
IB Sheet 8

서버 모듈

개발자가이드

IBSheet8 서버모듈 개발자 가이드

초판발행/ 2023년 12월 18일

지은이/ 전인수

펴낸이/ 정광천

펴낸 곳/



08394 서울시 구로구 디지털로 242(구로동) 한화비즈메트로 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. 서버모듈 개요	7
1. 서버모듈 개요	9
1.1 서버모듈이란?	9
1.2 서버모듈 특징	9
Chapter 2. 서버모듈 구조	11
2. 서버 모듈 구조	12
2.1 서버 모듈 흐름	12
2.1.1 일반 Download	12
2.1.2 일반 load	13
2.1.3 DirectDownload	14
2.1.4 DirectLoad	15
Chapter 3. 서버모듈 설치	17
3. 서버모듈의 설치 및 환경 설정	19
3.1 서버모듈 사양	19
3.2 서버모듈의 설치	19
3.2.1 서버에 jar 파일 로드	19
3.2.2 서버에 jsp 파일 로드	21
3.2.3 jsp 파일 경로 설정	22
3.2.4 jsp 파일 환경 설정	23
Chapter 4. API 레퍼런스	25
4. API 레퍼런스	26
4.1 API 사용	26
4.2 IBSheetDown	26
IBSheetDown()	26

IBSheetDown(HttpServletRequest, HttpServletResponse)	26
IBSheetDown(HttpServletRequest, HttpServletResponse, Int)	27
appendDataArr(data)	27
appendDataList(data)	27
appendDataMap(data)	27
appendMultiData(data)	27
close()	28
downToBrowser()	28
downToStream(OutputStream)	28
getExtendParam(key)	28
getFileName()	28
saveToFile(path)	28
setDirectRequestData()	29
setDownFinish()	29
setEncoding(encoding)	29
setFileHeader()	29
setFontName(fontName)	29
setFontSize(size)	30
setHeaderBackColor(color)	30
setHeaderFontBold(bold)	30
setHeaderFontColor(color)	30
setHeaderFontName(name)	31
setHeaderFontSize(fontSize)	31
setWordWrap(isWordWrap)	31
setMarkupTagDeilmiter(st1, st2, ct1, ct2)	31
setDelimMode(mode)	32
setFileType(fileType, ignoreFileExtention)	32
setNumberExMode(numberExMode)	33
setXlsxTempFolderMode(useXssfMode)	33
setImageProcessType(typ)	33
setUseGroupTree(useGroupTree)	34
setExcelRowHeight(excelRowHeight)	34

setExcelHeaderRowHeight(excelHeaderRowHeight)	34
setService(HttpServletRequest, HttpServletResponse)	34
setTempRoot(tempRoot)	35
setTextLine(textLine)	35
setTextSplitter(textSplitter)	35
setTreeChar(treeChar)	35
setWebServerDomain(webServerDomain)	36
4.3 IBSheetLoad	36
IBSheetLoad()	36
IBSheetLoad(HttpServletRequest, HttpServletResponse)	36
IBSheetLoad(HttpServletRequest, HttpServletResponse, String)	36
close()	36
directLoadExcel()	36
getExtendParam(key)	37
getUploadFile()	37
getUploadFileName()	37
sendDirectTo(url)	37
sendDirectToFP()	38
setEncoding(encoding)	38
setMarkupTagDeilimiter(startTag1, startTag2, closingTag1, closingTag2)	38
setDelimMode(mode)	38
setStrictHeaderMatch(strictHeaderMatch)	39
setMaxRows(maxRows)	39
setMaxColumns(maxColumns)	39
setSkipEmptyRow(skipEmptyrow)	40
setMaxFileSize(maxFileSize)	40
setService(HttpRequest, HttpResponse)	40
setTextSplitter(textSplitter)	40
writeToBrowser()	40
writeToStream(OutputStream)	40
4.4 PrintPdf	41
Print2Pdf()	41

Print2Pdf(HttpServletRequest, HttpServletResponse)	41
print()	41
saveTo(path)	41
setDPI(dpi)	41
setFontFolder(path)	42
setPageEncoding(encoding)	42
setRequest(HttpServletRequest, HttpServletResponse)	42
4.5 클라이언트 API	42
4.6 기타 API 사용	44
4.7.1 DRM	44
4.7.2 Direct Export	46
4.7.3 Direct import	49
4.7.4 header 및 footer 기능	52

Chapter 1. 서버모듈 개요

1. 서버모듈 개요

1.1 서버모듈이란?

- IB Sheet 8에서 사용할 수 있는 파일 다운로드 및 업로드 기능을 제공하는 라이브러리이다.

1.2 서버모듈 특징

➤ Excel Import/Export

IBSheet위에 조회 되거나 작성된 데이터를 엑셀 파일에 다운로드 할 수 있으며, 엑셀 파일에 작성한 데이터를 IBSheet 위에 로딩하여 표현할 수 있다.

또한 DB에서 조회된 데이터를 엑셀 파일에 바로 다운로드 가능하며, 엑셀 파일을 서버로 전달하여 분석한 데이터를 DB로 바로 적용할 수 있는 기능을 제공한다.

➤ Text Import/Export

IBSheet위에 조회되거나 작성된 데이터를 txt형태로 다운로드 할 수 있으며, txt 파일의 내용을 IBSheet 위에 로딩하여 표현할 수 있다.

➤ Pdf Export

IBSheet위에 조회되거나 작성된 데이터를 pdf형태로 다운로드 할 수 있으며, Font를 설정하여 Font지정이 가능하다.

