**Scenario Logic:**

* **Daytime**: Dwarves hunt orcs (dwarves = predator, orcs = prey)
* **Night time**: Orcs hunt dwarves (orcs = predator, dwarves = prey)
* The **roles flip every X turns** (every 20 time steps).
* Each agent has basic behaviors:
  + **Predator**: Seeks enemies in sight, chases, attacks if close.
  + **Prey**: Tries to escape or hide.
* Agents can die or reproduce based on events.
* Simple terrain (like safe zones, shadows, or caves) can be added later.

**Step-by-Step**

**Step 1: Set Up Your Environment**

Use **Python** with:

* **pygame** → for game-style visualization (moving sprites on grid)
* OR **matplotlib** for simpler visuals (dot plots)
* **random** and maybe math for logic

**Step 2: Create the Grid World**

* Define a basic 2D grid (e.g., 50x50)
* Make agents move inside this grid, looping at edges (torus-style or bounded)

**Step 3: Create the Agent Classes**

We’ll need two main agent types:

* **Orc**: with attributes like **x**, **y**, **is\_predator**, **alive**
* **Dwarf**: same structure
* A method like **.move()** and **.act()** depending on role (**predator**/**prey**)

**Step 4: Handle Day/Night Cycle**

* Every X steps (say 20), flip roles:

**Step 5: Implement Basic Behaviors**

* When predator:
  + Scan in vision range
  + Move toward closest enemy
  + Attack if adjacent
* When prey:
  + Move away from predator
  + Maybe hide in terrain (later)

**Step 6: Visualize Movement**

* Use **pygame** to draw the grid and represent orcs/dwarves in different colors.
* Optional: show who is predator with a ring/highlight

**Step 7: Add Basic Stats**

* Count how many orcs/dwarves remain
* Track how many kills each side gets
* Plot or display in-game