는 내용이 있는 경우 Get 방식의 QueryString으로 연결하여 설정 Default=""
DownTreeHide	bool	선택	트리의 접혀진 행 다운 여부 Default=0(다운 안받음)

➤ **Example**

```
// 텍스트 파일로 내려 받기
mySheet.Down2Text();

// 다운로드할 파일명을 text로 정의하여다운로드 받는다.
mySheet.Down2Text({FileName:'text' });

// 콤보는 코드로 받고, 헤더와 합계는 제외하고 컬럼은 왼쪽에서 3개의 컬럼만 다운로드 한다.
mySheet.Down2Text({DownCombo:'CODE', DownRows:'', DownCols:'0|1|2',
DownHeader:0, DownSum:0});
```

➤ **제공 버전**

7.0.0.0



Down2TextUrl Method

➤ 기능

텍스트 다운로드 기능을 처리할 서버 페이지 경로를 확인 및 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetDown2TextUrl()
--------	-----	---------------------------------

➤ Info

Return	String, 설정된 경로값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// 텍스트 다운로드 설정 경로를 확인한다.  
var url = mySheet.GetDown2TextUrl();
```

➤ Syntax

Syntax	Set	ObjId. SetDown2TextUrl (Url)
--------	-----	-------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Url	String	필수	설정할 서버 페이지 Url

➤ Example

```
// 텍스트 다운로드 경로를 설정한다.  
mySheet.SetDown2TextUrl("/jsp/Down2Text.jsp");
```

➤ 제공 버전

7.0.0.0

DownloadingImage Method

➤ 기능

파일 다운로드 중 대기 이미지 파일의 위치를 설정하거나 확인한다.

이 속성은 기본적으로 제공하는 파일 업로드 중 대기 이미지를 사용자가 원하는 이미지로 변경설정 가능하기 위한 속성이다.

➤ Syntax

Syntax	Get	ObjId. GetDownloadingImage()
--------	-----	-------------------------------------

➤ Info

Return	String, 현재 설정값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//현재 설정되어있는 다운로드 중 대기 이미지 경로를 확인한다.
Alert(mySheet.GetDownloadingImage());
```

➤ Syntax

Syntax	Set	ObjId. SetDownloadingImage (Url)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Url	String	필수	이미지 URL

➤ Example

```
//다운로드 중 대기 이미지를 변경한다.
mySheet.SetDownloadingImage( "/sheet/imgDownload.gif");
```



- 제공 버전
7.0.0.0

DragMode Method

➤ 기능

마우스 드래깅시 처리 방법을 설정하거나 확인 한다.

설정 값에 따른 처리방식은 아래와 같다.

설정값	내용	
0 (Default)	일반	셀 또는 행 범위 선택
	Ctrl Key 사용	행 드래깅
1	일반	행 드래깅
	Ctrl Key 사용	셀 또는 행 범위 선택

➤ Syntax

Syntax	Get	ObjId. GetDragMode()
--------	-----	-----------------------------

➤ Info

Return	Boolean, 설정된 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//DragMode 설정 값 확인
var mode = mySheet.GetDragMode();
```

➤ Syntax

Syntax	Set	ObjId. SetDragMode (Mode)
--------	-----	----------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Mode	Boolean	필수	드래그 처리 설정 값 (Default=0)



➤ **Example**

```
// DragMode 방식 설정 (마우스 드래깅시 행 드래깅 처리 방식 설정)
mySheet.SetDragMode(1);
```

➤ **제공 버전**

7.0.0.0

Editable Method

➤ 기능

전체적인 Edit 허용 여부를 확인하거나 설정한다.

전체적으로 Edit가 불가능하면 다른 설정에 관계없이 모든 Edit는 불가능하다.

전체 Edit 가능 여부는 다음과 같이 결정된다.

Editable	ColEditable	RowEditable	CellEditable	셀의 Edit 가능 여부
불가	무관	무관	무관	불가
가능	불가	불가	가능/불가	가능/불가
가능	가능	가능/불가	가능/불가	가능/불가

➤ Syntax

Syntax	Get	ObjId. GetEditable()
--------	-----	-----------------------------

➤ Info

Return	Boolean, 설정된 편집 여부 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

//Edit 가능 여부 확인 mySheet.GetEditable();

➤ Syntax

Syntax	Set	ObjId. SetEditable(Edit)
--------	-----	---------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Edit	Boolean	필수	설정할 편집 여부 값



➤ **Example**

```
//초기 로드 시 전체 적인 Edit 가능 설정  
mySheet.SetEditable(1);
```

➤ **제공 버전**
7.0.0.0

EditEnterBehavior Method

➤ 기능

데이터를 Edit 후 Enter 키를 눌렀을 때 동작을 확인하거나 설정한다
이 속성은 EnterBehavior Method와 구분하여 사용해야 한다. EnterBehavior Method 는 편집상태가 아닌 단순히 포커스가 있는 상태에서 Enter 키를 누른 경우이고, 이 속성은 편집상태에서 입력을 종료하기 위해 Enter 키를 누른 상태로 구분하여 사용한다.

Newline 속성값 설정시 InitColumns에서 MultiLineText 속성이 1로 설정된 컬럼에 대하여 개행처리가 된다.

값	설명
tab	편집 완료 후 오른쪽 셀로 포커스 이동
down	Down 키를 누른 것 처럼 아래셀로 이동
newline	줄바꿈되어 개행 처리 됨 (가능한 경우)
none	편집 완료 후 해당 셀 포커스 유지
editTab	편집 완료 후 다음 편집 가능한 셀로 이동

➤ Syntax

Syntax	Get	ObjId. GetEditEnterBehavior()
--------	-----	--------------------------------------

➤ Info

Return	String, 설정한 속성 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// Enter키의 동작을 확인한다.
mySheet.GetEditEnterBehavior( );
```



➤ **Syntax**

Syntax	Set	ObjId. SetEditEnterBehavior (Mode)
--------	-----	---

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Mode	String	필수	편집상태에서 Enter key 입력시 설정할 속성 값. Default="tab"

➤ **Example**

```
// Edit 후 Enter를 누르면 무조건 아래로 이동하게 한다.  
mySheet.SetEditEnterBehavior( "down");
```

➤ **제공 버전**

7.0.0.0	
7.0.13.16	"editTab" 모드 추가

EditableColorDiff Method

➤ 기능

편집이 불가능한 셀을 색상으로 구분하여 표시할지 여부를 확인하거나 설정한다.

설정 가능한 종류는 다음과 같다.

종류	설명
0	편집불가능한 셀을 구분없이 표시
1	편집불가능한 셀을 css에서 설정한 색상으로 표시
2	편집불가능한 셀을 css의 설정값과 기본배경색의 조합으로 표시

➤ Syntax

Syntax	Get	ObjId.GetEditableColorDiff()
--------	-----	------------------------------

➤ Info

Return	Integer, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

//편집 불가능한 셀의 표시방법 확인 mySheet.GetEditableColorDiff ();
--

➤ Syntax

Syntax	Set	ObjId.SetEditableColorDiff(Mode)
--------	-----	----------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명



Mode	Integer	필수	설정 값
------	---------	----	------

➤ **Example**

```
//편집 불가능한 셀을 구분하지 않음  
mySheet.SetEditableColorDiff (0);
```

➤ **제공 버전**

7.0.0.0

EditArrowBehavior Method

➤ 기능

편집모드시 화살표키(상,하,좌,우)에 대한 셀의 포커스 이동 동작을 확인하거나 설정한다.

화살표키	셀 상하 이동	셀 좌우 이동
0	불가	불가
1	가능	불가
2	불가	가능
3	가능	가능

➤ Syntax

Syntax	Get	ObjId. GetEditArrowBehavior()
--------	-----	--------------------------------------

➤ Info

Return	Integer, 설정된 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

// 설정값 확인 mySheet.GetEditArrowBehavior();
--

➤ Syntax

Syntax	Set	ObjId. SetEditArrowBehavior(behavior)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
behavior	Integer	필수	설정할 동작 값 (Default=1)



➤ **Example**

```
// 편집모드인 상태에서 화살표키(상,하,좌,우)를 눌렀을때 상하,좌우셀로 포커스이동가능  
mySheet.SetEditArrowBehavior(3);
```

➤ **제공 버전**

7.0.0.0

EditTabBehavior Method

➤ 기능

데이터를 Edit 후 Tab 키를 눌렀을 때 동작을 확인하거나 설정한다
설정값에 대한 동작은 다음과 같다.

값	설명
0	다음 편집이 가능한 셀로 이동 (편집상태 유지)
1	편집여부와 무관하게 다음셀로 이동 (다음셀이 편집 가능한 경우면 편집상태 유지)
2	편집여부와 무관하게 다음셀로 이동 (다음셀의 편집 가능여부와 무관하게 편집상태 종료)

➤ Syntax

Syntax	Get	ObjId. GetEditTabBehavior()
--------	-----	------------------------------------

➤ Info

Return	String, 설정한 속성 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

// Tab키의 동작을 확인한다. mySheet.GetEditTabBehavior();

➤ Syntax

Syntax	Set	ObjId. SetEditTabBehavior (Mode)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명



Mode	Number	필수	편집상태에서 Tab key 입력시 설정할 속성 값. Default="0"
------	--------	----	--

➤ **Example**

```
// Edit 후 Tab을 누르면 무조건 다음 셀로 이동하게 한다.  
mySheet.SetEditTabBehavior(1);
```

➤ **제공 버전**

7.0.0.0

Ellipsis Method

➤ 기능

말줄임 표시 기능을 사용할 것인지 여부를 설정하거나 확인한다.

셀에 설정된 Text의 길이가 해당 컬럼의 너비 보다 클 때 기본적으로 잘려서 표시된다. 이러한 Text를 "..."와 같은 말 줄임 표시로 설정할 때 이 속성을 사용한다.

이외에도 InitColumns에 Wrap:1속성을 사용하면 내용이 잘리지 않고 자동줄바꿈 기능을 설정하여 사용할 수도 있다.

➤ Syntax

Syntax	Get	ObjId. GetEllipsis()
--------	-----	-----------------------------

➤ Info

Return	Boolean, 사용가능 여부 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

// 설정값 확인 mySheet.GetEllipsis();

➤ Syntax

Syntax	Set	ObjId. SetEllipsis (Flag)
--------	-----	----------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Flag	Boolean	필수	말줄임 표시기능 사용여부, Default=0



➤ **Example**

```
// 말줄임 기능 사용하기  
mySheet.SetEllipsis(1);
```

➤ **제공 버전**
7.0.0.0

Enable Method

➤ 기능

이 속성값을 0로 설정하면 마우스나 키보드를 사용한 User Interface 기능은 모두 사용 불가능하고, 기타 제품이 제공하는 Method나 Property는 코딩으로 호출되므로 사용할 수 있다.

➤ Syntax

Syntax	Get	ObjId. GetEnable()
--------	-----	---------------------------

➤ Info

Return	Boolean, 사용가능 여부 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//User Interface 사용가능 여부 값을 확인한다.
mySheet.GetEnable();
```

➤ Syntax

Syntax	Set	ObjId. SetEnable (Enable)
--------	-----	----------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Enable	Boolean	필수	User Interface 사용 가능 여부

➤ Example

```
//사용 불가능 상태로 변경한다.
mySheet.SetEnable(0);
```



```
//사용 가능 상태로 변경한다.  
mySheet.SetEnable(1);
```

➤ **제공 버전**

7.0.0.0

EnterBehavior Method

➤ 기능

셀에서 포커스를 두고 TAB키를 누르면 옆셀로 포커스가 이동하고, Enter 키를 누르면 Edit를 시작한다. 이런 Enter의 기능을 다른 기능으로 사용할 때 이 속성을 설정한다.

기본적으로 Edit를 시작하지만 Enter를 눌렀을 때 TAB키를 누른 것 처럼 포커스를 이동하고자 한다면 이 속성을 "tab"이라고 설정한다

값	설명
"tab"	탭을 누른 것 처럼 옆셀로 이동
"edit"	Edit를 시작 (Default)
"down"	아래로 내려간다
"none"	아무런 동작 하지 않음

➤ Syntax

Syntax	Get	ObjId. GetEnterBehavior()
--------	-----	----------------------------------

➤ Info

Return	String, 설정한 속성 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

//설정을 확인하다. mySheet.GetEnterBehavior();
--

➤ Syntax

Syntax	Set	ObjId. SetEnterBehavior (Mode)
--------	-----	---------------------------------------

➤ Info



Return	None		
Parameter	Type	필수여부	설 명
Mode	String	필수	Enter key 입력시 설정할 속성 값 Default="edit"

➤ **Example**

```
//Enter를 누르면 무조건 옆으로 이동하게 한다.  
mySheet.SetEnterBehavior("tab");
```

➤ **제공 버전**

7.0.0.0

EtcData Method

➤ 기능

데이터 정보 이외의 정보를 확인하거나 설정한다.

이 속성은 조회 함수를 이용하여 조회 결과 또는 저장 처리 결과에서 추가 정보를 조회 한 경우 기타 정보로 저장할 수 있다.

기타 정보는 키 이름과 키 값으로 구성 되며 속성을 바로 설정하거나 확인할 수 있다.

XML 포맷인 경우는 <ETC-DATA> 태그를 사용하며 다음과 같이 사용한다.

기본 구조	<ETC-DATA> <ETC KEY="키이름">키값</ETC> <ETC KEY="name">홍길동</ETC> <ETC KEY="age">30</ETC> </ETC-DATA>
조회 시	<SHEET> 태그와 <DATA> 태그 사이에 설정한다. 예) <?xml version='1.0' ?> <SHEET> <ETC-DATA> <ETC KEY="name">홍길동</ETC> <ETC KEY="age">30</ETC> </ETC-DATA> <DATA> <TR> <TD>CWOOF-171</TD> <TD>17평</TD> <TD>2040000</TD> <TD>2101200</TD> </TR> </DATA> </SHEET>



저장 시	<p><SHEET> 태그와 <RESULT> 태그 사이에 설정한다. 예)</p> <pre><?xml version='1.0' ?> <SHEET> <ETC-DATA> <ETC KEY="name">홍길동 </ETC> <ETC KEY="age">30</ETC> </ETC-DATA> <RESULT Code="0" Message="저장성공" /> </SHEET></pre>
------	--

JSON 포맷인 경우는 etc 키를 사용하며 다음과 같이 사용한다.

기본 구조	etc: { "키이름":"키값", name: "홍길동", age: 30}
조회 시	<p>예)</p> <pre>{ etc:{ name: "홍길동", age: 30}, data:[{C1: "CWOFF-171", C2: "17평"}, ...] }</pre>
저장 시	<p>예)</p> <pre>{ etc:{ name: "홍길동", age: 30}, result:[{Code:0, Message: "저장성공"}, ...] }</pre>

➤ **Syntax**

Syntax	Get	ObjId. GetEtcData (KeyName)
--------	-----	------------------------------------



➤ **Info**

Return	String, 키에 설정된 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
KeyName	String	필수	기타정보 키 이름

➤ **Example**

```
// XML을 통해서 조회된 기타 정보를 TextBox에 설정한다.  
document.form1.txtAge.value = mySheet.GetEtcData("age")  
  
// 저장 처리 완료 후 기타 정보를 이용하여 페이지를 이동한다.  
mySheet.DoSave("save.html");  
location.href = "/site/showmaster.html?keyinfo=" + mySheet.GetEtcData("전표  
번호")
```



➤ **Syntax**

Syntax	Set	ObjId. SetEtcData (KeyName, Value)
--------	-----	---

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
KeyName	String	필수	기타정보 키 이름
Value	String	필수	기타정보 키 값

➤ **Example**

<pre>// 기타 정보의 값을 변경한다. mySheet.SetEtcData("age", 40); // 새로운 기타정보를 생성한다. mySheet.SetEtcData("Pay", 2000000);</pre>
--

➤ **제공 버전**

7.0.0.0



EtcDataString Method

➤ 기능

EtcData 객체에 설정된 키와 값을 모두 조합하여 "키이름1=키값1&키이름2=키값2"와 같은 QueryString 형태로 반환 한다.

➤ Syntax

Syntax	ObjId. EtcDataString ([UrlEncode])
--------	---

➤ Info

Return	String, EtcData에 설정된 키와 값에 대한 QueryString 문자열		
Parameter	Type	필수여부	설 명
UrlEncode	Boolean	선택	인코딩 처리 여부 (Default = 0)

➤ Example

```
// EtcData 영역의 값을 문자열로 받는다.  
var param = mySheet.EtcDataString();  
//저장시 인자로 넘긴다.  
mySheet.DoSave("Save.jsp", Param);
```

➤ 제공 버전

7.0.0.0



ExportData Method

➤ 기능

시트내의 데이터를 json,xml,csv 형식으로 추출한다.

옵션을 통해 지정한 컬럼만 추출도 가능하다.

```
<?xml version="1.0" encoding="UTF-8" ?>
<data>
  <row>
    <COLSAVENAME1><![CDATA[data....]]></COLSAVENAME1>
    <COLSAVENAME2><![CDATA[data....]]></COLSAVENAME2>
    ....
  </row>
</data>
```

[xml 형식 추출]

```
{"data":[
{" COLSAVENAME1": data," COLSAVENAME2":"data2"...}
....
]}
```

[json 형식 추출]

```
"data1,data2,.....WrWn...."
```

[csv형식 추출]

➤ Syntax

Syntax	ObjId. ExportData (info)
--------	---------------------------------

➤ Info

Return	Type으로 지정한 형식의 데이터		
Parameter	Type	필수여	설 명

		부																										
info.Type	String	필수	json,xml,csv 중에 하나 선택																									
info.Cols	String	선택	추출할 컬럼(Default:모든컬럼)																									
info.FormattedText	boolean	선택	포맷이 적용된 문자열형식으로 추출할지 여부(Default:0)																									
info.StyleProperty	boolean	선택	추출데이터에 편집여부,배경색,글자색에 대한 데이터도 포함할 지 여부 (json 형식인 경우에만 사용가능) <table><tr><th>대상</th><th>속성</th><th>설명</th></tr><tr><td rowspan="3">Row</td><td>Edit</td><td>행 편집여부</td></tr><tr><td>BackColor</td><td>행 배경색</td></tr><tr><td>FontColor</td><td>행 글자색</td></tr><tr><td rowspan="7">Cell</td><td>Edit</td><td>셀 편집여부</td></tr><tr><td>BackColor</td><td>셀 배경색</td></tr><tr><td>FontColor</td><td>셀 글자색</td></tr><tr><td>FontBold</td><td>셀 글자 굵기</td></tr><tr><td>FontItalic</td><td>셀 글자 italic</td></tr><tr><td>FontStrike</td><td>셀 글자 strike</td></tr><tr><td>FontUnderline</td><td>셀 글자 밑줄</td></tr></table>	대상	속성	설명	Row	Edit	행 편집여부	BackColor	행 배경색	FontColor	행 글자색	Cell	Edit	셀 편집여부	BackColor	셀 배경색	FontColor	셀 글자색	FontBold	셀 글자 굵기	FontItalic	셀 글자 italic	FontStrike	셀 글자 strike	FontUnderline	셀 글자 밑줄
대상	속성	설명																										
Row	Edit	행 편집여부																										
	BackColor	행 배경색																										
	FontColor	행 글자색																										
Cell	Edit	셀 편집여부																										
	BackColor	셀 배경색																										
	FontColor	셀 글자색																										
	FontBold	셀 글자 굵기																										
	FontItalic	셀 글자 italic																										
	FontStrike	셀 글자 strike																										
	FontUnderline	셀 글자 밑줄																										

➤ **Example**

```
var param = {"Type":"json","Cols":"0|2|3|4"};
var json = mySheet.ExportData(param);
```

➤ **제공 버전**

7.0.13.21

7.0.13.44 (FormattedText , StyleProperty 추가)



ExtendLastCol Method

➤ 기능

마지막 컬럼의 너비를 전체 너비에 맞게 자동으로 맞추어줄지 여부를 확인하거나 설정한다.

이 속성은 Object 너비에 비해서 컬럼 전체 너비의 합이 작을 때 마지막 컬럼을 Object 너비에 맞게 자동으로 넓어지는 기능을 사용하기 위함이다.

➤ Syntax

Syntax	Get	ObjId. GetExtendLastCol()
--------	-----	----------------------------------

➤ Info

Return	Boolean, 확장 설정 여부 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
없음			

➤ Example

//설정 여부를 확인한다. mySheet.GetExtendLastCol()
--

➤ Syntax

Syntax	Set	ObjId. SetExtendLastCol (Extend)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Extend	Boolean	필수	마지막 컬럼 너비 확장 여부 Default= 0

➤ Example



```
//마지막 컬럼을 전체 너비에 맞춘다..  
mySheet.SetExtendLastCol(1)
```

- **제공 버전**
7.0.0.0



FilteredRowCount Method

➤ 기능

필터행이 있는 경우 필터가 적용되어 있으면 필터링 된 행의 갯수(출력된 행)를 반환하며 필터가 적용되어 있지 않으면 전체 행 수를 반환한다.

필터행이 없는 경우 -1을 반환한다.

행추가 시 추가된 행의 데이터가 필터 조건에 맞지 않을 수 있으나 행 갯수에는 반영된다. 정확한 결과를 얻으려면 행 추가하여 데이터 수정 후 재 필터링을 해야 한다.

➤ Syntax

Syntax	ObjId.FilteredRowCount()
--------	--------------------------

➤ Info

Return	Integer. 필터링 된 행 갯수		
Parameter	Type	필수여부	설 명
			없음

➤ Example

//필터링 된 행 개수를 가져온다. Alert(mySheet.FilteredRowCount());

➤ 제공 버전

7.0.0.0

FindCheckedRow Method

➤ 기능

대상 컬럼의 체크된 행의 Index를 구분자 "|"로 연결한 문자열 또는 배열집합을 반환한다.

Opt 인자의 RowLevel, ParentRow, Recursive 인자는 트리구조의 시트에서 트리 기준 컬럼에 대한 체크박스 사용 모드에서만 유효한 인자 속성이다.

RowLevel, ParentRow 속성은 동시에 같이 사용할 수 없으며 두 속성 모두 설정이 있는 경우 ParentRow 속성을 우선 처리 한다.

2개행 이상의 단위데이터행 구조인 경우 Col 인자에 Index를 설정할 경우 모든행에 대해서 처리를 하고 SaveName으로 설정할 경우 해당 SaveName인 행에 대해서만 처리 한다.

➤ Syntax

Syntax	ObjId.FindCheckedRow(Col, [Opt])
--------	----------------------------------

➤ Info

Return	String Array, 체크된 행번호를 " " 연결한 문자열 또는 배열		
Parameter	Type	필수여부	설 명
Col	Number / String	필수	특정 컬럼의 Index 또는 SaveName
Opt.RowLevel	Number	선택	처리 대상 트리 레벨 (Default:0)
Opt.ParentRow	Index	선택	처리 대상 부모 행의 Index
Opt.Recursive	Boolean	선택	하위 노드에 대한 재귀 처리 여부 (Default:1)
Opt.ReturnArray	Boolean	선택	배열집합 반환 여부 (Default:0)

➤ Example



```
//체크된 행번호를 가져온다.  
//인자->1, 결과->1|3|4|5|6  
var sRow = mySheet.FindCheckedRow(1);  
var sRow = mySheet.FindCheckedRow("pass_yn");  
  
// 트리기준 컬럼의 체크박스에 대한 사용  
// 트리 레벨이 2인 행 가운데 체크된 목록 산출  
var res = mySheet.FindCheckedRow(1, {RowLevel:2, Recursive:0});  
  
// Index가 5인 행의 자식행 가운데 체크된 목록을 배열집합으로 산출  
var arRes = mySheet.FindCheckedRow(1, {  
    ParentRow:5,  
    Recursive:1,  
    ReturnArray:1  
});
```

➤ 제공 버전

7.0.0.0	
7.0.9.0	트리 구조의 트리체크에 대한 처리 지원 (RowLevel, ParaenRow 속성 인자 추가) 반환값 형태 설정 기능 추가 (ReturnArray)



FindFilterRow Method

➤ 기능

필터행의 인덱스를 반환한다.

➤ Syntax

Syntax	ObjId.FindFilterRow()
--------	-----------------------

➤ Info

Return	Long, 필터행의 인덱스 (Default=-1)		
Parameter	Type	필수여부	설 명
			없음

➤ Example

//필터행의 행 번호를 가져온다. var filterRow = mySheet.FindFilterRow();
--

➤ 제공 버전

7.0.13.32	
-----------	--



FindStatusRow Method

➤ 기능

트랜잭션 상태에 해당하는 행번호를 ";"로 조합하여 반환한다.

트랜잭션 상태는 RIUD의 문자열을 "|"로 연결하여 설정하면 해당하는 트랜잭션인 행의 번호를 모두 찾아서 "|"로 조합하여 반환한다.

➤ Syntax

Syntax	ObjId.FindStatusRow(sStatus)
--------	------------------------------

➤ Info

Return	String, 트랜잭션 상태에 해당하는 행번호를 ";"로 조합한 문자열 (Default="")		
Parameter	Type	필수여부	설 명
sStatus	String	필수	찾고자 하는 트랜잭션 상태 코드를 " "로 연결한 문자열

➤ Example

```
// 수정, 삭제인 행을 알아오기
//인자->U|D, 결과->1;3;4;5;6
var sRow = mySheet.FindStatusRow("U|D");

//받은 결과를 배열로 생성한다.
var arrow = sRow.split(";");
```

➤ 제공 버전

7.0.0.0

FindSubSumRow Method

➤ 기능

ShowSubSum 함수를 이용하여 표시된 소계의 행 번호를 "|"로 연결하여 문자열로 반환한다. StdCol 인자를 설정하지 않으면 모든 소계 행을 찾고, StdCol을 설정할 경우 해당 컬럼으로 계산된 소계 행만 찾아서 문자열로 반환한다.

➤ Syntax

Syntax	ObjId.FindSubSumRow([StdCol])
--------	-------------------------------

➤ Info

Return	String, 소계의 행 번호를 " "로 연결한 문자열		
Parameter	Type	필수여부	설 명
StdCol	Long/String	선택	소계를 표시한 기준 컬럼의 컬럼Index 또는 SaveName. Default=""(전체컬럼)

➤ Example

<pre>//표시된 모든 소계의 행 번호를 가져온다. var sRow = mySheet.FindSubSumRow(); //1컬럼으로 기준으로 표시된 소계 행의 행번호를 가져온다. var sRow = mySheet.FindSubSumRow(1);</pre>

➤ 제공 버전

7.0.0.0



FindSumRow Method

➤ 기능

합계행의 인덱스를 확인한다.

합계행이 2개 이상인 경우 (2개행 이상의 단위데이터행 구조인 경우) 합계행의 인덱스를 구분자 "|" 로 연결한 문자열을 반환 한다.

➤ Syntax

Syntax	ObjId.FindSumRow()
--------	--------------------

➤ Info

Return	String, 합계행의 인덱스 (Default=-1)		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
//합계행의 행 번호를 가져온다.  
var sumRow = mySheet.FindSumRow();
```

➤ 제공 버전

7.0.0.0

FindText Method

➤ 기능

컬럼내에 특정 텍스트를 찾아서 행번호를 반환한다.

기본적으로 데이터 행 처음부터 끝까지 찾고자 하는 텍스트와 대소문자를 포함한 전체가 같은 것을 찾을 수 있고, 인자의 설정에 따라 앞문자가 같은 데이터 행을 찾을 수 있다.

동일한 문자열을 컬럼에서 찾을 수 없으면 -1을 반환한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 모든행에 대해서 처리를 하고 SaveName으로 설정할 경우 해당 SaveName이 포함된 행에 대해서만 처리 한다.

➤ Syntax

Syntax	ObjId. FindText (Col,SearchText,[StartRow],[FullMatch],[CaseSensitive])
--------	--

➤ Info

Return	Long, 찾아진 행번호 Default=-1		
Parameter	Type	필수여부	설 명
Col	Number / String	필수	찾고자 하는 컬럼의 Index 또는 SaveName
SearchText	String	필수	찾을 문자열
StartRow	Number	선택	시작 행의 Index, Default="첫행"
FullMatch	Number	선택	글자 동일 종류, Default= -1
CaseSensitive	Boolean	선택	대소문자 구분 여부, Default=1

FullMatch 인자는 글자를 찾을 때 옵션으로 다음과 같이 처리된다.

FullMatch 값	기능
-1	SearchText와 전체가 같은 행을 찾는다.
0	SearchText와 앞 부분이 같은 행을 찾는다.



1	SearchText와 뒤 부분이 같은 행을 찾는다.
2	SearchText와 가운데 부분이 같은 행을 찾는다.

➤ **Example**

```
// 2컬럼내에 '한국'으로 시작하는 데이터의 행 위치 찾기
var Row1 = mySheet.FindText(2, "한국", 0, 0, 0);

//전체 글자가 "한국"인 데이터의 행 위치 찾기
var Row1 = mySheet.FindText(2, "한국", 0);

//뒤에 글자가 "은행"인 데이터의 행 위치 찾기
var Row1 = mySheet.FindText(2, "은행", 0, 1);

//글자 중에 "은"이라는 글자가 들어가는 행 위치 찾기
var Row1 = mySheet.FindText(2, "은", 0, 2);

//대소문자 구분하지 않고 찾기
var Row1 = mySheet.FindText(2, "Bank", 0, 2, 0);
```

➤ **제공 버전**

7.0.0.0

FitColWidth Method

➤ 기능

전체 넓이를 기준으로 모든 컬럼의 Width를 재조정하여 수평 스크롤이 생기지 않도록 한다.

시트 너비가 0 이거나 Visible 컬럼이 없는 경우 별도 처리없이 -1을 반환한다. Width 인자에 값을 설정하는 경우 전체 넓이 100%를 기준으로 설정한 값을 적용한다. 인자에 값은 각 컬럼의 너비를 "|"로 묶어서 설정하며 인자 값에 설정한 개수만큼 컬럼의 Width를 %단위로 재조정한다.

Width 인자에 설정한 값의 숫자가 컬럼수와 맞지 않거나 총합이 100%보다 큰 경우 수평 스크롤이 생길 수 있다.

Width 인자가 ""이거나 인자없이 호출하는 경우 컬럼의 FitColWidth 속성이 1(true)로 설정한 컬럼들의 너비 비율을 유지하면서 Object 전체 너비에 맞게 컬럼 너비를 재조정한다.

➤ Syntax

Syntax	ObjId.FitColWidth([Width])
--------	----------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Width	String	선택	각 컬럼의 %단위 너비의 조합, Default=""

➤ Example

<pre>// 컬럼들의 너비 비율을 유지하며 전체적으로 너비 재조정하기 mySheet.FitColWidth(); // % 단위로 재설정하기 mySheet.FitColWidth("10 20 40 30");</pre>
--



- 제공 버전
7.0.0.0

FitSize Method

➤ 기능

모든 행의 높이와 모든 컬럼의 너비를 재조정한다.

RowHeight 인자가 1인 경우 모든 행의 높이를 데이터의 높이에 맞게 재조정하고, ColumnWidth 인자가 1인 경우 모든 컬럼의 너비를 컬럼 내의 가장 넓은 글자의 너비에 맞게 재조정한다.

➤ Syntax

Syntax	ObjId. FitSize (RowHeight, ColumnWidth)
--------	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
RowHeight	Boolean	필수	행 높이 변경 여부
ColumnWidth	Boolean	필수	컬럼 너비 변경 여부

➤ Example

<pre>//행 높이만 재조정하기 mySheet.FitSize(1, 0); //컬럼 너비만 재조정하기 mySheet.FitSize(0, 1); //모두 재조정하기 mySheet.FitSize(1, 1);</pre>

➤ 제공 버전

7.0.0.0



FitSizeCol Method

➤ 기능

대상 컬럼의 너비를 컬럼 내의 가장 넓은 글자의 너비에 맞게 재조정한다.
재조정의 범위는 전역속성 FitSizeColMode 설정을 따른다.

➤ Syntax

Syntax	ObjId.FitSizeCol(Col)
--------	-----------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Col	Number/ String	필수	대상 컬럼의 Index 또는 SaveName

➤ Example

```
//SaveName 이 "sTarget"인 컬럼에 대한 처리  
mySheet.FitSizeCol("sTarget");
```

➤ 제공 버전

7.0.4.0	
---------	--

FocusAfterProcess Method

➤ 기능

조회 완료 후 포커스를 데이터 행으로 설정할지 여부를 확인하거나 설정한다.
기본적으로 DoSearch 함수를 이용하여 데이터 조회를 완료하면 포커스를 데이터 행의 첫행에 두도록 처리한다. 그러나 이 속성을 false로 설정하면 조회 완료하여도 기존 다른 컨트롤에 있는 포커스를 뺏어오지 않는다.

➤ Syntax

Syntax	Get	ObjId. GetFocusAfterProcess()
--------	-----	--------------------------------------

➤ Info

Return	Boolean, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//조회 후 포커스 여부
mySheet.GetFocusAfterProcess();
```

➤ Syntax

Syntax	Set	ObjId. SetFocusAfterProcess(mode)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
mode	Boolean	필수	포커스 설정 여부 (Default=1)

➤ Example

```
//조회 후 포커스를 두지 않음
mySheet.SetFocusAfterProcess(0);
```



- 제공 버전
7.0.0.0

FocusAfterRowTransaction Method

➤ 기능

행 추가, 삭제, 이동, 복사 후 포커스를 이동 할지 여부를 확인 하거나 설정 한다.

2개 이상의 행에 대한 연속 처리시 불필요한 포커스 처리를 막을 수 있어서 처리 속도를 개선 할 수 있다.

➤ Syntax

Syntax	Get	ObjId. GetFocusAfterRowTransaction()
--------	-----	---

➤ Info

Return	Boolean, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// 포커스를 이동 할지 여부를 확인
mySheet.GetFocusAfterRowTransaction();
```

➤ Syntax

Syntax	Set	ObjId. SetFocusAfterRowTransaction(mode)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
mode	Boolean	필수	포커스 설정 여부 (Default=1)

➤ Example

```
// 현재 포커스행 하위로 10개 행을 추가후 맨 마지막 추가된 행에 포커스
설정
```



```
mySheet.SetFocusAfterRowTransaction(0);  
var new Row = null;  
for (var i = 0; i < 10; i++) {  
    newRow = mySheet.DataInsert();  
}  
mySheet.SetSelectRow(newRow);
```

➤ **제공 버전**

7.0.0.0

FocusEditMode Method

➤ 기능

셀에 포커스가 들어갔을 때 Edit 가능한 셀은 기본적으로 단순 포커스 상태로 둘것인지 Edit 상태로 둘것인지 여부를 설정하거나 확인한다.

이 속성은 Edit 가능한 셀에 대한 Edit 모드를 설정하거나 확인한다.

설정값	내용
0	포커스가 들어가면 그냥 포커스 상태 (기본값)
1	포커스가 들어가면 바로 편집 상태
2	컬럼타입이 Combo, ComboEdit만 포커스 상태, 나머지는 모두 편집 상태

➤ Syntax

Syntax	Get	ObjId. GetFocusEditMode()
--------	-----	----------------------------------

➤ Info

Return	Integer, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//포커스가 갔을 때 Edit 상태로 표시 할지 여부 확인
mySheet.GetFocusEditMode();
```

➤ Syntax

Syntax	Set	ObjId. SetFocusEditMode (Mode)
--------	-----	---------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명



Mode	Integer	필수	설정 여부 (Default=0)
------	---------	----	-------------------

➤ **Example**

```
// 포커스가 갔을 때 Edit가 아닌 단순 포커스 상태 유지  
mySheet.SetFocusEditMode(0);  
  
//포커스가 갔을 때 Edit 상태로 표시  
mySheet.SetFocusEditMode(1);
```

➤ **제공 버전**

7.0.0.0

FrozenCol Method

➤ 기능

좌측 고정컬럼을 동적으로 설정 확인한다.

기존에 설정된 고정컬럼 수 보다 많은 수를 설정하는 경우 추가로 고정컬럼을 설정 처리하고, 적은 수를 설정한 경우 초과되는 컬럼수만큼 고정컬럼을 해제 처리한다.

메인 섹션의 컬럼을 모두 고정으로 설정하는 경우 및 시트에서 보여지는 영역 내에서 설정한 고정컬럼을 모두 표현할 수 없는 경우 시트가 정상적으로 동작하지 않을 수 있다. (주의)

➤ Syntax

Syntax	Get	ObjId. GetFrozenCol ()
--------	-----	-------------------------------

➤ Info

Return	Integer, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// 고정컬럼 설정 값을 확인한다.
Alert(mySheet.GetFrozenCol());
```

➤ Syntax

Syntax	Set	ObjId. SetFrozenCol (Count)
--------	-----	------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Rows	Integer	필수	설정할 고정컬럼 개수 (Default=0)



➤ **Example**

```
// 3개의 고정컬럼을 설정한다.  
mySheet.SetFrozenCol(3);
```

➤ **제공 버전**

7.0.0.0

FrozenRows Method

➤ 기능

고정행 출력 여부 및 출력 옵션을 설정하거나 확인한다.

상단에 설정한 개수의 행을 고정행으로 출력한다.

조회 전 호출 시 데이터 조회 완료 시점에 해당 개수만큼 조회된 데이터를 고정행으로 출력한다.

고정행에 합계행, 필터행은 포함되지 않으며 고정행 생성 시 합계행, 필터행이 있는 경우 합계행, 필터행의 아래에 생성된다.

고정행은 마우스를 이용한 행 선택 및 마우스 드래그를 통한 행 이동을 지원하지 않는다.

서버페이지 조회 방식, 단위데이터행, 데이터머지, 트리 시트 및 소계를 사용하는 경우 해당 기능을 지원하지 않는다.

메인 섹션의 행을 모두 고정행으로 설정하는 경우 및 시트에서 보여지는 영역 내에서 설정한 고정행을 모두 표현할 수 없는 경우 시트가 정상적으로 동작하지 않을 수 있다. (주의)

➤ Syntax

Syntax	Get	ObjId. GetFrozenRows()
--------	-----	-------------------------------

➤ Info

Return	Integer, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// 고정행 설정 값을 확인한다.
Alert(mySheet.GetFrozenRows());
```

➤ Syntax



Syntax	Set	ObjId. SetFrozenRows (Rows)
--------	-----	------------------------------------

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Rows	Integer	필수	설정할 고정행 개수 (Default=0)

➤ **Example**

```
// 3개의 고정행을 설정한다.  
mySheet.SetFrozenRows(3);
```

➤ **제공 버전**

7.0.0.0

GetCellProperty Method

➤ 기능

InitColumns 또는 InitCellProperty 에서 설정한 속성의 정보를 확인한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

➤ Syntax

Syntax	ObjId. GetCellProperty (Row, Col, PropName)
--------	--

➤ Info

Return	String/ Boolean/ Integer, 설정한 컬럼의 속성값		
Parameter	Type	필수여부	설 명
Row	Long	필수	특정 셀의 Row Index
Col	Long / String	필수	특정 셀의 컬럼 Index 또는 SaveName
PropName	String	필수	확인하고자 하는 속성명

➤ Example

<pre>//데이터 타입을 읽어옴 var iType = mySheet.GetCellProperty(1, 1, "Type"); //데이터 SaveName을 읽어옴 var sSaveName = mySheet.GetCellProperty(1, 1, "SaveName");</pre>

➤ 제공 버전

7.0.0.0



GetChildNodeCount Method

➤ 기능

트리구조에서 특정 행의 하위(다음레벨) 노드의 개수를 확인한다.

➤ Syntax

Syntax	ObjId. GetChildNodeCount (Row)
--------	---------------------------------------

➤ Info

Return	Integer, 하위 노드의 개수		
Parameter	Type	필수여부	설 명
Row	Long	필수	특정 행의 Row Index

➤ Example

```
//1번째행의 하위노드 개수  
var childCount = mySheet.GetChildNodeCount(1);
```

➤ 제공 버전

7.0.0.0

GetChildRows Method

➤ 기능

트리구조에서 특정 행의 자식행들의 Index를 "|"로 조합하여 반환한다

MaxLevel 인자를 설정하는 경우 해당 레벨까지의 자식 행들까지만 리턴되며 설정하지 않는 경우 모든 자식행들이 리턴된다.

➤ Syntax

Syntax	ObjId. GetChildRows (Row, [MaxLevel])
--------	--

➤ Info

Return	String, 대상 자식행들의 문자열 조합		
Parameter	Type	필수여부	설 명
Row	Long	필수	대상 행의 Index
MaxLevel	Integer	선택	확인할 자식행의 제한 레벨 (-1인 경우 모든 자식행), Default=-1

➤ Example

<pre>// 2번째 행의 모든 자식행 확인 var childRows = mySheet.GetChildRows(2); // 2번째 행의 자식레벨중 3레벨까지만 확인 var childRows = mySheet.GetChildRows(2, 3); // 2번째 행의 레벨기준으로 하위 +2레벨까지 확인 var myLevel = mySheet.GetRowLevel(2); var childRows = mySheet.GetChildRows(2, myLevel+2);</pre>

➤ 제공 버전

7.0.0.0



GetColSortInfo Method

➤ 기능

현재 적용되어 있는 컬럼별 Sort 정보를 확인 한다.

반환되는 값은 Sort 적용 순서로 아래와 같은 형식의 객체에 대한 배열 집합이다.

Property	Type	설 명
Col	Number/String	컬럼의 Index (단위데이터행 구조인 경우 SaveName)
SortOrder	String	해당 컬럼의 정렬 방식 ("asc","desc")

➤ Syntax

Syntax	ObjId.GetColSortInfo()
--------	------------------------

➤ Info

Return	Object[], 컬럼별 Sort 정보 객체의 배열 집합		
Parameter	Type	필수여부	설 명
			없음

➤ Example

```
// 현재의 Sort 정보를 확인 한다.  
var sortInfo = mySheet.GetColSortInfo();
```

➤ 제공 버전

7.0.13.22	
-----------	--

GetComboInfo Method

➤ 기능

특정 셀의 콤보 정보를 확인한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

Flag 인자 값은 다음과 같다.

설정값	설명
"Text"	콤보텍스트
"Code"	콤보코드
"SelectedIndex"	선택된 콤보의Item Index

➤ Syntax

Syntax	ObjId. GetComboInfo (Row,Col,Flag)
--------	---

➤ Info

Return	String, 콤보 텍스트 또는 코드		
Parameter	Type	필수여부	설 명
Row	Long	필수	특정 셀의 Row Index
Col	Long / String	필수	특정 셀의 컬럼 Index 또는 SaveName
Flag	String	필수	"Text", "Code" 선택

➤ Example

```
//콤보코드와 텍스트를 가져온다.
var sText = mySheet.GetComboInfo(0,2, "Text");
var sCode = mySheet.GetComboInfo(0,2, « Code ») ;
```



```
//각각 배열로 구성한다.  
var arrText = sText.split("|");  
var arrCode = sCode.split("|");  
  
//2행의 2컬럼의 콤보 코드를 이용하여 콤보텍스트를 가져온다.  
For(i=0; i<arrCode.length; i++) {  
    if(mySheet.GetCellValue(2,2) == arrCode[i]) {  
        alert(arrText[i]);  
        break;  
    }  
}
```

- **제공 버전**
7.0.0.0



GetCurrentPage Method

➤ 기능

조회 방식이 smGeneral 이 아닌 경우 전체데이터를 기준으로 현재 화면 위에 올려진 부분이 몇번째 페이지인지를 리턴한다.

➤ Syntax

Syntax	ObjId.GetCurrentPage()
--------	------------------------

➤ Info

Return	Integer, 현재 페이지 번호
--------	--------------------

➤ Example

```
// 현재 페이지 넘버.  
var pageNum = mySheet.GetCurrentPage();
```

➤ 제공 버전

7.0.0.0



GetDataFirstRow Method

➤ 기능

데이터 행의 시작 인덱스를 확인한다.

헤더, 필터, 합계행은 인덱스에서 제외되며, 데이터행이 존재하지 않는 경우 -1을 반환한다.

➤ Syntax

Syntax	ObjId.GetDataFirstRow()
--------	-------------------------

➤ Info

Return	Number, 데이터행의 시작 인덱스 (Default=-1)		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// 데이터행의 마지막 인덱스를 확인한다.  
var startRow = mySheet.GetDataFirstRow();
```

➤ 제공 버전

7.0.13.37	
-----------	--



GetDataRows Method

➤ 기능

단위 데이터 행의 설정 개수를 확인 한다.

➤ Syntax

Syntax	ObjId. GetDataRows()
--------	-----------------------------

➤ Info

Return	Integer, 설정 되어 있는 단위 데이터 행의 개수		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// 단위데이터행의 개수를 확인한다.  
var dataRows = mySheet.GetDataRows();
```

➤ 제공 버전

7.0.0.0



GetDataLastRow Method

➤ 기능

데이터 행의 마지막 인덱스를 확인한다.

헤더, 필터, 합계행은 인덱스에서 제외되며, 데이터행이 존재하지 않는 경우 -1을 반환한다.

➤ Syntax

Syntax	ObjId.GetDataLastRow()
--------	------------------------

➤ Info

Return	Number, 데이터행의 마지막 인덱스 (Default=-1)		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// 데이터행의 마지막 인덱스를 확인한다.  
var lastRow = mySheet.GetDataLastRow();
```

➤ 제공 버전

7.0.13.37	
-----------	--



GetEditText Method

➤ 기능

Edit 중인 글자를 확인한다.

