Carrera	Asignatura	Clave	No. Practica	Unidad	Tema
Ing. Sistemas Computacionales	**Graficación**	**SCC-1010**	**2**	**Figuras 3d graficacion**	**Representación y visualización De objetos en 3d**
Facilitador:	Ing. S.C. Erika Muciño Mancilla	**Cal:**	Grupo: **5S3**	**Nombre del Alumno:**	Morua Bernabe Miguel Uriel

INTRODUCCION

Las figuras tridimensionales (3D) son objetos que tienen tres dimensiones: ancho, largo y alto, lo que les permite tener profundidad y volumen. Estas figuras pueden ser poliedros, que están limitados por polígonos y no tienen superficies curvas, como el cubo y el tetraedro, o cuerpos redondos, que tienen superficies curvas, como la esfera y el cilindro.

Los poliedros tienen caras, aristas y vértices, y pueden ser regulares si todas sus caras son congruentes. Por otro lado, los cuerpos redondos, como la esfera, no tienen vértices y su superficie es completamente curva.

Las figuras 3D son esenciales en muchas disciplinas. En ingeniería, se utilizan para diseñar estructuras como edificios y puentes. En medicina, permiten la visualización de órganos y tejidos para diagnósticos y tratamientos. En educación, las impresoras 3D ayudan a enseñar conceptos complejos en ciencias y matemáticas.

Estas figuras son fundamentales para modelar y visualizar estructuras complejas con precisión, lo que las hace indispensables en diversas aplicaciones prácticas.

