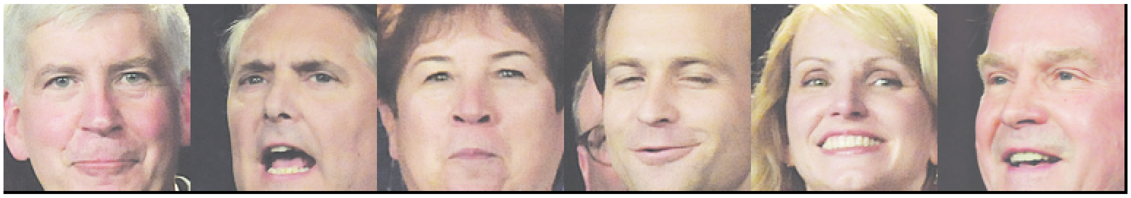
Final_project

September 12, 2020

```
In [2]: import zipfile
        from PIL import Image
        import pytesseract
        import cv2 as cv
        import numpy as np


        zip_file = zipfile.ZipFile("readonly/small_img.zip","r")
        lst_files = zip_file.infolist()

        lst_dic_objects = [] #list of dictionaries objects. Each dictionary encompasses three 

        face_cascade = cv.CascadeClassifier('readonly/haarcascade_frontalface_default.xml')


        for file in lst_files:#Loop to creat a data structure where each object is a python di

            new_dic={}
            zip_open = zip_file.open(file)

            image = Image.open(zip_open) #Turning the Zip file into a Pill Image.
            text = pytesseract.image_to_string(image) # getting the text from each Image using

            cv_image = np.asarray(image) #converting the image as a Nandarray.
            gray = cv.cvtColor(cv_image, cv.COLOR_BGR2GRAY) # turning the image to gray.
            faces = face_cascade.detectMultiScale(gray, scaleFactor =1.3, minNeighbors = 5) #g

            new_dic["image"] = image #Appending the image to the dictionary.
            new_dic["faces"] = faces #Appending the faces to the dictionary.
            new_dic["text"] = text # Appending the text to the dictionary.

            lst_dic_objects.append(new_dic)
        lst_cropped_image = []  #list of sublist, each sublist has all the cropped faces of an
        def cropped_faces(image, faces): #function to cropped the faces from each image.
            lst_cropped_faces = []
            for x,y,w,h in faces:
                image_cropped = image.crop((x,y,x+w,y+h))
                lst_cropped_faces.append(image_cropped)
            lst_cropped_image.append(lst_cropped_faces)

        def create_contact_sheet(lst_of_cropped_images): #function that creates a contact shee
            contact_sheet=Image.new("RGB", (770, 130), (0,0,0))
            x=0
            y=0
            size = (128, 128)
            for faces in lst_of_cropped_images:
                faces.thumbnail((size))

                contact_sheet.paste(faces, (x, y))
                if x + 128 == contact_sheet.width:
                    x=0
                    y=y+128
                else:
                    x=x+128

            return display(contact_sheet)


        for dic in lst_dic_objects:
            image = dic["image"]
            faces = dic["faces"]
            text = dic["text"]
            if "Christopher" in text:
                cropped_faces(image,faces)
            else:
                print("There were no faces in this file.")

        for lst in lst_cropped_image:
            create_contact_sheet(lst)
```

```
There were no faces in this file.
There were no faces in this file.
```



```
In [ ]:
```

```
In [ ]:
```