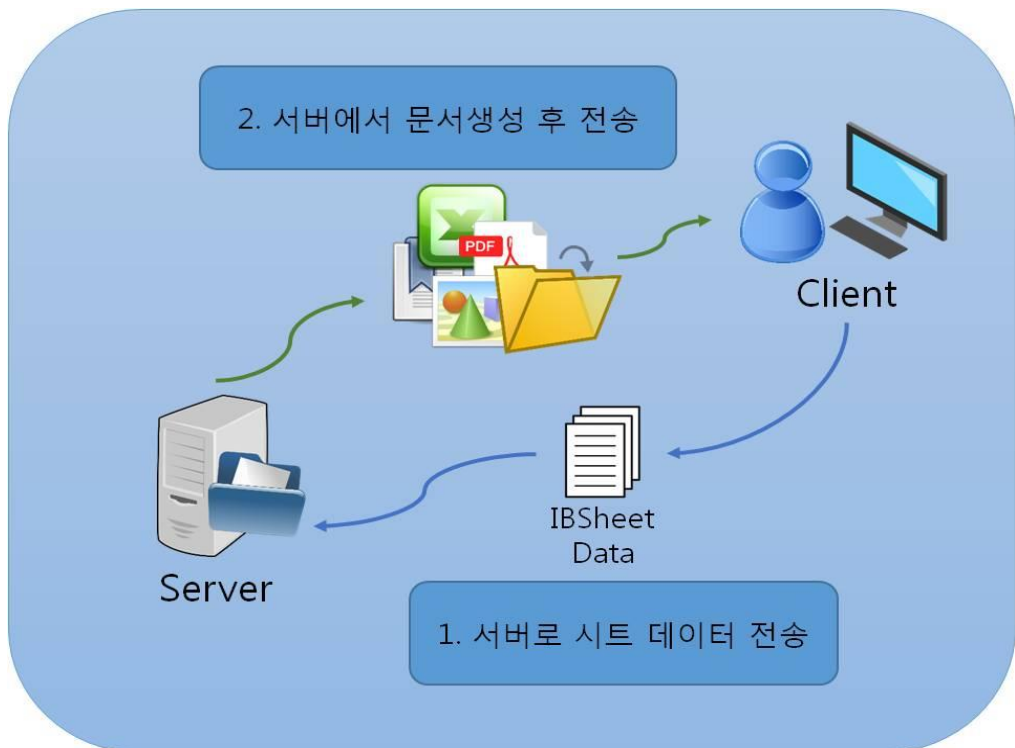
구분	타입	파일 형식	비고
export	엑셀	xls, xlsx	
	텍스트	txt, csv	
	PDF	pdf	
import	엑셀	xls, xlsx	
	텍스트	txt, csv	

