**ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ**

**«ОПТИКО-МЕХАНІЧНИЙ ФАХОВИЙ КОЛЕДЖ КИЇВСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ ІМЕНІ ТАРАСА ШЕВЧЕНКА»**

Циклова комісія програмування та інформаційних технологій

**КУРСОВА РОБОТА**

**з дисципліни «Інструментальні засобі візуального програмування»**

на тему: «Проектування прикладного програмного забезпечення

“Каса автостанції”»

|  |  |  |
| --- | --- | --- |
| ЗАХИЩЕНО З ОЦІНКОЮ  Національна шкала \_\_\_\_\_\_\_\_\_\_\_\_  Кількість балів \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Керівник  \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Таран В.В.  (підпис) |  | ВИКОНАВ |
| студент 3-го курсу гр. ІПЗ-31 спеціальність 121 «Інженерія програмного забезпечення» |
| Бовкун Павло Олександрович  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (підпис виконавця) |

Члени комісії \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Таран В. В.

(підпис)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Заболотний В.О.

(підпис)

м. Київ – 2022

**ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ**

**«ОПТИКО-МЕХАНІЧНИЙ ФАХОВИЙ КОЛЕДЖ КИЇВСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ ІМЕНІ ТАРАСА ШЕВЧЕНКА»**

ЗАТВЕРДЖУЮ

В.о. директора

\_\_\_\_\_\_\_\_\_\_\_ Гаприндашвілі Б. В.

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2022р.

|  |  |  |
| --- | --- | --- |
| **Дисципліна:** | Інструментальні засоби візуального програмування | |
| **Спеціальність:** | 121 «Інженерія програмного забезпечення» | |
| **Курс ІІІ** | **Група IПЗ-31** | **Семестр** **6** |

**ЗАВДАННЯ  
на курсову роботу студента**

|  |
| --- |
| Бовкуна Павла Олександровича |
| **1.Тема курсової роботи:**  Проектування прикладного програмного забезпечення  «Каса автостанції» |
| Затверджена протоколом циклової комісії № 06 від 25 січня 2022 р. |
| **2.Технічні вимоги:** ЄСПД, ЄСКД, ДСТУ 3008-95 |
| **3.Термін здачі студентом закінченої роботи: «**\_\_» червня 2022 р. |
| **4.Вихідні дані до виконання роботи:** |
| * Проаналізувати предметну область, виконати опис проектованої системи |
| і правил її функціонування; |
| * Розробити графічний інтерфейс користувача для роботи з його |
| елементами: головне вікно програми, система меню форм, довідкова система та інформація про програму; |
| **5.Зміст пояснювальної записки** (перелік питань, які підлягають розробці): |
| вступ; постановка задачі; характеристика апаратно-програмної частини; |
| опис мови і середовища програмування; розробка алгоритму та його опис; |
| створення та тестування програми; висновки; перелік умовних позначень; |
| список літератури; програмний лістинг. |
| **6. Дата видачі завдання:** «05» квітня 2022 р. |

|  |  |  |
| --- | --- | --- |
| Керівник роботи |  | Таран Вікторія Валеріївна |
| Завдання прийняв до виконання |  | Бовкун Павло Олександрович |

«05» квітня 2022 р.

**КАЛЕНДАРНИЙ ПЛАН-ГРАФІК**

**виконання курсової роботи з дисципліни**

**«Інструментальні засоби візуального програмування»**

|  |  |  |
| --- | --- | --- |
| **№ з/п** | **Етапи** | **Термін виконання** |
|  | Обрання та затвердження теми, попередня постановка задачі | 01.04-05.04.2022 |
|  | Підбір літератури, веб-джерел, ознайомлення зі стандартами | 06.04-12.04.2022 |
|  | Вивчення та аналіз предметної області об'єкту курсової роботи | 13.04-19.04.2022 |
|  | Підготовка оглядової частини | 20.04-26.04.2022 |
|  | Розробка архітектури та загальної структури системи | 27.04-03.05.2022 |
|  | Програмна реалізація системи | 04.05-10.05.2022 |
|  | Оформлення роботи, передача її на рецензування керівникові | 11.05-17.05.2022 |
|  | Рецензування роботи, уточнення назви, висновків, підготовка презентаційних матеріалів | 18.05-24.05.2022 |
|  | Попередній захист курсової роботи | 25.05-07.06.2022 |
|  | Захист курсової роботи | 08.06.2022 |

|  |  |  |
| --- | --- | --- |
| Керівник роботи |  | Таран Вікторія Валеріївна |
| Студент |  | Бовкун Павло Олександрович |
|  |  | «\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2022 р. |

**Зміст**

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

4

ОМФК.0121КР.10003ПЗ

Розроб.

Бовкун П.О.

Перевір.

Таран В. В.

Реценз.

Заболотний В. О.

Н. Контр.

3

Затверд.

Проектування прикладного програмного забезпечення «Каса автостанції»

Літ.

