```
 1: # Progressive examples toward simple Python interactivity
 2: # Brygg Ullmer, Clemson University
 3: # Written and presented in-class on 2022-10-13
 4:
 5: import tkinter as tk
 6: from functools import partial
 7:
 8: ################# tkinter interactive grid class #################
 9:
10: class enoIgridTk:
11:
12:   numButtons      = 10 #how many UN SDGs
13:   numPerRow       = 4
14:   buttonWidth     = 10 #button width
15:   igridParent     = None
16:   igridFrame      = None
17:   useImageLabels  = False
18:   imageLabelDir   = None
19:   callbackFunc    = None
20:   imageMapNorm    = None
21:   imageMapDs      = None
22:   buttonMapIdx    = None
23:   lastIdxSelected = None
24:
25:   ################# constructor #################
26:
27:   def __init__(self, tkParent, **kwargs):
28:     self.callbackFunc = self.buttonCallback #default, overridable
29:     self.__dict__.update(kwargs) #allow class fields to be passed in constructor
30:
31:     self.buildGui(tkParent)
32:
33:   ################# button callback #################
34:
35:   def buttonCallback(self, whichButton):
36:     print("Callback %i pressed" % whichButton)
37:
38:     if self.useImageLabels and whichButton in self.imageMapDs:
39:       self.dimUnselected(whichButton)
40:       b1   = self.buttonMapIdx[whichButton-1]
41:       imgN = self.imageMapNorm[whichButton-1]
42:       b1.configure(image=imgN)
43:
44:     self.lastIdxSelected = whichButton
45:
46:   ################# dimUnselected #################
47:
48:   def dimUnselected(self, selectedIdx):
49:     indices = self.buttonMapIdx.keys()
50:     for i in indices:
51:       if i == selectedIdx-1: continue
52:       b    = self.buttonMapIdx[i]
53:       imgD = self.imageMapDs[i]
54:       b.configure(image=imgD)
55:
56:   ################# generate image filenames #################
57:
58:   def genImageFn(self, idx):
59:     result = "%s/%02i.png" % (self.imageLabelDir, idx)
60:     return result
61:
62:   def genImageNormFn(self, idx):
63:     result = "%s/norm/%02i.png" % (self.imageLabelDir, idx)
64:     return result
65:
66:   def genImageDsFn(self, idx):
67:     result = "%s/ds/%02i.png" % (self.imageLabelDir, idx)
68:     return result
```

```
    69:
    70:   ################# build gui #################
    71:
    72:   def buildGui(self, tkParent):
    73:     self.igridParent = tkParent
    74:     self.igridFrame  = tk.Frame(tkParent)
    75:     self.igridFrame.pack(expand=1)
    76:
    77:     rowFrame = tk.Frame(self.igridFrame) # invisible bundle of UI widgets
    78:     rowFrame.pack(expand=1)
    79:     colNum   = 1
    80:     self.buttonMapIdx = {}
    81:
    82:     if self.useImageLabels:
    83:       self.imageMapNorm = {}
    84:       self.imageMapDs   = {}
    85:
    86:     for i in range(self.numButtons):
    87:       cb = partial(self.callbackFunc, i+1)
    88:
    89:       if self.useImageLabels:
    90:         imgNFn = self.genImageNormFn(i+1); imgDFn = self.genImageDsFn(i+1)
    91:         imgN  = tk.PhotoImage(file=imgNFn); self.imageMapNorm[i] = imgN
    92:         imgD  = tk.PhotoImage(file=imgDFn); self.imageMapDs[i]   = imgD
    93:         b1    = tk.Button(rowFrame, image=imgN, command=cb)
    94:       else:
    95:         buttonLabel = "B%i" % (i+1)
    96:         b1 = tk.Button(rowFrame, text=buttonLabel, command=cb, width=self.buttonWi
dth)
    97:
    98:       self.buttonMapIdx[i] = b1
    99:
   100:       b1.pack(side=tk.LEFT)
   101:       colNum += 1
   102:
   103:       if colNum  > self.numPerRow:
   104:         rowFrame = tk.Frame(self.igridFrame);
   105:         rowFrame.pack(expand=1, side=tk.TOP)
   106:         colNum = 1
   107:
   108:       rowFrame.pack()
   109:
   110: ### end ###
```