

## Лабораторна робота №6

**Тема:** «Основи об'єктно-орієнтованого програмування. Модулі та пакети».

**Мета:** вивчити способи створення та підключення модулів та пакетів. Основи ООП. Методи і атрибути класів та робота з ними. Побудова програми у стилі ООП.

### Завдання:

1. Вивчити матеріал лекцій 18, 19, 20, 21, 22 та 23.
2. Виконати індивідуальне завдання лабораторної роботи, вибране відповідно до варіанту.

### Теоретичні основи:

#### **Модулі й пакети**

**Модулем у мові Python називають** будь-який файл з програмним кодом.

Кожний модуль може імпортувати інший модуль, одержуючи, таким чином, доступ до атрибутів (змінних, функцій і класів), оголошених усередині імпортованого модуля.

**Слід відмітити**, що імпортований модуль може містити програму не тільки мовою Python – так, можна імпортувати скомпільований модуль, написаний мовою C.

Усі програми, які ми запускали раніше, були розташовані в модулі з назвою `"__main__"`. Одержати ім'я модуля дозволяє визначений атрибут `__name__`.

#### **Інструкція `import`**

Імпортувати модуль дозволяє інструкція `import`.

Інструкція `import` має наступний формат:

```
import    <Назва модуля1> [ <as Псевдонім1> ] [ , ... ,  
                <Назва модуляN> [ <as ПсевдонімN> ] ]
```

**Після імпортування** модуля його назва стає ідентифікатором, через який можна одержати доступ до атрибутів, визначених усередині модуля.

**Доступ до атрибутів** модуля здійснюється за допомогою точкової нотації. Наприклад, звернутися до константи `pi`, розташованої усередині модуля `math`, можна так: `math.pi`

Функція `getattr()` дозволяє одержати значення атрибута модуля по його назві, заданій у вигляді рядка. За допомогою цієї функції можна сформулювати назву атрибута динамічно під час виконання програми.

Формат функції:

```
getattr( <Об'єкт модуля> , <Атрибут> [ , <Значення за  
замовчуванням> ] )
```

Якщо зазначений атрибут не знайдений, виконується виключення `AttributeError`. Щоб уникнути виводу повідомлення про помилку, можна в третьому параметрі вказати значення, яке буде повертатися, якщо атрибут не існує.

Перевірити існування атрибута дозволяє функція

`hasattr` (`<Об'єкт>`, `<Назва атрибута>`).

Якщо атрибут існує, функція повертає значення `True`.

Якщо назва модуля занадто довга, і її незручно вказувати щоразу для доступу до атрибутів модуля, то можна створити псевдонім. Псевдонім задають після ключового слова **as**. Доступ до атрибутів модуля може здійснюватися за допомогою ідентифікатора-псевдоніма.

Кожний модуль Python **може бути скомпільований** і перетворений в особливе внутрішнє представлення (байт-код), – для прискорення виконання коду.

Файли з відкомпільованим кодом зберігаються в папці `__pycache__`, що автоматично створюється в папці, де перебуває сам файл з початковим, невідкомпільованим кодом модуля, і мають імена виду:

`<ім'я файлу з початковим кодом>.cpython-<перші дві цифри номера версії Python>.pyc`.

Для одержання відкомпільованого файлу необхідно виконати код, показаний у прикладі:

**Приклад.**

```
>>> import py_compile
>>> py_compile.compile('tests.py')
'__pycache__\tests.cpython-35.pyc'
```

Таким чином, після запуску цього коду відкомпільований `tests.py` буде збережений у файлі

`tests.cpython-35.pyc`.

Для імпортування модуля достатньо мати тільки файл з відкомпільованим кодом, файл із початковим кодом у цьому випадку не потрібний.

**Імпортування модуля виконується тільки при першому виклику** інструкції `import` (або `from`). При кожному виклику інструкції `import` перевіряється наявність об'єкта модуля в словнику `modules` з модуля `sys`. Якщо посилання на модуль перебуває в цьому словнику, то модуль повторно імпортуватися не буде.

