

## Download Files Using PL/SQL and a Gateway (ORDS, mod\_plsql, EPG)

This article gives some examples of the PL/SQL needed to download files using a PL/SQL gateway, such as ORDS, mod\_plsql or the embedded PL/SQL gateway (EPG).

- [Background](#)
- [BFILE](#)
- [BLOB](#)
- [Programatically](#)
- [APEX](#)

### Background

Since Oracle 8i there have been a number of ways to download files using PL/SQL and a gateway. The way you call these will depend on the gateway used and will also vary slightly if you are using APEX to make the call. The following articles explain the gateway part of the puzzle.

- [Oracle REST Data Services \(ORDS\) : File Upload and Download \(mod\\_plsql Style\)](#)
- [Oracle REST Data Services \(ORDS\) : RESTful Web Services Handling Media \(Documents and Images\)](#)
- [File Upload and Download Procedures in Oracle Database 10g \(mod\\_plsql\)](#)
- [Images from Oracle Over HTTP](#)
- [DBMS\\_EPG - The Embedded PL/SQL Gateway in Oracle 10g Database Release 2](#)
- [APEX Tips : File Download From a Button or Link](#)

The rest of this article is focused on the PL/SQL code required to initiate the download of a file, based on various sources.

### BFILE

In this example we source the file from the database file system using a directory object.

```
CREATE OR REPLACE DIRECTORY file_dir AS '/path/to/my/images/';  
GRANT READ ON DIRECTORY file_dir TO apex_ws_user;
```

With the directory object in place, we can now use a BFILE to read the file and push it out over the gateway using the WPG\_DOCLOAD package.

```

CREATE OR REPLACE PROCEDURE get_file (p_dir          IN  VARCHAR2,
                                      p_file_name     IN  VARCHAR2,
                                      p_mime_type     IN  VARCHAR2) IS

    l_bfile BFILE;
BEGIN
    l_bfile := BFILENAME(p_dir, p_file_name);
    sys.OWA_UTIL.mime_header(p_mime_type, FALSE);
    sys.HTP.p('Content-Length: ' || DBMS_LOB.getlength(l_bfile));
    sys.HTP.p('Content-Disposition: filename="' || p_file_name || '"');
    sys.OWA_UTIL.http_header_close;

    sys.WPG_DOCLOAD.download_file(l_bfile);
EXCEPTION
    WHEN OTHERS THEN
        HTP.p('Whoops');
END;
/

```

## BLOB

In this example the media is stored in a BLOB column of a DOCUMENTS table, which also include a MIME\_TYPE column.

```

CREATE OR REPLACE PROCEDURE get_file (p_file_name IN VARCHAR2) IS
    l_blob_content documents.blob_content%TYPE;
    l_mime_type    documents.mime_type%TYPE;
BEGIN
    SELECT blob_content,
           mime_type
    INTO   l_blob_content,
           l_mime_type
    FROM   documents
    WHERE  file_name = p_file_name;

    sys.OWA_UTIL.mime_header(l_mime_type, FALSE);
    sys.HTP.p('Content-Length: ' || DBMS_LOB.getlength(l_blob_content));
    sys.HTP.p('Content-Disposition: filename="' || p_file_name || '"');
    sys.OWA_UTIL.http_header_close;

    sys.WPG_DOCLOAD.download_file(l_blob_content);
EXCEPTION
    WHEN OTHERS THEN
        HTP.p('Whoops');
END;
/

```

## Programatically

In some cases, you want to create the contents of a file programatically and just push it through the gateway using the HTP package. This example pushes out some columns of the EMP table in CSV format.

```
CREATE OR REPLACE PROCEDURE get_bfile (p_file_name IN VARCHAR2,
                                         p_mime_type IN VARCHAR2) IS

    l_clob CLOB      := '';
    l_offset NUMBER := 1;
    l_chunk  NUMBER := 3000;
BEGIN
    FOR cur_rec IN (SELECT empno || ',' || ename || '' as data FROM emp) LOOP
        l_clob := l_clob || cur_rec.data || CHR(10);
    END LOOP;

    sys.OWA_UTIL.mime_header(p_mime_type, FALSE);
    sys.HTP.p('Content-Length: ' || DBMS_LOB.getlength(l_bfile));
    sys.HTP.p('Content-Disposition: filename="' || p_file_name || '"');
    sys.OWA_UTIL.http_header_close;

    -- Handle CLOBS larger than 32K.
    LOOP
        EXIT WHEN l_offset > LENGTH(p_clob);
        HTP.prn(SUBSTR(p_clob, l_offset, l_chunk));
        l_offset := l_offset + l_chunk;
    END LOOP;
EXCEPTION
    WHEN OTHERS THEN
        HTP.p('Whoops');
END;
/
```

## APEX

The previous examples all work when called from APEX, but you will often see two extra lines included. The example below is a repeat of the BLOB example, but includes the extra APEX related lines. These could be added to the other examples if required.

```
CREATE OR REPLACE PROCEDURE get_file (p_file_name IN VARCHAR2) IS
    l_blob_content documents.blob_content%TYPE;
    l_mime_type     documents.mime_type%TYPE;
BEGIN
    SELECT blob_content,
           mime_type
    INTO   l_blob_content,
           l_mime_type
    FROM   documents
    WHERE  file_name = p_file_name;
```

```
sys.HTP.init;
sys.OWA_UTIL.mime_header(l_mime_type, FALSE);
sys.HTP.p('Content-Length: ' || DBMS_LOB.getlength(l_blob_content));
sys.HTP.p('Content-Disposition: filename="' || p_file_name || '"');
sys.OWA_UTIL.http_header_close;

sys.WPG_DOCLOAD.download_file(l_blob_content);
apex_application.stop_apex_engine;
EXCEPTION
  WHEN apex_application.e_stop_apex_engine
    THEN RAISE;
  WHEN OTHERS THEN
    HTP.p('Whoops');
END;
/
```

For more information see:

- [Oracle REST Data Services \(ORDS\) : File Upload and Download \(mod\\_plsql Style\)](#)
- [Oracle REST Data Services \(ORDS\) : RESTful Web Services Handling Media \(Documents and Images\)](#)
- [File Upload and Download Procedures in Oracle Database 10g \(mod\\_plsql\)](#)
- [Images from Oracle Over HTTP](#)
- [DBMS\\_EPG - The Embedded PL/SQL Gateway in Oracle 10g Database Release 2](#)
- [APEX Tips : File Download From a Button or Link](#)

Hope this helps. Regards Tim...

[Back to the Top.](#)

Created: 2019-08-20 Updated: 2023-10-11

Contact Us



[Home](#) | [Articles](#) | [Scripts](#) | [Blog](#) | [Certification](#) | [Videos](#) | [Misc](#) | [About](#)

About Tim Hall

Copyright & Disclaimer