

```
package myowngames;
import java.util.*;
public class EclipseOfDawn {
    public static void main(String[] args) {
        Game game = new Game();
        game.start();
    }
}

class Game {
    private Player player;
    private Village village;
    private Storyline storyline;
    private List<NPC> npcs;
    private Currency currency;
    private List<Faction> factions;
    private Map<String, Location> locations;
    public Game() {
        player = new Player("Survivor");
        village = new Village("Hope Haven", 100);
        storyline = new Storyline();
        npcs = new ArrayList<>();
        factions = new ArrayList<>();
        currency = new Currency("Caps", 0);
        locations = new HashMap<>();
    }
}
```

```
        initializeNPCs();
        initializeFactions();
        initializeLocations();
    }
    private void initializeNPCs() {
        npcs.add(new NPC("Trader Joe", "A resourceful merchant", new Random().nextInt(50) + 50));
        npcs.add(new NPC("Wanderer Sarah", "A mysterious wanderer with useful information", 0));
    }
    private void initializeFactions() {
        factions.add(new Faction("The Machinists", "Experts in creating and maintaining advanced machines."));
        factions.add(new Faction("The Guardians", "Warriors dedicated to protecting humanity from mutants and raiders."));
        factions.add(new Faction("The Relic Hunters", "Seekers of pre-war artifacts to harness their power."));
    }
    private void initializeLocations() {
```

```
        locations.put("Machinist HQ", new
Location("Machinist HQ", "A hub of technological
innovation.", "The Machinists"));
        locations.put("Guardian Outpost", new
Location("Guardian Outpost", "A fortified base to
combat threats.", "The Guardians"));
        locations.put("Relic Vault", new Location("Relic
Vault", "A secretive facility full of artifacts.", "The
Relic Hunters"));
    }
    public void start() {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Welcome to the Nuclear
Survival Game!");
        System.out.println("Your village is under threat
from raiders. Can you save it?");
        boolean playing = true;
        while (playing) {
            System.out.println("\nWhat would you like to
do?");
            System.out.println("1. Explore for
resources");
            System.out.println("2. Craft items");
```

```
        System.out.println("3. Upgrade  
weapons/armor");  
        System.out.println("4. Check village status");  
        System.out.println("5. Fight raiders or  
mutants");  
        System.out.println("6. Interact with NPCs");  
        System.out.println("7. Trade");  
        System.out.println("8. View storyline  
progress");  
        System.out.println("9. Travel to a location");  
        System.out.println("10. Exit game");  
        int choice = scanner.nextInt();  
        switch (choice) {  
            case 1:  
                player.explore();  
                break;  
            case 2:  
                player.craftItem();  
                break;  
            case 3:  
                player.upgradeEquipment();  
                break;  
            case 4:  
                village.checkStatus();
```

```
        break;
    case 5:
        fightEnemies();
        break;
    case 6:
        interactWithNPCs();
        break;
    case 7:
        trade();
        break;
    case 8:
        storyline.displayProgress();
        break;
    case 9:
        travelToLocation();
        break;
    case 10:
        playing = false;
        System.out.println("Exiting game.
Goodbye!");
        break;
    default:
        System.out.println("Invalid choice. Try
again.");
```

```

    }
}
scanner.close();
}
private void fightEnemies() {
    System.out.println("Choose your enemy: 1.
Raiders, 2. Mutants");
    Scanner scanner = new Scanner(System.in);
    int enemyType = scanner.nextInt();
    if (player.getHealth() > 0) {
        int damage = new Random().nextInt(50) + 1;
        if (enemyType == 1) {
            System.out.println("You fought raiders and
took " + damage + " damage.");
        } else {
            System.out.println("You fought mutants
and took " + damage + " damage.");
        }
        player.takeDamage(damage);
    } else {
        System.out.println("You are too weak to fight.