Chapter 2. 서버모듈 구조

2. 서버 모듈 구조

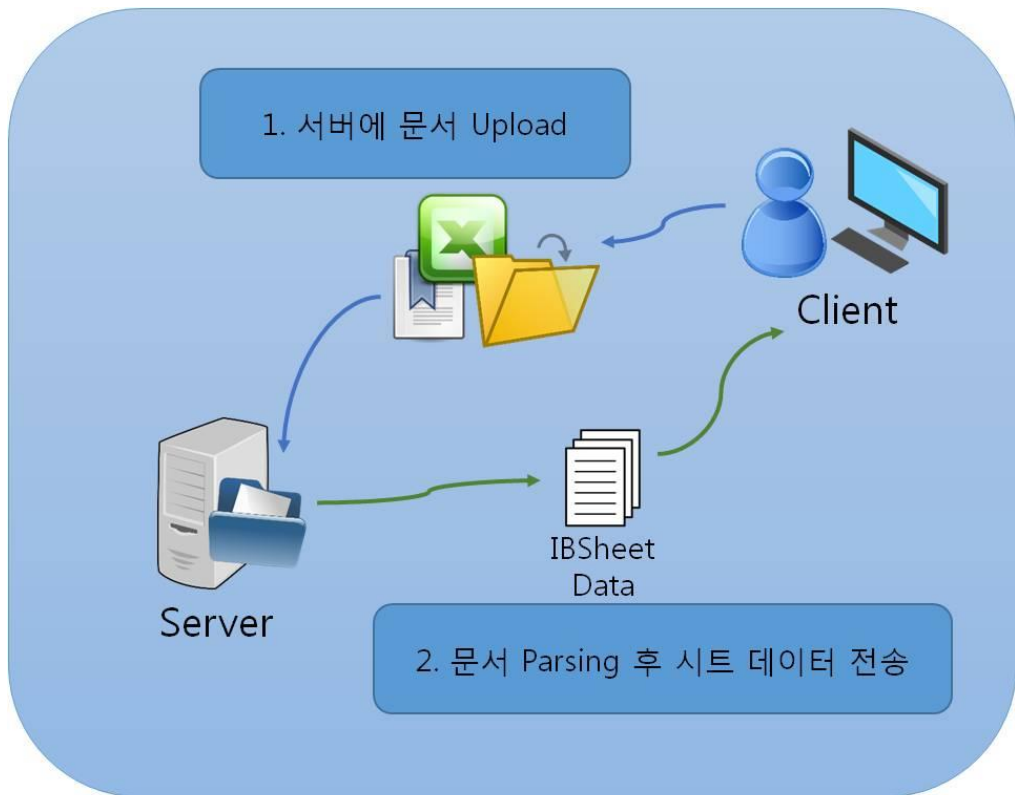
2.1 서버 모듈 흐름

2.1.1 일반 Download



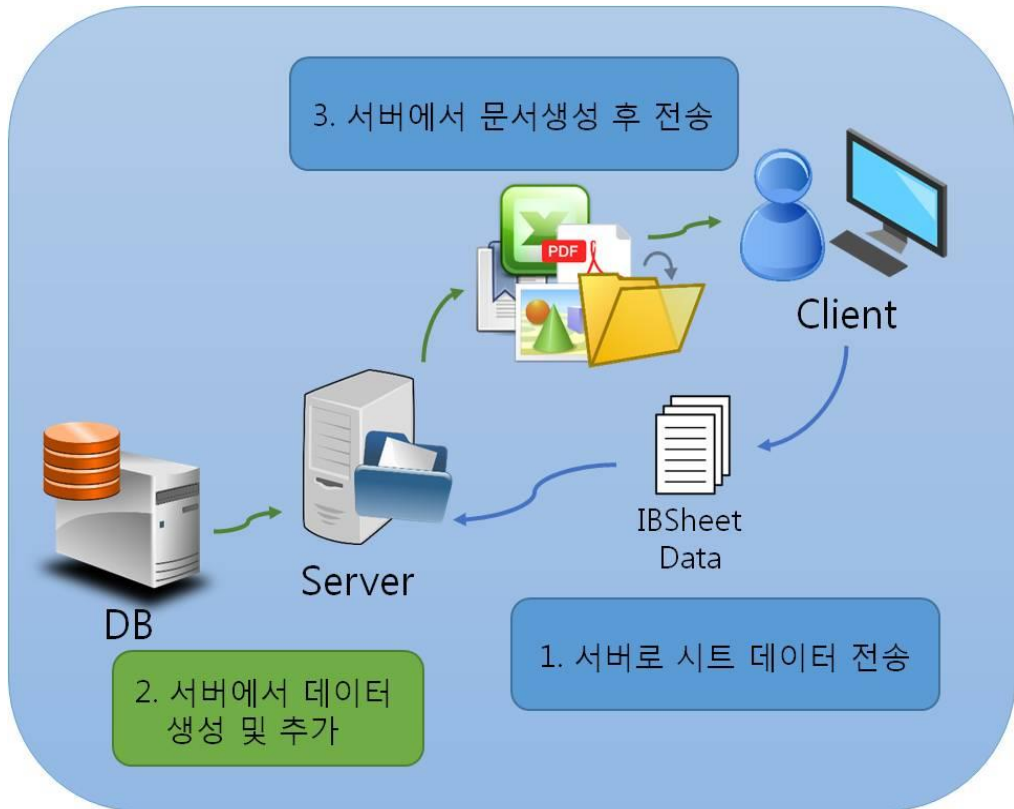
IBSheet-Server Export Module

2.1.2 일반 load



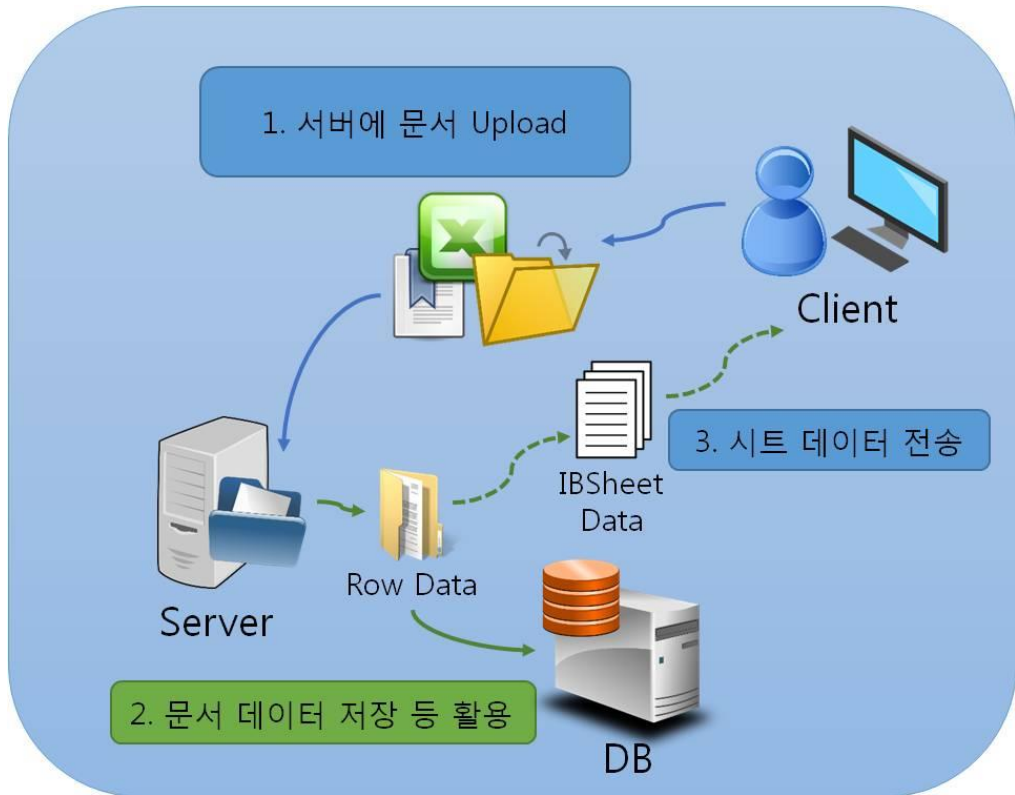
IBSheet-Server Import Module

2.1.3 DirectDownload



IBSheet-Server Direct Export Module

2.1.4 DirectLoad



IBSheet-Server Direct Import Module

Chapter 3. 서버모듈 설치

3. 서버모듈의 설치 및 환경 설정

3.1 서버모듈 사양

최소 사양	권장 사양
JDK 버전 1.8	JDK 버전 1.8
Dynamic Web Module 3.1	Dynamic Web Module 3.1

3.2 서버모듈의 설치

3.2.1 서버에 jar 파일 로드

서버모듈을 이용한 환경을 구축하기 위해서 제공받은 모든 jar 파일들을 서버 라이브러리 폴더(wlib)에 복사하여 옮겨 놓는다. lib 폴더 안에 파일을 붙여넣기 한 경우 WAS 서버를 재부팅해야 정상으로 적용된다.

배포되는 파일 중 ibsheet8-x.x.x.jar 이외에 이미 서버에서 필요 버전이상의 같은 라이브러리를 사용한다면 설치하지 않아도 문제되지는 않는다.

예)

```
₩Tomcat8.5₩lib₩commons-codec-1.6.jar
₩Tomcat8.5₩lib₩commons-logging-1.1.3.jar
₩Tomcat8.5₩lib₩batik-all.xml.jar
₩Tomcat8.5₩lib₩poi-3.13-all.jar
₩Tomcat8.5₩lib₩ibsheet7-2.0.0.jar
```

- 제공되는 라이브러리 파일은 아래와 같다.

<POI3 버전 사용시>

배포 파일	필수 여부	비고
lbsheet8-x.x.x.jar	○	
batik-all-xml.jar (batik 1.7)	○	batik 1.6 이상
commons-codec-1.6.jar	○	1.5 이상
commons-logging-1.1.3.jar	○	1.1.3 이상
ib-itext.jar	X	
poi-3.13-20191017.jar	○	
poi-ooxml-3.13-20191017.jar	○	
poi-ooxml-schemas-3.13-20191017.jar	○	
servlet-api.jar	○	
xmlbeans-2.3.0.jar	○	poi 3.10이상

<POI4 버전 사용시>

배포 파일	필수 여부	비고
lbsheet8-x.x.x.jar	○	
batik-all-xml.jar	○	batik 1.6 이상
commons-codec-1.13.jar	○	
commons-collections4-4.4.jar	○	
commons-compress-1.19.jar	○	
commons-logging-1.1.3.jar	○	
commons-math3-3.6.1.jar	○	
curvesapi-1.06.jar	○	
ib-itext.jar	X	
poi-4.1.2.jar	○	
poi-ooxml-4.1.2.jar	○	
poi-ooxml-schemas-4.1.2.jar	○	
servlet-api.jar	○	
SparseBitSet-1.2.jar	○	
xmlbeans-3.1.0.jar	○	

3.2.2 서버에 jsp 파일 로드

서버모듈의 기능 사용을 위해서는 해당 기능에 맞는 jsp 파일이 서버에 존재해야 한다. 제공받은 jsp 파일들을 웹 서버의 특정 폴더에 복사하여 설치한다.

