

INNORIX Platform WP v9.2 Manual

INDEX

I. 제품 구성	3
A. Full Product.....	3
B. Simple Product.....	3
II. 설치 및 동작확인.....	4
A. Full Product.....	4
B. Simple Product.....	5
III. 제품 기본 설정	7
A. 클라이언트측 환경 설정.....	7
B. 서버측 환경 설정.....	8
IV. 제품 상세 설정	8
A. 컨트롤 리스트 옵션 목록.....	8
B. 클라이언트 측 메소드 목록	11
C. 서버측 메서드 목록	32
D. 클라이언트 측 이벤트 목록	34
E. 서버 측 이벤트 목록.....	43
V. 활용	45
A. UI 구성.....	45
B. 커스텀 페이지 구성	49
C. 부가기능	52

I. 제품 구성

A. Full Product.

LicenseKey.txt

라이선스 키 파일

/innorix	제품구성 공통 리소스
innorix.js	전체 제품 통합 js 파일
innorix.css	전체 제품 통합 css 파일
/install	배포 Agent 파일 및 Install.html / 리소스
/img	제품 구성 이미지
/icon	제품 구성 아이콘
/WEB-INF/lib	WAS 라이브러리 디렉토리
InnorixJAVA-WP-{reversion}.jar	서버측 JAVA 모듈
/innorix/exam	예제 파일 모음
upload.html	업로드 화면구성 기본예제
upload.jsp	업로드 서버측 처리 기본예제
download.html	다운로드 화면구성 기본예제
download.jsp	다운로드 서버측 처리 기본예제

B. Simple Product.

/innorix	제품구성 공통 리소스
innorix.js	전체 제품 통합 js 파일
innorix.css	전체 제품 통합 css 파일
/img	제품 구성 이미지
/icon	제품 구성 아이콘
/WEB-INF/lib	WAS 라이브러리 디렉토리
InnorixJAVA-WP-{reversion}.jar	서버측 JAVA 모듈
/innorix/exam	예제 파일 모음
upload.html	업로드 화면구성 기본예제
upload.jsp	업로드 서버측 처리 기본예제

download.html

다운로드 화면구성 기본예제

download.jsp

다운로드 서버측 처리 기본예제

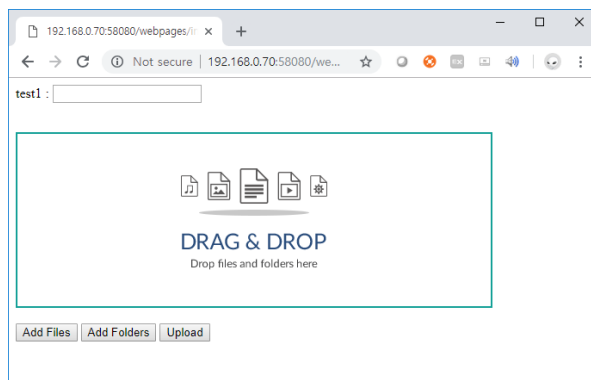
II. 설치 및 동작확인.

A. Full Product.

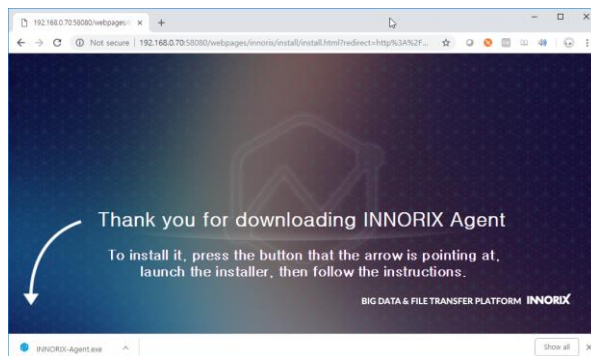
- 1) WEB 또는 WAS의 Context root에 제품파일의 innorix 디렉토리를 복사합니다.
- 2) WAS의 WEB-INF/lib 경로에 InnorixJAVA-WP-{reversion}.jar 파일을 복사합니다.
- 3) Innorix/innorix.js 상단, var innorix_license 변수에 LicenseKey.txt 파일의 키값을 입력합니다.

```
// innorix.js  
  
var innorix_license = "FGujyWVygW3onRabv8="; // 라이선스 키값 입력
```

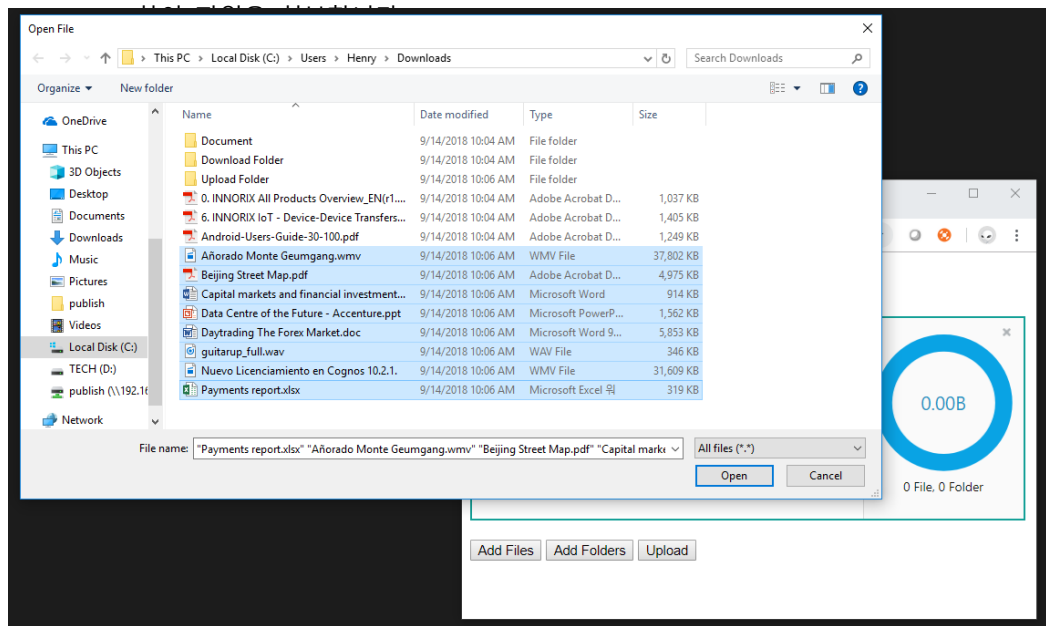
- 4) 웹 브라우저로 <http://{ServerAddress:port}/{context root}/innorix/exam/upload.html> 주소에 접속하여 기본동작을 확인합니다.



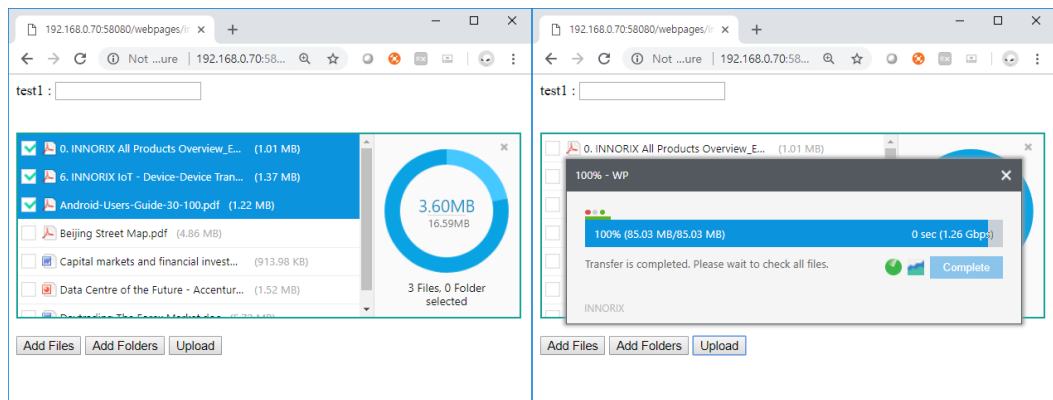
- 5) 최초 페이지 접근 시, INNORIX Agent 설치를 위해 install.html로 자동으로 이동하며, 다운로드 되는 INNORIX Agent를 설치합니다.



- 6) 설치가 완료된 후, 예제 페이지로 자동으로 이동하며, 화면 내 [파일추가] 버튼을 클릭하거나,



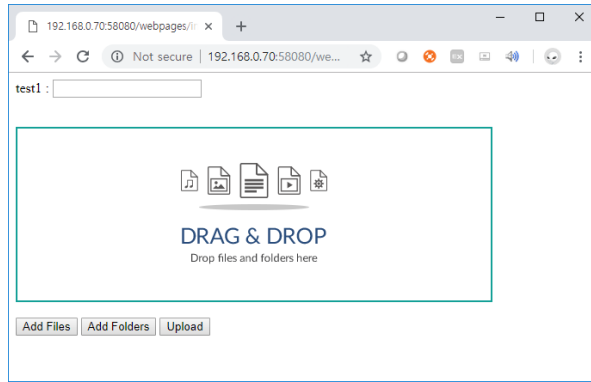
- 7) 컨트롤 리스트에 파일이 첨부된 후, 화면 내 [업로드] 버튼을 클릭하여 전송 동작을 확인합니다.



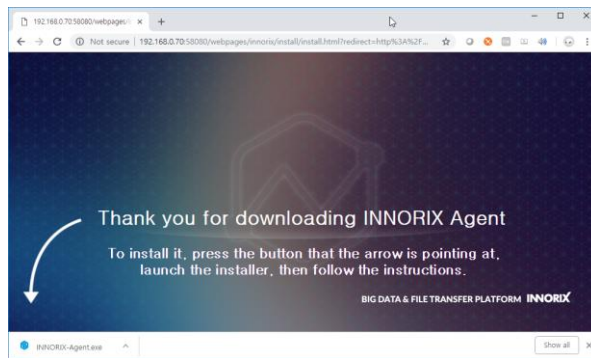
- 8) WEB 또는 WAS의 context root의 innorix/exam/data 디렉토리에서 업로드 한 파일이 정상적으로 저장됨을 확인하실 수 있습니다.

B. Simple Product.

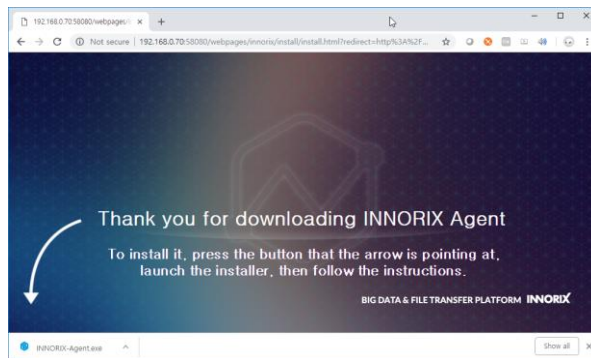
- 1) <http://opentransfer.com/~>를 통해 Simple Product package를 다운로드 받습니다.
- 2) WEB 또는 WAS의 Context root에 제품파일의 innorix 디렉토리를 복사합니다.
- 3) WAS의 WEB-INF/lib 경로에 InnorixJAVA-WP-{reversion}.jar 파일을 복사합니다.
- 4) 웹 브라우저로 <http://{ServerAddress:port}/{context root}/innorix/exam/upload.html> 주소에 접속하여 기본동작을 확인합니다.



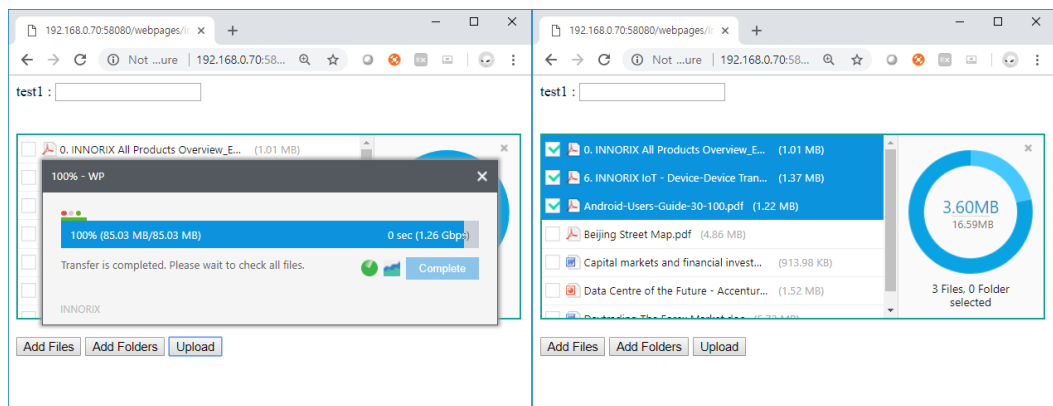
- 5) 최초 페이지 접근 시, INNORIX Agent 설치를 위해 install.html로 자동으로 이동하며, 다운로드 되는 INNORIX Agent를 설치합니다.



- 6) 설치가 완료된 후, 예제 페이지로 자동으로 이동하며, 화면 내 [파일추가] 버튼을 클릭하거나, Drag&Drop하여 파일을 첨부합니다.



- 7) 컨트롤 리스트에 파일이 첨부된 후, 화면 내 [업로드] 버튼을 클릭하여 전송 동작을 확인합니다.



- 8) WEB 또는 WAS의 context root의 innorix/exam/data 디렉토리에서 업로드 한 파일이 정상적으로 저장됨을 확인하실 수 있습니다.

III.제품 기본 설정

A. 클라이언트측 환경 설정

- 1) HTML 페이지에서 경로 및 컨트롤의 옵션과 이벤트, 메소드를 구성할 수 있습니다.

```

<link rel="stylesheet" href="../innorix.css">                                     // innorix.css 경로 설정
<script src="../innorix.js"></script>                                           // innorix.js 경로 설정
<script>
// 컨트롤 생성
var control = new Object();
control = innorix.create({
    el                : "#fileControl",                                         // 컨트롤 출력 객체 ID
    uploadUrl         : "../upload.jsp",                                         // 파일 업로드 URL
    installUrl        : "../install/install.html",                             // install 페이지 URL
    maxFileCount      : 100,                                                     // 첨부가능 파일 전체 개수
    maxFileSize       : 1000000,                                                 // 첨부가능 전체 사이즈(bytes)
    maxTotalSize      : 10000,                                                  // 첨부가능 파일 사이즈(bytes)
    width             : 550,                                                    // 컨트롤 출력 너비(pixel)
    height            : 200                                                     // 컨트롤 출력 높이(pixel)
})

// 업로드 완료 이벤트
control.on('uploadComplete', function(p) {
    console.log(p);
});
...

