# Accessing Amazon AWS S3

## Helpful Guides

* Uploading files to S3: <http://docs.aws.amazon.com/AmazonS3/latest/dev/UploadObjSingleOpJava.html>
* Getting files from S3: <http://docs.aws.amazon.com/AmazonS3/latest/dev/RetrievingObjectUsingJava.html>

## AWS Security Credential

* Working with Credentials: <http://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/credentials.html>
* To get your S3’s access key and secret access key which you will need to authenticate your app’s access to your S3, on your AWS console homepage go to: My Security Credentials > Access Keys > Create New Access Key
  + This will give you an **Access Key** and **Secret Access Key**.
    - Save these in a “**credentials**” file in “C:/Users/YOUR\_USER\_NAME/.aws”. See “AWS Credentials File Format” section in “Working with Credentials” link above for proper format.

## JARs

* If not using Maven, drop these JARs into your Tomcat’s lib folder.
* **Amazon AWS SDK for Java**: <https://aws.amazon.com/sdk-for-java/>
* **Apache Commons FileUpload**: <http://commons.apache.org/proper/commons-fileupload/download_fileupload.cgi>
* **Apache Commons IO**: <http://commons.apache.org/proper/commons-io/download_io.cgi>

## HTML

* To have a form that uploads a file: <form enctype=”multipart/form-data”>
* To allow a user to upload a file: <input type=”file”>

## Java

* Important classes from Amazon AWS SDK: **AmazonS3Client** for connecting to your S3, **PutObjectRequest** for uploading to your S3, **ObjectListing** for getting objects from your S3.
* Method to upload to your S3: amazonS3Client.**putObject**(new PutObjectRequest(bucketName, objectKeyName, fileStream, objectMetadata))
* Method to get objects/files in a particular bucket: amazonS3Client.**listObjects**(bucketName)

## Demo

* **S3Demo** is a Java Dynamic Web Project built in Eclipse that demoes accessing an S3 bucket to upload a file and list the names of the files that are in there.