Аркушів

28

ІПЗ-31

Реферат 5

Скорочення 6

Вступ 7

1. Постановка задачі 8

2. Проектування структури додатку 9

3. Проектування інтерфейсу додатку 10

4. Опис алгоритму додатку та розробка програмного коду 14

5. Тестування додатку 15

6. Опис використаних програмних засобів ООП 19

Висновок 22

Список використаної літератури 23

Додаток 24

**Реферат**

Зм.

Арк.

№ документу

Підпис

Дата

Арк.

5

ОМФК.0121КР.10003ПЗ

Курсова робота складається із реферату, скорочень, вступу, шістьох розділів, висновків, списку використаної літератури та одного додатку. Кількість використаних джерел літератури – 7. Робота містить 23 рисунки. Загальний обсяг роботи становить 28 сторінок, основний текст викладено на 18 сторінках.

Ключові слова: КАСА, КВИТОК, ТРАНСПОРТ, ВОКЗАЛ.

Темою курсової роботи є проектування прикладного програмного забезпечення «Каса автостанції».

Предметна область – автовокзальні каси.

Мета – покращити досвід користування касою автовокзалу (і не тільки) для користувачів та співробітників каси шляхом автоматизації деяких процесів.

**Скорочення**

Зм.

Арк.

№ документу

Підпис

Дата

Арк.

6

ОМФК.0121КР.10003ПЗ

і т. д. – і так далі

ІЗВП – інструментальні засоби візуального програмування

ООП – об’єктно-орієнтоване програмування

ПЗ – програмне забезпечення

С# – багатопарадигмальна мова програмування «Сі Шарп».

**Вступ**

Зм.

Арк.

№ документу

Підпис

Дата

Арк.

7

ОМФК.0121КР.10003ПЗ

Після дослідження предметної області я з’ясував, що автовокзали потребують програмної системи, яка б змогла покращити досвід використання кас автостанцій для клієнтів і не тільки. Досягненню цієї мети допоможе прикладне ПЗ «Каса автостанції», яке, до того ж, зможе підвищити продуктивність касира шляхом автоматизації певного переліку дій: прийом замовлень квитків та послуг по інтернету з їх сортуванням, угрупованням, фільтруванням тощо з подальшим наданням їх [квитків, послуг] на касі або через додаток (наприклад, в електронному форматі для друку); надання інформації по запиту користувача (довідка, характеристики автобусів тощо) і т. д.

У цій курсовій роботі буде розроблятися прикладне ПЗ за допомогою мови програмування С# із застосуванням принципів ІЗВП та ООП.

Методологія об'єктно-орієнтованого програмування заснована на уявленні програми у вигляді сукупності об'єктів, кожен з яких є екземпляром певного класу, а класи утворюють ієрархію спадкування.

Об'єктно-орієнтоване програмування дозволяє швидко і з мінімальною кількістю помилок написати програму, код якої буде зрозумілим для інших розробників і, отже, легко змінюваним при необхідності. Мова програмування є об'єктно-орієнтованою, коли виконуються такі умови: підтримуються об'єкти або абстракції даних, що мають інтерфейс у вигляді іменованих операцій, та власні дані з обмеженням доступу до них; об'єкти відносяться до відповідних типів; типи (класи) можуть успадковувати атрибути супертипу (базового класу).

1. **Постановка задачі**

Зм.

Арк.

№ документу

Підпис

Дата

Арк.

8

ОМФК.0121КР.10003ПЗ

1. Створити прикладне програмне забезпечення «Каса автостанції», в якому би застосовувались основні принципи об’єктно-орієнтованого програмування.
2. Прикладне програмне забезпечення повинне створюватись мовою програмування C# у середовищі Microsoft Visual Studio.
3. Додаток має дати змогу вибрати та зареєструвати білет на автобус, переглядати характеристики автобусів.
4. **Проектування структури додатку**

Зм.

Арк.

№ документу

Підпис

Дата

Арк.

9

ОМФК.0121КР.10003ПЗ

Структура додатку «Каса автостанції» складається із базового класу Company та похідних класів MainMenu (Головне меню), Registration (Регістрація) та Passenger (Пасажир).

Зв’язок між базовим та похідними класами:

Базовий (батьківський) клас Company містить поля: numberofReis, Prise\_Reis(1-4). Крім того, базовий клас містить і властивості: numberofReis {get,set}, Prise\_Reis(1-4){get,set}.

Похідний клас MainMenu містить метод Menu () : void. Він відповідає за головне меню програми.

Клас Registration містить метод registr() : void в якому реалізується реєстрація білетів покупця.

1. **Проектування інтерфейсу додатку**

Зм.

Арк.

№ документу

Підпис

Дата

Арк.

10

ОМФК.0121КР.10003ПЗ

Додаток «Каса автостанції» виконано у консольному варіанті. Він надає можливість вибрати покупцю білет, який він буде замовляти.

Початок роботи в Microsoft Visual Studio здійснюється запуском на виконання файлу devenv.exe ("C:\Program Files (x86)\Microsoft Visual Studio 10.0\Common7\IDE\devenv.exe").