```

```
<input type="button" value="업로드" onclick="control.upload();"/> // 파일 업로드 메소드
```

B. 서버측 환경 설정

- 1) upload.jsp와 download.jsp 등 서버 측에서 저장경로 지정 등의 설정을 구성할 수 있습니다.

```
String directory = "/wwwroot/INNORIX/data"; // 업로드 저장경로 설정
```

```
InnorixUpload uploader = new InnorixUpload(request, response, maxPostSize, directory);
```

IV.제품 상세 설정

A. 컨트롤 리스트 옵션 목록

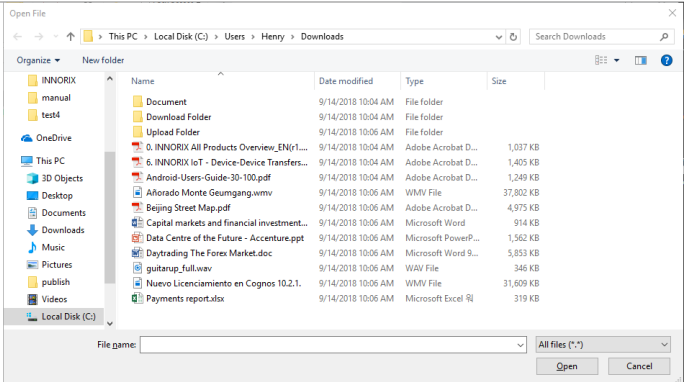
속성	타입	설명	기본값
addDuplicateFile	Boolean	중복파일 추가 유무	true
agent	Boolean	true : Agent 설치형 모드 사용 false : HTML5 모드 사용 ※ 모드 간 동작차이는 별첨 참조	true
allowExtension	String Array	정의된 확장자 파일만 추가 e.g ["exe", "msi", "cab"] ※ limitExtension과 동일 확장자 정책 지정 시, limitExtension을 우선 적용	
autoCheckVersion	Boolean	INNORIX Agent 버전체크 이벤트 활성화	false
autoReTransfer	Boolean	전송 에러발생 시 자동 재전송	true
cancelConfirmation	Boolean	취소 시, alert창 출력 옵션	false
checkLocation	Boolean	유저의 위도, 경도 위치정보 체크	false
downloadPath	String	기본 다운로드 경로를 지정 e.g "c:WWdownload"	
downloadType	String	다운로드 URL 타입을 지정 direct : http://abc.com/file.dat 형식 stream : http://abc.com/down.jsp?fid=33 형식	stream
maxFileSize	Number	첨부가능 파일 전체 사이즈(bytes)	무제한
maxTotalSize	Number	첨부가능 1개 파일 전체 사이즈(bytes)	무제한
limitExtension	String Array	정의된 확장자 파일을 제외 e.g ["exe", "msi", "cab"]	
draggableTransferWindow	Boolean	전송창 드래그로 이동 제어	true

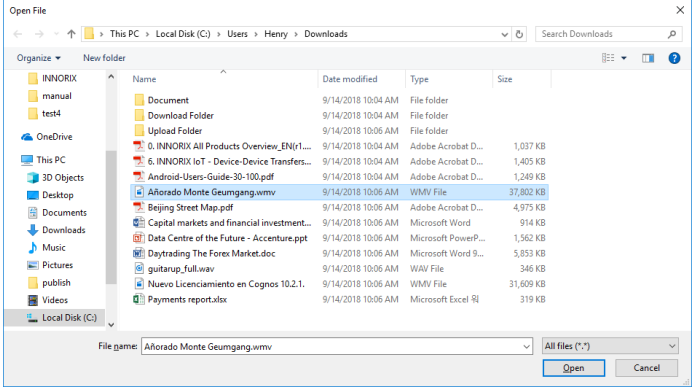
enableDropZone	Boolean	드롭존 활성화	true
encryptMetaData	Boolean	전송 메타정보 암호화	false
folderIntact	Boolean	폴더첨부 유무	true
height	Number	리스트 컨트롤 높이(pixel)	200
iframeView	Boolean	Iframe에서 전송창 출력 확장	true
iframeViewCssUrl	String	Parent frame에서 참조할 innorix.css url	
iframeViewLeft	Number	전송창 출력할 장축 기준 위치 (pixel)	
iframeViewLocation	String	전송창 출력할 위치 top : frame 내 최상단에서 출력 parent : 바로 상위 frame 에서 출력	
iframeViewMarginLeft	Number	출력 영역 내 장축 기준 마진 (pixel)	
iframeViewMarginTop	Number	출력 영역 내 상단 기준 마진 (pixel)	
iframeViewTop	Number	전송창 출력 할 상단 기준 위치 (pixel)	
initSpeedCheck	Boolean	스피드체크 구간 활성화 여부	false
installPopupWindow	Boolean	Install.html 팝업으로 출력 여부	false
installPopupWindowHeight	Number	출력되는 팝업창 높이 (pixel)	
installPopupWindowLeft	Number	출력되는 팝업창 장축 기준 위치 (pixel)	
installPopupWindowTop	Number	출력되는 팝업창 상단 기준 위치 (pixel)	
installPopupWindowWidth	Number	출력되는 팝업창 폭 (pixel)	
installUrl	String	Agent 설치 안내 URL e.g installUrl : "install/install.html"	
isHighSpeed	Boolean	고속전송 활성 여부	true
isMobile	Boolean	모바일에서 App 연동 여부	false
isSliceTransfer	Boolean	조각 전송 활성 여부	true
limitExtension	String Array	정의된 확장자 파일 첨부 제한 e.g ["exe", "msi", "cab"] ※ allowExtension과 동일 확장자 정책 지정 시, limitExtension을 우선 적용	
limitRate	Number	전송 대역폭 제한 (Kbyte)	
loadTransfer	Boolean	이어 전송 기능 활성 여부	true
locationUrl	String	위치정보를 전달할 서버 URL e.g "http://demo.innorix.com"	
maxFileCount	Number	첨부가능 파일 전체 개수	무제한

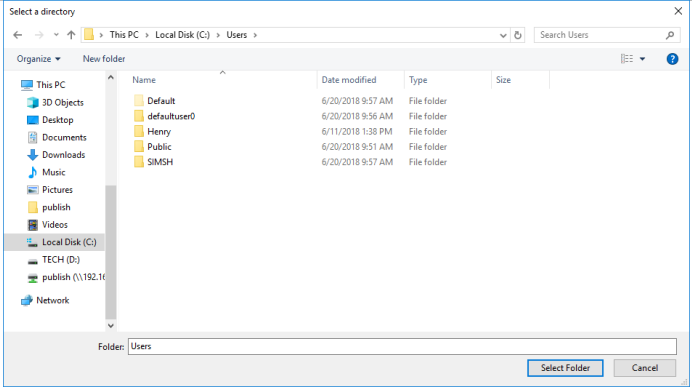
maxFileSize	Number	첨부가능 개별파일 최대 용량 (Byte)	무제한
maximumErrorCount	Number	최대 에러 허용 횟수	무제한
maxMassFileListCount	Number	대량파일 전송모드 전환 기준 파일 개수	1000
maxTotalSize	Number	첨부가능 전체파일 최대 용량 (Byte)	무제한
monitorServer	String	전송 정보를 전달할 MRT 서버 URL e.g "http://demo.innorix.com/dispatch/mt/transfer"	
platformDepend	Boolean	INNORIX Platform 연동 여부	false
progressTitle	String	전송창 타이틀 출력 문구	WP
resumeDownloadType	String	다운로드 이어받기 정책 overwrite : 덮어쓰기 relay : 이어받기 numbering : 파일명 자동 넘버링 저장 confirm : 사용자 확인	confirm
resumeType	String	업로드/다운로드 이어전송 통합 정책 overwrite : 덮어쓰기 relay : 이어받기 numbering : 파일명 자동 넘버링 저장 confirm : 사용자 확인	upload : relay / download : confirm
resumeUploadType	String	업로드 이어올리기 정책 overwrite : 덮어쓰기 relay : 이어올리기	relay
retryCount	Number	재전송 횟수	무제한
retryDelay	Number	재전송 간격 (sec)	3
showFolderItems	Boolean	폴더구조의 리스트 출력 여부	true
showFolderItemsTree	Boolean	폴더트리 구조 리스트 출력 여부	true
showGraph	Boolean	우측 그래프 출력	true
showQrCode	Boolean	QR코드 출력	true
showTransferFileStatus	Boolean	리스트 전송창 모드에서 파일별 전송상태 출력	false
showTransferWindow	Boolean	전송창 출력	true
skipAgentCheck	Boolean	에이전트 자동체크 비활성화	false
transferMode	String	업로드, 다운로드 모드 설정 both : 업로드, 다운로드 혼합사용 upload : 업로드 기능만 사용 download : 다운로드 기능만 사용	both

transferStart	JSON Text	전송 자동시작 유무 e.g { "upload": "auto", "download": "manual" } auto : 자동으로 전송시작 manual : 사용자 확인 후 전송	"upload": "auto", "download": "manual"
transferWindowMode	Boolean	리스트 전송창 모드 출력	false
transferWindowModeOpen	Boolean	리스트 전송창 모드에서 다운로드 폴더 열기 추가	false
transferWindowPolicy	Boolean	리스트 전송창 모드에서 이어전송 옵션 추가	false
uploadUrl	String	업로드 URL	
useContextMenu	Boolean	우클릭 컨텍스트 메뉴 출력	true
useCrypt	Boolean	암호화 전송 설정	false
useProxy	Boolean	프록시 설정	false
width	Number	리스트 컨트롤 높이 (pixel)	200
showFileListHeader	Boolean	리스트 헤더에 선택툴팁 출력	false
skipErrorFile	Boolean	에러발생 시, 해당 파일 건너뛰기	false
useFuncBtn	Boolean	파일 아이템 이동, 삭제 기능버튼 출력여부	true
cancelConfirmation	Boolean	전송 취소 시, confirm창 출력	false
useDownloadAction	Boolean	다운로드 시작-완료 시점 캐치	false

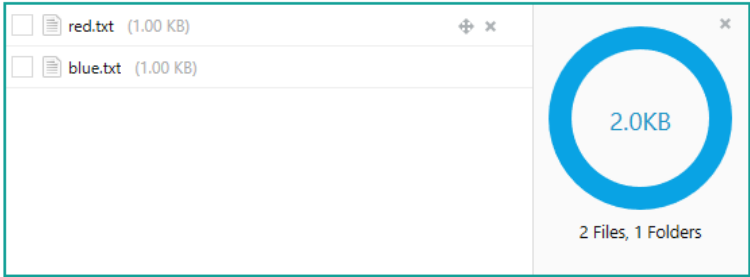
B. 클라이언트 측 메소드 목록

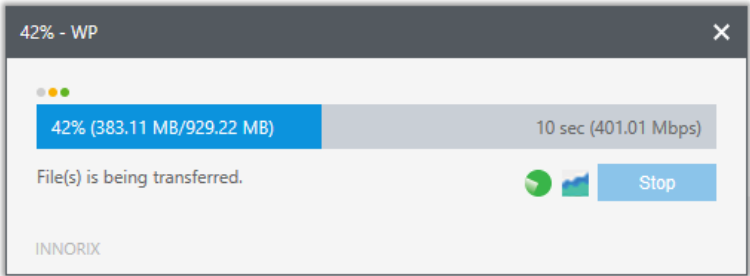
메소드명	openFileDialog()
리턴타입	없음 (void)
입력 파라미터	없음
설명	멀티 파일 선택창 출력
예제 화면	

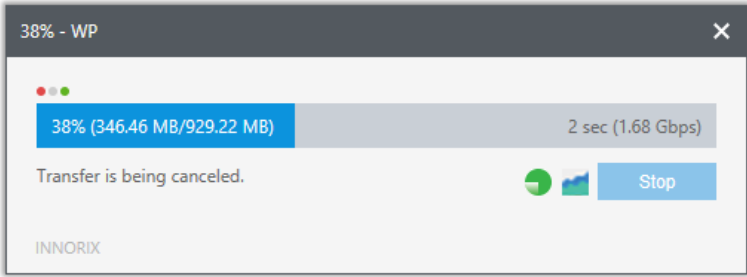
메소드명	openFileDialogSingle()
리턴타입	없음 (void)
입력 파라미터	없음
설명	단일 파일 선택창 출력
예제 화면	

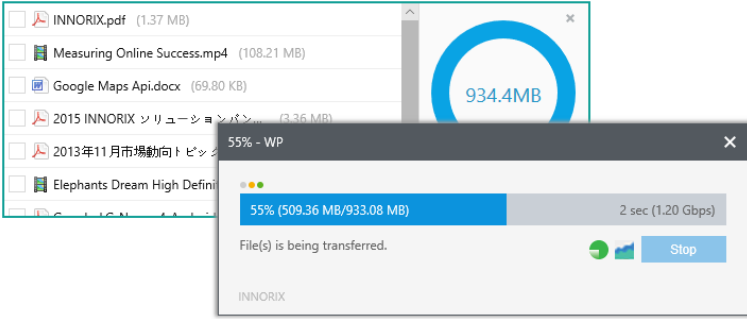
메소드명	openFolderDialog()
리턴타입	없음 (void)
입력 파라미터	없음
설명	폴더 선택창 출력
예제 화면	

메소드명	addFiles()
리턴타입	없음 (void)
입력 파라미터	# JSON Object <pre>[{</pre>

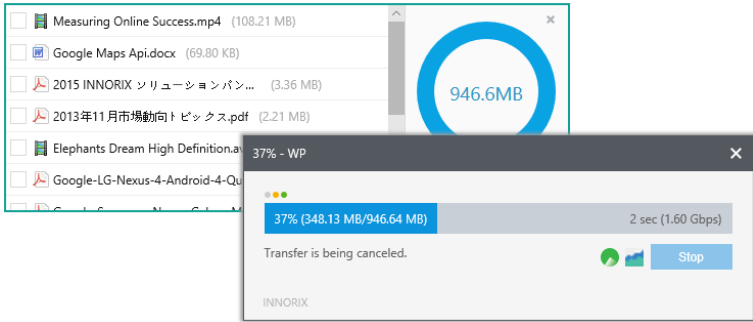
	<pre> "filePath": "C:\\data\\red.txt", // 파일 경로 (String) "fileSize": 1024, // 파일 사이즈(Number) / Byte "uploadUrl": "upload.jsp" // 파일별 업로드 URL (* optional) (String) }, { "filePath": "C:\\data\\blue.txt", // 파일 경로 (String) "fileSize": 1024, // 파일 사이즈(Number) / Byte "uploadUrl": "upload.jsp" // 파일별 업로드 URL (* optional) (String) }] </pre>
설명	업로드 파일 추가
예제 화면	