➤ Syntax

Syntax	ObjId. GetEditText()
--------	-----------------------------

➤ Info

Return	String, 편집중인 글자 (Default="")		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// 편집중인 글자를 확인한다.  
Function mySheet_OnKeyUp(Row, Col, KeyCode, Shift){  
    var editTxt= "Edit중인 글자 = " + mySheet.GetEditText();  
    editTxt += "실제글자 = " + mySheet.GetCellValue(Row,Col);  
    alert(editTxt);  
}
```

➤ 제공 버전

7.0.0.0

GetFilterParam Method

➤ 기능

SearchMode:3 은, 디비에서 부분 데이터를 가져와 조회하는 것이므로 필터링 기능 사용이 불가능하다. 이 경우 필터링 행의 필터값과, 필터링 할려는 옵션 값을 서버에 넘겨서 디비에서 조회시 사용하여, 필터링한 것과 같은 데이터를 구하여 시트에서 조회하기 위한 QueryString 문자열을 구해오는 함수이다. **AllFilter**가 0인 경우, 전체 컬럼을, 1인 경우 필터링이 되어 있는 컬럼들만 대상으로 한다. 단 필터 셀에 값이 들어있어도, 옵션값이 0(사용 안함)인 경우는 무시한다.

QueryString은 한 컬럼당 SaveName=CellValue&SaveName_opt=OptionValue 형식으로 구성되며 컬럼의 연결자 역시 "&"를 사용한다.

➤ Syntax

Syntax	ObjId. GetFilterParam ([AllFilter], [UrlEncode])
--------	---

➤ Info

Return	String, 조회 조건 Query String		
Parameter	Type	필수여부	설 명
AllFilter	Boolean	선택	전체 필터링 여부, Default =0
UrlEncode	Boolean	선택	UrlEncode 여부, Default= 1

➤ Option

0	사용 안 함	1	같다
2	같지 않다	3	작은
4	작거나 같은	5	큰
6	크거나 같은	7	단어로 시작함
8	단어로 시작하지 않음	9	단어로 끝남
10	단어로 끝나지 않음	11	포함함
12	포함하지 않음		



➤ **Example**

```
//필터행 전체 컬럼을 Param 형식의 스트링으로 구해오기.  
var FilterStr = mySheet.GetFilterParam(1);  
  
//필터링 된 컬럼들만 Param 형식의 스트링으로 구해오기.  
var FilterStr = mySheet.GetFilterParam(0);
```

➤ **제공 버전**

7.0.0.0



GetFirstChildRow Method

➤ 기능

트리구조에서 특정 행의 첫번째 자식 행의 Index를 확인한다.
자식행이 없는 경우는 -1를 반환한다.

➤ Syntax

Syntax	ObjId. GetFirstChildRow (Row)
--------	--------------------------------------

➤ Info

Return	Long, 첫번째 자식 행의 Index		
Parameter	Type	필수여부	설 명
Row	Long	필수	대상 행의 Index

➤ Example

```
// 2번째 행의 첫번째 자식 행을 확인한다.  
var firstChild = mySheet.GetFirstChildRow(2);
```

➤ 제공 버전

7.0.0.0



GetGroupCol Method

➤ 기능

현재 설정되어있는 그룹 기준 컬럼을 SaveName확인한다.

2개 이상의 컬럼이 설정되어 있는 경우는 "|"를 구분자로 하여 연결된 문자열로 리턴 된다.

➤ Syntax

Syntax	ObjId.GetGroupCol()
--------	---------------------

➤ Info

Return	String, 현재 설정된 그룹 기준 컬럼 정보		
Parameter	Type	필수여부	설 명
없음			

➤ Example

// 그룹행을 설정한다. mySheet.ShowGroupRow(); // 현재 설정되어 있는 그룹 기준 컬럼을 확인한다 var Cols = mySheet.GetGroupCol();

➤ 제공 버전

7.0.0.0



GetGroupRow Method

➤ 기능

대상 컬럼을 기준으로 생성된 그룹행을 확인 한다.

대상 그룹행의 Index를 구분자 "|"로 연결된 문자열로 반환 한다.

➤ Syntax

Syntax	ObjId. GetGroupRow (Col)
--------	---------------------------------

➤ Info

Return	String, 현재 설정된 그룹 기준 컬럼 정보		
Parameter	Type	필수여부	설 명
Col	Number/ String	필수	대상 컬럼의 Index 또는 SaveName

➤ Example

```
// SaveName이 "sDept"인 컬럼을 기준으로 생성된 그룹행 확인  
console.log("GroupRow :", mySheet.GetGroupRow("sDept"));
```

➤ 제공 버전

7.0.13.22	
-----------	--



GetLastChildRow Method

➤ 기능

트리구조에서 특정 행의 마지막 자식 행의 Index를 확인한다.

자식행이 없는 경우는 -1를 반환한다.

➤ Syntax

Syntax	ObjId. GetLastChildRow (Row)
--------	-------------------------------------

➤ Info

Return	Long, 마지막 자식 행의 Index		
Parameter	Type	필수여부	설 명
Row	Long	필수	대상 행의 Index

➤ Example

// 2번째 행의 마지막 자식 행을 확인한다. var lastChild = mySheet.GetLastChildRow(2);
--

➤ 제공 버전

7.0.0.0



GetMergedEndCell Method

➤ 기능

전체 머지된 셀중 머지 마지막 셀의 Row와 Col 정보를 Row, Col 형태의 String으로 리턴한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

➤ Syntax

Syntax	ObjId. GetMergedEndCell (Row, Col)
--------	---

➤ Info

Return	String, Row와 Col 정보를 "Row, Col"형태의 String으로 리턴		
Parameter	Type	필수여부	설 명
Row	Long	필수	대상 행의 Index
Col	Long / String	필수	대상이 되는 컬럼 또는 SaveName

➤ Example

<pre>var endMergeCell = mySheet.GetMergedEndCell(4,5);</pre>
--

➤ 제공 버전

7.0.0.0

GetMergedStartCell Method

➤ 기능

전체 머지된 셀중 머지 시작 셀의 Row와 Col 정보를 Row, Col 형태의 String으로 리턴한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

➤ Syntax

Syntax	ObjId.GetMergedStartCell(Row, Col)
--------	------------------------------------

➤ Info

Return	String, Row와 Col 정보를 "Row, Col"형태의 String으로 리턴		
Parameter	Type	필수여부	설 명
Row	Long	필수	대상 행의 Index
Col	Long / String	필수	대상이 되는 컬럼 또는 SaveName

➤ Example

```
var startMergeCell = mySheet.GetMergedStartCell(4,5);
```

➤ 제공 버전

7.0.0.0



GetNextSiblingRow Method

➤ 기능

트리구조에서 특정 행에 대한 동일 부모, 동일 레벨상의 다음 행의 Index를 확인 한다.

해당 다음 행이 없는 경우는 -1로 반환 한다..

➤ Syntax

Syntax	ObjId. GetNextSiblingRow (Row)
--------	---------------------------------------

➤ Info

Return	Long, 동일 레벨상 다음 행의 Index		
Parameter	Type	필수여부	설 명
Row	Long	필수	대상 행의 Index

➤ Example

```
// 6번째 행의 동일레벨 상의 다음 행을 확인한다.  
var next = mySheet.GetNextSiblingRow(6);
```

➤ 제공 버전

7.0.0.0



GetParentRow Method

➤ 기능

트리구조에서 특정 행의 부모 행의 Index를 확인한다.

부모 행이 없는 경우 -1로 반환한다.

➤ Syntax

Syntax	ObjId. GetParentRow (Row)
--------	----------------------------------

➤ Info

Return	Long, 부모 행의 Index		
Parameter	Type	필수여부	설 명
Row	Long	필수	대상 행의 Index

➤ Example

```
// 5번째 행의 부모행을 확인한다.  
var parent = mySheet.GetParentRow(5);
```

➤ 제공 버전

7.0.0.0



GetPrevSiblingRow Method

➤ 기능

트리구조에서 특정 행에 대한 동일 부모, 동일 레벨상의 이전 행의 Index를 확인 한다.

해당 이전 행이 없는 경우는 -1로 반환 한다..

➤ Syntax

Syntax	ObjId. GetPrevSiblingRow (Row)
--------	---------------------------------------

➤ Info

Return	Long, 동일 레벨상 이전 행의 Index		
Parameter	Type	필수여부	설 명
Row	Long	필수	대상 행의 Index

➤ Example

```
// 6번째 행의 동일레벨 상의 이전 행을 확인한다.  
var previous = mySheet.GetPrevSiblingRow(6);
```

➤ 제공 버전

7.0.0.0

GetSaveData Method

➤ 기능

저장 페이지를 호출하여 저장 처리를 완료한 후 생성된 결과를 문자열로 반환한다.

이 함수는 DoAllSave 또는 DoSave 함수와 달리 저장할 QueryString을 인자로 받아서 저장처리를 하며, 저장결과에 대한 처리 없이 데이터 자체로 반환한다. 이 함수를 통해서 반환된 저장 데이터는 LoadSaveData 함수의 인자로 사용하면 저장 결과를 IBSheet 내부에 표현할 수 있다.

SaveString 에 대한 인코딩이 필요할 경우 함수 호출전에 처리를 해야 한다.

➤ Syntax

Syntax	ObjId. GetSaveData (PageUrl, [SaveString], [Param], [Opt])
--------	---

➤ Info

Return	String, 저장 데이터의 문자열		
Parameter	Type	필수여부	설 명
PageUrl	String	필수	저장 처리할 페이지 URL
SaveString	String	선택	저장할 Query String, Default=""
Param	String	선택	저장 조건 Query String, Default=""
Opt.RegHeader	Object	선택	요청헤더 설정 값 객체, Default={}

➤ Example

```
//저장 String 가져오기 - 트랜잭션이 발생한 것만 저장할 경우
var SaveStr = mySheet.GetSaveString();

//저장할 내역이 없거나 저장 validation에서 오류가 발생한 경우
if (SaveStr == "" || SaveStr == "KeyFieldError") return;

//저장 처리 결과 읽어오기
```



```
var rtnData = mySheet.GetSaveData("save.jsp", SaveStr);  
//돌아온 결과를 시트에 반영  
mySheet.LoadSaveData(rtnData);
```

- **제공 버전**
7.0.0.0

GetSaveJson Method

➤ 기능

저장 대상의 데이터를 Json 객체로 반환한다.

이 함수를 호출하면 저장 객체를 생성하며 필수 입력 항목을 확인하고, OnValidation 이벤트가 발생하여 사용자 Validation 로직도 처리한다.

Col 인자는 AllSave인자가 0일 때 사용하며 저장 대상이 되는 컬럼을 설정한다. 2개행 이상의 단위데이터행 구조인 경우 Col 인자에 Index를 설정할 경우 Json 객체 생성시 각각의 행 단위로 생성되고, SaveName으로 설정할 경우 단위데이터행 단위로 생성 한다.

Validation 체크 결과가 실패인 경우 결과 코드와 메시지가 반환된다.

- 처리대상 행이 없는 경우

Code : "IBS000", Message : "NoTargetRows"

- 필수입력 누락인 경우

Code : "IBS010", Message : "KeyFieldError"

- Validation 오류인 경우

Code : "IBS020", Message : "InvalidInputError"

➤ Syntax

Syntax	ObjId. GetSaveJson ([Option])
--------	--------------------------------------

➤ Info

Return	Object, 저장할 데이터의 Json 객체		
Parameter	Type	필수여부	설 명
Option.AllSave	Boolean	선택	전체 저장 여부, (Default =0)
Option.StdCol	Number / String	선택	대상이 되는 기준 컬럼의 Index 또는 SaveName, (Default =상태 컬럼)
Opt.StdColValue	String	선택	기준컬럼의 추출 대상 값을 구분 자 ' ' 로 연결한 문자열



Opt.ValidKeyField	Boolean	선택	KeyField 체크 여부 (Default=1)
Opt.ValidFullInput	Boolean	선택	FullInput 체크 여부 (Default=1)

➤ **Example**

```
//모든 행의 데이터를 객체로 받기  
var SaveJson = mySheet.GetSaveJson({AllSave:1});  
  
//저장 대상의 데이터를 객체로 받기 (트랜잭션이 발생한 행)  
var SaveJson = mySheet.GetSaveJson();  
  
// 4번째 컬럼이 Check된 행의 데이터를 객체로 받기  
var SaveJson = mySheet.GetSaveJson({StdCol:4});
```

➤ **제공 버전**

7.0.0.0	
7.0.13.27	StdColValue 속성 추가

GetSaveString Method

➤ 기능

저장 시 사용되는 데이터의 Query String을 문자열로 반환한다.

이 함수는 보안 모듈을 사용하여 저장하는 경우 저장할 문자열을 반환하여 보안모듈에서 암호화 하여 저장 처리할 수 있도록 한다.

이 함수를 호출하면 저장 String을 생성하며 필수 입력 항목을 확인하고, OnValidation 이벤트가 발생하여 사용자 Validation 로직도 처리한다.

AllSave인자는 1일때 DoAllSave 함수를 호출할 때 사용되는 저장String과 동일하며, 0 일때 DoSave함수를 호출할 때 사용되는 저장String과 동일하다.

UrlEncode는 1일 때 Query String의 한글 문자를 모두 Encoding한다.

Col 인자는 AllSave인자가 0일 때 사용하며 저장 대상이 되는 컬럼을 설정한다.

Mode인자는 Query String 문자열 조합방법을 설정한다.

설정값에 따른 Query String 결과는 다음과 같다.

Validation 체크 결과가 실패인 경우 기본으로 **KeyFieldError** 문자열이 반환되고, ibmsg의 SYS_InvalidGetSaveString에 값이 설정되어 있는 경우 해당 값으로 반환된다.

Mode	설명	예
1	셀기준 조합 (SaveName 기준 Array 형식)	sSeq=1&sStatus=R..
2	컬럼기준 조합 (SaveName 기준 구분자 조합 형식)	sSeq=1 2&sStatus=R U..

"선택" 인자는 json 형식으로 속성 설정이 가능하다.(Example 참고)



➤ **Syntax**

Syntax	ObjId. GetSaveString ([Opt])
--------	-------------------------------------

➤ **Info**

Return	String, 저장할 Query String		
Parameter	Type	필수여부	설 명
Opt.AllSave	Boolean	선택	전체 저장 여부 (Default=0)
Opt.UrlEncode	Boolean	선택	UrlEncode 여부 (Default=1)
Opt.Col	Long / String	선택	대상이 되는 기준 컬럼 또는 SaveName (Default = 상태 컬럼)
Opt.Prefix	String	선택	저장시 SaveName 앞에 붙이고자 하는 문자열 (Default="")
Opt.Mode	Integer	선택	Query String 문자열 조합방법을 설정 Mode=1, Mode=2 (Default = 1)
Opt.Delim	String	선택	Mode=2일때, 연결될 구분자 설정 (Default = " ")
Opt.StdColValue	String	선택	기준컬럼의 추출 대상 값을 구분자 ' ' 로 연결한 문자열
Opt.ValidKeyField	Boolean	선택	KeyField 체크 여부 (Default=1)
Opt.ValidFullInput	Boolean	선택	FullInput 체크 여부 (Default=1)

➤ **Example**

```
//DoAllSave 함수 사용과 동일한 저장 String 가져오기
var SaveStr = mySheet.GetSaveString({"AllSave":1});

//DoSave 함수 사용과 동일한 저장 String 가져오기
// - 트랜잭션이 발생한 것만 저장할 경우
var SaveStr = mySheet.GetSaveString();

//DoSave 함수 사용과 동일한 저장 String 가져오기
// - 체크박스 컬럼타입의 Check된 것만 저장하는 경우
```



```
// (해당 컬럼의 Index가 4일 때)
var SaveStr = mySheet.GetSaveString({"Col":4});

//DoSave 함수 사용과 동일한 저장 String 가져오기
// 돌아오는 값은 pre_AA=1&pre_BB=2&pre_CC=3 형태가 된다.
var SaveStr = mySheet.GetSaveString({"Prefix":"pre_"});
```

➤ see also

[GetSaveJson](#) Method

➤ 제공 버전

7.0.0.0	
7.0.12.4	인자구조 변경, ValidKeyField, ValidFullInput 인자 속성 추가 (이전 방식의 인자구조는 이전 가이드 참고)
7.0.13.27	StdColValue 속성 추가



GetSearchData Method

➤ 기능

조회 페이지를 호출하여 조회 처리를 완료한 후 생성된 조회결과 데이터를 문자열로 반환한다. 이 함수는 DoSearch 함수와 달리 조회결과에 대한 처리 없이 조회 데이터 자체를 반환한다. 이 함수를 통해서 반환된 조회 데이터는 LoadSearchData 함수의 인자로 사용하면 조회 결과를 IBSheet 내부에 표현할 수 있다.

➤ Syntax

Syntax	ObjId. GetSearchData (PageUrl, [Param], [Opt])
--------	---

➤ Info

Return	String, 조회 데이터의 문자열		
Parameter	Type	필수여부	설 명
PageUrl	String	필수	조회 처리할 페이지 URL
Param	String	선택	조회 조건 Query String, Default=""
Opt ReqHeader	Object	선택	요청헤더 설정 값 객체, Default={}

➤ Example

```
//조회 데이터 읽어오기
var sXml = mySheet.GetSearchData("list.jsp");

//조회 결과 내용을 표현하기
mySheet.LoadSearchData(sXml);
```

➤ 제공 버전

7.0.0.0



GetSelectionCols Method

➤ 기능

사용자가 드레그한 영역의 열을 구분자로 구분하여 반환한다.
구분자를 설정하지 않으면 기본적으로 "|"로 묶어서 반환한다.

➤ Syntax

Syntax	ObjId. GetSelectionCols ([DeliChar])
--------	---

➤ Info

Return	String, 선택되어있는 행번 조합 문자열		
Parameter	Type	필수여부	설 명
DeliChar	String	선택	조합구분자, Default=" "

➤ Example

```
// 구분자로 연결하여 선택된 열번 가져오기
var sColStr = mySheet.GetSelectionCols("/");

//자바 스크립트 배열로 만들기
var arr = sColStr.split("/");
for (i=0; i<arr.length(); i++) {
    alert(arr[i] + " 컬럼이 선택되었음");
}
```

➤ 제공 버전

7.0.0.0



GetSelectionRows Method

➤ 기능

사용자가 드래그를 통해 선택한 영역의 행을 구분자로 구분하여 반환한다.
구분자를 설정하지 않으면 기본적으로 "|"로 묶어서 반환한다.

➤ Syntax

Syntax	ObjId. GetSelectionRows ([DeliChar])
--------	---

➤ Info

Return	String, 선택되어있는 행번 조합 문자열		
Parameter	Type	필수여부	설 명
DeliChar	String	선택	조합구분자, Default=" "

➤ Example

```
//"/" 구분자로 연결하여 선택된 행번 가져오기  
var sRowStr = mySheet.GetSelectionRows("/");  
  
//자바 스크립트 배열로 만들기  
var arr = sRowStr.split("/");  
for (i=0; i<arr.length(); i++) {  
    alert(arr[i] + " 행이 선택되었음");  
}
```

➤ 제공 버전

7.0.0.0



GetSheetHtml Method

➤ 기능

현재 시트의 Html 소스를 반환한다.

시트에 적용된 스타일을 문자열로 가진 style과 html 소스를 문자열로 가진 body를 객체로 반환한다.

* 현재 화면에 보여지는 시트 상태 그대로의 html 코드를 리턴하므로 히든 설정 및 컬럼 사이즈 등이 동일하게 출력되며, 화면에 로드 되지 않은 자료는 출력되지 않습니다.

➤ Syntax

Syntax	ObjId.GetSheetHtml()
--------	----------------------

➤ Info

Return	Object, (style, body 로 이루어진 객체)		
Parameter	Type	필수여부	설 명
			없음

➤ Example

```
// 시트 Html 가져오기
var code = mySheet.GetSheetHtml();
document.getElementById("styleText").value = code.style;
document.getElementById("htmlText").value += code.body;
```

➤ 제공 버전

7.0.0.0



GetSubSumInfo Method

➤ 기능

현재 시트의 소계 설정 정보를 확인 한다.

SaveName 인자를 설정하지 않거나, 0으로 설정한 경우 컬럼 정보는 Index로 반환 하고, SaveName 인자를 1로 설정한 경우 SaveName으로 반환 한다.

반환되는 값은 객체 배열 형태로 [ShowSubSum](#) 메소드의 인자 형태와 동일 하다.

➤ Syntax

Syntax	ObjId.GetSubSumInfo(SaveName)
--------	-------------------------------

➤ Info

Return	Object		
Parameter	Type	필수여부	설 명
SaveName	Boolean	필수	컬럼 정보를 SaveName으로 반환 여부

➤ Example

```
// 소계 정보 확인 하기 (SaveName)
var subsumInfo = mySheet.GetSubSumInfo(1);
```

➤ 제공 버전

7.0.13.34	
-----------	--

GoToBaseSheet Method

➤ 기능

피벗 설정 팝업으로부터 피벗 테이블이 화면에 표시되어 있는 경우 이 기능을 사용하면 해당 피벗 테이블의 원본 시트로 이동 한다.

피벗 테이블 설정 팝업의 [원본시트이동] 버튼의 기능과 동일하게 동작 한다.

➤ Syntax

Syntax	ObjId.GoToBaseSheet()
--------	-----------------------

➤ Info

Return	None.		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// 피벗 테이블의 원본 시트로 이동
mySheet.GoToBaseSheet();
```

➤ 제공 버전

7.0.0.0



GoToFirstPage Method

➤ 기능

페이징 모드로 설정되어 있는 경우 맨 처음 페이지로 이동한다.

이 기능을 사용하기 위해서는 SetConfig 함수에서 SearchMode를 페이징 모드로 설정 하여야 한다.

➤ Syntax

Syntax	ObjId.GoToFirstPage()
--------	-----------------------

➤ Info

Return	None.		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// 페이징 모드 설정
var cfg = {SearchMode:1, Page:10};
mySheet.SetConfig(cfg);

// 맨 처음 페이지로 이동
mySheet.GoToFirstPage();
```

➤ 제공 버전

7.0.0.0



GoToLastPage Method

➤ 기능

페이징 모드로 설정되어 있는 경우 맨 마지막 페이지로 이동한다.

이 기능을 사용하기 위해서는 SetConfig 함수에서 SearchMode를 페이징 모드로 설정 하여야 한다.

➤ Syntax

Syntax	ObjId.GoToLastPage()
--------	----------------------

➤ Info

Return	Boolean		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// 페이징 모드 설정
var cfg = {SearchMode:1, Page:10};
mySheet.SetConfig(cfg);

// 맨 마지막 페이지로 이동
mySheet.GoToLastPage();
```

➤ 제공 버전

7.0.0.0



GoToNextPage Method

➤ 기능

페이징 모드로 설정되어 있는 경우 다음 페이지로 이동한다.

이 기능을 사용하기 위해서는 SetConfig 함수에서 SearchMode를 페이징 모드로 설정 하여야 한다.

➤ Syntax

Syntax	ObjId.GoToNextPage()
--------	----------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// 페이징 모드 설정
var cfg = {SearchMode:1, Page:10};
mySheet.SetConfig(cfg);

// 다음 페이지로 이동
mySheet.GoToNextPage();
```

➤ 제공 버전

7.0.0.0



GoToPageNum Method

➤ 기능

페이징 모드로 설정되어 있는 경우 인자로 입력한 페이지로 이동한다.

이 기능을 사용하기 위해서는 SetConfig 함수에서 SearchMode를 페이징 모드로 설정 하여야 한다.

➤ Syntax

Syntax	ObjId.GoToPageNum(index)
--------	--------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
index	Integer	필수	이동하고자 하는 Page의 Index

➤ Example

```
// 페이징 모드 설정
var cfg = {SearchMode:1, Page:10};
mySheet.SetConfig(cfg);

// 5번째 페이지로 이동
mySheet.GoToPageNum(5);
```

➤ 제공 버전

7.0.0.0



GoToPrevPage Method

➤ 기능

페이징 모드로 설정되어 있는 경우 이전 페이지로 이동한다.

이 기능을 사용하기 위해서는 SetConfig 함수에서 SearchMode를 페이징 모드로 설정 하여야 한다.

➤ Syntax

Syntax	ObjId.GoToPrevPage()
--------	----------------------

➤ Info

Return	None.		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// 페이징 모드 설정
var cfg = {SearchMode:1, Page:10};
mySheet.SetConfig(cfg);

// 이전 페이지로 이동
mySheet.GoToPrevPage();
```

➤ 제공 버전

7.0.0.0

GroupActionMenu Method

➤ 기능

그룹행 영역에 설정된 컨텍스트 메뉴를 설정하거나 확인 한다.

MenuText, MenuCode를 "|" 연산자로 조합한 문자열로 설정한다.

MenuCode 인자 설정시 사전 정의 코드를 사용하는 경우 별도 코드 작성없이 해당 기능이 처리되고, 그렇지 않은 경우에는 OnSelectMenu 이벤트에서 처리할 수 있다.

MenuText를 설정하지 않는 경우 기본메뉴로 설정처리 된다.

➤ Syntax

Syntax	Get	ObjId. GetGroupActionMenu()
--------	-----	------------------------------------

➤ Info

Return	Object, MenuText, MenuCode로 이루어진 객체		
Parameter	Type	필수여부	설 명

➤ Example

```
// 그룹행의 컨텍스트 메뉴 확인
console.log("groupMenu:", mySheet.GetGroupActionMenu());
```

➤ Syntax

Syntax	Set	ObjId. SetGroupActionMenu (MenuText, MenuCode)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
MenuText	String	필수	메뉴명을 구분자 " "로 연결한 문자열



MenuCode	String	필수	메뉴코드를 구분자 " "로 연결한 문자열
----------	--------	----	------------------------

기본으로 제공되는 사전 정의 코드는 아래와 같다.

Code	기능
_ibInitGroupCols	그룹 정보 초기화
_ibSaveGroupCols	그룹 정보 저장
_ibResetGroupCols	그룹 정보 저장 취소

➤ **Example**

```
//컬럼 저장 메뉴 구성  
mySheet.SetGroupActionMenu("그룹 저장|그룹 저장 취소",  
"_ibSaveGroupCols|_ibResetGroupCols");
```

➤ **제공 버전**

7.0.13.40	
------------------	--

HeaderActionMenu Method

➤ 기능

헤더 컨텍스트 메뉴를 설정하거나 확인한다.

SetConfig 또는 ibsheet.cfg 에서 UseHeaderActionMenu 를 1로 설정한 경우에만 사용 가능하다.

UseHeaderActionMenu를 설정하고 SetHeaderActionMenu를 설정하지 않는 경우 자동으로 기본 메뉴가 출력된다.

MenuText, MenuCode를 "|" 연산자로 조합한 문자열로 설정한다.

MenuCode를 시트에서 제공하는 고정코드를 설정하는 경우 제품 내에서 해당 기능을 처리 한다.

MenuText를 설정하지 않는 경우 기본메뉴로 설정처리 된다.

➤ Syntax

Syntax	Get	ObjId. GetHeaderActionMenu()
--------	-----	-------------------------------------

➤ Info

Return	Object, MenuText, MenuCode로 이루어진 객체		
Parameter	Type	필수여부	설 명

➤ Example

```
// 설정된 헤더메뉴 확인
var menu = mySheet.GetHeaderActionMenu();
console.log(menu.MenuText); // 설정된 MenuText 확인
console.log(menu.MenuCode); // 설정된 MenuCode 확인
```

➤ Syntax

Syntax	Set	ObjId. SetHeaderActionMenu (MenuText, MenuCode)
--------	-----	--

➤ Info



Return	None		
Parameter	Type	필수여부	설 명
MenuText	String	선택	" " 문자로 연결한 메뉴명 문자열
MenuCode	String	선택	" " 문자로 연결한 메뉴코드 문자열. 시트에서 제공하는 고정 코드를 설정하는 경우 제품내에서 해당 기능을 처리한다.

* 고정코드

Code	설 명
_ibColSortAsc	대상 컬럼 오름 차순 정렬
_ibColSortDesc	대상 컬럼 내림 차순 정렬
_ibColHidden	대상 컬럼 숨김.
_ibCancelColHidden	헤더메뉴로 숨긴 컬럼 출력
_ibSaveColPosition	현재 컬럼정보 저장 (위치, 컬럼 숨김여부, 너비)
_ibResetColPosition	저장된 컬럼정보 삭제
_ibShowFilter	필터행 출력
_ibHideFilter	필터행 숨김

➤ Example

// aaa 메뉴 클릭시 필터를 출력하고, bbb 메뉴 클릭시 필터를 숨기는 헤더 메뉴를 설정한다.

```
mySheet.SetHeaderActionMenu("aaa|*-|bbb", "_ibShowFilter|_ibHideFilter");
```

➤ 제공 버전

7.0.0.0	
7.0.13.6	_ibColSortAsc, _ibColSortDesc 고정 코드 추가



HeaderBackColor Method

➤ 기능

헤더 행의 바탕색을 설정하거나 확인한다.

색상 지정은 WebColor값을 사용하여 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetHeaderBackColor()
--------	-----	------------------------------------

➤ Info

Return	String, 현재 배경 색상 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

//헤더의 배경색 확인한다. mySheet.GetHeaderBackColor();
--

➤ Syntax

Syntax	Set	ObjId. SetHeaderBackColor (Color)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Color	String	필수	헤더행의 배경 색상

➤ Example

//헤더의 배경색을 빨간색으로 설정한다. mySheet.SetHeaderBackColor("#FF0000");
--

➤ 제공 버전



7.0.0.0

HeaderCheck Method

➤ 기능

헤더의 전체체크박스 값을 확인 및 설정 한다.

설정의 경우 마우스로 클릭하여 처리되는것과 달리 단순 .체크/언체크만 설정할 수 있다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 모든행에 대해서 처리를 하고 SaveName으로 설정할 경우 해당 SaveName이 포함된 행에 대해서만 처리 한다.

➤ Syntax

Syntax	Get	ObjId. GetHeaderCheck (Row, Col)
--------	-----	---

➤ Info

Return	Boolean, 현재 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Number	필수	대상 행의 Index
Col	Number / String	필수	대상 컬럼의 Index 또는 SaveName

➤ Example

```
// 헤더의 3번째 컬럼의 전체 체크박스 값을 확인 한다.
var CheckValue = mySheet.GetHeaderCheck(0, 3);
```

➤ Syntax

Syntax	Set	ObjId. SetHeaderCheck (Row, Col, Value)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명



Row	Number	필수	대상 행의 Index
Col	Number / String	필수	대상 컬럼의 Index 또는 SaveName
Value	Boolean	필수	체크박스에 설정하고자 하는 값

➤ **Example**

```
// 헤더의 3번째 컬럼의 전체 체크박스 값을 연체크 설정 한다..  
mySheet.SetHeaderCheck(0, 3, 0);
```

➤ **제공 버전**

7.0.0.0

HeaderCursor Method

➤ 기능

헤더의 마우스 오버시 커서의 모양을 설정한다.

"Default","Pointer" 로 설정이 가능하며 기본적으로 "Default"값이 설정된다.

➤ Syntax

Syntax	Get	ObjId. GetHeaderCursor()
--------	-----	---------------------------------

➤ Info

Return	String		
Parameter	Type	필수여부	설 명

➤ Example

```
// 헤더의 마우스 커서포인트를 확인한다.
mySheet.GetHeaderCursor();
```

➤ Syntax

Syntax	Set	ObjId. SetHeaderCursor (Cursor)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Cursor	String	필수	마우스 모양 설정

➤ Example

```
// 헤더의 마우스 커서포인트를 손가락 모양으로 변경한다.
mySheet.SetHeaderCursor('Pointer');
```



- 제공 버전
7.0.13.60



HeaderFontBold Method

➤ 기능

헤더 행의 글자 볼드 여부를 설정하거나 확인한다. 기본적으로 헤더 글자를 볼드하지 않지만 볼드하게 사용할 때 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetHeaderFontBold ()
--------	-----	------------------------------------

➤ Info

Return	Boolean, 현재 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//헤더의 글자 Bold 설정 확인한다.
mySheet.GetHeaderFontBold();
```

➤ Syntax

Syntax	Set	ObjId. SetHeaderFontBold (Bold)
--------	-----	--

➤ Info

Return	Boolean, 현재 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Bold	Boolean	필수	Header행의 Font Bold 여부

➤ Example

```
//헤더의 글자를 Bold로 설정한다.
mySheet.SetHeaderFontBold(1);
```

➤ 제공 버전



7.0.0.0



HeaderFontColor Method

➤ 기능

헤더 행의 글자색을 설정하거나 확인한다.

➤ Syntax

Syntax	Get	ObjId. GetHeaderFontColor ()
--------	-----	-------------------------------------

➤ Info

Return	String, 현재 설정값(Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//헤더의 글자색을 검은색으로 설정한다.  
mySheet.GetHeaderFontColor();
```

➤ Syntax

Syntax	Set	ObjId. SetHeaderFontColor (Color)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Color	String	필수	설정하고자 하는 WebColor

➤ Example

```
//헤더의 글자색을 검은색으로 설정한다.  
mySheet.SetHeaderFontColor("#000000");
```

➤ 제공 버전

7.0.0.0



HeaderRows Method

➤ 기능

헤더행의 개수를 확인한다.

이 속성은 InitHeaders() 함수를 통해서 설정된 헤더 행의 개수를 의미한다

➤ Syntax

Syntax	ObjId.HeaderRows()
--------	--------------------

➤ Info

Type	Long, 헤더행의 개수	
Parameter	Type	설 명
없음		

➤ Example

//헤더 행의 개수를 확인한다. Alert("헤더 행의 개수는 " + mySheet.HeaderRows() + "개 입니다.");

➤ 제공 버전

7.0.0.0

HeaderRowHeight Method

➤ 기능

헤더 행의 행높이를 설정하거나 확인한다. 값은 픽셀 단위로 설정한다

➤ Syntax

Syntax	Get	ObjId. GetHeaderRowHeight()
--------	-----	------------------------------------

➤ Info

Return	Integer, 현재 설정값(Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//헤더 행의 높이를 확인
mySheet.GetHeaderRowHeight();
```

➤ Syntax

Syntax	Set	ObjId. SetHeaderRowHeight (Height)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Height	Integer	필수	설정하고자 하는 픽셀값

➤ Example

```
//헤더 행의 높이를 30 픽셀로 설정.
mySheet.SetHeaderRowHeight(30);
```

➤ 제공 버전

7.0.0.0



HideFilterRow Method

➤ 기능

IBSheet 상단 고정행의 필터행을 삭제한다.

필터행이 있는 경우 삭제되고, 없으면 아무일도 발생하지 않는다.

➤ Syntax

Syntax	ObjId.HideFilterRow()
--------	-----------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
없음			

➤ Example

//필터행 삭제하기 mySheet.HideFilterRow();
--

➤ 제공 버전

7.0.0.0

HideGroupRow Method

➤ 기능

시트의 그룹행을 삭제하거나 숨김처리 한다.

그룹행 삭제시에는 그룹핑 상태인 경우 모두 초기화 되며 숨김시에는 그룹핑 상태가 그대로 유지 된다.

숨김상태인 경우 ShowGroupRow 메소드를 통해 보이도록 처리할 수 있다.

➤ Syntax

Syntax	ObjId.HideGroupRow([Del])
--------	---------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Del	Boolean	선택	삭제 여부 (Default="1")

➤ Example

<pre>//그룹행 삭제 mySheet.HideGroupRow(); // 그룹행 숨김 mySheet.HideGroupRow(0)</pre>
--

➤ 제공 버전

7.0.10.0	
----------	--



HideProcessDlg Method

➤ 기능

시트에 표시되어 있는 대기중 이미지를 닫는다.

➤ Syntax

Syntax	ObjId.HideProcessDlg()
--------	------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// 대기중 이미지 닫기  
mySheet.HideProcessDlg();
```

➤ 제공 버전

7.0.0.0

HideSubSum Method

➤ 기능

ShowSubSum 함수를 호출하면 다음 조회시 조회 데이터 사이에 소계행이 삽입된다. 이 메서드를 사용하면 ShowSubSum에서 시작한 소계기능을 다음 조회에서 더 이상 추가되지 않도록 한다.

이 메서드는 호출 즉시 동작하는 것이 아니라, 다음 조회시에 반영된다.

➤ Syntax

Syntax	ObjId.HideSubSum([StdCol])
--------	----------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
StdCol	Long/String	선택	소계를 표시한 기준 컬럼의 컬럼Index 또는 SaveName. Default=""(전체컬럼)

➤ Example

```
//1컬럼을 중심으로 소계 계산하여 표시하기
var info = [{StdCol:1, SumCols:"3|4|5"}];
mySheet.ShowSubSum(info);

//표시된 소계 숨기기
mySheet.HideSubSum();
```

➤ 제공 버전

7.0.0.0



HighlightAfterSort Method

➤ 기능

소트 후 하이라이트 처리 방법을 확인하거나 설정한다.

설정 가능한 값은 아래와 같다.

설정값	설명
0	포커스 클리어
1	이전 포커스 유지 및 해당 셀의 위치로 이동
2	이전 포커스 유지 및 scroll 위치 초기화

➤ Syntax

Syntax	Get	ObjId. GetHighlightAfterSort()
--------	-----	---------------------------------------

➤ Info

Return	Number, 현재 설정값(Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

//소트 이후 설정 확인 mySheet.GetHighlightAfterSort();

➤ Syntax

Syntax	Set	ObjId. SetHighlightAfterSort (Mode)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Mode	Number	필수	Sort 후 포커스 설정방법 (Default=1)



➤ **Example**

```
//소트 이후 기존에 선택하고 있던 데이터를 찾아감.  
mySheet.SetHighlightAfterSort(1);
```

➤ **제공 버전**

7.0.0.0	
7.0.13.16	Mode:2 추가



IBCloseCalendar Method

➤ 기능

외부의 컨트롤에서 사용한 달력팝업을 닫도록 설정한다.

➤ Syntax

Syntax	ObjId.IBCloseCalendar()
--------	-------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
없음			

```
// 달력 팝업을 띄운다.  
mySheet.IBShowCalendar("20121116", {Format:"yyyy/MM/dd", X:300, Y:600});  
  
// 띄운 달력 팝업 Dialog를 닫는다.  
mySheet.IBCloseCalendar();
```

➤ 제공 버전

7.0.0.0

IBShowCalendar Method

➤ 기능

달력 팝업을 외부의 컨트롤에서 사용하도록 설정한다

일반적으로 Callback 함수를 지정하는 경우 첫번째 인자로 선택한 날짜 문자열이 전달되며, 그 외 Callback 함수로 전달하고자 하는 인자가 있는 경우 CallbackParam 을 설정한다. CallbackParam으로 설정된 객체는 Callback 함수의 두번째 인자로 전달된다.

➤ Syntax

Syntax	ObjId.IBShowCalendar(val, [obj]);
--------	-----------------------------------

➤ Info

Return	String, (달력 팝업 Dialog에서 선택한 날짜값)		
Parameter	Type	필수여부	설 명
val	String	필수	날짜데이터값 (Default=현재날짜)
obj	Object	선택	기능을 json 형태로 설정한다.

세부내용

컬럼별로 설정 할 수 있는 속성은 다음과 같다.

속성	타입	설명
Format	String	날짜포맷 (Default="Ymd")
Target	String, Object	"Mouse" (마지막 마우스 위치 사용시, Default) 또는 달력 버튼 Object (달력 버튼 위치 사용시)
Result	Object	선택 결과를 설정할 객체 (input 객체)
X	Integer	(좌표값 사용시), X 축 좌표값
Y	Integer	(좌표값 사용시), Y 축 좌표값
Callback	String	Function Name Return 시 사용되는 함수명
CallbackParam	Object	Callback 함수에서 받을 인자 객체



CalButtons	String	달력팝업의 사용할 버튼 설정 사용하고자 하는 버튼을 " " 로 연결하여 설정한다. <table><tr><th>설정값</th><th>내용</th></tr><tr><td>Close</td><td>취소 버튼</td></tr><tr><td>Today</td><td>오늘일자 입력 버튼</td></tr><tr><td>Yesterday</td><td>어제일자 입력 버튼</td></tr></table>	설정값	내용	Close	취소 버튼	Today	오늘일자 입력 버튼	Yesterday	어제일자 입력 버튼
설정값	내용									
Close	취소 버튼									
Today	오늘일자 입력 버튼									
Yesterday	어제일자 입력 버튼									
CalButtonAlign	String	달력팝업에서 버튼을 사용하는 경우 정렬방법을 설정 <table><tr><th>설정값</th><th>내용</th></tr><tr><td>Left</td><td>왼쪽정렬</td></tr><tr><td>Center</td><td>가운데정렬 (기본값)</td></tr><tr><td>Right</td><td>오른쪽정렬</td></tr></table>	설정값	내용	Left	왼쪽정렬	Center	가운데정렬 (기본값)	Right	오른쪽정렬
설정값	내용									
Left	왼쪽정렬									
Center	가운데정렬 (기본값)									
Right	오른쪽정렬									

➤ Example

```
// 날짜데이터값 입력
var val = document.getElementById("DateText").value;

// 달력 팝업 Dialog 위치 : X, Y 좌표값 사용시
7. var obj = { Format: "Ymd", X:300, Y:600, CallBack: "Test" };

// 달력 팝업 Dialog 위치 : 마우스 마지막 위치 사용시
7. var obj = { Format: "Ymd", Target:"Mouse", CallBack: "Test" };

// 달력 팝업 Dialog 위치 : 달력 버튼 위치 사용시
7. var obj = { Format:"Ymd",
Target:document.getElementById("DateBtn"), CallBack: "Test" };

// CalButtons 속성 : 달력 닫기 버튼 옵션 설정 사용시
7. var obj = { Format:"Ymd",
Target:document.getElementById("DateBtn"), CallBack: "Test",
CalButtons : "Close"};
```



// 달력 닫기 버튼 옵션 설정 사용시 정렬 설정: 왼쪽 정렬

```
7. var obj = { Format:"Ymd",  
    Target:document.getElementById("DateBtn"), CallBack: "Test",  
    CalButtons : "Close", CalButtonAlign : "Left" };
```

// fnName : 함수명으로 function 생성

```
function Test (date){
```

// 달력 팝업 Dialog서 날짜 선택시 리턴값

```
    document.getElementById("DateText").value = date;  
}
```

➤ **제공 버전**

7.0.0.0



ImageList Method

➤ 기능

N번째 이미지 파일의 웹서버 경로를 확인하거나 설정한다.

Type이 Image인 셀에 표시하는 이미지의 경로를 미리 설정하여 1회 다운 받아 재사용하기 위해 이미지 파일을 기억을 기억하기 위한 속성이다 .

조회 및 신규행입력시 사용된다.

이미지 파일의 경로는 현재 페이지를 기준으로 상대경로를 사용하거나 웹서버 루트 이하의 절대경로를 설정하도록 한다.

인덱스로 설정할 경우 0부터 시작하며 원하는 개수만큼 설정하도록한다.

[설정하는 방법]

```
mySheet.SetImageList(0, "../image/btn_search.gif");
```

```
mySheet.SetImageList(1, "/common/image/btn_cal.gif");
```

[이미지관련 값변경시 사용하는 함수 정리]

이미지설정	상황	Image
ImageList	이미지변경	SetCellValue,SetCellImage SetSearchingImage,SetSavingImage, SetWaitImage,SetKeyFieldImage
	값변경	해당없음

➤ Syntax

Syntax	Get	ObjId. GetImageList (Index)
--------	-----	------------------------------------

➤ Info

Return	String, 해당 인덱스에 설정한 Image경로 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Index	Integer	필수	Image 인덱스

➤ Example



```
// 해당 인덱스에 설정한 이미지경로 확인  
mySheet.GetImageList(0);
```

➤ **Syntax**

Syntax	Set	ObjId. SetImageList (Index, Url)
--------	-----	---

➤ **Info**

Return	None (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Index	Integer	필수	Image 인덱스
Url	String	필수	설정할 Image 경로

➤ **Example**

```
// 0 인덱스에 이미지경로 설정  
mySheet.SetImageList(0, "../image/btn_search.gif");  
  
// 0 인덱스에 저장된 이미지 사용(Type:Image)  
mySheet.SetCellValue(3, 5, 0);  
mySheet.SetCellImage(3, 5, 0);  
mySheet.SetSearchingImage(0);  
mySheet.SetSavingImage(0);  
mySheet.SetWaitImage(0);  
mySheet.SetKeyFieldImage(0);
```

➤ **제공 버전**

7.0.0.0



InitCellProperty Method

➤ 기능

조회 또는 행 추가 후 특정 셀의 속성 정의를 기본 컬럼 속성 정의와 다르게 하고자 하는 경우에 이 함수를 사용한다.

(단, 날짜포맷 설정은 Ymd, Ym, Md, YmdHms, YmdHm 포맷에 대하여 설정이 가능하다.)

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

InitColumns Method에서 정의한 컬럼의 타입이 Seq, Status, DelCheck, AutoSum, AutoAvg 인 경우는 해당 함수를 사용 할 수 없으며 설정이 가능한 Type은 다음과 같다.

Type	설명
CheckBox	CheckBox 형태 데이터
Combo	Edit 불가능 Combo 데이터
ComboEdit	자동완성형태의 Combo 데이터
Date	날짜형 타입
Float	실수형 타입
Image	Edit 불가능한 단순 이미지 표현 형태 데이터
Int	정수형 타입
Popup	팝업을 사용한 데이터
PopupCode	팝업메뉴의 코드 집합
PopupText	팝업메뉴의 문자열 집합
Text	기본 문자열 데이터

이 함수에서 설정이 가능한 속성은 다음과 같다.

속성	타입	설명
Type	String	컬럼의 데이터 타입(필수)



AcceptKeys	String	<p>입력허용키 설정</p> <p>"N" : 숫자만 허용</p> <p>"E" : 영문만 허용</p> <p>"N E" : 숫자, 영문만 허용</p> <p>"N E [사용자정의문자]" : 숫자, 영문과 사용자정의문자로 설정한 문자 허용</p> <p>예) "N [abc!]" 설정 시 숫자와 a, b, c, ! 문자만 입력 가능</p> <p>사용자 정의 문자열은 꼭 대괄호로 묶어서 설정해야 하며 사용자정의문자와 숫자,영문 설정 문자의 합집합으로 처리된다.</p> <p>AcceptKeys 와 ExceptKeys를 둘다 설정하고 설정키 내에 동일 문자가 있는 경우 해당 문자는 입력불가 처리된다.</p>
Align	String	데이터의 정렬
ApproximateType	Number	근사값처리방식 (1:반올림, 2:내림, 3:올림)
ComboCode	String	콤보 리스트의 코드 집합
ComboText	String	콤보 리스트의 문자열 집합
Cursor	String	마우스 모양 설정
CustomDate	Number	사용자 정의 날짜 사용여부
Edit	Boolean	편집가능 여부
EditLen	Number	데이터의 입력가능한 글자수
EmptyToReplaceChar	String	빈값인 경우 대체하여 보여줄 문자 설정
ExceptKeys	String	<p>입력제외키 설정</p> <p>"N" : 숫자 입력 불가</p> <p>"E" : 영문 입력 불가</p> <p>"N E" : 숫자, 영문 입력 불가</p> <p>"N E [사용자정의문자]" : 숫자, 영문과 사용자정의문자로 설정한 문자 입력 불가</p> <p>예) "N [abc!]" 설정 시 숫자와 a, b, c, ! 문자만 입력 불가</p>



		<p>사용자 정의 문자열은 꼭 대괄호로 묶어서 설정해야 하며 사용자정의문자와 숫자,영문 설정 문자의 합집합으로 처리된다.</p> <p>AcceptKeys 와 ExceptKeys를 둘다 설정하고 설정키 내에 동일 문자가 있는 경우 해당 문자는 입력불가 처리된다.</p>
Format	String	데이터의 Mask 적용 형태
FormatFix	Boolean	GetCellValue 시 GetCellText 값을 반환할지 여부 (Default=0) 특히 True 로 설정 시 저장할 때 Format 이 설정된 상태로 저장한다.
HoverUnderline	Boolean	마우스오버시 언더라인 여부
Image	String	이미지의 경로
ImgAlign	String	이미지 표시 위치
ImgHeight	Number	이미지 높이
ImgWidth	Number	이미지 너비
InputCaseSensitive	Number	<p>입력시 영문에 대한 대소구문 자동 치환 처리에 대한 설정</p> <p>0 : 별도 처리 하지 않음 (기본값)</p> <p>1 : 대문자로 치환</p> <p>2 : 소문자로 치환</p>
ItemCode	String	멀티체크 아이템의 코드를 구분자 " "로 연결한 문자열
ItemText	String	멀티체크 아이템의 텍스트를 구분자 " "로 연결한 문자열
MaxCheck	Number	멀티체크의 최대 선택 개수 설정
MultiLineText	Boolean	다중라인 입력여부
PointCount	Number	컬럼타입이 Float 인 경우 소수점 이하의 자리수
PopupButton	Boolean	팝업버튼 표시 여부
PopupCode	String	팝업메뉴의 코드 집합

PopupText	String	팝업메뉴의 문자열 집합
Radiolcon	Number	데이터 셀내의 체크박스 계열의 이미지 모양 설정
ZeroToReplaceChar	String	값이 0인 경우 대체하여 보여줄 문자 설정

각 속성의 상세 설명은 InitColumns Method 설명을 참조 한다.

