

Міністерство освіти і науки України  
Центральноукраїнський національний технічний університет  
Механіко-технологічний факультет  
Кафедра кібербезпеки та програмного забезпечення  
Дисципліна: Об'єктно-орієнтоване програмування

**Лабораторна робота №6**  
**Тема: «ШАБЛОНИ В С++»**

Виконав: ст. гр. КН-24  
Куріщенко П. В.  
Перевірив: асистент  
Козірова Н. Л.

**Мета:** ознайомитись з основними поняттями шаблоні та навчитись їх програмно реалізовувати мовою C++.

### **Варіант – 14 - 10 = 4**

Створи програму для керування складом товарів у магазині. Кожен товар має назву, ціну та кількість на складі. Необхідно створити клас Product, який містить ці дані.

Крім того, створи шаблонний клас Inventory, який представляє інвентар магазину. Цей клас має методи для додавання товару на склад, видалення товару зі складу за назвою, виведення списку всіх товарів та пошуку товару за ціною.

Забезпеч універсальність класу Inventory, використовуючи шаблонний параметр для типу товару. Це дозволить тобі працювати з різними типами товарів (наприклад, продукти харчування, електроніка, одяг тощо).

Для демонстрації роботи програми:

1. Створи об'єкт Inventory для зберігання товарів на складі.
1. Додай до інвентаря декілька товарів різних типів.
2. Виведи на екран список всіх товарів на складі.
3. Знайди та виведи на екран всі товари за певною ціною.
4. Видали з інвентаря товар за назвою.
5. Знову виведи на екран оновлений список товарів на складі.

Не забудь використовувати шаблонний клас Inventory для зберігання товарів будь-якого типу. Також, забезпеч, щоб клас Product мав методи доступу до своїх полів (назва, ціна, кількість).

## Результат:

```
*All items*
Name: Apple, price: 17.5uah, amount: 120
Name: Laptop, price: 25000uah, amount: 5
Name: Book, price: 299uah, amount: 45
Name: Backpack, price: 899uah, amount: 20

Enter price to find an item>> 25000
Name: Laptop, price: 25000uah, amount: 5

Enter name to remove an item>> Book

*Changed items*
Name: Apple, price: 17.5uah, amount: 120
Name: Laptop, price: 25000uah, amount: 5
Name: Backpack, price: 899uah, amount: 20
```

## Листинг product.h:

```
#ifndef PRODUCT_H
#define PRODUCT_H

#include <QString>
#include <iostream>

using namespace std;

class Product
{
public:
    Product(const QString&, double, unsigned short);

    QString getName() const;
    double getPrice() const;

    void printItem() const;

private:
    QString name;
    double price;
    unsigned short amount;
};

#endif // PRODUCT_H
```

## Листинг product.cpp:

```
#include "product.h"

Product::Product(const QString& n, double p, unsigned short a)
    : name(n), price(p), amount(a) {}
```

```

QString Product::getName() const{ return name; }

double Product::getPrice() const{ return price; }

void Product::printItem() const{
    cout << "Name: " << name.toStdString()
    << ", price: " << price << "uah"
    << ", amount: " << amount << endl;
}

```

## Лістинг inventory.h:

```

#ifndef INVENTORY_H
#define INVENTORY_H

#include <QList>
#include <iostream>

using namespace std;

template <typename T>
class Inventory
{
    QList<T> items;

public:
    Inventory() = default;

    Inventory(const initializer_list<T>& list) {
        for (const auto& item : list) items.append(item);
    }

    void addItem(const T& item) { items.append(item); }

    void removeByName(const QString& name) {
        for (int index = 0; index < items.size(); ++index)
            if (items[index].getName() == name) {
                items.removeAt(index);
                return;
            }
        cout << "Item wasn't found(" << endl;
    }

    void showAllItems() const {
        if (items.isEmpty()) {
            cout << "List of items is empty" << endl;
            return;
        }
        for (const auto& item : items) item.printItem();
    }

    void findByPrice(double price) const {
        bool found = false;
        for (const auto& item : items)
            if (item.getPrice() == price) {
                item.printItem();
                found = true;
            }
    }
}

```

```

    if (!found) cout << "Item wasn't found(" << endl;
}
};

#endif // INVENTORY_H

```

### Лістинг main.cpp:

```

#include "product.h"
#include "inventory.h"

int main()
{
    double price = 0;
    string name;
    Inventory<Product> store = {
        Product("Apple", 17.5, 120),
        Product("Laptop", 25000.0, 5),
        Product("Book", 299.0, 45),
        Product("Backpack", 899.0, 20)
    };

    cout << "* All items *" << endl;
    store.showAllItems();

    cout << "\nEnter price to find an item>> ";
    cin >> price;
    store.findByPrice(price);

    cout << "\nEnter name to remove an item>> ";
    cin >> name;
    store.removeByName(QString::fromStdString(name));

    cout << "\n* Updated items *" << endl;
    store.showAllItems();

    return 0;
}

```