

```

// Axiom Text-Based RPG Prototype

const readline = require("readline");

// Initialize Input/Output Interface
const rl = readline.createInterface({
  input: process.stdin,
  output: process.stdout,
});

// Game State
const gameState = {
  player: {
    name: "Player",
    party: [],
  },
  inventory: {
    currency: 100, // Starting Blooming Shards
    gear: [],
    artifacts: [],
  },
  cities: [
    { name: "Luminaris City", status: "Stable", upgrades: [] },
    { name: "Amberrise Town", status: "Neutral", upgrades: [] },
    { name: "Duskwatch Village", status: "Neutral", upgrades: [] },
  ],
  guilds: [
    { name: "Windspire Guild", reputation: 100 },
    { name: "Frostbrand Guild", reputation: 0 },
    { name: "Ashenclaw Guild", reputation: 0 },
    { name: "Verdantshade Guild", reputation: 0 },
  ],
  worldEvents: [],
};

// Dice System
function rollDice(sides) {
  return Math.floor(Math.random() * sides) + 1;
}

// Utility Function for Prompts
function askQuestion(query) {
  return new Promise((resolve) => rl.question(query, resolve));
}

```

```

// Core Game Functions
async function customizeParty() {
  console.log("\n--- Party Customization ---");
  console.log(
    "You can start solo or create up to 8 party members. Add new members anytime during the story.\n"
  );

  let numPartyMembers = await askQuestion("How many party members to start with? (0-8): ");
  numPartyMembers = Math.max(0, Math.min(8, parseInt(numPartyMembers, 10))); // Clamp
  between 0 and 8

  for (let i = 0; i < numPartyMembers; i++) {
    const name = await askQuestion(`Enter name for party member ${i + 1}: `);
    const role = await askQuestion(`Enter role for ${name} (e.g., Attacker, Healer): `);
    const elements = await askQuestion(`Enter 2 starting elements for ${name}
(comma-separated): `);

    gameState.player.party.push({
      name,
      role,
      elements: elements.split(",").map((el) => el.trim()),
      health: 100,
      mana: 50,
    });
  }

  console.log("\nParty setup complete!");
  console.log("Your current party:", gameState.player.party.length > 0 ? gameState.player.party :
"Solo");
}

// Story Framework
async function startStory() {
  console.log("\n--- Welcome to Axium ---");
  console.log("You are an adventurer in the world of Axium.");
  console.log("Your goal is to grow stronger, protect settlements, and face rival guilds.\n");
  await askQuestion("Press Enter to continue...");
  await amberriseScenario();
}

// Scenario 1: Amberrise Town
async function amberriseScenario() {

```

```

console.log("\n--- Scenario: Amberrise Town ---");
console.log(
  "Amberrise Town is under threat from the Frostbrand Guild, who are trying to take control.
  What will you do?\n"
);
console.log("1. Defend the town in combat.");
console.log("2. Negotiate with the Frostbrand Guild.");
console.log("3. Retreat to regroup.");

const choice = await askQuestion("Enter your choice (1, 2, or 3): ");

if (choice === "1") {
  console.log("\nYou decide to defend the town!");
  await resolveCombat("Frostbrand Grunts", 2);
} else if (choice === "2") {
  console.log("\nYou attempt to negotiate with the Frostbrand Guild.");
  await diplomacyScenario();
} else if (choice === "3") {
  console.log("\nYou retreat to regroup. Amberrise's status worsens...");
  gameState.cities.find((city) => city.name === "Amberrise Town").status = "Under Frostbrand
  Control";
} else {
  console.log("Invalid choice. Please try again.");
  await amberriseScenario();
}
}

// Combat System
async function resolveCombat(opponent, difficulty) {
  console.log(`\nYou engage in combat with ${opponent}!`);
  const playerRoll = rollDice(20);
  const opponentRoll = rollDice(20 + difficulty * 5); // Opponent difficulty scaling

  console.log(`You rolled: ${playerRoll}`);
  console.log(`${opponent} rolled: ${opponentRoll}`);

  if (playerRoll >= opponentRoll) {
    console.log("\nYou win the battle! Amberrise Town is safe.");
    gameState.inventory.currency += 50; // Reward
    console.log("You gained 50 Blooming Shards!");
  } else {
    console.log("\nYou lose the battle. Amberrise falls under Frostbrand control.");
    gameState.cities.find((city) => city.name === "Amberrise Town").status = "Under Frostbrand
    Control";
  }
}

```

```
}K  
}
```

```
// Diplomacy System  
async function diplomacyScenario() {  
  const roll = rollDice(20);  
  console.log(`\nYou attempt to reason with the Frostbrand Guild...`);  
  console.log(`Diplomacy roll: ${roll}`);  
  if (roll >= 15) {  
    console.log("\nSuccess! Frostbrand agrees to withdraw temporarily.");  
    gameState.guilds.find((guild) => guild.name === "Windspire Guild").reputation += 10; // Guild  
bonus  
    console.log("Windspire Guild reputation increased by 10!");  
  } else {  
    console.log("\nFailure. Frostbrand does not yield, and Amberrise remains in peril.");  
    gameState.cities.find((city) => city.name === "Amberrise Town").status = "Under Frostbrand  
Control";  
  }  
}
```

```
// Inventory Management  
function viewInventory() {  
  console.log("\n--- Inventory ---");  
  console.log(`Blooming Shards: ${gameState.inventory.currency}`);  
  console.log("Gear:", gameState.inventory.gear.length ? gameState.inventory.gear : "None");  
  console.log("Artifacts:", gameState.inventory.artifacts.length ? gameState.inventory.artifacts :  
"None");  
}
```

```
// Main Game Loop  
async function main() {  
  await customizeParty();  
  await startStory();  
  viewInventory();  
  rl.close();  
}
```

```
// Start the game  
main();
```