DESARROLLO

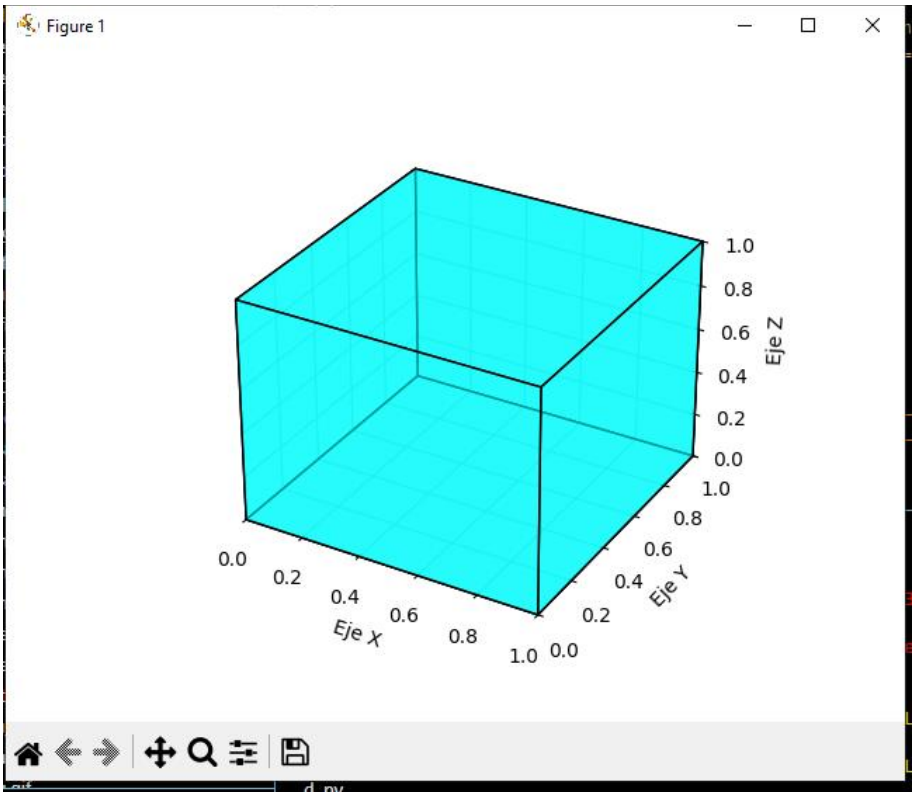
CUBO EN 3D

CUBO.PY > ...

```python
# Importamos las bibliotecas necesarias
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d.art3d import Poly3DCollection
import numpy as np

# Definimos los vértices del cubo en un array numpy
vertices = np.array([
    [0, 0, 0], [1, 0, 0], [1, 1, 0], [0, 1, 0],  # Base inferior
    [0, 0, 1], [1, 0, 1], [1, 1, 1], [0, 1, 1]   # Base superior
])

# Definimos las caras del cubo (cada cara es una lista de índices de vértices)
caras = [
    [vertices[0], vertices[1], vertices[2], vertices[3]],  # Cara inferior
    [vertices[4], vertices[5], vertices[6], vertices[7]],  # Cara superior
    [vertices[0], vertices[3], vertices[7], vertices[4]],  # Cara lateral izquierda
    [vertices[1], vertices[2], vertices[6], vertices[5]],  # Cara lateral derecha
    [vertices[0], vertices[1], vertices[5], vertices[4]],  # Cara frontal
    [vertices[3], vertices[2], vertices[6], vertices[7]]   # Cara trasera
]

# Creamos la figura y el objeto de ejes 3D
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')

# Añadimos las caras al gráfico como una colección de polígonos
ax.add_collection3d(Poly3DCollection(caras, color='cyan', edgecolor='k', alpha=0.6))

# Ajustamos los límites de los ejes
ax.set_xlim([0, 1])
ax.set_ylim([0, 1])
ax.set_zlim([0, 1])
```

```python
# Etiquetas de los ejes
ax.set_xlabel("Eje X")
ax.set_ylabel("Eje Y")
ax.set_zlabel("Eje Z")

# Mostramos el gráfico
plt.show()
```

Realizo: Ing. S.C. Erika Muciño Mancilla



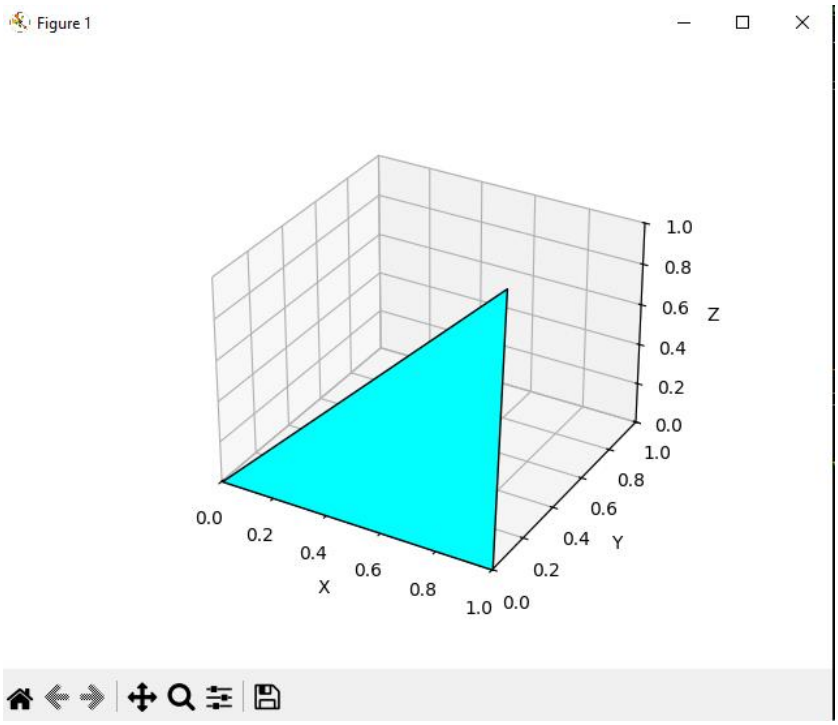
TRIANGULO 3D

codificacion2.py 7 | esfera.py 3 | triangulo3d.py | CUBO.PY 3

triangulo3d.py > ...

```python
# Importamos las bibliotecas necesarias
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d.art3d import Poly3DCollection
import numpy as np

# Definimos los vértices del triángulo en un array numpy
vertices = np.array([
    [0, 0, 0],  # Primer vértice en el origen
    [1, 0, 0],  # Segundo vértice en el eje x
    [0.5, 1, 0.5]  # Tercer vértice un poco elevado en z y desplazado en y
])

# Creamos una figura y un eje 3D
fig = plt.figure()
ax = fig.add_subplot(111, projection="3d")

# Definimos la cara del triángulo usando los vértices
caras = [[vertices[0], vertices[1], vertices[2]]]

# Creamos la colección de polígonos (caras) y la añadimos al eje
ax.add_collection3d(Poly3DCollection(caras, color="cyan", edgecolor="k"))

# Configuramos los límites de los ejes para visualizar bien el triángulo
ax.set_xlim([0, 1])
ax.set_ylim([0, 1])
ax.set_zlim([0, 1])

# Etiquetas de los ejes
ax.set_xlabel("X")
ax.set_ylabel("Y")
ax.set_zlabel("Z")
```