Як правило, використовується для цього Головне меню: Пуск / Всі програми / Microsoft Visual Studio 2010 / Microsoft Visual Studio 2010.

У головному меню вибираємо: File / New / Project… (рис. 1)

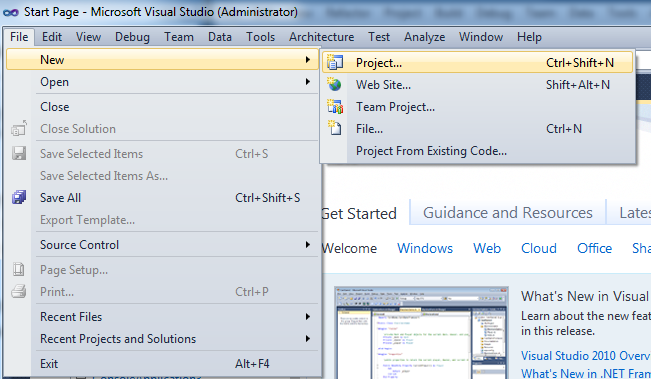
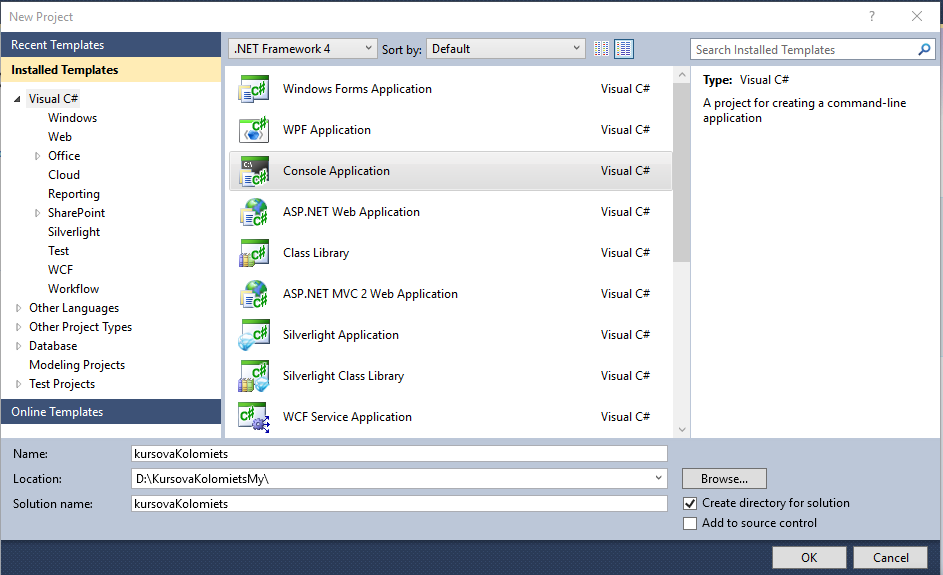


 Рисунок 1. Створення нового проекту.

З’явиться діалогове вікно «New Project» (рис. 2), в якому слід вибрати мову Visual C#, тип додатку Console Application, вказати назву проекту та його розміщення.



Зм.

Арк.

№ документу

Підпис

Дата

Арк.

11

ОМФК.0121КР.10003ПЗ

Рисунок 2. Діалогове вікно «New Project».

Задавши правильно ці обов’язкові параметри, відкривається головний файл Program.cs (рис. 3), в якому міститься метод Main(), який є точкою входу у програму.

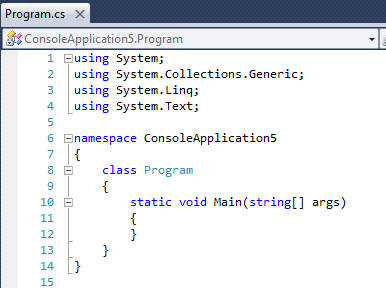
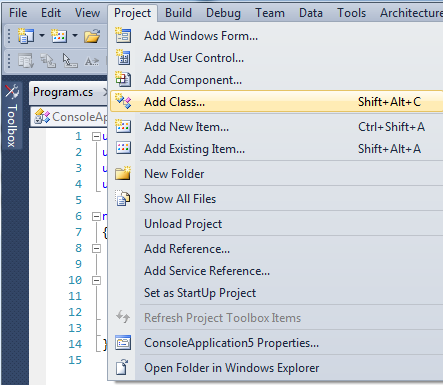


Рисунок 3. Файл Program.cs з методом Main.

Далі по черзі додаємо класи з такими назвами: (рис. 4, 5): Company – базовий клас; Registration, Passenger, MainMenu – похідні класи.



Зм.

Арк.

№ документу

Підпис

Дата

Арк.

12

ОМФК.0121КР.10003ПЗ

Рисунок 4. Додавання класу.

