Command 패턴

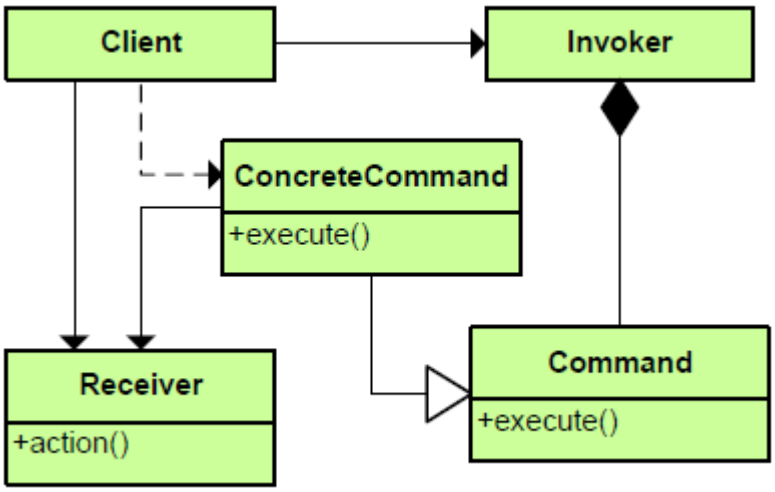
타입

Behavioral Pattern

문제

해결

클래스 다이어그램



예제

command.cpp

```
#include <iostream>

using namespace std;

class Giant
{
public:
    Giant()
    {
        m_id = s_next++;
    }
    void fee()
```

```
    {
        cout << m_id << "-fee  ";
    }
    void phi()
    {
        cout << m_id << "-phi  ";
    }
    void pheaux()
    {
        cout << m_id << "-pheaux  ";
    }
private:
    int m_id;
    static int s_next;
};
int Giant::s_next = 0;

class Command
{
public:
    typedef void(Giant:: *Action)();
    Command(Giant *object, Action method)
    {
        m_object = object;
        m_method = method;
    }
    void execute()
    {
        (m_object->*m_method)();
    }
private:
    Giant *m_object;
    Action m_method;
};

template <typename T>
class Queue
{
public:
    Queue()
    {
        m_add = m_remove = 0;
    }
    void enque(T *c)
    {
        m_array[m_add] = c;
        m_add = (m_add + 1) % SIZE;
    }
    T *deque()
    {
```

```
        int temp = m_remove;
        m_remove = (m_remove + 1) % SIZE;
        return m_array[temp];
    }
private:
    enum
    {
        SIZE = 8
    };
    T *m_array[SIZE];
    int m_add, m_remove;
};

int main()
{
    Queue<Command> que;
    Command *input[] =
    {
        new Command(new Giant, &Giant::fee), new Command(new Giant,
&Giant::phi),
        new Command(new Giant, &Giant::pheaux), new Command(new Giant,
&Giant
        ::fee), new Command(new Giant, &Giant::phi), new Command(new
Giant,
        &Giant::pheaux)
    };

    for (int i = 0; i < 6; i++)
        que.enque(input[i]);

    for (int i = 0; i < 6; i++)
        que.deque()->execute();
    cout << '\n';
}
```

참고

http://en.wikibooks.org/wiki/C%2B%2B_Programming/Code/Design_Patterns#Command