파일명	내용/역할
Down2Excel.jsp	엑셀다운로드 전용인 Down2Excel 메소드를 적용하기 위한 서버 쪽 응용 프로그램 파일
LoadExcel.jsp	엑셀로딩 전용인 LoadExcel 메소드를 적용하기 위한 서버 쪽 응용 프로그램 파일
Down2Text.jsp	IBSheet 데이터를 txt 형태 파일로 내리게 하기 위한 jsp 파일
LoadText.jsp	txt 파일의 내용을 IBSheet 위에 올리기를 위한 jsp 파일
DirectDown2Excel.jsp	DB에서 조회한 데이터를 바로 엑셀 파일로 내리게 하기 위한 jsp 파일
DirectLoadExcel.jsp	IBSheet에서 조회한 데이터를 DB에 바로 적용할 수 있도록 하기 위한 jsp 파일
Down2Pdf.jsp	IBSheet 데이터를 pdf로 변환하여 다운로드 처리를 담당하는 jsp 파일

3.2.3 jsp 파일 경로 설정

시트를 생성할 때 설정하는 Cfg 프로퍼티의 Export 속성을 통해 jsp 파일이 위치한 경로를 설정할 수 있다.

```
// 서버모듈 jsp 파일들의 공통 경로를 설정하는 경우
Options.Cfg = {
    "Export": {
        "Url": "/assets/ibsheet/jsp"
    }
}

// jsp별로 개별 경로를 설정하는 경우
Options.Cfg = {
    "Export": {
        "Down2ExcelUrl": "/assets/ibsheet/jsp/Down2Excel.jsp",
        "LoadExcelUrl": "/assets/ibsheet/jsp/LoadExcel.jsp",
        ...
    }
}
```

서버모듈에 필요한 ibsheet.cfg 속성은 아래와 같다.

속성 명	내용/역할
Url	서버모듈 jsp 파일의 경로 (해당 Url 위의 jsp 파일 호출)
Down2ExcelUrl	엑셀 다운로드 처리를 담당하는 jsp 파일의 경로
LoadExcelUrl	엑셀 업로드 처리를 담당하는 jsp 파일의 경로
Down2TextUrl	텍스트 다운로드 처리를 담당하는 jsp 파일의 경로
LoadTextUrl	텍스트 업로드 처리를 담당하는 jsp 파일의 경로
Down2PdfUrl	PDF 다운로드 처리를 담당하는 jsp 파일의 경로
DirectDown2ExcelUrl	(DB 조회한)데이터를 엑셀 다운로드 처리를 담당하는 jsp 파일의 경로
DirectLoadExcelUrl	(서버로 직접) 엑셀 업로드 처리를 담당하는 jsp 파일의 경로

3.2.4 jsp 파일 환경 설정

- 인코딩 설정

사용하는 서버에 따라 인코딩 설정이 다르기 때문에 각 jsp 파일에 해당 서버에 맞는 인코딩 설정을 한다.

참고1. Down2Excel.jsp

```
IBSheetDown down = null;  
down = new IBSheetDown(request, response);  
down.setEncoding("UTF-8");
```

- PDF 폰트 경로 설정

PDF파일 다운로드를 위해서는 서버에 한글폰트 지원을 위한 ttf 파일이 필요하다.
이 폰트 폴더에 대한 경로를 jsp에서 설정해야 한다.

참고1. Down2Pdf.jsp

```
Print2Pdf ibPdf = new Print2Pdf(request, response);  
ibPdf.setFontFolder("D:/down");
```


Chapter 4. API 레퍼런스

4. API 레퍼런스

4.1 API 사용

서버모듈에서 제공하는 API 사용을 하기 위해서는 생성자를 필수로 호출하여 객체를 생성해야 합니다.

예)

```
// request, response를 포함한 IBSheetDown 생성자 호출
IBSheetDown down = new IBSheetDown(request, response);

//IBSheetDown 객체의 인코딩 설정
down.setEncoding("UTF-8");

//IBSheetDown 객체의 다운로드 파일명을 반환
down.getFileName();
```

4.2 IBSheetDown

XLSX, XLS, CSV, TEXT 파일 다운로드

IBSheetDown()

생성자, request와 response 할당이 없는 경우 setService로 등록해야 정상 동작한다.

IBSheetDown(HttpServletRequest, HttpServletResponse)

request와 response를 포함한 생성자

IBSheetDown(HttpServletRequest, HttpServletResponse, Int)

DelimMode 1을 사용할 경우의 생성자, 시트에 포함될 문자열중 STX(Wu0002), ETX(Wu0003 이 포함된 경우에만 1로 설정

appendDataArr(data)

시트에 String[] 형태의 데이터를 한 줄 덧붙인다. (DirectDownExcel 전용)

Return	없음		
Parameter	Type	필수여부	설 명
data	String[]	필수	엑셀에 추가할 시트 데이터

appendDataList(data)

시트에 List 형태의 데이터를 한 줄 덧붙인다. (DirectDown2Excel 전용)

Return	없음		
Parameter	Type	필수여부	설 명
data	List<String>	필수	엑셀에 추가할 시트 데이터

appendDataMap(data)

시트에 Map 형태의 데이터를 한 줄 덧붙인다. (DirectDown2Excel 전용)

Return	없음		
Parameter	Type	필수여부	설 명
data	Map<String, String>	필수	엑셀에 추가할 시트 데이터

appendMultiData(data)

시트에 데이터를 여러 줄 덧붙인다. (DirectDown2Excel 전용)

Return	없음		
--------	----	--	--

Parameter	Type	필수여부	설 명
data	List<Object>	필수	엑셀에 추가할 시트 데이터

close()

다운로드를 종료하고 메모리를 해제한다.

downToBrowser()

생성된 문서를 브라우저를 통해 다운로드 한다.

downToStream(OutputStream)

생성된 문서를 OutputStream을 통해 전송한다.

getExtendParam(key)

시트에서 전달한 ExtendParam 정보를 받는다.

Return	String, key에 해당하는 값		
Parameter	Type	필수여부	설 명
key	String	필수	ExtendParam에 설정한 Key값

getFileName()

생성된 파일명을 반환한다.

Return	String, 파일명		
Parameter	Type	필수여부	설 명
없음			

saveToFile(path)

생성된 문서를 서버의 특정 폴더에 저장한다.