메소드명	upload()
리턴타입	없음 (void)
입력 파라메터	없음
설명	모든 파일 업로드 시작
예제 화면	

메소드명	uploadCancel()
리턴타입	없음 (void)
입력 파라미터	없음
설명	진행 중인 업로드 취소
예제 화면	

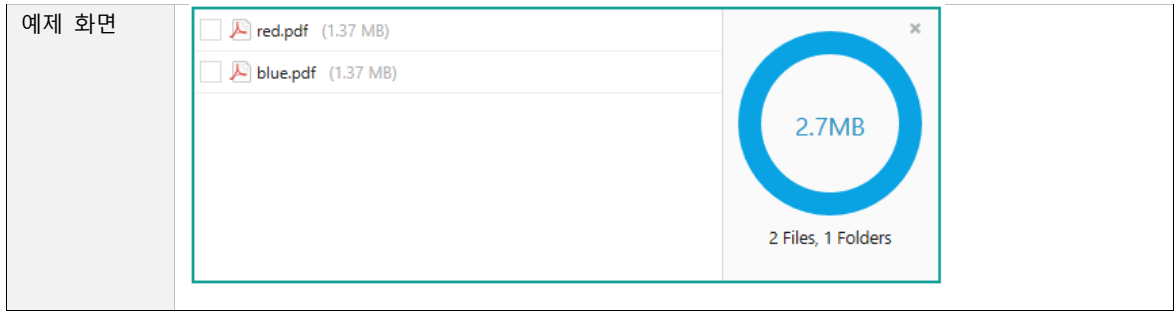
메소드명	removeAllFiles()
리턴타입	없음 (void)
입력 파라미터	없음
설명	모든 파일 제거
메소드명	download()
리턴타입	없음 (void)
입력 파라미터	없음
설명	모든 파일 다운로드 시작
예제 화면	

메소드명	downloadSelectedFiles()
리턴타입	없음 (void)
입력 파라미터	없음

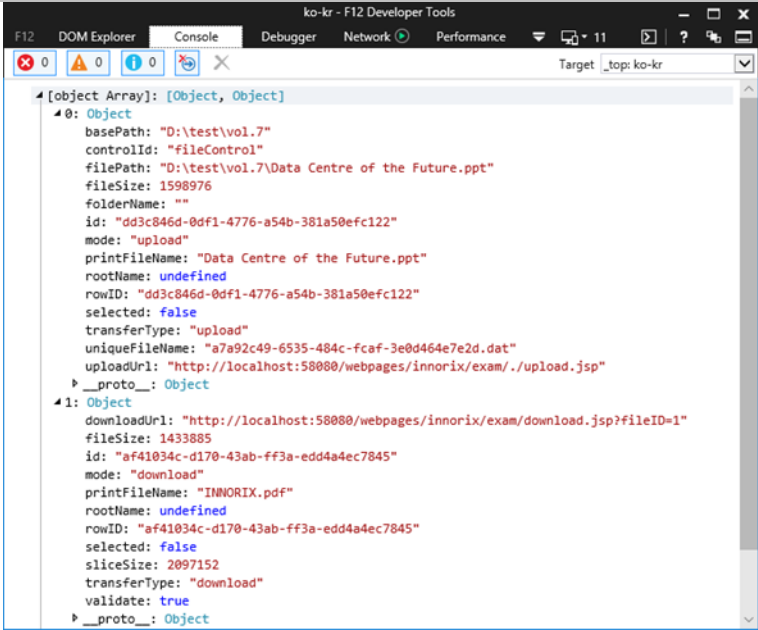
설명	선택 파일 다운로드
예제 화면	

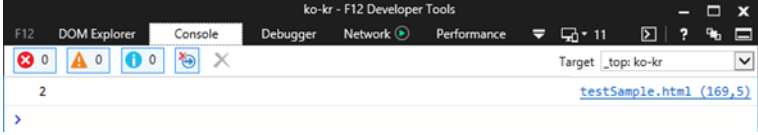
메소드명	downloadAndOpen()
리턴타입	없음 (void)
입력 파라미터	없음
설명	마지막 선택 인덱스 파일의 다운로드 완료 후 파일 열기

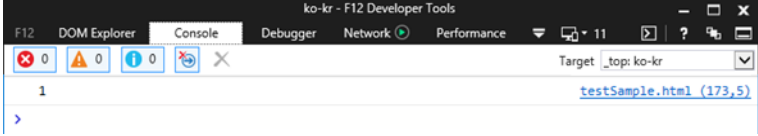
메소드명	presetDownloadFiles()
리턴타입	없음 (void)
입력 파라미터	<p># JSON Object</p> <pre>[{ printFileName: "red.pdf ", // 파일출력명 (String) fileSize: 1433885, // 파일사이즈 (Number) / Byte downloadUrl: "http://서버주소/download.jsp?fileID=111"//다운로드URL (String) }, { printFileName: "blue.pdf", // 파일출력명 (String) fileSize: 1433885, // 파일사이즈 (Number) / Byte downloadUrl: "http://서버주소/download.jsp?fileID=222"//다운로드URL (String) }]</pre>
설명	다운로드 파일 구성



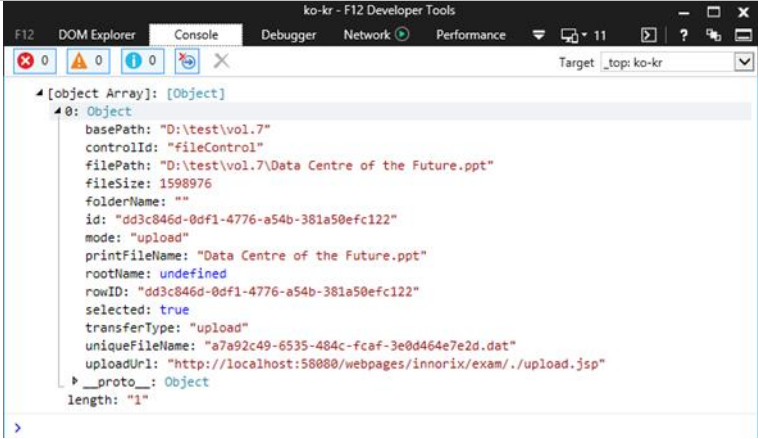
메소드명	getAllFiles()
리턴타입	<pre> # JSON Object - 업로드 파일 정보 [{ "basePath":"D: WtestWvol.7", // 첨부경로 (String) "controlId":"fileControl", // 컨트롤ID (String) "filePath":""D:WtestWvol.7WData Centre of Future.ppt"",// 파일 전체경로 (String) "fileSize":2506093, // 파일 사이즈(Number) / Byte "folderName":"", // 폴더명 (String) "id":"fda07096-917b-4dde-fe23-bb0114fbe8ad", // 파일ID (String) "mode":"upload", // 첨부모드 (String) "printFileName":" Data Centre of Future.ppt ", // 출력파일명 (String) "rootName":"", // 폴더정보 (String) "rowID":"fda07096-917b-4dde-fe23-bb0114fbe8ad", // 파일리스트ID (String) "selected":false, // 파일선택여부 (Boolean) "transferType":"upload", // 전송유형 (String) "uniqueFileName":"950491e5.....db6d5529e404.dat", // 파일고유저장명 (String) "uploadUrl":"http://localhost/webpages/.upload.jsp"// 업로드URL (String) },] # JSON Object - 다운로드 파일 정보 [</pre>

	<pre> { "downloadUrl ":"http://localhost/download.jsp?fileID=1",// 다운로드 URL (String) "fileSize ":1433885, // 파일사이즈 (Number) / Byte "id":"8d6f7747-414e-4ad6-bc84-aa4392edf7d6", // 파일ID (String) "mode":"download", // 첨부모드 (String) "printFileName":"INNORIX.pdf", // 출력파일명 (String) "rowID":"8d6f7747-414e-4ad6-bc84-aa4392edf7d6", // 파일리스트ID (String) "selected":true, // 파일선택여부 (Boolean) "sliceSize":2097152, // 파일조각사이즈 (Number) / Byte "transferType":"download", // 전송타입 (String) "validate":true // 파일유효여부 (Boolean) } </pre>
입력 파라미터	없음
설명	모든 파일 정보 출력
예제 화면	

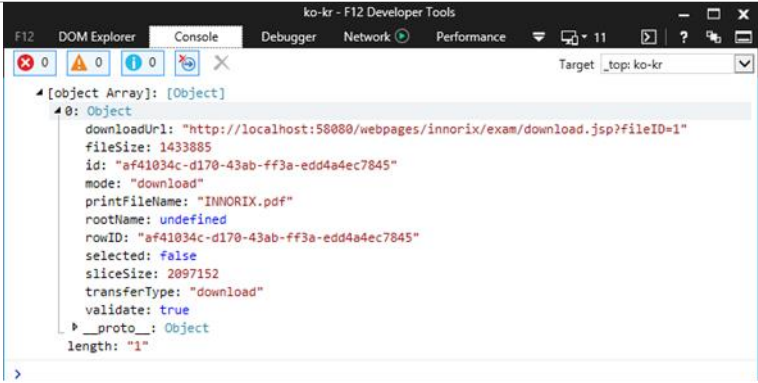
메소드명	getFileCount()
리턴타입	(Number)
입력 파라미터	없음
설명	모든 파일 개수(Number) 출력
예제 화면	

메소드명	getSelectedFileCount()
리턴타입	(Number)
입력 파라미터	없음
설명	선택 파일 개수(Number) 출력
예제 화면	

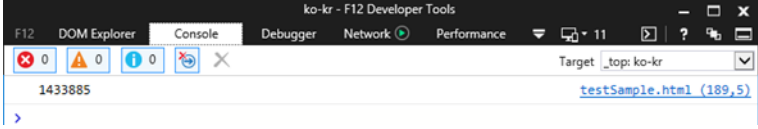
메소드명	getUploadFiles()
리턴타입	# JSON Object [{ "basePath": "D:\test\vol.7", // 첨부경로 (String) "controlId": "fileControl", // 컨트롤ID (String) "filePath": "D:\test\vol.7\Data Centre of Future.ppt", // 파일 전체경로 (String) "fileSize": 2506093, // 파일 사이즈(Number) / Byte "folderName": "", // 폴더명 (String) "id": "fda07096-917b-4dde-fe23-bb0114f8e8ad", // 파일ID (String) "mode": "upload", // 첨부모드 (String) "printFileName": "Data Centre of Future.ppt", // 출력파일명 (String) "rootName": "", // 폴더정보 (String) }]

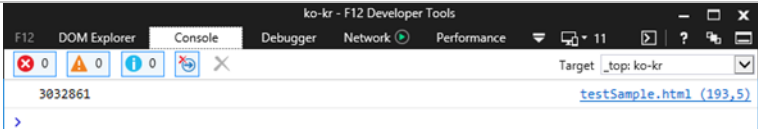
	<pre> "rowID":"fda07096-917b-4dde-fe23-bb0114fbe8ad", // 파일리스트ID (String) "selected":false, // 파일선택여부 (Boolean) "transferType":"upload", // 전송유형 (String) "uniqueFileName":"950491e5.....db6d5529e404.dat", // 파일고유저장명 (String) "uploadUrl":"http://localhost/webpages/./upload.jsp"// 업로드URL (String) },] </pre>
입력 파라미터	없음
설명	업로드 전체 파일 정보 출력
예제 화면	

메소드명	getDownloadFiles()
리턴타입	<p># JSON Object – 다운로드 파일 정보</p> <pre> [{ "downloadUrl":"http://localhost/download.jsp?fileID=1",// 다운로드 URL (String) "fileSize ":1433885, // 파일사이즈 (Number) / Byte "id":"8d6f7747-414e-4ad6-bc84-aa4392edf7d6", // 파일ID (String) "mode":"download", // 첨부모드 (String) "printFileName":"INNORIX.pdf", // 출력파일명 (String) "rowID":"8d6f7747-414e-4ad6-bc84-aa4392edf7d6", // 파일리스트ID (String) "selected":true, // 파일선택여부 (Boolean) }] </pre>

	<pre> "sliceSize":2097152, // 파일조각사이즈 (Number) / Byte "transferType":"download", // 전송타입 (String) "validate":true // 파일유효여부 (Boolean) }]</pre>
입력 파라미터	없음
설명	다운로드 전체 파일 정보 출력
예제 화면	

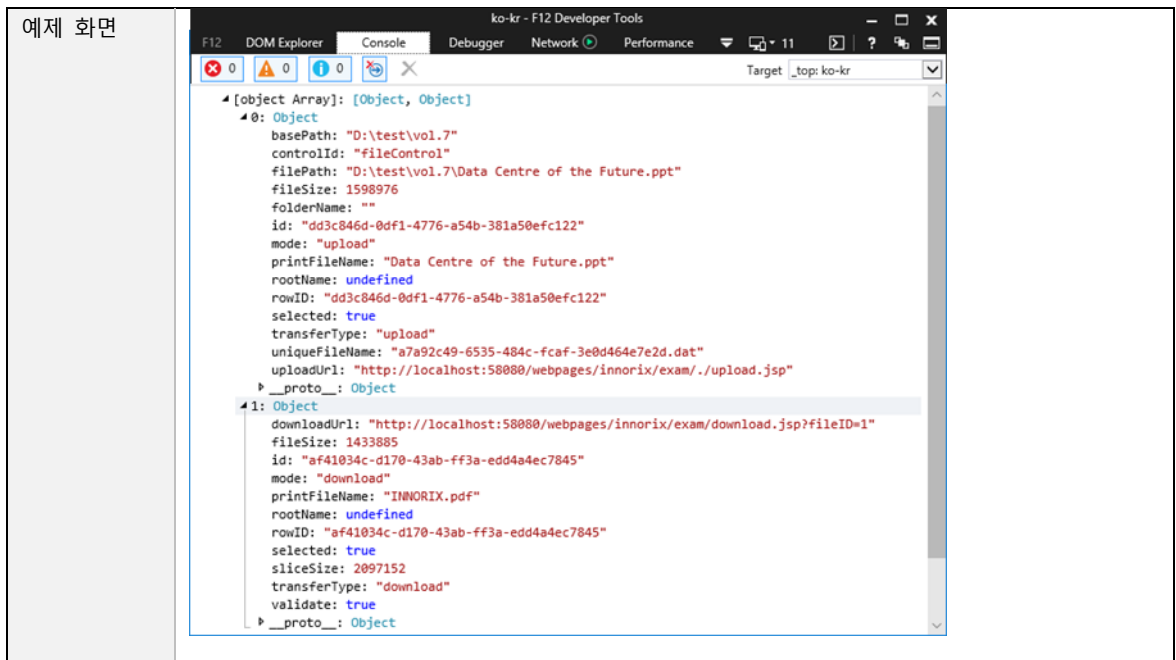
메소드명	getUploadFileSize()
리턴타입	(Number)
입력 파라미터	없음
설명	업로드 대상 전체 파일 사이즈(Number) 출력
예제 화면	