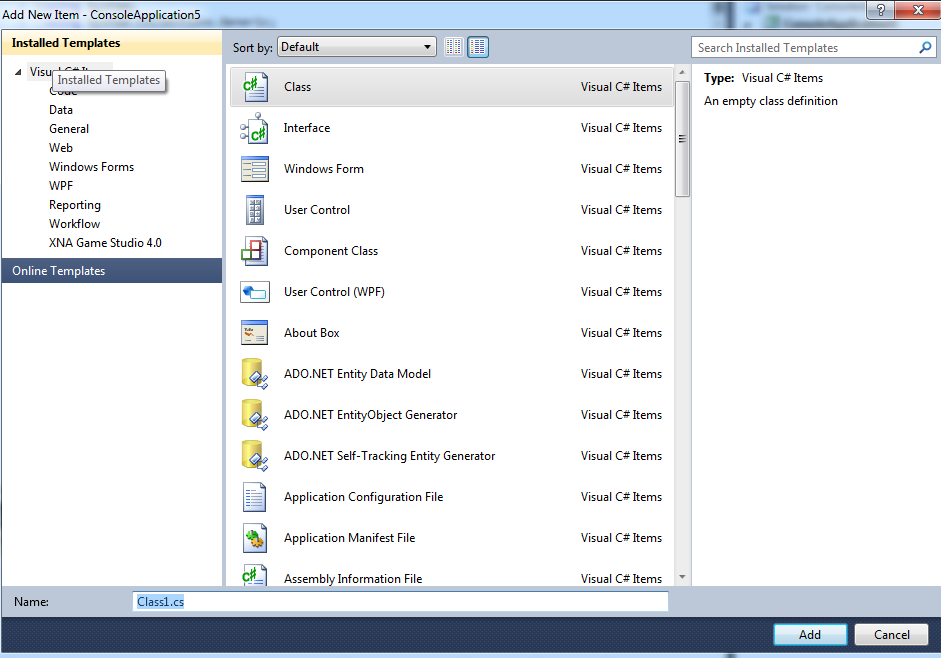


Рисунок 5. Створення назви класу.

Щоб мати можливість користуватись діалоговими вікнами (наприклад, MessageBox.Show (response + " - неправильно введений номер автобуса!");), додамо посилання на бібліотеку System.Windows.Forms.dll через діалогове вікно «Add Reference» (рис. 6) та рядок підключення using System.Windows.Forms;.

Зм.

Арк.

№ документу

Підпис

Дата

Арк.

13

ОМФК.0121КР.10003ПЗ

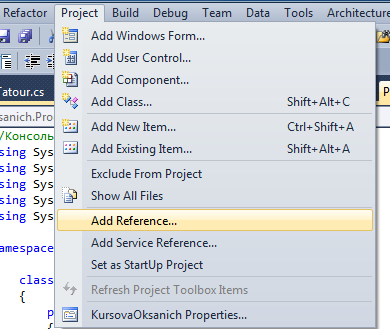


Рисунок. 6. Відкриття вікна «Add Reference».

Переходимо на вкладку .NET, виділяємо System.Windows.Forms і натискаємо Ok (рис. 7).

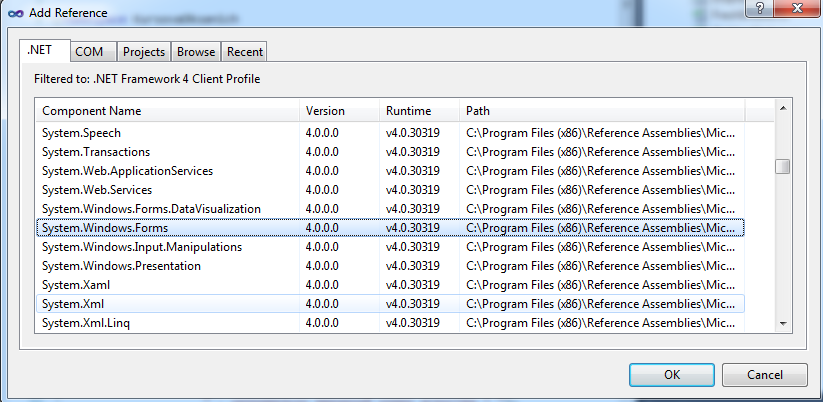


Рисунок 7. Додавання посилання у «Add Reference».

1. **Опис алгоритму додатку та розробка програмного продукту**

Зм.

Арк.

№ документу

Підпис

Дата

Арк.

14

ОМФК.0121КР.10003ПЗ

Див. Додаток.

1. **Тестування додатку**

Зм.

Арк.

№ документу

Підпис

Дата

Арк.

15

ОМФК.0121КР.10003ПЗ

Для проведення тестування програми завантажимо програму на виконання. Після компіляції додатку ми бачимо вікно (рис. 8):

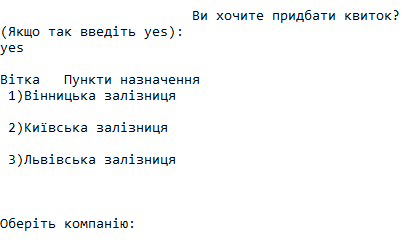


Рисунок 8. Вікно програми (1).