Return	없음		
Parameter	Type	필수여부	설 명
path	String	필수	저장할 서버 폴더 명

setDirectRequestData()

HttpRequest에 SHEETDATA라는 Attribute가 저장되어 있는 경우, 해당 데이터를 다운로드 한다. (DirectDown2Excel 전용)

setDownFinish()

다운로드 종료 정보를 시트에 전달하고, 화면의 대기 이미지를 닫기 위한 FileDownloadToken 쿠키를 생성한다.

setEncoding(encoding)

파일을 다운로드받는 Html 페이지의 인코딩에 맞춰 인코딩을 설정한다.

Return	없음		
Parameter	Type	필수여부	설 명
encoding	String	필수	페이지 인코딩 타입 UTF-8, EUC-KR 등

setFileHeader()

다운로드 할 파일의 정보를 브라우저 헤더로 전송한다.

setFontName(fontName)

문서에 사용될 글꼴을 설정한다(HeaderFontName도 같이 설정된다.).

Return	없음		
Parameter	Type	필수여부	설 명
fontName	String	필수	엑셀 파일에 설정할 글꼴

setFontSize(size)

문서에 사용될 글자 크기를 설정한다.(HeaderFontSize도 같이 설정된다.)

Return	없음		
Parameter	Type	필수여부	설 명
size	Int	필수	엑셀 파일에 설정할 글자 크기

setHeaderBackColor(color)

헤더 부분의 배경색을 설정한다.

Return	없음		
Parameter	Type	필수여부	설 명
color	String	필수	엑셀 파일의 헤더 배경색

setHeaderFontBold(bold)

헤더 부분 글자의 볼드 처리 여부를 설정한다.

Return	없음		
Parameter	Type	필수여부	설 명
bold	boolean	필수	엑셀 파일의 헤더 글자 볼드 처리 여부 (default: false)

setHeaderFontColor(color)

헤더 부분의 글자색을 설정한다.

Return	없음		
Parameter	Type	필수여부	설 명
color	String	필수	엑셀 파일의 헤더 글자 색상

setHeaderFontName(name)

헤더 부분에 사용될 글꼴을 설정한다.

Return	없음		
Parameter	Type	필수여부	설 명
name	String	필수	엑셀 파일의 헤더 글꼴

setHeaderFontSize(fontSize)

헤더 부분에 사용될 글자 크기를 설정한다.

Return	없음		
Parameter	Type	필수여부	설 명
fontSize	Int	필수	엑셀 파일의 헤더 글자 크기

setWordWrap(isWordWrap)

문서에 줄바꿈 적용 여부를 설정한다.

Return	없음		
Parameter	Type	필수여부	설 명
isWordWrap	boolean	필수	엑셀 파일의 줄바꿈 허용 여부

setMarkupTagDeilmiter(st1, st2, ct1, ct2)

기본 마크업 태그 외에 별도의 태그를 사용할 때 설정한다.

IBSheet8 환경설정(ibsheet.cfg)의 MarkupTagDelimiter 설정 값과 동일해야 한다.

Return	없음		
Parameter	Type	필수여부	설 명
startTag1	String	필수	Start Tag 시작 문자열 (default: "<")
startTag2	String	필수	Start Tag 완료 문자열 (default: ">")
closingTag1	String	필수	Closing Tag 시작 문자열(default: "</")

closingTag2	String	필수	Closing Tag 완료 문자열(default: ">")
-------------	--------	----	----------------------------------

setDelimMode(mode)

시트 문자열에 STX(₩u0002), ETX(₩u0003)이 포함된 경우 설정한다.

Return	없음		
Parameter	Type	필수여부	설 명
mode	boolean	필수	0: 시트 구분자로 STX, ETX 문자를 사용한다. (default) 1: 시트 구분자로 변형된 문자열을 사용한다(시트에 설정이 되어 있어야 한다.).

setFileType(fileType, ignoreFileExtention)

다운받을 문서의 타입을 설정한다.

Return	없음		
Parameter	Type	필수여부	설 명
fileType	String	필수	xls: xls 형식으로 다운로드 xlsx: xlsx 형식으로 다운로드 excel: FileName에 설정한 값으로 다운로드, 확장자가 없는 경우 기본 xls 형식으로 다운로드
ignoreFileExtention	boolean	선택	클라이언트 사이드에서 설정한 파일명의 확장자를 제거하고 서버 사이드에서 설정한 확장자를 붙일지 여부 설정

setNumberExMode(numberExMode)

엑셀에 포함될 숫자 형식 데이터를 '통화'나 '사용자정의'가 아닌 '숫자' 서식으로 다운로드하고 싶을 때 사용한다. 클라이언트 사이드에서 down2Excel 옵션으로 설정한 numberExMode보다 우선적으로 적용된다.

Return	없음		
Parameter	Type	필수여부	설 명
numberExMode	boolean	필수	숫자 데이터를 '통화', '사용자정의'가 아닌 '숫자' 서식으로 다운받을지 여부 설정

setXlsxTempFolderMode(useXssfMode)

POI 라이브러리를 사용중이고 XLSX 형식의 파일을 다운로드할 때 임시폴더를 사용하지 않고 메모리에서 직접 다운받도록 설정한다.

메모리 사용량이 늘어나므로 임시폴더를 사용할 수 없는 환경에서만 설정하길 권장.

Return	없음		
Parameter	Type	필수여부	설 명
useXssfMode	boolean	필수	임시폴더 사용 여부 설정

setImageProcessType(typ)

엑셀 다운로드시 포함된 이미지의 비율을 맞추고 싶을 때 설정한다.

Return	없음		
Parameter	Type	필수여부	설 명
typ	Int	필수	0: 셀의 가로/세로에 꼭차게 이미지 처리 1: 셀의 중앙에 이미지를 원본 크기로 표시 2: 이미지의 원본 가로/세로 비율을

			유지하면서 셀에 맞춤
--	--	--	-------------

setUseGroupTree(useGroupTree)

트리 사용 시트에서 트리 구조를 엑셀 파일에 반영할지 여부를 설정한다.

Return	없음		
Parameter	Type	필수여부	설 명
useGroupTree	boolean	필수	트리구조 엑셀 반영 여부 설정

setExcelRowHeight(excelRowHeight)

엑셀 문서의 행 높이를 설정하고 싶은 경우에 설정한다.

클라이언트 사이드에서 설정하는 excelRowHeight보다 우선적으로 반영된다.

Return	없음		
Parameter	Type	필수여부	설 명
excelRowHeight	Int	필수	엑셀 행 높이 설정

setExcelHeaderRowHeight(excelHeaderRowHeight)

엑셀 문서의 헤더행 높이를 설정하고 싶은 경우에 설정한다.

