Практична робота №5
Фундаментальні принципи ООП. Абстрагування даних та параметричний поліморфізм
Система оцінювання

№	Тема	К-ть балів
1.	Завдання 1	1
2.	Завдання 2	1
3.	Завдання 3	1
4.	Завдання 4	1
5.	Завдання 5	1
	Всього за практичну роботу	**5**
6.	ІНДЗ-1	1
7.	ІНДЗ-2	1
8.	ІНДЗ-3	1
	Всього	**8**

Завдання на практичне заняття

1. ***(Абстрактні класи)*** Розробіть абстрактний клас Figure, який передбачатиме наступні члени:
    - Приватне внутрішнє поле name (назва фігури);
    - Конструктор з одним параметром, який ініціалізує поле name переданим значенням;
    - Властивість Name для доступу до внутрішнього поля name;
    - Абстрактна властивість Area2, призначена для знаходження площі фігури;
    - Абстрактний метод Area, призначений для знаходження площі фігури;
    - Віртуальний метод Print, який відображає назву фігури.

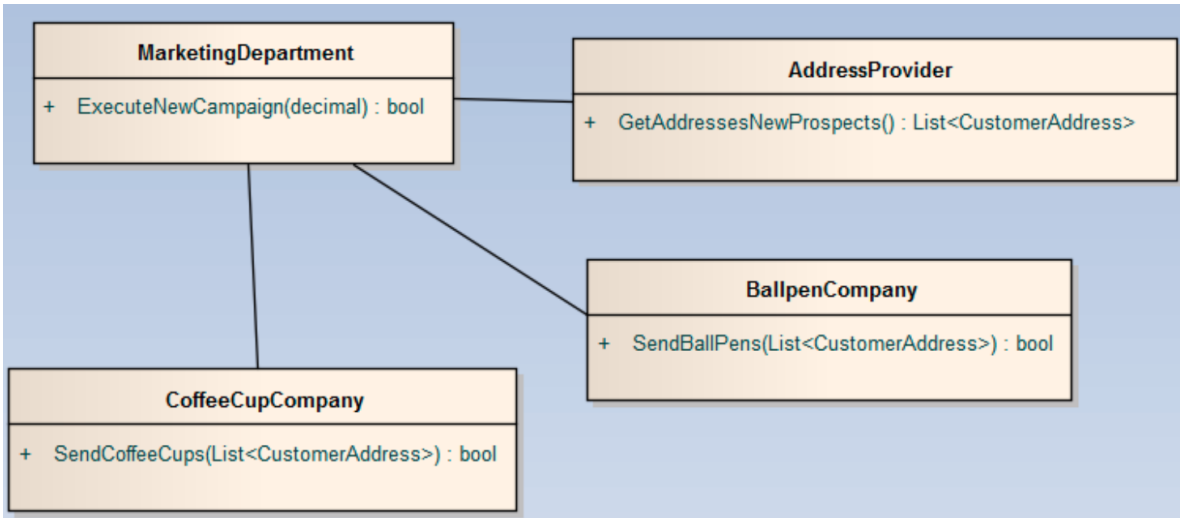
    Створіть клас Triangle, який успадковує (розширяє) можливості класу Figure. Даний клас передбачає наступні члени:
    - Приховані внутрішні поля a, b, c (сторони трикутника);
    - Конструктор з 4ма параметрами;
    - Методи-аксесори SetABC(), GetABC() для доступу до полів класу. Кожний метод отримує 3 параметри – довжини сторін трикутника;
    - Властивість Area2 визначає площу трикутника за трьома його сторонами;
    - Метод Area, який повертає площу трикутника за трьома його сторонами;
    - Віртуальний метод Print для відображення значень внутрішніх полів класу. Метод звертається до аналогічного методу з базового класу.

    Доповніть додаток класом TriangleColor, який успадковує можливості класу Triangle. Цей клас має наступні члени:
    - Приховане внутрішнє поле color;

- Конструктор з 5ма параметрами, який викликатиме конструктор базового класу;
- Властивість Color для доступу до внутрішнього поля color;
- Властивість Area2, яка викликає властивість базового класу з тією ж назвою, щоб обчислити площу трикутника;
- Метод Area, який повертає площу трикутника за його сторонами;
- Віртуальний метод Print для відображення значень внутрішніх полів класу. Метод звертається до аналогічного методу з базового класу.

