

STL

Об'єктно-орієнтоване програмування

Задача 1: Керування рецептами

Реалізуйте програму для керування рецептами. Кожен рецепт має назву, список інгредієнтів та інструкції з приготування. Використовуйте клас для представлення рецепту, `std::vector` для зберігання рецептів та функцію `find_if` для пошуку рецептів за інгредієнтами або назвою.



Інструкції:

- Створіть клас `Recipe` з властивостями `назва`, `список інгредієнтів` та `інструкції`.
- Використайте `std::vector<Recipe>` для зберігання рецептів.
- Напишіть функції для додавання, видалення, пошуку та виведення рецептів.
- Використовуйте функцію `find_if` для пошуку рецептів за інгредієнтами чи назвою рецепту.

Задача 2: Система обміну повідомленнями з використанням `std::weak_ptr`

Ви розробляєте систему обміну повідомленнями між користувачами. Кожен користувач може відправляти повідомлення і отримувати їх. Кожне повідомлення має відправника, який також є користувачем системи. Для уникнення циклічних посилань між користувачами та повідомленнями використовуйте `std::weak_ptr`.



Класи:

User:

- Має ім'я (`std::string name`).
- Зберігає отримані повідомлення у векторі `std::vector<std::shared_ptr<Message>> receivedMessages`.
- Має метод `sendMessage(const std::string& content)`, який створює нове повідомлення і відправляє його в систему.
- Має метод `receiveMessage(const std::shared_ptr<Message>& msg)`, який додає отримане повідомлення до свого списку отриманих.
- Має метод `printReceivedMessages()`, який виводить всі отримані повідомлення користувача на консоль.
- Використовує `std::weak_ptr<User>` у класі `Message` для зберігання посилання на відправника.

Message:

- Має вміст (`std::string content`) та відправника (`std::weak_ptr<User> sender`).
- Має метод `getContent()`, який повертає вміст повідомлення.
- Має метод `getSender()`, який повертає відправника у вигляді `std::shared_ptr<User>`. Використовуйте `sender.lock()` для отримання `shared_ptr`, щоб звернутися до об'єкту `User`.

Головна програма:

- Створіть декілька об'єктів User.
- Використайте методи `sendMessage` та `printReceivedMessages` для взаємодії між користувачами та виведенням отриманих повідомлень.