МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. І. Сікорського

Кафедра

Автоматизованих Систем Обробки Інформації та Управління

**КУРСОВА РОБОТА**

з Програмування. «Структури даних та алгоритми»

на тему:  
Онлайн-ресурс клубу любителів

туризму

Студента 1-го курсу, групи ІС-91

Хмелiнiна Андрiя Михайловича

Спеціальності 126  
«Інформаційні управляючі системи та технології»

Керівник ст.вик. Вітковська І. І.

Кількість балів: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Національна оцінка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |  |
| --- | --- | --- | --- |
| Члени комісії |  |  |  |
|  | (підпис) |  | (вчене звання, науковий ступінь, прізвище та ініціали) |
|  |  |  |  |
|  | (підпис) |  | (вчене звання, науковий ступінь, прізвище та ініціали) |

Київ – 2020 рік

# **ЗМІСТ**

Вступ

1. Основна чатина

А. Основнiзавдання програми

Б. Функцiональнi вимоги до програми

1. Календарний план
2. UML дiаграма класiв
3. Таблиця з описом методiв програми
4. Iнструкцiя по використанню
5. Тестування
6. Висновок

# **Вступ**

# ПОСТАНОВКА ЗАДАЧІ

1. Спроектувати об’єктно-орієнтовану модель предметної галузі згідно з варіантом, визначивши необхідні для цього класи та їх структуру. При об’єктно-орієнтованому проектуванні продемонструвати застосування всіх видів відношень.
2. Написати програму мовою С#, у якій реалізувати попередньо спроектовану об’єктно-орієнтовану модель.
3. Предметну галузь реалізувати окремим проектом – динамічною бібліотекою (С#).
4. Інтерфейс користувача, наприклад, введення\виведення з консолі, реалізовувати окремим проектом (С#). Код інтерфейсу має бути простим (демонструється використання класів предметної галузі шляхом створення об’єктів та їх застосування).

Клуб любителів туризму

В клубі надається інформація про країни, що цікаві для туризму,

тематики турів, туристичні тури по певних місцях, туристичні

агенції, що надають послуги таких турів. Користувач сервісу

клубу може обрати собі тур та агенцію згідно його вподобань до

запланованої дати подорожі.

Основна частина

Програма «Туристичний клуб» буде використовуватись для керуванням власним туристичним клубом та для того щоб ним користуватись.

Основні завдання нашої програми:

* допомагати користувачеві поступати до туристичного клубу;
* адмiнiструвати клубом;
* надавати iнформацiю про тур-агенства та тури;
* можливiсть додавати користувачевi тури за характеристиками по бажанню користувача

Також ми повиннi передбачити те, що користувач може вводити недопустимi даннi:

* замiсть чисел вводиться знак або лiтера;
* некоректна вiдповiдь на питання формату так/нi
* при видаленнi користувача з клубу, сам клуб повинен пропасти у користувача

При виборi туру користувачу потрiбно опиратися на якiсь характеристики. Саме для цього буде реалiзована можливiсть для користувача вибору туру за такими параметрами:

- максимальна цiна

- агенство, яке предоставляє тур

- дата туру

- тематика туру

- країна туру

Функціональні вимоги до програми

Для користувача має iснувати можливiсть «iцiалiзуватися» у системi. Користувач повинен мати змогу як вступити до туристичного клубу, так i вийти з нього. Якщо користувач виходить з клубу або його выдаляє сам клуб то вiн має отримати повiдомлення.

Щоб адмiнiстрацiя клубу не створила туру з вiд’ємною цiною буде викликатися повiдомлення про помилку та цiна буде встановлюватися за замовчуванням 7777777, поки адмiнicтрацiя не перестворить тур, цiна буде залишатися такою високою, щоб такий тур не замовляли користувачi, помилка буде внесена до log файлу. Коли ж буде траплятися помилка щодо вiдповiдей типу так/нi, користувачу також виведе повiдомлення i питання задається знову.