메소드명	getDownloadFileSize()
리턴타입	(Number)
입력 파라미터	없음
설명	다운로드 대상 전체 파일 사이즈(Number) 출력
예제 화면	

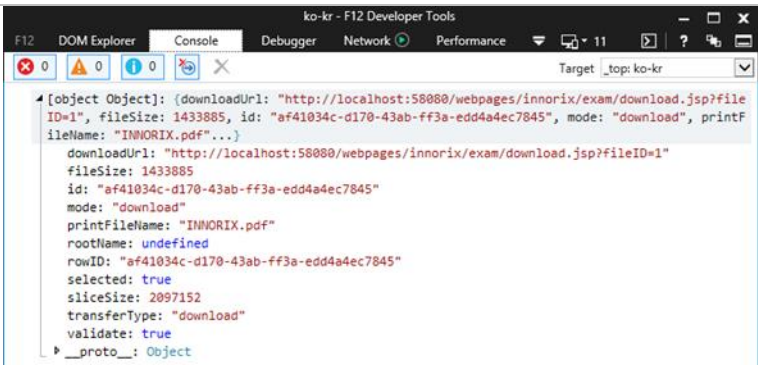
메소드명	getTotalSize()
리턴타입	(Number)
입력 파라미터	없음
설명	전체 파일 사이즈(Number)를 출력
예제 화면	

메소드명	getSelectedFiles()
리턴타입	<p># JSON Object – 업로드 파일 정보</p> <pre>[{ "basePath": "D:\test\vol.7", // 첨부경로 (String) "controllId": "fileControl", // 컨트롤ID (String) "filePath": "D:\test\vol.7\Data Centre of Future.ppt", // 파일 전체경로 (String) "fileSize": 2506093, // 파일 사이즈(Number) / Byte "folderName": "", // 폴더명 (String) "id": "fda07096-917b-4dde-fe23-bb0114fbe8ad", // 파일ID (String) "mode": "upload", // 첨부모드 (String) "printFileName": "Data Centre of Future.ppt", // 출력파일명 (String) "rootName": "", // 폴더정보 (String) "rowID": "fda07096-917b-4dde-fe23-bb0114fbe8ad", // 파일리스트ID (String) }]</pre>

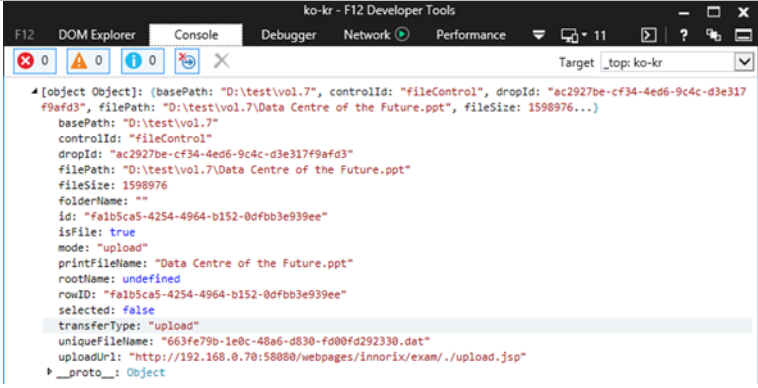
	<pre> "selected":true, // 파일선택여부 (Boolean) "transferType":"upload", // 전송유형 (String) "uniqueFileName":"950491e5-...-db6d5529e404.dat", // 파일고유저장명 (String) "uploadUrl":"http://localhost/webpages/./upload.jsp"// 업로드URL (String) },] # JSON Object – 다운로드 파일 정보 [{ "downloadUrl ":"http://localhost/download.jsp?fileID=1",// 다운로드 URL (String) "fileSize ":1433885, // 파일사이즈 (Number) / Byte "id":"8d6f7747-414e-4ad6-bc84-aa4392edf7d6", // 파일ID (String) "mode":"download", // 첨부모드 (String) "printFileName":"INNORIX.pdf", // 출력파일명 (String) "rowID":"8d6f7747-414e-4ad6-bc84-aa4392edf7d6", // 파일리스트ID (String) "selected":true, // 파일선택여부 (Boolean) "sliceSize":2097152, // 파일조각사이즈 (Number) / Byte "transferType":"download", // 전송타입 (String) "validate":true // 파일유효여부 (Boolean) }] </pre>
입력 파라미터	없음
설명	선택 파일 정보를 출력

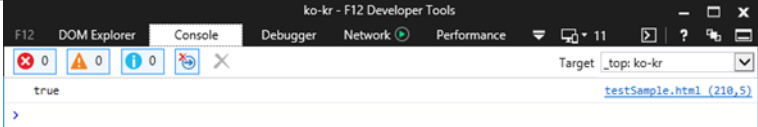


메소드명	getFileByIndex()
리턴타입	<p># JSON Object – 업로드 파일 정보</p> <pre>[{ "basePath": "D:\test\vol.7", // 첨부경로 (String) "controlId": "fileControl", // 컨트롤ID (String) "filePath": "D:\test\vol.7\Data Centre of Future.ppt", // 파일 전체경로 (String) "fileSize": 2506093, // 파일 사이즈(Number) / Byte "folderName": "", // 폴더명 (String) "id": "fda07096-917b-4dde-fe23-bb0114fbe8ad", // 파일ID (String) "mode": "upload", // 첨부모드 (String) "printFileName": "Data Centre of Future.ppt ", // 출력파일명 (String) "rootName": "", // 폴더정보 (String) "rowID": "fda07096-917b-4dde-fe23-bb0114fbe8ad", // 파일리스트ID (String) "selected": true, // 파일선택여부 (Boolean) "transferType": "upload", // 전송유형 (String) "uniqueFileName": "950491e5-...-db6d5529e404.dat", // 파일고유저장명 (String) }]</pre>

	<pre> "uploadUrl":"http://localhost/webpages/./upload.jsp"// 업로드URL (String) },] # JSON Object – 다운로드 파일 정보 [{ "downloadUrl":"http://localhost/download.jsp?fileID=1",// 다운로드 URL (String) "fileSize ":1433885, // 파일사이즈 (Number) / Byte "id":"8d6f7747-414e-4ad6-bc84-aa4392edf7d6", // 파일ID (String) "mode":"download", // 첨부모드 (String) "printFileName":"INNORIX.pdf", // 출력파일명 (String) "rowID":"8d6f7747-414e-4ad6-bc84-aa4392edf7d6", // 파일리스트ID (String) "selected":true, // 파일선택여부 (Boolean) "sliceSize":2097152, // 파일조각사이즈 (Number) / Byte "transferType":"download", // 전송타입 (String) "validate":true // 파일유효여부 (Boolean) }] </pre>
입력 파라미터	(Number)
설명	index(Number) 파일 정보를 출력
예제 화면	 <p>The screenshot shows a web browser's developer console with the 'Console' tab selected. It displays a JSON object returned from a server, containing file metadata such as download URL, file size, ID, mode, print filename, row ID, selection status, slice size, transfer type, and validation status. The object is expanded to show its internal structure.</p>

메소드명	getFileById()
리턴타입	<pre> # JSON Object – 업로드 파일 정보 [{ "basePath":"D:\test\vol.7", // 첨부경로 (String) "controllId":"fileControl", // 컨트롤ID (String) "filePath":""D:\test\vol.7\Data Centre of Future.ppt",// 파일 전체경로 (String) "fileSize":2506093, // 파일 사이즈(Number) / Byte "folderName":"", // 폴더명 (String) "id":"fda07096-917b-4dde-fe23-bb0114fbe8ad", // 파일ID (String) "mode":"upload", // 첨부모드 (String) "printFileName":" Data Centre of Future.ppt ", // 출력파일명 (String) "rootName":"", // 폴더정보 (String) "rowID":"fda07096-917b-4dde-fe23-bb0114fbe8ad", // 파일리스트ID (String) "selected":true, // 파일선택여부 (Boolean) "transferType":"upload", // 전송유형 (String) "uniqueFileName":"950491e5-...-db6d5529e404.dat", // 파일고유저장명 (String) "uploadUrl":"http://localhost/webpages./upload.jsp"// 업로드URL (String) },] # JSON Object – 다운로드 파일 정보 [{ "downloadUrl ":"http://localhost/download.jsp?fileID=1",// 다운로드 URL (String) "fileSize ":1433885, // 파일사이즈 (Number) / Byte "id":"8d6f7747-414e-4ad6-bc84-aa4392edf7d6", // 파일ID (String) "mode":"download", // 첨부모드 (String) "printFileName":"INNORIX.pdf", // 출력파일명 (String) </pre>

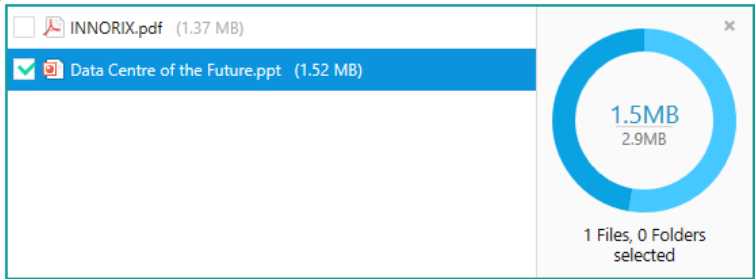
	<pre> "rowID":"8d6f7747-414e-4ad6-bc84-aa4392edf7d6", // 파일리스트ID (String) "selected":true, // 파일선택여부 (Boolean) "sliceSize":2097152, // 파일조각사이즈 (Number) / Byte "transferType":"download", // 전송타입 (String) "validate":true // 파일유효여부 (Boolean) }]</pre>
입력 파라미터	(String)
설명	id(String)에 해당하는 파일 정보 출력
예제 화면	

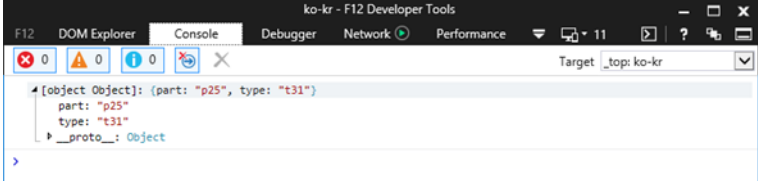
메소드명	getAgentMode()
리턴타입	(Boolean) – true : Agent 모드, false : HTML5 모드
입력 파라미터	없음
설명	사용중인 컨트롤의 Agent 모드, HTML5 모드 사용여부 확인
예제 화면	

메소드명	removeFileByIndex()
리턴타입	없음 (void)
입력 파라미터	(Number)
설명	index(Number) 파일 제거

메소드명	removeFileById()
리턴타입	없음 (void)
입력 파라미터	(String)
설명	id(String)에 해당하는 파일 제거

메소드명	removeSelectedFiles()
리턴타입	없음 (void)
입력 파라미터	없음
설명	선택 파일 제거

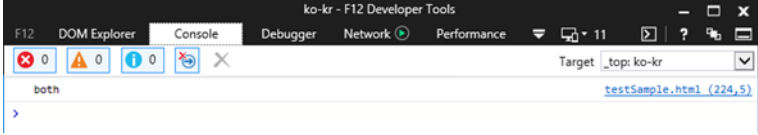
메소드명	addSelectFilesById()
리턴타입	없음 (void)
입력 파라미터	(String)
설명	id(String)에 해당하는 파일 선택
예제 화면	

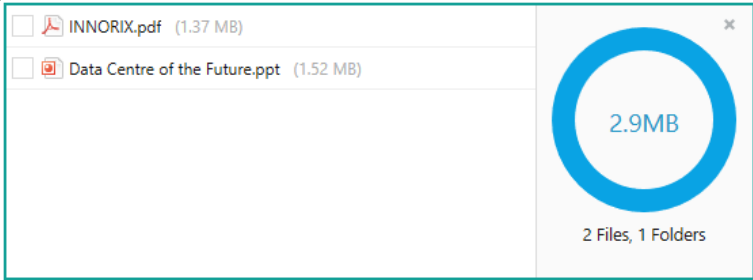
메소드명	setPostData()
리턴타입	없음 (void)
입력 파라미터	<p># JSON Object</p> <pre>{ "type" : "t31", "part" : "p25", }</pre> <p>업로드시 POST Param 으로 type = "t31", part = "p25" 값이 전달됨</p> <pre>var postObj = new Object(); postObj.type = "t31", postObj.part = "p25" control.setPostData(postObj); control.upload();</pre>
설명	POST Param 구성
예제 화면	

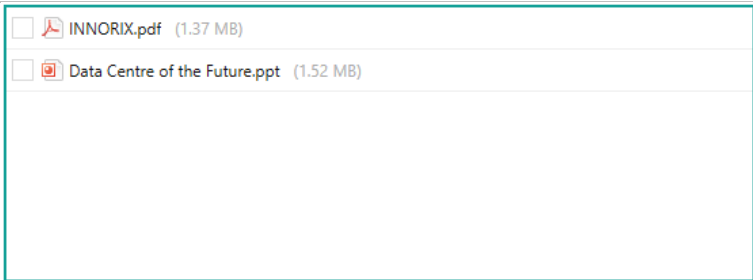
메소드명	setSize()
리턴타입	없음 (void)
입력 파라미터	Width(Number), Height(Number)
설명	컨트롤 크기 조절 -

메소드명	setCookie()
리턴타입	없음 (void)
입력 파라미터	(String) 컨트롤 로딩 완료시 웹 페이지의 세션정보 공유를 위한 쿠키설정의 예 control.on('loadComplete', function (p) { control.setCookie("JSESSIONID= <%=session.getId()%>"); }...)
설명	웹 쿠키(세션) 정보 유지

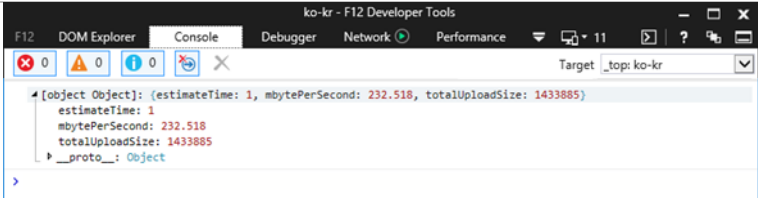
메소드명	destory()
리턴타입	없음 (void)
입력 파라미터	없음
설명	컨트롤 객체 destroy