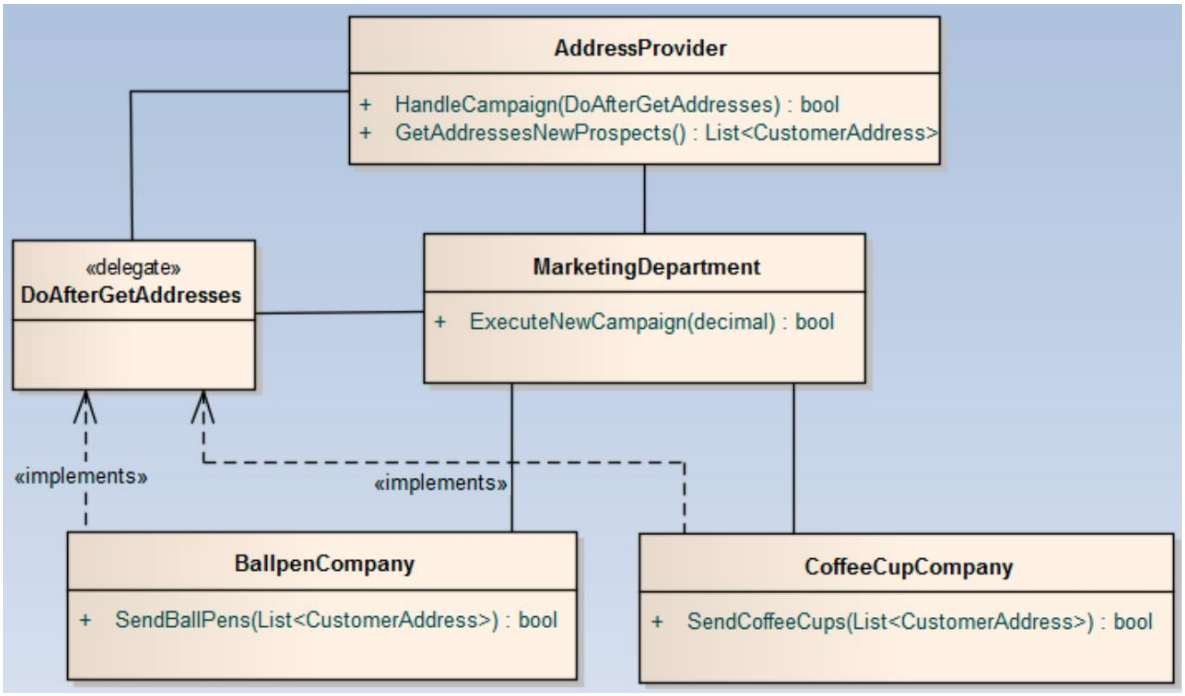
2. *Детально опишіть* у звіті кроки, представлені в статті, після чого продемонструйте роботу представленого додатку.
3. ***(Делегати)*** Нехай потрібно змоделювати роботу маркетингового відділу компанії. Його робота – проводити рекламні кампанії з адресами потенційних клієнтів. Якщо бюджет відділу не перевищує 10000, тоді в розсилку на адреси потенційних клієнтів потрапить тільки компанія BallpenCompany (виробник кулькових ручок), інакше – і компанія CoffeeCupCompany (виробник чашок). Основна ідея: отримуємо адреси від деякого провайдера адрес та відправляємо їх у BallpenCompany або CoffeeCupCompany залежно від бюджету.