이 함수를 사용하기 위해서는 Type을 필수로 설정 해야 한다.

➤ Syntax

Syntax	ObjId.InitCellProperty(Row, Col, info);
--------	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	대상 행의 Index
Col	Long / String	필수	대상이 되는 컬럼 Index 또는 SaveName
info	Object	필수	셀의 속성 정의 객체

➤ Example

<pre>// Text 컬럼 타입으로 변경 var info = {Type: "Text", Align: "Center", Edit: 0}; mySheet.InitCellProperty(2, "sCombo", info); // 허용키, 제외키 설정 var info = {Type:"Text",AcceptKeys:"N [abcd!@#]", ExceptKeys:"[890!]"}; mySheet.InitCellProperty(3, 4 ,info);</pre>

➤ 제공 버전

7.0.0.0	
7.0.7.2	팝업버튼 사용여부 설정을 위한 PopupButton 속성 추가
7.0.12.3	ItemCode, ItemText, MaxCheck 속성 추가



7.0.13.12	Radiolcon 속성 추가
7.0.13.24	EmptyToReplaceChar, ZeroToReplaceChar 속성 추가

InitColumns Method

➤ 기능

각 컬럼의 데이터 타입, 포맷 및 기능들을 설정한다..

➤ Syntax

Syntax	ObjId.InitColumns(Cols, DataRow);
--------	-----------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Cols	Object	필수	각 컬럼의 기능을 json 형태로 설정한다.
DataRow	Number	선택	단위데이터행 개수 단위데이터행 구조를 사용하는 경우 해당 인자의 값과 Cols 인자의 배열 길이가 동일해야 한다.

➤ 세부내용

컬럼 별로 설정 할 수 있는 속성은 다음과 같다.

※ 컬럼별 설정이 가능한 속성여부는 [Appendix#1.컬럼 타입별 정의 가능한 속성 참조](#)

속성	타입	설명
Type	String	컬럼의 데이터 타입 (필수)
AcceptKeys	String	입력허용키 설정 "N" : 숫자만 허용 "E" : 영문만 허용 "N E" : 숫자, 영문만 허용 "N E [사용자정의문자]" : 숫자, 영문과 사용자정의문자로 설정한 문자 허용



		<p>예) "N [abc!]" 설정 시 숫자와 a, b, c, ! 문자만 입력 가능</p> <p>사용자 정의 문자열은 꼭 대괄호로 묶어서 설정해야 하며 사용자정의문자와 숫자,영문 설정문자의 합집합으로 처리된다.</p> <p>AcceptKeys 와 ExceptKeys를 둘다 설정하고 설정키 내에 동일 문자가 있는 경우 해당 문자는 입력불가 처리된다.</p>
ActionMenu	Object[]	컨텍스트 메뉴
Align	String	데이터의 정렬
AllowNull	Boolean	숫자계열 컬럼에 빈값 허용 여부 설정
ApproximateType	Number	근사값처리방식 설정
BackColor	String	배경 색상
ButtonUrl	String/Number	팝업 버튼의 이미지 경로 또는 이미지 리스트의 Index
CalcLogic	String / Object	컬럼별 계산식 문자열 또는 설정 객체
CaseSensitive	Boolean	필터링시 대소구분 처리 여부 (Default=1)
Chart	Object	스파크라인 차트 타입에 대한 세부 속성 설정
CheckSaveName	String	트리기준컬럼의 체크박스에 대한 SaveName
ClassName	String	버튼의 사용자정의 클래스 postfix 명
ColMerge	Number	컬럼의 셀 병합 처리 방법
ColSpan	Number	ColSpan 범위 값 (단위데이터행 구조에서만 사용 가능)



ComboCode	String	콤보 리스트의 코드 집합
ComboFilter	Number	ComboEdit 컬럼 타입의 필터링 처리 모드 설정
ComboText	String	콤보 리스트의 문자열 집합
Cursor	String	마우스 모양 설정
CustomDate	Boolean	사용자 정의 날짜 사용여부
DefaultValue	String	신규입력시 기본값 설정
Edit	Boolean	편집가능 여부
EditLen	Number	데이터의 입력가능한 글자수
EditPointCount	Number	편집시점에서의 소수점 자리수 입력 제한 개수 설정
Ellipsis	Boolean	말줄임 여부
EmptyToReplaceChar	String	빈값인 경우 대체하여 보여줄 문자 설정
EnterMode	Boolean	다중라인 입력모드(MultiLineText)에서 Enter 키 입력에 대한 개행 처리 여부 (Default:0)
ExceptKeys	String	<p>입력제외키 설정</p> <p>"N" : 숫자 입력 불가</p> <p>"E" : 영문 입력 불가</p> <p>"N E" : 숫자, 영문 입력 불가</p> <p>"N E [사용자정의문자]" : 숫자, 영문과 사용자정의문자로 설정한 문자 입력 불가</p> <p>예) "N [abc!]" 설정 시 숫자와 a, b, c, ! 문자만 입력 불가</p> <p>사용자 정의 문자열은 꼭 대괄호로 묶어서 설정해야 하며 사용자정의문자와 숫자,영문 설정문자의 합집합으로 처리된다.</p>



		AcceptKeys 와 ExceptKeys를 둘다 설정하고 설정키 내에 동일 문자가 있는 경우 해당 문자는 입력불가 처리된다.
ExcludeEmpty	Boolean	AutoSum 컬럼타입의 평균 또는 건수 계산시 빈값을 포함 할지 여부 (Default=0)
FalseValue	String	1 이외의 CheckBox 형태 컬럼의 False 값 지정. "F" 으로 지정한 경우 0 대신 "F"를 False 값으로 사용 가능.
FitColWidth	Boolean	FitColWidth 메소드 호출시 비율에 의한 너비 재조정 허용 여부 (Default=1)
Focus	Boolean	포커스 허용 여부
FontBold	Boolean	폰트 굵기 여부
FontColor	String	폰트 색상
FontUnderline	Boolean	폰트 언더라인 여부
Format	String / Array	데이터의 Mask 적용 형태
FormatFix	Boolean	GetCellValue 시 GetCellText 값을 반환 할지 여부 (Default=0) 특히 True 로 설정시 저장할 때 Format 이 설정된 상태로 저장한다.
FullInput	Boolean	컬럼타입이 단일행 문자열인 경우 전체길이(EditLen)만큼의 입력 여부
GroupSumType	String	그룹핑 처리시 부모노드의 셀에 처리할 계산 방식 설정 (Default="") "Sum" : 합계계산 "Count" : 건수계산
HeaderCheck	Boolean	헤더에 전체 체크 표시 여부 (Default=1)
Hidden	Boolean	컬럼 숨김 여부
HoverUnderline	Boolean	마우스오버시 언더라인 여부



Image	String	이미지 표현시 Url
ImgAlign	String	이미지의 정렬
ImgHeight	Number	이미지 높이
ImgWidth	Number	이미지 너비
InputCaseSensitive	Number	입력시 영문에 대한 대소구문 자동 치환 처리에 대한 설정 0 : 별도 처리 하지 않음 (기본값) 1 : 대문자로 치환 2 : 소문자로 치환
InsertEdit	Boolean	트랜잭션이 "입력"인 상태에서 데이 터의 Edit 가능 여부
ItemCode	String	멀티체크 아이템의 코드를 구분자 " "로 연결한 문자열
ItemText	String	멀티체크 아이템의 텍스트를 구분자 " "로 연결한 문자열
KeyField	Boolean	필수 입력 항목
LevelSaveName	String	트리레벨의 데이터 저장 또는 조회 시 사용하는 변수명
MaxCheck	Number	멀티체크의 최대 선택 개수 설정 (Default : 0)
MaximumValue	Number	숫자포맷일 때 편집시 입력할수 있 는 최대값
MinimumValue	Number	숫자포맷일 때 편집시 입력할수 있 는 최소값
MinWidth	Number	컬럼 최소 가로크기
MultiLineText	Boolean	다중라인 입력여부
NumberSort	Boolean	숫자형 Sort 처리 사용 여부
PointCount	Number	컬럼타입이 Float 인 경우 소수점 이 하의 자리수
PopupButton	Boolean	팝업버튼 사용 여부
PopupCheckEdit	Boolean	팝업메뉴 설정시 편집가능 여부
PopupCode	String	팝업메뉴의 코드 집합



PopupText	String	팝업메뉴의 문자열 집합
Radiolcon	Boolean	데이터 셀내의 체크박스 계열의 이미지 모양 설정
RowMerge	Boolean	대상 컬럼의 RowMerge 허용 여부 (Default=1)
RowSpan	Number	RowSpan 범위 값 (단위데이터행 구조에서만 사용 가능)
Save	Boolean	저장 또는 저장관련 메소드에서 해당 컬럼의 값 포함 여부 (Default=1)
SaveName	String	데이터 저장 또는 조회시 사용하는 변수명
ShowCol	Number	멀티콤보 설정 시 컬럼 값으로 출력할 컬럼 설정
Sort	Boolean	헤더 클릭시 소트 가능 여부 (Default=1)
StaticPassword	Boolean	값이 0인 경우 데이터 길이만큼 '*'의 수를 표현한다.
SumType	String	합계 계산 방식 설정 (Default="Sum")
ToolTip	Boolean/String	셀의 풍선도움말 표시 여부 또는 설정할 문자열
ToolTipText	String	헤더행의 풍선 도움말에 설정할 문자열
Transaction	Boolean	트랜잭션 허용 여부 (Default=1)
TreeCheck	Boolean	트리 기준컬럼에서 체크박스 사용여부
TreeCol	Boolean	트리 기준컬럼
TrueValue	String	1 이외의 CheckBox 형태 컬럼의 True 값 지정. "M" 으로 지정한 경우 1 대신 "M"을 True 값으로 사용 가능

		능.
UpdateEdit	Boolean	트랜잭션이 "조회"인 상태에서 데이터의 Edit 가능 여부
VAlign	String	컬럼 상/하 정렬 (Default="middle")
Validation	Boolean	콤보에디트 컬럼 유효성 검사 사용 여부 설정 (Default=0)
Width	Number	컬럼의 너비
Wrap	Boolean	자동 줄바꿈 여부
ZeroToReplaceChar	String	값이 0인 경우 대체하여 보여줄 문자 설정

Type은 해당 컬럼에 대한 Type 설정으로 다음과 같은 값을 설정 할 수 있다.

Type	설명
Text	기본 문자열 데이터 타입
Status	행에 대한 트랜잭션 상태를 표현하는 데이터 타입
DelCheck	행에 대한 삭제여부를 설정하는 CheckBox 형태 데이터 타입
CheckBox	CheckBox 데이터 타입
DummyCheck	이벤트를 발생하지 않는 CheckBox 데이터 타입
Radio	데이터 행 중 하나의 데이터만 선택하는 Radio 데이터 타입
Combo	DropDown 리스트 데이터 타입
ComboEdit	편집 및 필터링이 가능한 DropDown 리스트 데이터 타입 (모바일지원안함)
AutoSum	합계행에 합계를 표현하는 숫자 데이터 타입
Image	이미지 형태의 데이터 타입
Int	정수형 숫자 데이터 타입
Float	실수형 숫자 데이터 타입
Date	날짜 데이터 타입
Popup	우측에 팝업 버튼을 갖는 읽기 전용 문자열 데이터 타입
Pass	Password 데이터 타입
Seq	행의 생성 순서값을 표현하는 데이터 타입



Html	Html 태그형태를 표현하는 데이터 타입
Result	저장 처리 결과를 표시하는 데이터 타입
Sparkline	스파크라인 차트를 표현하는 데이터 타입
Button	버튼 형태를 표현하는 데이터 타입

※ 모바일에서는 **ComboEdit** 타입은 **Combo** 타입으로 대체되어 표현된다.

Align은 데이터에 대한 정렬 설정 Left, Center, Right 값을 설정한다.

DataAlign	설명
Left	좌측 정렬 데이터(Default)
Center	가운데 정렬 데이터
Right	우측 정렬 데이터

AcceptKeys는 컬럼의 입력허용키를 설정한다.

"N" : 숫자만 허용

"E" : 영문만 허용

"N|E" : 숫자, 영문만 허용

"N|E|[사용자정의문자]" : 숫자, 영문과 사용자정의문자로 설정한 문자 허용

예) "N|[abc!]" 설정 시 숫자와 a, b, c, ! 문자만 입력 가능

사용자 정의 문자열은 꼭 대괄호로 묶어서 설정해야 하며 사용자정의문자와 숫자,영문 설정문자의 합집합으로 처리된다.

AcceptKeys 와 **ExceptKeys**를 둘다 설정하고 설정키 내에 동일 문자가 있는 경우 해당 문자는 입력불가 처리된다.

ActionMenu는 컬럼의 컨텍스트 메뉴를 설정하는 속성이다. 항목의 Text, Code, Icon 을 갖는 객체의 배열로 설정을 기본으로 하며, 약식 설정 방법인 항목의 Text 배열로 설정 할 수 있다. 항목의 Code 속성을 설정하지 않거나, 약식 방법으로 설정하는 경우 Code는 Text 설정값과 동일한 값으로 설정 된다. 메뉴 선택시 OnSelectMenu 이벤트를 발생하고, 시트의 ActionMenu와 같이 사용시 컬럼의 ActionMenu 설정이 우선적으로 처리된다.

설정 방법은 아래와 같다.

```
// 일반 설정
var actionMenu = [
```

```
{'Text':'메뉴01', Code:'01'}, {'Text':'메뉴02', Code:'02'}, {'Text':'메뉴02', Code:'03'}
];

// 일반 설정 : Code 생략
var actionMenu = [
{'Text':'메뉴01'}, {'Text':'메뉴02'}, {'Text':'메뉴02'}
];

// 약식 설정
var actionMenu = ['메뉴01', '메뉴02', '메뉴03'];
```

BackColor는 컬럼의 배경색상을 설정 한다.

ButtonUrl은 컬럼타입이 Popup일 때 표시되는 팝업 버튼의 이미지를 변경하고자 할 때 사용하는 속성 이다. 이 속성을 설정하지 않은 경우 기본적으로 테마의 "popup.gif" 이미지를 표시 하고, 날짜 포맷을 설정한 경우 "calendar.gif" 이미지를 표시 한다. 별도의 이미지를 사용하고자 하는 경우에 이 속성을 이용하여 설정 한다. 설정 값은 설정하고자 하는 이미지의 경로 문자열 또는 이미지리스트의 Index값으로 설정 하며, 설정 시 설정된 테마의 경로를 기준으로 설정하여야 한다. 이미지의 사이즈는 12*12 로 사용해야 올바르게 표현이 된다. 동적 변경이 아닌 경우에는 테마의 해당 이미지를 교체하여 사용하는 것을 권장 한다.

CalcLogic 속성은 행 내의 컬럼간 계산식을 설정할 때 사용하는 속성이다. 설정 방법은 문자열 형식의 설정 방법과 함수 형식의 설정 방법이 있다.

문자열 형식의 설정은 컬럼의 값을 사용하여 사칙연산의 간단한 계산식을 설정하는 경우에 사용하며, 계산식에 컬럼을 포함하는 경우 해당 컬럼의 Index 또는 SaveName을 기호 "|"로 감싸서 표현 한다. 사용 예제는 아래와 같다.

```
// SaveName이 "sBase"인 컬럼과 Index가 3인 컬럼의 합을 이용한 계산
"CalcLogic" : "(|sBase| + |3|) * 2"
```

함수 형식의 설정은 아래와 같이 세부 속성을 설정 한다.



Parameter	Type	필수여부	설 명
TriggerCols	String	선택	대상 컬럼의 Index 또는 SaveName 을 구분자 " "로 연결한 문자열
Function	Function	필수	계산식 처리 함수

Function 속성은 계산식 처리시 호출할 함수를 설정하며, 익명 또는 전역함수를 설정 할 수 있다. 함수 호출시 전달되는 인자는 아래와 같다.

Parameter	Type	설 명
Shet	Object	대상 시트 객체
Row	Number	대상 행의 Index
Col	Number	대상 컬럼의 Index

TriggerCols 속성을 설정하지 않은 경우 대상행의 컬럼의 데이터가 변경시 계산식 함수가 호출 되고, 특정 컬럼이 변경된 경우에만 계산식 함수를 호출 하는 경우 해당 컬럼을 TriggerCols에 설정 한다. 일반적으로 계산식 함수에서 사용하는 컬럼들의 정의하여 사용 한다. 사용예제는 아래와 같다.

```
// 익명함수 사용 방법
"CalcLogic": {
  "TriggerCols": "sBase",
  "Function": function(info) {
    var res = info.Sheet.GetCellValue(info.Row, "sBase") * 2;
    return res;
  }
}

// 전역 함수 사용 방법 (별도 정의)
function fnCalcSample(info) {
  var res = info.Sheet.GetCellValue(info.Row, "sBase") * 2;
  return res;
}
"CalcLogic": {
  "TriggerCols": "sBase",
  "Function": fnCalcSample
}
```


}

※ 계산식 함수는 모든 행에 대해서 대상컬럼이 변경되는 시점마다 호출되기 때문에 복잡한 로직을 사용하는 경우 성능 저하의 원인이 될 수 있습니다.

CaseSensitive는 필터링 처리시 영문의 대소구분 처리 여부를 설정한다.

Chart는 스파크라인 컬럼 타입 사용시 차트에 대한 세부 속성을 설정 한다. 이 속성은 스파크라인 컬럼 타입에서만 유효하며 세부 속성은 다음과 같다.

Parameter	Type	필수여부	설 명	
Type	String	선택	스파크라인 차트의 타입	
			설정값	설명
			Area	Area 차트 (Default)
			Line	Line 차트
			Column	Column 차트
			WinLose	승-패 차트
Data	String	필수	차트를 구성할 컬럼의 Index 또는 SaveName을 구분자 " "로 연결한 문자열	
Color	String	선택	양수 값(영역)에 대한 색상 값 (Default="#4f81bd")	
NeagtiveColor	String	선택	음수 값(영역)대 대한 색상 값 (Default="#c0504d")	

스파크라인 차트 사용시 ibchart.js 를 인클루드해야 사용 할수 있으며 IE 브라우저는 9이상에서만 지원 한다. (타 브라우저는 모두 지원)

lbcharts.js 를 인클루드 하지 않거나 지원하지 않는 브라우저에서는 차트가 표현되지 않고 빈 여백으로 표시 된다.

CheckSaveName은 트리구조에서 트리 기준컬럼에 체크박스를 사용시 체크박스의 값을 서버에 전달시 사용할 SaveName을 설정한다. 해당 SaveName이 설정되어 있는 경우에만 저장 관련 메소드 호출시 해당 체크값을 수집하게 된다. 이 속성은 해당 컬럼에 TreeCol, TreeCheck 가 모두 true로 설정되어 있는 경우에만 유효하다.



ClassName은 "Button" 컬럼타입의 버튼에 대한 사용자정의 클래스를 적용하고자 하는 경우 사용한다. 이 속성을 설정하지 않는 경우 기본 클래스로 적용되고, 설정 시 기본 Class의 Postfix로 적용된다. 메인테마 기준 기본 Class 및 속성 설정에 대한 Class는 아래와 같다.

구분	기본 Class 명	"Red"로 설정한 Class 명
기본 스타일	GMCellButtonBase	GMCellButtonBaseRed
활성화 상태의 스타일	GMCellButton	GMCellButtonRed
비활성화 상태의 스타일	GMCellButton0	GMCellButtonRed0
활성화의 Hover	GMCellButton: hover	GMCellButtonRed: hover
비활성화의 Hover	GMCellButton0: hover	GMCellButtonRed0: hover

해당 css Class는 반드시 해당 테마의 ibsheet.css 에 정의되어야 한다.

ColMerge는 데이터 컬럼에 대한 세로 머지 처리방법을 설정한다. 설정할 수 기본적으로있는 값은아래와 같으며, 설정이 없는 경우는 기본값 1로 처리 한다.

설정값	설명
0	셀 병합 사용 안함
1	셀 병합 사용 함
2	가상 셀 병합 방법으로 사용 함

셀병합은 기본적으로 table의 colspan, rowspan 속성을 이용하기 때문에, 범위가 큰 경우 대상 행을 모두 렌더링을 해야하는 문제로 성능이 저하 될 수 있다. 이런 경우 설정 값 2로 사용하면 실제 셀병합을 하지 않고, 가상으로 처리하는 방법을 사용하기 때문에 성능을 향상 시킬 수 있다.

ColSpan은 단위 데이터 행 구조에서 단위 데이터 행 범위 내에서 고정으로 적용 할 ColSpan 범위를 설정 한다. 이 속성 설정시 데이터의 동일 유무와 관계없이 셀 병합 처리를 하여 표현 된다. 이 속성은 단위 데이터 행 구조에서만 유효하며, 해당 속성을 사용하기 위해서는 MergeSheet 설정값을 3 또는 8로 설정 하여야 한다. (연관 속성 : RowSpan)

ComboCode는 Type이 "Combo" 또는 "ComboEdit"인 경우 저장될 코드 항목을 "|"로 연결하여 구성하며 위의 ComboText 항목의 개수와 동일 하여야 한다.

ComboFilter는 Type이 "ComboEdit"인 경우 목록에 대한 필터링 처리 방법을 설정 한다. 설정값에 따른 처리는 다음과 같다.

ComboFilter	설명
0	필터 기능 사용 안함
1	특정 부분 일치 형태의 필터링 모드
2	앞문자열 일치 형태의 필터링 모드

ComboText는 Type이 "Combo" 또는 "ComboEdit"인 경우 화면에 보여질 문자열 항목을 "|"로 연결하여 구성한다.

콤보리스트를 다중 컬럼 형태로 출력하려면 문자열을 "wt"로 연결하여 구성한다.

예) ComboText:"가wt나wt다|라wt마wt바"

Cursor는 해당 컬럼에 마우스 오버 시 커서의 모양을 설정한다.

"Default","Pointer" 로 설정이 가능하며 기본적으로 "Default"값이 설정된다.

DefaultValue는 신규입력시 기본값을 설정한다. 기본적으로 ""이 설정된다.

Edit는 데이터의 편집가능 여부를 설정한다. 기본적으로 1이 설정된다.

EditLen는 데이터의 입력 가능한 글자 수를 설정한다.

EditPointCount는 실수 형태의 포맷에서, 편집시 소수점 자리 수에 대한 입력 제한 개수를 설정 한다. 이 속성은 편집시에 대해서만 유효하다.

Ellipsis는 컬럼 텍스트의 표시글자가 잘릴 경우 말 줄임 여부를 설정한다. 기본적으로 0값이 설정된다

EmptyToReplaceChar는 데이터 타입이 "Int", "Float", "AutoSum"이고, 해당 셀의 값이 빈값일 때, 화면에 빈 값을 대체하여 표시할 문자 또는 문자열을 설정하는 기능이다. 화면 표시에만 해당 설정한 문자열로 보여지게 되고, 내부에서 처리되는 값은 빈값으로 처리되기 때문에 저장 또는 문서 연동시에는 해당 설정값으로 처리되지 않는다.

EnterMode는 다중라인 입력모드(MultiLineText)에서 Enter 키 입력에 대한 개행 처리 여부를 설정 한다. 이 속성은 MultiLineText 속성이 1(true)인 경우에만 유효하며, EditEnterBehavior 메소드 설정보다 우선 처리 된다.

ExceptKeys는 컬럼의 입력제외키를 설정한다.

"N" : 숫자 입력 불가

"E" : 영문 입력 불가

"N|E" : 숫자, 영문 입력 불가

"N|E|[사용자정의문자]" : 숫자, 영문과 사용자정의문자로 설정한 문자 입력 불가

예) "N|[abc!]" 설정 시 숫자와 a, b, c, ! 문자만 입력 불가

사용자 정의 문자열은 꼭 대괄호로 묶어서 설정해야 하며 사용자정의문자와 숫자,영문 설정문자의 합집합으로 처리된다.

AcceptKeys 와 ExceptKeys를 둘다 설정하고 설정키 내에 동일 문자가 있는 경우 해당 문자는 입력불가 처리된다.

ExcludeEmpty는 AutoSum 컬럼 타입인 경우 빈 값을 평균 또는 건수 계산시 포함할지 여부를 설정 한다. 설정 하지 않는 경우는 기본적으로 0값이 설정 (포함하여 계산)된다. 빈 값에 대한 비교이므로 해당 기능을 사용시에는 포맷 이 NullInteger 또는 NullFloat이 설정되어야 한다.

FitColWidth는 FitColWidth 메소드 호출시 현재의 컬럼 너비 비율에 맞게 너비를 재조정 처리를 허용할지 여부를 설정한다. 설정값이 1인 경우는 컬럼의 너비 비율에 맞게 너비가 재조정되며, 설정값이 0인 경우는 현재의 컬럼 너비를 그대로 유지 한다. 이 속성은 FitColWidth 메소드의 인자를 설정하지 않고 호출하는 경우에만 적용된다.

FontColor는 컬럼의 폰트색상을 설정한다.

Format은 데이터에 대한 Mask 적용 형태를 설정하는 것으로 다음과 같은 값을 설정한다. 포맷 설정은 기본제공 외에 사용자가 직접 정의 해서 설정 할

수 있으며 ((ex) Format="#,###,#0") Type 값에 따라 설정한 Format이 적용 된다.

Format	설명	사용가능타입
Ymd	"년월일" 형태. lbmsg의 SYS_d 포맷을 따른다	Date,Text,Popup
Ym	"년월" 형태, lbmsg의 SYS_m 포맷을 따른다.	Date,Text,Popup
Md	"월일" 형태. lbmsg의 SYS_M 포맷을 따른다.	Date,Text,Popup
Hms	"시분초" 형태. lbmsg의 SYS_T 포맷을 따른다.	Date,Text,Popup
Hm	"시분" 형태. lbmsg의 SYS_t 포맷을 따른다.	Date,Text,Popup
YmdHms	"년월일시분초" 형태. lbmsg의 SYS_G 포맷을 따른다	Date,Text,Popup
YmdHm	"년월일시분" 형태. lbmsg의 SYS_g 포맷을 따른다	Date,Text,Popup
Integer	정수 형태, 기본 0	Int,AutoSum,AutoAvg
NullInteger	널 정수 형태, 기본 널	Int,AutoSum,AutoAvg
Float	실수 형태, 기본 0	Float,AutoSum,AutoAvg
NullFloat	널 실수 형태, 기본 널	Float,AutoSum,AutoAvg
IdNo	주민등록번호 형태	Text
SaupNo	사업자번호 형태	Text
PostNo	우편번호 형태 (6자리, 5자리 지원)	Text
CardNo	카드번호 형태	Text
PhoneNo	전화번호 형태	Text
Number	숫자 형태 (특정 포맷없이 숫자키만 입력 받는 형태)	Text

※ 사용자 숫자 포맷 가이드



"#", "0", ",", "." 조합으로 기본제공 포맷 이외의 포맷을 정의하여 사용 할 수 있다. 사용가능한 타입은 Int, Float, AutoSum, AutoAvg 타입이다.
사용방법은 아래 표와 같다.

구분	기호	설명
숫자	#	해당위치에 값이 있는 경우 해당 값을 표시하고, 없는 경우는 빈값으로 표시한다. 소수점 이하에서 사용하는 경우는 표시할 자리수를 결정하며 입력된 값의 자리수가 #의 개수보다 많은 경우 반올림하여 개수만큼 표시한다. 예) Input= 12345.678, Format=#,###.##,Output=12,345.68 Input= 0.789 Format=#,###.##,Output=0.79
	0	해당위치에 값이 있는 경우 해당 값을 표시하고, 없는 경우는 0으로 표시한다. 소수점 이하에서 "#"과 동일하게 동작한다. 예) Input=123456.7 Format=#,##0.00 Result=123,456.70 Input= 0.7 Format=#,##0.00 Result=0.70
소수점	.	소수점이 있는 경우 실수표현이 가능하며 소수점을 표현하는 경우 ibmsg의 SYS_DecimalSeparator에 설정되어 있는 값으로 표현된다.
천단위구분자	,	천단위 구분자를 넣은 경우 천단위 마다 구분자를 표시하며 ibmsg 의 SYS_GroupSeparator에 설정되어 있는 값으로 표현된다. 또한, 소수점 바로 앞에 사용하는 경우는 콤마 개수만큼 1000으로 나눈 값으로 표현한다. 예) Input= 123456.789 Format=0,.00 Result=123.46 Input= 123456.789 Format=0,,.00 Result=0.12
퍼센트	%%	마지막에 퍼센트 기호를 넣는 경우 자동으로 %를 표현

	%	한다. 예) Input=123.456 Format=#,##0.00WW% Result=123.46%
Indian Numbering 표현		Indian Numbering 포맷은 1,00,00,000 과 같이 천단위 구분자 이상에서는 두자리 마다 구분자를 넣는 방식을 사용한다. 이와 같은 포맷 설정시 "#,##,###" 와 같이 설정하여 사용 할 수 있다. 예) Input=123456 Format=#,##,### Result=1,23,456 Input=12345678 Format=#,##,### Result=1,23,45,678

※ 사용자 문자 포맷 가이드

"#", "*", 그외 조합으로 기본제공 포맷 이외의 포맷을 정의 하여 사용 할 수 있다. 사용가능한 타입은 Text타입이다.

사용방법은 아래 표와 같다.

구분	기호	설명
문자	#	해당위치에 값이 있는 경우 해당 값을 표시하고, 없는 경우는 빈값으로 표시한다. 예) Input=7907211022553, Format=#####-****###, Result=790721-****553
	*	
	그외	#,*이외의 문자 숫자에 대해선 그대로 표시한다.

※ "Text", "Popup" 컬럼 타입에서의 다중 포맷 가이드

한 컬럼내에 2개 이상의 포맷을 설정하고자 하는 경우 Format 속성에 해당하는 포맷 문자열을 배열 형식으로 설정한다. 이와 같이 설정하는 경우 입력된 문자열의 길이에 따라 해당 길이와 일치하는 포맷 문자열이 적용되며, 일치하는 포맷 문자열이 없는 경우 가장 첫번째 설정한 포맷 문자열을 기준으로 설정 된다.



편집 시 문자열 입력 길이와 같이 적용 포맷이 명확하지 않는 경우 배열의 가장 첫번째 설정한 포맷 기준으로 처리 되므로, 길이가 가장 긴 포맷을 첫번째에 설정을 권장한다.

(예) 주민번호, 사업자번호 포맷 설정 : Format : ["IdNo", "SaupNo"]

"1234567890123" → "123456-7890123" : "IdNo" 적용

"1234567890" → "123-45-67890" : "SaupNo" 적용

"1234567" → "123456-7" : "IdNo" 적용

FormatFix는 GetCellValue 시 GetCellText 값을 반환 할지 여부를 설정한다. 특히 True 로 설정시 저장할 때 Format 이 설정된 상태로 저장한다. 기본적으로 0값이 설정된다.

FullInput은 컬럼타입이 단일 행 문자열인 경우 EditLen 만큼 모두 입력해야 하는 경우 설정한다. 기본적으로 0값이 설정된다.

GroupSumType은 그룹행 설정에 의한 데이터 구성시 부모행에 표현할 계산 방식을 설정한다. 합계 또는 건수 계산시 모든 자식행을 포함하여 계산한다. 이 속성은 ShowGroupRow 메소드를 통한 그룹행 사용시에만 유효한 속성이다. 설정값은 대소구분없이 설정이 가능하며, 설정할 수 있는 값은 아래와 같다.

설정값	설명
""	별도 계산 처리 없음 (Default)
"Sum"	합계 계산 처리
"Count"	건수 계산 처리

HeaderCheck는 컬럼에 헤더의 전체체크 표시 여부를 설정한다. 기본적으로 1값이 설정된다.

Hidden은 데이터 컬럼 숨김 여부를 설정한다.

HoverUnderline은 마우스 오버 시 셀 값에 언더라인 처리 여부를 설정한다. 기본적으로 0값이 설정된다.

Image는 컬럼에 이미지를 같이 표현하는 경우에 설정한다. 컬럼타입이 "Text", "Int", "Float", "AutoSum", "AutoAvg" 인 경우에 사용할 수 있다. 여기에서 설정하는 이미지는 컬럼에 대한 기본 이미지이며 조회 데이터 구조에서 셀 별로 설정이 가능하다.

ImgAlign는 Image 속성을 사용한 경우 이미지의 위치를 설정한다. "Left"로 설정하는 경우 텍스트의 왼쪽에 이미지가 표시되고 "Right"로 설정하는 경우 텍스트의 오른쪽에 이미지가 표시된다. 설정을 하지 않는 경우 기본값은 "Left"이다.

ImgHeight는 Type이 "Image" 이거나 "Image" 속성을 사용하는 경우 표시되는 이미지의 높이 값을 설정한다. 기본적으로 0값이 설정되며 0으로 설정하는 경우 표시되는 이미지는 원본 사이즈로 표시된다

ImgWidth는 Type이 "Image" 이거나 "Image" 속성을 사용하는 경우 표시되는 이미지의 너비 값을 설정한다. 기본적으로 0값이 설정되며 0으로 설정하는 경우 표시되는 이미지는 원본 사이즈로 표시된다.

InsertEdit는 트랜잭션 상태가 입력인 데이터에 해서 Edit 가능 여부를 설정하며, 기본적으로 1값이 설정된다.

ItemCode는 한 셀안에 2개이상의 체크박스(멀티 체크) 구성시 해당 아이템별 코드를 구분자 "|"로 연결한 문자열로 설정한다. 해당 속성은 컬럼타입이 "CheckBox", "DummyCheck"에서만 사용이 가능 하다.

ItemText는 한 셀안에 2개이상의 체크박스(멀티 체크) 구성시 해당 아이템별 텍스트(라벨)을 구분자 "|"로 연결한 문자열로 설정한다. 해당 속성은 컬럼타입이 "CheckBox", "DummyCheck"에서만 사용이 가능 하다.

KeyField는 데이터의 필수입력 항목 여부를 설정하는 것으로 1인 경우 저장 함수 호출 시 셀에 데이터가 없는 경우 경고 메시지를 표시하고, Edit 하도록 유도한다. 기본적으로 0값이 설정된다.



LevelSaveName은 트리 형태의 조회인 경우 데이터를 저장할 때 사용하는 트리 레벨의 변수 명을 설정한다. 변수 명을 설정하지 않을 경우 해당 행의 트리 레벨은 서버로 전달되지 않는다.

MaxCheck는 멀티 체크를 구성시 최대 허용 체크 개수를 설정한다. 설정값이 0인 경우는 최대 허용에 대한 제한을 두지 않는다. 해당 속성은 "CheckBox", "DummyCheck" 컬럼타입에 ItemText 속성이 설정된 경우에만 유효 하다.

MaximumValue는 Format이 숫자와 관련된 포맷인 Integer, Float, NullInteger, NullFloat 일 때, 편집 시 입력할 수 있는 최대값을 설정한다.

MinimumValue는 Format이 숫자와 관련된 포맷인 Integer, Float, NullInteger, NullFloat 일 때, 편집 시 입력할 수 있는 최소값을 설정한다.

MultiLineText는 데이터타입이 "Text"인 경우 다중라인 입력여부를 설정한다. 기본적으로 0값이 설정된다.

NumberSort는 데이터타입이 "Text", "Popup" 인 경우 Sort 시 문자형 정렬방식으로 처리하게 되는데, 이를 숫자형 정렬방식으로 사용하고자 하는 경우에 이 속성을 사용한다. 설정하지 않은 경우 기본값은 0으로 문자형 정렬방식으로 처리 된다.

PointCount는 컬럼타입이 Float 인 경우 소수점 이하의 자리 수를 설정한다. 설정하지 않은 경우 Format 설정 값을 따르며 설정한 경우 Format 설정 값은 무시되고 재설정 된다.

PopupButton 속성은 대상 셀에 팝업 버튼 사용여부에 대한 컬럼단위로 설정할 수 있는 속성이다. "Text", "Int", "Float", "AutoSum", "AutoAvg" 컬럼타입에서 설정 할 수 있으며, 별도 설정이 없는 경우 기본값인 0(false)로 처리 된다.

PopupCheckEdit는 컬럼팝업을 설정 했을 경우 선택한 행의 데이터 편집가능 여부를 확인하여 편집 가능한 데이터만 변경하는 경우는 1로 설정한다. 기본적으로 0값이 설정된다.

PopupCode는 PopupText에 매핑되는 코드값을 "|"로 연결하여 구성하며 PopupText 항목의 개수와 동일하여야 한다. 이 속성을 설정하는 경우에는 팝업 메뉴에서 항목을 선택 시 해당 컬럼의 옆 컬럼에 매핑되는 코드 값이 설정된다.

PopupText는 마우스 오른쪽 버튼 클릭시 보여지는 팝업 메뉴 목록을 설정할 항목을 "|"로 연결하여 구성한다.

Radiolcon은 CheckBox, DummyCheck, DelCheck, Radio 컬럼타입에 대하여 데이터 셀에 표현되는 이미지 모양을 설정한다. 1로 설정하는 경우에는 라디오 버튼 모양으로 표시되고 0으로 설정하는 경우에는 체크박스 모양으로 표시된다. 기본값은 Radio 컬럼타입은 1 그외의 컬럼타입은 0이다.

RowMerge는 데이터 컬럼에 대한 가로 머지 허용 여부를 설정한다. 기본적으로 1값이 설정된다.

RowSpan은 단위 데이터 행 구조에서 단위 데이터 행 범위 내에서 고정으로 적용 할 RowSpan 범위를 설정 한다. 이 속성 설정시 데이터의 동일 유무와 관계없이 셀 병합 처리를 하여 표현 된다. 이 속성은 단위 데이터 행 구조에서 만 유효하며, 해당 속성을 사용하기 위해서는 MergeSheet 설정값을 3 또는 8로 설정 하여야 한다. (연관 속성 : ColSpan)

Save 속성은 저장 또는 저장 관련 메소드에서 해당 컬럼의 값을 포함하여 서버로 전달 또는 추출할지 여부를 설정 한다. 일반적으로 **SaveName** 속성을 정의 해야하고, 저장시 서버로 전달이 필요 없는 경우 이 속성값을 0(false)로 설정 한다. (**SaveName** 속성을 설정하지 않으면 기본적으로 전달하지 않는다.) 이 속성은 **SaveName** 속성이 설정되어 있는 경우에만 유효하게 동작 한다.

SaveName은 데이터를 저장할 때 사용하는 변수 명을 설정한다. 변수 명을 설정하지 않을 경우 기본적으로 컬럼 순서대로 C1, C2, ... 로 설정된다.

ShowCol 속성은 ComboText에서 다중컬럼 형태의 콤보를 사용할때 콤보 선택시 보여지는 기준 컬럼을 설정 한다. 설정하지 않은 경우 ComboText에 처음 문자열이 보여진다.

Sort는 헤더 셀을 클릭 시 소트 처리 가능 여부를 설정한다. 기본적으로 1값이 설정된다.

StaticPassword는 Pass 타입에서 ibmsg의 SYS_Password 값으로 고정표현할지 데이터 길이에 따라 '*'를 표현할지 여부를 선택한다. 기본속성은 1이며 고정된 값을 표현한다.

SumType은 AutoSum 컬럼에 대한 합계 계산 방식을 설정한다. 설정값에 따라 합계, 평균, 건수, 최대값, 최소값을 계산하여 합계행에 표시 한다. 컬럼 중 AutoSum 컬럼이 1개 이상인 경우에 합계행은 설정에 따라 상단 또는 하단에 표시된다. 해당 속성의 설정값은 대소구분없이 설정이 가능하며 설정에 따른 계산방식은 다음과 같다.

설정값	설명
"Sum"	합계 계산 처리 (Default)
"Avg"	평균 계산 처리
"Count"	건수 계산 처리
"Max"	최대값 계산 처리
"Min"	최소값 계산 처리

ToolTip은 데이터 영역에 풍선도움말 사용 여부를 설정한다. Boolean 으로 설정한 경우 사용함/사용안함으로 설정되고, 설정인 경우 셀의 데이터와 동일한 내용의 풍선도움말을 표시 한다. 문자열로 설정한 경우 해당 문자열로 풍선도움말을 표시 한다.

ToolTipText는 헤더 행의 풍선도움말에 표시할 문자열을 설정한다. 기본적으로 빈 값이 설정된다.

Transaction은 Status 컬럼이 있는 경우 해당 셀의 값 변경시 트랜잭션 관리 허용여부를 설정 한다. 해당 속성의 설정을 0으로 설정시 해당 셀의 값을 변경하더라도 Status 컬럼의 트랜잭션 상태가 변경되지 않는다.

TreeCheck는 트리 기준컬럼에 체크박스 사용여부에 대한 설정이다. TreeCol이 설정되어있는 컬럼에서만 해당 속성이 유효하고, 1(true)로 설정시 트리기준 컬럼의 셀별 좌측에 체크박스가 표시 된다.

TreeCol은 트리형태 조회의 경우 기준컬럼 여부를 설정한다.

TrueValue, FalseValue는 CheckBox 형태인 컬럼에서 기존에 지원하던 0,1 이외의 지정한 값을 사용할 수 있도록 사용자가 지정할 수 있게 하는 기능이다. 예를 들어 TrueValue:"T", FalseValue:"F"로 지정하게 되면 1 대신 "T", 0 대신 "F" 를 사용하여 값을 나타낼 수 있다.

UpdateEdit는 트랜잭션 상태가 조회인 데이터에 대해 Edit 가능 여부를 설정하며, 기본적으로 1값이 설정된다.

Width는 컬럼에 대한 픽셀 단위 너비 설정으로 설정하지 않을 경우 헤더 Text의 너비에 맞게 자동 설정된다.

Wrap는 컬럼너비에 따라 자동 줄 바꿈 여부를 설정한다. 기본적으로 0값이 설정된다.

VAlign는 컬럼 세로 정렬 기능을 설정한다. "Top"으로 설정할 경우 텍스트 및 이미지 등을 최상단으로 정렬하고, "Bottom"으로 설정할 경우 최하단으로 정렬한다. 설정을 하지 않는 경우 기본값은 "Middle"로 중앙으로 정렬된다.

ZeroToReplaceChar는 데이터 타입이 "Int", "Float", "AutoSum"이고, 해당 셀의 값이 0일 때, 화면에 0값을 대체하여 표시할 문자 또는 문자열을 설정하는 기능이다. 화면 표시에만 해당 설정한 문자열로 보여지게 되고, 내부에서 처리되는 값은 0으로 처리되기 때문에 저장 또는 문서 연동시에는 해당 설정값으로 처리되지 않는다.

➤ **Example**

```
//각 컬럼의 기능을 설정한다.  
var cols = [
```



```
{Type:"Status",Width:60,SaveName:"sStatus",Align:"Center"},
{Type:"DelCheck",Width:60,SaveName:"sDelete",Align:"Center"},
{Type:"Text",Width:100,SaveName:"JOB",Align:"Center"},
{Type:"Combo",Width:100,SaveName:"DEPTNO",Align:"Center"
,ComboText:comboDataArr[0],ComboCode:comboDataArr[1]},
{Type:"Text",Width:60,SaveName:"EMPNO",Align:"Center"},
{Type:"Text",Width:150,SaveName:"ENAME",Align:""},
{Type:"Date",Width:120,SaveName:"HIREDATE",Format:"Ymd"
,Align:"Center",EditLen:8},
{Type:"Text",Width:120,SaveName:"MGR",Align:"Center"},
{Type:"Int",Width:120,SaveName:"SAL",Align:"Right",Format:"NullInteger"},
{Type:"Int",Width:60,SaveName:"COMM",Align:"Right",Format:"Integer"}
];
mySheet.InitColumns(cols);
```

➤ 제공 버전

7.0.0.0	
7.0.1.0	1. ComboFilter 속성에 대한 앞문자열 일치 형태의 필터링 모드 추가 2. ComboEdit 컬럼 Validation 설정 추가
7.0.4.4	다중라인 입력모드에 대한 EnterMode 속성 추가
7.0.7.2	팝업버튼 사용여부 설정을 위한 PopupButton 속성 추가
7.0.9.0	TreeCheck, CheckSaveName 속성 추가
7.0.10.0	GroupSumType 속성 추가
7.0.11.0	스파크라인 컬럼타입 추가, Chart 속성 추가
7.0.12.0	SumType 속성 추가, ExcludeEmpty 속성 적용범위 확장(건수) AutoAvg 컬럼타입 삭제
7.0.12.2	RowMerge, FitColWidth 속성 추가
7.0.12.3	ItemCode, ItemText, MaxCheck 속성 추가
7.0.13.3	Transaction 속성 추가
7.0.13.5	"Text", "Popup" 컬럼타입에서의 다중 포맷 설정 지원 "PostNo" 포맷에 대한 기존 6자리, 신규 5자리 포맷지원



7.0.13.9	"Button" 컬럼타입 추가, ClassName 속성 추가
7.0.13.13	EditPointCount 속성 추가
7.0.13.16	ActionMenu 속성 추가
7.0.13.24	EmptyToReplaceChar, ZeroToReplaceChar 속성 추가
7.0.13.29	Save 속성 추가 CalcLogic 속성 개선 (Object 타입 설정 기능 추가)
7.0.13.34	NumberSort 속성 추가
7.0.13.50	ColMerge 속성 개선 (가상 셀병합 모드 추가, 타입변경)
7.0.13.53	StaticPassword 속성 추가



InitComboNoMatchText Method

➤ 기능

Combo 항목을 표시 시, Combo 항목에 없는 항목을 설정하거나 조회 하였을 경우 보여지는 글자를 설정한다.

Show 인자를 1(true)로 설정 하면 조회데이터의 내용이 ComboCode에 해당하는 값이 존재하지 않는 경우 ShowText 인자에 설정한 값이 보여진다.

InsertItem 인자를 1(true)로 설정할 경우, InitColumns 에서 설정한 적이 없는 ComboText 또는 ComboCode 값이 조회되면 그 값으로 아이템을 추가한다.

➤ Syntax

Syntax	ObjId.InitComboNoMatchText(Show, [ShowText], [InsertItem])
--------	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Show	Boolean	필수	콤보 항목 이외의 글자를 보여줄지 여부를 설정
ShowText	String	선택	콤보 항목 이외인 경우 표시할 글자, Default = ""
InsertItem	Boolean	선택	콤보 항목 이외의 값이 오는 경우 그 값을 해당 셀에 아이템으로 삽입한다. Default=0

➤ Example

//Combo 항목이 없는 경우 항목 없음 표시하기 mySheet.InitComboNoMatchText(1, "항목없음");
--

➤ 제공 버전

7.0.0.0

InitHeaders Method

➤ 기능

헤더에 들어갈 타이틀과 헤더기능을 지정한다.