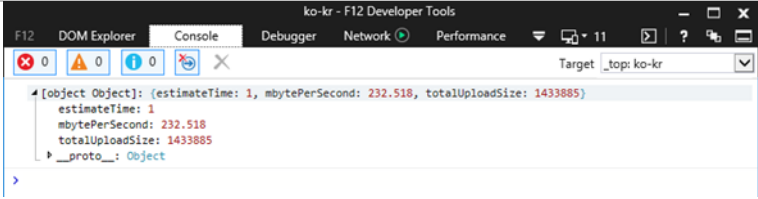
메소드명	getTransferMode()
리턴타입	(String) upload : 업로드 모드 download : 다운로드 모드 both : 업로드, 다운로드 모드 혼합 사용
입력 파라미터	없음
설명	전송모드 출력
예제 화면	 <p>The screenshot shows the F12 Developer Tools interface with the Console tab selected. The output of the <code>getTransferMode()</code> method is displayed as <code>both</code>. The target is set to <code>_top: ko-kr</code> and the source is <code>testSample.html (224,5)</code>.</p>

메소드명	showGraph()
리턴타입	없음 (void)
입력 파라미터	(Boolean) – true, false
설명	용량 그래프 출력
예제 화면	

메소드명	hideGraph()
리턴타입	없음 (void)
입력 파라미터	Boolean) – true, false
설명	용량 그래프 숨김
예제 화면	

메소드명	estimateTime()
리턴타입	# JSON Object <pre>{ "mbytePerSecond":149.781, // 초당 전송속도 (MB/s) (Number) "totalUploadSize":28311552, // 파일 사이즈 (Byte) (Number) "estimateTime":1 // 예상시간 (Second) (Number) }</pre>
입력 파라미터	없음

설명	전송 예상 시간을 출력
예제 화면	

메소드명	estimateTime()
리턴타입	# JSON Object <pre>{ "mbytePerSecond":149.781, // 초당 전송속도 (MB/s) (Number) "totalUploadSize":28311552, // 파일 사이즈 (Byte) (Number) "estimateTime":1 // 예상시간 (Second) (Number) }</pre>
입력 파라미터	없음
설명	전송 예상 시간을 출력
예제 화면	

메소드명	stopUploadServerProcessBar ()
리턴타입	없음 (void)
입력 파라미터	없음 (void)
설명	업로드 프로그레스바 상단 애니메이션 중지

메소드명	stopDownloadServerProcessBar ()
리턴타입	없음 (void)
입력 파라미터	없음 (void)
설명	다운로드 프로그레스바 상단 애니메이션 중지

메소드명	startTransferProgress()
리턴타입	없음 (void)
입력 파라미터	없음 (void)
설명	팝업모드에서 다운로드 바로시작

메소드명	closeTransferWindow()
리턴타입	없음 (void)
입력 파라미터	없음 (void)
설명	전송창 바로 닫힘

메소드명	setAgentMode()
리턴타입	없음 (void)
입력 파라미터	Boolean) – true, false
설명	Agent 동작 모드 동적 할당

메소드명	execute()
리턴타입	없음 (void)
입력 파라미터	String - fileName
설명	업로드 대상 파일 실행

C. 서버측 메서드 목록

메소드명	setOverwrite()
리턴타입	없음 (void)
입력 파라미터	Boolean) – true, false
설명	중복파일 덮어쓰기

메소드명	setAutoDecryption ()
리턴타입	없음 (void)
입력 파라미터	Boolean) – true, false
설명	암호화파일 자동 복호화

메소드명	run()
리턴타입	없음 (void)
입력 파라미터	없음 (void)
설명	업로드 시작

메소드명	setFileName()
리턴타입	없음 (void)
입력 파라미터	String
설명	업로드 파일명 리네임

메소드명	setDirectory ()
리턴타입	없음 (void)
입력 파라미터	String
설명	업로드 저장경로 지정

메소드명	InnorixCustomError ()
리턴타입	없음 (void)
입력 파라미터	InnorixCustomError.set(error Code, error Message, error Title, retry);
설명	커스텀에러 구성

메소드명	InnorixCustomValue ()
리턴타입	없음 (void)
입력 파라미터	InnorixCustomValue.set(parameter, value);
설명	커스텀밸류 구성

메소드명	setSaveFolderTree()
리턴타입	없음 (void)
입력 파라미터	Boolean) – true, false
설명	폴더첨부 시, 파일만 저장

D. 클라이언트 측 이벤트 목록

이벤트명	파라미터	설명
loadComplete		로딩 완료
beforeAddFile	# JSON Object <pre>{ "basePath": "C:\\\\data", //첨부경로(String) "controlId": "fileControl", //컨트롤ID(String) "filePath": "C:\\\\data\\\\blue.pdf", //전체경로(String) "fileSize": 9437184, //파일사이즈 (Number) / Byte "mode": "upload", //전송모드(String) "uploadUrl": "http://{Server}/upload.jsp", //업로드URL(String) }</pre>	파일 추가 전
afterAddFiles	# JSON Object <pre>[{ "basePath": "D:\\", // 첨부경로 (String) "controlId": "fileControl", // 컨트롤ID (String) "filePath": "D:\\Data.ppt", // 파일 전체경로 (String) "fileSize": 2506093, // 파일 사이즈(Number) / Byte "folderName": "", // 폴더명 (String) "id": "fda07096-....114fbe8ad", // 파일ID (String) "mode": "upload", // 첨부모드 (String) "printFileName": "Data.ppt ", // 출력파일명 (String) "rootName": "", // 폴더정보 (String) "rowID": "fda070....114fbe8ad ", // 파일리스트ID (String) "selected": true, // 파일선택여부 (Boolean) "transferType": "upload", // 전송유형 (String) }]</pre>	파일 추가 후

	<pre> "uniqueFileName": "950..404.dat", // 파일고유저장명 (String) "uploadUrl": "http://{Server}/upload.jsp" //업로드URL (String) }, { "downloadUrl": "http://{Server}/download.jsp?fileID=1", // 다운로드 URL (String) "fileSize": 1433885, // 파일사이즈 (Number) / Byte "id": "8d6f7747-...92edf7d6", // 파일ID (String) "mode": "download", // 첨부모드 (String) "printFileName": "INNORIX.pdf", // 출력파일명 (String) "rowID": "8d6f7747-...92edf7d6", // 파일리스트ID (String) "selected": true, // 파일선택여부 (Boolean) "sliceSize": 2002, // 파일조각사이즈(Number) / Byte "transferType": "download", // 전송타입 (String) "validate": true // 파일유효여부 (Boolean) }] </pre>	
removeFiles	<pre> # JSON Object [{ "basePath": "D:\\", // 첨부경로 (String) "controlId": "fileControl", // 컨트롤ID (String) "filePath": "D:\\Data.ppt", // 파일 전체경로 (String) "fileSize": 2506093, // 파일 사이즈(Number) / Byte "folderName": "", // 폴더명 (String) "id": "fda07096-....114fbe8ad", // 파일ID (String) "mode": "upload", // 첨부모드 (String) "printFileName": "Data.ppt ", // 출력파일명 (String) "rootName": "", // 폴더정보 (String) "rowID": "fda070....114fbe8ad ", // 파일리스트ID (String) "selected": true, // 파일선택여부 (Boolean) "transferType": "upload", // 전송유형 (String) "uniqueFileName": "950..404.dat", // 파일고유저장명 (String) "uploadUrl": "http://{Server}/upload.jsp" //업로드URL (String) }, { "downloadUrl": "http://{Server}/download.jsp?fileID=1", // 다운로드 URL (String) "fileSize": 1433885, // 파일사이즈 (Number) / Byte "id": "8d6f7747-...92edf7d6", // 파일ID (String) "mode": "download", // 첨부모드 (String) "printFileName": "INNORIX.pdf", // 출력파일명 (String) "rowID": "8d6f7747-...92edf7d6", // 파일리스트ID (String) "selected": true, // 파일선택여부 (Boolean) "sliceSize": 2097152, // 파일조각사이즈(Number) / Byte "transferType": "download", // 전송타입 (String) "validate": true // 파일유효여부 (Boolean) }] </pre>	파일 삭제 후

	<pre> }] </pre>	
beforeRemoveFiles	<pre> # JSON Object [{ "basePath":"D:\W", // 첨부경로 (String) "controlId":"fileControl", // 컨트롤ID (String) "filePath":"","D:\WData.ppt", // 파일 전체경로 (String) "fileSize":2506093, // 파일 사이즈(Number) / Byte "folderName":"", // 폴더명 (String) "id":"fda07096-....114fbe8ad", // 파일ID (String) "mode":"upload", // 첨부모드 (String) "printFileName":" Data.ppt ", // 출력파일명 (String) "rootName":"", // 폴더정보 (String) "rowID":" fda070....114fbe8ad ", // 파일리스트ID (String) "selected":true, // 파일선택여부 (Boolean) "transferType":"upload", // 전송유형 (String) "uniqueFileName":"950..404.dat", // 파일고유저장명 (String) "uploadUrl":"http://{Server}/upload.jsp" //업로드URL (String) }, { "downloadUrl ":"http://{Server}/download.jsp?fileID=1", // 다운로드 URL (String) "fileSize ":1433885, // 파일사이즈 (Number) / Byte "id":"8d6f7747-...92edf7d6", // 파일ID (String) "mode":"download", // 첨부모드 (String) "printFileName":"INNORIX.pdf", // 출력파일명 (String) "rowID":"8d6f7747-...92edf7d6", // 파일리스트ID (String) "selected":true, // 파일선택여부 (Boolean) "sliceSize":2097152, // 파일조각사이즈(Number) / Byte "transferType":"download", // 전송타입 (String) "validate":true // 파일유효여부 (Boolean) }] </pre>	파일 삭제 전 false 시 파일 삭제 안함
addFileError	<pre> # JSON Object [{ "basePath":"D:\W", // 첨부경로 (String) "controlId":"fileControl", // 컨트롤ID (String) "filePath":"","D:\WData.ppt", // 파일 전체경로 (String) "fileSize":2506093, // 파일 사이즈(Number) / Byte "mode":"upload", // 첨부모드 (String) "uploadUrl":"http://{Server}/upload.jsp", //업로드URL (String) } "message":"*.pdf file can not be attached.", //에러내용 (String) "*.ppt file can not be attached."] </pre>	파일 추가 에러

	<p>"The maximum limit of a single file is 1.00 MB."</p> <p>"The maximum number of files is 1."</p> <p>"The maximum limit of total files is 1.00 MB"</p> <p>"The same file is already attached."</p> <p>"type": "limitExtension" //제한사유 (String)</p> <p>"allowExtension"</p> <p>"maxFileSize"</p> <p>"maxFileCount"</p> <p>"maxTotalSize"</p> <p>"addDuplicateFile"</p> <p>]</p>	
onSelectRows	<p># JSON Object</p> <pre>[{ "basePath": "D:\\", // 첨부경로 (String) "controlId": "fileControl", // 컨트롤ID (String) "filePath": "D:\\Data.ppt", // 파일 전체경로 (String) "fileSize": 2506093, // 파일 사이즈(Number) / Byte "folderName": "", // 폴더명 (String) "id": "fda07096-....114fbe8ad", // 파일ID (String) "mode": "upload", // 첨부모드 (String) "printFileName": " Data.ppt ", // 출력파일명 (String) "rootName": "", // 폴더정보 (String) "rowID": " fda070....114fbe8ad ", // 파일리스트ID (String) "selected": true, // 파일선택여부 (Boolean) "transferType": "upload", // 전송유형 (String) "uniqueFileName": "950..404.dat", // 파일고유저장명 (String) "uploadUrl": "http://{Server}/upload.jsp" //업로드URL (String) }, { "downloadUrl": "http://{Server}/download.jsp?fileID=1", // 다운로드 URL (String) "fileSize": 1433885, // 파일사이즈 (Number) / Byte "id": "8d6f7747-...92edf7d6", // 파일ID (String) "mode": "download", // 첨부모드 (String) "printFileName": "INNORIX.pdf", // 출력파일명 (String) "rowID": "8d6f7747-...92edf7d6", // 파일리스트ID (String) "selected": true, // 파일선택여부 (Boolean) "sliceSize": 2097152, // 파일조각사이즈(Number) / Byte "transferType": "download", // 전송타입 (String) "validate": true // 파일유효여부 (Boolean) }]</pre>	파일 선택시
onUnSelectRows	<p># JSON Object</p> <pre>[{</pre>	파일 선택 해제 시

	<pre> "basePath":"D:\W", // 첨부경로 (String) "controlId":"fileControl", // 컨트롤ID (String) "filePath":"","D:\WData.ppt", // 파일 전체경로 (String) "fileSize":2506093, // 파일 사이즈(Number) / Byte "folderName":"", // 폴더명 (String) "id":"fda07096-....114fbe8ad", // 파일ID (String) "mode":"upload", // 첨부모드 (String) "printFileName":" Data.ppt ", // 출력파일명 (String) "rootName":"", // 폴더정보 (String) "rowID":" fda070....114fbe8ad ", // 파일리스트ID (String) "selected":false, // 파일선택여부 (Boolean) "transferType":"upload", // 전송유형 (String) "uniqueFileName":"950..404.dat", // 파일고유저장명 (String) "uploadUrl":"http://{Server}/upload.jsp" //업로드URL (String) }, { "downloadUrl ":"http://{Server}/download.jsp?fileID=1", // 다운로드 URL (String) "fileSize ":1433885, // 파일사이즈 (Number) / Byte "id":"8d6f7747-...92edf7d6", // 파일ID (String) "mode":"download", // 첨부모드 (String) "printFileName":"INNORIX.pdf", // 출력파일명 (String) "rowID":"8d6f7747-...92edf7d6", // 파일리스트ID (String) "selected":false, // 파일선택여부 (Boolean) "sliceSize":2097152, // 파일조각사이즈(Number) / Byte "transferType":"download", // 전송타입 (String) "validate":true // 파일유효여부 (Boolean) } } </pre>	
uploadStart	<pre> # JSON Object { "files":[// 업로드파일정보 (Array) { "basePath":"D:\W", // 첨부경로 (String) "controlId":"fileControl", // 컨트롤ID (String) "filePath":"","D:\WData.ppt", // 파일 전체경로 (String) "fileSize":2506093, // 파일 사이즈(Number) / Byte "folderName":"", // 폴더명 (String) "id":"fda07096-....114fbe8ad", // 파일ID (String) "mode":"upload", // 첨부모드 (String) "printFileName":" Data.ppt ", // 출력파일명 (String) "rootName":"", // 폴더정보 (String) "rowID":" fda070....114fbe8ad ", // 파일리스트ID (String) "selected":false, // 파일선택여부 (Boolean) "transferType":"upload", // 전송유형 (String) "uniqueFileName":"950..404.dat", // 파일고유저장명 (String) "uploadUrl":"http://{Server}/upload.jsp" //업로드URL (String) </pre>	업로드 시작