Потрібно вибрати зі списку потрібний вам пункт (рис. 9). Допустимо, ви вибрали пункт 1, далі вам надається 5 варіантів напрямків відправлень, але для детальних характеристик нам потрібно вибрати дані одного з варіантів поїздки, щоб побачити дані автобуса, час польоту і ціну.

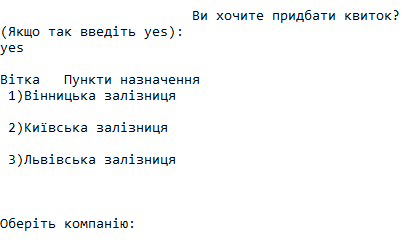


Рисунок 9. Вікно програми (2).

При відмові від поїздки ми отримуємо повідомлення (рис. 10).

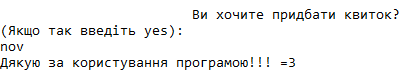
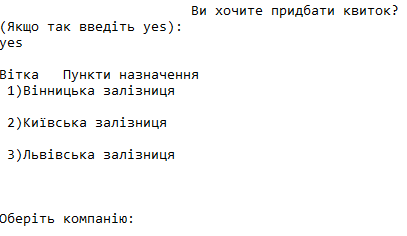


Рисунок 10. Повідомлення при відмові.

Програма продовжує свою роботу до тих пір, поки ми її не вимкнемо або введемо некоректні дані. Якщо введені дані є правильними, то програма запропонує вибрати вам маршрут поїздки (рис. 11, 12).



Зм.

Арк.

№ документу

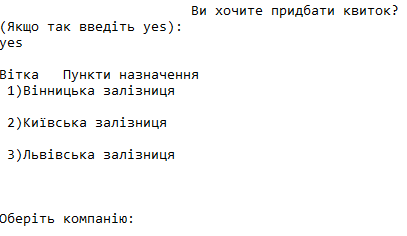
Підпис

Дата

Арк.

16

ОМФК.0121КР.10003ПЗ



Рисунки 11, 12. Робота програми після введення неправильних даних.

Після того, як з’являється вибір маршруту, потрібно вибрати один із трьох напрямків: Вінницька залізниця, Київська залізниця, Львівська залізниця.

Виберемо, наприклад, Київську залізницю (рис. 13).

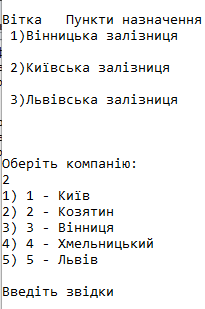


Рисунок 13. Вибір маршруту.

Після чого у вас з’являється 5 напрямків руху.

Якщо ви виберете варіант 2, вам буде запропоновано вибрати напрям руху. Наприклад, виберемо Київ. Після вам буде запропоновано вибрати тип місця (рис. 14).

Зм.

Арк.

№ документу

Підпис

Дата

Арк.

17

ОМФК.0121КР.10003ПЗ

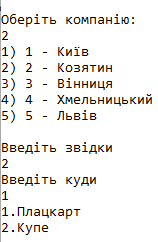
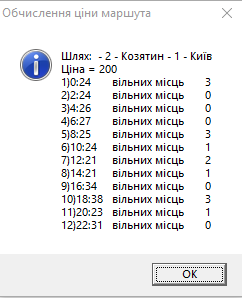


Рисунок 14. Вибір типу місця.

Коли виберемо місце, ми отримаємо повну інформацію про напрям руху, час руху і його вартість (рис. 15).

****

Зм.

Арк.

№ документу

Підпис

Дата

Арк.

18

ОМФК.0121КР.10003ПЗ

Рисунок 15. Вікно з повною інформацією про поїздку.

1. **Опис використаних програмних засобів ООП**

Зм.

Арк.

№ документу

Підпис

Дата

Арк.

19

ОМФК.0121КР.10003ПЗ

Із розвитком програмування виникла ідея поєднати в межах однієї сутності дані і код, що безпосередньо опрацьовує ці дані. Ця сутність отримала назву об’єкт, а відповідний підхід до створення програм – об’єктно-орієнтоване програмування.

ООП – це парадигма програмування, яка розглядає програму як сукупність гнучко пов’язаних між собою об’єктів. Кожен об’єкт має суттєві характеристики (абстракції), які відрізняють його від інших об’єктів. Розрізняють абстракції стану та поведінки об’єкта.

Стан (дані об’єкта) характеризується переліком та значенням певних ознак. Поведінка (функціонал об’єкта) визначається набором операцій, які виконуються об’єктом, або над об’єктом. Кожен об’єкт є екземпляром (представником) певного класу. Відповідно, клас – це відповідна абстракція об’єктів.