➤ Syntax

Syntax	ObjId.InitHeaders(Headers, [Info])
--------	------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Headers	Object[]	필수	헤더의 행별 정보를 설정
Info	Object	선택	헤더의 공통 정보를 설정

세부설정

설정변수		타입	설명	
Headers	Text	String	헤더에 표시할 Text를 " "로 연결한 문자열	
	Align	String	헤더 텍스트의 정렬방법 Default="Center"	
	RowMerge	Boolean	헤더행의 가로머지 허용 여부 Default=1	
Info	Sort	Number	헤더 클릭을 통한 컬럼 Sort 허용 여부	
			설정값	설명
			0	사용 안함
			1	Sort 기능 사용 (Default)
			2	Sort 아이콘만 표시
	3	colSpan 설정이 아닌 헤더 셀만 Sort 기능 사용		
	ColMove	Boolean	헤더 컬럼 이동 가능 여부 Default=1	



	ColResize	Boolean	컬럼 너비 조정 허용 여부 Default=1
	HeaderCheck	Boolean	헤더에 전체 체크 표시 여부 Default=1

➤ **Example**

```
//2줄 헤더타이틀을 설정한다.
var headers = [
{Text:"상태|삭제|직원정보|직원정보|직원정보", Align:" Center"} ,
{Text:"상태|삭제|직원명 |직원번호|입사일", Align:" Center"}
];
var info = {Sort:1, ColMove:1, ColResize:0, HeaderCheck:0};
mySheet.InitHeaders(headers,info);
```

➤ **제공 버전**

7.0.0.0	
7.0.4.3	헤더행 별 RowMerge 속성 추가
7.0.4.6	Sort 속성 개선 (아이콘만 표시하는 기능 추가)
7.0.13.24	Sort 속성 개선 (colSpan 에 대한 예외 기능 추가)



IsDataModified Method

➤ 기능

데이터 행의 트랜잭션 여부를 확인한다.

트랜잭션 상태가 "조회" 이외의 데이터가 한건이라도 있으면 1을 반환하고, 없으면 0을 반환한다.

➤ Syntax

Syntax	ObjId.IsDataModified()
--------	------------------------

➤ Info

Return	Boolean, 데이터 행의 트랜잭션 여부		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// 트랜잭션이 발생하지 않은 경우 메시지 표시 후 저장 취소 처리
if(!mySheet.IsDataModified()) {
    alert("저장할 내역이 없습니다. 저장은 취소합니다.");
// 트랜잭션이 발생한 경우 저장 처리
} else {
    mySheet.DoSave("sheet_save.jsp");
}
```

➤ 제공 버전

7.0.0.0



IsFocused Method

➤ 기능

시트의 포커스 상태를 확인 한다.

➤ Syntax

Syntax	ObjId.IsFocused()
--------	-------------------

➤ Info

Return	Boolean, 포커스 상태.		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// 시트의 Focus 여부 확인  
console.log("focused", mySheet.IsFocused());
```

➤ 제공 버전

7.0.13.51	
-----------	--

IsHaveChild Method

➤ 기능

트리형 데이터에서 특정 행에 Child 레벨의 행이 존재하는지 여부를 확인한다.
Child 레벨이 존재할 때 1를 반환하고, 존재하지 않을 때 0를 반환한다.
IncludeDelRow 인자는 Row 인자에 설정된 행 번호 아래에 존재하는 Child 행 중 "삭제"상태의 행을 포함 할것인가 여부를 의미하며, 1인 경우 "삭제" Child 행을 포함하고, 0 인경우 포함하지 않는다. Default=0이다.

➤ Syntax

Syntax	ObjId. IsHaveChild (Row, [IncludeDelRow])
--------	--

➤ Info

Return	Boolean, Child행의 존재 여부.		
Parameter	Type	필수여부	설 명
Row	Long	필수	특정 행의 Row Index
IncludeDelRow	Boolean	선택	Child 레벨의 행 중 "삭제" 상태의 행을 포함할것인가 여부, Default=0

➤ Example

```
//4행의 Child 존재 여부를 확인한다.
If(mySheet.IsHaveChild(4)) {
    alert("4행에 자식 레벨이 존재합니다.");
}else{
    alert("4행에 자식 레벨이 존재하지 않습니다.");
}
//4행의 삭제된 Child도 포함하여 확인한다.
var bResult = mySheet.IsHaveChild(4, 1);
```

➤ 제공 버전



7.0.0.0

KeyFieldImage Method

➤ 기능

필수 입력 항목에 대한 이미지 파일의 위치를 설정하거나 확인한다.

이 속성은 기본적으로 제공하는 필수 입력 항목 이미지를 사용자가 원하는 이미지로 변경설정 가능하기 위한 속성이다.

➤ Syntax

Syntax	Get	ObjId. GetKeyFieldImage()
--------	-----	----------------------------------

➤ Info

Return	String, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// 필수 입력 항목 이미지를 확인한다.
mySheet.GetKeyFieldImage();
```

➤ Syntax

Syntax	Set	ObjId. SetKeyFieldImage (imgUrl)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
imgUrl	String	필수	설정할 Url

➤ Example

```
// 필수 입력 항목 이미지를 변경한다.
mySheet.SetKeyFieldImage("/sheet/imgKeyField.gif");
```



- 제공 버전
7.0.0.0



LastCol Method

➤ 기능

마지막 컬럼의 index를 반환한다.

➤ Syntax

Syntax	ObjId.LastCol()
--------	-----------------

➤ Info

Return	Long, 마지막 컬럼 인덱스		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
//마지막 컬럼의 index를 확인한다.  
Alert("마지막 컬럼의 index는 " + mySheet.LastCol() + " 번 입니다.");
```

➤ 제공 버전

7.0.0.0



LastRow Method

➤ 기능

마지막 행의 index를 반환한다.

Mode 인자를 설정하지 않거나 0으로 설정하는 경우 데이터만의 마지막 행의 인덱스도 아니고, 화면에 보여지는 마지막 행의 인덱스도 아닌 마지막 행의 인덱스를 반환한다.

마지막 행은 합계 행일 수도 있고, 데이터 행일 수도 있고, 헤더 행일 수도 있으므로 주의 하여 사용한다.

화면에 보여지는 마지막 행의 인덱스를 반환받기 원하는 경우 Mode 인자를 1로 설정한다.

➤ Syntax

Syntax	ObjId.LastRow([Mode=0])
--------	-------------------------

➤ Info

Return	Long, 마지막행의 Index	
Parameter	Type	설 명
Mode	Integer	마지막행 반환 모드 0 : 마지막 행인덱스 반환 (기본값, 기존동일) 1 : 현재 화면상의 마지막 행 Index 반환

➤ Example

//마지막 행의 인덱스를 확인한다. Alert("마지막 행의 index는 " + mySheet.LastRow() + " 번 입니다."); // 화면에 보여지는 마지막 행의 인덱스를 확인한다. Alert("화면상 마지막 행의 index는 " + mySheet.LastRow(1) + " 번 입니다.");

➤ 제공 버전

7.0.0.0

LeftCol Method

➤ 기능

가장 좌측에 표시되는 컬럼을 확인하거나 설정한다. 틀고정 컬럼이 있을 경우 고정 컬럼 이후 가장 좌측에 표시 된다. 이 속성의 설정으로 포커스는 변경되지 않지만 가장 좌측에 표시되는 컬럼이 바뀌므로 자동으로 수평 스크롤이 움직이게 된다. LeftCol은 Hidden 컬럼을 포함하여 0 컬럼을 기준으로 설정된다.

➤ Syntax

Syntax	Get	ObjId. GetLeftCol()
--------	-----	----------------------------

➤ Info

Return	Long, 가장좌측 끝 컬럼 인덱스 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//컬럼으로 설정됨을 확인
mySheet.GetLeftCol();
```

➤ Syntax

Syntax	Set	ObjId. SetLeftCol (Col)
--------	-----	--------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Col	Long/ String	필수	특정컬럼의 Column Index 또는 SaveName

➤ Example



```
//고정컬럼이 설정되어 있을경우
//4컬럼을 수평스크롤 가장 좌측에 보이는 컬럼으로 설정
mySheet.SetLeftCol(4);

//고정컬럼이 설정되어 있지 않을경우
//6컬럼을 수평스크롤 가장 좌측에 보이는 컬럼으로 설정
mySheet.SetLeftCol(6);
```

➤ **제공 버전**

7.0.0.0

LoadSaveData Method

➤ 기능

저장 데이터를 함수의 인자로 전달 받아 결과를 IBSheet 내부에 처리한다.

이 함수는 보안 모듈이 사용 될 경우 복호화된 저장 데이터를 읽어 들이기 위한 것이다. 저장 데이터는 GetSaveData 함수를 통해 읽어 들일 수 있고, 이 함수에 인자로 설정하면 처리 결과를 표현하고, OnSaveEnd 이벤트를 발생한다.

➤ Syntax

Syntax	ObjId.LoadSaveData(Content, [Opt])
--------	------------------------------------

➤ Info

Return	None.		
Parameter	Type	필수여부	설 명
Content	String	필수	저장XML 또는 저장JSON 문자열
Opt.CallBack	Function	선택	CallBack 함수
Opt.Event	Boolean	선택	완료 이벤트 발생 여부 Default = 1

➤ Example

<pre>//저장 작업 Ajax 처리 var rtnData = mySheet.GetSaveData("Action.do",param); //저장 결과 표현하기 mySheet.LoadSaveData(rtnData);</pre>

➤ 제공 버전

7.0.0.0	
7.0.13.43	CallBack, Event 속성 추가



LoadSearchChildData Method

➤ 기능

DoSearchChild 메소드의 기능을 url이 아닌 객체 또는 문자열로 처리하는 기능으로 트리구조에서 자식노드 조회 데이터(xml or json)를 함수의 인자로 전달 받아 아이비시트에 로딩한다. 이 함수는 보안 모듈이 사용 될 경우 복호화된 조회 데이터를 읽어 들이기 위한 것이다. 조회 데이터는 GetSearchData 함수를 통해 서버로부터 가져올 수 있고, 가져온 데이터 문자열을 이 함수의 인자로 설정하면 조회 데이터를 아이비시트에 로딩하고, OnSearchEnd 이벤트를 발생한다.

➤ Syntax

Syntax	ObjId. LoadSearchChildData (Row, Data, [Opt])
--------	--

➤ Info

Return	None.		
Parameter	Type	필수여부	설 명
Row	Long	필수	대상 부모의 행 인덱스
Data	String / Object	필수	조회 데이터 문자열 또는 JSON 객체
Opt	Object	선택	옵션 인자 기능을 json 형태로 설정한다.

Opt 옵션 인자 속성은 다음과 같다.

WaitDlg	Boolean	대기이미지 표시 여부 (Default=1)
Append	Boolean	기존 자식 데이터에 Append 여부 (Default=0)

➤ Example

//조회 데이터 읽어오기 var Row = mySheet.GetSelectRow();
--



```
var Data;  
Data=mySheet.GetSearchData('list2_Child_Json.jsp',  
'ChildLevel='+ (parseInt(mySheet.GetRowLevel(Row))+1));  
var Opt = {Append: 1, WaitDlg: 1};  
  
//조회 결과 내용을 표현하기  
mySheet.LoadSearchChildData(Row, Data, Opt);
```

➤ **제공 버전**

7.0.0.0

LoadSearchData Method

➤ 기능

조회 데이터(xml or json)를 함수의 인자로 전달 받아 아이비시트에 로딩한다.

조회 데이터는 GetSearchData 함수나 혹은 별도의 ajax 모듈을 통해 서버로부터 가져올 수 있고, 가져온 데이터 문자열을 이 함수의 인자로 설정하면 조회 데이터를 아이비시트에 로딩하고, OnSearchEnd 이벤트를 발생한다.

기본적으로 이 함수가 호출되면 기존 데이터는 지워지고 Content데이터가 로딩된다.

하지만 Append 인자를 사용하는 경우, 기존데이터를 맨 아래 행에 Content데이터를 붙여넣는다.

AppendRow 인자는 Append값이 true 인 경우, Content데이터를 붙여넣을 행을 지정할 수 있다.

Fx 인자는 컬럼의 타입에 따라 사용할 수 있는 포맷 유형을 제한하고 대신 조회속도를 향상 시킨다. Fx에서 사용할 수 있는 포맷 형식에 관한 자세한 내용은 이 메뉴얼 마지막 장 Appendix 부분에 설명되어 있다.

➤ Syntax

Syntax	ObjId. LoadSearchData (Content, [Opt])
--------	---

➤ Info

Return	None.		
Parameter	Type	필수여부	설 명
Content	String	필수	조회XML 또는 조회JSON 문자열
Opt.Append	Boolean	선택	Append 조회 여부, Default=0
Opt.AppendRow	Number	선택	Append 조회 시 데이터를 붙여넣을 위치 설정
Opt.CalBack	Function	선택	CallBack 함수
Opt.Event	Boolean	선택	완료 이벤트 발생 여부



			Default = 1
Opt.Fx	Boolean	선택	포맷팅된 데이터 조회 처리 여부, Default=0
Opt.Sync	Boolean	선택	동기 조회 여부 이 함수는 기본적으로 비동기 형태로 동작한다. Default=0
Opt.Wait	Boolean	선택	대기 이미지 표시 여부 동기 조회인 경우에는 설정값에 상관없이 대기 이미지를 표시하지 않는다. Default=1,

➤ **Example**

```
//1. 조회된 데이터가 없음을 로딩
var data = {Data:[]};
mySheet.LoadSearchData(data);

//2. 서버로부터 데이터를 가져와 로딩
//조회 데이터 읽어오기
var sXml = mySheet.GetSearchData(" list.jsp","sa_nm=chris&sa_no=2007128");

//조회 결과 내용을 표현하기
mySheet.LoadSearchData(sXml);

// 조회 결과 내용을 기존데이터에 Append 하여 표현하기
var opt = { Append : 1 };
mySheet.LoadSearchData(sXml, opt);
```

➤ **제공 버전**

7.0.0.0	
7.0.5.4	Fx:2 옵션 추가



7.0.6.0	대기이미지 표시여부 속성 추가 (Wait)
7.0.13.43	CallBack, Event 속성 추가

LoadExcel Method

➤ 기능

엑셀로부터 데이터를 읽어 들여 데이터 영역에 표시한다.

전달되는 인자의 형식은 JSON 타입으로 설정 하고자 하는 정보를 JSON 형태로 구성하여 전달한다.

Append 속성을 1로 설정하는 경우 기존 데이터에 현재 엑셀 데이터를 첨부하여 로드 할 수 있다.

ColumnMapping 속성은 엑셀의 특정 컬럼들만 IBSheet의 첫컬럼부터 차례대로 로딩할 때에 사용한다. 기본값은 널. 이 값이 널이 아니면, Mode 가 HeaderMatch 값을 유지하고 있어도 무시되고 이 기능이 우선적으로 처리된다.

FileExt 속성은 업로드 가능한 파일 확장자를 지정한다. 확장자는 "."을 제외한 "xls" 등으로 입력해야 하며, 기본값은 ""으로 "xls|xlsx"로 입력한 것과 동일하다. 여러 개의 확장자는 "|"로 구분하여 입력한다.

MaxFileSize 속성은 처리 대상 파일의 사이즈를 제한하고자 할 때 사용한다. 설정 단위는 MB이며, 설정 하지 않은 경우 파일 사이즈의 제한없이 사용 할 수 있다. 이 속성은 IE 브라우저의 경우 11이상부터 지원 하고, 그외의 브라우저에서는 모두 지원 한다.

Mode 속성은 IBSheet 헤더의 타이틀과 엑셀의 헤더 타이틀을 서로 매칭하여 로딩할지의 여부와 엑셀 로딩시 헤더의 포함 유무를 나타낸다.

Mode 값	설명
HeaderMatch	엑셀 문서의 컬럼들이 뒤섞여 있더라도 헤더내의 타이틀들을 비교하여 그 순서대로 찾아서 로딩한다.
NoHeader	엑셀 문서 내에 헤더가 없이 첫 행부터 데이터로 구성되어 있으므로 차례대로 로딩한다.



HeaderSkip	엑셀문서의 첫 행부터 그리드의 헤더 행 수 만큼은 헤더데이터 이지만 무시하도록 하고 그 이후의 행부터 데이터로 간주하여 차례대로 로딩한다. IBSheet의 헤더가 두 줄인 경우 엑셀의 세번째 데이터 행부터 로딩한다.
------------	--

StartRow 속성과 **EndRow** 속성은 읽어올 영역의 시작과 마지막 행의 번호를 의미하며, 설정하지 않을 시 기본적으로 엑셀 영역의 처음부터 끝까지 읽어들이는다. 엑셀에서 행번호는 1부터 시작한다.

StartCol 속성은 읽어올 컬럼의 시작번호를 의미하며, 설정하지 않을 시 기본적으로 엑셀 첫컬럼부터 끝까지 읽어들이는다. 엑셀에서 컬럼번호는 1부터 시작한다.

UseDOM 속성은 2007 형식(*.xlsx)에 대한 엑셀파일의 파싱 방법을 설정한다. true인 경우 엑셀 파일에 DOM 기반 파싱을 수행하며, false인 경우 엑셀 파일 파싱에 SAX방식을 사용 한다.

LoadExcel 호출시 SAX(default)파싱 방식을 사용하여 대용량처리시 메모리를 적게 사용하고 빠른처리가 되지만 엑셀파일의 데이터 파싱중 서식부분에서 오류가 생길 수 있어서 IBSheet에 로드되는 값이 다르게 표현될 수 있다. 이때 true를 사용하여 로드를 하는 경우 데이터 값을 그대로 표현 가능하며, 2003, 2007 형식의 엑셀간의 호환성에 유리하다.

WorkSheetName 속성은 읽어올 엑셀파일의 WorkSheet 이름을 선택한다. 엑셀의 WorkSheet 선택은 WorkSheetName 인자의 값을 우선적으로 읽어들이고, 이를 실패한 경우, WorkSheetNo 인자의 값으로 WorkSheet를 선택한다.

(사용 예)

ColumnMapping:'1|2|3' // IBSheet의 첫번째~ 세번째 컬럼을 엑셀의 1번, 2번, 3번 컬럼들의 값을 차례대로 로딩한다.

ColumnMapping:'7||7' // IBSheet의 첫번째와 세번째 컬럼에만 엑셀의 7번째 컬럼의 값으로 로딩한다.

➤ **Syntax**

Syntax	ObjId. LoadExcel ([Opt])
--------	---------------------------------

➤ **Info**

Parameter	Type	필수여부	설 명
Opt.Append	Boolean	선택	기존 데이터에 추가 여부 Default=0
Opt.ColumnMapping	String	선택	엑셀 컬럼 번호 Default=""
Opt.EndRow	String	선택	엑셀 로딩완료 행번호 Default="0"
Opt.ExtendParam	String	선택	서버로 전달할 추가 파라미터 들을 a=1&b=2 형태로 넣어 준다. Default=""
Opt.FileExt	String	선택	업로드 가능한 파일 확장자 Default=""
Opt.MaxFileSize	Number	선택	최대 허용 파일 사이즈 (단위:MB)
Opt.Mode	String	선택	로딩 방식 (헤더매칭등) Default="HeaderMatch"
Opt.StartRow	String	선택	엑셀 로딩 행번호 Default="1"
Opt.StartCol	String	선택	엑셀 로딩 컬럼번호 Default="1"
Opt.UseDOM	Boolean	선택	엑셀 로딩시 DOM 기반 파싱 수행 여부 Default="0"
Opt.WorkSheetNo	String	선택	엑셀WorkSheet번호 Default="1"
Opt.WorkSheetName	String	선택	엑셀WorkSheet이름 Default=""



➤ **Example**

```
// 엑셀로 로딩하기
mySheet.LoadExcel();

// 헤더타이틀을 서로 비교하여 동일한 컬럼끼리 로딩을 하되 7행부터 9 행
// 까지만 읽어들인다.
mySheet.LoadExcel({Mode:'HeaderMatch',StartRow:'7',EndRow:'9'});

// IBSheet의 첫번째 컬럼에 엑셀의 5번째 컬럼의 값을 로딩하고, IBSheet 의
// 5번째 컬럼에 엑셀의 1번째 컬럼의 값을 로딩함.
mySheet.LoadExcel({ColumnMapping:'5|4|3|2|1'});
```

➤ **제공 버전**

7.0.0.0	
7.0.13.27	MaxFileSize 속성 추가

LoadExcelBuffer Method

➤ 기능

1개의 엑셀 문서로부터 여러 개의 시트에 로딩한다.

LoadExcelBuffer 의 IsBuffer 인자를 true 로 설정하면 이후로 실행되는 LoadExcel은 실제로 동작하지 않으며 모두 내부 메모리에 버퍼링된다. 이후에 LoadExcelBuffer 의 IsBuffer 인자를 false 로 설정하는 순간 파일 업로드 창이 뜨게된다.

버퍼링되는 동안 지정되는 FileExt 속성은 제일 마지막에 설정된 값이 유효하다.

➤ Syntax

Syntax	ObjId. LoadExcelBuffer (IsBuffer)
--------	--

➤ Info

Parameter	Type	필수여부	설 명
IsBuffer	bool	필수	버퍼링 여부

➤ Example

```
// 엑셀 파일 즉시 로딩하기
mySheet.LoadExcel();

//이후로는 버퍼링한다. 아무 동작 안함.
mySheet.LoadExcelBuffer(true);

// mySheet에 엑셀파일의 첫번째 시트의 헤더타이틀을 서로 비교하여 동일한 컬럼끼리 로딩을 하되 7행부터 9 행까지만 로딩하기로 예약함
mySheet.LoadExcel({Mode:'HeaderMatch',StartRow:'7',EndRow:'9',
WorksheetNo:1});
```



// mySheet2에 엑셀파일의 두번째 시트를 IBSheet의 첫번째 컬럼에 엑셀의 5번째 컬럼의 값을 로딩하고, IBSheet 의 5번째 컬럼에 엑셀의 1번째 컬럼의 값을로딩하기로 예약함

```
mySheet2.LoadExcel({ColumnMapping:'5|4|3|2|1', WorkSheetNo:2});
```

// 버퍼링된 모든 엑셀 로딩 자료와 파일을 서버로 보내기 위해 파일 업로드 창을 띄운다

```
mySheet.LoadExcelBuffer(false);
```

➤ **제공 버전**

7.0.0.0

LoadExcelUrl Method

➤ 기능

엑셀 업로드 기능을 처리할 서버 페이지 경로를 확인 및 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetLoadExcelUrl()
--------	-----	---------------------------------

➤ Info

Return	String, 설정된 경로값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// 엑셀 업로드 설정 경로를 확인한다.
var url = mySheet.GetLoadExcelUrl();
```

➤ Syntax

Syntax	Set	ObjId. SetLoadExcelUrl (Url)
--------	-----	-------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Url	String	필수	설정할 서버 페이지 Url

➤ Example

```
// 엑셀 업로드 경로를 설정한다.
mySheet.SetLoadExcelUrl("/jsp/LoadExcel.jsp");
```

➤ 제공 버전

7.0.0.0

LoadText Method

➤ 기능

텍스트 파일로부터 데이터를 읽어 들여 데이터 영역에 표시한다.

전달되는 인자의 형식은 JSON 타입으로 설정 하고자 하는 정보를 JSON 형태로 구성하여 전달한다.

Append 속성을 1로 설정하는 경우 기존 데이터에 현재 조회 데이터를 첨부하여 조회 할 수 있다.

ColSeparator 속성은 데이터의 컬럼 구분자를 설정한다. 기본값은 "wt"이다.

FileExt 속성은 업로드 가능한 파일 확장자를 지정한다. 확장자는 "."을 제외한 "txt" 등으로 입력해야 하며, 기본값은 ""으로 "txt"로 입력한 것과 동일하다. 여러 개의 확장자는 "|"로 구분하여 입력한다.

MaxFileSize 속성은 처리 대상 파일의 사이즈를 제한하고자 할 때 사용한다. 설정 단위는 MB이며, 설정 하지 않은 경우 파일 사이즈의 제한없이 사용 할 수 있다. 이 속성은 IE 브라우저의 경우 11이상부터 지원 하고, 그외의 브라우저에서는 모두 지원 한다.

Mode 속성은 IBSheet 헤더의 타이틀과 엑셀의 헤더 타이틀을 서로 매칭하여 로딩할지의 여부와 엑셀 로딩시 헤더의 포함 유무를 나타낸다.

Mode 값	설명
HeaderMatch	텍스트 문서의 컬럼들이 뒤섞여 있더라도 헤더내의 타이틀들을 비교하여 그 순서대로 찾아서 로딩한다.
NoHeader	텍스트 문서 내에 헤더가 없이 첫 행부터 데이터로 구성되어 있으므로 차례대로 로딩한다.

HeaderSkip	텍스트 문서의 첫 행부터 그리드의 헤더 행 수 만큼은 헤더데이터이지만 무시하도록 하고 그 이후의 행부터 데이터로 간주하여 차례대로 로딩한다. IBSheet의 헤더가 두 줄인 경우 파일의 세번째 데이터 행부터 로딩한다.
------------	---

➤ Syntax

Syntax	ObjId.LoadText([Opt])
--------	-----------------------

➤ Info

Parameter	Type	필수여부	설 명
Opt.Mode	String	선택	로딩 방식 (헤더매칭등) Default="HeaderMatch"
Opt.ColSeparator	String	선택	데이터의 컬럼 구분자. Default="\t"
Opt.Append	Boolean	선택	기존 데이터에 추가 여부 Default=0
Opt.FileExt	String	선택	업로드 가능한 파일 확장자 Default=""

➤ Example

<pre>// 텍스트 파일로 로딩하기 mySheet. LoadText(); // 헤더타이틀을 서로 비교하여 동일한 컬럼끼리 로딩을 하며, 컬럼 구분자는'\t' 로 구분되어 있다 mySheet.Load Text({Mode:'HeaderMatch', Deli:'\t'});</pre>

➤ 제공 버전

7.0.0.0	
7.0.13.27	MaxFileSize 속성 추가 ColSeparator 속성 추가 (Deli 속성 대체)





LoadTextUrl Method

➤ 기능

텍스트 업로드 기능을 처리할 서버 페이지 경로를 확인 및 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetLoadTextUrl ()
--------	-----	---------------------------------

➤ Info

Return	String, 설정된 경로값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// 텍스트 업로드 설정 경로를 확인한다.  
var url = mySheet.GetLoadTextUrl();
```

➤ Syntax

Syntax	Set	ObjId. SetLoadTextUrl (Url)
--------	-----	------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Url	String	필수	설정할 서버 페이지 Url

➤ Example

```
// 텍스트 업로드 경로를 설정한다.  
mySheet.SetLoadTextUrl("/jsp/LoadText.jsp");
```

➤ 제공 버전

7.0.0.0

MergeSheet Method

➤ 기능

전체적인 머지 종류를 설정하거나 확인한다.

전체 머지가 되지 않는 경우 어떤 셀도 머지 되지 않으며, 전체 머지 가능 상태에서 가로, 세로 머지 가능 여부를 설정하여 머지 되도록 한다.

- ※ 머지된 데이터는 SetConfig에서 설정한 Page 크기에 의해 나누어 진다. 단 머지 유형이 msPrevColumnMerge을 사용하는 경우에는 SetConfig에 PrevColumnMergeMode:0 를 설정하면 앞컬럼 기준으로 머지된 데이터를 표현할 수 있다. 하지만 이 방식은 한번에 머지된 행을 모두 화면 위로 올리므로 머지된 행의 개수가 많아지면 속도가 저하될 수 있다.

SetConfig Method의 MergeSheet 설정과 동일한 기능을 한다.

상수값	종류	설명
0	msNone	머지 없음
1	msAll	전부 머지 가능
2	msPrevColumnMerge	앞 컬럼이 머지 된 경우 해당 행 안에서 머지 가능
3	msFixedMerge	단위데이터 행 구조에서 고정 셀 병합 기능
4	msBaseColumnMerge	기준컬럼 머지 영역 범위 내에서 머지 처리 기능 (좌측으로부터 첫번째 ColMerge 속성이 1인 컬럼이 기준컬럼이 된다.)
5	msHeaderOnly	헤더만 머지
7	msPrevColumnMerge + msHeaderOnly	앞 컬럼이 머지 된 경우 해당 행 안에서 머지기능 + 헤더 머지
8	msFixedMerge + msHeaderOnly	단위데이터 행 구조에서 고정 셀 병합 기능 + 헤더 머지
9	msBaseColumnMerge + msHeaderOnly	기준컬럼 머지 영역 범위 내에서 머지 처리 기능 + 헤더 머지

➤ Syntax



Syntax	Get	ObjId. GetMergeSheet ()
--------	-----	--------------------------------

➤ **Info**

Return	Integer, 현재 설정된 머지 종류 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ **Example**

```
//머지 방식 확인  
mySheet.GetMergeSheet();
```

➤ **Syntax**

Syntax	Set	ObjId. SetMergeSheet (Merge)
--------	-----	-------------------------------------

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Merge	Integer	필수	머지 종류

➤ **Example**

```
//초기 로드 시 전체 머지 종류를 설정  
mySheet.SetMergeSheet( msHeaderOnly);
```

➤ **제공 버전**

7.0.0.0



MaximumValue Method

➤ 기능

Format이 숫자와 관련된 포맷인 Integer, Float, NullInteger, NullFloat일 때, 편집시 입력할 수 있는 최대값을 확인하거나 설정한다.

최대값이 설정되어있는 셀을 수정시 설정한 값보다 큰값을 입력하면 경고 메시지를 표시한다. (ibmsg 파일의 SYS_MaximumBigValue)

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

기본적으로 설정된 최대값과 최소값은 다음과 같다.

Format	최소값	최대값
Integer NullInteger	-999999999999999	999999999999999
Float NullFloat	-999999999999999.00	999999999999999.00

➤ Syntax

Syntax	Get	ObjId. GetMaximumValue (Row, Col)
--------	-----	--

➤ Info

Return	Long, 설정된 최대값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 행 Index
Col	Long/ String	필수	해당 셀의 컬럼 Index 또는 SaveName

➤ Example



```
//설정된 숫자포맷의 최대값을 확인한다.  
mySheet.GetMaximumValue(1, 12);
```

➤ **Syntax**

Syntax	Set	ObjId. SetMaximumValue (Row, Col, Value)
--------	-----	---

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 행 Index
Col	Long/ String	필수	해당 셀의 컬럼 Index 또는 SaveName
Value	Long	필수	설정할 최대값

➤ **Example**

```
//숫자포맷의 최대값을 설정한다.  
mySheet.SetMaximumValue(1, 12, 5000);
```

➤ **제공 버전**

7.0.0.0



MinimumValue Method

➤ 기능

Format이 숫자와 관련된 포맷인 Integer, Float, NullInteger, NullFloat일 때, 편집시 입력할 수 있는 최소값을 확인하거나 설정한다.

최소값이 설정되어있는 셀을 수정시 설정한 값보다 작은값을 입력하면 경고 메시지를 표시한다. (ibmsg 파일의 SYS_MinimumBigValue)

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

기본적으로 설정된 최대값과 최소값은 다음과 같다.

Format	최소값	최대값
Integer NullInteger	-999999999999999	999999999999999
Float NullFloat	-999999999999999.00	999999999999999.00

➤ Syntax

Syntax	Get	ObjId. GetMinimumValue (Row, Col)
--------	-----	--

➤ Info

Return	Long, 설정된 최소값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 데이터의 행 Index
Col	Long/ String	필수	해당 데이터의 컬럼 Index 또는 SaveName

➤ Example



```
//설정된 숫자포맷의 최소값을 확인한다.  
mySheet.GetMinimumValue(1, 12);
```

➤ **Syntax**

Syntax	Set	ObjId. SetMinimumValue (Row, Col, Value)
--------	-----	---

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 데이터의 행 Index
Col	Long/ String	필수	해당 데이터의 컬럼 Index 또는 SaveName
Value	Long	필수	설정할 최소값

➤ **Example**

```
// 숫자포맷의 최소값을 설정한다.  
mySheet.SetMinimumValue(1, 12, 1000);
```

➤ **제공 버전**

7.0.0.0



MouseCol Method

➤ 기능

마우스가 눌러졌을 때 선택된 컬럼 번호를 반환한다. 데이터 영역이 아닌 경우는 -1을 반환한다.

➤ Syntax

Syntax	ObjId. MouseCol()
--------	--------------------------

➤ Info

Return	Long, 마우스위치가 있는 컬럼 인덱스		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// 마우스가 눌러졌을 때 컬럼 번호를 가져온다.  
Function mySheet_OnMouseDown(Button, Shift, X, Y) {  
    // 눌린 컬럼 확인  
    alert(mySheet.MouseRow() + "행 " + mySheet.MouseCol() + "컬럼이 눌  
림");  
}
```

➤ 제공 버전

7.0.0.0

MouseHoverMode Method

➤ 기능

시트내의 행 또는 셀에 마우스 커서가 위치했을 때 Hover 동작 모드를 설정한다.

설정값	설명
0	사용안함 (Default)
1	셀단위 Hover
2	행단위 Hover

➤ Syntax

Syntax	Get	ObjId. GetMouseHoverMode()
--------	-----	-----------------------------------

➤ Info

Return	String, 마우스 Hover 모드 설정 값		
Parameter	Type	필수여부	설 명

➤ Example

```
//마우스 Hover 모드 설정값을 확인한다.
mySheet.GetMouseHoverMode();
```

➤ Syntax

Syntax	Set	ObjId. SetMouseHoverMode (Mode)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Mode	Number	필수	설정할 모드 값

➤ Example



```
//마우스 Hover 모드를 행단위로 설정 한다.  
mySheet.SetMouseHoverMode(2);
```

➤ **제공 버전**

7.0.8.0	
----------------	--



MousePointer Method

➤ 기능

브라우저의 특성상 해당기능이 잘 동작하지 않는 경우가 많으므로 이 기능보다는 **DataLinkMouse** 기능을 사용할 것을 권한다.

마우스의 모양을 설정하거나 확인한다. 설정할수 있는 값은 아래와 같다.

설정값	마우스 모양
Default	기본 화살표 모양
Hand	손가락 모양

이 속성은 DataLinkMouse 속성과 혼동하지 않도록 한다.

➤ Syntax

Syntax	Get	ObjId. GetMousePointer ()
--------	-----	----------------------------------

➤ Info

Return	String, 마우스 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//마우스의 모양을 확인한다.
mySheet.GetMousePointer("Default");
```

➤ Syntax

Syntax	Set	ObjId. SetMousePointer (Pointer)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Pointer	String	필수	설정할 마우스 모양



➤ **Example**

```
function mySheet_OnMouseMove(Button, Shift, X, Y) {  
    //마우스 위치가 2컬럼 일때만 마우스 손가락 모양  
    if(mySheet.MouseCol() == 2) {  
        mySheet.SetMousePointer("Hand");  
    } else {  
        mySheet.SetMousePointer("Default");  
    }  
}
```

➤ **제공 버전**

7.0.0.0



MouseRow Method

➤ 기능

마우스가 눌러졌을 때 선택된 행 번호를 반환한다. 데이터 영역이 아닌 경우는 -1을 반환한다.

➤ Syntax

Syntax	ObjId. MouseRow()
--------	--------------------------

➤ Info

Return	Long, 마우스포인트가 있는 셀의 행 Index	
Parameter	Type	설 명
		없음

➤ Example

```
//마우스가 눌러졌을 때 행 번호를 가져온다.  
Function mySheet_OnMouseDown(Button, Shift, X, Y){  
    //눌린 행 확인  
    alert(mySheet.MouseRow() + "행이 눌림");  
}
```

➤ 제공 버전

7.0.0.0



MouseToolTipText Method

➤ 기능

마우스에 표시될 푹션도움말을 확인하거나 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetMouseToolTipText()
--------	-----	-------------------------------------

➤ Info

Return	String, 설정된 문자열		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
//설정된 MouseToolTipText 값을 확인한다.  
mySheet.GetMouseToolTipText();
```

➤ Syntax

Syntax	Set	ObjId. SetMouseToolTipText(Value)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Value	String	필수	설정할 문자열

➤ Example

```
// MouseMove 이벤트에서 마우스 위치의 X, Y 좌표값을 ToopTip으로 설정  
한다.  
Function mySheet_OnMouseMove(button, shift, x, y) {  
    var value = "마우스 좌표 = " + x + "," + y;  
    mySheet.SetMouseToolTipText(value);  
}
```

}.

➤ **제공 버전**
7.0.0.0



MoveColumnFail Method

➤ 기능

OnBeforeColumnMove 이벤트 안에서 사용하는 속성이며, 컬럼이동 실패 여부를 설정한다. OnBeforeColumnMove 이벤트 안에서 이 속성을 1로 설정하면 컬럼 이동은 실패하고, OnAfterColumnMove 이벤트는 발생하지 않는다.

➤ Syntax

Syntax	ObjId. MoveColumnFail (Flag)
--------	-------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Flag	Boolean	필수	컬럼이동실패여부

➤ Example

```
//0컬럼이 3컬럼 이후의 위치로 이동하고자 할때 컬럼 이동을 취소한다
function mySheet_OnBeforeColumnMove(Col, NewPos) {
    if(Col==0 && NewPos > 3) {
        mySheet.MoveColumnFail(1);
    }else{
        mySheet.MoveColumnFail(0);
    }
}
```

➤ 제공 버전

7.0.0.0

MoveColumnPos Method

➤ 기능

특정 컬럼을 새로운 컬럼 위치로 이동한다. Event 인자를 1로 설정한 경우 OnBeforeColumnMove Event와 OnAfterColumnMove Event를 발생하므로 OnBeforeColumnMove Event 에서 처리할수 있는 컬럼 이동 취소 기능을 할수 있다. 0으로 설정할 경우 이벤트 없이 컬럼을 이동한다.

➤ Syntax

Syntax	ObjId. MoveColumnPos (Col, NewPos, [Event])
--------	--

➤ Info

Return	Boolean, 이동 성공 여부		
Parameter	Type	필수여부	설 명
Col	Long/String	필수	이동할 컬럼의 Index 또는 SaveName
NewPos	Long/String	필수	이동될 위치의 컬럼 Index 또는 SaveName
Event	Boolean	선택	이동되면서 발생하는 이벤트인 OnBeforeColumnMove Event와 OnAfterColumn Event를 발생여부, Default=1

➤ Example

```
//1컬럼을 9컬럼으로 이동한다.
mySheet.MoveColumnPos(1, 9);

//각 컬럼의 SaveName으로 설정한 경우
mySheet.MoveColumnPos("stockNm", "payAmt");
```

➤ 제공 버전

7.0.0.0



PagingPosition Method

➤ 기능

페이지 네비게이션을 사용여부를 설정하거나 확인 한다.

사용시 건수 정보 영역에 설정값에 따라 좌측 또는 우측에 표시되며, CountPosition 설정이 1 ~ 4의 값으로 설정이 되어 있어야 한다.

건수 정보 또는 선택 요약 정보의 설정보다 페이지 네비게이션 설정이 우선시 되기 때문에 페이지 네비게이션 설정영역에 위치한 건수 정보 또는 선택 요약 정보는 표시 되지 않는다.

페이지 네비게이션이 불필요한 조회모드를 설정한 경우 설정값과 상관없이 0 으로 설정 된다 (SearchMode:smGeneral)

설정 값 별로 개수 정보 표시 영역 정보는 다음과 같다.

설정값	0	1	2
표시위치	표시하지 않음	좌측	우측

➤ Syntax

Syntax	Get	ObjId. GetPagingPosition()
--------	-----	-----------------------------------

➤ Info

Return	Number, 페이지 네비게이션의 설정 위치 값		
Parameter	Type	필수여부	설 명

➤ Example

// 현재 설정된 페이지 네비게이션 설정 위치값을 확인 한다. mySheet.GetPagingPosition()

➤ Syntax



Syntax	Set	ObjId. SetPagingPosition (Position)
--------	-----	--

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Position	Number	필수	설정할 페이지 네비게이션 위치 값

➤ **Example**

```
// 페이지 네비게이션을 좌측 상단, 건수 정보를 우측 상단에 표시 한다.  
mySheet.SetCountPosition(2);  
mySheet.SetPagingPosition(1)
```

➤ **제공 버전**

7.0.0.0



PageCount Method

➤ 기능

조회 모드(SearchMode)에서 사용되는 Page 속성의 값을 설정하거나 확인 한다.

➤ Syntax

Syntax	Get	ObjId. GetPageCount ()
--------	-----	-------------------------------

➤ Info

Return	Number, Page 개수 설정값		
Parameter	Type	필수여부	설 명

➤ Example

```
// 현재 설정된 페이지 개수를 확인 한다..  
mySheet.GetPageCount()
```

➤ Syntax

Syntax	Set	ObjId. SetPageCount (Page)
--------	-----	-----------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Page	Number	필수	설정할 페이지 개수

➤ Example

```
// 페이지 개수를 50으로 설정 한다.  
mySheet.SetPageCount(50);
```

➤ 제공 버전



7.0.2.0	
---------	--



PopupButtonImage Method

➤ 기능

대상 셀의 팝업 버튼 이미지를 변경한다.

Image 인자는 사용할 이미지의 Url 또는 이미지리스트에 설정한 이미지 인덱스를 사용할 수 있다.

Image 인자를 상대경로 Url로 설정하는 경우 테마폴더를 기준으로 설정한다.

단위데이터행 구조인 경우 Col인자에 인덱스를 설정하면 단위데이터 첫번째 행에 대해서 처리하고, SaveName을 설정하면 설정행을 기준으로한 단위데이터내의 SaveName 컬럼에 대해서 처리한다.

➤ Syntax

Syntax	ObjId.PopupButtonImage(Row, Col, Image)
--------	---

➤ Info

Return	설정불가능 셀인경우 -1 반환		
Parameter	Type	필수여부	설 명
Row	Long	필수	설정할 셀의 행 인덱스
Col	Long/String	필수	설정할 셀의 컬럼 인덱스 또는 SaveName
Image	Long/String	필수	설정할 이미지 Url 또는 이미지인덱스

➤ Example

```
// 셀의 팝업버튼 이미지를 변경
mySheet.PopupButtonImage(2, 10, "/image/ic_popup.gif");

// sPopup 컬럼의 팝업버튼 이미지를 이미지리스트에 설정한 이미지로 변경
mySheet. PopupButtonImage (4, "sPopup", 1);
```



- 제공 버전
7.0.0.0



RangeBackColor Method

➤ 기능

셀의 범위 영역의 배경색을 설정 한다.

그 영역이 데이터 영역이든 헤더 영역이든 셀의 배경색을 설정 할 수 있으며, 영역이 헤더와 데이터 영역을 벗어나는 경우 에러메시지 없이 배경색 설정은 취소 된다.

➤ Syntax

Syntax	Set	ObjId. RangeBackColor (Row1,Col1,Row2,Col2,Color)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row1	Number	필수	범위 시작 셀의 Row Index
Col1	Number	필수	범위 시작 셀의 Column Index
Row2	Number	필수	범위 종료 셀의 Row Index
Col2	Number	필수	범위 종료 셀의 Column Index
Color	String	필수	WebColor

➤ Example

//영역에 배경색 설정하기 mySheet.RangeBackColor(1,0,10,10, "#FFFF00");

➤ 제공 버전

7.0.13.16	
-----------	--

RangeFontBold Method

➤ 기능

셀의 범위 영역의 글자볼드 여부를 설정 한다.

그 영역이 데이터 영역이든 헤더 영역이든 셀의 글자볼드를 설정할 수 있으며, 영역이 헤더와 데이터 영역을 벗어나는 경우 에러메시지 없이 글자볼드 설정은 취소 된다.

➤ Syntax

Syntax	Set	ObjId. RangeFontBold (Row1, Col1, Row2, Col2, Bold)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row1	Number	필수	범위 시작 셀의 Row Index
Col1	Number	필수	범위 시작 셀의 Column Index
Row2	Number	필수	범위 종료 셀의 Row Index
Col2	Number	필수	범위 종료 셀의 Column Index
Bold	Boolean	필수	볼드 설정 여부

➤ Example

//영역 글자 볼드 설정하기 mySheet.RangeFontBold(1, 0, 2, 2, 1);
--

➤ 제공 버전

7.0.13.16	
-----------	--



RangeFontColor Method

➤ 기능

셀의 범위 영역의 글자색을 설정 한다.

그 영역이 데이터 영역이든 헤더 영역이든 셀의 글자색을 설정할 수 있으며, 영역이 헤더와 데이터 영역을 벗어나는 경우 에러메시지 없이 글자색 설정은 취소 된다.

➤ Syntax

Syntax	Set	ObjId. RangeFontColor (Row1, Col1, Row2, Col2, Color)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row1	Number	필수	범위 시작 셀의 Row Index
Col1	Number	필수	범위 시작 셀의 Column Index
Row2	Number	필수	범위 종료 셀의 Row Index
Col2	Number	필수	범위 종료 셀의 Column Index
Color	String	필수	WebColor

➤ Example

//영역에 글자색 설정하기 mySheet.RangeFontColor(1,0,10,10, "FF0000");
--

➤ 제공 버전

7.0.13.16	
-----------	--

RangeText Method

➤ 기능

특정 영역의 셀값을 Format이 적용된 형태로 화면에 보여지는 값 그대로 확인하거나 설정한다.

그 영역이 데이터 영역이든 헤더 영역이든 셀의 텍스트를 설정하거나 확인할 수 있으며, 영역이 헤더와 데이터 영역을 벗어나는 경우 에러메시지 없이 문자열 설정은 취소되고, 확인 시 0이 반환된다.

➤ Syntax

Syntax	Get	ObjId. GetRangeText (Row1,Col1,Row2,Col2,[ColSeparator],[RowSeparator])
--------	-----	--

➤ Info

Return	String, 현재 문자열값(Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row1	Long	필수	범위 시작 셀의 Row Index
Col1	Long	필수	범위 시작 셀의 Column Index
Row2	Long	필수	범위 종료 셀의 Row Index
Col2	Long	필수	범위 종료 셀의 Column Index
ColSeparator	String	선택	컬럼과 컬럼 사이를 구분하는 구분자. Default=" "
RowSeparator	String	선택	행과 행 사이를 구분하는 구분자. Default="^"

➤ Example

```
//특정 영역의 셀값을 Format이 적용된 형태로 화면에 보여지는 값 그대로 확인한다.
mySheet.GetRangeText(1, 1, 2, 2, "|", "^");
```



➤ **Syntax**

Syntax	Set	ObjId. SetRangeText (sData,Row1,Col1,Row2,Col2, [ColSeparator], [RowSeparator])
--------	-----	---

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
sData	String	필수	문자열
Row1	Long	필수	범위 시작 셀의 Row Index
Col1	Long	필수	범위 시작 셀의 Column Index
Row2	Long	필수	범위 종료 셀의 Row Index
Col2	Long	필수	범위 종료 셀의 Column Index
ColSeparator	String	선택	컬럼과 컬럼 사이를 구분하는 구분자. Default=" "
RowSeparator	String	선택	행과 행 사이를 구분하는 구분자. Default="^"

➤ **Example**

<pre>//1,1 셀에 A 를 2,2셀에 D를 설정한다. mySheet.SetRangeText("A B^C D", 1, 1, 2, 2, " ", "^"); //1,7 셀부터 5,10 셀까지 ** 로 설정한다. mySheet.SetRangeText("**", 1, 7, 5, 10);</pre>

➤ **제공 버전**

7.0.0.0

RangeValue Method

➤ 기능

특정 영역의 셀값을 Format이 적용되지 않은 실제로 저장 시 사용하는 값으로 확인하고, 설정한다

그 영역이 데이터 영역이든 헤더 영역이든 셀의 값을 설정하거나 확인할 수 있으며, 영역이 헤더와 데이터 영역을 벗어나는 경우 에러메시지 없이 문자열 설정은 취소되고, 확인 시 0이 반환된다.

