```
#Morse Code Translator
#Icon found from http://icons8.com
import tkinter
from tkinter import IntVar, END, DISABLED, NORMAL
from playsound import playsound
from PIL import ImageTk, Image

#Define window
root = tkinter.Tk()
root.title('Morse Code Translator')
root.iconbitmap('morse.ico')
root.geometry('500x350')
root.resizable(0,0)

#Define fonts colors
button_font = ('SimSun', 10)
root_color = "#778899"
frame_color = "#dcdcdc"
button_color = "#c0c0c0"
text_color = "#f8f8ff"
root.config(bg=root_color)

#Define funtions
def convert():
    """Call the appropriate conversion function based off radio button values"""
    #English to morse code:
    if language.get() == 1:
        get_morse()
    elif language.get() == 2:
        get_english()


def get_morse():
    """Convert an English message to morse code"""
    #String to hold morse code message
    morse_code = ""

    #Get the input text and standardize it to lower case
    text = input_text.get("1.0", END)
    text = text.lower()

    #Remove any letters of symbols not in our dict keys
    for letter in text:
        if letter not in english_to_morse.keys():
            text = text.replace(letter, '')

    #Break up into individual words based on space " " and put into a list
    word_list = text.split(" ")

    #Turn each individual word in word_list into a list of letters
    for word in word_list:
        letters = list(word)
        #For each letter, get the morse code representation and append it to the string
        morse_code
        for letter in letters:
            morse_char = english_to_morse[letter]
            morse_code += morse_char
            #Seperate individual letters with a space
            morse_code += " "
        #Seperate individual words with a |
        morse_code += "|"

    output_text.insert("1.0", morse_code)


def get_english():
    """Convert a morse code message to english"""
```

```
    #String to hold English message
    english = ""

    #Get the input text
    text = input_text.get("1.0", END)

    #Remove any letters or symbols not in our dict keys
    for letter in text:
        if letter not in morse_to_english.keys():
            text = text.replace(letter, '')

    #Break up each word based on | and put into a list
    word_list = text.split("|")

    #Turn each word into a list of letters
    for word in word_list:
        letters = word.split(" ")
        #For each letter, get the English representation and add it to the string English
        for letter in letters:
            english_char = morse_to_english[letter]
            english += english_char
        #seperate individual words with a space
        english += " "

    output_text.insert("1.0", english)


def clear():
    """Clear both text fields"""
    input_text.delete("1.0", END)
    output_text.delete("1.0", END)


def play():
    """Play tones for corresponding dots and dashes"""
    #Determine where the morse code is
    if language.get() == 1:
        text = output_text.get("1.0", END)
    elif language.get() == 2:
        text = input_text.get("1.0", END)

    #Play the tones (., -, " " , |)
    for value in text:
        if value == ".":
            playsound('dot.mp3')
            root.after(100)
        elif value == "-":
            playsound('dash.mp3')
            root.after(200)
        elif value == " ":
            root.after(300)
        elif value == "|":
            root.after(700)


def show_guide():
    """Show a morse code guide in a second window"""
    #Image 'morse' needs to be a global variable to put on our window
    #Window 'guide' needs to be global to close in another function.
    global morse
    global guide

    #Create second window relative to the root window
    guide = tkinter.Toplevel()
    guide.title("Morse Guide")
    guide.iconbitmap('morse.ico')
    guide.geometry('350x350+'+ str(root.winfo_x()+500) + "+" + str(root.winfo_y()))
```

```
134         guide.config(bg=root_color)
135
136         #Create the image, label, and pack
137         morse = ImageTk.PhotoImage(Image.open('morse_chart.jpg'))
138         label = tkinter.Label(guide, image=morse, bg=frame_color)
139         label.pack(padx=10, pady=10, ipadx=5, ipady=5)
140
141         #Create a close button
142         close_button = tkinter.Button(guide, text="Close", font=button_font,
            bg=button_color, command=hide_guide)
143         close_button.pack(padx=10, ipadx=50)
144
145         #Disabel the guide button
146         guide_button.config(state=DISABLED)
147
148
149     def hide_guide():
150         """Hide the guide"""
151         guide_button.config(state=NORMAL)
152         guide.destroy()
153
154
155     #Create our morse code dictionaries
156     english_to_morse = {'a': '.-', 'b': '-...', 'c': '-.-.', 'd': '-..',
157                 'e': '.', 'f': '..-.', 'g': '--.', 'h': '....',
158                 'i': '..', 'j': '.---', 'k': '-.-', 'l': '.-..',
159                 'm': '--', 'n': '-.', 'o': '---', 'p': '.--.',
160                 'q': '--.-', 'r': '.-.', 's': '...', 't': '-',
161                 'u': '..--', 'v': '...-', 'w': '.--', 'x': '-..-',
162                 'y': '-.--', 'z': '--..', '1': '.----',
163                 '2': '..---', '3': '...--', '4': '....-', '5': '.....',
164                 '6': '-....', '7':  '--...', '8': '---..', '9': '----.',
165                 '0': '-----', ' ':' ', '|':'|', "":"" }
166
167     morse_to_english = dict([(value, key) for key, value in english_to_morse.items()])
168
169     #Define layout
170     #Create frames
171     input_frame = tkinter.LabelFrame(root, bg=frame_color)
172     output_frame = tkinter.LabelFrame(root, bg=frame_color)
173     input_frame.pack(padx=16, pady=(16,8))
174     output_frame.pack(padx=16, pady=(8,16))
175
176     #Layout for the input frame
177     input_text = tkinter.Text(input_frame, height=8, width=30, bg=text_color)
178     input_text.grid(row=0, column=1, rowspan=3, padx=5, pady=5)
179
180     language = IntVar()
181     language.set(1)
182     morse_button = tkinter.Radiobutton(input_frame, text="English --> Morse Code",
        variable=language, value=1, font=button_font, bg=frame_color)
183     english_button = tkinter.Radiobutton(input_frame, text="Morse Code --> English",
        variable=language, value=2, font=button_font, bg=frame_color)
184     guide_button = tkinter.Button(input_frame, text="Guide", font=button_font,
        bg=button_color, command=show_guide)
185
186     morse_button.grid(row=0, column=0, pady=(15,0))
187     english_button.grid(row=1, column=0)
188     guide_button.grid(row=2, column=0, sticky="WE", padx=10)
189
190     #Layout for the output frame
191     output_text = tkinter.Text(output_frame, height=8, width=30, bg=text_color)
192     output_text.grid(row=0, column=1, rowspan=4, padx=5, pady=5)
193
194     convert_button = tkinter.Button(output_frame, text="Convert", font=button_font,
        bg=button_color, command=convert)
195     play_button = tkinter.Button(output_frame, text="Play Morse", font=button_font,
```

```
    bg=button_color, command=play)
196 clear_button = tkinter.Button(output_frame, text="Clear", font=button_font,
    bg=button_color, command=clear)
197 quit_button = tkinter.Button(output_frame, text="Quit", font=button_font,
    bg=button_color, command=root.destroy)
198 convert_button.grid(row=0, column=0, padx=10, ipadx=50) #convert ipadx defines column
    width
199 play_button.grid(row=1, column=0, padx=10, sticky="WE")
200 clear_button.grid(row=2, column=0, padx=10, sticky="WE")
201 quit_button.grid(row=3, column=0, padx=10, sticky="WE")
202
203 #Run the root window's main loop
204 root.mainloop()
```