Сценарій 1



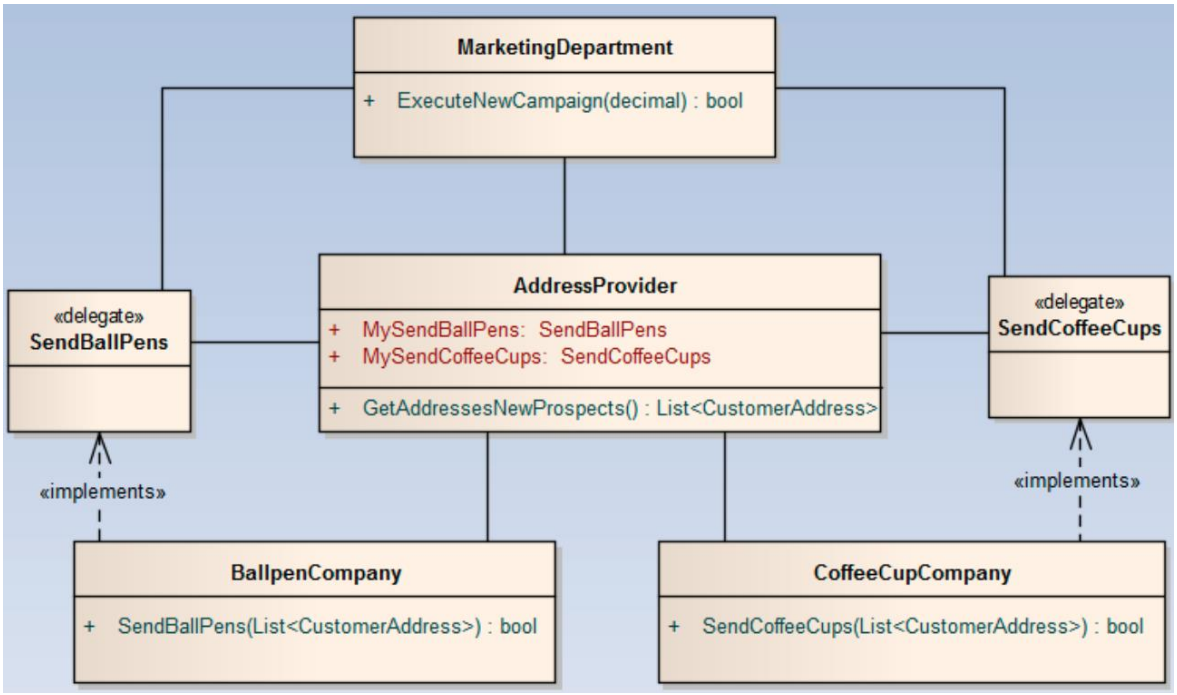
Сценарій 2

Працівники відділу подумали, що було б краще, якби провайдер адрес сам міг вирішувати, що зі своїми адресами робити. Вирішено створити делегат, який слугуватиме обгорткою для метода, що буде викликатись. Клас Marketing Department буде заповнювати обгортку методом за вибором (SendBallpens чи SendCoffeeCups), а потім передаватиме обгортку в AddressProvider. Останній прийматиме обгортку і викликатиме її, навіть не знаючи, який конкретно метод буде виконано:



Сценарій 3

AddressProvider має зв'зяок з BallpenCompany та CoffeeCupCompany. Замість розкриття цих компаній для MarketingDepartment, AdressProvider робить доступними 2 делегати: один для надсилання кулькових ручок, інший – для надсилання чашок для кави.



Реалізуйте код для всіх трьох ситуацій та сформуйте уявлення про роль делегатів.

4. ***(Події)*** Напишіть програму, яка моделює відвідування онлайн-заходу. Вимоги наступні:
    - Користувачі надають на вхід свій логін, після чого додаток відображає повідомлення «Вітаємо, <логін>».
    - Існує перелік забанених організаторами логінів користувачів. Як тільки хтось із цього переліку вводить свій логін, додаток викликає подію: видає звукове сповіщення (нескінченний періодичний сигнал біпера) та надсилає e-mail адміністрації (виведе про це повідомлення). Додаток у цей час переходить у зациклений режим, єдиний спосіб виходу – завершення роботи додатку (натиснення Ctrl + C).

    Продемонструйте введення двох звичайних користувачів та одного забаненого.

5. (Скоро буде…)

ІНДЗ

1. (Знаходження класів, які реалізують інтерфейс) Розгляньте статтю та виведіть перелік класів (а також загальну їх кількість), які реалізують інтерфейс з переліку відповідно до Вашого номеру в списку підгрупи:

№	Назва інтерфейсу
1	IComparable
2	IDisposable
3	IFormattable
4	IEnumerable
5	ICollection
6	IEnumerator
7	IList
8	IDictionary
9	IAsyncResult
10	IConvertible
11	IEquatable
12	IObservable
13	IProgress
14	IServiceProvider
15	IAsyncDisposable

2. *Детально опишіть* рекомендовані практики для реалізації інтерфейсу IEnumerable та продемонструйте роботу запропонованих у статті алгоритмів.

3. Створіть власний узагальнений інтерфейс ICounter<T> – лічильник, який матиме два члени:
   - Властивість Count (без сеттера), яка має тип int;
   - Метод Get(), який за переданим індексом повертатиме елемент послідовності.

   На базі даного інтерфейсу опишіть інтерфейс IPersons<T>, який на додачу передбачатиме метод для додавання елементу в послідовність (метод нічого не повертатиме).
   Передбачте узагальнений клас People<T>, який реалізуватиме інтерфейс IPersons<T>. Даний клас міститиме два поля: розмір послідовності і послідовність (масив узагальненого типу). Також додайте конструктор, який за умовчанням буде ініціалізувати розмір нулем, послідовність – на 10 елементів. Реалізуйте методи, передбачені в інтерфейсі IPersons<T> відповідно до логіки роботи масиву.
   Доповніть додаток класом Employee, який матиме наступні члени:
   - Властивість для присвоєння номеру робітникові;
   - Властивості для імені та прізвища робітника;
   - Властивість для встановленої погодинної оплати;
   - Конструктор для попередніх 4х атрибутів класу із значеннями за умовчанням «0, Іван, Іванов, 50»;
   - Заміщений метод ToString(), який організує форматований вивід інформації по робітнику.

   В управляючому класі створіть 3 робітника та IPersons-масив з Employee-об’єктів. Додайте робітників у масив, а потім виведіть збережену інформацію щодо них.