

2012

Display Company Logo for Apex Applications

Displaying Company Logo Step by Step

Basic requirement in most implementation is displaying company logo. This document shows how to achieve it in Oracle Application Express.



Display Company Logo for Apex Application

A basic requirement in most implementations is displaying company logo. This document shows how to achieve it in Application Express.

Implementation:

Brief Steps are

- Upload company logo image in Shared Components > Images > Create button.
- Go to Shared Components > Definition (Under Application section). In Logo section, enter logo type as *Image* and logo as `#WORKSPACE_IMAGES#logo.JPG`. `#WORKSPACE_IMAGES#` is a substitution string which is replaced with actual location of image.
- To make logo clickable, enter logo type as *Text* and logo as

Why I switched to logo type as Text to implement clickable logo?

When logo type Image is chosen, APEX uses *img* tag to display image. You can use logo attributes (field below logo) to set image attributes like width, height etc. There is no *href* attribute for *img* tag unlike anchor tag. So I wrote html tags bundling anchor and image tags, to put them under logo type *Text*. `#WORKSPACE_IMAGES#` substitution string is still applicable in HTML text.

Lets see the detailed steps.

- Go to Shared Components > Images (Under Files section). This displays all the available images in application. Choose create to upload new image. Note that application field is optional.
- Go to Shared Components > Definition (Under Application section). For non-clickable logo, choose logo type as Image and logo as `#WORKSPACE_IMAGES#logo.JPG`. I've used logo attributes for setting alternate text and title. Alternate text (*alt*) is used when image cannot display.
- For clickable logo, choose logo type as Text and logo. This HTML text is image tag wrapped around anchor tag with *href* and *target* attributes.
- Apply Changes and run the application.

How `#WORKSPACE_IMAGES#` substitution string works?

Viewing page source is good way to check how substitution strings are replaced. In case of `#WORKSPACE_IMAGES#logo.JPG` substitution string, it is replaced with a function call `wwv_flow_file_mgr.get_file?p_security_group_id=5937624817042064053&p_fname=logo.JPG` where `p_security_group_id` is workspace identifier. Package `wwv_flow_file_mgr` is in `FLWS_020100` which APEX

3.2 schema. `wwv_flow_file_mgr.get_file` fetches file from `APEX_WORKSPACE_FILES` table which contains images, css files and javascript files. Since `APEX_WORKSPACE_FILES` is one-stop table for above files, to reference css/javascript files in HTML code you can use same substitution string `#WORKSPACES_IMAGES#` like `#WORKSPACE_IMAGES#custom.css` or `#WORKSPACE_IMAGES#custom.js`

What is difference between #WORKSPACE_IMAGES# and #APP_IMAGES# substitution strings?

APEX Workspace can contain one or more applications. If application field is left NULL when creating an image, image take global or workspace scope. It means it is not attached to any specific application.

`#WORKSPACE_IMAGES#` substitution string is used for accessing workspace and application images. It is replaced

with `wwv_flow_file_mgr.get_file?p_security_group_id=&p_fname=`

`#APP_IMAGES#` substitution string is used for accessing only application images. It is replaced with

`wwv_flow_file_mgr.get_file?p_security_group_id=&p_flow_id=&p_fname=`

Note the extra parameter `p_flow_id` added to same function call.

If image is workspace image, `#APP_IMAGES#` substitution string cannot be used.

What is difference between #WORKSPACE_IMAGES# substitution string and /i/ notation for accessing images?

`/i/` notation can also be used for accessing images. `/i/` points to physical directory on the server where images are located. Suppose if APEX3.2 is installed on windows, `/i/` points to `\apex\images`. For above requirement if I placed logo file in physical directory, I can use logo Image as `/i/Logo.JPG`

Which notation to use? Personally I prefer `#WORKSPACE_IMAGES#` notation as I can upload images from APEX application which does not require access to server physical directories. I can also see the images in Shared Components > Images when they are uploaded from APEX application.