Основні переваги концепції ООП: моделювання предметів та явищ реального світу; можливість створювати користувацькі типи даних (класи); приховування деталей реалізації (інкапсуляція); можливість повторного використання коду (наслідування); інтерпретація викликів процедур та функцій на етапі виконання (поліморфізм).

Інкапсулювання – це механізм, що зв’язує в одне ціле функції і дані, якими вони маніпулюють, а також захищає їх від зовнішнього доступу і неправильного застосування. В об’єктно-орієнтованій мові функції і всі необхідні дані можуть зв’язуватись в об’єкти. В межах об’єкта функції та дані можуть бути або закритими для інших об’єктів (private), або захищеними (protected), або відкритими (public).

*Модифікатори доступу.*

Зм.

Арк.

№ документу

Підпис

Дата

Арк.

20

ОМФК.0121КР.10003ПЗ

Модифікатор private забороняє доступ до полів і методів класу поза самого класу, тобто вони будуть доступні тільки в межах самого класу. Модифікатор public відкриває доступ поза оголошенням базового класу чи похідних класів. Метод, оголошений із іншого класу, може вільно звертатись до таких полів і методів. Модифікатор protected відкриває доступ до полів і методів базового класу тільки для похідних класів, але не для зовнішніх по відношенню до класу методів. Модифікатор internal представляє комбінацію модифікаторів public і protected. Він обмежує доступ відмічених ним полів і методів межами одної збірки.

Успадкування – це властивість, з допомогою якої один об’єкт може набувати властивостей іншого зі збереженням ієрархії. Без використання успадкування кожний об’єкт повинен явно визначати всі свої характеристики, а з наслідуванням об’єкт повинен визначати тільки ті якості, які роблять його унікальним в межах свого класу. Визначення похідного класу може ґрунтуватись на визначенні базового. Тобто новий клас отримає все від суперкласу та доповнить це своїм. У похідного класу може бути один (одиничне спадкування) або більше (множинне спадкування) базових класів.

Поліморфізм також є однією з важливих концепцій ООП. Його застосування дозволяє значно полегшити розробку складних програм. Термін «поліморфізм» з грецької означає «наявність багатьох форм». З ним тісно пов’язані такі поняття, як абстрактні класи, віртуальні методи, перевантаження методів і властивостей.

Віртуальні методи – один із найважливіших прийомів реалізації поліморфізму. Вони дозволяють створювати загальний код, який може працювати як з об’єктами базового класу, так і з об’єктами будь-якого його класу-нащадка. При цьому базовий клас визначає спосіб роботи з об’єктами і будь-які його нащадки можуть представляти конкретну реалізацію цього способу. Клас, який містить хоча б один метод, що не має реалізації у цьому ж класі (абстрактний метод), називається абстрактним. Об’єкт такого класу створювати не можна. Нащадки абстрактного класу повинні обов’язково представити реалізацію для всіх його абстрактних методів, інакше вони будуть абстрактними. Можливість присвоювати різну функціональність одному методу називається перевантаженням методу.

Зм.

Арк.

№ документу

Підпис

Дата

Арк.

21

ОМФК.0121КР.10003ПЗ

**Висновок**

Зм.

Арк.

№ документу

Підпис

Дата

Арк.

22

ОМФК.0121КР.10003ПЗ

У ході виконання курсової роботи мовою C# у середовищі Visual Studio було створено додаток «Каса автостанції». Це було продемонстровано на прикладі можливостей об’єктно-орієнтованого програмування та інструментального середовища програмування.

Результати:

1) Створена програма дає можливість користувачу ознайомитись з інформацією про доступні на даний момент білети (маршрут, їхня ціна тощо). Також користувач може отримати довідкову інформацію.

2) В програмі були реалізовані основні елементи класів за допомогою принципів ООП.

**Список використаної літератури**

Зм.

Арк.

№ документу

Підпис

Дата

Арк.

23

ОМФК.0121КР.10003ПЗ

1. Джеффри Рихтер. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C#. 4-е изд. — СПб.: Питер, 2018. — 896 с.: ил. — (Серия «Мастер-класс»).

2. Скит, Джон. С# для профессионалов: тонкости программирования (3-е издание). : Пер. с англ. — М. : ООО “И.Д. Вильямс”, 2019. – 608 с. : ил. — Парал. тит. англ.

3. Кристиан Нейгел, Билл Ивьен, Джей Глинн, Карли Уотсон, Морган Скиннер - С# 2008 и платформа .Net 3.5 для профессионалов.

4. https://uk.wikipedia.org/wiki/Об%27єктно-орієнтоване\_програмування

5. https://uk.wikipedia.org/wiki/Успадкування\_(програмування)

6. http://www.rsdn.ru/article/dotnet/datagridview20.xml

7. http://nullpro.info/category/c/samouchitel-c/ – Самоучитель по C#, 2013.

Додаток

Зм.

Арк.

№ документу

Підпис

Дата

Арк.

