

Python Implementation of the Satellite Command System

Description

This directory contains the Python implementation of the Satellite Command System. The implementation simulates the control of a satellite's orientation, solar panel status, and data collection through a series of commands.

Adherence to SOLID Principles and Evaluation Criteria

SOLID Principles

- **Single Responsibility Principle (SRP):** Each class and method in this implementation has a single responsibility. For instance, the `Satellite` class manages the state of the satellite, while separate action classes handle specific operations.
- **Open/Closed Principle (OCP):** The system is open for extension but closed for modification. New functionality (like additional satellite commands) can be added with new classes without altering existing code.
- **Liskov Substitution Principle (LSP):** The action classes are interchangeable without affecting the system's behavior, as they all inherit from a common `SatelliteAction` interface.
- **Interface Segregation Principle (ISP):** The `SatelliteAction` interface is minimal, ensuring that classes implementing the interface don't depend on methods they don't use.
- **Dependency Inversion Principle (DIP):** High-level modules (like the main application flow) depend on abstractions (such as the `SatelliteAction` interface), not on concrete classes of lower-level modules.

Evaluation Criteria

- **Code Quality:** Best practices, such as clear naming conventions, modular design, and documentation, are followed to ensure code quality.
- **Functionality:** The implementation correctly simulates satellite operations, maintaining its state through various commands.
- **Global Convention:** The code is easy to understand and follows Python conventions for readability and maintainability.
- **Gold Standards:** Exception handling and logging mechanisms are implemented. Logging provides insights into the operations performed and the state of the satellite.

Requirements

- Python 3.x

Installation and Execution

To run the Python implementation, navigate to the `Python/` directory and execute the following command:

```
python satellite_command_system.py
```

Features

- Initialize the satellite with default settings.
- Rotate the satellite in specified directions.
- Activate and deactivate solar panels.
- Collect data based on the status of the solar panels.

For more details, refer to the comments within the `satellite_command_system.py` file.