In [26]:

```
import cv2
import numpy as np
from matplotlib import pyplot as plt
from IPython.display import display, Image

#creating facecascade
face_cascade = cv2.CascadeClassifier("D:\\Nishi\\Masters\\SEM III\\Project\\Cosmoshop\\clie
display(Image(filename='D:\\Nishi\Masters\\SEM III\\Project\\Cosmoshop\client\static\\image
#loading image to matrix
img = cv2.imread("D:\\Nishi\Masters\\SEM III\\Project\\Cosmoshop\client\static\\images\\fac

#converting into grayscale image
gray_img = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
faces = face_cascade.detectMultiScale(gray_img,scaleFactor = 1.05,minNeighbors=10)
for x,y,w,h in faces :
    cropped_img = img[y:y+h,x:x+w]
    edges = cv2.Canny(cropped_img,130,1000)
    number_of_edges = np.count_nonzero(edges)
if number_of_edges > 1000:
    print("Wrinkle Found ")
else:
    print("No Wrinkle Found ")
```

No Wrinkle Found

In [30]:

```
#creating facecascade
face_cascade = cv2.CascadeClassifier("D:\\Nishi\\Masters\\SEM III\\Project\\Cosmoshop\\clie
display(Image(filename='D:\\Nishi\Masters\\SEM III\\Project\\Cosmoshop\client\static\\image
#loading image to matrix
img = cv2.imread("D:\\Nishi\Masters\\SEM III\\Project\\Cosmoshop\client\static\\images\\Wri

#converting into grayscale image
gray_img = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
#search the coordinates of an image
faces = face_cascade.detectMultiScale(gray_img,scaleFactor = 1.05,minNeighbors=10)
for x,y,w,h in faces :
    cropped_img = img[y:y+h,x:x+w]
    edges = cv2.Canny(cropped_img,110,1000)
    number_of_edges = np.count_nonzero(edges)

if number_of_edges > 1000:
    print("Wrinkle Found ")
else:
    print("No Wrinkle Found ")
```

```
Wrinkle Found
```

In [ ]: