You will use first 40 minutes of this lab working with Oracle Intermedia for content based image retrieval. In content based image retrieval, an image (and not just the text) is used as a query. You will see in this lab how it is done through SQL statement in Oracle Intermedia. The SQL will return a set of images from the database of images and text, as a query result. Note that in content based image search several similar images may be retrieved from the database. The similarity we will use here is based on color, texture, shape and position features. **The second part of the lab is devoted to project 2.**

In this part of the lab you will do the following:

- Create signatures for all the images in table CSE480Images that you have created in last week's lab.

- Choose one of the database images as a query image (note that a query image could be any image and does not have to be an image from the database). and create the signature for the query image.

- Search CSE480Images table for database image signatures that are close to the query image signature.

- Display the images of the query results from the jpg file to determine how closely they match.

- Vary the degree of closeness and see the effect on query results.

To be able to do content based retrieval, you need to create signature for each image in the table and store it in the signature column of the table. Note that the database search is based on these signatures of the images.
1. The following example shows how a signature is created and stored in table CSE480Images for photo_id = 1. You do not need to run it. Just try to understand the code and go to the next step.

DECLARE
-- create objects t_image and image_s"g
  t_image  ORDSYS.ORDImage;
  image_s"g ORDSYS.ORDImageSignature;
BEGIN
-- get object id of the object with $photo\_id =1$ into $t\_image$.
  SELECT photo, photo_s"g INTO
    t_image, image_s"g FROM CSE480Images
  WHERE photo_id=1 FOR UPDATE;
-- Initialize image_s"g object with the signature of object t_image
  image_s"g.generateSignature(t_image);
-- Store the signature into the table CSE480Images
  UPDATE CSE480Images SET photo_s"g = image_s"g
    WHERE photo_id =1;
  commit;
END;
/

(a) What values actually get stored into the variables, t_image and image_s"g?
(b) Which object the method generateSignature operates on?
(c) What does the statement UPDATE accomplish?

2. You run the following code to create signatures for all the images. The code is not complete. You need to complete the where clause marked *** and then run the code.

DECLARE
  t_photo_id number;
  t_image  ORDSYS.ORDImage;
  image_s"g ORDSYS.ORDImageSignature;
  cursor c1 is
    select photo_id, photo, photo_s"g from CSE480Images for update;
BEGIN
  open c1;
loop
fetch c1 into t_photo_id, t_image, image_sig;
EXIT WHEN c1%NOTFOUND;
-- generate a signature
image_sig.generateSignature(t_image);
UPDATE CSE480Images SET photo_sig = image_sig
WHERE ***
end loop;
commit;
close c1;
END;
/

After running the above code you answer the following:

(a) What does the cursor statement above do?
(b) What does the statement UPDATE achieve?

3. You need to do the following for displaying information:
SET SERVEROUTPUT ON

4. Now you do a query on the database.

DECLARE
-- threshold provides the closeness of similarity
threshold NUMBER;
-- create signature object for query image
compare_sig ORDSYS.ORDSImageSignature;
annotation VARCHAR2(255);
-- create image object for query results
photo ORDSYS.ORDMAGE;
-- Define cursor for matching. Set weights for the visual attributes.
CURSOR getphotos IS
-- You read lecture note for explanation of the WHERE clause below.
SELECT annotation FROM CSE480Images S
WHERE ORDSYS.IMGSimilar(S.photo_sig, compare_sig,
'color'="1.0" texture="0.0" shape="0.0"
location="0.0", threshold)=1 AND photo_id <> 1;
BEGIN
-- select query signature as that of photo_id=1 in the database.
SELECT P.photo_id INTO compare.sig FROM cse480images P
WHERE P.photo_id = 1;
-- Set the threshold value.
threshold := 50;
-- Retrieve rows for matching images.
OPEN getphotos;
-- go through one row at a time of the cursor getphotos
LOOP
  FETCH getphotos INTO annotation;
  EXIT WHEN getphotos%NOTFOUND;
  dbms_output.put_line( annotation);
END LOOP;
CLOSE getphotos;
END;
/

5. Display the images of the query result using
display file.jpg
All of them may not be closely related to the query image.

6. Try increasing or decreasing thresholds and the weights to see the effect
on query results. Which values of threshold and weights are giving the
best results?