Design Patterns

CS 121, second year students

Assignment 6

November 15, 2021. Due to December 1.

Overview

In this assignment you should implement Facade and Bridge patterns.

Description

- **Facade pattern task**.

1. Given the multiple classes represent different functionality of a file processing, one has to implement a facade which simplifies access and operations. You can modify recommended classes.

2. Implement file readers based on the file extension. These classes **must** be able to read real pdf and docx files. You can use external libraries for this task.

PdfParser
+content: str
+read_file(filename: str) -> None +parse_content() -> None

DocxParser
+content: str
+read_file(filename: str) -> None +parse_content() -> None

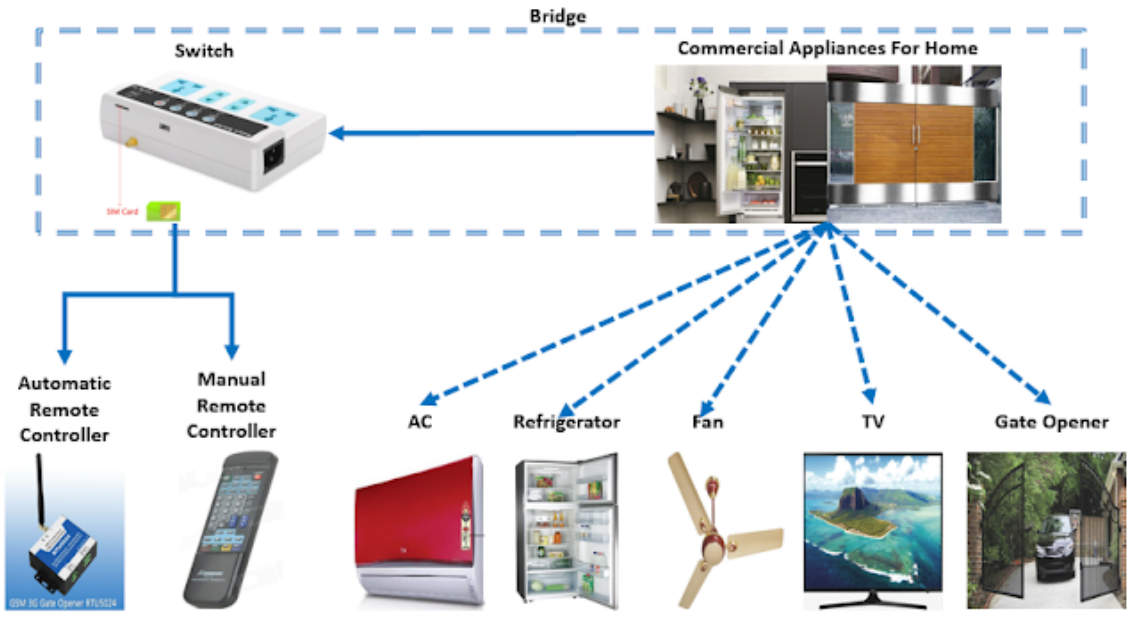
3. Create a class for entity extraction. Use external libraries or regular expressions [1,2,3].

Extractor
+content: str
+extract_dates() -> List[str] +extract_numbers () -> List[float] +extract_names() -> List[str]

4. Provide facade class **Processor** which aggregates or composites parsers and extractor. Also, add testing code.

❖ Bridge pattern task.

1. Imagine you are working on the IoT project. Implement the following diagram:



2. Each Appliances class must contain start (), stop () methods with appropriate logic, such as power off. Switch and the corresponding Remote Controllers must contain the required methods turn_on (), turn_off. Complete the rest of the logic at your discretion.
3. Provide testing of your code.

References:

1. https://www.pythonforbeginners.com/files/how-to-extract-a-date-from-a-txt-file-in-python
2. https://spacy.io/
3. https://stackoverflow.com/questions/4289331/how-to-extract-numbers-from-a-string-in-python
4. *Eric Freeman and others*. **Design Patterns (Head First).**
5. *Sean Bradley*. **Design Patterns In Python. Common GoF (Gang of Four) Design Patterns Implemented In Python.**
6. *Robert Martin*. **Clean Code: A Handbook of Agile Software Craftsmanship.**
7. *Mark Lutz*. **Python, 5 ed.**
8. *Dan Bader*. **Clean Python.**