➤ Syntax

Syntax	Get	ObjId. GetRangeValue (Row1,Col1,Row2,Col2,[ColSeparator],[RowSeparator])
--------	-----	---

➤ Info

Return	String, 현재 문자열값(Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row1	Long	필수	범위 시작 셀의 Row Index
Col1	Long	필수	범위 시작 셀의 Column Index
Row2	Long	필수	범위 종료 셀의 Row Index
Col2	Long	필수	범위 종료 셀의 Column Index
ColSeparator	String	선택	컬럼과 컬럼 사이를 구분하는 구분자. Default=
RowSeparator	String	선택	행과 행 사이를 구분하는 구분자. Default=^

➤ Example

```
// 특정 영역의 셀값을 Format이 적용되지 않은 실제로 저장 시 사용하는 값으로 확인한다.
mySheet.GetRangeValue("A|B^C|D", 1, 1, 2, 2, "|", "^");
```

➤ Syntax



Syntax	Set	ObjId. SetRangeValue (sData,Row1,Col1,Row2,Col2, [ColSeparator],[RowSeparator])
--------	-----	---

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
sData	String	필수	문자열
Row1	Long	필수	범위 시작 셀의 Row Index
Col1	Long	필수	범위 시작 셀의 Column Index
Row2	Long	필수	범위 종료 셀의 Row Index
Col2	Long	필수	범위 종료 셀의 Column Index
ColSeparator	String	선택	컬럼과 컬럼 사이를 구분하는 구분자. Default=
RowSeparator	String	선택	행과 행 사이를 구분하는 구분자. Default=^

➤ **Example**

//1,1 셀에 A 를 2,2셀에 D를 설정한다. mySheet.SetRangeValue("A B^C D", 1, 1, 2, 2, " ", "^"); //1,7 셀부터 5,10 셀까지 ** 로 설정한다. mySheet.SetRangeValue("**", 1, 7, 5, 10);
--

➤ **제공 버전**

7.0.0.0

RedrawSum Method

➤ 기능

합계 내용 계산 여부를 확인하거나 설정한다.

합계를 표시하는 컬럼의 값을 SetCellValue Method등으로 변경하면 합계를 계산하느라고 속도가 느려지는 증상이 발생한다. 하나의 데이터를 설정하면서 합계를 다시 계산하기 때문이다. 이때 데이터를 모두 변경한 다음에 합계를 계산하는 기능을 설정함으로써 데이터 변경의 속도를 향상시킬수 있다. 이 속성에 주의할 점은 0으로 사용 후 반드시 1로 설정 하여야 한다.

➤ Syntax

Syntax	Get	ObjId. GetRedrawSum()
--------	-----	------------------------------

➤ Info

Return	Boolean, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//합계 내용 계산 여부를 확인한다.
mySheet.GetRedrawSum();
```

➤ Syntax

Syntax	Set	ObjId. SetRedrawSum (Redraw)
--------	-----	-------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Redraw	Boolean	필수	합계행 계산 여부

➤ Example



```
//변경해야 할 데이터가 많은 경우 사용  
mySheet.SetRedrawSum(0);  
for(var i=1; i<100; i++) mySheet.SetCellValue(i,1, 10000, 0);  
//RedrawSum 가 1가 되는 순간 합계와 데이터가 한꺼번에 표시  
mySheet.SetRedrawSum(1);
```

➤ **제공 버전**

7.0.0.0



RemoveAll Method

➤ 기능

헤더 행을 제외한 모든 데이터 행을 지운다.

➤ Syntax

Syntax	ObjId.RemoveAll()
--------	-------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
없음			

➤ Example

// 모든 데이터 지우기 mySheet.RemoveAll();

➤ 제공 버전

7.0.0.0



RemoveFooterRow Method

➤ 기능

생성된 하단 고정행을 삭제한다.

➤ Syntax

Syntax	ObjId. RemoveAll ()
--------	----------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// 생성된 하단 고정행을 삭제한다.  
mySheet.RemoveFooterRow();
```

➤ 제공 버전

7.0.13.60

RenderSheet Method

➤ 기능

행 추가 또는 컬럼 숨김 기능을 동시에 여러 개 사용하는 경우 속도개선을 위해 랜더링을 마지막에 한꺼번에 처리 하기 위한 기능을 한다.

주의! 이 속성을 0(false)으로 사용 후 반드시 1(true)로 설정 하여야 한다. 그렇지 않으면 Sheet의 모양이 그려지지 않아서 깨지는 현상이 있을 수 있다.

아래와 같은 기능을 동시에 여러 개 수행할 때 사용하면 속도 개선을 할 수가 있다.

※ Render 인자를 2로 설정하는 경우 시트 영역 전체에 대한 강제 랜더링 처리를 할 수 있다.

메소드	설명
ColHidden	컬럼 숨김
DataInsert	행 추가
InitCellProperty	셀의 속성 재 설정

➤ Syntax

Syntax	ObjId. RenderSheet (Render)
--------	------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Render	Boolean	필수	렌더링 여부 (Default=1)

➤ Example

```
// 30개의 행추가를 RenderSheet를 이용하여 속도개선 처리
mySheet.RenderSheet(0);
```



```
for (var i = 0; i < 30; i++) {  
    mySheet.DataInsert();  
}  
mySheet.RenderSheet(1);
```

➤ **제공 버전**

7.0.0.0



RemoveEtcData Method

➤ 기능

EtcData 객체에 설정된 키와 값을 모두 제거 한다.

➤ Syntax

Syntax	ObjId. RemoveEtcData()
--------	-------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
//EtcData에 설정된 모든 데이터 지우기  
mySheet.RemoveEtcData();
```

➤ 제공 버전

7.0.0.0



ReNumberSeq Method

➤ 기능

Seq 컬럼타입에 대한 데이터 순번을 재설정 한다.

Order 인자의 설정에 따라 오름차순 또는 내림차순으로 설정 할 수 있으며 설정 값은 대소구분없이 처리 된다.

➤ Syntax

Syntax	ObjId.ReNumberSeq([Order])
--------	----------------------------

➤ Info

Return	None								
Parameter	Type	필수여부	설 명						
Order	String	선택	넘버링 처리 방식 (Default="asc")						
			<table><tr><th>설정값</th><th>설명</th></tr><tr><td>asc</td><td>오름 차순</td></tr><tr><td>desc</td><td>내림 차순</td></tr></table>	설정값	설명	asc	오름 차순	desc	내림 차순
			설정값	설명					
			asc	오름 차순					
desc	내림 차순								

➤ Example

<pre>//컬럼의 Type이 Seq인 컬럼의 숫자를 1부터 다시 매긴다. mySheet.ReNumberSeq(); // 역순 넘버링 처리 mySheet.ReNumberSeq("desc");</pre>

➤ 제공 버전

7.0.0.0	
7.0.13.5	Order 인자 추가



Reset Method

➤ 기능

IBSheet에 설정된 모든 기본 속성을 제거하고 초기상태로 변경한다.

➤ Syntax

Syntax	ObjId. Reset ([KeepTheme])
--------	-----------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
KeepTheme	Boolean	선택	적용되어 있는 테마 유지 여부, Default=0

➤ Example

//초기 상태로 변경하기 mySheet.Reset();

➤ 제공 버전

7.0.0.0

ReturnCellData Method

➤ 기능

조회된 데이터의 특정 셀의 값을 트랜잭션이 발생하기 이전의 조회된 데이터 되돌린다.

조회된 데이터로 되돌리지 못하는 경우는 다음과 같다.

1. 조회된 데이터가 아닌 경우, 예를 들면 입력 상태의 데이터
2. Status 컬럼이 존재하지 않는 경우
3. 다음의 데이터 타입인 경우 : Status, Seq

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

➤ Syntax

Syntax	ObjId. ReturnCellData (Row,Col)
--------	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	데이터 셀의 Row Index
Col	Long / String	필수	데이터 셀의 Column Index 또는 SaveName

➤ Example

```
//초기 상태로 변경하기
mySheet.ReturnCellData(Row,Col);
```

➤ 제공 버전

7.0.0.0



ReturnColumnPos Method

➤ 기능

이동된 컬럼의 위치를 처음 위치로 되돌린다.

➤ Syntax

Syntax	ObjId.ReturnColumnPos()
--------	-------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
없음			

➤ Example

//초기 위치로 되돌린다. mySheet.ReturnColumnPos();
--

➤ 제공 버전

7.0.0.0



ReturnData Method

➤ 기능

특정 행의 데이터를 조회 상태의 문자열로 변경한다.

조회된 데이터로 되돌리지 못하는 경우는 다음과 같다

1. 조회된 데이터가 아닌 경우, 예를 들면 입력 상태의 데이터
2. 상태컬럼이 존재하지 않는 경우

➤ Syntax

Syntax	ObjId.ReturnData(Row)
--------	-----------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	데이터 행의 Row Index

➤ Example

//초기 상태로 변경하기 mySheet.ReturnData(2);

➤ 제공 버전

7.0.0.0



RowBackColor Method

➤ 기능

행 전체의 배경색을 설정하거나 확인한다. 데이터 영역의 배경색만 처리한다. 행이 존재하지 않는 경우 에러메시지 없이 배경색 설정은 취소되고, 확인 시 0이 반환된다.

색상 설정은 WebColor 값을 이용하여 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetRowBackColor (Row)
--------	-----	-------------------------------------

➤ Info

Return	String, 배경 색상 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	Row Index

➤ Example

//1행의 배경색을 확인한다. mySheet.GetRowBackColor(1);

➤ Syntax

Syntax	Set	ObjId. SetRowBackColor (Row,BackColor)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	Row Index
BackColor	String	필수	WebColor 색상 값

➤ Example



```
//1행의 배경색을 회색으로 설정한다.  
mySheet.SetRowBackColor(1,"#C0C0C0");  
  
//3행의 배경색을 빨강색으로 설정한다.  
mySheet.SetRowBackColor(1,"#FF0000");  
//2행 배경색을 1행 배경색으로 설정한다.  
mySheet.SetRowBackColor(2, mySheet.GetRowBackColor(1));
```

➤ **제공 버전**

7.0.0.0



RowBackColorD Method

➤ 기능

각행의 트랜잭션 상태가 삭제인 행의 배경색을 확인하거나 설정한다.

메서드가 적용되면 기존에 시트위에 존재하는 데이터의 색상은 변경되지 않고, 새로 추가되는 항목에 한하여 배경색이 적용된다.

색상 설정은 WebColor 값을 이용하여 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetRowBackColorD ()
--------	-----	-----------------------------------

➤ Info

Return	String, 설정색상 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//트랜잭션상태가 삭제인 행의 배경색을 확인한다.
mySheet.GetRowBackColorD();
```

➤ Syntax

Syntax	Set	ObjId. SetRowBackColorD (BackColor)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
BackColor	String	필수	WebColor값

➤ Example

```
//트랜잭션상태가 삭제인 행의 배경색을 회색으로 설정한다.
mySheet.SetRowBackColorD("#C0C0C0");
```



```
//트렌젝션상태가 삭제인 행을 1행 배경색으로 설정한다.  
mySheet.SetRowBackColorD(mySheet.GetRowBackColor(1));
```

- **제공 버전**
7.0.0.0



RowBackColorI Method

➤ 기능

각행의 트랜잭션 상태가 입력인 행의 배경색을 확인하거나 설정한다.

메서드가 적용되면 기존에 시트위에 존재하는 데이터의 색상은 변경되지 않고, 새로 추가되는 항목에 한하여 배경색이 적용된다.

➤ Syntax

Syntax	Get	ObjId. GetRowBackColorI ()
--------	-----	-----------------------------------

➤ Info

Return	String, 설정 색상 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
BackColor	String	필수	WebColor 색상 값

➤ Example

```
//트랜잭션상태가 입력인 행의 배경색을 확인한다.  
mySheet.GetRowBackColorI();
```

➤ Syntax

Syntax	Set	ObjId. SetRowBackColorI (BackColor)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
BackColor	String	필수	WebColor 색상 값

➤ Example

```
//트랜잭션상태가 입력인 행의 배경색을 회색으로 설정한다.  
mySheet.SetRowBackColorI("#C0C0C0");
```



```
//트렌젝션상태가 입력인 행을 1행 배경색으로 설정한다.  
mySheet.SetRowBackColorl(mySheet.GetRowBackColor(1));
```

- **제공 버전**
7.0.0.0



RowBackColorU Method

➤ 기능

각행의 트랜잭션 상태가 수정인 행의 배경색을 확인하거나 설정한다
메서드가 적용되면 기존에 시트위에 존재하는 데이터의 색상은 변경되지 않고,
새로 추가되는 항목에 한하여 배경색이 적용된다.

➤ Syntax

Syntax	Get	ObjId. GetRowBackColorU ()
--------	-----	-----------------------------------

➤ Info

Return	String, 설정색상 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
BackColor	String	필수	WebColor 색상 값

➤ Example

```
//트랜잭션상태가 수정인 행의 배경색을 확인한다.  
mySheet.GetRowBackColorU();
```

➤ Syntax

Syntax	Set	ObjId. SetRowBackColorU (BackColor)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
BackColor	String	필수	WebColor 색상 값

➤ Example

```
//트랜잭션상태가 수정인 행의 배경색을 회색으로 설정한다.  
mySheet.SetRowBackColorU("#C0C0C0");
```



//트렌젝션상태가 수정인 행을 1행 배경색으로 설정한다.

```
mySheet.SetRowBackColorU(mySheet.GetRowBackColor(1));
```

➤ **제공 버전**

7.0.0.0



RowCount Method

➤ 기능

전체 데이터 행 개수를 확인한다.

Status 값을 설정하지 않은 경우 조회된 데이터 행 개수와 신규 입력된 행까지 포함한 전체 데이터 행 개수를 확인한다.

Status 값에 따라 조회/입력/수정/삭제 각 상태의 행 개수를 확인 할 수 있다.

➤ Syntax

Syntax	ObjId.RowCount([Status])
--------	--------------------------

➤ Info

Return	Long, 전체 또는 특정 트랜잭션 상태의 데이터행 개수		
Parameter	Type	필수여부	설 명
Status	String	선택	트랜잭션 코드. Default="전체건수"

➤ Example

```
alert("전체 건수 는 " + mySheet.RowCount() + " 건입니다. ");  
alert("조회 건수 는 " + mySheet.RowCount("R") + " 건입니다. ");  
alert("입력 건수 는 " + mySheet.RowCount("I") + " 건입니다. ");  
alert("수정 건수 는 " + mySheet.RowCount("U") + " 건입니다. ");  
alert("삭제 건수 는 " + mySheet.RowCount("D") + " 건입니다. ");
```

➤ 제공 버전

7.0.0.0

RowData Method

➤ 기능

행의 데이터를 각 컬럼의 SaveName을 이용하여 Json 객체로 생성하여 반환하거나 설정한다.

2개행 이상의 단위데이터행 구조인 경우 단위데이터행 전체를 반환하거나 설정한다.

➤ Syntax

Syntax	ObjId. GetRowData (Row)
--------	--------------------------------

➤ Info

Return	Object, 해당 행의 데이터 객체		
Parameter	Type	필수여부	설 명
Row	Number	필수	대상 행의 Index

➤ Example

```
// 1행의 Json 객체를 가져온다.
var rowJosn = mySheet.GetRowData(1);
```

➤ Syntax

Syntax	ObjId. SetRowData (Row, Data, Opt)
--------	---

➤ Info

Return	Object, 해당 행의 데이터 객체		
Parameter	Type	필수여부	설 명
Row	Number	필수	대상 행 또는 추가할 위치의 Row Index
Data	Object/ String	필수	Json 형태의 행 데이터 객체 또는 문자열
Opt.Add	Boolean	선택	행 추가 여부 (Default=0)



Opt.Level	Number	선택	트리구조에서 행 추가인 경우 추가될 행의 트리 레벨 값
Opt.Event.	Boolean	선택	셀값 변경시 OnChange 이벤트 발생 여부 (Default = 1)
Opt.StatusMode	Number	선택	상태값 처리 모드 (Default = 1) 0 : 대상 행의 상태값을 변경하지 않음 1 : 인자의 상태값을 무시하고 대상 행의 데이터 변경에 따른 상태값 처리 2 : 인자의 상태값을 그대로 적용

➤ **Example**

```
// 1행을 해당 Json 객체로 설정한다.  
var data = {sName:"홍길동", sAge:20};  
mySheet.SetRowData(1, data);  
  
// 1행의 데이터를 2행에 적용  
mySheet.SetRowData(2, mySheet.GetRowData(1));
```

➤ **제공 버전**

7.0.5.0	
----------------	--



(주) 아이비 리더스

Inno Biz Leaders Co., Ltd.



RowDelete Method

➤ 기능

특정의 단일 또는 다중 데이터 행을 삭제 한다.

Confirm 인자를 1로 설정한 경우 삭제 전 확인 메시지를 표시 한다.

Row 인자를 설정하지 않은 경우 현재 선택되어 있는 행에 대해서 기능을 수행 한다.

➤ Syntax

Syntax	ObjId. RowDelete ([Row],[Confirm])
--------	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Number/String	선택	삭제할 행의 Row Index, 또는 행의 Index를 " " 구분자로 연결한 문자열 Default ="현재 선택 행"
Confirm	Boolean	선택	삭제 전 메시지 표시 여부 Default=0

➤ Example

//확인 메시지 없이 1행 삭제하기 mySheet.RowDelete(1, 0); // 3, 7, 10번 행 삭제하기 mySheet.RowDelete("3 7 10");
--

➤ 제공 버전

7.0.0.0

RowDraggable Method

➤ 기능

특정 행의 마우스 드래그 가능여부를 설정하거나 확인 한다.

설정 값에 따른 처리방식은 아래와 같다.

설정값	내용
0	행 드래그 불가
1 (Default)	행 드래그 가능

➤ Syntax

Syntax	Get	ObjId. GetRowDraggable (row)
--------	-----	-------------------------------------

➤ Info

Return	Boolean, 설정된 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
row	Long	선택	행의 Row Index

➤ Example

```
// Drag 가능여부 설정 값 확인
var drag = mySheet.GetRowDraggable(3);
```

➤ Syntax

Syntax	Set	ObjId. SetRowDraggable (row, drag)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
row	Long	선택	행의 Row Index
drag	Boolean	필수	드래그 가능여부 설정 값 (Default=1)



➤ **Example**

```
// 3행의 Drag 가능여부 설정 (마우스 드래킹시 행 드래킹 가능여부 설정)
mySheet.SetRowDraggable(3, 1); // 드래그 가능
mySheet.SetRowDraggable(3, 0); // 드래그 불가
```

➤ **제공 버전**

7.0.0.0

RowEditable Method

➤ **기능**

특정 행의 Edit 가능 여부를 확인하거나 설정한다.

행의 Edit 가능 여부는 전체 Edit 가능 여부가 가능일 때 변경 가능하다.

단, ColEditable 설정값이 불가인 경우는 RowEditable 설정이 무시된다

행의 Edit 가능 여부는 다음과 같이 결정된다.

Editable	ColEditable	RowEditable	CellEditable	셀의 Edit 가능 여부
불가	무관	무관	무관	불가
가능	불가	불가	가능/불가	가능/불가
가능	가능	가능/불가	가능/불가	가능/불가

➤ **Syntax**

Syntax	Get	ObjId. GetRowEditable (Row)
--------	-----	------------------------------------

➤ **Info**

Return	Boolean, Edit 가능 여부 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 행의 Row Index

➤ **Example**

```
//1행의 Edit 가능 여부를 확인한다.
mySheet.GetRowEditable(1,0);
```



➤ **Syntax**

Syntax	Set	ObjId. SetRowEditable (Row, Editable)
--------	-----	--

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 행의 Row Index
Editable	Boolean	필수	행의 Edit 가능 여부 (Set일때만 사용)

➤ **Example**

//1행의 Edit 가능 여부를 0으로 설정한다. mySheet.SetRowEditable(1,0);

➤ **제공 버전**

7.0.0.0



RowExpanded Method

➤ 기능

트리 형태 일 때 행의 Child Level이 펼쳐져 있는 상태인지 여부를 확인하거나 펼침 여부를 설정한다.

펼침 여부 설정시(Set Method) Event 인자를 0(false)로 설정하는 경우 관련 이벤트인 OnBeforeExpand, OnAfterExpand 이벤트는 발생하지 않는다.

➤ Syntax

Syntax	Get	ObjId. GetRowExpanded (Row)
--------	-----	------------------------------------

➤ Info

Return	Boolean, 행의 자식행들의 펼쳐짐 여부 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	Row Index

➤ Example

```
//2행의 펼쳐져 있지 않다면, 2행의 트리를 펼친다.
If(mySheet.GetRowExpanded(2) == 0){
    mySheet.SetRowExpanded(2, 1);
}
```

➤ Syntax

Syntax	Set	ObjId. SetRowExpanded (Row, Expand, [Event])
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Number	필수	대상 행의 Index
Expand	Boolean	필수	자식행의 펼쳐짐 여부



Event	Boolean	선택	관련 이벤트 발생여부 Default = 1
-------	---------	----	----------------------------

➤ **Example**

```
//2행의 펼쳐져 있지 않다면, 2행의 트리를 펼친다.  
If(mySheet.GetRowExpanded(2) == 0){  
    mySheet.SetRowExpanded(2, 1);  
}
```

➤ **제공 버전**

7.0.0.0	
7.0.4.2	Set 메소드의 이벤트 발생여부 인자 추가



RowFontColor Method

➤ 기능

행 전체의 글자색을 설정하거나 확인한다.

➤ Syntax

Syntax	Get	ObjId. GetRowFontColor (Row)
--------	-----	-------------------------------------

➤ Info

Return	String, 설정 색상 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	Row Index

➤ Example

//1행의 글자색을 확인한다. mySheet.GetRowFontColor(1);

➤ Syntax

Syntax	Set	ObjId. SetRowFontColor (Row,Color)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	Row Index
Color	String	필수	WebColor 색상 값

➤ Example

//1행의 글자색을 회색으로 설정한다. mySheet.SetRowFontColor(1, "192,192,192"); //2행 글자색을 1행 글자색으로 설정한다.



```
mySheet.SetRowFontColor(2,mySheet.GetRowFontColor(1));
```

- **제공 버전**
7.0.0.0



RowHeight Method

➤ 기능

특정 행의 높이를 설정하거나 확인한다.

픽셀 단위의 높이를 설정할 수 있고, 행이 존재하지 않는 경우 에러메시지는 표시하지 않지만 처리는 취소된다.

➤ Syntax

Syntax	Get	ObjId. GetRowHeight (Row)
--------	-----	----------------------------------

➤ Info

Return	Integer, 현재 행 높이 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	설정할 행 Index

➤ Example

//1행의 높이를 확인한다. mySheet.GetRowHeight(1);

➤ Syntax

Syntax	Set	ObjId. SetRowHeight (Row, Height)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	설정할 행 Index
Height	Integer	필수	설정할 행 높이

➤ Example

//50 픽셀로 높이 수정 mySheet.SetRowHeight(1, 50);
--



```
//3행의 높이를 2행의 높이와 동일하게 변경  
mySheet.SetRowHeight(3, mySheet.GetRowHeight(2));
```

- **제공 버전**
7.0.0.0



RowHeightMax Method

➤ 기능

모든 데이터 행의 최대 높이를 확인하거나 설정한다.

데이터 행의 높이를 자동으로 설정 한 경우에만 적용이 되며 DataRowHeight 보다 작게 설정한 경우 DataRowHeight의 값으로 설정 된다.

이 기능은 행의 높이를 자동으로 설정 하고 특정 높이 이상 커지지 않게 하고 자 하는 경우에 사용한다.

➤ Syntax

Syntax	Get	ObjId. GetRowHeightMax()
--------	-----	---------------------------------

➤ Info

Return	Integer, 행의 최대 높이 (Get Method인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//설정한 최소 높이값 확인한다.  
mySheet.GetRowHeightMax();
```

➤ Syntax

Syntax	Set	ObjId. SetRowHeightMax (MaxHeight)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
MaxHeight	Integer	필수	해당 행에 설정 할 최대 높이값

➤ Example



```
//최소 높이를 50픽셀로 설정한다.  
mySheet.SetRowHeightMax(50);
```

- **제공 버전**
7.0.0.0



RowHeightMin Method

➤ 기능

모든 행의 최소 높이를 확인하거나 설정한다. 최소 높이는 5픽셀 이하로 할 수 없으므로 5픽셀 이상의 값을 설정하도록 한다. 기본값은 21픽셀로 설정되어 있다.

➤ Syntax

Syntax	Get	ObjId. GetRowHeightMin()
--------	-----	---------------------------------

➤ Info

Return	Integer, 행의 최소 높이 (Get Method인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//설정한 최소 높이값 확인한다.
mySheet.GetRowHeightMin();
```

➤ Syntax

Syntax	Set	ObjId. SetRowHeightMin (MinHeight)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
MinHeight	Integer	필수	행에 설정 할 최소 높이값

➤ Example

```
//최소 높이를 10픽셀로 설정한다.
mySheet.SetRowHeightMin(10);
```




- 제공 버전
7.0.0.0



RowHidden Method

➤ 기능

행의 숨기여부를 설정하거나 확인한다.

'|' 구분자로 Row Index를 연결하여 조합하면 다수행의 숨김여부를 설정할 수 있다. (단, 다수행에 대해 Read 기능은 지원되지 않는다.)

➤ Syntax

Syntax	Get	ObjId. GetRowHidden (Row)
--------	-----	----------------------------------

➤ Info

Return	Boolean, 설정 값 (Get : 1 이면 숨기 상태, 0이면 보이는 상태)		
Parameter	Type	필수여부	설 명
Row	Long /String	필수	특정 행의 Row Index 혹은 구분자 " "로 연결된 문자열 (Set에서만 사용)

➤ Example

//1행이 숨겨진 여부를 확인하여 숨겨진 경우 표시되도록 설정한다. If(mySheet.GetRowHidden(1)){ mySheet.SetRowHidden(1,0); }
--

➤ Syntax

Syntax	Set	ObjId. SetRowHidden (Row,Hidden)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long /String	필수	특정 행의 Row Index 혹은 구분자 " "로 연결된 문자열



			(Set에서만 사용)
Hidden	Boolean	필수	숨김여부

➤ **Example**

//1행이 숨겨진 여부를 확인하여 숨겨진 경우 표시되도록 설정한다.

```
If(mySheet.GetRowHidden(1)){  
    mySheet.SetRowHidden(1,0);  
}
```

//다수의 행을 동시에 숨기는 경우

```
mySheet.SetRowHidden("2|5|7|10", 1);
```

➤ **제공 버전**

7.0.0.0



RowLevel Method

➤ 기능

행의 트리 레벨을 설정하거나 확인한다.

트리가 존재하는 경우 행의 트리 레벨을 설정하거나 확인하며 트리 레벨은 0 부터 시작한다.

➤ Syntax

Syntax	Get	ObjId. GetRowLevel (Row)
--------	-----	---------------------------------

➤ Info

Return	Integer, 현재 행의 트리 레벨 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 행의 Row Index

➤ Example

```
//1행의 트리 레벨을 확인한다.
mySheet.GetRowLevel(1);
```

➤ Syntax

Syntax	Set	ObjId. SetRowLevel (Row,Level)
--------	-----	---------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 행의 Row Index
Level	Integer	필수	해당 행의 트리 레벨값(Set)

➤ Example

```
//트리 레벨을 1 증가 시킨다.
mySheet.SetRowLevel(1,mySheet.GetRowLevel(1) + 1);
```



- 제공 버전
7.0.0.0.



RowMerge Method

➤ 기능

행의 가로 머지 여부를 확인하거나 설정한다.

가로 머지가 가능하기 위해서는 전체 머지가 가능하여야 하고, 해당 데이터 행이 존재해야 한다.

➤ Syntax

Syntax	Get	ObjId. GetRowMerge (Row)
--------	-----	---------------------------------

➤ Info

Return	Boolean, 현재 행의 가로 머지 여부 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 행의 Row Index

➤ Example

```
// 첫번째 행의 가로 머지를 허용여부를 확인한다.
mySheet.GetRowMerge(1);
```

➤ Syntax

Syntax	Set	ObjId. SetRowMerge (Row, Merge)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 행의 Row Index
Merge	Boolean	필수	가로머지 허용 여부

➤ Example

```
// 첫번째 행의 가로 머지를 허용하도록 설정한다.
mySheet.SetRowMerge(1, 1);
```



- 제공 버전
7.0.0.0.



RowSaveStr Method

➤ 기능

행의 데이터를 각 컬럼의 SaveName을 이용하여 저장 시 사용되는 Query String 형태로 조합하여 반환한다.

SaveName1=Value1&SaveName2=Value2&...

위와 같은 형태로 반환하며, 한글은 UrlEncoding 되어 반환한다.

➤ Syntax

Syntax	ObjId. RowSaveStr (Row, [Opt])
--------	---------------------------------------

➤ Info

Return	String, 행의 데이터		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 행의 Row Index
Opt.ValidKeyField	Boolean	선택	KeyField 체크 여부 (Default=1)
Opt.ValidFullInput	Boolean	선택	FullInput 체크 여부 (Default=1)

➤ Example

```
// 1행의 SaveString을 가져온다.  
var RowParam = mySheet.RowSaveStr(1);
```

➤ 제공 버전

7.0.0.0	
7.0.12.4	ValidKeyField, ValidFullInput 인자 속성 추가



RowSumable Method

➤ 기능

행의 합계 계산 대상인지 여부를 확인하거나 설정한다.

해당 속성에 대한 설정은 합계, 소계, 트리소계에 적용 된다.

➤ Syntax

Syntax	Get	ObjId. GetRowSumable (Row)
--------	-----	-----------------------------------

➤ Info

Return	Boolean, 설정 값		
Parameter	Type	필수여부	설 명
Row	Number	필수	대상 행의 Index

➤ Example

```
//1행에 대한 합계 계산 대상 여부를 확인 한다..
var sumable = mySheet.GetRowSumable(1);
```

➤ Syntax

Syntax	Set	ObjId. SetRowSumable (Row, Sum)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Number	필수	대상 행의 Index
Sum	Boolean	필수	합계 계산 포함 여부

➤ Example

```
//5행을 합계계산에서 제외 한다.
mySheet.SetRowSumable(5, 0);
```



}

- **제공 버전**
7.0.0.0



RowTop Method

➤ 기능

특정 행의 상단 시작위치의 offsetTop 값을 확인 한다.

Row 인자 설정이 잘못된 경우는 별도 처리없이 -1을 리턴 한다.

시트 컨테이너(table)의 최상단위치를 기준으로하여 처리 된다. (기준점 : 0)

동일한 행인 경우에도 세로 스크롤 위치에 따라서 반환값이 달라질 수 있다.

➤ Syntax

Syntax	ObjId. RowTop (Row)
--------	----------------------------

➤ Info

Type	Long, 행의 상단 위치	
Parameter	Type	설 명
Row	Long	Row Index

➤ Example

//행의 상단 위치를 파악한다. var iTop = mySheet.RowTop(1);
--

➤ 제공 버전

7.0.0.0



SaveNameCol Method

➤ 기능

InitColumns에서 설정된 SaveName을 이용하여 해당하는 컬럼 번호를 확인한다. SaveName에 해당하는 컬럼이 존재하지 않는 경우 -1을 반환한다.

➤ Syntax

Syntax	ObjId. SaveNameCol (SaveName)
--------	--------------------------------------

➤ Info

Return	Long, 컬럼의 Index		
Parameter	Type	필수여부	설 명
SaveName	String	필수	저장 변수명

➤ Example

```
// 변수명으로 컬럼 번호를 가져온다.  
var Col = mySheet.SaveNameCol("stockNm");
```

➤ 제공 버전

7.0.0.0

SavingImage Method

➤ 기능

저장 중 대기 이미지 파일의 위치를 설정하거나 확인한다.

이 속성은 기본적으로 제공하는 처리 중 대기 이미지를 사용자가 원하는 이미지로 변경설정 가능하기 위한 속성이다.

➤ Syntax

Syntax	Get	ObjId. GetSavingImage()
--------	-----	--------------------------------

➤ Info

Return	String, 현재 설정값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//현재 설정되어있는 저장 중 대기 이미지 경로를 확인한다.
Alert(mySheet.GetSavingImage());
```

➤ Syntax

Syntax	Set	ObjId. SetSavingImage (Url)
--------	-----	------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Url	String	필수	이미지 URL

➤ Example

```
//저장 중 대기 이미지를 변경한다.
mySheet.SetSavingImage( "/sheet/imgSave.gif");
```



- 제공 버전
7.0.0.0

ScrollInfoFormat Method

➤ 기능

세로 스크롤 지연 처리 시간동안 표시되는 메시지 포맷을 설정하거나 확인 한다. 설정 시 아래의 예약어와 컬럼 정보의 조합으로 일반 텍스트 형식과 간단한 마크업 태그 형식으로 설정 할수 있다. 마크업 태그 형식으로 설정시 복잡도에 따라 성능에 영향을 줄 수 있기 때문에 가급적이면 간단한 형식으로 사용하여야 한다. 사용 가능한 예약어는 아래와 같으며, 별도 설정이 없는 경우 기본값인 "TOPROW / TOTALROWS" 로 처리 된다.

예약어	설명
TOTALROWS	전체 건수
TOPROW	스크롤 이동시 상단에 위치할 행의 Index
컬럼의 Index 또는 SaveName	TOPRow 의 해당 컬럼의 셀 정보 컬럼정보 사용시 해당 값의 앞뒤에 " " 문자를 사용하여야 한다. (예) 3 , sText}

➤ Syntax

Syntax	Get	ObjId. GetScrollInfoFormat ()
--------	-----	--------------------------------------

➤ Info

Return	String, 현재 설정값		
Parameter	Type	필수여부	설 명

➤ Example

```
//현재 설정되어있는 스크롤 정보 메시지 포맷을 확인 한다.
Console.log("ScrollInfoFormat :", mySheet.GetScrollInfoFormat());
```

➤ Syntax

Syntax	Set	ObjId. SetScrollInfoFormat (Format)
--------	-----	--



➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Format	String	필수	설정할 메시지 포맷 문자열

➤ **Example**

```
// 기본값에 컬럼인덱스 3번의 셀 정보 표시 설정
mySheet.SetScrollInfoFormat("[TOPROW / TOTALROWS] |3|");

// 기본값에 컬럼의 SaveName이 "sName"인 셀 정보 표시 설정
mySheet.SetScrollInfoFormat("[TOPROW / TOTALROWS] |sName|");

// 간단한 마크업 태그를 이용한 설정 (셀 정보에 blue 색상 적용)
var format = "[TOPROW / TOTALROWS] <font color='blue'>|sName|</font>";
mySheet.SetScrollInfoFormat(format);
```

➤ **제공 버전**

7.0.13.11	
-----------	--

ScrollInfoPosition Method

➤ 기능

세로 스크롤 지연 처리 시간동안 표시되는 메시지의 출력 위치를 설정하거나 확인 한다. 설정 가능한 값은 아래와 같다.

예약어	설명
none	표시 하지 않음
center	화면 중앙에 표시 (Default)
scroll	스크롤 위치에 표시

➤ Syntax

Syntax	Get	ObjId. GetScrollInfoPosition()
--------	-----	---------------------------------------

➤ Info

Return	String, 현재 설정값		
Parameter	Type	필수여부	설 명

➤ Example

```
//현재 설정되어있는 스크롤 정보 메시지 포맷을 확인 한다.
Console.log("ScrollInfoPosition :", mySheet.GetScrollInfoPosition());
```

➤ Syntax

Syntax	Set	ObjId. SetScrollInfoPosition(Pos)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Pos	String	필수	메시지 출력 위치 (Default:center)



➤ **Example**

```
// 메시지 출력을 스크롤 위치에 표시되도록 설정  
mySheet.SetScrollInfoPosition("scroll");
```

➤ **제공 버전**

7.0.13.11	
------------------	--

SearchingImage Method

➤ 기능

조회 중 대기 이미지 파일의 위치를 설정하거나 확인한다.

이 속성은 기본적으로 제공하는 처리 중 대기 이미지를 사용자가 원하는 이미지로 변경설정 가능하기 위한 속성이다.

➤ Syntax

Syntax	Get	ObjId. GetSearchingImage()
--------	-----	-----------------------------------

➤ Info

Return	String, 현재 설정값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//현재 설정되어있는 조회 중 대기 이미지 경로를 확인한다.
Alert(mySheet.GetSearchingImage());
```

➤ Syntax

Syntax	Set	ObjId. SetSearchingImage (Url)
--------	-----	---------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Url	String	필수	이미지 URL

➤ Example

```
//저장 중 대기 이미지를 변경한다.
mySheet.SetSearchingImage( "/sheet/imgSearch.gif");
```



- 제공 버전
7.0.0.0



SearchRows Method

➤ 기능

조회 XML을 통해서 조회된 전체 행 개수를 확인한다.

➤ Syntax

Syntax	ObjId. SearchRows()
--------	----------------------------

➤ Info

Return	Long, 조회된 전체 행 개수		
Parameter	Type	필수여부	설 명
없음			

➤ Example

//행 개수 확인 alert(mySheet.SearchRows());

➤ 제공 버전

7.0.0.0

SelectCell Method

➤ 기능

특정 셀을 선택한다. 이 함수를 이용할 경우 Edit 가능한 경우 Edit 가능한 경우 Edit 속성을 1로 설정하면 셀이 선택됨과 동시에 편집모드가 된다. Edit 가능한 필요하지 않다면 Edit 속성을 0으로 설정하여 호출한다.

EditText 속성은 Edit 속성이 1일 때 사용되는 값으로 편집 모드시 표시할 문자열을 설정 한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

➤ Syntax

Syntax	ObjId. SelectCell (Row, Col, [Opt])
--------	--

➤ Info

Return	None		
Parameter	Type	필수 여부	설 명
Row	Number	필수	선택할 셀의 Row Index
Col	Number / String	필수	선택할 셀의 Column Index 또는 SaveName
Opt.Edit	Boolean	선택	셀 선택시 편집모드 여부를 설정 Default=0
Opt.EditText	String	선택	Edit 속성이 1일때 편집상태의 문자열 설정. Defulat= ""
Opt.SelectEditText	Boolean	선택	편집시 기존 값에 대한 select 처리 여부



			Default = 1
Opt.Event	Boolean	선택	OnSelectCell 이벤트 발생 여부 Default = 1
Opt.Focus	Boolean	선택	셀 선택후 포커스 처리 여부 Default = 1

➤ **Example**

```
// 2행을 선택하고 그 행 아래에 새로운 행 생성하기  
mySheet.SelectCell(2, 0);  
mySheet.DataInsert();
```

➤ **제공 버전**

7.0.0.0	
7.0.4.4	편집시 기존 값에 대한 select 처리 여부 설정 기능 추가 (인자 구조 설정방식 변경)
7.0.13.14	Fous 인자 속성 추가



SelectCol Method

➤ 기능

현재 선택된 셀의 컬럼 Index를 확인하거나 설정한다.

SelectRow 속성과 함께 사용하며, 두 속성을 모두 사용한 SelectCell 함수를 이용할 수 있다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 모든행에 대해서 처리를 하고 SaveName으로 설정할 경우 해당 SaveName이 포함된 행에 대해서만 처리 한다.

➤ Syntax

Syntax	Get	ObjId. GetSelectCol()
--------	-----	------------------------------

➤ Info

Return	Number, 현재 선택된 컬럼 Index (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
없음			

➤ Example

//선택된 컬럼을 확인한다. Alert(mySheet.GetSelectCol());



➤ **Syntax**

Syntax	Set	ObjId. SetSelectCol (Col, [UseEvent])
--------	-----	--

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Col	Number / String	필수	현재 선택된 셀의 Column Index 또는 SaveName
UseEvent	Boolean	선택	OnSelectCell 이벤트 발생 여부 Default = 1 (이벤트 발생)

➤ **Example**

//인덱스가 2인 컬럼을 선택한다. mySheet.SetSelectCol(2);

➤ **제공 버전**

7.0.0.0



SelectionMode Method

➤ 기능

셀 선택 모드를 설정하거나 확인한다. 설정가능한 종류는 아래와 같다.

종류	설명
0	셀 단위 선택
1	행 단위 선택
3	Ctrl 키를 이용하여 연결되지 않은 다중의 행을 선택 선택된 행번호는 GetSelectionRows() 함수 이용 확인

➤ Syntax

Syntax	Get	ObjId. GetSelectionMode()
--------	-----	----------------------------------

➤ Info

Return	Integer, 현재 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

// 현재 설정되어 있는 선택 모드를 확인한다. Alert(mySheet.GetSelectionMode());
--

➤ Syntax

Syntax	Set	ObjId. SetSelectionMode (Mode)
--------	-----	---------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Mode	Integer	필수	선택 모드 종류 (0 : 셀단위, 1 : 행단위)



➤ **Example**

```
//셀 단위 선택으로 설정한다.  
mySheet.SetSelectionMode(0);
```

➤ **제공 버전**
7.0.0.0



SelectionSummaryInfoElement Method

➤ 기능

선택 요약 정보를 시트 외부의 DOM 엘리먼트에 표현하기 위한 설정을 하거나 설정된 엘리먼트를 확인 한다.

설정이 가능한 DOM 엘리먼트는 input 과 div, span 등의 innerText 를 지원하는 DOM 엘리먼트이다.

➤ Syntax

Syntax	Get	ObjId. GetSelectionSummaryInfoElement ()
--------	-----	---

➤ Info

Return	object, 설정된 DOM 엘리먼트 객체		
Parameter	Type	필수여부	설 명

➤ Example

```
// 선택요약정보를 표시할 DOM 엘리먼트 객체를 확인 한다.
var elem = mySheet.GetSelectionSummaryInfoElement();
```

➤ Syntax

Syntax	Set	ObjId. SetSelectionSummaryInfoElement (Element)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Element	Object/ String	필수	설정할 DOM 엘리먼트 객체 또는 Id

➤ Example



```
// 선택요약정보를 표시할 DOM 엘리먼트를 설정한다.  
// ("summaryElem" 아이디의 DOM 엘리먼트가 생성되어 있어야 함)  
mySheet.SetSelectionSummaryInfoElement("summaryElem");  
}
```

➤ **제공 버전**

7.0.0.0



SelectRow Method

➤ 기능

현재 선택된 셀의 행 Index를 확인하거나 설정한다.

SelectCol 속성과 함께 사용하며, 두 속성을 모두 사용한 SelectCell 함수를 이용할 수 있다.

➤ Syntax

Syntax	Get	ObjId. GetSelectRow()
--------	-----	------------------------------

➤ Info

Return	Long, 현재 선택된 행 Index (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

//선택된 행을 확인한다. Alert(mySheet.GetSelectRow());
--

➤ Syntax

Syntax	Set	ObjId. SetSelectRow (Row, [UseEvent])
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	선택할 행 Index
UseEvent	Boolean	선택	OnSelectCell 이벤트 발생 여부 Default = 1 (이벤트 발생)

➤ Example



```
//인덱스가 2인 행을 선택한다.  
mySheet.SetSelectRow(2);
```

- **제공 버전**
7.0.0.0



SendComboData Method

➤ 기능

데이터 타입이 Combo일때 서버로 전송되는 데이터를 설정하거나 확인한다.
컬럼에 대해서 저장 메서드를 통하여 데이터가 서버로 전달될 때, 콤보에 설정
했던 Code가 전달된다. 하지만 이 메서드를 사용하면 Code 대신에 눈에 보이
는 Text를 전달하거나 Code와 Text를 동시에 전달할 수도 있다.

