

How to Upload Files In Android

You can use following codes to upload files with extra parameters in Android app:

```
public String uploadFile(String uri, String fileName) throws Exception {
    String lineEnd = "\r\n";
    String twoHyphens = "--";
    String boundary = "*****";
    int bytesRead, bytesAvailable, bufferSize;
    byte[] buffer;
    int maxBufferSize = 1 * 1024 * 1024;
    FileInputStream fileInputStream = context.openFileInput(fileName);
    URL url = new URL(uri);

    // Open a HTTP connection to the URL
    HttpURLConnection conn = (HttpURLConnection) url.openConnection();

    conn.setDoInput(true); // Allow Inputs
    conn.setDoOutput(true); // Allow Outputs
    conn.setUseCaches(false); // Don't use a Cached Copy
    conn.setRequestMethod("POST");
    conn.setRequestProperty("Content-Encoding", "gzip");
    conn.setRequestProperty("Connection", "Keep-Alive");
    //conn.setRequestProperty("Content-Type", "text/html; charset=UTF-8");
    conn.setRequestProperty("Content-Type", "multipart/form-data;boundary=" + boundary);
    conn.setRequestProperty("appbundle_product[playlist]", fileName);
    DataOutputStream dos = new DataOutputStream(conn.getOutputStream());

    dos.writeBytes(twoHyphens + boundary + lineEnd);
    String id=params.get("appbundle_product[id]").toString();
    dos.writeBytes("Content-Disposition: form-data; name=\"appbundle_product[id]\""+ lineEnd);
    //dos.writeBytes("Content-Type: text/plain; charset=UTF-8" + lineEnd);
    //dos.writeBytes("Content-Length: " + name.length() + lineEnd);
    dos.writeBytes(lineEnd);
    dos.writeBytes(id); // mobile_no is String variable
    dos.writeBytes(lineEnd);

    dos.writeBytes(twoHyphens + boundary + lineEnd);
    String playlistType=params.get("appbundle_product[playlist_type]").toString();
    dos.writeBytes("Content-Disposition: form-data; name=\"appbundle_product[playlist_type]\""+
lineEnd);
    //dos.writeBytes("Content-Type: text/plain; charset=UTF-8" + lineEnd);
    //dos.writeBytes("Content-Length: " + name.length() + lineEnd);
    dos.writeBytes(lineEnd);
    dos.writeBytes(playlistType); // mobile_no is String variable
    dos.writeBytes(lineEnd);

    dos.writeBytes(twoHyphens + boundary + lineEnd);
    String macValue=params.get("appbundle_product[mac_value]").toString();
    dos.writeBytes("Content-Disposition: form-data; name=\"appbundle_product[mac_value]\""+
lineEnd);
    //dos.writeBytes("Content-Type: text/plain; charset=UTF-8" + lineEnd);
    //dos.writeBytes("Content-Length: " + name.length() + lineEnd);
    dos.writeBytes(lineEnd);
    dos.writeBytes(macValue); // mobile_no is String variable
    dos.writeBytes(lineEnd);

    dos.writeBytes(twoHyphens + boundary + lineEnd);
    dos.writeBytes("Content-Disposition: form-data; name=appbundle_product[playlist];filename="
+ fileName + "" + lineEnd);
    dos.writeBytes(lineEnd);
```

```

// create a buffer of maximum size
bytesAvailable = fileInputStream.available();

bufferSize = Math.min(bytesAvailable, maxBufferSize);
buffer = new byte[bufferSize];

// read file and write it into form...
bytesRead = fileInputStream.read(buffer, 0, bufferSize);

while (bytesRead > 0) {
    dos.write(buffer, 0, bufferSize);
    bytesAvailable = fileInputStream.available();
    bufferSize = Math.min(bytesAvailable, maxBufferSize);
    bytesRead = fileInputStream.read(buffer, 0, bufferSize);
}
// send multipart form data necessary after file data...
dos.writeBytes(lineEnd);
dos.writeBytes(twoHyphens + boundary + twoHyphens + lineEnd);

// Responses from the server (code and message)
int serverResponseCode = conn.getResponseCode();
String serverResponseMessage = conn.getResponseMessage();
//close the streams //
BufferedReader br = new BufferedReader(new InputStreamReader((conn.getInputStream())));
StringBuilder sb = new StringBuilder();
String line;
while ((line = br.readLine()) != null) {
    sb.append(line).append("\n");
}

fileInputStream.close();
dos.flush();
dos.close();
return sb.toString();
}

```

Usage

```

1 private class UploadFileTask extends AsyncTask {
2
3     @Override
4     protected Object doInBackground(Object[] params) {
5
6         String result = null;
7         try {
8             result = uploadFile(getString(R.string.upload_playlist_request), "test.txt");
9             return result;
10        } catch (Exception e) {
11            return e.getMessage();
12        }
13    }
14 }

```