	<pre> },] "progress":0, // 진행상태 (Number) / % "retries":0, // 재시도횟수 (Number) / Times "speed":0, // 전송속도 (Number) / Byte/s "state":"Before", // 전송단계 (String) "totalSize":2506093, // 전체용량 (Number) / Byte "transferID":"d77-...29d", // 전송ID (String) "transferSize":0, // 전송용량 (Number) / Byte "type":"upload" // 전송타입 (String) } </pre>	
uploadComplete	<pre> # JSON Object { "files":[// 업로드파일정보 (Array) { "clientFileName":"Data.ppt", // 출력파일명 (String) "clientFilePath":"","D:₩Data.ppt", // 파일 전체경로 (String) "controlId":"fileControl", // 컨트롤ID (String) "basePath":"D:₩", // 첨부경로 (String) "customeValue":""," // 개발자정의값 (String) "fileSize":2506093, // 파일 사이즈(Number) / Byte "folderName":""," // 폴더명 (String) "fileState":"complete", // 파일전송단계 (String) "isFolder":false, // 폴더여부 (Boolean) "rootName":""," // 폴더정보 (String) "rowID":" fda070....114fbe8ad ", // 파일리스트ID (String) "serverFileName":"Data.dat", // 저장파일명 (sting) "serverFilePath":"C:/{{Server}}/Data.dat", // 저장경로 (String) "uploadUrl":"http://{{Server}}/upload.jsp" // 업로드URL (String) },] "progress": 100, // 진행상태 (Number) / % "retries":0, // 재시도횟수 (Number) "speed": 8584793, // 전송속도 (Number) / Byte/s "state":"Complete", // 전송단계 (String) "stausMessage":{" // 상태메시지 (Object) "errorCode":false, // 에리코드 (Boolean/String) "id":"transferCompleted" // 전송상태 (String) }, "totalSize":2506093, // 전체용량 (Number) / Byte "transferID":"d77-...29d", // 전송ID (String) "transferSize": 2506093, // 전송용량 (Number) / Byte "type":"upload" // 전송타입 (String) } </pre>	업로드 완료
uploadCancel	<pre> # JSON Object { </pre>	업로드 취소

	<pre> "files":[// 업로드파일정보 (Array) { "clientFileName":"Data.ppt", // 출력파일명 (String) "clientFilePath":""D:\WData.ppt"", // 파일 전체경로 (String) "controlId":"fileControl", // 컨트롤ID (String) "basePath":""D:\W", // 첨부경로 (String) "customeValue":"" // 개발자정의값 (String) "fileSize":2506093, // 파일 사이즈(Number) / Byte "folderName":""," // 폴더명 (String) "fileState":"wait", // 파일전송단계 (String) "isFolder":false, // 폴더여부 (Boolean) "rootName":""," // 폴더정보 (String) "rowID":"" fda070....114fbe8ad ", // 파일리스트ID (String) "serverFileName":"Data.dat", // 저장파일명 (sting) "serverFilePath":"C:/\{Server}/Data.dat", // 저장경로 (String) "uploadUrl":"http://\{Server}/upload.jsp" // 업로드URL (String) },] "progress": 20, // 진행상태 (Number) / % "retries":0, // 재시도횟수 (Number) "speed": 8584793, // 전송속도 (Number) / Byte/s "state":"Cancel", // 전송단계 (String) "stausMessage":{" // 상태메시지 (Object) "errorCode":false, // 에러코드 (Boolean/String) "id":"trnasferring" // 전송상태 (String) }, "totalSize":2506093, // 전체용량 (Number) / Byte "transferID":"d77-...29d", // 전송ID (String) "transferSize": 1203022, // 전송용량 (Number) / Byte "type":"upload" // 전송타입 (String) } </pre>	
uploadError	<pre> # JSON Object { "files":[// 업로드파일정보 (Array) { "clientFileName":"Data.ppt", // 출력파일명 (String) "clientFilePath":""D:\WData.ppt"", // 파일 전체경로 (String) "controlId":"fileControl", // 컨트롤ID (String) "basePath":""D:\W", // 첨부경로 (String) "customeValue":"" // 개발자정의값 (String) "fileSize":2506093, // 파일 사이즈(Number) / Byte "folderName":""," // 폴더명 (String) "fileState":"wait", // 파일전송단계 (String) "isFolder":false, // 폴더여부 (Boolean) "rootName":""," // 폴더정보 (String) "rowID":"" fda070....114fbe8ad ", // 파일리스트ID (String) "serverFileName":"Data.dat", // 저장파일명 (sting) </pre>	업로드 에러

	<pre> "serverFilePath": "C:/Server/Data.dat", // 저장경로 (String) "uploadUrl": "http://Server/upload.jsp" // 업로드 URL (String) },] "progress": 20, // 진행상태 (Number) / % "retries": 3, // 재시도횟수 (Number) "speed": 8584793, // 전송속도 (Number) / Byte/s "state": "Error", // 전송단계 (String) "statusMessage": { // 상태메시지 (Object) "errorCode": "404", // 에러코드 (String) "id": "errorSocket" // 전송상태 (String) }, "totalSize": 2506093, // 전체용량 (Number) / Byte "transferID": "d77-...29d", // 전송ID (String) "transferSize": 1203022, // 전송용량 (Number) / Byte "type": "upload" // 전송타입 (String) } </pre>	
downloadStart	<pre> # JSON Object { "files": [// 다운로드파일정보 (Array) { "downloadUrl": "http://Server/download.jsp?fileID=1", // 다운로드 URL (String) "fileSize": 1433885, // 파일사이즈 (Number) / Byte "id": "8d6f7747-...92edf7d6", // 파일ID (String) "mode": "download", // 첨부모드 (String) "printFileName": "INNORIX.pdf", // 출력파일명 (String) "rowID": "8d6f7747-...92edf7d6", // 파일리스트ID (String) "selected": false, // 파일선택여부 (Boolean) "sliceSize": 2097152, // 파일조각사이즈 (Number) / Byte "transferType": "download", // 전송타입 (String) "validate": true // 파일유효여부 (Boolean) },], "progress": 0, // 진행상태 (Number) / % "retries": 0, // 재시도횟수 (Number) / Times "speed": 0, // 전송속도 (Number) / Byte/s "state": "Before", // 전송단계 (String) "totalSize": 1433885, // 전체용량 (Number) / Byte "transferID": "d77-...29d", // 전송ID (String) "transferSize": 0, // 전송용량 (Number) / Byte "type": "download" // 전송타입 (String) } </pre>	다운로드 시작
downloadComplete	<pre> # JSON Object { "files": [// 다운로드파일정보 (Array) </pre>	다운로드 완료

	<pre> { "controlId":"fileControl" // 컨트롤ID (String) "fileName":"INNORIX.pdf", // 파일명 (String) "filePath":"C:\download" // 저장경로 (String) "fileState":"complete", // 파일전송단계 (String) "fileURL":" http://{Server}/download.jsp?fileID=1", // 다운로드 URL (String) "isFolder":false, // 폴더여부 (Boolean) "origFileSize":1433885, // 원본파일사이즈(Number) / Byte "rowID":"8d6f7747-...92edf7d6", // 파일리스트ID (String) "saveFileSize":1433885, // 저장파일사이즈 (Number) / Byte },] "progress":100, // 진행상태 (Number) / % "retries":0, // 재시도횟수 (Number) / Times "speed": 11029884, // 전송속도 (Number) / Byte/s "state":"Complete", // 전송단계 (String) "stausMessage":{" // 상태메시지 (Object) "errorCode":false, // 에리코드 (Boolean/String) "id":"trnasferring" // 전송상태 (String) }, "totalSize":1433885, // 전체용량 (Number) / Byte "transferID":"d77-...29d", // 전송ID (String) "transferSize":1433885, // 전송용량 (Number) / Byte "type":"download" // 전송타입 (String) } </pre>	
downloadCancel	<pre> # JSON Object { "files":[// 다운로드파일정보 (Array) { "controlId":"fileControl" // 컨트롤ID (String) "fileName":"INNORIX.pdf", // 파일명 (String) "filePath":"C:\download" // 저장경로 (String) "fileState":"wait", // 파일전송단계 (String) "fileURL":" http://{Server}/download.jsp?fileID=1", // 다운로드 URL (String) "isFolder":false, // 폴더여부 (Boolean) "origFileSize":1433885, // 원본파일사이즈(Number) / Byte "rowID":"8d6f7747-...92edf7d6", // 파일리스트ID (String) "saveFileSize":1433885, // 저장파일사이즈 (Number) / Byte },] "progress":50, // 진행상태 (Number) / % "retries":0, // 재시도횟수 (Number) / Times "speed": 11029884, // 전송속도 (Number) / Byte/s </pre>	다운로드 취소

	<pre> "state": "Cancel", "stausMessage": { "errorCode": false, "id": "trnsferring" }, "totalSize": 1433885, "transferID": "d77-...29d", "transferSize": 72563, "type": "download" </pre>	<pre> // 전송단계 (String) // 상태메시지 (Object) // 에러코드 (Boolean/String) // 전송상태 (String) // 전체용량 (Number) / Byte // 전송ID (String) // 전송용량 (Number) / Byte // 전송타입 (String) </pre>	
installPopupBlocked			install팝업모드 사용 시, 브라우저 팝업설정체크
dropzoneShow			드롭존 활성화 시
dropzoneHide			드롭존 비활성화 시
agentReady			인스톨완료 이벤트
onDbClickRows			더블클릭 시

E. 서버 측 이벤트 목록

이벤트명	파라미터	설명
getFileInfo	<pre> _action // 전송액션 플래그 _origin_filename // 원본파일명 _filesize // 파일사이즈 _folder // 하위폴더정보 _clientpath // 첨부파일 클라이언트 경로 _compressed // 압축여부 _rootPath // 최상위경로 _subdir // 서브디렉토리 경로 _encrypt // 암호화 여부 _transferId // 전송 ID _slice_transfer // 중계전송 여부 _duplicationFile // 중복파일처리 _empty_folder // 빈폴더존재 여부 </pre>	업로드 파일정보 획득
attachFile	<pre> _action // 전송액션 플래그 _origin_filename // 원본파일명 _new_filename // 저장파일명 </pre>	업로드 시작

	_filesize // 파일사이즈 _folder // 하위폴더정보 _clientpath // 첨부파일 클라이언트 경로 _serverpath // 첨부파일 서버저장 경로 _compressed // 압축여부 _rootPath // 최상위경로 _subdir // 서브디렉토리 경로 _encrypt // 암호화 여부 _transferId // 전송 ID _slice_transfer // 중계전송 여부 _duplicationFile // 중복파일처리 _empty_folder // 빈폴더존재 여부 _cookie // 쿠키정보 _start_offset // 조각시작점 _end_offset // 조각끝점 _orig_start_offset // 전송시점 기준 시작점	
attachFileComplete	_action // 전송액션 플래그 _origin_filename // 원본파일명 _new_filename // 저장파일명 _filesize // 파일사이즈 _folder // 하위폴더정보 _filepath // 첨부파일 서버저장 경로 _compressed // 압축여부 _rootPath // 최상위경로 _subdir // 서브디렉토리 경로 _encrypt // 암호화 여부 _transferId // 전송 ID _slice_transfer // 중계전송 여부 _duplicationFile // 중복파일처리 _empty_folder // 빈폴더존재 여부 _isfolder // 폴더존재 여부 _check_integrity // 무결성체크여부 _integrity_crc32 // crc32 체크 _integrity_md5 // md5 체크 _merging // 머징 여부	모든 일 업로드 완료
DownloadStart		다운로드 시작
DownloadCancel		다운로드 취소
DownloadError		다운로드 에러
DownloadComplete		다운로드 완료

V. 활용

A. UI 구성

1) 컨트롤 리스트

a. 크기조절

제공드리는 컨트롤 옵션과 메소드를 통하여, 첨부파일 리스트를 출력할 컨트롤 리스트의 크기를 자유롭게 구성할 수 있습니다.

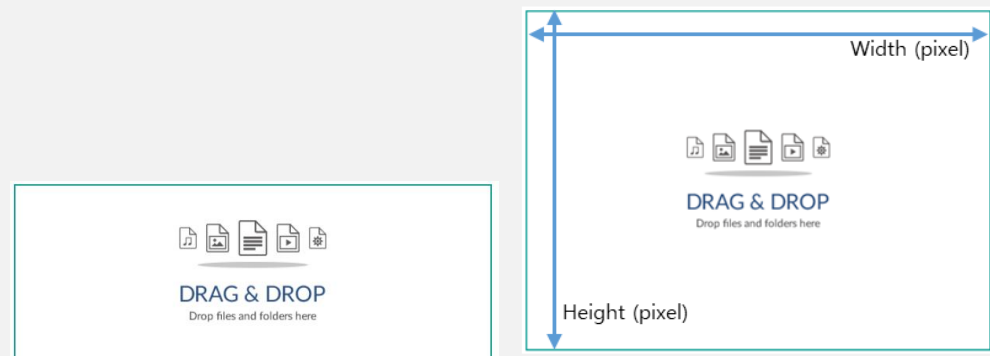
컨트롤 옵션을 통해 컨트롤 리스트 크기 조절

`control.option. width: 550 (pixel)`

`control.option.height : 200 (pixel)`

컨트롤 메소드를 통해 컨트롤 리스트 크기 조절

`control.setSize(550, 400);`



b. 배경이미지

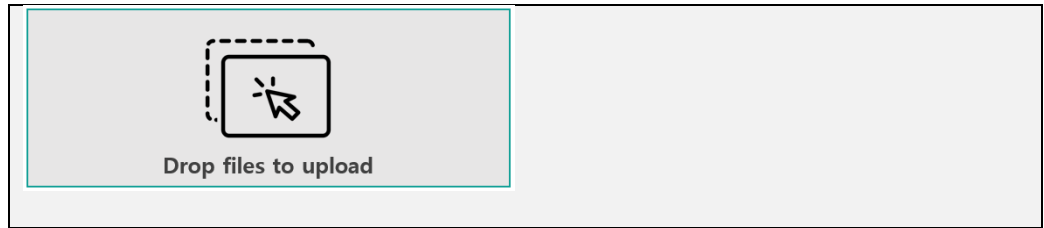
드래그드롭 유도 이미지는 `img/` 디렉토리 내에 위치하고 있습니다.

기본값이 아닌 커스터마이징 디자인을 원하는 경우, 해당 파일을 변경한 뒤, 파일명만 `dropzone.png`로 수정해주시면 반영되겠습니다.