➤ Syntax

Syntax	Get	ObjId. GetSendComboData (DataRow, Col)
--------	-----	---

➤ Info

Return	String, 현재 선택된 콤보 컬럼의 전송될 데이터 형식 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
DataRow	Number	필수	단위 데이터 행 Index
Col	Number / String	필수	대상 컬럼의 Index 또는 SaveName

➤ Example

//3컬럼에 대해서 저장시 콤보 전송 형식을 알아온다. mySheet.GetSendComboData(0,3);
--

➤ Syntax

Syntax	Set	ObjId. SetSendComboData (DataRow, Col, Type)
--------	-----	---

➤ Info

Return	String, 현재 선택된 콤보 컬럼의 전송될 데이터 형식 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
DataRow	Long	필수	단위 데이터 행 Index



Col	Long / String	필수	특정 컬럼의 Column Index 또는 SaveName
Type	String	선택	서버 전송 옵션 설정 ("Text" 또는 "code"), Default="code"

➤ **Example**

//3컬럼에 대해서 저장시 Code 와 Text 값이 같이 전달되도록 한다.

```
mySheet.SetSendComboData(0,3,"Code|Text");
```

//Dept 컬럼에 대해서 저장시 Code 대신 Text 값이 전달되도록 한다.

```
mySheet.SendComboData(0,"Dept","Text");
```

➤ **제공 버전**

7.0.0.0



SetBlur Method

➤ 기능

시트에 포커스를 뺀다

➤ Syntax

Syntax	ObjId. SetBlur ()
--------	--------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// 시트에서 포커스를 뺀다.  
mySheet.SetBlur();
```

➤ 제공 버전

7.0.0.0

SetCellImageStyle Method

➤ 기능

Image 속성을 사용하는 경우 해당 셀의 이미지에 대한 설정을 한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 첫번째 행의 해당인덱스 셀에 대해서 처리를 하고 SaveName으로 설정할 경우 설정한 행의 단위데이터행 내 해당 SaveName 셀에 대해서 처리한다.

이 함수를 통해 설정이 가능한 속성은 다음과 같다.

속성	타입	설명
Image	Integer	이미지의 경로
ImgAlign	String	이미지의 표시 위치
ImgWidth	Integer	이미지의 너비
ImgHeight	Integer	이미지의 높이

➤ Syntax

Syntax	ObjId. SetCellImageStyle (Row, Col, Style);
--------	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	대상 행의 Index
Col	Logn/String	필수	대상 컬럼의 Index 또는 SaveName
Style	Object	필수	셀의 이미지관련 속성 객체

➤ Example

```
//3번째 행의 3번째 컬럼에 이미지를 왼쪽에 표시.
var style = {Image: "myImage.gif", ImgHeight:20, ImgAlign:"Left"};
mySheet.SetCellImageStyle(3, 3, style);
```



```
//3번째 행의 3번째 컬럼에 ImageList의 1번째 이미지를 오른쪽에 표시.  
mySheet.SetImageList(0, "image0.gif");  
mySheet.SetImageList(1, "image1.gif");  
var style = {Image: 1, ImgAlign:"Right"};  
mySheet.SetCellImageStyle(3, 3, style);
```

➤ **제공 버전**

7.0.0.0

SetColProperty Method

➤ 기능

InitColumns Method를 통해 정의후 특정 컬럼의 속성 정의를 동적으로 변경하여 사용 하고자 하는 경우에 이 메소드를 사용한다. 동적으로 변경된 속성은 데이터 재조회시 적용되므로 데이터 조회 이전에 변경하여야 한다.

이 메소드를 통해 재설정이 가능한 속성은 다음과 같다.

속성	타입	설명
AcceptKeys	String	<p>입력허용키 설정</p> <p>"N" : 숫자만 허용</p> <p>"E" : 영문만 허용</p> <p>"N E" : 숫자, 영문만 허용</p> <p>"N E [사용자정의문자]" : 숫자, 영문과 사용자정의문자로 설정한 문자 허용</p> <p>예) "N [abc!]" 설정 시 숫자와 a, b, c, ! 문자만 입력 가능</p> <p>사용자 정의 문자열은 꼭 대괄호로 묶어서 설정해야 하며 사용자정의문자와 숫자,영문 설정문자의 합집합으로 처리된다.</p> <p>AcceptKeys 와 ExceptKeys를 둘다 설정하고 설정키 내에 동일 문자가 있는 경우 해당 문자는 입력불가 처리된다.</p>
ActionMenu	Object[]	컨텍스트 메뉴
Align	String	데이터의 정렬
AllowNull	Boolean	숫자계열 컬럼에 빈값 허용 여부 설정
ApproximateType	Number	근사값처리방식 (1:반올림, 2:내림, 3:



		올림)
ButtonUrl	String/Number	팝업 버튼의 이미지 경로 또는 이미지 리스트의 Index
CalcLogic	String / Object	컬럼별 계산식 문자열 또는 설정 객체 세부설명은 InitColumns 메소드 참고
Chart	Object	스파크라인 차트 타입에 대한 세부 속성 설정 세부설명은 InitColumns 메소드 참고
ComboCode	String	콤보 리스트의 코드 집합
ComboText	String	콤보 리스트의 문자열 집합
CustomDate	Number	사용자 정의 날짜 사용여부
DefaultValue	String	신규입력시 기본값 설정
Edit	Boolean	편집가능 여부
EditLen	Number	데이터의 입력가능한 글자수
EmptyToReplaceChar	String	빈값인 경우 대체하여 보여줄 문자 설정
EnterMode	Boolean	다중라인 입력모드(MultiLineText)에서 Enter 키 입력에 대한 개행 처리 여부 (Default=0)
ExceptKeys	String	입력제외키 설정 "N" : 숫자 입력 불가 "E" : 영문 입력 불가 "N E" : 숫자, 영문 입력 불가 "N E[사용자정의문자]" : 숫자, 영문과 사용자정의문자로 설정한 문자 입력 불가 예) "N [abc!]" 설정 시 숫자와 a, b, c, ! 문자만 입력 불가 사용자 정의 문자열은 꼭 대괄호로 묶어



		서 설정해야 하며 사용자정의문자와 숫자,영문 설정문자의 합집합으로 처리된다. AcceptKeys 와 ExceptKeys를 둘다 설정하고 설정키 내에 동일 문자가 있는 경우 해당 문자는 입력불가 처리된다.
FalseValue	String	0 이외의 CheckBox 형태 컬럼의 False값 지정. "F" 으로 지정한 경우 0 대신 "F"을 True 값으로 사용 가능.
FitColWidth	Boolean	FitColWidth 메소드 호출시 비율에 의한 너비 재조정 허용 여부
Focus	Boolean	포커스 허용 여부
Format	String	데이터의 Mask 적용 형태
FormatFix	Boolean	GetCellValue 시 GetCellText 값을 반환 할지 여부. 특히 True 로 설정시 저장할 때 Format 이 설정된 상태로 저장한다.
HoverUnderline	Boolean	마우스오버시 언더라인 여부
Image	String	이미지 표현시 Url
ImgHeight	Number	이미지 높이
ImgWidth	Number	이미지 너비
InputCaseSensitive	Number	입력시 영문에 대한 대소구문 자동 치환 처리에 대한 설정 0 : 별도 처리 하지 않음 (기본값) 1 : 대문자로 치환 2 : 소문자로 치환
ItemCode	String	멀티체크 아이템의 코드를 구분자 " "로 연결한 문자열
ItemText	String	멀티체크 아이템의 텍스트를 구분자 " "로 연결한 문자열
KeyField	Boolean	필수 입력 항목



MinWidth	Number	컬럼 최소 가로크기
MaxCheck	Number	멀티체크의 최대 선택 개수 설정
MultiLineText	Boolean	다중라인 입력여부
NumberSort	Boolean	숫자형 Sort 처리 사용 여부
PointCount	Number	컬럼포맷이 Float 인 경우 소수점 이하의 자리수
PopupCode	String	팝업메뉴의 코드 집합
PopupText	String	팝업메뉴의 문자열 집합
Radiolcon	Boolean	데이터 셀내의 체크박스 계열의 이미지 모양 설정
ShowCol	Number	멀티콤보 설정 시 컬럼 값으로 출력할 컬럼 설정
ToolTip	Boolean/String	셀의 풍선도움말 표시 여부 또는 설정할 문자열
Transaction	Boolean	트랜잭션 허용 여부 (Default=1)
TreeCol	Boolean	트리 기준컬럼 설정여부
TrueValue	String	1 이외의 CheckBox 형태 컬럼의 True 값 지정. "M" 으로 지정한 경우 1 대신 "M"을 True 값으로 사용 가능.
Width	Number	컬럼의 너비
ZeroToReplaceChar	String	값이 0인 경우 대체하여 보여줄 문자 설정
StaticPassword	Boolean	값이 0인 경우 데이터 길이만큼 '*'의 수를 표현한다.

각 속성의 상세 설명은 InitColumns Method 설명을 참조 한다.

➤ Syntax

Syntax	ObjId. SetColProperty (DataRow, Col, Prop);
--------	--

➤ Info

Return	None
--------	------

Parameter	Type	필수여부	설 명
DataRow	Number	필수	단위데이터행 Index
Col	Number / String	필수	대상 컬럼의 Index 또는 SaveName
Prop	Object	필수	컬럼의 속성 정의 객체

➤ **Example**

```
// 3번째 컬럼의 콤보 목록을 변경
var info = {ComboText: "신규|진행중|완료 ", ComboCode: " 01|02|03" };
mySheet.SetColProperty(0, 3, info);

// 단위데이터행 1번째, 3번째 컬럼의 콤보 목록을 변경
var info = {ComboText: "신규|진행중|완료 ", ComboCode: " 01|02|03" };
mySheet.SetColProperty(1, 3, info);

// 4번째 컬럼에 허용키, 제외키 설정
var info = {AcceptKeys:"N|E|[$%]",ExceptKeys:"[123ab]"};
mySheet.SetColProperty(0, 4 ,info);
```

➤ **제공 버전**

7.0.0.0	
7.0.4.4	다중라인 입력모드에 대한 EnterMode 속성 추가
7.0.7.2	팝업버튼 사용여부 설정을 위한 PopupButton 속성 추가
7.0.11.0	Chart 속성 추가
7.0.12.2	FitColWidth 속성 추가
7.0.12.3	ItemCode, ItemText, MaxCheck 속성 추가
7.0.13.3	Transaction 속성 추가
7.0.13.16	ActionMenu 속성 추가
7.0.13.22	CalcLogic 속성 추가
7.0.13.24	EmptyToReplaceChar, ZeroToReplaceChar 속성 추가
7.0.13.29	CalcLogic 속성 개선 (Object 타입 설정 기능 추가)
7.0.13.34	NumberSort 속성 추가



7.0.13.53

StaticPassword 속성 추가

SetConfig Method

➤ 기능

시트의 초기화시 필요한 속성을 설정 한다.

초기화 이후 재설정이 필요한 경우에는 반드시 Refresh 인자를 1(true)로 설정 하여야 설정한 속성들이 적용되며, 이 경우 해당 시트의 데이터는 모두 초기화 되므로, 사용시 반드시 조회 이전에 호출 하여야 한다.

➤ Syntax

Syntax	ObjId. SetConfig (Cfg, [Refresh])
--------	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Cfg	Object	필수	전달되는 인자의 형식은 JSON 타입으로 설정 하고자 하는 정보를 JSON 형태로 구성하여 전달한다.
Refresh	Boolean	선택	설정 후 갱신 처리 여부 (Default=0)

< 세부내용 >

설정변수	타입	설명
AutoCloseDialog	boolean	콤보목록, 달력팝업, 액션메뉴 팝업의 자동닫기 여부 설정. (Default=0) 자동닫기 설정시 MouseOut 시점에 자동으로 닫힌다.
AutoFitColWidth	string/ array	FitColWidth에 대한 적용 시점 설정 적용하고자 하는 설정값을 구분자 " "로 연결하여 설정한다. (Default : "")
AutoSumCalcMode	number	별합된 셀에 대한 합계 계산 처리 방법 설정 (Default:0)
CachePageCount	number	페이지 캐싱 개수



ChildPage	number	트리구조에서 자식행의 페이지 단위 개수 (Default=0)						
ColPage	number	한번에 표시할 컬럼의 개수 (Default=사용안함)						
ComboMaxHeight	number	콤보리스트 최대 높이 설정						
CurrentColInfo	string	설정하고자 하는 컬럼 정보 문자열						
DataRowMerge	boolean	전체행의 가로머지 허용여부 (Default=0)						
DeferredHScroll	boolean	가로 스크롤에 대한 지연 처리 모드 사용 여부 (Default=0)						
DeferredVScroll	boolean	세로 스크롤에 대한 지연 처리 모드 사용 여부 (Default=0)						
DragCell	boolean	드래그시 셀 단위 드래그 여부 설정 (Default=0)						
DragMode	number	드래그 처리 방법 (Default=0)						
FilterOperator	string	필터링시 컬럼간 연산자 설정 (Default= "and")						
FitSizeColHeaderMode	boolean	컬럼에 대한 FitSize 처리시 컬럼간 셀 병합이 되어있는 헤더를 포함하여 처리할지 여부 설정 <table><tr><td>설정값</td><td>내용</td></tr><tr><td>0</td><td>포함하여 처리 (Default)</td></tr><tr><td>1</td><td>제외하고 처리</td></tr></table>	설정값	내용	0	포함하여 처리 (Default)	1	제외하고 처리
설정값	내용							
0	포함하여 처리 (Default)							
1	제외하고 처리							
FitSizeColMode	boolean	헤더의 컬럼 사이를 더블클릭시 처리 되는 FitSize 처리 모드 설정 (Default=0)						
FocusSumRow	boolean	합계행의 셀에 대한 포커스 허용 여부 설정 (Default=0)						
FrozenCol	number	좌측 고정컬럼의 수 (Default=0)						
FrozenColRight	number	우측 고정컬럼의 수 (Default=0)						
GroupSort	boolean	그룹핑시 기준컬럼에 대한 Sort 처리						



		여부 설정 (Default=1)
HeaderCheckMode	boolean	전체체크 선택시 필터링된 행만 체크한다. (Default=0)
HeaderMergeMode	boolean	헤더의 머지 방식 설정 (Default=0)
ImageStatus	boolean	상태값을 이미지로 표시할지 여부여부 (Default:0)
InvalidInputBehavior	number	편집 완료시점에 입력값이 invalid 인 경우에 대한 처리 방법 설정
InvalidInputCallBack	function	편집 완료시점에 입력값이 invalid 인 경우에 콜백설정
KeyFieldPosition	string	필수 입력 표시 위치 설정 (Default="Left")
MaxSort	number	헤더클릭을 이용한 연계 소팅 최대 개수를 설정 (Default=3)
MergeSheet	number	머지 종류 (Default=0)
MouseHoverMode	number	시트내의 행 또는 셀에 MouseOver에 대한 Hover 방식을 설정
NewRowDeleteMode	boolean	"입력" 상태인 행의 삭제 컬럼 체크시 confirm 메시지 사용 여부 설정 (Default=0)
NextPageCall	number	조회 모드가 서버페이징, allBack함수명 경우 다음 페이지를 호출하는 시점 설정 %에 해당하는 숫자값으로 60~90 사이만 설정 가능
NoFocusMode	boolean	포커스 행 표시 사용여부 설정 (Default = 0)
Page	number	한번에 표시할 행의 개수 (Default=20)
PopupCheckEditMode	boolean	컬럽팝업 메뉴를 통한 값 설정시 편집가능여부에 따른 처리 여부



		(Default=1)
PrevColumnMergeMode	number	LazyLoad 방식인 경우 앞컬럼 머지 방식 설정 (0:페이지 전체, 1:페이지내 (Default))
SelectionSummary	string	선택영역 요약 정보 표시 설정
SearchMode	number	조회 방식 설정 (Default:2)
SizeMode	number	사이즈 방식 설정 (Default=0)
SmartResize	boolean	OnResize를 OnSmartResize 동작처럼 변환여부
SortEventMode	boolean	OnSort 이벤트 발생시 정렬된 모든 컬럼의 정보를 반환 할지 여부. (Default:0)
SPage	number	서버 요청 페이지 단위 행의 개수
SumPosition	boolean	합계행 위치 (1: 하단 고정(Default), 0: 상단 고정)
TabStop	boolean	시트의 TabStop 허용 여부 Default = 1
ToolTip	boolean	셀의 풍선도움말을 표시한다. (Default=0)
TouchScrolling	number	터치스크롤 방식 설정 (0:사용안함, 1:일반사용(Default), 2:지연이동)
TreeDragIconMode	number	트리구조에서 드래그시 드래그 위치 아이콘 표시 모드 (Default=0)
TreeNodeIcon	boolean	트리 노드 아이콘 사용 여부
TreeNodeToggleMode	number	마우스 입력을 통한 Tree 노드 토글시 처리 방법 (Default = 0)
UseEditMask	boolean	편집시 사용자포맷, 날짜에 대한 IMaskEdit 사용 여부 (Default:0)
UseFindDialog	boolean	찾기 팝업 다이얼로그 사용 여부

		(Default:1)
UseGroupActionMenu	boolean	그룹행에서 컨텍스트 메뉴 사용 여부 설정 (GroupActionMenu 참조) 0:사용안함-Default, 1:사용
UseHeaderActionMenu	boolean	헤더 컨텍스트 메뉴 사용 여부 설정 헤더메뉴를 사용함으로 설정하고 헤더메뉴를 설정하지 않는경우 자동으로 기본 메뉴가 출력된다. (HeaderActionMenu 참조) 0:사용안함-Default, 1:사용
UseHeaderSortCancel	boolean	헤더 클릭을 통한 컬럼 Sort 처리시 취소 모드 사용 여부
UseNoDataRow	number	조회된 데이터가 없는 경우, '조회된 데이터가 없습니다' 문구 표시 여부 설정 (0:사용안함, 1:사용(Default))
UsePivotDialog	boolean	피벗 설정 팝업 다이얼로그 사용 여부 (Default:1)
VScrollMode	number	세로스크롤바 표시 설정 (0:Auto(Default), 1:고정)

AutoFitColWidth는 FitColWidth를 적용하고자 하는 시점을 설정하여 별도 코딩 없이 FitColWidth를 적용하고자 할 때 사용한다. 아래의 설정값을 구분자 "\""로 연결할 문자열 또는 배열집합으로 설정 한다. 설정하지 않은 경우 기본값은 "" 이다.

설정값	내용
search	조회 및 로드 시점
resize	시트 Resize 시점
init	초기화 및 RemoveAll 호출 시점
colhidden	컬럼 숨김/보임 시점
rowtransaction	로우 추가/삭제/숨김/보임 이후

AutoSumCalcMode는 AutoSum 타입의 컬럼에 셀 병합이 있는 경우 병합된 셀에 대한 합계 계산 방식을 설정 한다. 설정 값에 따라 아래와 같이 처리 된다.

설정값	내용
0	모든 셀에 대한 합계 계산 (Default)
1	병합된 셀의 부속셀은 제외 하고 계산

CachePageCount는 조회모드가 smLazyLoad 또는 smServerPaging 인 경우 렌더링 된 페이지의 최대 보관 개수를 설정 한다. 보관되는 페이지 개수가 많을 수록 브라우저의 성능 저하현상이 발생할 수 있으므로 주의 해서 사용해야 한다.

ChildPage는 트리구조에서 자식행에 대한 LazyLoad 설정으로 페이지징 개수 단위를 설정한다. 값을 설정한 경우에만 적용된다.

(이 속성을 사용하려면 SearchMode 를 smLazyLoad로 설정하여야 한다.)

ComboMaxHeight은 콤보타입의 콤보목록이 펼쳐질때의 최대 높이 값 설정으로 0으로 설정하는 경우는 브라우저 높이에 자동으로 설정된다. 단위는 픽셀 단위로 설정한다.

CurrentColInfo 속성은 이전 컬럼 들의 위치, 너비, 숨김 정보를 그대로 설정 하설정 할 때 사용한다. 설정 값은 반드시 이전 화면에서의 [GetCurrentColInfo](#) 메소드를 통한 반환값을 사용하여야 한다.

DataRowMerge는 전체행의 가로 머지 허용여부를 설정한다. 기본값은 0이다.

DeferredHScroll 속성은 가로 스크롤 이동에 대한 지연 처리 사용여부를 설정 한다. 사용으로 설정한 경우, DeferredScrollTime 전역속성으로 설정한 시간동안 스크롤 움직임이 없는 경우, 그때 스크롤 이동 처리를 수행 한다. 마우스 휠을 사용한 스크롤 이동이 해당 속성에 영향을 주지 않는다.

DeferredVScroll 속성은 세로 스크롤 이동에 대한 지연 처리 사용여부를 설정한다. 사용으로 설정한 경우, **DeferredScrollTime** 전역속성으로 설정한 시간동안 스크롤 움직임이 없는 경우, 그때 스크롤 이동 처리를 수행한다. 세로 스크롤의 경우에는 스크롤 지연 처리시간 동안 스크롤 이동에 대한 정보를 메시지 창을 통해 표시해 주며, [ScrollInfoFormat](#), [ScrollInfoPosition](#) 메소드를 통해 표시 위치와 표시 정보를 설정할 수 있다. 마우스 휠을 사용한 스크롤 이동이 해당 속성에 영향을 주지 않는다.

DragMode는 드래그 처리 방법 설정으로 기본값은 0이다.

설정값	내용
-1	- 드래그 사용 안함
0 (Default)	- 일반 : 셀 또는 행 범위 선택 - Ctrl 키 이용 : 행 드래깅
1	- 일반 : 행 드래깅 - Ctrl 키 이용 : 셀 또는 행 범위 선택

FilterOperator 속성은 필터링시 컬럼간이 처리 연산자를 설정한다.

설정가능한 값은 아래와 같으며, 설정하지 않은 경우 기본값은 "and" 이다.

설정값	설명
and	필터링시 컬럼간 AND(논리곱) 연산 처리
or	필터링시 컬럼간 OR(논리합) 연산 처리

FocusSumRow 속성은 합계행의 셀에 대한 포커스 허용 여부를 설정할 때 사용한다. 기본 값은 0으로 포커스를 허용 하지 않는다.

FrozenCol은 좌측 고정 컬럼의 개수를 설정한다. 기본값은 0이다.

FrozenColRight은 우측 고정 컬럼의 개수를 설정한다. 기본값은 0이다.

GroupSort는 그룹핑 시 기준컬럼의 데이터에 대한 Sort 처리 여부를 설정한다. 설정값이 0인 경우 현재 상태의 값을 기준으로 근접해 있는 동일한 값에 대해서 그룹핑 처리를 한다.

HeaderMergeMode는 헤더의 가로 머지 방식을 설정한다. 설정을 0으로 하는 경우 ColMerge 속성 설정값에 상관없이 머지 처리를 하며, 1로 하는 경우 ColMerge 속성에 따라 머지 처리한다.

ImageStatus는 상태 값을 이미지로 표현할지 여부를 설정 한다. 표현되는 이미지를 변경하고자 하는 경우, 테마의 ibsheet.css 파일의 (메인테마 기준) GMStatusR, GMStatusI, GMStatusU, GMStatusD 클래스의 이미지를 변경하면 된다.

InvalidInputBehavior 속성은 편집시 잘못된 값을 입력후 편집 종료시점에 대한 처리방법을 설정한다. "Date" 타입의 표맷에 맞지않는 날짜값 입력, "Int", "Float" 타입에서의 최대값, 최소값 범위를 넘는 값 입력, "EditLen" 속성을 설정시 최대 길이 초과 입력과 같은 경우가 대상 범위에 포함 된다.
설정 값에 따른 처리 방법은 아래와 같다.

설정값	내용
0	alert 메시지 후 해당 셀 편집상태 유지 (Default)
1	alert 메시지 후 해당 셀 편집종료(입력값 취소 처리)
2	alert 메시지 후 해당 셀 편집상태 유지(입력값 취소 처리)

InvalidInputCallBack 속성은 편집시 잘못된 값을 입력후 편집 종료하였을 때, 유효성 체크 이후 발생하는 함수를 설정한다. 해당 함수에는 오브젝트 형태의 리턴값을 가지게 되는데, 리턴값은 아래와 같다.

설정값	내용
Info.Raw	대상 행의 Index
Info.Col	대상 컬럼의 Index
Info.InputValue	셀 입력 값

KeyFieldPosition은 필수 입력 표시 위치 설정으로 기본값은 Left 이다.

설정값	설명
Left	헤더 텍스트의 왼쪽에 위치
Right	헤더 텍스트의 오른쪽에 위치

MergeSheet는 머지의 종류 설정으로 기본값은 msNone이다.

상수값	설정값	설명
0	msNone	머지 없음
1	msAll	전부 머지 가능
2	msPrevColumnMerge	앞 컬럼이 머지 된 경우 해당 행 안에서 머지 가능
3	msFixedMerge	단위데이터 행 구조에서 고정 셀 병합 기능
4	msBaseColumnMerge	기준컬럼 머지 영역 범위 내에서 머지 처리 기능 (좌측으로부터 첫번째 ColMerge 속성이 1인 컬럼이 기준컬럼 이 된다.)
5	msHeaderOnly	헤더만 머지
7	msPrevColumnMerge + msHeaderOnly	앞 컬럼이 머지 된 경우 해당 행 안에서 머지기능 + 헤더 머지
8	msFixedMerge + msHeaderOnly	단위데이터 행 구조에서 고정 셀 병합 기능 + 헤더 머지
9	msBaseColumnMerge + msHeaderOnly	기준컬럼 머지 영역 범위 내에서 머지 처리 기능 + 헤더 머지

MouseHoverMode는 시트내의 행 또는 셀에 마우스 커서가 위치했을 때 Hover 동작 모드를 설정한다. 설정하지 않은 경우의 기본값은 0이다.

설정값	설명
0	사용안함 (Default)
1	셀단위 Hover
2	행단위 Hover

Page는 페이지징 단위 값 설정으로 페이지징 모드, Lazy Load 모드 실시간 서버 처리 모드 설정시 페이지징 개수 단위를 설정한다. 기본값은 20이다.



PopupCheckEditMode는 컬럼의 팝업메뉴를 통한 값 설정시 편집가능 여부에 따른 처리 여부를 설정한다. 설정값이 1인 경우에는 편집가능한 셀에 대해서 값을 설정하고, 영역 선택에 의한 값 설정시 영역내에 숨김행은 제외 하고 처리한다. 설정값이 0인 경우에는 편집여부와 무관하게 값을 설정하고, 영역 선택에 의한 값 설정시 영역내에 숨김행이 있는 경우 해당 숨김행도 포함하여 처리 한다.

SelectionSummary는 선택영역 요약 정보 표시 설정에 대해 다음의 옵션을 구분자 "|" 로 연결하여 설정하며, 설정한 경우 건수 정보 표시위 반대 위치여 표시 된다.

설정값	설명
EmptyCell	빈값의 셀을 포함
NotEmptyCell	빈값의 셀을 포함하지 않음
DelRow	삭제상태의 행을 포함
NotDelRow	삭제상태의 행을 포함하지 않음

SearchMode는 조회 방식 모드 설정으로 일반 모드, 페이징 모드, Lazy Load 모드, 실시간 서버 처리 모드를 설정 할 수 있다. 기본값은 2 이다.

상수값	설정값	설명
0	smGeneral	일반 조회 모드 전체 데이터를 조회 하고 모든 데이터를 화면에 표시한다.
1	smClientPaging	페이징 모드 전체 데이터를 조회 하고 화면에서 Page 속성 설정값만큼 페이징 처리를 한다.
2	smLazyLoad	Lazy Load 모드 전체 데이터를 조회 하고 화면에서 Page 속성 설정값 단위로 스크롤 위치에 따라 데이터를 화면에 표시한다.
3	smServerPaging	실시간 서버 처리 모드 (스크롤링 방식) 스크롤 위치에 해당하는 데이터만 서버에서 결과를 받아 화면에 표시한다. 화면에 표시되

		지 않는 데이터는 소멸된다.
4	smServerPaging2	실시간 서버 처리 모드(페이지인덱스 방식) 페이지에 해당하는 데이터만 서버에서 결과를 받아 화면에 표시한다. 화면에 표시되지 않는 데이터는 소멸된다.

SizeMode는 사이즈 처리 방식 설정으로 IB Sheet 객체 생성에서 설정한 사이즈를 기준으로 응용 설정이 가능하다. 기본값은 sizeAuto 이다.

상수값	설정값	설명
0	sizeAuto	설정된 높이, 너비 그대로 사용 (Default)
1	sizeNoVScroll	높이를 스크롤 없이 자동 설정 생성시 설정한 높이값은 무시된다.
2	sizeNoHScroll	너비를 스크롤 없이 자동 설정 생성시 설정한 너비값은 무시된다.
3	sizeNoBothScroll	높이, 너비 모두 스크롤 없이 자동 설정 생성시 설정한 높이, 너비값은 무시된다.
4	sizeAdvancedAuto	설정된 높이 이전까지 스크롤 없이 높이 자동 설정

SmartResize는 Resize 처리에 대한 개선 방안이다.

설정을 1(true)로 하는 경우에는 **OnSmartResize**이벤트는 **발생하지 않고** OnResize이벤트가 OnSmartResize이벤트 처럼 0.3(300mm)초 단위로 동작한다.

SumPosition은 합계행의 위치 설정으로 기본값은 posBottom이다.

상수값	설정값	설명
1	posBottom	하단 고정행에 위치
0	posTop	상단 고정행에 위치

SPage 속성은 실시간 서버 처리 페이징 모드 에서 서버에 요청할 페이지 단위 행의 개수를 설정 하는 속성이다. 이 속성은 실시간 서버 처리 페이징 모드 에서 만 유효 하며, 설정하지 않는 경우 Page 속성 설정 값으로 자동으로 설정된다.



(*참고 : 실시간 서버 처리 페이징 모드는 SearchMode 속성의 설정을 4 또는/smServerPaging2로 설정 함을 의미 한다.)

TabStop은 페이지에서 Tab키 입력에 의한 Tab 이동시 시트를 포함할지 여부에 대한 설정으로 해당 속성을 0(false)로 설정하는 경우 시트를 제외한 input element에 대한 Tab 네비게이션을 구성 할 수 있다. 이 속성의 기본값은 1(true) 이다.

ToolTip은 셀의 풍선 도움말 설정으로 기본값은 0이다. 1로 설정하는 경우 마우스 오버시 셀의 내용이 풍선 도움말로 표시된다.

TreeDragIconMode는 트리구조에서 드래드 아이콘 표시 방법으로서 기본값은 0이다

설정값	내용
0	이동이 가능한 위치에 맞게 아이콘 표시 (Default)
1	이동 위치와 상관 없이 모든 아이콘을 표시

TreeNodeIcon 속성은 트리구조에서 트리 노드에 대한 아이콘 표시 여부를 설정 한다.

TreeNodeToggleMode 속성은 마우스 입력을 통한 Tree 노드 토글시 처리 방법을 설정 한다.

설정값	내용
0	토글후 해당 셀에 포커스 이동 처리 방법 (Default)
1	셀의 포커스 이동 없이 토글 처리 방법

UseEditMask 속성은 Text의 사용자 포맷 또는 날짜 포맷에서 IBMaskEdit을 연동하여 마스킹 기능 사용 여부를 설정 한다. IBMaskEdit 제품 버전 7.3.0.0 이상인 경우에만 사용이 가능 하며, IE8 이상의 브라우저 및 jQuery 1.7.0 이상에서만 사용 할 수 있다 (IBMaskEdit 제약 사항)

UseFindDialog 속성은 시트에서 찾기 팝업 다이얼로그 사용여부를 설정 한다.

해당 속성을 0으로 설정하면 단축키(Ctrl + Alt + F) 또는 ShowFindDialog 메소드 호출시 찾기 팝업 다이얼로그를 출력하지 않는다. 해당 속성의 기본값은 1이다.

UseHeaderActionMode는 시트의 헤더 영역에서 컨텍스트 메뉴 사용여부를 설정 한다. 이 속성을 사용으로 설정한 경우 기본적으로 아래 내용의 메뉴가 설정되며, 이를 변경하고자 하는 경우 HeaderActionMenu 메소드를 이용하여 설정 할 수 있다. 해당 메뉴의 라벨은 ibmsg 파일에서 변경이 가능 하다.

"컬럼 정보 저장/저장 취소" 는 브라우저의 localStorage를 이용하여 처리하기 때문에 저장시 동일한 브라우저에서만 저장정보를 사용할 수 있으며, **IE의 경우 9이상**에서만 사용이 가능하다. (일부 OS및 에 따라 IE8에서도 사용가능)

메뉴명	내용	라벨 정의(ibmsg)
오름차순 정렬	대상 컬럼 오름 차순 정렬	SYS_ColSortAsc
내림차순 정렬	대상 컬럼 내림 차순 정렬	SYS_ColSortDesc
컬럼 숨기기	대상 컬럼 숨기기	SYS_ColHidden
모든 컬럼 숨기기 취소	"컬럼 숨기기" 메뉴를 이용하여 숨긴 컬럼 모두 출력	SYS_CancelColHidden
컬럼 정보 저장	현재 상태의 모든 컬럼에 대한 위치, 숨김여부, 너비 정보 저장	SYS_SaveColInfo
컬럼 정보 저장 취소	저장된 컬럼 정보 초기화	SYS_ResetColInfo
필터행 사용	필터행 출력	SYS_ShowFilter
필터행 사용 안함	필터행 숨김	SYS_HideFilter

UseHeaderSortCancel 속성은 헤더의 컬럼 클릭시 컬럼에 대한 Sort 취소 기능 사용여부를 설정 한다. 해당 속성 설정에 따라 아래와 같이 동작 한다.

설정값	설명
0	헤더의 컬럼 클릭시 오름차순 정렬/내림차순 정렬 순으로 처리
1	헤더의 컬럼 클릭시 오름차순 정렬/내림차순 정렬/정렬 취소 순으로 처리

UsePivotDialog 속성은 시트에서 피벗 설정 팝업 다이얼로그 사용여부를 설정



한다. 해당 속성을 0으로 설정하면 단축키(Ctrl + Alt + T) 또는 ShowPivotDialog 메소드 호출시 피벗 설정 팝업 다이얼로그를 출력하지 않는다. 해당 속성의 기본값은 1이다.

➤ **Example**

```
//페이징 모드 조회 예제
cfg= {SearchMode:1, Page:20};
mySheet.SetConfig(cfg);
```

➤ **제공 버전**

7.0.0.0	
7.0.4.2	TabStop 속성 추가
7.0.8.0	AutoFitColWidth, MouseHoverMode 속성 추가
7.0.11.0	PopupCheckEditMode 속성 추가
7.0.12.2	Refresh 인자 추가
7.0.13.5	FitSizeColHeaderMode, FilterOperator 속성 추가
7.0.13.6	UseHeaderActionMode 의 오름차순 정렬, 내림차순 정렬 기본 메뉴 추가
7.0.13.10	InvalidInputBehavior 속성 추가
7.0.13.11	DeferredHScroll, DeferredVScroll 속성 추가
7.0.13.14	CachePageCount 속성 추가
7.0.13.17	ImageStatus 속성 추가
7.0.13.22	AutoSumCalcMode 속성 추가
7.0.13.26	GroupSort 속성 추가
7.0.13.30	CurrentColInfo 속성 추가
7.0.13.33	UseEditMask 속성 추가
7.0.13.34	UseFindDialog, UsePivotDialog 속성 추가
7.0.13.48	UseHeaderSortCancel 속성 추가
7.0.13.51	FocusSumRow 속성 추가
7.0.13.52	TreeNodeToggleMode 속성 축
7.0.13.54	InvalidInputCallBack 속성 추가
7.0.13.56	TreeNodeIcon 속성 추가



7.0.13.59

SPage 속성 추가



SetDown2ExcelConfig Method

➤ 기능

- Down2Excel 메소드의 기본 속성을 정의 한다. 이 메소드를 통해 정의한 속성은 Down2Excel 메소드의 기본값으로 처리가 된다.

➤ Syntax

Syntax	ObjId.SetDown2ExcelConfig([Opt])
--------	----------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Opt	Object	선택	기본으로 정의할 속성 객체 세부 사항은 Down2Excel 참고

➤ Example

```
// FileName, Merge 속성에 대한 기본값 정의
mySheet.SetDown2ExcelConfig({
  "FileName": "excel2",
  "Merge": 1
});
```

➤ 제공 버전

7.0.0.0	
---------	--



SetDown2TextConfig Method

➤ 기능

- Down2Text 메소드의 기본 속성을 정의 한다. 이 메소드를 통해 정의한 속성은 Down2Text 메소드의 기본값으로 처리가 된다.

➤ Syntax

Syntax	ObjId.SetDown2TextConfig([Opt])
--------	---------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Opt	Object	선택	기본으로 정의할 속성 객체 세부 사항은 Down2Text 참고

➤ Example

```
// DownHeader 속성에 대한 기본값 정의  
mySheet.SetDown2TextConfig({  
    "DownHeader": 1  
});
```

➤ 제공 버전

7.0.13.29	
-----------	--



SetEndEdit Method

➤ 기능

셀 편집을 종료 한다. 편집중인 내용을 저장하고 종료하는 경우에는 Save 인자를 1(true)로 설정하고 취소하고 종료하는 경우에는 Save 인자를 0(false)로 설정한다.

➤ Syntax

Syntax	ObjId. SetEndEdit (Save)
--------	---------------------------------

➤ Info

Return	Boolean, 종료 성공 여부		
Parameter	Type	필수여부	설 명
Save	Boolean	필수	편집중인 내용 저장 여부

➤ Example

```
// 저장 관련 로직 처리 함수
var result = mySheet.SetEndEdit(1);

if (!result) {
    return;
}

// 저장 처리 로직 진행
```

➤ 제공 버전

7.0.0.0

SetFilterOption Method

➤ 기능

필터행 사용시 컬럼의 필터 옵션값을 설정할 때 사용한다.

➤ Syntax

Syntax	ObjId. SetFilterOption (Col, Option, [Event])
--------	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Col	Number / String	필수	컬럼의 Index 또는 SaveName
Option	Number	필수	설정 할 Option 값
Event	Boolean	선택	이벤트 발생 여부 (Default : 1)

➤ Option Value

0	사용 안 함		
1	같음	2	같지 않음
3	작음	4	같거나 작음
5	큼	6	같거나 큼
7	단어로 시작함	8	단어로 시작하지 않음
9	단어로 끝남	10	단어로 끝나지 않음
11	포함함	12	포함하지 않음

➤ Example

```
// 필터행이 1행일 경우 2컬럼의 필터링 할 값 설정
mySheet.SetCellValue(1, 2, "포함");

// 필터행 2컬럼의 필터 설정 - 문자열 '포함' 과 같은 단어 필터링.
mySheet.SetFilterOption (2, 1);
```



➤ 제공 버전

7.0.0.0	
7.0.13.54	Event 인자 추가

SetFilterValue Method

➤ 기능

필터행 사용시 컬럼의 필터값을 설정할 때 사용한다.

➤ Syntax

Syntax	ObjId. SetFilterValue (Col, Value, Option, [Event])
--------	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Col	Number / String	필수	컬럼의 Index 또는 SaveName
Value	String	필수	필터에 설정할 값
Option	Number	선택	설정 할 Option 값
Event	Boolean	선택	이벤트 발생 여부 (Default : 1)

➤ Option Value

0	사용 안 함		
1	같음	2	같지 않음
3	작음	4	같거나 작음
5	큼	6	같거나 큼
7	단어로 시작함	8	단어로 시작하지 않음
9	단어로 끝남	10	단어로 끝나지 않음
11	포함함	12	포함하지 않음

➤ Example

```
// 5컬럼에 "서울"이 포함된 문자열 필터링하기
mySheet.SetFilterValue(5, "서울", 11);
```

➤ 제공 버전



7.0.0.0	
7.0.13.54	Event 인자 추가

SetFindDialog Method

➤ 기능

단축키 Ctrl+Shift+F 를 이용한 찾기 다이얼로그의 기본 정보를 설정 한다.

FullMatch 인자는 찾기 종류 항목을 설정 한다. FindText Method의 FullMatch 설정과 동일 하며 설정값은 아래와 같다.

FullMatch 값	기능
-1	SearchText와 전체가 같은 행을 찾는다.
0	SearchText와 앞 부분이 같은 행을 찾는다.
1	SearchText와 뒤 부분이 같은 행을 찾는다.
2	SearchText와 가운데 부분이 같은 행을 찾는다.

➤ Syntax

Syntax	ObjId. SetFindDialog (Show,[Col],[FullMatch],[FirstStart],[CaseSensitive],[KeepDialog])
--------	--

➤ Info

Return	None.		
Parameter	Type	필수여부	설 명
Show	Boolean	필수	찾기 다이얼로그 사용 여부
Col	Long/String	선택	찾을 컬럼의 Index 또는 SaveName 빈값인 경우 전체 컬럼, Default=""
FullMatch	Integer	선택	텍스트 일치 종류 설정, Default=-1
FirstStart	Boolean	선택	시작 위치 설정, Default=1
CaseSensitive	Boolean	선택	대소 구분 설정, Default=0
KeepDialog	Boolean	선택	찾기 후 창 닫기 여부 설정, Default=0

➤ Example



//찾기창 사용, 대상 컬럼:5, 시작위치:포커스행 다음, 일치종류:가운데일치
mySheet.SetFindDialog(1, 5, 2, 0);

- **제공 버전**
7.0.0.0



SetFocus Method

➤ 기능

시트에 포커스를 설정 한다.

➤ Syntax

Syntax	ObjId. SetFocus ()
--------	---------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
없음			

➤ Example

// 시트에 포커스를 설정한다. mySheet.SetFocus();
--

➤ 제공 버전

7.0.0.0



SetHeaderMode Method

➤ 기능

헤더의 모드를 설정한다.

InitHeaders 메소드에서 설정한 모드 설정값을 초기화 이후 변경하고자 할 때 사용한다.

➤ Syntax

Syntax	ObjId. SetHeaderMode (Mode)
--------	------------------------------------

➤ Info

Return	None			
Parameter	Type	필수여부	설 명	
Mode.ColMove	Boolean	선택	헤더 컬럼 이동 가능 여부	
Mode.ColResize	Boolean	선택	컬럼 너비 ReSize 여부	
Mode.HeaderCheck	Boolean	선택	헤더에 전체 체크 표시 여부	
Mode.Sort	Number	선택	헤더 클릭을 통한 컬럼 Sort 허용 여부	
			설정값	설명
			0	사용 안함
			1	Sort 기능 사용 (Default)
			2	Sort 아이콘만 표시
3	colSpan 설정이 아닌 헤더 셀만 Sort 기능 사용			

➤ Example

```
// 헤더의 마우스 드래그를 이용한 컬럼 이동 및 리사이즈 기능을 제한한다.
mySheet.SetHeaderMode({
    ColResize:0,
```



ColMode:0

});

➤ 제공 버전

7.0.2.0	
7.0.4.6	Sort 속성 개선 (아이콘만 표시하는 기능 추가)
7.0.13.24	Sort 속성 개선 (colSpan 에 대한 예외 기능 추가)



SetLoadExcelConfig Method

➤ 기능

LoadExcel 메소드의 기본 속성을 정의 한다. 이 메소드를 통해 정의한 속성은 LoadExcel 메소드의 기본값으로 처리가 된다.

➤ Syntax

Syntax	ObjId.SetLoadExcelConfig([Opt])
--------	---------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Opt	Object	선택	기본으로 정의할 속성 객체 세부 사항은 LoadExcel 참고

➤ Example

```
// MaxFileSize, FileExt 속성에 대한 기본값 정의  
mySheet.SetLoadExcelConfig({  
    "FileExt": "xlsx",  
    "MaxFileSize": 10  
});
```

➤ 제공 버전

7.0.13.27	
-----------	--



SetLoadTextConfig Method

➤ 기능

LoadText 메소드의 기본 속성을 정의 한다. 이 메소드를 통해 정의한 속성은 LoadText 메소드의 기본값으로 처리가 된다.

➤ Syntax

Syntax	ObjId.SetLoadTextConfig([Opt])
--------	--------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Opt	Object	선택	기본으로 정의할 속성 객체 세부 사항은 LoadText 참고

➤ Example

```
// Mode 속성에 대한 기본값 정의
mySheet.SetLoadTextConfig({
    "Mode": "HeaderMatch"
});
```

➤ 제공 버전

7.0.13.29	
-----------	--



SetMergeCell Method

➤ 기능

특정 셀을 기준으로 인접한 셀과 강제로 머지한다.

반드시 조회용 화면에서만 사용해야 한다.

일반적인 경우 인접한 셀과 값이 동일한 경우에만 자동으로 머지되지만, 이 메소드를 호출하면 지정된 셀로부터 특정한 크기로 무조건적(인접한 셀들과 값이 동일하지 않더라도)으로 강제 머지되어 표시된다.

해당 메소드 사용시 대상 행들에 대한 갱신 처리가 발생한다. 따라서 동일한 행의 여러 컬럼에 대한 처리시 성능저하가 발생하게 되는데, 이런 경우 배열 형식의 인자로 다중 처리하는 방식을 사용하면 성능 저하 문제를 해결 할 수 있다. 또한 중복되지 않은 대상 행이 많은 경우에는 Refresh 인자를 0(false)로 설정 후 해당 메소드를 수행하고, 이후 RenderSheet 메소드를 호출하여 성능 저하 문제를 해결 할 수 있다.

➤ Syntax

Syntax	ObjId. SetMergeCell (Row, Col, Rows, Cols, Refresh)
	ObjId. SetMergeCell ([[Row, Col, Rows, Cols], ...], Refresh);

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Number	필수	강제머지할 셀의 Row Index
Col	Number	필수	강제머지할 셀의 Column Index
Rows	Number	필수	강제머지할 셀의 Row 개수
Cols	Number	필수	강제머지할 셀의 Col 개수
Refresh	Number	선택	대상영역에 대한 갱신 여부, Default=1



➤ **Example**

```
// (1,10)부터(2,11)까지 2x2 창문형태의 셀을 강제로 머지하여 표시한다.  
mySheet.SetMergeCell(1, 10, 2, 2);  
  
// 배열 형식의 인자를 이용한 다중 처리  
// Index가 3, 4, 5인 컬럼에 대하여 5행과 6행에 대한 셀 병합을 한다.  
mySheet.SetMergeCell([  
    [5, 3, 2, 1], [5, 4, 2, 1], [5, 5, 2, 1]  
]);  
  
// RenderSheet 메소드를 이용한 성능 개선  
mySheet.SetMergeCell(1, 1, 50, 3, 0); // 대상 행이 50개. Refresh인자 0 설정  
mySheet.RenderSheet(2); // 시트 강제 랜더링 처리
```

➤ **제공 버전**

7.0.0.0	
7.0.13.2	멀티 처리를 위한 배열 형식의 인자 지원
7.0.13.5	Refresh 인자 추가



SetRowHaveChildValue Method

➤ 기능

트리구조에서 조회시 HaveChild 속성 설정값을 변경한다.

HaveChild 속성을 설정하는 경우 설정값에 따라서 하위노드 확장 아이콘이 설정이 되는데 이 메소드를 이용하여 설정되어 있는 HaveChild 속성값을 변경할 수 있다.

➤ Syntax

Syntax	ObjId. SetRowHaveChildValue (Row, HaveChild)
--------	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	대상 행의 Index
HaveChild	Boolean	필수	속성 설정값

➤ Example

```
// 3행의 HaveChild 속성을 0으로 변경한다.  
mySheet.SetRowHaveChildValue(3, 0);
```

➤ 제공 버전

7.0.0.0

SetSelectRange Method

➤ 기능

특정 셀의 범위 영역에 대해서 셀렉션 설정을 한다. 시트에서 마우스 드래그를 이용한 셀렉션 처리를 이 메소드를 통해서 동일하게 할 수 있다.

인자를 설정하지 않거나 잘못 설정한 경우 이전 시트의 셀렉션 설정은 클리어 된다. 범위 종료셀의 행 또는 컬럼 정보를 설정하지 않은 경우 범위 시작 셀의 정보로 자동 설정된다.

➤ Syntax

Syntax	ObjId.SetSelectRange(Row1, Col1, Row2, Col2)
--------	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row1	Number	필수	범위 시작 셀의 행 Index
Col1	Number/ String	필수	범위 시작 셀의 컬럼 Index 또는 SaveName
Row2	Number	선택	범위 종료 셀의 행 Index
Col2	Number/ String	선택	범위 종료 셀의 컬럼 Index 또는 SaveName

➤ Example

```
// (2,2)부터(2,10)까지 셀렉션 처리를 한다..
mySheet.SetSelectRange(2, 2, 2, 10);

// 이전 셀렉션 처리를 클리어 한다.
mySheet.SetSelectRange();
```

➤ 제공 버전



7.0.7.0	
---------	--

SetSplitMergeCell Method

➤ 기능

셀 병합(머지) 되어 있는 셀 영역을 취소(분할) 처리 한다.

동일한 값에 의한 자동 머지 또는 SetMergeCell 메소드를 이용한 머지에 대해서 취소 할 수 있다.

최소 한 경우 각 셀의 값은 셀 병합(머지) 이전의 값을 유지하게 된다.

➤ Syntax

Syntax	ObjId.SetSplitMergeCell(Row, Col)
--------	-----------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Number	필수	대상 셀의 행 Index
Col	Number/ String	필수	대상 셀의 컬럼 Index 또는 SaveName

➤ Example

// (1,1)부터(2,2)까지 2x2 셀 병합 영역을 분할 한다. mySheet.SetSplitMergeCell(1, 1);

➤ 제공 버전

7.0.0.0



SheetWidth Method

➤ 기능

전체 너비를 설정하거나 확인한다. 값은 픽셀 단위로 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetSheetWidth()
--------	-----	-------------------------------

➤ Info

Return	Integer, 설정되어 있는 너비 픽셀값(Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// 현재 설정되어있는 전체 너비를 확인한다.
Alert(mySheet.GetSheetWidth());
```

➤ Syntax

Syntax	Set	ObjId. SetSheetWidth (Width)
--------	-----	-------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Width	Integer	필수	설정하고자 하는 시트 너비 픽셀값

➤ Example

```
//전체 너비를 800px로 설정한다.
mySheet.SetSheetWidth(800);
```

➤ 제공 버전

7.0.0.0



SheetHeight Method

➤ 기능

전체 높이를 설정하거나 확인한다. 값은 픽셀 단위로 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetSheetHeight ()
--------	-----	---------------------------------

➤ Info

Return	Integer, 설정되어 있는 높이 픽셀값(Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// 현재 설정되어있는 전체 높이를 확인한다.
Alert(mySheet.GetSheetHeight());
```

➤ Syntax

Syntax	Set	ObjId. SetSheetHeight (Height)
--------	-----	---------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Height	Integer	필수	설정하고자 하는 시트 높이 픽셀값

➤ Example

```
//전체 높이를 800px로 설정한다.
mySheet.SetSheetHeight(800);
```

➤ 제공 버전

7.0.0.0



ShowButtonImage Method

➤ 기능

팝업과 콤보의 버튼 이미지를 표시하는 종류를 확인하거나 설정한다
초기 로드 시 버튼 이미지 표시 종류를 설정한다
설정할 표시 종류는 다음과 같다.

종류	설명
0	Focus가 있을 때만 콤보, 달력, 팝업 이미지 표시
1	Edit 가능할때 달력, 팝업 이미지 표시
2	항상 달력, 팝업 이미지 표시
3	Edit 가능할때 콤보, 달력, 팝업 이미지 표시 (Default)
4	항상 콤보, 달력, 팝업 이미지 표시

➤ Syntax

Syntax	Get	ObjId. GetShowButtonImage()
--------	-----	------------------------------------

➤ Info

Return	Integer, 현재 설정 값(Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// 팝업과 콤보의 버튼 이미지를 표시하는 종류를 확인한다.
mySheet.GetShowButtonImage();
```

➤ Syntax

Syntax	Set	ObjId. SetShowButtonImage(type)
--------	-----	--

➤ Info

Return	None
--------	------



Parameter	Type	필수여부	설 명
type	Integer	필수	설정 값 (Default =3)

➤ **Example**

// Focus가 있을경우만 콤보와, 팝업이미지 표시된다.

mySheet.SetShowButtonImage(0);

NO	상태	<input type="checkbox"/> 삭제	문자열	콤보	콤보에디트	팝업	팝업에디트
1		<input type="checkbox"/>	장순연	대기	서울대학교	대한민국	SK텔레콤
2		<input type="checkbox"/>	김정호	진행중	고려대학교	일본	삼성전자
3		<input type="checkbox"/>	정상호	대기	울산대학교	대한민국	아시아나항공

// Edit가 가능할 때 팝업이미지가 표시된다.

mySheet.SetShowButtonImage(1);

NO	상태	<input type="checkbox"/> 삭제	문자열	콤보	콤보에디트	팝업	팝업에디트
1	삭제	<input checked="" type="checkbox"/>	장순연	대기	서울대학교	대한민국	SK텔레콤
2	삭제	<input checked="" type="checkbox"/>	김정호	진행중	고려대학교	일본	삼성전자
3		<input type="checkbox"/>	정상호	대기	울산대학교	대한민국	아시아나항공

// 항상 팝업이미지가 표시 된다.

mySheet.SetShowButtonImage(2);

NO	상태	<input type="checkbox"/> 삭제	문자열	콤보	콤보에디트	팝업	팝업에디트
1		<input type="checkbox"/>	장순연	대기	서울대학교	대한민국	SK텔레콤
2		<input type="checkbox"/>	김정호	진행중	고려대학교	일본	삼성전자
3		<input type="checkbox"/>	정상호	대기	울산대학교	대한민국	아시아나항공

// Edit 가능할때 콤보와 팝업이미지가 표시된다.

mySheet.SetShowButtonImage(3);

NO	상태	<input type="checkbox"/> 삭제	문자열	콤보	콤보에디트	팝업	팝업에디트
1	삭제	<input checked="" type="checkbox"/>	장순연	대기	서울대학교	대한민국	SK텔레콤
2		<input type="checkbox"/>	김정호	진행중	고려대학교	일본	삼성전자
3		<input type="checkbox"/>	정상호	대기	울산대학교	대한민국	아시아나항공



// 항상 콤보와 팝업 이미지가 표시된다.

mySheet.SetShowButtonImage(4);

NO	상태	<input type="checkbox"/> 삭제	문자열	콤보	콤보에디트	팝업	팝업에디트
1	삭제	<input checked="" type="checkbox"/>	장순연	대기	서울대학교	대한민국	SK텔레콤
2	삭제	<input checked="" type="checkbox"/>	김정호	진행중	고려대학교	일본	삼성전자
3		<input type="checkbox"/>	정상호	대기	울산대학교	대한민국	아시아나항공

➤ 제공 버전

7.0.0.0



ShowCalendar Method

➤ 기능

Text 타입의 컬럼에 포맷으로 날짜 포맷을 설정한 경우 OnClick 이벤트에서 ShowCalendar()를 실행하면 해당 컬럼 클릭 시 달력팝업을 출력해준다.

