

# Guideline: StorageBox Utility

## (Java & Php) version 1.2

### StorageBox Utility Class

[ Provided with fincom.jar ]

```
package com.finlogic.util.sbutility;

public class StorageBoxImpl
{
    ...
    // Protected Parameterized Constructor
    protected StorageBoxImpl(int projectId)

    // Protected Methods
    protected String getFile(String sbFilePath) throws IllegalArgumentException,
        UnsupportedEncodingException, IOException

    protected void putFile(PutFileBean putFileBean) throws Exception

    protected boolean deleteFile(String sbFilePath) throws Exception

    protected boolean isFileExist(String sbFilePath) throws Exception

    protected List<SBFileBean> getList(String sbDirPath) throws IllegalArgumentException,
        UnsupportedEncodingException, IOException

    ...
}
```

## How To Use

### Dependencies:

Java - get updated **fincom.jar** and set in classpath.  
Php - get FinFunction.inc (updated),  
StorageBoxImpl.inc,  
PutFileBean.inc,  
SBFileBean.inc from samba share - devweb3.nj/html/sbutility

### Property File Update

Add below required properties:

[ to use localdisk as storagebox - developer's machine]

```
sbgetagent=file:///opt/application_storage/storage_box/  
sbputagent=file:///opt/application_storage/storage_box/  
sbdeleteagent=file:///opt/application_storage/storage_box/  
sbisexistagent=file:///opt/application_storage/storage_box/  
sblistagent=file:///opt/application_storage/storage_box/
```

**Note:**

/opt/application\_storage/storage\_box/ path in above property interprets as local storagebox path

Create below directory if not exists.

/opt/application\_storage/storage\_box/

[ to use dev storagebox]

```
sbgetagent=http://192.168.3.80:8081  
sbputagent=http://192.168.3.80:8081  
sbdeleteagent=http://192.168.3.80:8081  
sbisexistagent=http://192.168.3.80:8081  
sblistagent=http://192.168.3.80:8081
```

**Note:**

Confirm below property exists

tempfiles\_path=/opt/application\_storage/temp\_files/

Create below directory if not exists

/opt/application\_storage/temp\_files/cache/

As shown below, **create** a **class** once for a project and use the same while require any operation with storagebox.

**JAVA**

```
import com.finlogic.util.sbutility.PutFileBean;
import com.finlogic.util.sbutility.StorageBoxImpl;
import java.io.IOException;
import java.io.UnsupportedEncodingException;

public class SBCommonOperation extends StorageBoxImpl
{
    private static SBCommonOperation sbCommon = new sbCommonOperation();

    private SBCommonOperation()
    {
        super(0); // Must- provide project id
    }

    public static SBCommonOperation getSBCommonOperation()
    {
        return sbCommon;
    }

    @Override
    public String getFile(String sbFilePath) throws IllegalArgumentException,
    UnsupportedEncodingException, IOException
    {
        return super.getFile(sbFilePath);
    }

    @Override
    public void putFile(PutFileBean putFileBean) throws Exception
    {
        super.putFile(putFileBean);
    }

    @Override
    public boolean deleteFile(String sbFilePath) throws Exception
    {
        return super.deleteFile(sbFilePath);
    }

    @Override
    public boolean isFileExist(String sbFilePath) throws Exception
    {
        return super.isFileExist(sbFilePath);
    }

    @Override
    protected List<SBFileBean> getList(String sbDirPath) throws IllegalArgumentException,
    UnsupportedEncodingException, IOException
    {
        return super.getList(sbDirPath);
    }
}
```

- Get instance of SBCommonOperation class  
SBCommonOperation sbCommon = SBCommonOperation.getsbCommonOperation();
- Available methods-  
sbCommon.getFile(), sbCommon.putFile(), sbCommon.deleteFile(), sbCommon.isFileExist(), sbCommon.getList()

