```
import numpy as nm
import cv2 as cv



# template1.png is the template
# certificate
template_path = r'C:\Users\Mayukha Thumiki\Desktop\Certificate_template.png'

# Excel file containing names of
# the participants
#details_path = 'gsocOrgsList.xlsx'

# Output Paths
output_path = 'C:\\Users\\Mayukha Thumiki\\Documents\\All_docs\\'

# Setting the font size and font
# colour
font_size = 1.5
font_color = (0,0,0)

# Coordinates on the certificate where
# will be printing the name (set
# according to your own template)
coordinate_y_adjustment = 3
coordinate_x_adjustment = 7

# loading the details.xlsx workbook
# and grabbing the active sheet
#obj = openpyxl.load_workbook(details_path)
#sheet = obj.active
```

```
# printing for the first 10 names in the
# excel sheet
#for i in range(1,11):

    # grabs the row=i and column=1 cell
    # that contains the name value of that
    # cell is stored in the variable certi_name

f=open(r"C:\Users\Mayukha Thumiki\Documents\All_docs\Names.txt",'r')
N=f.readlines()
for i in range(len(N)):
    certi_name = N[i].rstrip('\n')

    # read the certificate template
    img = cv.imread(template_path)

    # choose the font from opencv
    font = cv.FONT_HERSHEY_SCRIPT_COMPLEX

    # get the size of the name to be
    # printed
    text_size = cv.getTextSize(certi_name, font, font_size, 4)[0]

    # get the (x,y) coordinates where the
    # name is to written on the template
    # The function cv.putText accepts only
    # integer arguments so convert it into 'int'.
    text_x = (img.shape[1] - text_size[0]) / 2 + coordinate_x_adjustment
    text_y = (img.shape[0] + text_size[1]) / 2 - coordinate_y_adjustment
    text_x = int(text_x)
    text_y = int(text_y)
```

```
cv.putText(img, certi_name,(text_x ,text_y ),font,font_size,font_color, 4)


# Output path along with the name of the
# certificate generated
certi_path = output_path + certi_name + '.png'


# Save the certificate
cv.imwrite(certi_path,img)
```

```
import numpy as nm
import cv2 as cv
import datetime

tdy=datetime.date.today()
date_now=tdy.strftime("%d/%m/%Y")



# template1.png is the template
# certificate
template_path = r'C:\Users\Mayukha Thumiki\Desktop\Certificate_template.png'

# Excel file containing names of
# the participants
#details_path = 'gsocOrgsList.xlsx'

# Output Paths
output_path = 'C:\\Users\\Mayukha Thumiki\\Documents\\All_docs\\'

# Setting the font size and font
# colour
font_size = 1.5
font_color = (0,0,0)

# Coordinates on the certificate where
# will be printing the name (set
# according to your own template)
coordinate_y_adjustment = 1
coordinate_x_adjustment = 7

# loading the details.xlsx workbook
```

```
# and grabbing the active sheet
#obj = openpyxl.load_workbook(details_path)
#sheet = obj.active

# printing for the first 10 names in the
# excel sheet
#for i in range(1,11):

    # grabs the row=i and column=1 cell
    # that contains the name value of that
    # cell is stored in the variable certi_name

f=open(r"C:\Users\Mayukha Thumiki\Documents\All_docs\Names.txt",'r')
N=f.readlines()
for i in range(len(N)):
    certi_name = N[i].rstrip('\n')

    # read the certificate template
    img = cv.imread(template_path)

    # choose the font from opencv
    font = cv.FONT_HERSHEY_SCRIPT_COMPLEX

    # get the size of the name to be
    # printed
    text_size = cv.getTextSize(certi_name, font, font_size, 2)[0]
    datets=cv.getTextSize(date_now, font, font_size,1)[0]

    # get the (x,y) coordinates where the
    # name is to written on the template
    # The function cv.putText accepts only
```

```
# integer arguments so convert it into 'int'.
text_x = int((img.shape[1] - text_size[0]) / 2 + coordinate_x_adjustment)
text_y = int((img.shape[0] + text_size[1]) / 2 - coordinate_y_adjustment)
now_x = int((img.shape[1] - datets[0]) / 2 + (coordinate_x_adjustment+103))
now_y = int((img.shape[0] + datets[1]) / 2 - (coordinate_y_adjustment-183))


cv.putText(img, certi_name,(text_x ,text_y ),font,font_size,font_color, 2)
cv.putText(img, date_now,(now_x ,now_y ),font,0.5,font_color, 1)


# Output path along with the name of the
# certificate generated
certi_path = output_path + certi_name + '.png'


# Save the certificate
cv.imwrite(certi_path,img)
```