import networkx as nx  
import matplotlib.pyplot as plt  
import random  
  
  
def generate\_network(num\_nodes\_range, num\_edges\_range):  
 num\_nodes = random.randint(\*num\_nodes\_range)  
 num\_edges = random.randint(\*num\_edges\_range)  
  
 G = nx.gnm\_random\_graph(num\_nodes, num\_edges)  
  
 # Assign random weights (free flow travel time, capacity, and actual flow) to edges  
 for (u, v) in G.edges():  
 G.edges[u, v]['free\_flow\_time'] = round(random.uniform(5, 30), 2)  
 G.edges[u, v]['capacity'] = random.randint(100, 500)  
 G.edges[u, v]['actual\_flow'] = random.randint(50, G.edges[u, v]['capacity'])  
  
 return G  
  
  
def visualize\_network(G):  
 pos = nx.spring\_layout(G)  
 edge\_labels = {edge: f"FFT: {G.edges[edge]['free\_flow\_time']}\nCap: {G.edges[edge]['capacity']}\nFlow: {G.edges[edge]['actual\_flow']}" for edge in G.edges()}  
  
 plt.figure(figsize=(12, 8))  
 nx.draw(G, pos, with\_labels=True, node\_size=700, node\_color="skyblue", font\_size=12, font\_weight="bold",  
 font\_family="Arial")  
  
 # Adjust edge labels position to avoid overlap  
 edge\_label\_pos = nx.draw\_networkx\_edge\_labels(G, pos, edge\_labels=edge\_labels, font\_color='red', font\_size=8, rotate=False)  
  
 plt.title("Network Topology", fontname="Arial", fontsize=15)  
 plt.show()  
  
  
# Generate and visualize the network  
num\_nodes\_range = (10, 20)  
num\_edges\_range = (20, 40)  
G = generate\_network(num\_nodes\_range, num\_edges\_range)  
visualize\_network(G)