Heal up first.");
    }
}
}

```

```
private void interactWithNPCs() {
    System.out.println("Available NPCs:");
    for (int i = 0; i < npcs.size(); i++) {
        System.out.println((i + 1) + ". " +
npcs.get(i).getName() + " - " +
npcs.get(i).getDescription());
    }
    System.out.println("Choose an NPC to interact
with:");
    Scanner scanner = new Scanner(System.in);
    int npcChoice = scanner.nextInt();
    if (npcChoice > 0 && npcChoice <= npcs.size())
{
        NPC npc = npcs.get(npcChoice - 1);
        System.out.println("Interacting with " +
npc.getName());
        npc.interact();
    } else {
        System.out.println("Invalid choice.");
    }
}
private void trade() {
```

```
        System.out.println("Trading system coming soon. Your current currency: " + currency.getAmount() + " " + currency.getName());
    }
    private void travelToLocation() {
        System.out.println("Available locations:");
        int index = 1;
        for (String locName : locations.keySet()) {
            System.out.println(index + ". " + locName + " - " + locations.get(locName).getDescription());
            index++;
        }
        System.out.println("Choose a location to travel to:");
        Scanner scanner = new Scanner(System.in);
        int locationChoice = scanner.nextInt();
        if (locationChoice > 0 && locationChoice <= locations.size()) {
            String selectedLocation = (String) locations.keySet().toArray()[locationChoice - 1];
            Location location = locations.get(selectedLocation);
            System.out.println("Traveling to " + location.getName());
        }
    }
}
```



```
        location.visit();
    } else {
        System.out.println("Invalid choice.");
    }
}
}

class Player {
    private String name;
    private int health;
    private int energy;
    private int level;
    private int experience;
    private int strength;
    private int agility;
    private int intelligence;
    private List<String> skills;
    public Player(String name) {
        this.name = name;
        this.health = 100;
        this.energy = 100;
        this.level = 1;
        this.experience = 0;
        this.strength = 5;
        this.agility = 5;
```

```
this.intelligence = 5;
this.skills = new ArrayList<>();
}
public void explore() {
    int found = new Random().nextInt(20) + 1;
    energy -= 10;
    System.out.println("You explored the wasteland
and found " + found + " resources. Energy left: " +
energy);
}
public void craftItem() {
    if (energy >= 10) {
        energy -= 10;
        health += 20;
        System.out.println("You crafted a medkit and
healed yourself. Current health: " + health);
    } else {
        System.out.println("Not enough energy to
craft an item.");
    }
}
public void upgradeEquipment() {
    if (energy >= 20) {
        energy -= 20;
```

```
        System.out.println("You upgraded your  
equipment. Your chances of survival have  
improved!");  
    } else {  
        System.out.println("Not enough energy to  
upgrade equipment.");  
    }  
}  
  
public void levelUp() {  
    level++;  
    experience = 0;  
    strength += 2;  
    agility += 2;  
    intelligence += 2;  
    System.out.println("Level up! You are now level  
" + level + ". Stats increased!");  
}  
  
public void pickSkill(String skill) {  
    skills.add(skill);  
    System.out.println("You learned a new skill: " +  
skill);  
}  
  
public void takeDamage(int damage) {  
    health -= damage;
```

```
        if (health <= 0) {
            health = 0;
            System.out.println("You have died. Game
over.");
        } else {
            System.out.println("You now have " + health
+ " health remaining.");
        }
    }
    public int getHealth() {
        return health;
    }
}

class NPC {
    private String name;
    private String description;
    private int health;
    public NPC(String name, String description, int
health) {
        this.name = name;
        this.description = description;
        this.health = health;
    }
    public String getName() {
```

```
        return name;
    }
    public String getDescription() {
        return description;
    }
    public void interact() {
        System.out.println("You interact with " + name +
". " + description);
    }
}

class Location {
    private String name;
    private String description;
    private String faction;
    public Location(String name, String description,
String faction) {
        this.name = name;
        this.description = description;
        this.faction = faction;
    }
    public String getName() {
        return name;
    }
    public String getDescription() {
```

```
        return description;
    }
    public void visit() {
        System.out.println("Visiting " + name + " - " +
description);
    }
}
class Faction {
    private String name;
    private String description;
    public Faction(String name, String description) {
        this.name = name;
        this.description = description;
    }
}
class Currency {
    private String name;
    private int amount;
    public Currency(String name, int amount) {
        this.name = name;
        this.amount = amount;
    }
    public int getAmount() {
        return amount;
    }
}
```

```
}  
    public String getName() {  
        return name;  
    }  
}  
  
class Village {  
    private String name;  
    private int health;  
    public Village(String name, int health) {  
        this.name = name;  
        this.health = health;  
    }  
    public void checkStatus() {  
        System.out.println(name + " village status:  
Health = " + health);  
    }  
}  
  
class Storyline {  
    public void displayProgress() {  
        System.out.println("Storyline progress: You are  
saving the village from raiders!");  
    }  
}
```