클라이언트 사이드에서 설정하는 excelHeaderRowHeight보다 우선적으로 반영된다.

Return	없음		
Parameter	Type	필수여부	설 명
excelHeaderRowHeight	Int	필수	엑셀 헤더행 높이 설정

setService(HttpServletRequest, HttpServletResponse)

request와 response를 할당한다. (기본 생성자 사용시만 적용)

setTempRoot(tempRoot)

템플릿 파일이 저장된 경로를 지정한다. 지정하지 않으면 시트의 TempFile에 사용된 경로만을 이용한다.

Return	없음		
Parameter	Type	필수여부	설 명
tempRoot	String	필수	템플릿 파일의 경로

setTextLine(textLine)

텍스트 파일의 행 구분자를 설정한다. (시트의 RowDelim 설정이 우선 적용)

Return	없음		
Parameter	Type	필수여부	설 명
textLine	String	필수	텍스트 파일의 표시될 행의 레코드 구분자 (default : "\n")

setTextSplitter(textSplitter)

텍스트 파일의 열 구분자를 설정한다. (시트의 ColDelim 설정이 우선 적용)

Return	없음		
Parameter	Type	필수여부	설 명
textSplitter	String	필수	텍스트 파일의 표시될 열의 레코드 구분자 (default : "\t")

setTreeChar(treeChar)

트리 레벨 문자 설정을 한다.

Return	없음		
Parameter	Type	필수여부	설 명
treeChar	String	필수	트리 레벨 문자 설정

			(default : "Wu2026")
--	--	--	----------------------

setWebServerDomain(webServerDomain)

문서에 이미지가 포함된 경우 이미지의 Root 폴더/웹 서버 도메인 설정을 한다.
 설정하지 않으면 현재 서버 도메인을 사용한다.

Return	없음		
Parameter	Type	필수여부	설 명
webServerDomain	String	필수	이미지가 포함된 도메인 경로

4.3 IBSheetLoad

XLS, XLSX, CSV, TXT 파일 로드

IBSheetLoad()

생성자, request와 response 할당이 없으면 setService로 등록해야 한다.

IBSheetLoad(HttpServletRequest, HttpServletResponse)

request와 response를 포함한 생성자

IBSheetLoad(HttpServletRequest, HttpServletResponse, String)

파일이 업로드 될 임시 경로를 사용하는 경우에 설정한다.

임시폴더를 설정하지 않으면 웹 서버의 기본 tempDirectory를 사용한다.

close()

업로드를 종료하고 메모리를 해제한다.

directLoadExcel()

업로드 한 문서를 List<Map<String, String>> 형태로 반환한다.(DirectLoadExcel 전

용)

getExtendParam(key)

시트에서 전달한 ExtendParam 정보를 받는다.

Return	String, key에 해당하는 값		
Parameter	Type	필수여부	설 명
key	String	필수	ExtendParam에 설정한 Key값

getUploadFile()

업로드 한 파일 객체를 반환한다.

Return	Object, 파일객체		
Parameter	Type	필수여부	설 명
없음			

getUploadFileName()

업로드 한 파일의 파일명을 반환한다.

Return	String, 파일명		
Parameter	Type	필수여부	설 명
없음			

sendDirectTo(url)

업로드 한 문서를 Map 형태로 SHEETDATA라는 HttpRequest Attribute에 저장하고 특정 경로로 전달한다.

Return	없음		
Parameter	Type	필수여부	설 명

url	String	필수	데이터를 전달받을 페이지 경로
-----	--------	----	------------------

sendDirectToFP()

업로드 한 문서를 Map 형태로 SHEETDATA라는 HttpRequest Attribute에 저장하고 시트에서 전달받은 FP 경로로 전달한다.

setEncoding(encoding)

페이지의 인코딩을 설정한다.

Return	없음		
Parameter	Type	필수여부	설 명
encoding	String	필수	페이지 인코딩 타입 UTF-8, EUC-KR 등

setMarkupTagDeilimiter(startTag1, startTag2, closingTag1, closingTag2)

기본 마크업 태그 외에 별도의 태그를 사용할 때 설정한다.

IBSheet8 환경설정(ibsheet.cfg)의 MarkupTagDelimiter 설정 값과 동일해야 한다.

Return	없음		
Parameter	Type	필수여부	설 명
startTag1	String	필수	Start Tag 시작 문자열 (default: "<")
startTag2	String	필수	Start Tag 완료 문자열 (default: ">")
closingTag1	String	필수	Closing Tag 시작 문자열(default: "</")
closingTag2	String	필수	Closing Tag 완료 문자열(default: ">")

setDelimMode(mode)

시트 문자열에 STX(₩u0002), ETX(₩u0003)이 포함된 경우 설정한다.

Return	없음		
Parameter	Type	필수여부	설 명

mode	boolean	필수	0: 시트 구분자로 STX, ETX 문자를 사용한다. (default) 1: 시트 구분자로 변형된 문자열을 사용한다(시트에 설정이 되어 있어야 한다.).
------	---------	----	--

setStrictHeaderMatch(strictHeaderMatch)

엄격하게 헤더 매치를 검사할지 여부를 설정한다(설정시 시트 헤더가 엑셀에 하나라도 존재하지 않으면 데이터를 로드하지 않는다.).

Return	없음		
Parameter	Type	필수여부	설 명
strictHeaderMatch	boolean	필수	엄격한 헤더 매칭 검사 여부 설정

setMaxRows(maxRows)

LoadExcel 처리를 허용할 최대 행수를 설정한다.

로드하는 데이터가 지정한 행 수보다 많을시 에러를 발생시키고 처리가 종료된다.

Return	없음		
Parameter	Type	필수여부	설 명
maxRows	Int	필수	최대 행수 설정

setMaxColumns(maxColumns)

LoadExcel 처리를 허용할 최대 열수를 설정한다.

로드하는 데이터가 지정한 행 수보다 많을시 에러를 발생시키고 처리가 종료된다.

Return	없음		
Parameter	Type	필수여부	설 명
maxColumns	Int	필수	최대 열수 설정

setSkipEmptyRow(skipEmptyrow)

엑셀의 빈 행도 로드하도록 설정한다.

Return	없음		
Parameter	Type	필수여부	설 명
skipEmptyRow	boolean	필수	엑셀의 빈 행 로드할지 여부 설정

setMaxFileSize(maxFileSize)

업로드 가능한 최대 파일 사이즈를 설정한다.

