

How_to_upload_a_file_to_server_with_application/x-www-form-urlencoded

It is simple to upload a file using **application/x-www-form-urlencoded** combined with **base64**. The only problem is that the uploading fails with files exceeding few 10th Kb.

Client source:

```
import httplib, urllib
import base64, os.path

def imageToURL( aPath ):
    # read the binary data of the picture
    data = open(aPath, 'rb').read()
    # encoded it to base64
    encodedData = base64.encodestring( data )
    headers = { "Content-type": "application/x-www-form-urlencoded",
                "Accept": "text/plain",
                }

    params = urllib.urlencode({ u'fileName': os.path.split(aPath)[1],
                               u'data':encodedData})

    conn = httplib.HTTPConnection( "yourURL.xxx" )
    conn.request( "POST", "/uploaderFolder/image_uploader.php", params, headers )
    response = conn.getresponse( )
    # returns "True" or "False" if failed
    print response.read( )
    # status for debugging
    print response.status
    conn.close( )

if __name__ == "__main__":
    imageToURL("yourImage.jpg")
```

Server source (image_uploader.php):

```
<?php
// In this example a directory "images" needs to be present on the same directory where
// image_uploader.php is, with the necessary rights for the script to write data inside

if(isset($_POST['fileName'])){
    $filename = $_POST['fileName'];
}
else{
    // if not: just stop here
    print "False";
    die();
}

if(isset($_POST['data'])){
    $encodedData = $_POST['data'];
    $data = base64_decode($encodedData);
}
else{
    // if not: just stop here
    print "False";
    die();
}
```

How_to_upload_a_file_to_server_with_application/x-www-form-urlencoded

```
// full path for the image
$filepathname = 'images/'.$filename;

// write the file to the server into the images directory
$handle = fopen($filepathname, 'wb');
fputs($handle, $data, strlen($data));
fclose($handle);

// return
echo "True";
?>
```