```
/* The following code example is taken from the book
 * "The C++ Standard Library - A Tutorial and Reference, 2nd Edition"
 * by Nicolai M. Josuttis, Addison-Wesley, 2012
 *
 * (C) Copyright Nicolai M. Josuttis 2012.
 * Permission to copy, use, modify, sell and distribute this software
 * is granted provided this copyright notice appears in all copies.
 * This software is provided "as is" without express or implied
 * warranty, and with no claim as to its suitability for any purpose.
 */
#include <iostream>
#include <string>
#include <set>
#include <deque>
#include <algorithm>
#include <memory>

class Item {
  private:
    std::string name;
    float  price;
  public:
    Item (const std::string& n, float p = 0) : name(n), price(p) {
    }
    std::string getName () const {
        return name;
    }
    void setName (const std::string& n) {
        name = n;
    }
    float getPrice () const {
        return price;
    }
    float setPrice (float p) {
        price = p;
    }
};

int main()
{
    std::vector<std::reference_wrapper<Item>> books;  // elements are references

    Item f("Faust",12.99);
    books.push_back(f);    // insert book by reference

    // print books:
    for (const auto& book : books) {
        std::cout << book.get().getName() << ": "
                  << book.get().getPrice() << std::endl;
    }

    f.setPrice(9.99);   // modify book outside the containers
    std::cout << books[0].get().getPrice() << std::endl;  // print price of
inserted book

    // print books using type of the elements (no get() necessary):
```

```
    for (const Item& book : books) {
        std::cout << book.getName() << ": " << book.getPrice() << std::endl;
    }
}
```