## PHP

```
<?php
include '/var/www/html/sbutility/StorageBoxImpl.inc';
include '/var/www/html/sbutility/PutFileBean.php';

class SBCommonOperation extends StorageBoxImpl {

    private static $sbCommon;

    protected function __construct() {
        parent::__construct(0); // Must- provide project id
    }

    public function getSbCommonOperation() {
        if(!is_object(self::$sbCommon)) {
            self::$sbCommon = new SBCommonOperation();
        }

        return self::$sbCommon;
    }

    public function getFile($filepath) {
        return parent::getFile($filepath);
    }

    public function putFile($putFileBn) {
        return parent::putFile($putFileBn);
    }

    public function deleteFile($sbFilePath) {
        return parent::deleteFile($sbFilePath);
    }

    public function isFileExist($sbFilePath) {
        return parent::isFileExist($sbFilePath);
    }

    public function getList($sbDirPath) {
        return parent::getList($sbDirPath);
    }
}
?>
```

- Get instance of SBCommonOperation class  
\$sbCommon = SBCommonOperation::getsbCommonOperation();
- Available methods-  
sbCommon->getFile(), sbCommon->putFile(), sbCommon->deleteFile(), sbCommon->isFileExist(), sbCommon->getList()

## Functionality

### Methods

#### 1. getFile

##### Input Parameter

Name	Datatype	Description	Example
sbFilePath	String	provide relative path of storagebox file	sbFilePath="TRADING/capitalmarket/basket_order/0605/ORD060515182201.xls"

##### Output Parameter

Name	Datatype	Description	Example
localFilePath	String	returns absolute local file path that received from storagebox	/opt/application_storage/temp_files/cache/cache-22098894664517/ORD060515182201.xls

##### Validations

Sr No.	Exception	Description	Example
1.	IllegalArgumentException	provided null or empty input parameter- sbFilePath	sbFilePath= null sbFilePath=" "
2.	IllegalArgumentException	when provided sbFilePath is not a relative path	sbDirPath="/opt/application_storage/storage_box/TRADING/capitalmarket/basket_order/ORD060515182201.xls"
3.	IllegalArgumentException	when requested file is not a file	sbFilePath="TRADING/capitalmarket/basket_order/0605/"
4.	FileNotFoundException	when requested storagebox file is not available on storagebox	-
5.	IOException	when requested empty file	TRADING/capitalmarket/basket_order/ORD060515182201.xls is empty file with size 0.
6.	IllegalState	storagebox is not available for get operation	when manually make storagebox unavailable for maintenance.

## 2. putFile

### Input Parameter

Name	Datatype	Description	Example
putFileBean	PutFileBean	provide PutFileBean with mandatory values	-

### PutFileBean

Name	Datatype	Description	Example
sourceFile	File	set file objects of absolute local filepath need to be uploaded on storagebox	putFileBean.setSourceFile(new File("/opt/application_storage/temp_files/trading/basket_order/0605/ORD060515182201.xls"))
sbDestPath	String	set relative destination storagebox filepath to upload file at.	TRADING/capitalmarket/basket_order/0605/123BQ123.xls
removeFromSource	boolean	true – remove source file after successful fileupload at storagebox  false – do not remove source file after successful fileupload at storagebox (Default: true)	true / false
overwrite	boolean	true – overwrite file on storagebox if exist  false – do not overwrite file on storagebox if exist (Default: true)	true / false

### Output Parameter

Name	Datatype	Description	Example
-	-	-	-

### Validations

Sr No.	Exception	Description	Example
1.	IllegalArgument	source file object is not provided while requesting file upload	-
2.	IllegalArgument	when source filepath is not start with /opt/application_storage/temp_files/	putFileBean.setSourceFile(new File("/opt/apache-tomcat1/webapps/trading/basket_order/0605/ORD060515182201.xls"))
3.	IllegalArgument	provided sourcefile object is not a file	putFileBean.setSourceFile(new File("/opt/apache-tomcat1/webapps/trading/basket_order/0605"))
4.	IllegalArgument	provided sourcefile object refers to empty file that requested to upload	-
5.	IllegalArgument	provided null or empty destination storagebox file path	putFileBean.setSbDestPath("");