Для імпортування тільки деяких визначених ідентифікаторів з модуля можна скористатися інструкцією `from`. Формат інструкції:

```
from <Назв. модуля> import <Ідентифікатор1> [as <Псевдонім1>]
[, ..., <ІдентифікаторN> [as <Псевдонім N>]]
```

```
from <Назв. модуля> import (<Ідентифікатор1> [as
<Псевдонім1>], ..., <ІдентифікаторN> [as <Псевдонім N>]])
```

```
from <Назва модуля> import *
```

### Шляхи пошуку модулів

Якщо модулі розміщені в одній папці з файлом основної програми, немає необхідності набудовувати шляхи пошуку модулів, оскільки папка з файлом, що виконується, автоматично додається в початок списку шляхів.

Одержати повний список шляхів пошуку дозволяє наступний код:

Список `sys.path` містить шляхи пошуку, одержувані з наступних джерел:

- **шлях до поточного каталогу** з кодом основної програми;
- **значення змінної оточення PYTHONPATH.**
- **шляхи пошуку стандартних модулів;**
- **вміст файлів з розширенням `pth`,**

**Кожний шлях** (абсолютний або відносний) повинен бути розташований на окремому рядку.

### Повторне завантаження модулів

Як ви вже знаєте, модуль завантажується тільки один раз при першій операції імпорту. Усі наступні операції імпортування цього модуля будуть повертати вже завантажений об'єкт модуля, навіть якщо сам модуль був змінений. Щоб повторно завантажити модуль, слід скористатися функцією `reload()` з модуля `imp`.

Формат функції:

```
from imp import reload
reload(<Об'єкт модуля>)
```

### Пакети

**Пакетом** називають каталог з модулями, у якому розташований файл ініціалізації `__init__.py`.

Файл ініціалізації може бути порожнім або містити код, який буде виконаний при першому доступі до пакета.

У будь-якому разі він обов'язково повинен бути присутнім всередині каталогу з модулями.

Для встановлення пакетів на Python використовуємо систему управління пакетами **pip**.

**pip** – використовується для установки і управління програмними пакетами, написаними на Python.

Для версії Python 3.5 **pip** постачається разом з інтерпретатором Python.

Якщо з якоїсь причини **pip** відсутній, то він може бути завантажений з <https://bootstrap.pypa.io/>

Назва файлу: [get-pip.py](#)

Інструкція по **pip** розміщена:

<https://pip.pypa.io/en/latest/installing/>

### **Приклад. Як завантажити пакет PyGame під Windows**

1. Вибираємо «Пуск-Выполнить»
2. В рядку «Открыть» набираємо «cmd». ОК.
3. У вікні «Администратор» набираємо: `pip install PyGame`

При цьому має бути підключення до мережі Internet

## **Об'єктно-орієнтоване програмування**

Об'єктно-орієнтоване програмування (ООП) – це спосіб організації програми, що дозволяє використовувати той самий код багаторазово. На відміну від функцій і модулів, ООП дозволяє не тільки розділити програму на фрагменти, але й описати предмети реального світу у вигляді зручних сутностей – об'єктів, а також організувати зв'язки між цими об'єктами.

Основною «цеглинкою» ООП є клас. *Клас* – це складний тип даних, що включає набір змінних і функцій для керування значеннями, що зберігаються в цих змінних.

Змінні називають *атрибутами*, а функції – *методами*. Клас є фабрикою об'єктів, тобто дозволяє створити необмежену кількість екземплярів, заснованих на цьому класі.

ООП ґрунтується на трьох основних концепціях розробки: **інкапсуляція**, **спадкування** й **поліморфізм**.

### **Визначення класу й створення екземпляра класу**

Клас описують за допомогою ключового слова `class` за наступною схемою:

```
class <Назва класу>[(<Клас1>[, ... , <Класn>])]:  
    [""" Рядок документування """]  
    <Опис атрибутів і методів>
```

Інструкція створює новий об'єкт і присвоює посилання на нього ідентифікатору, зазначеному після ключового слова `class`.

