

```
# Define a package named geometry
# Create a module named cube inside the geometry package
# Create a module named sphere inside the geometry package
```

```
# geometry/cube.py
```

```
def cube_area(side_length):
    return 6 * side_length**2
```

```
def cube_volume(side_length):
    return side_length**3
```

```
# geometry/sphere.py
```

```
import math
```

```
def sphere_area(radius):
    return 4 * math.pi * radius**2
```

```
def sphere_volume(radius):
    return (4/3) * math.pi * radius**3
```

```
# main script
```

```
from geometry import cube, sphere
```

```
# Cube
```

```
cube_side_length = 5
cube_area_result = cube.cube_area(cube_side_length)
cube_volume_result = cube.cube_volume(cube_side_length)
```

```
print(f"Cube - Side Length: {cube_side_length}")
```

```
print(f"Cube Area: {cube_area_result}")
```

```
print(f"Cube Volume: {cube_volume_result}\n")
```

```
# Sphere

sphere_radius = 3

sphere_area_result = sphere.sphere_area(sphere_radius)

sphere_volume_result = sphere.sphere_volume(sphere_radius)


print(f"Sphere - Radius: {sphere_radius}")
print(f"Sphere Area: {sphere_area_result}")
print(f"Sphere Volume: {sphere_volume_result}")


import tkinter as tk
from tkinter import font

class FontChangerApp:

    def __init__(self, root):
        self.root = root
        self.root.title("Font Changer App")

        # Initial font settings
        self.current_font = font.nametofont("TkDefaultFont")
        self.label_text = "Hello, Tkinter!"
        self.create_widgets()

    def create_widgets(self):
        # Create a label with initial font settings
        self.label = tk.Label(self.root, text=self.label_text, font=self.current_font)
        self.label.pack(pady=20)

        # Font Name Checkbutton
```

```

self.font_name_var = tk.StringVar()

font_name_checkbutton = tk.Checkbutton(self.root, text="Font Name",
variable=self.font_name_var, command=self.update_font)

font_name_checkbutton.pack()


# Bold Checkbutton

self.bold_var = tk.BooleanVar()

bold_checkbutton = tk.Checkbutton(self.root, text="Bold", variable=self.bold_var,
command=self.update_font)

bold_checkbutton.pack()


# Font Size Checkbutton

self.font_size_var = tk.StringVar()

font_size_checkbutton = tk.Checkbutton(self.root, text="Font Size", variable=self.font_size_var,
command=self.update_font)

font_size_checkbutton.pack()


def update_font(self):

    # Update font based on check button states

    font_name = "Arial" if self.font_name_var.get() else self.current_font.cget("family")

    font_size = 14 if self.font_size_var.get() else self.current_font.cget("size")

    font_weight = "bold" if self.bold_var.get() else "normal"


    # Update label font

    new_font = font.Font(family=font_name, size=font_size, weight=font_weight)

    self.label.config(font=new_font)

    self.current_font = new_font


if __name__ == "__main__":

    root = tk.Tk()

    app = FontChangerApp(root)

    root.mainloop()

```