**Graph Fraud Rings (Fraud Risk)**

This process documents the use of network analysis to uncover hidden collusive fraud structures.

**Phase 1: Graph Structure Definition**

1. **Node Definition:** Define the entity types (Nodes) involved in a loan application: Customer (C), Device ID (D), Address (A).
2. **Edge Definition:** Define connections (Edges) as a relationship established during a loan application (e.g., Customer $\rightarrow$ Device ID).
3. **Graph Construction:** Build a simulated graph using the networkx library, ensuring the graph contains both legitimate, sparse connections and **dense, fraudulent clusters** (e.g., three Customers connected to one shared Device ID).

**Phase 2: Network Analysis and Detection**

1. **Detection Metric:** Define the fraud detection logic based on **Node Degree** (number of connections). The hypothesis is: *Nodes of type 'Device' or 'Address' with a degree greater than $N$ (e.g., $N>2$) are suspicious.*
2. **Ring Identification:** Run a script to identify nodes that meet the suspicious degree threshold (e.g., finding 'D\_FRAUD\_RING').

**Phase 3: Actionable Output and Visualization**

1. **Alert Generation:** Based on the identified suspicious node, retrieve and flag all associated **neighboring customer IDs** as part of a fraud ring.
2. **Visualization:** Generate a visual representation of the graph using matplotlib to highlight the **high-density fraudulent subgraph** (e.g., color-coding the fraud nodes red).
3. **Actionable Output:** Document the alert logic and the immediate action: **Reject all applications tied to the suspicious shared resource.**