날짜 데이터형에 '0000-00-00', '2013-00-00' 형태의 데이터를 입력하고 싶은 경우 Text, Popup 컬럼에 CustomDate 속성을 1로 설정한 경우에도 OnClick, OnPopupClick 에서 사용 가능하다.

CustomDate를 사용하는 경우 사용자 포맷을 지정해주어야 하며 년월일(####-##-##, ####/##/## 등), 년월(####-##, ####/## 등), 월일(##-##, ##/## 등) 타입이 가능하다.

Format으로 지정한 #의 갯수로 년월일(8개), 년월(6개), 월일(4개) 포맷이 결정된다.

년 포맷은 4자리 표기만 지원하고, 입출력 순서는 년월일 순서만 가능하다.

날짜 데이터값이 있는 경우 달력출력 시 해당날짜를 선택상태로 표시하며, 데이터가 없거나 날짜값이 아닌 경우 오늘 날짜를 선택상태로 출력해준다.

➤ Syntax

Syntax	ObjId. ShowCalendar()
--------	------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
없음			

➤ Example

<pre>// 텍스트 타입 컬럼 클릭시 달력 팝업 출력하기 function mySheet_OnClick(Row, Col) { mySheet.ShowCalendar(); }</pre>



```
}  
  
// Popup컬럼의 OnPopup 버튼 클릭시 달력 팝업 출력하기  
function mySheet_OnPopupClick(Row,Col) {  
    mySheet.ShowCalendar();  
}
```

➤ **제공 버전**

7.0.0.0



ShowDebugMsg Method

➤ 기능

디버깅용 메시지의 표시여부를 확인하거나 설정한다.

디버깅 메시지를 표시하도록 설정하면 시스템 팝업 메시지로 디버깅을 위한 메시지를 표시하고, 표시 불가로 설정하면 OnDebugMsg 이벤트가 발생하여 이벤트의 인자를 통해 메시지를 확인할 수 있다.

설정값은 다음과 같다.

종류	설명
-1	시스템 팝업 디버그 시작
0	모든 디버그 종료

➤ Syntax

Syntax	ObjId. ShowDebugMsg (Msg)
--------	----------------------------------

➤ Info

Return	Boolean, 디버깅용 메시지 표시 여부		
Parameter	Type	필수여부	설 명
Msg	Integer	필수	디버깅용 메시지 표시여부 설정 (Default=0)

➤ Example

<pre>//저장 처리 과정을 디버깅 메시지를 팝업으로 표시 mySheet.ShowDebugMsg(-1); //조회 mySheet.DoSearch("list.xml");</pre>

➤ 제공 버전

7.0.0.0

ShowColumnPopup Method

➤ 기능

InitColumns 함수를 이용하여 컬럼 팝업 기능이 설정된 컬럼에서 함수를 이용하여 강제로 팝업을 열어주는 기능이다.

팝업이 열리는 위치는 MousePos를 1로 설정하는 경우 마지막으로 마우스를 클릭한 (X,Y) 좌표에 팝업을 표시하고, MousePos를 0로 설정한 경우는 (Row,Col) 셀의 위치의 아래쪽에 팝업을 표시한다.

2개행 이상의 단위데이터행 구조에서 Col 인자에 Index를 설정할 경우 모든행에 대해서 처리를 하고 SaveName으로 설정할 경우 해당 SaveName이 포함된 행에 대해서만 처리 한다.

➤ Syntax

Syntax	ObjId.ShowColumnPopup(Row, Col, [MousePos])
--------	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Number	필수	선택할 셀의 Row Index
Col	Number / String	필수	선택할 셀의 Column Index 또는 SaveName
MousePos	Boolean	선택	마지막 마우스 위치 사용 여부, Default=1(마지막 위치 사용)

➤ Example

<pre>//현재 셀 위치에 팝업 열기 mySheet.ShowColumnPopup(1, 4, 0) //마우스를 클릭한 마지막 위치에서 팝업 열기 mySheet.ShowColumnPopup(1, 4,1)</pre>



- 제공 버전
7.0.0.0



ShowFilterRow Method

➤ 기능

IBSheet 상단 고정행으로 필터행을 추가한다.

이 함수를 사용하면 컬럼별 데이터에 대한 필터링 기능을 사용할 수 있다.

함수의 호출 시점은 초기화 이후 조회 처리전에 호출 하여야 한다.

➤ Syntax

Syntax	ObjId. ShowFilterRow()
--------	-------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
없음			

➤ Example

//필터행 추가하기 mySheet.ShowFilterRow();
--

➤ 제공 버전

7.0.0.0



ShowFindDialog Method

➤ 기능

사용자가 시트에서 ctrl+shift+F 입력시 표시되는 찾기 창을 표시한다.

➤ Syntax

Syntax	ObjId. ShowFindDialog()
--------	--------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
//찾기창을 표시한다.  
mySheet.ShowFindDialog();
```

➤ 제공 버전

7.0.0.0

ShowFooterRow Method

➤ 기능

그리드 하단에 고정되는 사용자 정의 행을 설정한다.

`data` 인자는 JSON 형식의 조회 데이터 구조를 사용한다.

형식은 Key:Value 형태이며, Key는 컬럼 초기화 구성시 설정한 SaveName 이며 Value는 해당 셀의 값이 된다.

속성은 Key 값을 초기화시 설정된 SaveName과 속성명을 '#' 으로 구분하여 구성해서 사용한다. (예, SaveName#Type)

`Type` 은 `Text`, `Int`, `Float` 의 데이터타입만 허용되고, 기타 속성에 대한 타입 및 설정은 InitColumns Method 의 속성 설명을 참고 한다.

➤ Syntax

Syntax	ObjId. ShowFooterRow (Data)
--------	------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Data	Object/ Array	필수	Footer 설정 객체

셀별 속성 설정은 해당 키값 과 속성명을 "#"으로 구분하여 사용하며 사용 가능한 속성은 다음과 같다 (예, SaveName#BackColor)

속성명	설 명
Type	데이터 타입 (Text, Int, Float)
Align	좌우 정렬 값
RowSpan	RowSpan 범위 값
ColSpan	ColSpan 범위 값
FontColor	폰트색상
FontBold	폰트볼드



BackColor	셀 배경색
Format	포맷설정
PointCount	실수 형태의 데이터 타입에서 소수점 자리수

➤ **Example**

// 하단 고정행을 2개로 만들며, TEXT_DATA 컬럼의 경우 2행, 2열의 머지를 한 형태로 행을 생성함.

```
mySheet.ShowFooterRow([  
    {"SEQ":"순번", "SEQ#Type":"Int", "TEXT_DATA":"홍길동",  
    "TEXT_DATA#BackColor":"blue", "TEXT_DATA#Align":"Center",  
    "TEXT_DATA#RowSpan":2, "TEXT_DATA#ColSpan":2, "IMAGE_DATA":"이미지",  
    "PASS_DATA":"비밀번호", "RADIO_DATA":"라디오"},  
    {"SEQ":"순번2", "COMBO_DATA":"콤보2","COMBOEDIT_DATA":"콤보에디트2",  
    "DATE_DATA":"날짜2","CHECKBOX_DATA":"체크박스2","RADIO_DATA":"라디오  
2"}  
]);
```

➤ **제공 버전**

7.0.13.60

ShowGroupRow Method

➤ 기능

시트의 헤더 상단에 그룹행을 추가 또는 보이도록 한다.

이 메소드를 사용하면 헤더 상단에 컬럼을 드래그 앤 드롭 할수 있는 그룹행이 생성되며 그룹행으로 그룹핑 하고자 하는 헤더의 컬럼을 마우스 드래그로 이동하면 해당 컬럼을 기준으로 데이터를 그룹핑 할 수 있다.

그룹핑의 해당 컬럼을 기준으로 트리 구조 형식으로 표현되며 그룹행의 우측 모두 펼치기/접기를 통해 자식행에 대한 모두 접기/펼치기를 할 수 있다.

그룹 기준 컬럼의 셀값은 기본적으로 해당 셀값으로 표현되고, Format 인자를 통해 셀값({%s})과 자식노드 개수({%c})에 대한 포맷을 지정하여 표현 할 수 있다. Format 인자는 생성하는 시점에서만 설정이 가능하고 숨김상태인 경우에는 재설정이 불가능 하다.

그룹행이 생성된 상태에서 이 메소드를 호출시 그룹 기준 컬럼 및 Format 을 변경 할 수 있다.

생성된 그룹행은 HideGroupRow 메소드를 통하여 삭제 또는 숨김 처리를 할 수 있다.

➤ Syntax

Syntax	ObjId. ShowGroupRow ([Cols], [Format])
--------	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Cols	Number/ String	선택	그룹핑 하고자 하는 컬럼의 Index 또는 SaveName을 구분자 " "로 연결한 문자열



Format	String	선택	아래의 키를 이용한 그룹기준 컬럼의 데이터 포맷 (Default="") {%s} : 해당 셀의 값 {%c} : 자식노드 개수
--------	--------	----	---

➤ **Example**

```
//그룹행 추가하기
mySheet.ShowGroupRow();

// 인덱스가 1, 3 인 컬럼 기준 그룹행 추가하기
mySheet.ShowGroupRow("1|3");

// SaveName이 sName, sDept 인 컬럼 기준 그룹행 추가하기
mySheet.ShowGroupRow("sName|sDept");

// Format 설정 (예) sName (5건)
mySheet.ShowGroupRow("sName", "{%s} ({%c}건)");

// Format 설정2(스타일 적용) 자식노드 개수에 color 적용
mySheet.ShowGroupRow("sName", "{%s} <font color='gray'>({%c}건)</font>");
```

➤ **제공 버전**

7.0.0.0	
7.0.10.0	Format 인자 추가
7.0.12.1	그룹기준 컬럼 및 Format 에 대한 동적 변경 기능 추가

ShowMsgMode Method

➤ 기능

Sheet에서 발생하는 각종 메시지를 시스템 팝업 형태로 사용할것인지 이벤트 방식을 사용할것인지 설정한다.

이 속성이 1이면 모든 메시지는 시스템 팝업으로 발생하고, 이 속성이 0이면 OnMessage Event가 발생한다. 메시지 표시창에 디자인을 적용해야 한다면, 버튼의 개수를 변경해야 할때 이 속성을 0으로 설정하고, OnMessage Event, ConfirmOK Method를 함께 사용한다.

➤ Syntax

Syntax	Get	ObjId. GetShowMsgMode()
--------	-----	--------------------------------

➤ Info

Return	Boolean, 현재 설정 값(Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//메시지 모드를 확인한다.
mySheet.GetShowMsgMode();
```

➤ Syntax

Syntax	Set	ObjId. SetShowMsgMode(Mode)
--------	-----	------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Mode	Boolean	필수	메시지 표시 여부



➤ **Example**

```
//메시지 모드를 설정한다.  
mySheet.SetShowMsgMode(0);  
  
//OnMessage 이벤트를 처리한다.  
Function mySheet_OnMessage(Msg, Level, IsConfirm) {  
    //메시지 표시하기  
    var win_result = window.showModalDialog(  
        "sheet_message.jsp?Msg=" + Msg + "&IsConfirm=" + IsConfirm,  
        'modalResult',  
        'dialogWidth:200px;dialogHeight:200px;center:yes;help:no;status:no;');  
    //Sheet로 메시지 결과를 반환한다.  
    If(IsConfirm) mySheet.ConfirmOK( win_result);  
}
```

➤ **제공 버전**

7.0.0.0

ShowPivotDialog Method

➤ 기능

사용자가 시트에서 ctrl+alt+T 입력시 표시되는 피벗 테이블 설정 팝업을 호출한다.

피벗 설정 다이얼로그를 통해 생성된 피벗 테이블 시트는 원본시트명 + "_Pivot"으로 시트명이 설정된다.

➤ Syntax

Syntax	ObjId. ShowPivotDialog()
--------	---------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
			없음

➤ Example

```
//피벗 테이블 설정 팝업을 호출한다.
mySheet.ShowPivotDialog();
```

➤ 제공 버전

7.0.0.0	
7.0.13.0	합계 비율 선택 기능 추가



ShowPivotSumRatio Method

➤ 기능

피벗 테이블에 대한 합계 비율을 적용 한다.

이 기능은 피벗 테이블 결과가 적용된 시트에서만 사용이 가능 하다.

➤ Syntax

Syntax	ObjId. ShowPivotSumRatio (Mode)
--------	--

➤ Info

Return	None			
Parameter	Type	필수여부	설 명	
Mode	Number	필수	합계 비율 모드 (Default : 0)	
			설정 값	설명
			0	기본 합계
			1	총 합계 비율
			2	열 합계 비율
			3	행 합계 비율

➤ Example

<pre>// 피벗 테이블 구성 완료시점에 총합계 비율을 설정 한다. Function mySheet_Pivot_OnSearchEnd() { mySheet_Pivot.ShowPivotSumRatio(1); }</pre>

➤ 제공 버전

7.0.13.0	
----------	--

ShowPivotTable Method

➤ 기능

대상 시트의 데이터를 기준으로 피벗 테이블을 동적으로 생성하여 결과를 표현 한다.

CreatePivotTable Method 와 달리 현재의 시트의 데이터를 사용하고 피벗 테이블 결과는 동적으로 시트를 생성하여 표현하는 점이 다르다. 또한, 이 메소드를 이용하면 피벗 설정 다이얼로그와 연계되어 사용할 수 있다.

동적으로 생성되는 피벗테이블 시트 객체는 현재 시트의 id + "_Pivot" 으로 생성 된다.

➤ Syntax

Syntax	ObjId.ShowPivotTable(opt)
--------	---------------------------

➤ Info

Return	None								
Parameter	Type	필수여부	설 명						
opt.Cols	string	필수	열 레이블 필드에 설정할 컬럼의 Index 또는 SaveName을 구분자 ' '로 연결한 문자열						
opt.Rows	string	필수	행 레이블 필드에 설정할 컬럼의 Index 또는 SaveName을 구분자 ' '로 연결한 문자열						
opt.Value	string	필수	계산 대상 컬럼의 Index 또는 SaveName을 구분자 ' '로 연결한 문자열						
opt.CalsSumRatio	Number	선택	합계 비율 모드 (Default:0)						
			<table><tr><th>설정 값</th><th>설명</th></tr><tr><td>0</td><td>기본 합계</td></tr><tr><td>1</td><td>총 합계 비율</td></tr></table>	설정 값	설명	0	기본 합계	1	총 합계 비율
			설정 값	설명					
			0	기본 합계					
1	총 합계 비율								



			2	열 합계 비율
			3	행 합계 비율

➤ **Example**

```
// Index가 5,6,7 인 컬럼을 행 레이블 필드로 설정하고,  
// Index가 8,9,10 인 컬럼을 열 레이블의 필드로 설정하여  
// Index가 13 인 컬럼에 대한 합계를 표시하는 피벗 테이블 구성  
mySheet.ShowPivotTable({  
    "Rows": "5|6|7",  
    "Cols": "8|9|10",  
    "Value": "13"  
});
```

➤ **제공 버전**

7.0.13.59	
-----------	--

ShowProcessDlg Method

➤ 기능

시트 중앙에 대기중 이미지를 표시 한다.

인자에 따라서 조회, 저장, 다운로드, 업로드의 대기 이미지를 표시 할 수 있으며 각각의 대기 이미지는 메소드에서 설정한 이미지를 사용한다.

표시된 이미지를 닫을 경우에는 HideProcessDlg Method 를 이용한다.

설정 인자에 따른 이미지 종류는 다음과 같다

Type	이미지 종류	Method (설정/확인)
Search	조회 대기 이미지	SearchingImage
Save	저장 대기 이미지	SavingImage
Download	다운로드 대기 이미지	DownloadingImage
Upload	업로드 대기 이미지	UploadingImage

➤ Syntax

Syntax	ObjId. ShowProcessDlg (Type)
--------	-------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Type	String	선택	대기 이미지 종류 (Default:"Search")

➤ Example

// 저장중 대기이미지를 표시 한다.. mySheet.ShowProcessDlg("Search");
--

➤ 제공 버전

7.0.0.0



ShowSortArrow Method

➤ 기능

헤더를 눌러 소트를 할 때 소트되는 방향을 화살표 이미지로 표시할지 여부를 설정하거나 확인한다.

➤ Syntax

Syntax	Get	ObjId. GetShowSortArrow()
--------	-----	----------------------------------

➤ Info

Return	Boolean, 화살표 이미지 표시여부(Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// 설정되어 있는 소트 방향 표시여부를 확인한다.  
Alert(mySheet.GetShowSortArrow());
```

➤ Syntax

Syntax	Set	ObjId. SetShowSortArrow (Arrow)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Arrow	Boolean	필수	화살표 표시 여부

➤ Example

```
// 소트 방향을 표시한다.  
mySheet.SetShowSortArrow(1);
```

➤ 제공 버전



7.0.0.0

ShowSubSum Method

➤ 기능

특정 컬럼의 데이터를 기준으로 소계와 누계를 계산하여 표시한다.

이 함수는 데이터를 조회하는 메소드를 호출하기 이전에 호출하여야 하며, 설정을 변경한 경우 다시 데이터를 조회하여야 적용된다.

SumCols 속성은 소계와 누계가 계산 되어야 할 컬럼의 Index를 "|"로 연결한 문자열로 해당 컬럼에 소계와 누계가 계산되어 표시된다.

ShowCumulate 속성을 1 설정하면 소계를 표시하고 그 아랫줄에 해당하는 소계의 누계가 계산되어 표시되고, 0으로 설정하면 소계만 계산하여 표시된다.

누계 기능은 다음과 같이 계산된다. 첫번째 누계는 첫번째 소계와 동일하며, 두번째 누계는 첫번째 소계와 두번째 소계를 합한 것이고, 마지막 누계는 이전 소계들을 모두 합한것이다. 누계 기능은 소계가 계산될때만 표시 되는 기능으로 처리된다.

AvgCols 속성은 평균으로 계산 되어야 할 컬럼의 Index를 "|"로 연결한 문자열로 해당하는 컬럼에 평균이 계산되어 표시된다.

CaptionText 속성은 소계를 대표글자의 포맷을 변경하여 설정할 때 사용한다. 설정하지 않은 경우는 "소계 : " + 기준값로 설정된다.

소계(누계) 표현시 "%s" 를 사용하며 기준 값은 "%col"을 사용하여 포맷을 설정 할 수있다.

예를 들어 "%s = %col"으로 설정한 경우 소계행은 소계 = 기준값, 누계행은 누계 = 기준값으로 표시된다.

소계를 표시하는 행은 첫번째 컬럼에 "소계 : " + 기준값으로 표시되고, 누계를 표시하는 행은 "누계 : " + 기준값으로 표시된다.

Position 속성은 소계/누계행의 생성위치를 설정한다. 설정 값에 따라 생성되는 위치는 아래와 같다.

설정값	생성 위치
bottom	소계 그룹의 하단 (Default)
top	소계 그룹의 상단
bottomAll	최하단
topAll	최상단

인자값은 JSON 형태의 포맷으로 설정 한다.

예) var info = {StdCol:2, SumCols:"2|3|4", ShowCumulate:1, AvgCols:"5|6"};

OtherColText 속성은 CaptionCol 이외의 컬럼에 문자열을 설정할 때 사용한다.

Mode 속성은 소계행 표시 방법을 설정 한다. 설정에 따른 표시 방법은 아래와 같다

설정값	설 명
0	모든 대상 그룹에 대한 소계행 표시 (Default)
1	대상 그룹이 2개 이상인 그룹에 대해서만 소계행 표시

➤ Syntax

Syntax	ObjId. ShowSubSum (info)
--------	---------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
StdCol	Number/ String	필수	기준 컬럼 Index 또는 SaveName
SumCols	String	필수	소계가 계산되어야 할 컬럼 Index 를 " "로 연결한 문자열



Sort	String	선택	기준 컬럼의 Sort 처리 방식 (Default="")	
			설정값	설명
			"asc"	오름차순 정렬
			"desc"	내림차순 정렬
			""	사용 안함
ShowCumulate	Boolean	선택	소계에 대한 누계 표시 여부. Default=0	
CaptionCol	Long	선택	소계 대표 글자인 "소계:" + 기준 값을 설정할 컬럼 Default=숨겨지지 않은 첫컬럼	
CaptionText	String	선택	소계 대표 글자 포맷을 설정 Default=소계(누계): + 기준값	
AvgCols	String	선택	소계 행에 평균으로 계산되어야 할 컬럼 Index를 " "로 연결한 문 자열. Default=""	
Position	String	선택	소계행 생성 위치 설정 Default="bottom"	
OtherColText	String	선택	CaptionCol 이외의 컬럼에 설정할 글자를 조합한 문자열 Default=""	
Mode	Number	선택	소계행 표시 방법 Default=0	

➤ **Example**

```
//1컬럼을 중심으로 소계만 계산하여 표시하기
var info = [
    {StdCol:1, SumCols:"2|3|4|5|6|7", Sort:"desc"}
];
mySheet.ShowSubSum(info);

// 누계 까지 표시하기
```



```
var info = [  
    {StdCol:1, SumCols:"2|3|4|5|6|7", ShowCumulate:1, Sort:"desc"}  
];  
mySheet.ShowSubSum(info);  
  
//0컬럼과 1컬럼의 소계에 대한 소계 표시  
var info = [{StdCol:0, SumCols:"6|7",Sort: "asc"}  
            ,{StdCol:1, SumCols:"2|3|4|5", Sort:"desc",CaptionCol:1}];  
mySheet.ShowSubSum(info);
```

➤ **제공 버전**

7.0.0.0	
7.0.13.12	Sort 속성의 타입 변경 및 오름차순 내림차순 설정 지원
7.0.13.38	Position 속성 추가 소계 후 컬럼 Sort 기능 지원
7.0.13.40	OtherColText 속성 추가
7.0.13.51	Mode 속성 추가



ShowToolTip Method

➤ 기능

마우스 커서가 위치한 셀의 내용을 툴팁으로 표현한다.

이 기능은 초기화 함수인 SetConfig 의 ToolTip 인자를 이용하여 설정 할 수도 있다.

함수의 호출 시점은 초기화 이후 조회 처리전에 호출 하여야 한다.

➤ Syntax

Syntax	ObjId. ShowToolTip (ToolTip)
--------	-------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
ToolTip	Boolean	필수	툴팁 표시 여부

➤ Example

//툴팁 설정 mySheet.ShowToolTip(1);

➤ 제공 버전

7.0.0.0

ShowTreeSubSum Method

➤ 기능

트리구조에서 특정 컬럼의 데이터를 기준으로 소계를 계산하여 표시한다.

이 함수는 데이터를 조회하는 메소드를 호출하기 이전에 호출하여야 하며, 설정을 변경한 경우 다시 데이터를 조회하여야 적용된다.

인자값은 JSON 형태의 포맷으로 설정 한다.

예) var info = {SumCols:"2|3|4", AvgCols:"5|6", SumEx:1};

➤ Syntax

Syntax	ObjId. ShowTreeSubSum (Info)
--------	-------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Info.SumCols	String	선택	합계가 계산되어야 할 컬럼의 Index 또는 SaveName을 " "로 연결한 문자열. Default=""
Info.AvgCols	String	선택	평균이 계산되어야 할 컬럼의 Index 또는 SaveName을 " "로 연결한 문자열. Default=""
Info.CountCols	String	선택	개수가 계산되어야 할 컬럼의 Index 또는 SaveName을 " "로 연결한 문자열. Default=""
Info.MaxCols	String	선택	최대값이 계산되어야 할 컬럼의 Index 또는 SaveName을 " "로 연결한 문자열. Default=""
Info.MinCols	String	선택	최소값이 계산되어야 할 컬럼의 Index 또는 SaveName을 " "로 연결한 문자열. Default=""



Info.SumEx	Boolean	선택	상태값이 "삭제"인 행에 대한 계산 포함 여부. Default=0
Info.ExceptNull	Boolean	선택	평균 컬럼에서 Null 값 포함 계산 여부. Default=0

➤ **Example**

```
//2번째 컬럼에 대한 트리 소계 표현
var info = SumCols: "2";
mySheet.ShowTreeSubSum(info);

// 삭제행은 계산에서 제외하기
var info = {SumCols: "2", SumEx: "1"};
mySheet.ShowTreeSubSum (info);
```

➤ **제공 버전**

7.0.0.0	
7.0.12.0	MaxCols, MinCols 인자 속성 추가



SubSumBackColor Method

➤ 기능

소계 행의 배경 색상을 확인하거나 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetSubSumBackColor()
--------	-----	------------------------------------

➤ Info

Return	String, 현재 설정된 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//소계 행 배경색상을 확인한다.  
mySheet.GetSubSumBackColor();
```

➤ Syntax

Syntax	Set	ObjId. SetSubSumBackColor (Color)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Color	String	필수	설정하고자 하는 WebColor 값

➤ Example

```
//소계 행 배경색상을 녹색으로 설정  
mySheet.SetSubSumBackColor("#00FF00");
```

➤ 제공 버전

7.0.0.0



ShowTreeLevel Method

➤ 기능

데이터가 트리 형태일 때 보여질 트리 레벨을 설정한다.

Level은 다음과 같은 결과가 처리된다.

Level	설명
0	모두 접기
-1	모두 펼치기, Default
그 외	해당 레벨 만큼만 펼치기

ChildStatus는 보여질 트리 레벨 이하 레벨인 Child의 상태를 설정한다.

ChildStatus는 다음과 같은 값을 설정할 수 있다.

ChildStatus	설명
0	이전 상태를 그대로 유지, Default
1	모두 접음
2	모두 펼침

➤ Syntax

Syntax	ObjId. ShowTreeLevel ([Level], [ChildStatus])
--------	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Level	Integer	선택	보여질 트리 레벨 수준. Default=-1
ChildStatus	Integer	선택	보여질 트리 레벨 이하 레벨의 펼침 상태를 설정한다. Default=0

➤ Example

<pre>//모두 접기 mySheet.ShowTreeLevel(0, 1);</pre>



```
//모두 펼치기  
mySheet.ShowTreeLevel(-1);
```

- **제공 버전**
7.0.0.0



SumBackColor Method

➤ 기능

합계 행의 배경 색상을 확인하거나 설정한다.

색상 설정은 WebColor 값을 사용하여 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetSumBackColor()
--------	-----	---------------------------------

➤ Info

Return	String, WebColor 값(Get Method인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

//합계행 배경색상 구하기 mySheet.GetSumBackColor();
--

➤ Syntax

Syntax	Set	ObjId. SetSumBackColor (Color)
--------	-----	---------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Color	String	필수	WebColor 값

➤ Example

//합계 행 배경색상을 녹색으로 설정 mySheet.SetSumBackColor("#00FF00");

➤ 제공 버전



7.0.0.0



SumFontBold Method

➤ 기능

합계 행의 글자를 굵게 할지 여부를 확인하거나 설정한다.

이 속성을 1로 설정하면 합계 행의 글자를 굵게 표시하고, 0로 설정하면 보통으로 표시한다. 합계행의 글자가 굵으면 해당 컬럼의 다른 데이터와 정렬이 맞지 않는 단점이 있으며, 기본적인 설정값은 0 이다.

➤ Syntax

Syntax	Get	ObjId. GetSumFontBold ()
--------	-----	---------------------------------

➤ Info

Return	Boolean, 현재 폰트 굵기 여부 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//합계행 글자 볼드 설정 확인한다.
mySheet.GetSumFontBold();
```

➤ Syntax

Syntax	Set	ObjId. SetSumFontBold (Bold)
--------	-----	-------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Bold	Boolean	필수	폰트 굵기 여부

➤ Example

```
//합계 행의 글자 볼드를 설정한다.
mySheet.SetSumFontBold(1);
```



- 제공 버전
7.0.0.0



SumFontColor Method

➤ 기능

합계 행의 글자 색상을 확인하거나 설정한다.

색상 설정은 WebColor 값을 사용하여 설정한다

➤ Syntax

Syntax	Get	ObjId. GetSumFontColor()
--------	-----	---------------------------------

➤ Info

Return	String, 현재 합계행 글자 색상 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

//합계행 폰트색상을 확인한다. mySheet.GetSumFontColor();

➤ Syntax

Syntax	Set	ObjId. SetSumFontColor (Color)
--------	-----	---------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Color	String	필수	WebColor 값

➤ Example

//합계 행 폰트색상을 녹색으로 설정 mySheet.SetSumFontColor("#00FF00");

➤ 제공 버전



7.0.0.0



SumRowHidden Method

➤ 기능

합계행의 숨기여부를 설정하거나 확인한다.

➤ Syntax

Syntax	Get	ObjId. GetSumRowHidden()
--------	-----	---------------------------------

➤ Info

Return	Boolean, 숨김 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//합계행의 숨김 여부를 확인하여 숨겨진 경우 표시되도록 설정한다.
If(mySheet.GetSumRowHidden() == 1) {
    mySheet.SetSumRowHidden(0);
}
```

➤ Syntax

Syntax	Set	ObjId. SetSumRowHidden (Hidden)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Hidden	Boolean	필수	숨김여부

➤ Example

```
//합계행의 숨김 여부를 확인하여 숨겨진 경우 표시되도록 설정한다.
If(mySheet.GetSumRowHidden() == 1) {
    mySheet.SetSumRowHidden(0);
}
```




}

- 제공 버전
7.0.0.0



SumValue Method

➤ 기능

합계 셀의 값을 Format이 적용되지 않은 형태로 확인하거나 설정한다.
자동 계산되는 합계 셀에는 설정이 불가능 하다.

➤ Syntax

Syntax	Get	ObjId. GetSumValue (DataRow, Col)
--------	-----	--

➤ Info

Return	String, 합계 셀의 Value값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
DataRow	Number	필수	단위 데이터행의 Index
Col	Number / String	필수	합계 셀의 Column Index 또는 SaveName

➤ Example

//첫번째 합계 행의 2컬럼의 값을 읽어온다. 값이 1,234.56 이면 //1234.56이 반환된다.. alert("합계는 " + mySheet.GetSumValue(0, 2) + "입니다.");
--

➤ Syntax

Syntax	Set	ObjId. SetSumValue (DataRow, Col, Value)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
DataRow	Number	필수	단위 데이터행의 Index
Col	Number / String	필수	합계 셀의 Column Index 또는 SaveName
Value	Number /	필수	포맷 적용 안된 형태의 CellValue 값



	String		
--	--------	--	--

➤ **Example**

<pre>//첫번째 합계 행의 2컬럼의 값을 5432.12 로 바꾼다. mySheet.SetSumValue(0, 2, 5432.12);</pre>

➤ **제공 버전**

7.0.0.0



TabBehavior Method

➤ 기능

셀 선택 상태에서 Tab 키를 눌렀을 때의 동작 모드를 확인하거나 설정한다
True(1)로 설정시 셀의 편집 가능 여부와 상관없이 모든 셀에 대해서 탭 이동을 처리하고, false(0)로 설정시 편집 불가능한 셀을 제외하고 편집 가능한 셀에 대해서만 탭 이동을 처리 한다.

➤ Syntax

Syntax	Get	ObjId. GetTabBehavior()
--------	-----	--------------------------------

➤ Info

Return	Boolean, 설정한 속성 값		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// Tab키의 동작을 확인한다.
mySheet.GetTabBehavior();
```

➤ Syntax

Syntax	Set	ObjId. SetTabBehavior (Mode)
--------	-----	-------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Mode	Boolean	필수	Tab key 입력시 설정할 속성 값. Default= "1"

➤ Example

```
// 편집 가능한 셀에 대해서만 탭 이동을 설정 한다.
```



```
mySheet.SetTabBehavior(0);
```

➤ **제공 버전**

7.0.0.0



Theme Method

➤ 기능

IBSheet 의 테마 디자인을 확인 하거나 설정한다.

설정하기 위해서는 사전에 테마 디자인을 작업 하여야 한다. (세부 사항은 가이드문서 마지막 부분의 **Appendix 3.Theme 만들기** 참조)

➤ Syntax

Syntax	Get	ObjId. GetTheme ()
--------	-----	---------------------------

➤ Info

Return	String, 현재 테마의 Prefix 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//현재 설정되어 있는 테마를 확인
var Prefix = mySheet.GetTheme();
```

➤ Syntax

Syntax	Set	ObjId. SetTheme (Prefix, Folder)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Prefix	String	필수	테마의 Prefix 값
Folder	String	필수	테마의 폴더 명

➤ Example

```
//Gray 테마를 적용한다.
mySheet.SetTheme("GG", "Gray");
```



- 제공 버전
7.0.0.0



ToolTipText Method

➤ 기능

셀의 풍선도움말을 설정하거나 확인한다.

헤더 행이든 데이터 행이든 셀 각각의 풍선도움말을 설정할거나 확인할 수 있고, 해당 셀에 마우스를 잠시 머무르고 있으면 풍선도움말이 표시된다.

➤ Syntax

Syntax	Get	ObjId. GetToolTipText (Row, Col)
--------	-----	---

➤ Info

Return	String, 현재 설정된 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long / String	필수	해당 셀의 Column Index 또는 SaveName

➤ Example

// 풍선 도움말 설정값을 확인 alert(mySheet.GetToolTipText(1, 1));

➤ Syntax

Syntax	Set	ObjId. SetToolTipText (Row, Col, ToolTip)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Long	필수	해당 셀의 Row Index
Col	Long / String	필수	해당 셀의 Column Index 또는 SaveName
ToolTip	String	필수	풍선 도움말 설정 값



➤ **Example**

```
//셀에 풍선 도움말 설정
```

```
mySheet.SetToolTipText(1,1, "셀의 금액은 " + mySheet.GetCellText(1,1) + "입  
니다. ");
```

➤ **제공 버전**

7.0.0.0



TopRow Method

➤ 기능

최상단의 행 번호를 확인하거나 설정한다.

상단 헤더 영역을 제외하고 데이터 영역 안에서 최상단의 행을 설정하며 해당 행이 선택된 상태로 설정되지는 않는다.

설정 시 잘못된 행번호를 사용하여도 에러 메시지는 없이 가장 가까운 행이 최상단으로 사용되고, 해당 행이 최상단에 되기 위해 아래에 표시될 행의 개수가 충분할 때 가능하다.

예를 들어 총 건수가 103건이고, 한번에 20개씩 보인다면 TopRow를 100이라고 설정하여도 아래에 적어도 18개가 보여야 하므로 TopRow는 85로 변경된다. 따라서 보여질 영역에 표시되어야 할 행에 개수가 맞춰진 다음에 TopRow가 결정된다.

➤ Syntax

Syntax	Get	ObjId. GetTopRow ()
--------	-----	----------------------------

➤ Info

Return	Long, 최상단의 행 번호 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

//최상단 행 번호를 확인한다. mySheet.GetTopRow();

➤ Syntax

Syntax	Set	ObjId. SetTopRow (Row)
--------	-----	-------------------------------

➤ Info

Return	None
--------	------



Parameter	Type	필수여부	설 명
Row	Long	필수	Row Index

➤ **Example**

```
//최상단 행 번호를 설정한다.  
mySheet.SetTopRow(100);
```

➤ **제공 버전**

7.0.0.0



TotalRows Method

➤ 기능

조회 조건에 따라 전체 조회해야 할 DB의 레코드 건수를 확인하거나 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetTotalRows()
--------	-----	------------------------------

➤ Info

Return	Long, 현재 설정된 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//전체 데이터 건수를 확인한다.  
Alert("전체 데이터 건수는 " + mySheet.GetTotalRows() + "입니다. ");
```

➤ Syntax

Syntax	Set	ObjId. SetTotalRows (Count)
--------	-----	------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Count	Long	필수	전체 건수 설정값

➤ Example

```
//전체 데이터 건수를 1000으로 설정한다.  
mySheet.SetTotalRows(1000);
```

➤ 제공 버전



7.0.0.0



TreeActionMode Method

➤ 기능

트리 형태일때 각종 기능 처리 관련 속성을 확인하거나 설정한다.

속성값은 다음과 같다.

설정값	기능
0	자식이 있으면 삭제 안됨, 부모가 삭제이면 삭제 취소 할수 없음, (Default)
1	삭제 체크시 자식까지 삭제 체크됨

➤ Syntax

Syntax	Get	ObjId. GetTreeActionMode ()
--------	-----	------------------------------------

➤ Info

Return	Integer, 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

//현재 설정되어 있는 TreeActionMode 값 확인한다. Alert(mySheet.GetTreeActionMode());
--

➤ Syntax

Syntax	Set	ObjId. SetTreeActionMode (Value)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Value	Integer	필수	TreeActionMode 값



➤ **Example**

//삭제 체크를 눌렀을 때 자식레벨까지 삭제되도록 한다.
mySheet.SetTreeActionMode(1);

➤ **제공 버전**

7.0.0.0



TreeCheckActionMode Method

➤ 기능

트리구조의 시트에서 트리 기준 컬럼에 대한 체크박스 사용 모드인 경우 체크 값 변경시 자식노드에 대한 처리 모드를 확인하거나 설정 한다.

설정할 수 있는 처리 모드는 아래와 같다.

처리모드	설정값	기능
일반	0	자식노드에 대한 별도 처리 없음. (기본값)
관계모드	1	체크박스가 편집가능한 행에 대해서 아래와 같이 처리된다. - 체크상태가 변경된 행의 자식행에 동일한 값 처리 - 자식행이 1개이상 체크된 경우 해당 부모행의 체크 상태를 체크 처리 - 자식행이 모두 언체크인 경우 해당 부모행의 체크 상태를 언체크 처리
관계모드 (Force)	2	체크박스의 편집가능여부와 무관하게 모든 행에 대해서 "관계모드"과 같이 처리 된다.

➤ Syntax

Syntax	Get	ObjId. GetTreeCheckActionMode()
--------	-----	--

➤ Info

Return	Number, 현재 설정 값		
Parameter	Type	필수여부	설 명

➤ Example

```
//현재 설정되어 있는 TreeCheckActionMode 값을 확인한다.
Console.log("TreeCheckActionMode = ", mySheet.GetTreeCheckActionMode());
```

➤ Syntax



Syntax	Set	ObjId. SetTreeCheckActionMode (Mode)
--------	-----	---

➤ **Info**

Return	None		
Parameter	Type	필수여부	설 명
Mode	Number	필수	설정 모드 값

➤ **Example**

```
// 트리체크모드를 관계모드(Force)로 설정한다.  
mySheet.SetTreeCheckActionMode(2);
```

➤ **제공 버전**

7.0.9.0	
----------------	--



TreeCheckEditable Method

➤ 기능

트리구조의 시트에서 트리 기준 컬럼에 대한 체크박스 사용 모드인 경우 해당 컬럼의 모든 체크박스의 편집가능 여부를 확인하거나 설정한다.

Get 메소드의 반환값은 Set 메소드의 설정값을 기준으로 처리 한다.

행 단위의 편집가능 여부 설정은 TreeCheckRowEditable 메소드를 참고 한다.

➤ Syntax

Syntax	Get	ObjId. GetTreeCheckEditable ()
--------	-----	---------------------------------------

➤ Info

Return	Boolean, 컬럼단위 체크박스의 편집 가능 여부		
Parameter	Type	필수여부	설 명
없음			

➤ Example

```
// 컬럼단위 체크박스의 편집 가능 여부 확인
var edit = mySheet.GetTreeEditable();
console.log("TreeCheckEditable = ", edit);
```

➤ Syntax

Syntax	Set	ObjId. SetTreeCheckEditable (Edit)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Edit	Boolean	필수	체크박스의 편집 가능 여부

➤ Example



```
// 트리 컬럼의 체크박스에 대한 비활성화 설정.  
mySheet.SetTreeCheckEditable(0);
```

➤ **제공 버전**

7.0.9.0	
----------------	--

TreeCheckRowEditable Method

➤ 기능

트리구조의 시트에서 트리 기준 컬럼에 대한 체크박스 사용 모드인 경우 대상 행의 체크박스의 편집가능 여부를 확인하거나 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetTreeCheckRowEditable (Row)
--------	-----	---

➤ Info

Return	Boolean, 컬럼단위 체크박스의 편집 가능 여부		
Parameter	Type	필수여부	설 명
Row	Number	필수	대상 행의 Index

➤ Example

```
// Index가 3인 행의 체크박스 편집가능여부 확인
var edit = mySheet.GetTreeRowEditable(3);
console.log("RowIndex = 3", "CheckEditable =", edit);
```

➤ Syntax

Syntax	Set	ObjId. SetTreeCheckRowEditable (Row, Edit)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Number	필수	대상 행의 Index
Edit	Boolean	필수	체크박스의 편집 가능 여부

➤ Example

```
// Index가 3인 행의 트리 컬럼 체크박스의 편집가능여부를 토글한다.
mySheet.SetTreeCheckRowEditable(3, !mySheet.GetTreeCheckRowEditable(3));
```



➤ 제공 버전

7.0.9.0	
---------	--



TreeCheckValue Method

➤ 기능

트리구조의 시트에서 트리 기준 컬럼에 대한 체크박스 사용 모드인 경우 해당 체크박스의 값을 확인하거나 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetTreeCheckValue (Row)
--------	-----	---------------------------------------

➤ Info

Return	Number, 해당 체크박스의 값		
Parameter	Type	필수여부	설 명
Row	Number	필수	대상 행의 Index

➤ Example

```
// Index 가 3인 행에 대한 체크박스 값 확인
var val = mySheet.GetTreeCheckValue(3);
console.log("RowIndex = 3", "Check =", val);
```

➤ Syntax

Syntax	Set	ObjId. SetTreeCheckValue (Row, Value, Event)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Number	필수	대상 행의 Index
Value	Boolean	필수	체크박스 설정 값
Event	Boolean	선택	설정시 OnTreeCheckChange 이벤트 발생 여부 (기본값:1)

➤ Example



```
// Index가 3인 행의 트리 컬럼 체크값을 checked 로 설정한다.  
mySheet.SetTreeCheckValue(3, 1);
```

➤ **제공 버전**

7.0.9.0	
----------------	--



TreeChildSort Method

➤ 기능

트리 구조의 시트에서 대상 행의 자식 노드에 대한 Sort 처리를 수행 한다.
Sort 처리시 대상 행의 1레벨 하위의 노드에 대해서만 처리한다.

➤ Syntax

Syntax	ObjId.TreeChildSort()
--------	-----------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Row	Number	필수	대상 행의 Index
Col	String	필수	Sort 기준 컬럼의 Index 또는 SaveName 을 구분자로 연결한 문자열
SortOrder	String	선택	컬럼에 대한 정렬 방식을 구분자로 연결한 문자열.(ASC, DESC) Default=ASC

➤ Example

```
// Index가 3인 행의 자식노드를 2번째 컬럼을 기준으로 정렬한다.
mySheet.TreeChildSort(3, 2, "ASC");
```

➤ 제공 버전

7.0.6.0	
---------	--

UnicodeByte Method

➤ 기능

한글 1자리의 바이트 수를 확인하거나 설정 한다.

자바스크립트에서는 모든 글자를 1바이트로 인식한다.

그러나 한글이나 일본어, 중국어와 같은 글자들은 DB의 언어 설정에 따라 2바이트 이상으로 인식되어야 하는 경우가 있다.

이런 경우 이 메소드를 이용하여 한글의 바이트 수를 설정 할 수 있다.

Utf-8 형식에서는 문자에 따라서 바이트수가 달라지게 되는데 이런 경우 설정 인자를 "utf-8"로 설정하면 utf-8 형식에 맞게 처리 할 수 있다.

➤ Syntax

Syntax	Get	ObjId. GetUnicodeByte ()
--------	-----	---------------------------------

➤ Info

Return	Integer, 현재 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

// 설정된 바이트 수를 확인한다. mySheet. GetUnicodeByte();

➤ Syntax

Syntax	Set	ObjId. SetUnicodeByte (Value)
--------	-----	--------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Value	Number/ String	필수	설정할 바이트 수 또는 설정모드 (Default=1)



➤ **Example**

```
// 한글 1자리의 바이트수를 3으로 설정한다.  
mySheet. SetUnicodeByte(3);
```

➤ **제공 버전**

7.0.0.0	
7.0.5.0	Utf-8 설정모드 추가

UseDefaultTime Method

➤ 기능

Format이 "Hms", "Hm" 인 경우 셀의 값이 공백인 상태에서 편집을 처음 시작하고자 할때 기본적으로 시스템 현재시간을 표시한다. 이 속성을 0으로 설정하면 시스템 현재 시간을 표시하지 않고 공백상태로 표시한다. 이 속성의 기본 값은 1이다

➤ Syntax

Syntax	Get	ObjId. GetUseDefaultTime()
--------	-----	-----------------------------------

➤ Info

Return	Boolean, 현재 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//현재 시간을 표시 설정 확인
mySheet.GetUseDefaultTime();
```

➤ Syntax

Syntax	Set	ObjId. SetUseDefaultTime(Value)
--------	-----	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Value	Boolean	필수	현재시간 표시 유무 (Default=1)

➤ Example

```
//현재 시간을 표시하지 않도록 설정
mySheet.SetUseDefaultTime(0);
```



- 제공 버전
7.0.0.0

UserAgent Method

➤ 기능

조회나 저장시 HTTP 헤더 정보에 포함되어 넘어가는 IBUserAgent의 값을 확인하거나 설정한다.

