```java
package ch.hevs.gdx2d.lunar.main;

import java.util.Random;

import com.badlogic.gdx.graphics.Texture;
import com.badlogic.gdx.math.Vector2;

import ch.hevs.gdx2d.lib.GdxGraphics;
import ch.hevs.gdx2d.lib.interfaces.DrawableObject;
import ch.hevs.gdx2d.lunar.physics.Constants;
import ch.hevs.gdx2d.lunar.physics.PhysicalObject;

public class Gegner extends PhysicalObject implements DrawableObject{

    private boolean destroyed;

    private static final Random rand = new Random();

    private Texture meteor;

    public Gegner(Vector2 p) {
        super(p, new Vector2(rand.nextFloat() * rand.nextInt(50) * (rand.nextBoolean() ? 1 : rand.nextBoolean() ? 1 : -0.05f),
                rand.nextFloat() * rand.nextInt(10) * (-1)), Constants.GEGNER_MASS, 40, 40);
        meteor = new Texture("data/images/meteor.png");
        destroyed = false;
    }

    @Override
    public void step() {
        this.force.y = -Constants.GRAVITY * this.mass;
    }

    @Override
    public void draw(GdxGraphics arg0) {
        if (!destroyed) {
            arg0.draw(meteor, position.x - 25, position.y - 30, 50, 50);
        }
    }

    @Override
    public void removedFromSim() {
        destroyed = true;
    }

    @Override
    public boolean notifyCollision(int energy) {
        return (energy >= Constants.DESTRUCTION_ENERGY);
    }

}
```