기본 전송창



커스터마이징 디자인 적용 예시

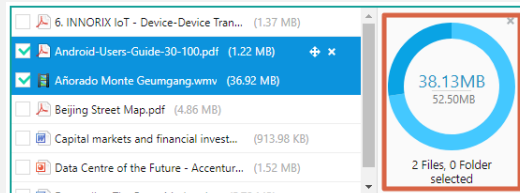


c. 헤더툴팁 / 그래프

전송창 내, 파일용량을 출력하는 그래프와 전체선택 및 선택삭제, 선택 다운로드 등의 기능버튼 및 컨텍스트 메뉴 등 다양한 방법의 기능을 컨트롤리스트 자체에서 제공드립니다. 별도의 버튼구성 없이도, 파일첨부 및 전송이 가능합니다.

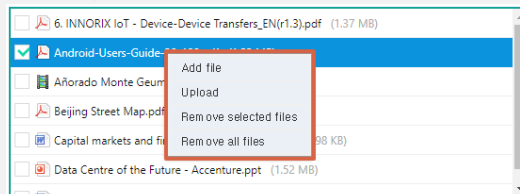
그래프

```
control.showGraph();
control.option.showGraph : true,
```



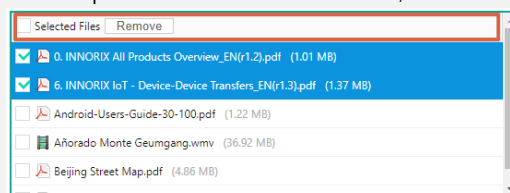
마우스 우클릭 컨텍스트 메뉴

```
control.option. useContextMenu : true,
```



리스트 헤더 메뉴

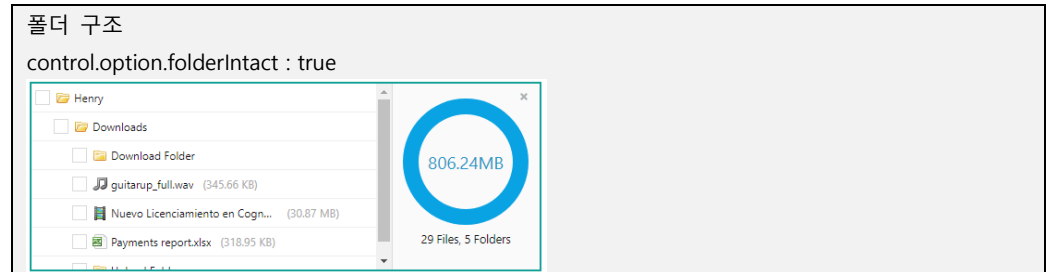
```
control.option. showFileListHeader : true,
```



d. 폴더 구조 출력

폴더 첨부 시, 폴더 트리 구조를 출력할 수 있으며 삭제 및 이동 등의 편집이 가능합니다.

다운로드 또한 폴더구조를 구성하여, 폴더 째 다운로드를 지원합니다.



2) 전송창

a. 전송 시작

업로드 또는 다운로드 버튼 없이도, 즉시 업로드 및 다운로드 동작을 진행할 수 있습니다.

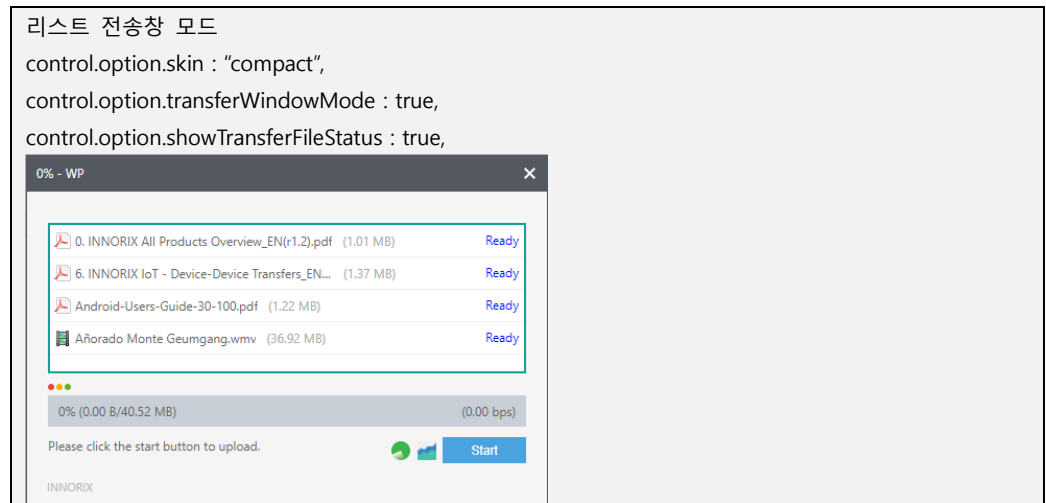
```
e.g) 첨부 시 바로 전송 진행
control.on('afterAddFiles', function(p){ //컨트롤 리스트로 파일 추가 후의 이벤트
control.upload(); // 업로드
});

e.g) 파일 선택 시, 다운로드
control.on('onSelectRows', function(p) { // 파일 선택 시
control.downloadSelectedFiles(); // 선택파일 다운로드
});

e.g) 다운로드 시, 경로 선택 창 출력
// 다운로드 경로선택 창 출력
function download() {
control.setDownloadPath(function(response){ // 경로선택
    if (response.result==true){
        control.downloadSelectedFiles(); // 선택파일을 선택경로로 다운로드
    }
});
}
```

b. 리스트 전송창 모드

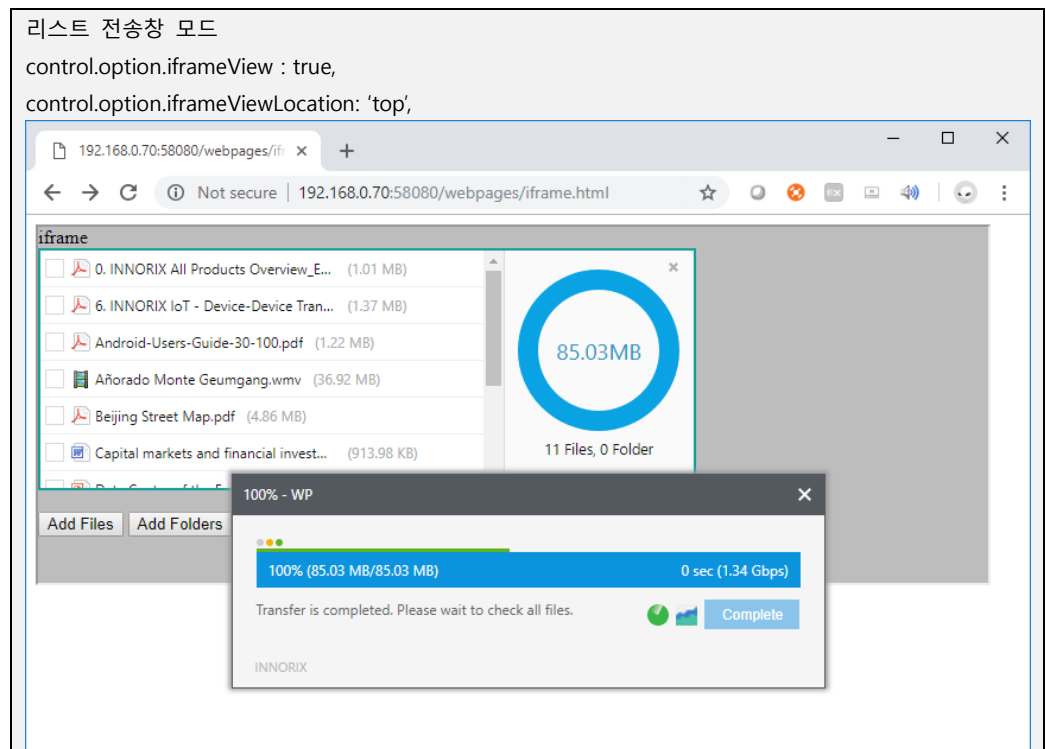
전송창에서 파일리스트를 출력하고, 전송 중인 파일들의 전송상태를 함께 표시할 수 있습니다.



3) iFrame

a. iframe 환경 구성

iframe 환경에서 프레임의 영역에 종속되지 않고 원하는 지점에서 전송창을 띄우고, 제어할 수 있습니다.



B. 커스텀 페이지 구성

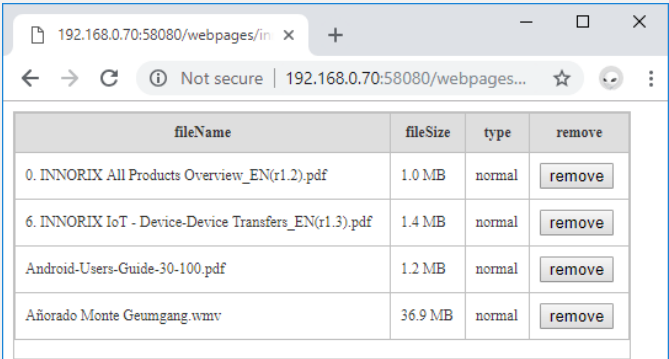
1) 커스텀 테이블

제공하는 기본 컨트롤리스트 형식이 아닌, 원하는 형태의 리스트스타일을 HTML객체를 통하여 자유롭게 구성이 가능합니다.

파일명, 파일사이즈와 같은 기본적인 정보 외에도 부가적인 정보를 함께 출력할 수 있습니다.

a. 커스텀 업로드 테이블

업로드 커스텀 테이블

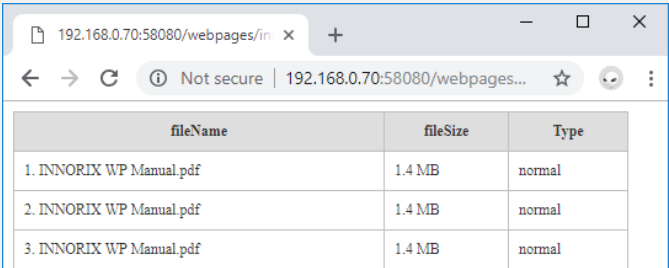


fileName	fileSize	type	remove
0. INNORIX All Products Overview_EN(r1.2).pdf	1.0 MB	normal	<button>remove</button>
6. INNORIX IoT - Device-Device Transfers_EN(r1.3).pdf	1.4 MB	normal	<button>remove</button>
Android-Users-Guide-30-100.pdf	1.2 MB	normal	<button>remove</button>
Añorado Monte Geumgang.wmv	36.9 MB	normal	<button>remove</button>

Add Files Upload

b. 커스텀 다운로드 테이블

다운로드 커스텀 테이블



fileName	fileSize	Type
1. INNORIX WP Manual.pdf	1.4 MB	normal
2. INNORIX WP Manual.pdf	1.4 MB	normal
3. INNORIX WP Manual.pdf	1.4 MB	normal

Download

2) 커스텀 드롭존

원하는 지점에 파일첨부가 가능한 드래그 드롭존을 구성할 수 있으며, 컨트롤리스트에 제약 없이 구성이 가능합니다.

a. 커스텀 드롭존 설정

드래그 드롭존 설정

// 특정 html 객체를 파일추가용 DropZone 으로 활성화
jQuery("#dropZone").on({
 "dragenter": function(jQueryevent) {
 control.setDropZone(jQueryevent, this);
 }
});

192.168.0.70:58080/webpages/in x +
← → ↻ ⓘ Not secure | 192.168.0.70:58080/we... ☆ ⚙ ⚠ 🔍

Drop files and folders here

☐ Añorado Monte Geumgang.wmv (36.92 MB)

☐ Beijing Street Map.pdf (4.86 MB)

☐ Capital markets and financial investment.docx (913.98 KB)

☐ Data Centre of the Future - Accenture.ppt (1.52 MB)

☐ 0. INNORIX All Products Overview_EN(r1.2).pdf (1.01 MB)

☐ 6. INNORIX IoT - Device-Device Transfers_EN(r1.3).pdf (1.37 MB)

Upload

3) 커스텀 에러

HTTP 상태에러 외의 시스템 구성과 맞지 않는 전송이 진행되는 경우, 커스텀 에러를 지정할 수 있으며, 에러코드와 에러타이틀, 에러메시지, 재전송여부에 대한 구성을 지원합니다.

커스텀 에러 설정

```
InnorixCustomError customError= new InnorixCustomError(Response); // 커스텀 에러 지정
customError.set("1006", "PathTooLong", "The path is too long.", false);
customError.run(); // 커스텀 에러 실행
```

에러 발생 시 해당 파일 건너뛰기 설정

```
control.option. skipErrorFile: true, // 에러 발생 시 skip
```

에러 발생 시 재시도 및 전송창 종료 설정

```
customError.set("1006", "PathTooLong", "The path is too long.", false); //에러발생 시 재시도 여부
control.on('uploadError', function(p){ // 전송 중 에러발생 이벤트
    setTimeout(function(){
        control.closeTransferWindow(); // 전송창 닫음
    }, 3000);
}
```

4) 커스텀 밸류

서버에서 파일처리 시 임의의 데이터를 추가 또는 업데이트를 할 수 있도록 커스텀밸류 구성을 지원합니다.

커스텀 밸류 설정

```
if (_action.equals("getFileInfo")){ //파일 별 전송 시작 지점
    uploader.setCustomValue("_test", _orig_filename + "_" + _filesize); // 커스텀 밸류 설정
}
```

5) 커스텀 포스트데이터

전송 시 파일 뿐만이 아닌, 임의의 데이터를 함께 전송하여 서버에서 처리할 수 있도록 구성이 가능합니다. 이러한 임의의 데이터는 한 전송 트랜잭션 당 또는 개별 파일마다 추가가 가능합니다.

포스트데이터 설정

```
var postObj = new Object(); // postdata 를 전달할 object 객체 생성
var fileId = "1234";
postObj_id = fileId; // object 에 각 id 할당
control.setPostData(postObj); // postdata 로 값 저장
```

파일당 포스트데이터 설정

```
var fileInfo = control.getAllFiles(); // 파일정보 추출
for (i=0; i<fileInfo.length; ++i){
    var postObj = new Object(); // postdata 를 전달할 object 객체 생성
    var fileId = fileInfo[i].id; // 각 파일 index 의 id 추출
    postObj_id = fileId; // object 에 각 id 할당
    control.setFilePostDataByIndex(i, postObj); // postdata 로 값 저장
}
```

C. 부가기능

1) MRT 연동

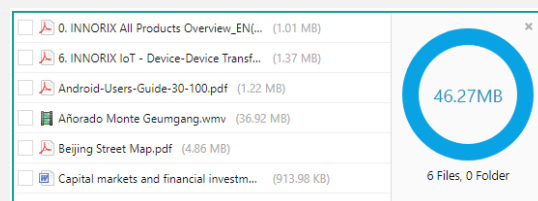
INNORIX WP의 전송내용을 MRT와 연동하여, 관리자는 MRT를 통해 실시간 전송 내용과, 전송 내역, 상태를 모니터링할 수 있습니다.

