

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
"КІЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО"  
НАВЧАЛЬНО-НАУКОВИХ ІНСТИТУТ АТОМНОЇ ТА ТЕПЛОВОЇ ЕНЕРГЕТИКИ  
КАФЕДРА ЦИФРОВИХ ТЕХНОЛОГІЙ В ЕНЕРГЕТИЦІ

ЛАБОРАТОРНА РОБОТА №1  
з дисципліни "Постреляційні бази даних"  
Тема: «Реалізація основних принципів ООП»

Виконав: студент групи ТР-52мп  
Томін Владислав Вікторович

Перевірив: Волков Олександр  
Володимирович

В ході виконання лабораторної роботи було використано мову програмування C#

## 1. Спадкування від абстрактного класу

```
5 // 1) Абстрактний клас + публічні/приватні властивості та методи
6 abstract class Person
7 {
8     public Guid Id { get; } = Guid.NewGuid();
9     public string Name { get; protected set; }
10    private DateTime _createdAt = DateTime.UtcNow; // приватне поле
11
12    public DateTime CreatedAtUtc => _createdAt;
13
14    // абстрактний метод (поліморфізм)
15    public abstract string Describe();
16
17    // перевантажені методи (overLoad)
18    public void Rename(string newName) => Name = newName;
19    public void Rename(string prefix, string newName) => Name = $"{prefix}{newName}";
20 }
21
22 // 2) Наслідування від абстрактного класу
23 class Teacher : Person
24 {
25     public string Subject { get; private set; }
26
27     public Teacher(string name, string subject)
28     {
29         Name = name; Subject = subject;
30     }
31
32     public override string Describe() => $"Teacher {Name} – {Subject}";
33 }
34
35 class Student : Person
36 {
37     // посилання на інший об'єкт (викладач як наставник)
38     public Teacher Advisor { get; private set; }
39
40     public List<int> Grades { get; } = new();
41
42     public Student(string name) { Name = name; }
43
44     public void SetAdvisor(Teacher t) => Advisor = t;
45
46     // перевантаження: додавання однієї оцінки або кількох
47     public void AddGrade(int g) => Grades.Add(g);
48     public void AddGrade(params int[] many) { Grades.AddRange(many); }
49
50     // приватний хелпер
51     private double Avg() => Grades.Count == 0 ? double.NaN : Grades.Average();
52
53     public override string Describe()
54     => $"Student {Name} – advisor: {Advisor?.Name ?? "(none)"}, grades: {Grades.Count}, avg: {Avg():F1}";
55 }
56 }
```

## 2. Публічні та приватні властивості

```
public Guid Id { get; }           // публічне поле
private DateTime _createdAt;      // приватне поле
private double Avg() => ...;      // приватний метод
public DateTime CreatedAtUtc => _createdAt; // публічне для читання
```

## 3. Перевантаження методів

```

// у Person
public void Rename(string newName) => Name = newName;
public void Rename(string prefix, string newName) => Name = $"{prefix}{newName}";

// у Student
public void AddGrade(int g) => Grades.Add(g);
public void AddGrade(params int[] many) => Grades.AddRange(many);

// у Classroom
public void Add(Person p) => _people.Add(p);
public void Add(params Person[] many) => _people.AddRange(many);

```

#### 4. Властивості простих типів

```

public Guid Id { get; }           // простий тип Guid
public string Name { get; protected set; } // string
private DateTime _createdAt;      // DateTime
public string Subject { get; private set; } // string
public List<int> Grades { get; } = new(); // int

```

#### 5. Посилання на інші об'єкти

```

public Guid Id { get; }           // простий тип Guid
public string Name { get; protected set; } // string
private DateTime _createdAt;      // DateTime
public string Subject { get; private set; } // string
public List<int> Grades { get; } = new(); // int

```

#### 6. Масиви/списки простих типів/об'єктів

```

public List<int> Grades { get; } = new(); // список простих типів (int)
private readonly List<Person> _people = new(); // список об'єктів

```

#### 7. Код створення об'єктів та заповнення властивостей

```

var t = new Teacher("Dr. Ada", "Algorithms");
var s = new Student("Vlad");
var room = new Classroom("CS-101");

// Заповнення властивостей
s.SetAdvisor(t);           // посилення на інший об'єкт
s.AddGrade(90);            // заповнює список оцінок
s.AddGrade(88, 95, 100);   // перевантаження
t.Rename("Prof. ", "Ada"); // змінює ім'я
s.Rename("Vladyslav");    // інше перевантаження
room.Add(t, s);           // додає об'єкти у список

// виклик усіх методів
Console.WriteLine(room);    // викликає ToString()
foreach (var p in room.People)
    Console.WriteLine(p.Describe()); // поліморфний виклик абстрактного Describe()

```