Користувач може передивлятись:

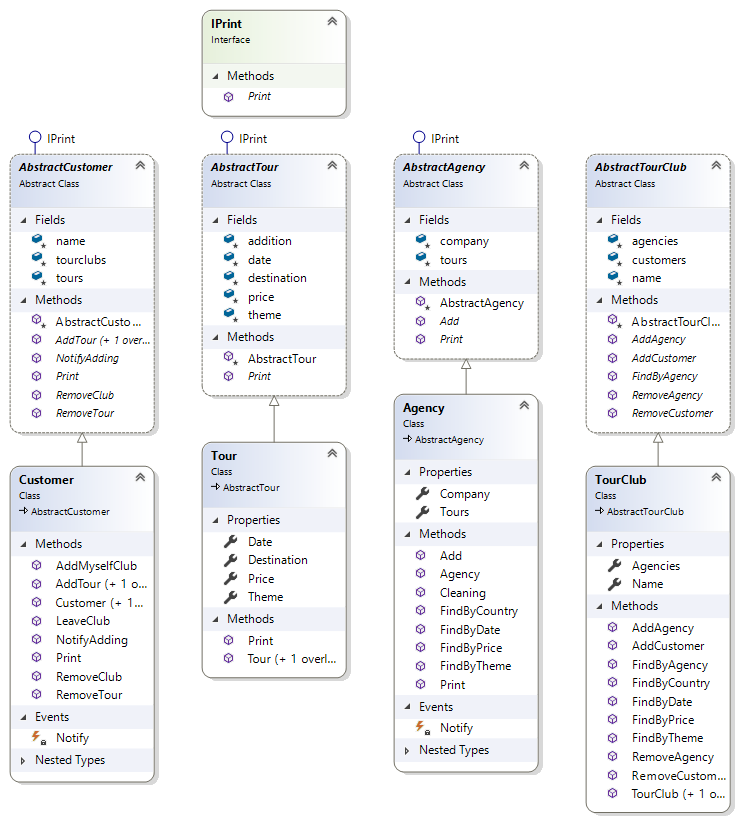
1. Агенства, тури яких вiн внiс до свого кошика
2. Самi тури та їх параметри:
3. Країна туру
4. Тематика туру
5. Цiна туру
6. Дата туру
7. Своє iм’я

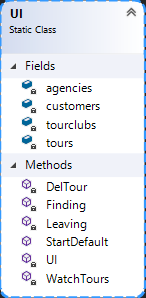
Програма реалiзована таким чином, що з часом можна її доповнювати змiни адже для кожного об’єкта є абстрактний клас батько, змiни можна робити доповнюючи новими версiями старих об’єктiв.

Календарний план

|  |  |  |  |
| --- | --- | --- | --- |
|  | Назва етапів курсової роботи | Термін виконання етапів роботи | Підпис керівника, студента |
| 1 | Отримання теми курсової роботи |  |  |
| 2 | Підготовка ТЗ |  |  |
| 3 | Аналіз предметної області |  |  |
| 4 | Проектування архітектури програмної системи |  |  |
| 5 | Розробка сценарію роботи програми |  |  |
| 6 | Узгодження з керівником інтерфейсу користувача |  |  |
| 7 | Розробка програмного забезпечення |  |  |
| 8 | Узгодження з керівником плану тестування |  |  |
| 9 | Тестування програми |  |  |
| 10 | Підготовка пояснювальної записки |  |  |
| 11 | Здача курсової роботи на перевірку |  |  |
| 12 | Захист курсової роботи |  |  |