➤ Syntax

Syntax	Get	ObjId. GetUserAgent ()
--------	-----	-------------------------------

➤ Info

Return	String, 현재 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// IBUserAgent의 값을 확인한다.
mySheet.GetUserAgent();
```

➤ Syntax

Syntax	Set	ObjId. SetUserAgent (Value)
--------	-----	------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Value	String	필수	User-Agent 설정값

➤ Example

```
// HTTP 헤더정보의 IBUserAgent 값을 My Agent Name 으로 전송
mySheet.SetUserAgent("My Agent Name");
```

➤ 제공 버전



7.0.0.0

UploadingImage Method

➤ 기능

파일 업로드 중 대기 이미지 파일의 위치를 설정하거나 확인한다.

이 속성은 기본적으로 제공하는 파일 업로드 중 대기 이미지를 사용자가 원하는 이미지로 변경설정 가능하기 위한 속성이다.

➤ Syntax

Syntax	Get	ObjId. GetUploadingImage()
--------	-----	-----------------------------------

➤ Info

Return	String, 현재 설정값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//현재 설정되어있는 업로드 중 대기 이미지 경로를 확인한다.
Alert(mySheet.GetUploadingImage());
```

➤ Syntax

Syntax	Set	ObjId. SetUploadingImage (Url)
--------	-----	---------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Url	String	필수	이미지 URL

➤ Example

```
//업로드 중 대기 이미지를 변경한다.
mySheet.SetUploadingImage( "/sheet/imgUpload.gif");
```



- 제공 버전
7.0.0.0

ValidateFail Method

➤ 기능

저장 함수를 호출하여 저장 처리 전에 저장할 각 데이터의 Validation 확인을 위한 OnValidation 이벤트에서 유효하지 않은 데이터를 발견하여 저장처리를 중단하고자 할 때 이 속성을 설정하여 처리를 중단한다.

OnEditValidation 이벤트에서 사용하는 경우 Fail 인자의 설정에 따라 다음과 같이 처리 할 수 있다. Fail 설정 값이 2 이고 Message 인자를 설정한 경우 Message 인자에 설정한 값을 메시지로 표시 한다.

Fail 설정 값	설명
1	Invalid 처리 (입력값 취소 / 편집 완료 처리)
2	Invalid 처리 (입력값의 상태로 편집상태 유지)

➤ Syntax

Syntax	ObjId. ValidateFail (Fail, [Message])
--------	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Fail	Number	필수	저장 처리 중단 여부
Message	String	선택	Fail 인자 설정이 2인 경우 표시할 메시지

➤ Example

```
// OnValidation 이벤트에서의 사용
function mySheet_OnValidation(Row, Col, Value) {
    if(Col == 2)    {
        if(Value=="원화" && mySheet.GetCellValue(Row,Col+1) >= 10000000) {
            alert("통화가 원화일 때 금액은 천만원 이상일수없습니다. ");
            mySheet.ValidateFail(1);
        }
    }
}
```



```
mySheet.SelectCell(Row, Col+1);
} else if(Value=="외화" && mySheet.GetCellValue(Row,Col+1) <
10000000) {
    alert("통화가 외화일 때 금액은 천만원 이상이어야 합니다. ");
    mySheet.ValidateFail(1);
    mySheet.SelectCell(Row, Col+1);
}
}
}

// OnEditValidation 이벤트에서의 사용
function mySheet_OnEditValidation(Row, Col, Value) {
    if(Col == 2 && Value > 100) {
        // alert 으로 메시지를 표시하고 편집상태를 유지한다.
        mySheet.ValidateFail(2, "금액 최대값은 100입니다");
    }
}
```

➤ **제공 버전**

7.0.0.0	
7.0.1.0	Fail시 편집 유지 모드 추가, Message 인자 추가



Visible Method

➤ 기능

IBSheet의 표시 여부를 설정하거나 확인한다.

0으로 설정하면 건수 정보를 포함한 모든 것이 숨겨져서 보이지 않으며, 1로 설정하면 모든 정보를 볼 수 있다.

➤ Syntax

Syntax	Get	ObjId. GetVisible()
--------	-----	----------------------------

➤ Info

Return	Boolean, 현재 설정 값 (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// 시트 표시 여부를 확인
if(mySheet.GetVisible()){
    alert("IBSheet를 표시함");
}else{
    alert("숨겨져 있음");
}
```

➤ Syntax

Syntax	Set	ObjId. SetVisible (Visible)
--------	-----	------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Visible	Boolean	필수	화면에 시트 표시 여부



➤ **Example**

```
// 시트를 안보이게 설정  
mySheet.SetVisible(0);
```

➤ **제공 버전**

7.0.0.0

WaitImage Method

➤ 기능

처리 중 대기 이미지 파일의 위치를 설정하거나 확인한다.

이 속성은 기본적으로 제공하는 처리 중 대기 이미지를 사용자가 원하는 이미지로 변경설정 가능하기 위한 속성으로 조회 중 대기 이미지와 저장 중 대기 이미지를 모두 포함하고 있는 처리 중 대기 이미지이다.

➤ Syntax

Syntax	Get	ObjId. GetWaitImage ()
--------	-----	-------------------------------

➤ Info

Return	String, 현재 설정 되어 있는 URL (Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
//처리 중 대기 이미지 위치를 확인한다.
mySheet.GetWaitImage();
```

➤ Syntax

Syntax	Set	ObjId. SetWaitImage (Url)
--------	-----	----------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Url	String	필수	이미지 Url

➤ Example

```
//처리 중 대기 이미지를 변경한다.
mySheet.SetWaitImage("/sheet/imgWait.gif");
```



- 제공 버전
7.0.0.0

WaitImageVisible Method

➤ 기능

각종 처리 중 대기 이미지의 표시여부를 설정하거나 확인한다.

조회,저장과 같은 EndUser의 대기 시간이 소요되는 함수를 호출하면 기본적으로 대기이미지를 표시한다.

이렇게 표시되는 대기 이미지를 어떤 이유로 사용하기 않고자 할 때 이 속성을 0으로 설정하여 대기 이미지를 더 이상 표시하지 않도록 한다.

이 속성의 설정에 따라서 모든 처리 중의 대기 이미지가 결정되어 표시되기도 하고, 표시 되지 않을수도 있다.

➤ Syntax

Syntax	Get	ObjId. GetWaitImageVisible ()
--------	-----	--------------------------------------

➤ Info

Return	Boolean, 현재 설정 값(Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

//처리 중 대기 이미지 표시 여부를 확인한다. Alert(mySheet.GetWaitImageVisible());

➤ Syntax

Syntax	Set	ObjId. SetWaitImageVisible (Visible)
--------	-----	---

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Visible	Boolean	필수	각종 대기 이미지 표시 여부



➤ **Example**

```
//처리 중 대기 이미지를 표시하지 않도록 설정한다..  
mySheet.SetWaitImageVisible(0);
```

➤ **제공 버전**

7.0.0.0



WaitTimeOut Method

➤ 기능

서버에 연결하여 응답을 대기하는 시간을 확인하거나 설정한다. (단위:초).

➤ Syntax

Syntax	Get	ObjId. GetWaitTimeOut ()
--------	-----	---------------------------------

➤ Info

Return	Integer, 현재 설정 값(Get Method 인 경우)		
Parameter	Type	필수여부	설 명

➤ Example

```
// 서버에 연결하여 응답을 대기하는 시간을 확인  
alert(mySheet.GetWaitTimeOut());
```

➤ Syntax

Syntax	Set	ObjId. SetWaitTimeOut (time)
--------	-----	-------------------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
time	Integer	필수	설정할 초시간 (Default=60)

➤ Example

```
// 서버에 연결하여 응답을 대기하는 시간을 확인하거나 설정  
mySheet.SetWaitTimeOut(120);
```

➤ 제공 버전

7.0.0.0





Chapter 10. IBSheet Global Module





10. IBSheet Global Module

10.1 Global Module 사용

IBSheet에서 제공하는 Global Module을 사용하기 위해서는 ibsheet.js 파일을 해당 페이지내에 반드시 include해야만 사용이 가능하다.

IBSheet가 제공하는 모듈은 다음과 같다.



10.2 Method List

IBCalendar.SetTheme Method

➤ 기능

외부 컨트롤에서 사용할 달력팝업의 테마 디자인을 설정한다.

설정하기 위해서는 사전에 테마 디자인을 작업 하여야 한다.

(세부 사항은 가이드문서 마지막 부분의 **Appendix 3.Theme 만들기** 참조)

➤ Syntax

Syntax	IBCalendar.SetTheme (Prefix, Folder);
--------	--

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
Prefix	String	필수	테마의 Prefix 값
Folder	String	필수	테마의 폴더 명

➤ Example

//Orange 테마를 적용한다. IBCalendar.SetTheme("OR", "Orange");
--

➤ 제공 버전

7.0.0.0



IBCalendar.Show Method

➤ 기능

달력 팝업을 외부의 컨트롤에서 사용하도록 설정한다

일반적으로 CallBack 함수를 지정하는 경우 첫번째 인자로 선택한 날짜 문자열이 전달되며, 그 외 CallBack 함수로 전달하고자 하는 인자가 있는 경우 CallBackParam 을 설정한다. CallBackParam으로 설정된 객체는 CallBack 함수의 두번째 인자로 전달된다.

➤ Syntax

Syntax	IBCalendar.Show (val, obj);
--------	------------------------------------

➤ Info

Return	String, (달력 팝업 Dialog에서 선택한 날짜값)		
Parameter	Type	필수여부	설 명
val	String	필수	날짜데이터값 (Default=현재날짜)
obj	Object	선택	기능을 json 형태로 설정한다.

세부내용

컬럼별로 설정 할 수 있는 속성은 다음과 같다.

속성	타입	설명	
CalButtonAlign	String	달력팝업에서 버튼을 사용하는 경우 정렬방법을 설정	
		설정값	내용
		Left	왼쪽정렬
		Center	가운데정렬 (기본값)
		Right	오른쪽정렬



CalButtons	String	달력팝업의 사용할 버튼 설정 사용하고자 하는 버튼을 " " 로 연결하여 설정한다.												
		<table><tr><th>설정값</th><th>내용</th></tr><tr><td>Close</td><td>취소 버튼</td></tr><tr><td>Today</td><td>오늘일자 입력 버튼</td></tr><tr><td>Yesterday</td><td>어제일자 입력 버튼</td></tr><tr><td>InputEmpty</td><td>빈값 입력 버튼</td></tr><tr><td>Ok</td><td>선택 버튼</td></tr></table>	설정값	내용	Close	취소 버튼	Today	오늘일자 입력 버튼	Yesterday	어제일자 입력 버튼	InputEmpty	빈값 입력 버튼	Ok	선택 버튼
		설정값	내용											
		Close	취소 버튼											
		Today	오늘일자 입력 버튼											
		Yesterday	어제일자 입력 버튼											
		InputEmpty	빈값 입력 버튼											
Ok	선택 버튼													
CallBack	String / Function	allBack함 수명 또는 함수 객체												
CallBackParam	Object	CallBack 함수에서 받을 인자 객체												
Format	String	날짜포맷 패턴(Default="yyyy/MM/dd")												
Holiday	String	달력에 표시할 사용자 정의 휴일 설정 파이프 연산자로 설정값을 연결한 문자열로 설정 할 수 있으며 포맷은 yyyyMMdd 만 허용한다. 매년, 매월에 대한 와일드카드로 * 을 사용한다. 설정 예) Holiday : "20120725 *0703 2012*27 **17" *0703 : 매년 7월 3일 2012*27 : 2012년 매월 27일 **17 : 매년, 매월 17일												
Result	Object	선택 결과를 설정할 객체 (input 객체)												
ShowYM	Boolean	년월 선택화면 우선 표시 여부												
Target	String / Object	"Mouse" (마지막 마우스 위치 사용시, Default) 또는 달력 버튼 Object (달력 버튼 위치 사용시)												
X	Number	(좌표값 사용시), X 축 좌표값												
Y	Number	(좌표값 사용시), Y 축 좌표값												



➤ **Example**

```
// 날짜데이터값 입력
var val = document.getElementById("DateText").value;

// 달력 팝업 Dialog 위치 : X, Y 좌표값 사용시
var obj = { Format:"yyyy/MM/dd", X:300, Y:600, CallBack:"Test" };

// 달력 팝업 Dialog 위치 : 마우스 마지막 위치 사용시
var obj = { Format:"yyyy/MM/dd", Target:"Mouse", CallBack: "Test" };

// 달력 팝업 Dialog 위치 : 달력 버튼 위치 사용시
var obj = { Target:document.getElementById("DateBtn"), CallBack: "Test" };

// CalButtons 속성 : 달력 닫기 버튼 옵션 설정 사용시
var obj = { Format:"Ymd", Target:document.getElementById("DateBtn"),
  CallBack: "Test" , CalButtons : "Close"};

// 달력 닫기 버튼 옵션 설정 사용시 정렬 설정: 왼쪽 정렬
var obj = { Format:"Ymd", Target:document.getElementById("DateBtn"),
  CallBack: "Test", CalButtons : "Close", CalButtonAlign : "Left" };

// 사용자 정의 휴일 설정
var obj = { Format:"Ymd", Target:document.getElementById("DateBtn"),
  CallBack: "Test", Holiday: " 20101215|*1203"};

// 달력 출력
IBCalendar.Show("20140415", obj);

// fnName : 함수명으로 function 생성
function Test (date){
  // 달력 팝업 Dialog서 날짜 선택시 리턴값
  document.getElementById("DateText").value = date;
```



```
}
```

➤ 제공 버전

7.0.0.0	
7.0.11.0	CalButtons 속성의 "InputEmpty" 값 추가
7.0.13.10	CalButtons 속성의 "Ok" 값 추가
7.0.13.16	ShowYM 속성 추가



IBCalendar.Close Method

➤ 기능

외부의 컨트롤에서 사용한 달력팝업을 닫도록 설정한다.

➤ Syntax

Syntax	IBCalendar.Close();
--------	----------------------------

➤ Info

Return	None		
Parameter	Type	필수여부	설 명
없음			

```
// 달력 팝업을 띄운다.  
IBCalendar.Show("20121116", {Format : "yyyy/MM/dd", X : 300, Y : 600}) ;  
  
// 띄운 달력 팝업 Dialog를 닫는다.  
IBCalendar.Close();
```

➤ 제공 버전

7.0.0.0



(주) 아이비 리더스

Inno Biz Leaders Co., Ltd.



Appendix



(주) 아이비 리더스

Inno Biz Leaders Co., Ltd.



1. 컬럼 타입별 정의 가능한 속성

InitColumns 메소드에서 컬럼 정의시 타입별 설정 가능한 속성은 아래의 표와 같다.

1.1. 텍스트 형식 타입

속성	Text	Popup
Type	O	O
AcceptKeys	O	X
Align	O	O
ApproximateType	X	X
BackColor	O	O
ButtonUrl	X	O
CalcLogic	X	X
CaseSensitive	O	X
Chart	X	X
CheckSaveName	O	O
ClassName	X	X
ColMerge	O	O
ColSpan	O	O
ComboCode	X	X
ComboFilter	X	X
ComboText	X	X
Cursor	O	O
DefaultValue	O	O
Edit	O	X
EditLen	O	X
EditPointCount	X	X



Ellipsis	O	O
EmptyToReplaceChar	X	X
ExcludeEmpty	X	X
FalseValue	X	X
Focus	O	O
FontBold	O	O
FontColor	O	O
FontUnderline	O	O
Format	O	O
FormatFix	O	O
FullInput	O	X
GroupSumType	O	O
HeaderCheck	X	X
Hidden	O	O
HoverUnderline	O	O
Image	O	X
ImgAlign	O	X
ImgHeight	O	X
ImgWidth	O	X
InsertEdit	O	O
ItemCode	X	X
ItemText	X	X
KeyField	O	O
LevelSaveName	O	O
MaxCheck	X	X
MaximumValue	X	X
JMinimumValue	X	X
MultiLineText	O	X
PointCount	X	X
PopupButton	O	X
PopupCheckEdit	O	O



PopupCode	O	O
PopupText	O	O
RadiolIcon	X	X
RowSpan	O	O
SaveName	O	O
Sort	O	O
SumType	X	X
ToolTipText	O	O
TreeCheck	O	O
TreeCol	O	O
TrueValue	X	X
UpdateEdit	O	O
VAlign	O	O
Validation	X	X
Width	O	O
Wrap	O	O
ZeroToReplaceChar	X	X

1.2. 날짜 형식 타입

속성	Date
Type	O
AcceptKeys	X
Align	O
ApproximateType	X
BackColor	O
ButtonUrl	X
CalcLogic	X
CaseSensitive	X
Chart	X
CheckSaveName	O
ClassName	X



ColMerge	O
ColSpan	O
ComboCode	X
ComboFilter	X
ComboText	X
Cursor	O
DefaultValue	O
Edit	O
EditLen	O
EditPointCount	X
Ellipsis	O
EmptyToReplaceChar	X
ExcludeEmpty	X
FalseValue	X
Focus	O
FontBold	O
FontColor	O
FontUnderline	O
Format	O
FormatFix	O
FullInput	O
GroupSumType	O
HeaderCheck	X
Hidden	O
HoverUnderline	O
Image	X
ImgAlign	X
ImgHeight	X
ImgWidth	X
InsertEdit	O
ItemCode	X



ItemText	X
KeyField	O
LevelSaveName	O
MaxCheck	X
MaximumValue	X
MinimumValue	X
MultiLineText	X
PointCount	X
PopupCheckEdit	O
PopupButton	X
PopupCode	O
PopupText	O
RadiolIcon	X
RowSpan	O
SaveName	O
Sort	O
SumType	X
ToolTipText	O
TreeCheck	O
TreeCol	O
TrueValue	X
UpdateEdit	O
VAlign	O
Validation	X
Width	O
Wrap	X
ZeroToREplaceChar	X

1.3. 숫자 형식 타입

속성	Int	Float	AutoSum
Type	O	O	O



AcceptKeys	X	X	X
ApproximateType	O	O	O
Align	O	O	O
BackColor	O	O	O
ButtonUrl	X	X	X
CalcLogic	O	O	O
CaseSensitive	X	X	X
Chart	X	X	X
CheckSaveName	O	O	O
ClassName	X	X	X
ColMerge	O	O	O
ColSpan	O	O	O
ComboCode	X	X	X
ComboFilter	X	X	X
ComboText	X	X	X
Cursor	O	O	O
DefaultValue	O	O	O
Edit	O	O	O
EditLen	O	O	O
EditPointCount	X	O	O
Ellipsis	O	O	O
EmptyToReplaceChar	O	O	O
ExcludeEmpty	X	X	X
FalseValue	X	X	X
Focus	O	O	O
FontBold	O	O	O
FontColor	O	O	O
FontUnderline	O	O	O
Format	O	O	O
FormatFix	O	O	O
FullInput	O	O	O



GroupSumType	O	O	O
HeaderCheck	X	X	X
Hidden	O	O	O
HoverUnderline	O	O	O
Image	O	O	O
ImgAlign	O	O	O
ImgHeight	O	O	O
ImgWidth	O	O	O
InsertEdit	O	O	O
ItemCode	X	X	X
ItemText	X	X	X
KeyField	O	O	O
LevelSaveName	O	O	O
MaxCheck	X	X	X
MaximumValue	O	O	O
MinimumValue	O	O	O
MultiLineText	X	X	X
PointCount	X	O	O
PopupButton	O	O	O
PopupCheckEdit	O	O	O
PopupCode	O	O	O
PopupText	O	O	O
Radiolcon	X	X	X
RowSpan	O	O	O
SaveName	O	O	O
Sort	O	O	O
SumType	X	X	O
ToolTipText	O	O	O
TreeCheck	O	O	O
TreeCol	O	O	O
TrueValue	X	X	X



UpdateEdit	O	O	O
VAlign	O	O	O
Validation	X	X	X
Width	O	O	O
Wrap	X	X	X
ZeroToReplaceChar	O	O	O

1.4. 체크박스 형식 타입

속성	DelCheck	CheckBox	DummyCheck	Radio
Type	O	O	O	O
AcceptKeys	X	X	X	X
Align	X	X	X	X
ApproximateType	X	X	X	X
BackColor	O	O	O	O
ButtonUrl	X	X	X	X
CalcLogic	X	X	X	X
CaseSensitive	X	X	X	X
Chart	X	X	X	X
CheckSaveName	O	O	O	O
ClassName	X	X	X	X
ColMerge	O	O	O	O
ColSpan	O	O	O	O
ComboCode	X	X	X	X
ComboFilter	X	X	X	X
ComboText	X	X	X	X
Cursor	X	X	X	X
DefaultValue	X	O	O	X
Edit	O	O	O	O
EditLen	X	X	X	X
EditPointCount	X	X	X	X
Ellipsis	O	O	O	O



EmptyToReplaceChar	X	X	X	X
ExcludeEmpty	X	X	X	X
FalseValue	O	O	O	O
Focus	O	O	O	O
FontBold	X	X	X	X
FontColor	X	X	X	X
FontUnderline	X	X	X	X
Format	X	X	X	X
FormatFix	O	O	O	O
FullInput	X	X	X	X
GroupSumType	O	O	O	O
HeaderCheck	O	O	O	X
Hidden	O	O	O	O
HoverUnderline	X	X	X	X
Image	X	X	X	X
ImgAlign	X	X	X	X
ImgHeight	X	X	X	X
ImgWidth	X	X	X	X
InsertEdit	X	O	O	O
ItemCode	X	O	O	X
ItemText	X	O	O	X
KeyField	O	O	O	O
LevelSaveName	O	O	O	O
MaxCheck	X	O	O	X
MaximumValue	X	X	X	X
MinimumValue	X	X	X	X
MultiLineText	X	X	X	X
PointCount	X	X	X	X
PopupButton	X	X	X	X
PopupCheckEdit	O	O	O	O
PopupCode	O	O	O	O



PopupText	O	O	O	O
Radiolcon	O	O	O	O
RowSpan	O	O	O	O
SaveName	O	O	O	O
Sort	O	O	O	O
SumType	X	X	X	X
ToolTipText	O	O	O	O
TreeCheck	O	O	O	O
TreeCol	O	O	O	O
TrueValue	O	O	O	O
UpdateEdit	X	O	O	O
VAlign	X	X	X	X
Validation	X	X	X	X
Width	O	O	O	O
Wrap	X	X	X	X
ZeroToReplaceChar	X	X	X	X

1.5. 콤보 형식 타입

속성	Combo	ComboEdit
Type	O	O
AcceptKeys	X	O
Align	O	O
ApproximateType	X	X
BackColor	O	O
ButtonUrl	X	X
CalcLogic	X	X
CaseSensitive	X	O
Chart	X	X
CheckSaveName	O	O
ClassName	X	X
ColMerge	O	O



ColSpan	O	O
ComboCode	O	O
ComboFilter	X	O
ComboText	O	O
Cursor	X	O
DefaultValue	O	O
Edit	X	O
EditLen	X	O
EditPointCount	X	X
Ellipsis	O	O
EmptyToReplaceChar	X	X
ExcludeEmpty	X	X
FalseValue	X	X
Focus	O	O
FontBold	O	O
FontColor	O	O
FontUnderline	O	O
Format	X	X
FormatFix	O	O
FullInput	X	O
GroupSumType	O	O
HeaderCheck	X	X
Hidden	O	O
HoverUnderline	O	O
Image	X	X
ImgAlign	X	X
ImgHeight	X	X
ImgWidth	X	X
InsertEdit	O	O
ItemCode	X	X
ItemText	X	X



KeyField	O	O
LevelSaveName	O	O
MaxCheck	X	X
MaximumValue	X	X
MinimumValue	X	X
MultiLineText	X	X
PointCount	X	X
PopupButton	X	X
PopupCheckEdit	O	O
PopupCode	O	O
PopupText	O	O
Radiolcon	X	X
RowSpan	O	O
SaveName	O	O
Sort	O	O
SumType	X	X
ToolTipText	O	O
TreeCheck	O	O
TreeCol	O	O
TrueValue	X	X
UpdateEdit	O	O
VAlign	O	O
Validation	X	O
Width	O	O
Wrap	X	O
ZeroToReplaceChar	X	X

1.6. Html 형식 타입

속성	Html	Button	Img
Type	O	O	O
AcceptKeys	X	X	X



Align	X	O	O
ApproximateType	X	X	X
BackColor	X	O	O
ButtonUrl	X	X	X
CalcLogic	X	X	X
CaseSensitive	X	X	X
Chart	X	X	X
CheckSaveName	X	X	O
ClassName	X	O	X
ColMerge	O	O	O
ColSpan	O	O	O
ComboCode	X	X	X
ComboFilter	X	X	X
ComboText	X	X	X
Cursor	O	O	O
DefaultValue	O	O	X
Edit	X	O	X
EditLen	X	X	X
EditPointCount	X	X	X
Ellipsis	X	X	X
EmptyToReplaceChar	X	X	X
ExcludeEmpty	X	X	X
FalseValue	X	X	X
Focus	O	O	O
FontBold	X	X	X
FontColor	X	X	X
FontUnderline	X	X	X
Format	X	X	X
FormatFix	X	X	O
FullInput	X	X	X
GroupSumType	X	X	O



HeaderCheck	X	X	X
Hidden	O	O	O
HoverUnderline	X	X	X
Image	X	X	X
ImgAlign	X	X	X
ImgHeight	X	X	O
ImgWidth	X	X	O
InsertEdit	X	X	O
ItemCode	X	X	X
ItemText	X	X	X
KeyField	O	O	O
LevelSaveName	X	X	O
MaxCheck	X	X	X
MaximumValue	X	X	X
MinimumValue	X	X	X
MultiLineText	X	X	X
PointCount	X	X	X
PopupButton	X	X	X
PopupCheckEdit	X	X	O
PopupCode	O	O	O
PopupText	O	O	O
Radiolcon	X	X	X
RowSpan	O	O	O
SaveName	O	O	O
Sort	O	O	O
SumType	X	X	X
ToolTipText	O	O	O
TreeCheck	X	X	O
TreeCol	O	O	O
TrueValue	X	X	X
UpdateEdit	X	X	O



VAlign	X	X	O
Validation	X	X	X
Width	O	O	O
Wrap	X	X	X
ZeroToReplaceChar	X	X	X

1.7 기타 형식 타입

속성	Result	Pass	Status	Seq	Sparkline
Type	O	O	O	O	O
AcceptKeys	X	X	X	X	X
Align	O	O	O	O	X
ApproximateType	X	X	X	X	X
BackColor	O	O	O	O	X
ButtonUrl	X	X	X	X	X
CalcLogic	X	X	X	X	X
CaseSensitive	O	X	X	X	X
Chart	X	X	X	X	O
CheckSaveName	O	O	O	O	X
ClassName	X	X	X	X	X
ColMerge	O	O	O	X	X
ColSpan	O	O	O	O	O
ComboCode	X	X	X	X	X
ComboFilter	X	X	X	X	X
ComboText	X	X	X	X	X
Cursor	O	O	O	O	X
DefaultValue	O	X	X	X	X
Edit	X	O	X	X	X
EditLen	X	O	X	X	X
EditPointCount	X	X	X	X	X
Ellipsis	O	O	O	O	X



EmptyToReplaceChar	X	X	X	X	X
ExcludeEmpty	X	X	X	X	X
FalseValue	X	X	X	X	X
Focus	O	O	O	O	O
FontBold	O	O	O	O	X
FontColor	O	O	O	O	X
FontUnderline	O	O	O	O	X
Format	X	X	X	X	X
FormatFix	O	O	O	O	X
FullInput	X	O	X	X	X
GroupSumType	O	O	O	O	X
HeaderCheck	X	X	X	X	X
Hidden	O	O	O	O	O
HoverUnderline	O	O	O	O	X
Image	X	X	X	X	X
ImgAlign	X	X	X	X	X
ImgHeight	X	X	X	X	X
ImgWidth	X	X	X	X	X
InsertEdit	O	O	O	O	X
ItemCode	X	X	X	X	X
ItemText	X	X	X	X	X
KeyField	O	O	O	O	O
LevelSaveName	O	O	O	O	X
MaxCheck	X	X	X	X	X
MaximumValue	X	X	X	X	X
MinimumValue	X	X	X	X	X
MultiLineText	X	X	X	X	X
PointCount	X	X	X	X	X
PopupButton	X	X	X	X	X
PopupCheckEdit	O	O	O	O	X
PopupCode	O	O	O	O	X



PopupText	O	O	O	O	X
Radiolcon	X	X	X	X	X
RowSpan	O	O	O	O	O
SaveName	O	O	O	O	O
Sort	O	O	O	O	X
SumType	X	X	X	X	X
ToolTipText	O	O	O	O	X
TreeCheck	O	O	O	O	X
TreeCol	O	O	O	O	X
TrueValue	X	X	X	X	X
UpdateEdit	O	O	O	O	X
VAlign	O	O	O	O	X
Validation	X	X	X	X	X
Width	O	O	O	O	O
Wrap	O	O	O	X	X
ZeroToReplaceChar	X	X	X	X	X

2. Fx(Formatted) 조회 모드의 허용 데이터 포맷

Dosearch 또는 LoadSearchData 메소드의 Fx 옵션 사용시 타입별 허용하는 포맷 값은 아래의 표와 같다.

컬럼타입	포맷	허용값	사용가능예	사용불가능예	비고
Text	일반	문자열			
	사용자 포맷	포맷을제외한 문자열	7912121022345	791212- 1022345	
	날짜	Date 타입 참고			
	날짜	Date 타입 참고			
Popup	일반	문자열			
Pass		문자열			
Date	날짜	날짜포맷 문자열	2012-02-16	20120216	년월일 구분자는 설정포맷과 틀려도됨
	시간	날짜포맷 문자열	13:15:16	131516	
	날짜+시간	날짜, 시간포맷 문자열	2012-02-16 13:15:16	2012021613151 6	날짜와 시간 사이에는 스페이스로 구분
Int		포맷을 제외한 숫자값	1234567	1,234,567	
Float		포맷을 제외한 숫자값	123456.789	123,456.789	
AutoSum	Int	포맷을 제외한 숫자값	1234567	1,234,567	
	Float	포맷을 제외한 숫자값	123456.789	123,456.789	
AutoAvg	Int	포맷을 제외한 숫자값	1234567	1,234,567	

	Float	포맷을 제외한 숫자값	123456.789	123,456.789	
CheckBox		0 또는 1		0, 1 을 제외한 문자열	
DelCheck		0 또는 1		0, 1 을 제외한 문자열	
DummyCheck		0 또는 1		0, 1 을 제외한 문자열	
Radio		0 또는 1		0, 1 을 제외한 문자열	
Combo		설정 ComboCode 값			InitCombo NoMatchText 사용불가
ComboEdit		설정 ComboCode 값			InitCombo NoMatchText 사용불가
Image		Image Url			
Seq		공백		문자열	
Status		I, D		공백, I, D 를 제외한 문자열	

Fx:2 가 7.0.5.4 버전 부터 추가되었다.

이 옵션을 사용하는 경우 아래와 같은 여러가지 기능의 제한이 생기는 대신 조회 속도를 크게 향상시킨다.

1. 조회 모드

smClientPaging, smLazyLoad 에서만 사용이 가능하다.

2. 데이터 포맷

XML 형식의 데이터는 지원하지 않으며 JSON형태만 지원한다. 그리고 JSON 형식에서 UseJsonAttribute를 통해 행단위,셀단위 서식은 지원하지 않는다.

3. 머지 유형

데이터 영역의 머지는 허용되지 않으며 msNone나msHeaderOnly 형식의 머지

만 사용할 수 있다.

4. 계산 기능

ShowSubSum()함수를 통한 소계/누계 기능은 지원하지 않고, InitColumns()함수의 Calculogin(열간 계산식) 기능도 지원하지 않는다.

5. 시트 구조

트리형식이나 다중라인 레코드 구조의 시트는 지원하지 않는다.

6. LoadSearchData의 Option 파라미터

Wait와 Sync를 제외한 다른 기능은 지원하지 않는다.

7. 이벤트

조회 후 발생하는 OnSearchEnd 이벤트만 지원하며, OnLoadData, OnRownSearchEnd, OnDecryption 과 같은 이벤트는 발생하지 않는다.

8. 그 외 옵션

DataAutoTrim,FrozenRows,DefaultValue 설정이 지원되지 않는다.

그외 각 컬럼의 Type 별 조회 허용 유형

아래 타입/포맷의 경우 조회 과정에서 서버로부터 넘어온 데이터를 무시하고 시트에 표시하지 않을 뿐 조회 된 이후에는 동일하게 동작한다.

컬럼 타입	세부 내용	불허 형식
Text,Popup,PopupEdit		Date 포맷사용 불가
CheckBox,DummyCheck	0/1 값만 사용가능	TrueValue/FalseValue값 무시
Radio,DelCheck		조회 데이터 표시 안함
Combo	ComboCode값 매핑	InitComboNoMatchText()
ComboEdit	ComboText값 매핑	설정 무시
Int,Float,AutoSum,AutoAvg	콤마가 제거된 숫자형식만 사용가능	
Date	Ymd,Ym,Md만 사용 가능하며, 조회시 구분자를 포함하여 데이터를 조회해야 함. ex) 1987-12-26	Hms,Hm,YmdHms 형식 사용 불가
Status,Result		조회 데이터 표시 안함



3. Theme 만들기

ibsheet는 기본적으로 ibsheet.js 파일이 위치한 폴더에 "Main"폴더 안에 있는 이미지 파일과 ibsheet.css 파일을 이용하여 디자인을 구성한다.

만약 다른 디자인을 만들거나 기존의 디자인을 수정하고 자 하는 경우에는 다음과 같이 수정한다.

1. Main 폴더를 하나 다른 이름으로 복사한다. (여기서는 "DeepBlue"라는 폴더로 복사했다고 가정하자)
2. 폴더안에 각종 버튼 이미지를 개발하는 프로젝트 테마에 맞게 변경한다.
3. 시트의 색상은 ibsheet.css 파일을 참고하게 되는데, 이 파일을 열어보면 모든 css class 명이 .GM으로 시작하는 걸 확인할 수 있다. 이 이름을 다른 이름으로 교체하여야만 사용이 가능하다. (여기서는 .GM을 .DB로 변경했다고 가정하자)
4. 끝으로 시트 초기화 구문에서 다음과 같이 테마를 적용할 수 있다.
`mySheet.SetTheme("DB","DeepBlue");`



4. 엑셀 다운시 머지 적용 처리

	DownCols	
	화면과 동일하게 컬럼 설정	화면과 다르게 컬럼 설정
DownRows 사용	X	X
DownRows 사용 안함	○	△

X : 머지 반영 안함

△ : 머지 반영 되지만 화면과 동일하지 않을 수 있음

○ : 화면과 동일하게 머지 반영

1. 머지 반영 안하는 경우

- DownRows 를 사용하면 Merge:1 로 설정해도 머지 반영 안함

2. 화면과 동일하게 머지되는 경우

- 화면에 보이는 컬럼 그대로 다운받는 경우

3. 머지 반영 되지만 화면과 동일하지 않을 수 있는 경우

- 화면에 보이는 컬럼과 다운 받는 컬럼이 다른 경우



5. 다중 검색어 및 범위 필터 기능 설정 및 사용법

OR, AND, From~To(이하 Range) 연산자에 대한 키워드를 포함한 문자열 구성한다.
각 연산자에 대한 키워드는 아래와 같다.

연산자	키워드
OR	or 또는
AND	and, & 또는 공백
Range	~

1. 설정 가이드

- 다중 필터를 사용하는 경우 입력 문자열이 길어 질 수 있기 때문에 입력한 검색어를 보기 어려울 수 있다. 이런 경우 전역속성 FilterInputPopup 을 1로 설정하면 필터 입력시 팝업 다이얼로그를 이용할 수 있다. (ibsheet.cfg 에서 설정)
- 팝업 다이얼로그 사용시 입력 창 하단에 사용 도움말을 표기 할 수 있다.
ibmsg Text 태그의 SYS_FilterInfoMessage 속성값으로 도움말을 정의 할수 있으며 해당 속성에 정의가 없으면 도움말은 표시되지 않는다.
- ibmsg 작성방법
 - i. 라인은 태그를 이용하여 구분한다.
 - ii. ibmsg 파일은 xml 포맷이기 때문에 "<", ">" 등과 같은 특수문자 표기는 다음의 url을 참고하여 표기 한다.
(http://www.w3.org/MarkUp/HTMLPlus/htmlplus_13.html)
 - iii. 작성예시
SYS_FilterInfoMessage = "검색어와 연산자는 공백으로 구분
AND 연산자 : and, 공백,
ex) 대한 and
민국OR 연산자 : or, |
ex) 대한 or 민국
범위 연산자 : ~
ex) 10 ~
20검색어에 연산자 또는 공백이 포함되는 경우
"" 사용
ex) "대한 민국", "You
and I""

2. 사용 가이드

- 검색어와 키워드 사이는 공백으로 분리 한다.
- 검색어에 공백 또는 연산자 키워드가 포함된 경우 " "를 사용한다.
예) "You and I ", "Dead or Alive", "대한 민국"
- 여러개의 연산자 키워드를 사용하는 경우 기본적으로 좌측부터 우측으로 처리 되고, 서로 다른 연산자는 Range > AND > OR 의 우선순위를 따른다.
 - i. 1 or 5 ~ 10 : 5 와 10 사이의 값 또는 1
 - ii. 1 ~ 5 or 10 : 1 과 5 사이의 값 또는 10
 - iii. 가 and 나 or 다 : "가", "나" 문자를 포함한 문자열 또는 "다" 문자를 포함한 문자열
- 날짜 타입에 대한 검색어 입력시 문자열 필터 옵션을 사용하는 경우 날짜 포맷을 직접 입력하여야 하고, 숫자 필터 옵션을 사용하는 경우에는 yyyyMMdd 형식의 포맷 구분자 없이 입력 할 수 있다.
예) 문자 필터 옵션 사용 : 2015/04
숫자 필터 옵션 사용 :
20150401 ~ 20150430 또는 2015/04/01 ~ 2015/04/30
- Range 필터를 사용하는 경우에는 문자 필터 옵션을 사용할 수 없다.
(문자 필터 옵션 사용시 "같음" 필터 옵션으로 변경 적용)

구분	필터 옵션
숫자 필터 옵션	같음
	같지 않음
	작음
	같거나 작음
	큼
	같거나 큼
문자열 필터 옵션	단어로 시작함
	단어로 시작하지 않음
	단어로 끝남
	단어로 끝나지 않음
	포함함
	포함하지 않음



6. 아이비시트에서 자주 사용되는 메서드 정리

1. Data Read/Write

Return값 유형	Method 명	내용
String	CellValue	셀 데이터 Read/Write
String	CellText	셀 데이터 Read/Write
JSON Object	RowData	행 데이터 Read/Write
Query String	RowSaveStr	행 데이터 Read
	LoadSearchData	조회 데이터 XML/JSON Write
	LoadSaveData	저장 데이터 XML/JSON Write
String	GetEditText	편집중인 셀의 데이터 Read
String	SumValue	합계행 안에 셀 데이터 Read/Write
JSON Object	GetSaveJson	시트 전체 데이터 Read
Query String	GetSaveString	시트 전체 데이터 Read
Object	GetSheetHtml	시트 전체 데이터를 HTML양식으로 Read

2. 최초 조회된 값 확인/변경

Return값 유형	Method 명	내용
	ReturnData	행을 최초 조회했던 값으로 변경
	ReturnCellData	셀을 최초 조회했던 값으로 변경
String	CellSearchValue	셀에 대한 조회 데이터 확인

3. 행/열 인덱스

Return값 유형	Method 명	내용
Int	LastRow	마지막 행의 Index
Int	LastCol	마지막 열의 인덱스
Int	HeaderRows	첫번째 데이터 행의 인덱스
Int	SearchRows	조회한 데이터 행수 (상태가 R인 데이터 행 수)
Int	TopRow	현재 화면에 보여지는 행중 최 상단의 행 인덱스
Int	LeftCol	현재 화면에 보여지는 열중에 최 좌측의 열 인덱스

Int	SelectRow	현재 선택된(포커스 된) 행의 인덱스
Int	SelectCol	현재 선택된(포커스 된) 열의 인덱스
	SelectCell	특정 셀로 포커스 이동
Int or String	FindSumRow	합계행의 인덱스

4. 찾기 기능

Return값 유형	Method 명	내용
Int	FindText	열에서 특정 String문자 찾기
String	FindSubSumRow	여러개의 소계행 인덱스 찾기
String	FindCheckedRow	체크박스 컬럼에 체크된 행 찾기
String	FindStatusRow	특정 상태를 갖는 모든 행 찾기
String	GetSelectionRows	마우스로 드레그 하여 선택한 영역 행 인덱스
String	GetSelectionCols	마우스로 드레그 하여 선택한 영역 열 인덱스

5. Tree 기능 관련 함수

Return값 유형	Method 명	내용
Int	RowLevel	노드의 Level(depth) Read/Write
Boolean	IsHaveChild	자식노드 존재 여부 확인
Int	GetPrevSiblingRow	자신과 같은 Level의 상위 행 인덱스
Int	GetNextSiblingRow	자신과 같은 Level의 하위 행 인덱스
Int	GetParentRow	부모 행 인덱스
String	GetChildRows	자식 행 전체 인덱스
Int	GetLastChildRow	마지막 자식 행 인덱스
Int	GetChildNodeCount	자식행 개수
	ShowTreeLevel	특정 Level까지 접거나 펼침

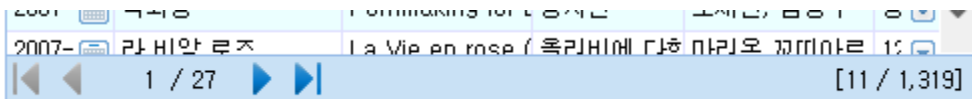
6. File import/export

Return값 유형	Method 명	내용
	Down2Excel	시트의 데이터를 엑셀파일로 export
	Down2Text	시트의 데이터를 text 파일로 export
	Down2Pdf	시트의 데이터를 PDF파일로 export
	DirectDown2Excel	시트의 헤더정보와 디비의 데이터 정보를 합



		쳐 엑셀파일로 export
	LoadExcel	엑셀 파일을 시트위로 import
	LoadText	text 파일을 시트위로 import
	DirectLoadExcel	엑셀 파일을 디비로 바로 저장
	Down2ExcelBuffer	두개 이상의 시트를 하나의 엑셀파일로 export
	SetDown2ExcelConfig	엑셀 export시 공통 설정
	OnDownFinish	다운로드 완료 후 이벤트
	OnBeforeDownload	서버로 데이터 전달 직전 이벤트
	OnLoadExcel	엑셀 파일 업로드 이후 이벤트

7. 건수정보 표시 줄 관련



Return값 유형	Method 명	내용
	CountPosition	건수표시 위치 설정
	CountFormat	건수표시 포맷 설정
	CountInfoElement	건수표시를 외부 객체에 설정/확인
	SelectionSummaryInfoElement	Summary정보를 외부 객체에 설정/확인
	SelectionSummary(in SetConfig)	드래그 영역에 대한 Summary정보 표시
	PagingPosition	페이지 네비게이션 버튼 표시

8. Tab,Enter 입력시 동작 결정

Return값 유형	Method 명	내용
	TabBehavior	포커스 상태에서 Tab키 다운시 동작방식 결정
	EnterBehavior	포커스 상태에서 Enter키 다운시 동작방식 결정
	EditTabBehavior	편집 중 Tab키 다운시 동작방식 결정
	EditEnterBehavior	편집 중 Enter키 다운시 동작방식 결정



	OnTab	Tab키를 통해 이동시 발생하는 이벤트
--	-------	-----------------------

9. 기타 자주 사용되는 메서드

RemoveAll 조회 데이터 전체 삭제

Reset 시트 객체 초기화

RowHidden 행 보이기/감추기

ColHidden 열 보이기/감추기

IsDataModified 시트의 수정 여부 확인

ColSaveName 열의 인덱스를 기준으로 SaveName을 확인

SaveNameCol 열의 SaveName을 기준으로 인덱스를 확인