이러한 모니터링은 각 전송구간에 대한 분석을 도우며, 에러 발생 시 빠르고 효과적으로 대응할 수 있습니다.

MRT 연동 설정

```
control.option.monitorServer : "http://{ServerIP:Port}/dispatch/mt/transfer", //MRT URL
control.option.url: 'http://{ServerIP:Port}', // INNORIX Platform URL
control.option.custom:{ storageUrl: "http://{StorageIP:Port}"}, //파일 저장 경로 URL
```

위와 같이 MRT가 설치된 서버의 URL과 MRT URL, 저장경로를 커스텀값으로 리턴하여, MRT 서버에서 자료조회가 가능합니다.



File ID	Filename	Size	Transfer Start Time	Transfer End Time
ad49806e-128a-49e0-a4a7-c2a6b3f09f1b	0. INNORIX All Products Overview_EN(1.2).pdf	1.01 MB	21 Sep 2018 10:11:03	21 Sep 2018 10:11:04

Product	Origin	Destination	# of Files	Total Size	Status	Cancel	Error	Transfer Date
WP	192.168.0.70	192.168.0.70	6	46.27 MB	Completed	0	0	21 Sep 2018 10:11:03

File ID	Filename	Size	Transfer Start Time	Transfer End Time
ad49806e-128a-49e0-a4a7-c2a6b3f09f1b	0. INNORIX All Products Overview_EN(1.2).pdf	1.01 MB	21 Sep 2018 10:11:03	21 Sep 2018 10:11:04
19b98f6f-6a61-49e0-a4a7-c2a6b3f09f1b	Android-Users-Guide-30-100.pdf	1.22 MB	21 Sep 2018 10:11:03	21 Sep 2018 10:11:04
96a2d7c2-c7f7-49e0-a4a7-c2a6b3f09f1b	6. INNORIX IoT - Device-Device Transfers_EN(1.3).pdf	1.37 MB	21 Sep 2018 10:11:03	21 Sep 2018 10:11:04
29067fe7-44ea-49e0-a4a7-c2a6b3f09f1b	Capital markets and financial investment.docx	913.98 KB	21 Sep 2018 10:11:04	21 Sep 2018 10:11:04
0f2727b6-a3af-49e0-a4a7-c2a6b3f09f1b	Beijing Street Map.pdf	4.86 MB	21 Sep 2018 10:11:04	21 Sep 2018 10:11:04
aaf38c1f-c8de-49e0-a4a7-c2a6b3f09f1b	Aforado Monte Geumgang.wmv	36.92 MB	21 Sep 2018 10:11:04	21 Sep 2018 10:11:04

2) QR코드 생성

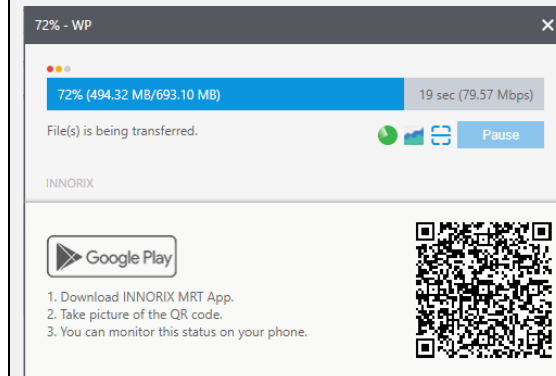
사용자는 자신의 전송이 진행되는 동안, PC 앞에서 전송을 지켜볼 필요가 없습니다.

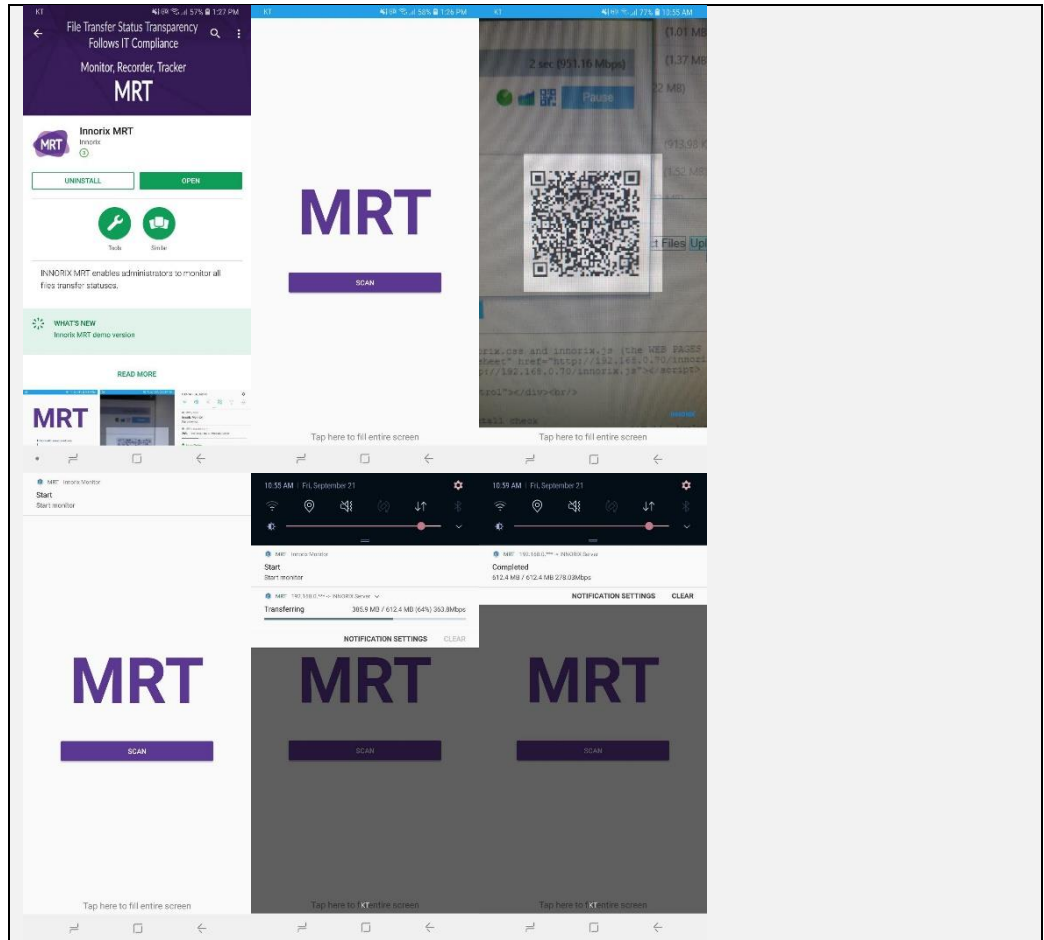
INNORIX에서 제공하는 INNORIX MRT Application을 통하여, 지금 전송하는 항목을 실시간으로 스마트폰을 통해 확인이 가능하며, 전송 완료, 에러 등 상태에 대해 확인할 수 있습니다.

MRT 연동 설정

control.option.showQrCode : true,

전송창에 활성화 된 QR코드를 통해, 스마트폰에서 전송상태를 확인할 수 있습니다.

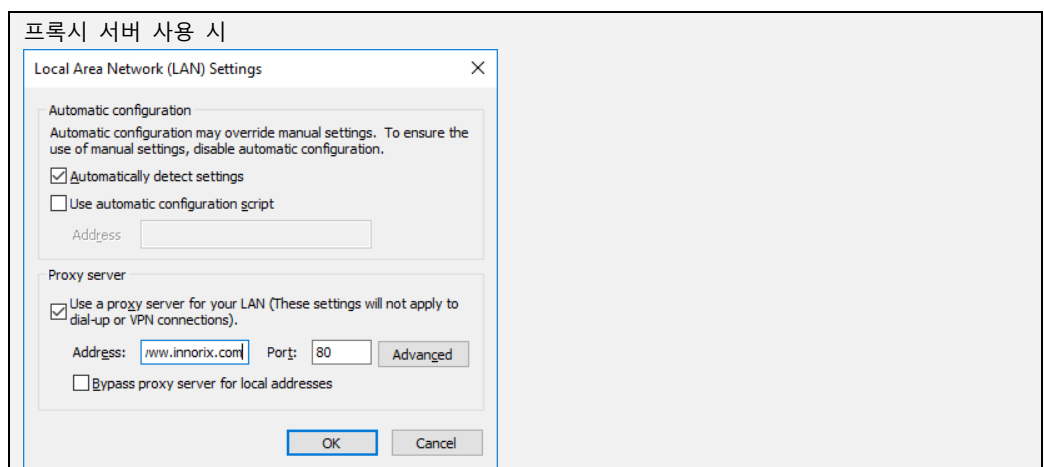




3) 프록시 환경 설정

단순한 네트워크 환경이 아닌, 프록시로 구성된 사내 네트워크에서는 다양한 정책과 제약으로 인하여 내부 사용자 간의 통신뿐만이 아닌, 외부 사용자와의 통신에도 제약이 있을 수 있으며, 이를 극복하기 위해 복잡한 예외처리를 진행해야 합니다.

INNORIX WP는 이러한 환경에서 간단한 설정만으로 프록시 서버에 자동으로 등록 또는 우회하여 전송할 수 있습니다.



```
control.option.useProxy : true,
```

파일 전송에 대한 요청/응답이 프록시 서버를 거치지 않도록 예외처리가 가능합니다.

```
innorix.js/
```

```
var WHITELIST = [www.innorix.com];
```

4) 파일서버 연동

해킹의 위협과 보안의 중요성으로 웹 서버와 파일서버를 분리하여 운영하는 경우에도, INNORIX WP는 전송데이터는 웹서버로, 파일은 파일서버로 저장할 수 있습니다.

업로드 경로 설정

upload.jsp 내에서 업로드 경로를 구성할 수 있습니다.

```
// 업로드 객체
```

```
new InnorixUpload(request, response, maxPostSize, directory);
```

```
// 저장 경로 구성
```

```
String directory = "/innorix/data";
```

```
// 동적으로 저장경로 변경 시
```

```
uploader.setDirectory(directory + "/" + name); // 저장경로 별도 지정
```

업로드 후 move 동작 설정

업로드 후, webServer가 아닌 다른 서버로 파일을 이동시킬 수 있습니다.

upload.html 완료 이벤트 구성

```
// 업로드 완료 이벤트
```

```
control.on('uploadComplete', function (p) {
```

```
    console.log('uploadComplete: event fire ');
```

```
    // 전송 완료 정보에서 파일 정보를 move.jsp 로 전송
```

```
    var files = JSON.stringify(p.files);
```

```
    jQuery.ajax({
```

```
        url: "move.jsp",
```

```
        type: 'post',
```

```
        data: {'files' : files}
```

```
    }).done(function (response) {
```

```
    }).fail(function () {
```

```
        console.log('move.jsp fail');
```

```
});
```

move.jsp 별도 구성

```
<%@ page import="java.io.*" %>
```

```
<%@ page import="java.nio.channels.FileChannel" %>
```

```
<%@ page import="org.json.JSONObject" %>
```

```
<%@ page import="org.json.JSONArray" %>
```

```

<%@ page import="org.json.JSONException" %>
<%@ page import="java.util.StringTokenizer" %>

<%
// 파일을 이동시킬 폴더 설정
String storagePath = "WWW192.168.0.70WWpublicWWuploadfileWWtestWW";
request.setCharacterEncoding("UTF-8");
java.util.Enumeration names = request.getParameterNames();

while (names.hasMoreElements()) {
    String name = (String)names.nextElement();
    String values[] = request.getParameterValues(name);

    // 전송한 파일 정보를 알아옴
    if (name.equals("files") && values != null)
    {
        JSONArray jsonFiles = new JSONArray(values[0]);
        for(int index = 0; index < jsonFiles.length(); ++index)
        {
            JSONObject jsonFileInfo = jsonFiles.getJSONObject(index);
            // 파일이 서버에 저장되어있는 경로와 파일 이름을 알아옴
            String serverFilePath = jsonFileInfo.getString("serverFilePath");
            String serverFileName = jsonFileInfo.getString("serverFileName");

            // 서버에 저장되어있는 파일을 storagePath 로 이동시킴
            File serverFile = new File(serverFilePath);
            if (serverFile.exists())
            {
                if (createFolder(storagePath))
                {
                    File storageFile = new File(storagePath + serverFileName);
                    // 파일이 존재하는 경우 리네임이 되지 않으므로 파일을 제거
                    if (storageFile.exists())
                    {
                        storageFile.delete();
                    }
                    boolean isMoved = serverFile.renameTo(storageFile);
                    // renameTo 로 리네임처리를 하지 못 한 경우 직접 파일을 이동시키도록
                    if (isMoved == false)
                    {
                        moveToFile(serverFile, storageFile);
                    }
                }
            }
        }
    }
}
}
}

```

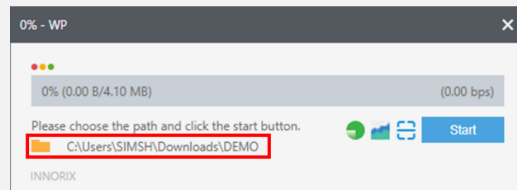

이하 생략

다운로드 경로 설정

1. 다운로드 경로 변경

전송창 내, 버튼아이콘을 클릭, 경로선택 다이얼로그를 출력할 수 있습니다.

이 때 처음 출력되는 경로는 사용자 기본 Download폴더 또는 최근 다운로드한 경로가 출력되었습니다.

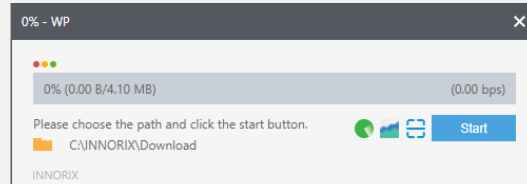


2. 기본 다운로드 경로 변경

클라이언트 옵션을 통하여, 최초 구성되는 다운로드 경로를 변경할 수 있습니다.

이 때, 경로가 존재하지 않을 경우, 해당 위치에 경로를 자동으로 생성합니다.

control.option.downloadPath : "C:\WINNORIX\Download"



3. 사용자 선택 다운로드 경로 변경

다운로드 시작 전, 자동으로 경로변경 다이얼로그를 출력,

사용자가 직접 다운로드 경로를 확인한 뒤 전송을 진행할 수 있습니다.

```
// 다운로드 folderDialog 출력 (선택한 경로를 response.result 로 리턴)
control.setDownloadPath(function(response){
    if (response.result==true){ // response.result 가 유효한 경로인 경우,
        control.download(); // 해당 경로에 다운로드 진행
    }
});
```

