```
import matplotlib.pyplot as plt
import numpy as np

mass = 10
air_resistance_coefficient = 0.25
initial_speed = 25
launch_angle = np.pi / 3

def euler_method(mass, initial_speed, launch_angle, air_resistance_coefficient=0, initial_height=0, gravity=9.8, time_step=0.01):
    velocity_x = initial_speed * np.cos(launch_angle)
    velocity_y = initial_speed * np.sin(launch_angle)
    position_x = 0
    position_y = initial_height
    time = 0

    x_positions = []
    y_positions = []

    is_first_iteration = False
    while not is_first_iteration or position_y > 0:
        is_first_iteration = True
        x_positions.append(position_x)
        y_positions.append(position_y)
        speed = np.sqrt(velocity_x**2 + velocity_y**2)
        acceleration_x = - (air_resistance_coefficient * speed * velocity_x) / mass
        acceleration_y = - gravity - (air_resistance_coefficient * speed * velocity_y) / mass

        velocity_x += acceleration_x * time_step
        velocity_y += acceleration_y * time_step

        position_x += velocity_x * time_step + 0.5 * acceleration_x * (time_step**2)
        position_y += velocity_y * time_step + 0.5 * acceleration_y * (time_step**2)

        time += time_step

    return x_positions, y_positions

x_positions_with_drag, y_positions_with_drag = euler_method(mass, initial_speed, launch_angle, air_resistance_coefficient=air_resistance_
x_positions_without_drag, y_positions_without_drag = euler_method(mass, initial_speed, launch_angle)

plt.figure(figsize=(9, 5))
plt.title('Comparison of Projectile with and without Air Resistance')
plt.plot(x_positions_without_drag, y_positions_without_drag, label='Without Air Resistance')
plt.plot(x_positions_with_drag, y_positions_with_drag, color='red', label='With Air Resistance')
plt.xlabel('Distance (x-axis) in m')
plt.ylabel('Height (y-axis) in m')
plt.legend()
plt.show()
```