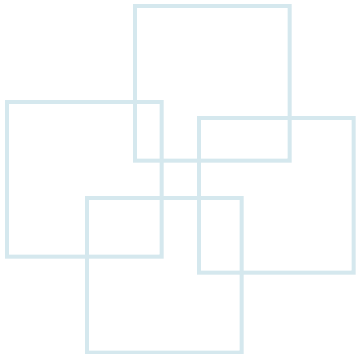
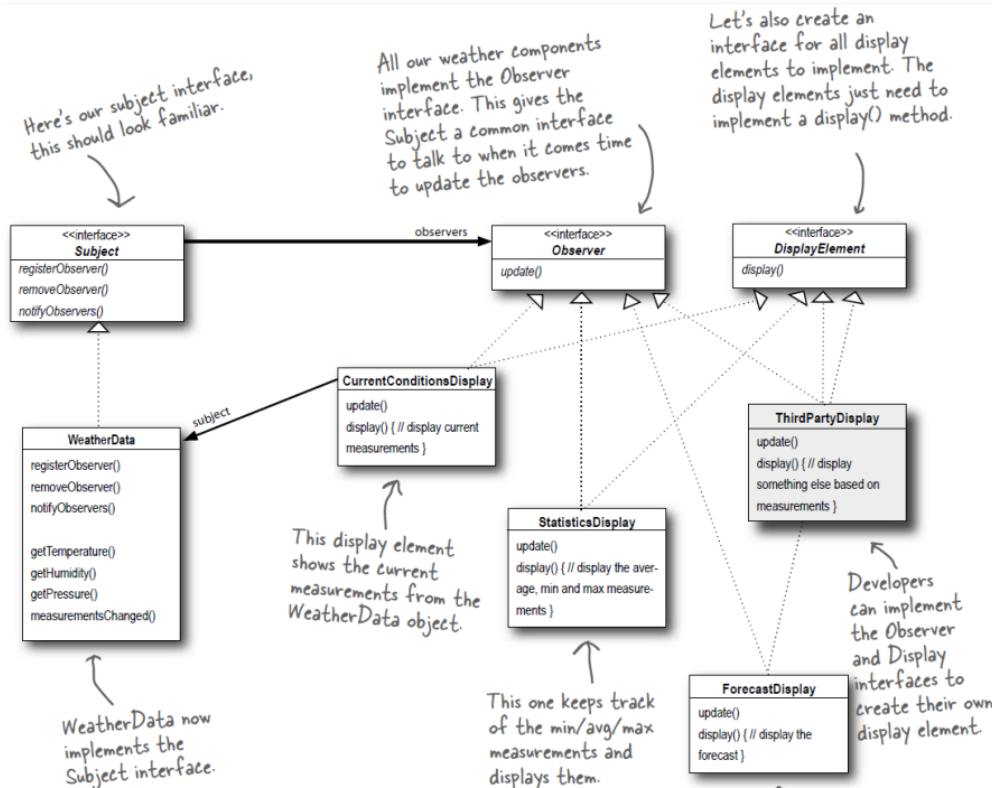
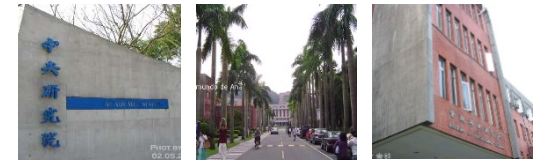


Project 2

Observer Pattern





Requirement

- Observer pattern practice
 - Write a JAVA/C++ project to implement a client-server program with the observer pattern
- Implement a server (i.e., subject) to support the following commands:
 - `send "string"`: send a string to all clients (i.e., observers)
 - `add client_name`: Create and add a new client `client_name` that subscribes to the server.
 - `del client_name`: Remove and delete a client `client_name` from the subscription list.
- The server supports two types clients:
- Each client would *print its name and the received string message* when the server sends a string message to all its clients.
 - `type c`: Print the message in capital
 - `type l`: Print the message in the lowercase (l: stands for lowercase of "L")
- Suppose that your program is called "cs".
 - Execute: `java cs`
 - Command:

```
# add c1 type=c
# add c2 type=l
# send "Hello World"
# del c1
# send "Hello World Again"
```

Output:

```
c1 HELLO WORLD
c2 hello world
c2 hello world again
```



Submission

- One person in one group.
- Write down a brief report in 2 pages to describe how you do and what you do/learn, including:
 - 1. Explain how you work on this project or how you make this project work.
 - 2. Depict and explain the framework or relation of objects in your program.
 - Block diagrams and UML-like diagrams are acceptable.
 - 3. What you learned and what are the main problems that you encounter?
- Compress your report, source code, and the executable/binary files in a compressed file named with **OOAD-P2-FullStudentID(YourName)**
- Email your project to TA coolivan3@outlook.com (徐雅姿) with the email title: **OOAD Project 2, FullStudentID(YourName)**
 - Write emails to ask TA if you have any question about this project.
- For the delayed submission, deduct two points for each day.
- The deadline is two weeks later (**2013/3/10 24:00**).