

---

# **КУРСОВА РОБОТА “ІГРАШКОВИЙ МАГАЗИН”**

**ВИКОНАВ: СОЛОДОВНИК ІЛЛЯ**

---



# Мета та актуальність

**Мета:** ПЗ для управління іграшковим магазином.

**Актуальність:** Автоматизація підвищує ефективність та знижує витрати.

```
"container">
class="row">
v class="col-md-6 col-lg-8"> <!--
<nav id="nav" role="navigation">
  <ul>
    <li><a href="index.html">Home</a>
    <li><a href="home-events.html">Home Events
    <li><a href="multi-col-menu.html">Multi-Column Menu
    <li class="has-children"> <a href="#">Dropdown Menu
      <ul>
        <li><a href="tall-button.html">Tall Button
        <li><a href="image-logo.html">Image Logo
        <li class="active"><a href="#">Active Item
      </ul>
    </li>
    <li class="has-children"> <a href="#">Another Dropdown
      <ul>
        <li><a href="variable-width.html">Variable Width
```



# Предмет роботи

Вивчення ООП на C++ для автоматизації магазину. Система включає модулі товарів, клієнтів, замовлень, звітів. Увага до гнучкості, масштабованості, взаємодії модулів.

```
17 string sInput;  
18 int iLength, iN;  
19 double dblTemp;  
20 bool again = true;  
21  
22 while (again) {  
23     iN = -1;  
24     again = false;  
25     getline(cin, sInput);  
26     system("cls");  
27     stringstream(sInput) >> dblTemp;  
28     iLength = sInput.length();  
29     if (iLength < 4) {  
30         again = true;  
31     }  
32 }
```



# Використані технології та інструменти

Мова: C++.

Середовище: Microsoft Visual Studio 2022.

Бази даних: Текстові файли.

Тестування: "Чорна скринька", "Біла скринька", Unit.



# Архітектура програми

Модульна структура.

Клас "toy" – основний елемент. Реалізовано основні операції (додавання, редагування, видалення, пошук, сортування)

```
out << "Enter rows and columns for second matrix: ";
cin >> r2 >> c2;

// entering elements of first matrix.
out << endl << "Enter elements of matrix 1:" << endl;
for (i = 0; i < r1; ++i)
    for (j = 0; j < c1; ++j)
    {
        out << "Enter element a" << i + 1 << j + 1 << " : ";
        cin >> a[i][j];
    }

// entering elements of second matrix.
out << "Enter elements of matrix 2:" << endl;
for (i = 0; i < r2; ++i)
    for (j = 0; j < c2; ++j)
    {
        out << "Enter element b" << i + 1 << j + 1 << " : ";
        cin >> b[i][j];
    }
```



# Функціонал продавця

Повний контроль над каталогом та продажами:

- Керування товарами.
- Відстеження продажів.
- Пошук та сортування.
- Коригування запасів, сповіщення.
- Збереження даних

```
resp_iter = self.stub.GetResponse()
statuses = {}
for data in resp_iter:
    status = Status(
        status_id=data.id, name=data.name
    )
    statuses[status_id] = status
```

# Функціонал покупця



## Покупець може:

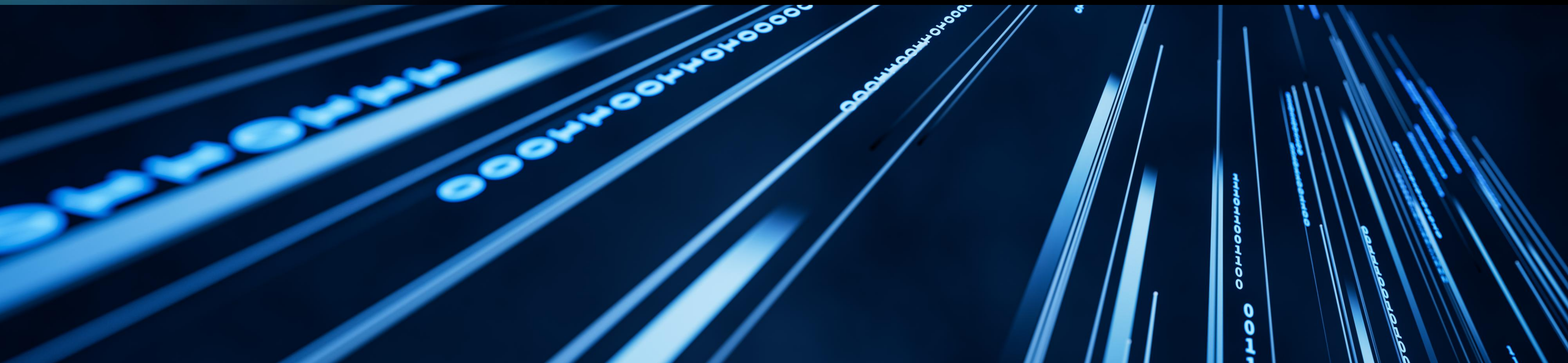
- Переглядати каталог.
- Додавати товари до кошика (з перевіркою наявності).
- Керувати кошиком.
- Оформляти замовлення. Система оновлює кількість товару.



# Тестування

Методи: "Чорна скринька", "Біла скринька", Модульне тестування.

Виправлені помилки: збереження даних, розрахунки цін, сортування. Тестування підвищило якість.





- Графічний інтерфейс (GUI).
- Авторизація.
- Розширена аналітика.
- Справжня БД (SQLite, MySQL).
- Розширений функціонал товарів.
- Управління замовленнями.
- Онлайн-оплата.



# Можливості вдосконалення





---

Програма "Іграшковий магазин" на C++  
автоматизує процеси. Реалізовано  
функціонал для продавця та покупця.  
Дані зберігаються у файлах, тестування  
усунуло помилки. Демонструє розуміння  
ООП та навички розробки ПЗ.

# Результати та ВИСНОВКИ





**ДЯКУЮ ЗА УВАГУ!!!**