Return	없음		
Parameter	Type	필수여부	설 명
maxFileSize	long	필수	업로드 가능한 최대 파일 사이즈 설정

setService(HttpRequest, HttpResponse)

request와 response를 할당한다. (기본 생성자 사용시에만 적용)

setTextSplitter(textSplitter)

텍스트 파일의 열 구분자를 설정 (시트의 Deli 설정이 우선 적용)

Return	없음		
Parameter	Type	필수여부	설 명
textSplitter	String	필수	열의 레코드 구분자 (default : "wt")

writeToBrowser()

업로드 한 파일을 파싱해서 브라우저의 시트에 전달한다.

writeToStream(OutputStream)

업로드 한 파일을 파싱해서 OutputStream으로 전달한다.

4.4 PrintPdf

Pdf 파일 다운로드

Print2Pdf()

생성자, request와 response 할당이 없으면 setService로 등록해야 한다.

Print2Pdf(HttpServletRequest, HttpServletResponse)

request와 response를 포함한 생성자

print()

생성된 문서를 브라우저를 통해 다운로드 한다.

saveTo(path)

생성된 문서를 서버의 특정 폴더에 저장한다.

Return	없음		
Parameter	Type	필수여부	설 명
path	String	필수	저장할 서버 경로

setDPI(dpi)

생성될 PDF 문서의 해상도를 지정한다. 기본 설정은 2000이다.

Return	없음		
Parameter	Type	필수여부	설 명
dpi	Integer	필수	PDF의 해상도 (default: 2000) 값이 작을 수록 크게 출력 50 ~ 32800 값 설정

setFontFolder(path)

TTF 폰트 파일이 저장된 경로를 지정한다.

Return	없음		
Parameter	Type	필수여부	설 명
path	String	필수	폰트 파일이 저장된 경로

setPageEncoding(encoding)

페이지의 인코딩을 설정한다.

Return	없음		
Parameter	Type	필수여부	설 명
encoding	String	필수	페이지 인코딩 타입 UTF-8, EUC-KR 등

setRequest(HttpServletRequest, HttpServletResponse)

request와 response를 할당한다. (기본 생성자 사용시에만 적용)

4.5 클라이언트 API

엑셀 다운로드 시 메모리 사용과 대량 데이터 처리를 위한 메소드 옵션을 제공한다. 옵션은 ExtendParam 속성에 사용하여 전달한다.

➤ Info

- ExtendParam에서 사용 가능한 옵션은 다음과 같다.

parameter	Type	필수여부	설명
BUFFER	Number	선택	Buffer 메모리 사용량(Default:0)
USE_SIMPLE_MODE	boolean	선택	셀 서식을 사용할지 여부(Default:0)

BUFFER 속성은 엑셀 다운로드 시 메모리에 몇 건의 데이터를 BUFFER 할지 여부이다. 100 ~ 1000 값을 설정 가능하며 5만 건 이상의 데이터에서는 빠른 데이터 처리를 위해 자동으로 100으로 동작한다.

USE_SIMPLE_MODE 속성은 고속모드로써 엑셀의 셀 서식을 제거하여 빠른 처리의 다운로드를 가능하게 해준다.

➤ Example

```
// 100건씩 메모리에 Buffer함
sheet.down2Excel({"extendParam":"BUFFER=100"});

// 셀 서식을 제거하여 엑셀 다운로드 처리
sheet.down2Excel({"extendParam":"USE_SIMPLE_MODE=1"});
```

4.6 기타 API 사용

4.7.1 DRM

문서 파일에 암호화/복호화

엑셀 파일에 DRM 적용이 필요한 경우 해당 jsp 파일에서 엑셀 파일에 대한 경로를 구하여 암호화/복호화 작업을 할 수 있다.

참고1. Down2Excel.jsp

```
IBSheetDown down = new IBSheetDown(request, response);

// 생성된 엑셀 문서를 바로 다운로드 받음
//down.downToBrowser();
// 다운로드 받을 파일 이름을 얻어서 서버에 저장
String fileName = down.getFileName();
down.saveToFile("d:/" + fileName);
// 생성된 엑셀 문서를 다운로드 처리
File file = new File("d:/" + fileName);
int fileLength = (int)file.length();

// file을 이용하여 이 부분에서 엑셀문서를 DRM 처리
// biz-logic 입력

response.setContentLength(fileLength);

try {
    if (file.isFile()) {
        down.setFileHeader();
        FileInputStream fileIn = new FileInputStream(file);
```

```
ServletOutputStream out3 = response.getOutputStream();

byte[] outputByte = new byte[fileLength];

while (fileIn.read(outputByte, 0, fileLength) != -1) {
    out3.write(outputByte, 0, fileLength);
}

fileIn.close();
out3.flush();
out3.close();
}
} finally {
    file.delete();
}
```

참고2. LoadExcel.jsp

```
IBSheetLoad load = new IBSheetLoad(request, response);

// 서버에 저장된 파일 객체
File uploadFileName = load.getUploadFile();

// uploadFileName을 이용하여 이 부분에서 엑셀문서를 DRM 처리
// biz-logic 입력

//브라우저에 데이터를 전달하여 시트에 로드
load.writeToBrowser();
```

4.7.2 Direct Export

DirectDown2Excel 에는 2가지 방법으로 사용할 수 있다.

1. 시트에 화면 페이지에서 DirectDown2Excel.jsp 페이지를 호출하기 전에 비즈니스 로직 페이지를 별도로 두어서 DB 데이터를 조회한 후 엑셀 다운로드 한다.
2. 비즈니스 로직 페이지를 별도로 필요로 하지 않으며, DirectDown2Excel.jsp 페이지를 바로 호출하여 데이터를 처리 후 엑셀 다운로드 한다.

- 방법 1.

```
// 시트 화면 페이지
// 1. DirectDown2Excel 메소드 호출
var param = {
    "url":"/bussinessList.jsp",
    "extendParam":"sa_nm=양진열&sa_no=980123",
    "fileName":"PersonList.xls"
};
sheet.directDown2Excel(param);

// bussinessList.jsp 페이지
// 2. 엑셀로 내려질 데이터를 DB에서 조회
String sa_name = request.getParameter("sa_nm");
String sa_no = request.getParameter("sa_no");

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.

// map 데이터 생성시 이름은 반드시 시트의 SaveName과 동일 해야 함.