24

ОМФК.0121КР.10003ПЗ

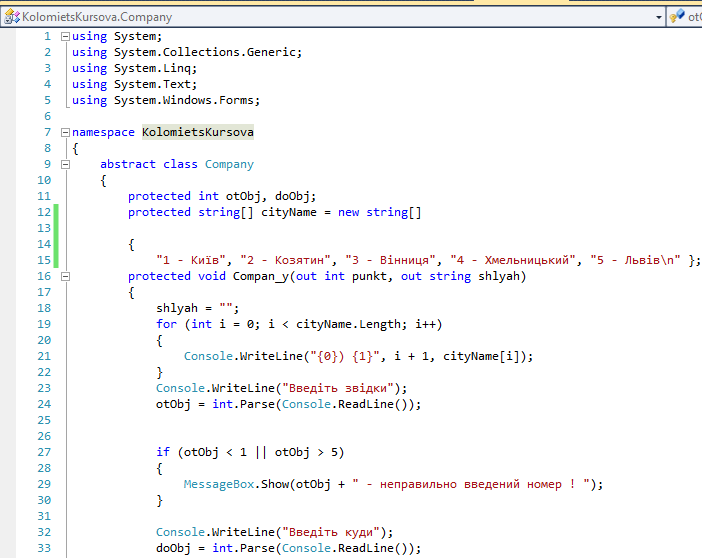
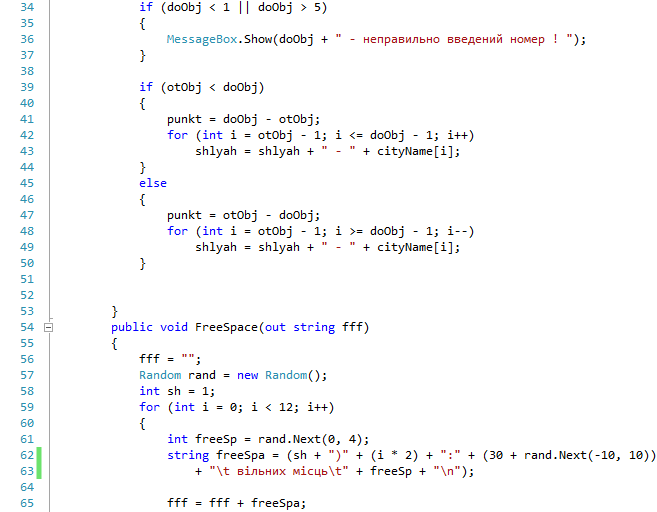
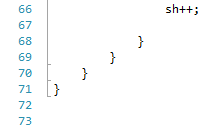


Рисунок 16. Код базового класу company з описом напряму і вартості руху (1/2).





Зм.

Арк.

№ документу

Підпис

Дата

Арк.

25

ОМФК.0121КР.10003ПЗ

Рисунок 17. Код базового класу company з описом напряму і вартості руху (2/2).

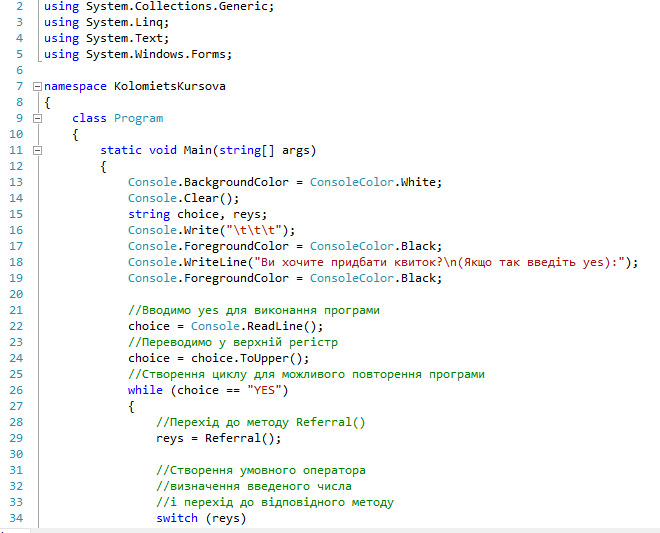


Рисунок 18. Код похідного класу MainMenu, що дає користувачу доступ до інших класів програми за допомогою об’єктів та оператора switch (1/3).



Зм.

Арк.

№ документу

Підпис

Дата

Арк.

26

ОМФК.0121КР.10003ПЗ

Рисунок 19. Код похідного класу MainMenu, що дає користувачу доступ до інших класів програми за допомогою об’єктів та оператора switch (2/3).