UML дiаграма класiв





Стандартні методи класів

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № п/п | Назва класу | Назва функції | Призначення функції | Опис вхідних параметрів | Опис вихідних параметрів |
| 1 | **Tour** | Print | Виведення у консоль всiх параметрiв класа Tour | - | - |
| 2 | **Agency** | Add | Добавляє ще один об’єкт типу Tour до списку | об’єкт типу **Tour** | - |
|  |  | Cleaning | Видаляє неактуальнi тури, якi вже неактуальнi бiльше мiсяця | - | - |
|  |  | FindByCountry | Пошук туру за країною | Назва країни у форматi **string** | Список турiв якi були знайденнi  **List<Tour>** |
|  |  | FindByDate | Пошук туру за датою | Дата туру у форматi **DateTime** з роком, мiсяцем та днем | Список турiв якi були знайденнi  **List<Tour>** |
|  |  | FindByPrice | Пошук туру за максимальною цiною | Максимальна цiна у форматi **double** | Список турiв якi були знайденнi  **List<Tour>** |
|  |  | FindByTheme | Пошук туру за тематикою | Тематика туру у форматi **string** | Список турiв якi були знайденнi  **List<Tour>** |
|  |  | Print | Виведення у консоль назву компанiї та для кожного туру зi списку викликає метод Print | - | - |
|  |  | подiя Notify | Подiя, яка одразу ж пiдписана на лямбда вираз який виводить рiзноманiтнi повiдомлення для користувача | Повiдомлення для користувача типу **string** | - |
| 3 | **Customer** | Print | Виведення у консоль iм’я користувача та для кожного туру з його списку викликає метод Print | - | - |
|  |  | RemoveClub | Викликається коли користувача видаляють з туристичного клубу, викликає повiдомлення та видаляє зi списку той клуб, з якого вигнали користувача | Клуб, з якого виганяють користувача типу **TourClub** | - |
|  |  | LeaveClub | Користувач може сам выйти з клубу | Клуб, з якого виходить користувач типу **TourClub** | - |
|  |  | RemoveTour | Видалення туру зi списку по номеру | Номер туру зi списку типу **int** | - |
|  |  | AddTour | Додавання туру або турiв до списку | Тур типу **Tour** або список турiв типу **List<Tour>** | - |
|  |  | NotifyAdding | Викликається при додаваннi користувача до клубу та викликає повiдомлення | Клуб, до якого додають користувача | - |
|  |  | AddMyselfClub | Додає користувача до клубу за його бажанням | Клуб, до якого додає себе користувач | - |
| 4 | **TourClub** | AddAgency | Додає до тур-агенцiю до списку | Тур-агенцiя типу **Agency** | - |
|  |  | AddCustomer | Додає користувача туристичного клубу до списку | Користувач типу **Customer** | - |
|  |  | RemoveAgency | Видаляє тур-агенцiю зi списку | Тур-агенцiя типу **Agency** | - |
|  |  | RemoveCustomer | Видалення користувача з туристичного клубу | Користувач типу **Customer** | - |
|  |  | FindByAgency | Шукає тури якi є у конкрутнiй тур-агенцiї | Iм’я компанiї тур-агенцiї типу **string** | Список турiв якi були знайденнi  **List<Tour>** |
|  |  | FindByCountry | Пошук туру за країною (викликається по всiх тур-агенцiях) | Назва країни у форматi **string** | Список турiв якi були знайденнi  **List<Tour>** |
|  |  | FindByDate | Пошук туру за датою (викликається по всiх тур-агенцiях) | Дата туру у форматi **DateTime** з роком, мiсяцем та днем | Список турiв якi були знайденнi  **List<Tour>** |
|  |  | FindByPrice | Пошук туру за максимальною цiною (викликається по всiх тур-агенцiях) | Максимальна цiна у форматi **double** | Список турiв якi були знайденнi  **List<Tour>** |
|  |  | FindByTheme | Пошук туру за тематикою (викликається по всiх тур-агенцiях) | Тематика туру у форматi **string** | Список турiв якi були знайденнi  **List<Tour>** |
| 5 | **Program** | WatchTours | Питає користувача чи показати список турiв, якi в нього є | Користувач типу **Customer** | - |
|  |  | Finding | Питає користувача чи робити пошук турiв, i за якими параметрами | Користувач типу **Customer**, туристичний клуб типу **TourClub** | Повертає значення типу **bool** яке свiдчить про те чи потрiбно питати користувача ще |
|  |  | Leaving | Питає користувача чи хоче вiн вийти з клубу | Користувач типу **Customer**, туристичний клуб типу **TourClub** | - |
|  |  | DelTour | Питає користувача чи хоче вiн видалити деякий тур зi списку | Користувач типу **Customer** | - |
|  |  | Start | Основний метод, де iнiцiалiзуються тури, туристичнi клуби та за бажанням користувача, сам користувач, несе у собi основний користувацький iнтерфейс, i просто викликається у Main() | - | - |
| 6 | **UI**  **(користувацький iнтерфейс)** | WatchTours | Питає користувача, чи хоче вiн подивитись тури якi обрав | Користувач типу **Customer** | - |
|  |  | Finding | Питає користувача про те чи хоче вiн обрати тури | Користувач типу **Customer**, туристичний клуб типу **TourClub** | - |
|  |  | Leaving | Питає користувача чи хоче вiн вийти з туристичного клубу | Користувач типу **Customer**, туристичний клуб типу **TourClub** | - |
|  |  | DelTour | Питає користувача чи хоче вiн видалити тури зi свого списку | Користувач типу **Customer** | - |
|  |  | StartDefault | Функцiя старту по замовчуванню: створює всi необхiднi об’єкти та заповнює їх за замовчуванням, та починає питати користувача | - | - |

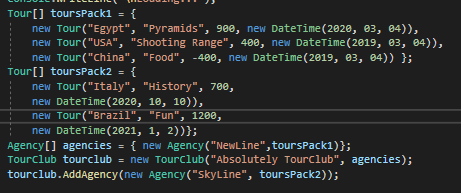
У таблицi вiдсутнi Абстрактнi класи, бо в них зазвичай є тiльки чисто вiртуальнi методи , якi реалiзуються у нащадках

Iнструкцiя по використанню

1. Створити користувача у программi
2. Вступити до клубу за бажанням
3. Вибирати тури за бажанням
4. Пiсля закiнчення, можна видалити свого користувача

