1 PSet-6.py

```
import numpy as np
import matplotlib.pyplot as plt

fileName = "Problem-Set-6"

# Grid points:
height = 200
width = 500

diameter = 50           # Diameter Cylinder

# Constants
rho = 3000              # kg/m^3
c = 840                 # J/(kg*C)
h = 28                  # W/(m^2*C)  Convective Heat Transfer Coefficient
k = 5.2                 # W/(m*C)    Thermal Conductivity
alpha = k / (rho * c)   # m^2/s      Thermal Diffusivity
#Fo = alpha * dt / (delta * delta)                          # Fourier Number
#Bi = h * delta / k                                         # Biot Number

T_initial = 10
T_right = 38
T_inf = 0

# Create array and initialize to T-initial
temperatures = np.zeros((width, height)) + T_initial
omega = np.zeros((width, height))
psi = np.zeros((width, height))


# Set the right boundary to T_right
for j in range(height):
    data[(width - 1), j] = T_right
```

```
34
35
36
37 # Print the data in the console (readable format)
38 #print(np.rot90(data))
39
40
41 figNum = 1
42 plt.figure(figNum)
43 plt.axes().set_aspect('equal')
44 plt.style.use('classic')
45 data_graphable = np.flipud(np.rot90(data))
46 heatmap = plt.pcolor(data_graphable)
47
48 plt.text(0.5, -0.02, "T = " + str(T_initial) + "\N{DEGREE SIGN}C",
49          horizontalalignment='center',
50          verticalalignment='top',
51          rotation=0,
52          clip_on=False,
53          transform=plt.gca().transAxes)
54 plt.text(0, 0.5, "Convective Boundary",
55          horizontalalignment='right',
56          verticalalignment='center',
57          rotation=90,
58          clip_on=False,
59          transform=plt.gca().transAxes)
60 plt.text(0.5, 1, "Insulated Surface",
61          horizontalalignment='center',
62          verticalalignment='bottom',
63          rotation=0,
64          clip_on=False,
65          transform=plt.gca().transAxes)
66 plt.text(1, 0.5, "T = " + str(T_right) + "\N{DEGREE SIGN}C",
67          horizontalalignment='left',
68          verticalalignment='center',
```

```
69          rotation=270,
70          clip_on=False,
71          transform=plt.gca().transAxes)
72
73 plt.axis("off")
74
75 plt.xlim(0, width)
76 plt.ylim(0, height)
77
78 cbar = plt.colorbar(heatmap)
79 cbar.set_label("Temperature (\N{DEGREE SIGN}C)")
80 plt.clim(np.amin(data), np.amax(data))
81
82 plt.savefig(fileName + "/images/" + fileName + "-Figure.png")
83 plt.show()
```