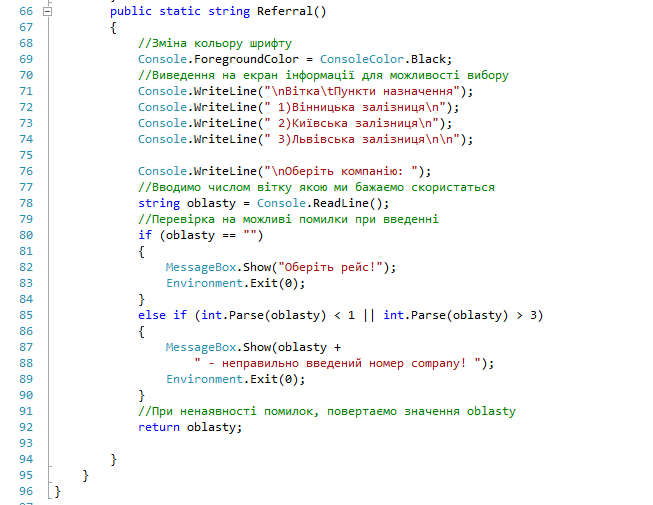
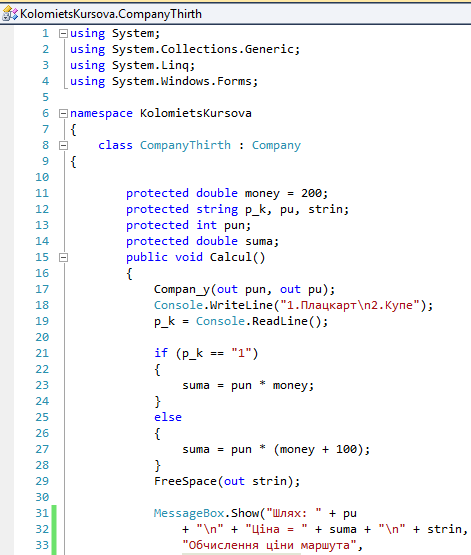


Рисунок 20. Код похідного класу MainMenu, що дає користувачу доступ до інших класів програми за допомогою об’єктів та оператора switch (3/3).



Зм.

Арк.

№ документу

Підпис

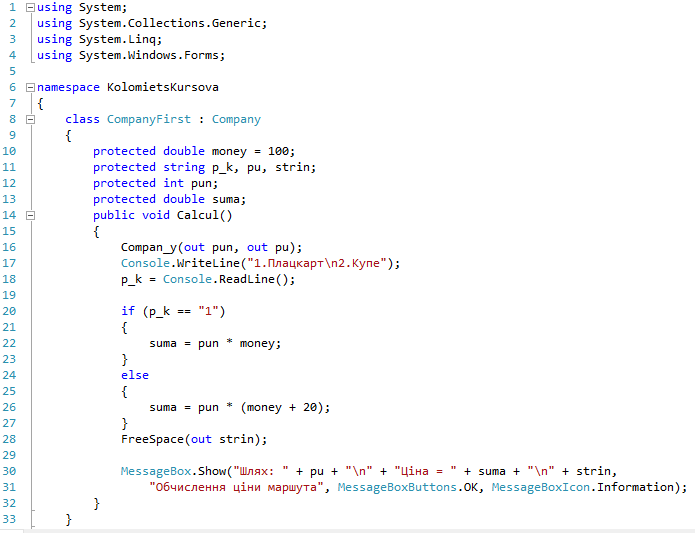
Дата

Арк.

27

ОМФК.0121КР.10003ПЗ

Рисунок 21. Код похідного класу Registration, що надає інформацію про маршрут.



Зм.

Арк.

№ документу

Підпис

Дата

Арк.

28

ОМФК.0121КР.10003ПЗ

Рисунок 22. Код класу СompanyFirst, який відповідає за виведення ціни.

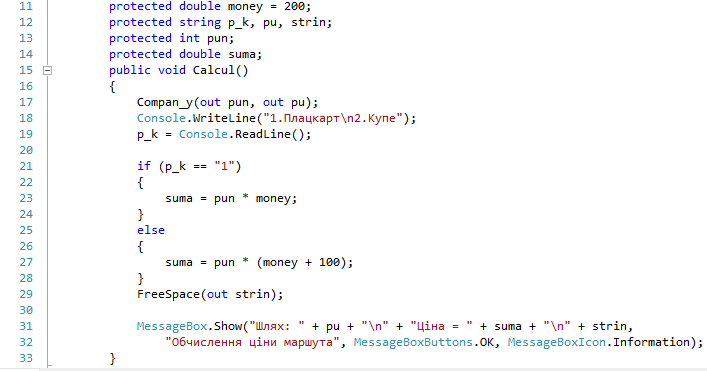
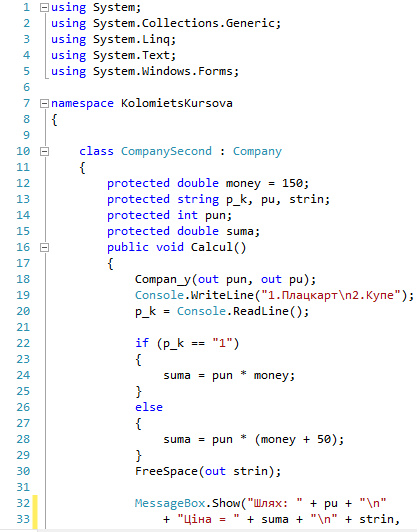


Рисунок 23. Код класу Company, що показує маршрут.