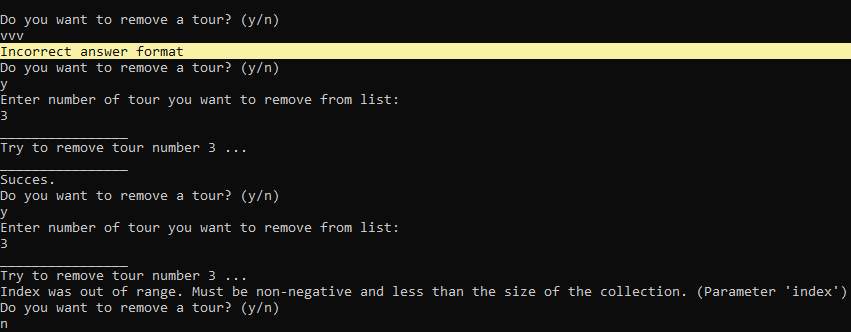
Тестування

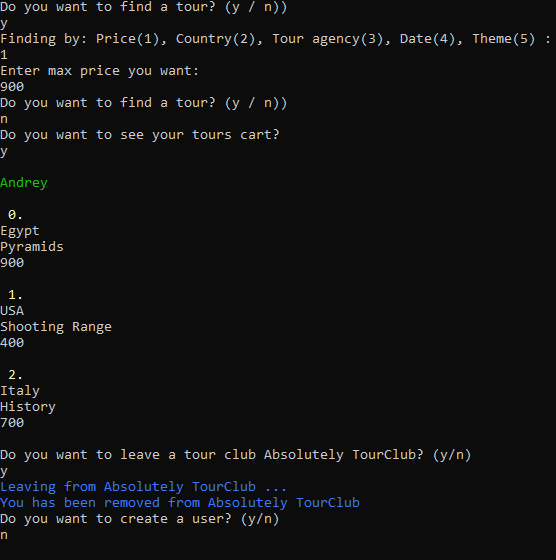
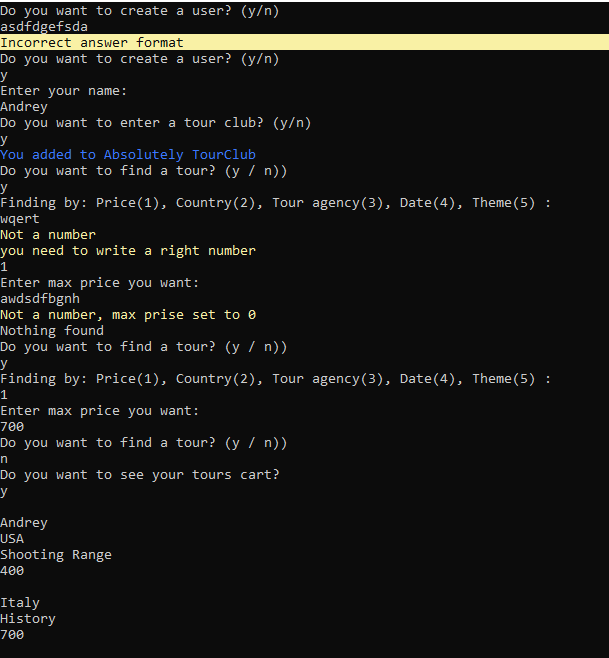
1. Iнiцiалiзацiя тур клубiв, турiв та тур-агенцiй







1. Перевiрка на те, якщо користувач вводить не коррекнi вiдповiдi

.

Висновок

Головною цiллю курсової роботи було спроектувати об’єктно-орієнтовану модель предметної галузі згідно з варіантом, визначивши необхідні для цього класи та їх структуру. При об’єктно-орієнтованому проектуванні продемонструвати застосування всіх видів відношень. А саме написати програму клуб любителів туризму, де надається інформація про країни, що цікаві для туризму, тематики турів, туристичні тури по певних місцях, туристичні агенції, що надають послуги таких турів. Зробити так, щоб користувач сервісу клубу мiг обрати собі тур та агенцію згідно його вподобань до запланованої дати подорожі.

Проблеми, з якими я стовкнувся: правильне конвертування введеного тексту до консолi у чисельний тип, неправильне iндексування по списку, з чого випливали помилки Out of range.

З усiма проблемами я справився за допомогою iнтернет ресурсiв та лiтератури:

Список допомiжних ресурсiв та лiтератури:

1. <https://metanit.com/sharp/tutorial/>
2. <https://docs.microsoft.com/en-us/dotnet/csharp/>