Realizo: Ing. S.C. Erika Muciño Mancilla



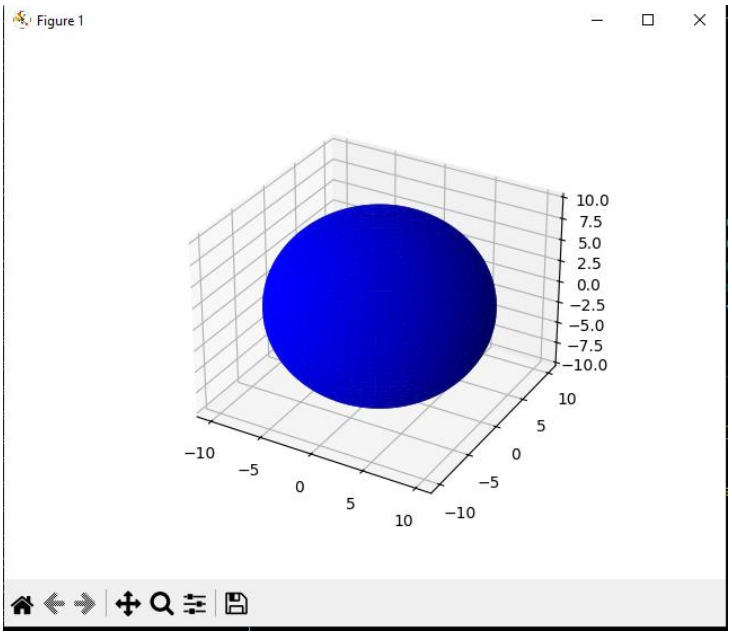
CIRCULO EN 3D

codificacion2.py 7 | esfera.py 3 | triangulo3d.py | CUBO.PY 3

esfera.py > ...

```python
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

# Crear una figura
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')

# Crear datos para la esfera
u = np.linspace(0, 2 * np.pi, 100)  # Genera 100 puntos entre 0 y 2π
v = np.linspace(0, np.pi, 100)      # Genera 100 puntos entre 0 y π
x = 10 * np.outer(np.cos(u), np.sin(v))  # Coordenada x de la esfera
y = 10 * np.outer(np.sin(u), np.sin(v))  # Coordenada y de la esfera
z = 10 * np.outer(np.ones(np.size(u)), np.cos(v))  # Coordenada z de la esfera

# Dibujar la esfera
ax.plot_surface(x, y, z, color='b')  # Dibuja la superficie de la esfera en color azul

# Mostrar la figura
plt.show()  # Muestra la figura en una ventana
```

Realizo: Ing. S.C. Erika Muciño Mancilla



CONCLUSION

La visualización de figuras geométricas en 3D con Python, usando la biblioteca matplotlib, es una técnica útil tanto para la educación como para proyectos de simulación y diseño. En esta serie de ejemplos, aprendimos a crear un cubo, un cono y un triángulo en el espacio tridimensional utilizando coordenadas y polígonos. Cada figura geométrica fue generada definiendo puntos específicos (vértices) y conectándolos para formar caras.

Crear y manipular figuras en 3D ayuda a comprender mejor las relaciones espaciales y las propiedades geométricas. También destaca cómo la programación permite automatizar visualizaciones complejas, simplificando tareas que serían difíciles de realizar a mano. En general, estos ejercicios demuestran que Python, con herramientas como matplotlib, es una poderosa opción para representar geometrías en 3D, apoyando áreas como la matemática, la física, la ingeniería y el diseño gráfico.

Realizo: Ing. S.C. Erika Muciño Mancilla

Evidencia de Aprendizaje	%	Indicador de Alcance						Evaluación Formativa de la Competencia
		A	B	C	D	E	F	Instrumento
Evaluación Escrita	30%							Cuestionario

Nombre y firma del docente:	Ing. S.C. Erika Muciño Mancilla
Nombre y Firma del estudiante:	