6.	IllegalArgument	when provided sbDestPath is not a relative path	sbDirPath="/opt/application_storage/storage_box/TRADING/capitalmarket/basket_order/ORD060515182201.xls"
7.	IllegalArgument	provided destination filename that is not valid. Not valid if contains special characters.	sbDirPath="/opt/application_storage/storage_box/TRADING/capitalmarket/basket_order/ORD\$060515182201.xls"
8.	IllegalArgument	provided sbDestPath is not a file	putFileBean.setSbDestPath("TRADING/capitalmarket/basket_order/0605.2015");
9.	IllegalArgument	if root dir on storagebox not exist. for shown example, TRADING1 is not exist on Storagebox. if subfolders are not exist, It will be created.	putFileBean.setSbDestPath("TRADING1/capitalmarket/basket_order/0605.2015");
10.	IllegalArgument	if subfolders are not exist it will be created only if their name is valid. Dir name is not valid if it contains special characters.	putFileBean.setSbDestPath("TRADING/capitalmarket/XY~Z/PQ\$12/123.xls");
11.	IllegalState	storagebox is not available for put operation	when manually make storagebox unavailable for maintenance purpose

### 3. deleteFile

#### Input Parameter

Name	Datatype	Description	Example
sbFilePath	String	provide relative path of storagebox file	sbFilePath="TRADING/capitalmarket/basket_order/0605/ORD060515182201.xls"

#### Output Parameter

Name	Datatype	Description	Example
status	boolean	returns true / false true – if deleted false – if not deleted	true / false

#### Validations

Sr No.	Exception	Description	Example
1.	IllegalArgument	provided null or empty input parameter- sbFilePath	sbFilePath= null sbFilePath=" "
2.	IllegalArgument	when provided sbFilePath is not a relative path	sbDirPath="/opt/application_storage/storage_box/TRADING/capitalmarket/basket_order/ORD060515182201.xls"
3.	IllegalArgument	requested to delete file on storagebox that is not exist.	-
4.	IllegalArgument	requested to delete file on storagebox- provided filepath that does not refers to a file.	sbFilePath="TRADING/capitalmarket/basket_order/0605/"
5.	IllegalState	storagebox is not available for deleteFile operation	when manually make storagebox unavailable for maintenance purpose



#### 4. isFileExist

##### Input Parameter

Name	Datatype	Description	Example
sbFilePath	String	provide relative path of storagebox file	sbFilePath="TRADING/capitalmarket/basket_order/0605/ORD060515182201.xls"

##### Output Parameter

Name	Datatype	Description	Example
status	boolean	Returns true / false true – if exist on storagebox false – if not exist on storagebox	true / false

##### Validations

Sr No.	Exception	Description	Example
1.	IllegalArgument	provided null or empty input parameter- sbFilePath	sbFilePath= null sbFilePath=" "
2.	IllegalArgument	when provided sbFilePath is not a relative path	sbDirPath="/opt/application_storage/storage_box/TRADING/capitalmarket/basket_order/ORD060515182201.xls"
3.	IllegalArgument	requested to check availability of a filepath that does not refers to a file.	sbFilePath="TRADING/capitalmarket/basket_order/0605/"
4.	IllegalState	storagebox is not available for isFileExist operation	when manually make storagebox unavailable for maintenance purpose

## 5. getList

- provides contents's detail of storagebox directory.

### Input Parameter

Name	Datatype	Description	Example
sbDirPath	String	provide relative path of storagebox directory.	sbDirPath="TRADING/capitalmarket"

### Output Parameter

Name	Datatype	Description	Example
List<SBFileBean>	SBFileBean	returns detail of contents underlying the specified sbDirPath directory.	returns List<SBFileBean> with x no. of SBFileBean objects. (x=no. of contents in Trading/capitalmarket)

### SBFileBean

Name	Datatype	Description	Example
sbFileName	String	content's relative path on storage box	TRADING/capitalmarket/basket_order
sbFileType	String	type of content (possible values- File / Directory)	Directory
sbFileLength	long	size of content.(if directory, it will always be zero)	0
sbFileLastModified	long	last modified time of content	1430916720000

### Validations

Sr No.	Exception	Description	Example
1.	IllegalArgumentException	provided null or empty input parameter- sbDirPath	sbDirPath= null sbDirPath=" "
2.	IllegalArgumentException	when provided sbDirPath is not a relative path	sbDirPath="/opt/application_storage/storage_box/TRADING/capitalmarket"
3.	IllegalArgumentException	when provided sbDirPath is not a directory	sbFilePath="TRADING/capitalmarket/basket_order/0605/sample.txt"
4.	FileNotFoundException	when provided sbDirPath is not exist.	-
5.	IllegalStateException	storagebox is not available for list operation	when manually make storagebox unavailable for maintenance.