```
Java.util.List li = new java.util.List();
java.util.Map mp = null;
while(rs.next()){
    mp = new Java.util.Map();
    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

```
String forwardPath = "./DirectDown2Excel.jsp";
if(!"".equals(forwardPath)){
    RequestDispatcher rd = request.getRequestDispatcher(forwardPath);
    rd.forward(request,response);
}
```

// 5. DirectDown2Excel.jsp 페이지

```
IBSheetDown down = new IBSheetDown(request, response);
// Request에 SHEETDATA로 저장된 데이터를 시트에 포함.
down.setDirectRequestData();
down.downToBrowser();
```

- 방법 2.

```
// 시트 화면 페이지
// 1. DirectDown2Excel 메소드 호출
var param = {
    "url": "../jsp/DirectDown2Excel.jsp"
    "extendParam": "sa_nm=양진열&sa_no=980123",
    "fileName": "PersonList.xls"
};

sheet.directDown2Excel(param);

// 2. DirectDown2Excel.jsp 페이지
IBSheetDown down = new IBSheetDown(request, response);
// 데이터를 직접 insert 하는 방법.
for (int i=0; i<100; i++) {
    Map map = new HashMap<String, String>();
    map.put("param", "10");
    down.appendDataMap(map);
}

down.downToBrowser();
```


4.7.3 Direct import

DirectLoadExcel 에는 2가지 방법으로 사용할 수 있다.

1. DirectLoadExcel.jsp 페이지 외에 엑셀파일의 데이터처리를 담당할 FP 페이지를 별도로 호출한다.
2. 별도 FP 페이지 없이 DirectLoadExcel.jsp만을 호출하여 엑셀파일의 데이터를 list로 받아서 가공한 데이터를 서버에 저장하거나 IBSheet에 로드 한다.

- 방법 1.

```
// 시트 화면 페이지
// 1. DirectLoadExcel 메소드 호출
var opt = {"extendParam":"FP=../fp_load.jsp"};
sheet.directLoadExcel(opt);

// 2. DirectLoadExcel.jsp 페이지
IBSheetLoad load = new IBSheetLoad(request, response);
// FP 페이지 호출
load.sendDirectToFP();

// 3. fp_load.jsp 페이지
String PRINT_STR = "";
List keys = new ArrayList();
// request 객체 안에 "SHEETDATA" 라는 이름으로 엑셀 내용을 받는다.
List li = (List)request.getAttribute("SHEETDATA");

for(int i=0;i<li.size();i++){
    Map mp = (Map)li.get(i);
    // 시트 헤더 내용
    if(i==0){
```

```
        Iterator it = mp.keySet().iterator();
        while(it.hasNext()){
            String key = (String)it.next();
            PRINT_STR += key+"\t\t";
            keys.add(key);
        }
        PRINT_STR += "\n";
    }

    // 시트 데이터 내용
    for(int c=0;c<keys.size();c++){
        PRINT_STR += mp.get(keys.get(c))+"\t\t";
    }
    PRINT_STR += "\n";

    PRINT_STR += li.get(i)+"\n";
}
```

- 방법 2.

```
// 시트 화면 페이지
// 1. DirectLoadExcel 메소드 호출
var opt = {"extendParam":"TestParam=test"}
sheet.directLoadExcel(opt);

// 2. DirectLoadExcel.jsp 페이지
IBSheetLoad load = new IBSheetLoad(request, response);

//ExtendParam 받기
//1. IBSheetLoad 선언 이후 ExtendParam 사용 가능
//String exParam = load.getExtendParam("TestParam");
//System.out.println("exParam [" + exParam + "]");
//2. getAttribute로 받기
```

```
//System.out.println("exParam [" + request.getAttribute("TestParam") + "]);  
  
// 엑셀 파일의 내용을 List<Map<String, String>> 형태로 받는다.  
List list = load.directLoadExcel();  
  
// 이 부분에서 list에 담긴 데이터를 가공  
  
// 데이터를 브라우저로 전송.  
//load.writeToBrowser();
```

4.7.4 header 및 footer 기능

엑셀다운시 보이는 IBSheet 헤더 위에 Header타이틀과 데이터 아래부분에 Footer 타이틀을 표시한다.

- 1) Header 및 Footer 영역에 라인별 텍스트를 설정할 수 있다.
- 2) 각 라인별 폰트 및 배경색 등 Style 설정을 설정할 수 있다.
- 3) Merge 설정은 사용자 정의로 사용할 수 없으며, 엑셀 다운시 보이는 IBSheet에 첫 컬럼 부터 IBSheet에 마지막 컬럼까지의 영역을 한줄로 Merge 하여 표현한다.
- 4) Header 및 Footer 설정시 기존에 TitleText, TitleAlign, UserMerge는 무시되어 처리되지 않으며, Header 및 Footer 설정이 없는 경우만 동작한다.

- Header 및 Footer에 설정할 수 있는 속성은 다음과 같다.

속성	type	필수여부	설명
Text	String	선택	텍스트 Default=""(사용안함)
Align	String	선택	정렬 Default="Center"
Height	String	선택	행 높이 Default=0
Style	Object	선택	스타일

- Style로 설정할 수 있는 속성은 아래와 같다.

속성	type	필수여부	설명
fontSize	Number	선택	폰트 크기 Default=0
fontName	String	선택	폰트명 Default="돋움"
fontColor	String	선택	폰트 색상 Default="검정색" (webColor : #000000)
fontBold	boolean	선택	폰트 굵기 여부 Default=0
fontUnderline	boolean	선택	폰트 밑줄 여부 Default=0
backColor	String	선택	배경색 Default="흰색" (webColor : #FFFFFF)
border	boolean	선택	외곽선 여부 Default=1

➤ Example

// 2행의 헤더 타이틀과 1행의 바닥 타이틀을 엑셀로 다운 받는다.

```
var opt = {
  "header":[
    {
      "text":"헤더1 ",
      "align":"Center",
      "height":"20",
      "style":{
        "fontSize": "10",
        "fontName": "",
        "fontColor": "",
        "fontBold": "",
        "fontUnderline": "",
        "backColor": ""
      }
    },
    {
      "text":"헤더2",
      "align":"Center",
      "height":"20",
      "style":{
        "fontSize": "10",
        "fontName": "",
        "fontColor": "",
        "fontBold": "",
        "fontUnderline": "",
        "backColor": ""
      }
    }
  ],
  "footer":[
    {
      "text": "",
```

```
"align": "",
"height": "",
"style":{
    "fontSize": "",
    "fontBold": "",
    "fontUnderline": "",
    "backColor": ""
}
}
]
}
sheet.down2Excel(opt);
```