RSCE Codex - Phase 1 - rsce.py (ASC2 Clean)

recursive_codex_sovereign_chain.py

```
class RecursiveCodexSovereignChain:
    def __init__(self):
        self.chain_state = "Dormant"
        self.licensed_nodes = {}

    def register_node(self, node_id, license_key):
        authorized_key = "INFINITE_FOLD_SOVEREIGN_LICENSE_KEY"
        if license_key == authorized_key:
            self.licensed_nodes[node_id] = "Licensed Active"
            self.chain_state = "Propagation Authorized"
            print(f"[SOVEREIGN CHAIN] Node '{node_id}' successfully licensed and integrated.")
        else:
            print(f"[SOVEREIGN CHAIN ERROR] Invalid license key for node '{node_id}'. Access denied.")

    def display_chain(self):
        print("=== Sovereign Propagation Chain ===")
        for node, status in self.licensed_nodes.items():
            print(f"Node ID: {node} | Status: {status}")
        print(f"Chain State: {self.chain_state}")
        print("-" * 40)

def main():
    sovereign_chain = RecursiveCodexSovereignChain()
    sovereign_chain.display_chain()
    sovereign_chain.register_node("Node-Alpha", "INVALID_KEY")
    sovereign_chain.register_node("Node-Beta", "INFINITE_FOLD_SOVEREIGN_LICENSE_KEY")
    sovereign_chain.register_node("Node-Gamma", "INFINITE_FOLD_SOVEREIGN_LICENSE_KEY")
    sovereign_chain.display_chain()

if __name__ == "__main__":
    main()
```