```cpp
#include <iostream>
#include "Page.cpp"
#include "Paragraph.cpp"
#include "Span.cpp"
#include "Image.cpp"

int main() {
    std::shared ptr<Page> page = std::make shared<Page>();
    std::shared ptr<Paragraph> p1 = std::make shared<Paragraph>();
    p1->addElement(std::make shared<Span>());
    p1->addElement(std::make shared<Span>());
    std::shared ptr<Paragraph> p2 = std::make shared<Paragraph>();
    p2->addElement(std::make shared<Image>());
    page->addElement(p1);
    page->addElement(p2);
    std::cout << page->render();
}


/* Image.cpp */
#include "PageElement.cpp"

class Image : public PageElement {
public:
    std::string render() {
        return "<img src=\"IMG\"/>";
    }
};

/* Page.cpp */
#include <forward list>
#include <list>
#include "PageElement.cpp"

class Page : public PageElement {
private:
    std::list< std::shared ptr<PageElement> >  mylist;
public:
    void addElement(std::shared ptr<PageElement> e){
       mylist.push back(std::move(e));
    }

    std::string render() {
        std::string rendered = "<page>\n";
        for (std::shared ptr<PageElement> e :  mylist)
            rendered += e->render() + "\n";
        return rendered + "</page>";
    }
};
/* PageElement.cpp */
#pragma once

class PageElement {
public:
    virtual std::string render() = 0;
};
```

```
65  /* Paragraph.cpp */
66  #include <forward list>
67  #include <list>
68  #include "PageElement.cpp"
69
70  class Paragraph : public PageElement {
71  private:
72      std::list< std::shared ptr<PageElement> >  mylist;
73  public:
74      void addElement(std::shared ptr<PageElement> e){
75         mylist.push back(std::move(e));
76      }
77
78      std::string render() {
79          std::string rendered = "<p>\n";
80          for (std::shared ptr<PageElement> e :  mylist)
81              rendered += e->render() + "\n";
82          return rendered + "</p>";
83      }
84  };
85
86  /* Span.cpp */
87  #include "PageElement.cpp"
88
89  class Span : public PageElement {
90  public:
91      std::string render() {
92          return "<span>TEXT</span>";
93      }
94  };
95
```