## JAVA Client Class

```
import com.finlogic.util.sbutility.PutFileBean;
import com.finlogic.util.sbutility.StorageBoxImpl;
import java.io.File;

public class Sbclient
{
    public static void main(String[] args)
    {
        SBCommonOperation sbCommon = SBCommonOperation.getSBCommonOperation ();
        try
        {
            //GETFILE
            String receivedPath =
            sbCommon.getFile("TRADING/capitalmarket/basket_order/0605/ORD060515182201.xls");

            System.out.println("getFile: Returned path- " + receivedPath);

            //PUTFILE
            File sourceFile = new File("/opt/apache-tomcat1/webapps/trading/basket/ORD060515182201.xls");
            String sbDestPath = "TRADING/capitalmarket/ORD060515182201.xls";
            PutFileBean putFileBean = new PutFileBean();
            putFileBean.setSourceFile(sourceFile);
            putFileBean.setSbDestPath(sbDestPath);
            putFileBean.setRemoveFromSource(true);
            putFileBean.setOverwrite(true);
            sbCommon.putFile(putFileBean);

            //IS FILE EXIST
            boolean status =
            sbCommon.isFileExist("TRADING/capitalmarket/basket_order/ORD060515182201.xls");
            System.out.println("status::" + status);

            //DELETEFILE
            boolean status1 =
            sbCommon.deleteFile("TRADING/capitalmarket/basket_order/ORD060515182201.xls");
            System.out.println("status::" + status1);

            //LISTFILES
            List<SBFileBean> sBeanList = sbCommon.getList("TRADING/capitalmarket");
            for (SBFileBean sBean : sBeanList)
            {
                System.out.println("filepath: "+sBean.getSbFileName());
                System.out.println("filetype: "+sBean.getSbFileType());
                System.out.println("file size:"+sBean.getSbFileLength());
                System.out.println("file last modified:"+sBean.getSbFileLastModified());
            }

        } catch (Exception e)
        {
            e.printStackTrace();
        }
    }
}
```

## **PHP Client Class**

```
<?php
include_once '/var/www/html/sbutility/SBCommonOperation.php';

$sbClient=SBCommonOperation::getSbCommonOperation();
try {

    //GETFILE
    $receivedPath = $sbClient->getFile('pdesk/cdesk/samp.txt');
    echo 'getFile: Returned path- ' . $receivedPath;

    //PUTFILE
    $sourceFile = '/opt/application_storage/temp_files/cdesk/ajax_functions.js';
    $sbDestPath = 'pdesk/tempcheck/ajax_functions.js';
    $putFileBean = new PutFileBean();
    $putFileBean->setSourceFile($sourceFile);
    $putFileBean->setSbDestPath($sbDestPath);
    $putFileBean->setRemoveFromSource(FALSE);
    $putFileBean->setOverwrite(TRUE);
    echo $sbClient->putFile($putFileBean);

    //ISFILEEXIST
    $status = $sbClient->isFileExist('pdesk/tempcheck/emptyfile.txt');
    echo "status:::" . $status;

    //DELETEFILE
    $status1 = $sbClient->deleteFile("TRADING/capitalmarket/basket_order/ORD060515182201.xls");
    echo "status:::" . $status1;

    //LISTFILES
    $var = $sbClient->getList('pdesk/tempcheck');
    $sbobj = new SBFileBean();
    for ($i = 0; $i < count($var); $i++) {
        $sbobj = $var[$i];
        $filepath = $sbobj->getSbFileName();
        $typeoffile = $sbobj->getSbFileType();
        $filelength = $sbobj->getSbFileLength();
        $lastmodified = $sbobj->getSbFileLastModified();
        echo '<br>filepath--> ' . $filepath;
        echo '<br>type of file--> ' . $typeoffile;
        echo '<br>filelength--> ' . $filelength;
        echo '<br>lastmodified--> ' . $lastmodified;
    }
} catch (Exception $e) {
    echo $e->getMessage();
}

?>
```