Це означає, що назва класу повинна повністю відповідати правилам іменування змінних. Після назви класу в круглих дужках можна вказати один або кілька базових класів через кому. Якщо ж клас не успадковує базові класи, то круглі дужки можна не вказувати.

Всі вирази всередині інструкції `class` виконуються при створенні класу, а не його екземпляра.

Створення атрибута класу аналогічно створенню звичайної змінної.

Метод всередині класу створюється так само, як і звичайна функція, за допомогою інструкції `def`.

Методам класу в першому параметрі, який обов'язково слід вказати явно, автоматично передають посилання на екземпляр класу. **Загальноприйнято** цей параметр називати ім'ям `self` (не обов'язково).

Доступ до атрибутів і методів класу всередині обумовленого методу проводиться через змінну `self` за допомогою точкової нотації – до атрибута `x` з методу класу можна звернутися так: `self.x`.

Щоб використовувати атрибути й методи класу, необхідно створити екземпляр класу згідно з наступним синтаксисом:

```
<Екземпляр класу> = <Назва класу>((<Параметри>))
```

При доступі до методів класу використовують такий формат:

```
<Екземпляр класу>.<Ім'я методу>((<Параметри>))
```

**При виклику методу не потрібно передавати посилання на екземпляр класу як параметр**, як у визначенні методу всередині класу. Посилання на екземпляр класу інтерпретатор передає автоматично.

Доступ до атрибутів класу здійснюється аналогічно:

```
<Екземпляр класу>.<Ім'я атрибута>
```

### Методи `__init__()` і `__del__()`

При створенні екземпляра класу інтерпретатор автоматично викликає метод ініціалізації `__init__()`. В інших мовах програмування такий метод прийнято називати конструктором класу.

Формат методу:

```
def __init__(self[, <Значення1>[, ... , <Значенняn>]]):  
<Інструкції>
```

За допомогою методу `__init__()` можна присвоїти початкові значення атрибутам класу. При створенні екземпляра класу параметри цього методу вказують після імені класу в круглих дужках:

```
<Екземпляр класу> = <Ім'я класу>([<Значення1>[, ... ,  
<Значенняn>]])
```

Якщо **конструктор** викликають при створенні об'єкта, то перед знищенням об'єкта автоматично викликається метод, називаний **деструктором**. У мові Python деструктор реалізується у вигляді визначеного методу `__del__()`. Метод не буде викликаний, якщо на екземпляр класу існує хоча б одне посилання.

## Інкапсуляція

Розглянемо концепцію розробки, що одержала назву «**інкапсуляція**». Ідея інкапсуляції полягає в тому, щоб сховати логіку функціонування від зовнішнього доступу, а користувачеві даного фрагмента коду надати тільки інтерфейс для його використання.

1. У цьому випадку ви можете звертатися до цього фрагмента з різних областей програми, що заощаджує розмір коду.

2. Якщо необхідно модифікувати інкапсульований фрагмент, то це не вплине на працездатність всієї програми.

В Python інкапсуляція виконується за допомогою методів класу.

## Перезавантаження операторів

В Python є можливість перехоплювати за допомогою спеціальних вбудованих методів виконання стандартних операцій і застосовувати нові варіанти цих операцій до екземпляра класу.

## Спадкування

Спадкування є найголовнішим поняттям ООП.

Припустимо, у нас є клас (наприклад, `Class1`). За допомогою спадкування ми можемо створити новий клас (наприклад, `Class2`), у якому буде реалізований доступ до всіх атрибутів і методів класу `Class1`.

**УВАГА!** Конструктор базового класу автоматично не викликається, якщо він перевизначений у підкласі.

Щоб викликати однойменний метод з базового класу, можна також скористатися функцією `super()`.

Формат функції:

```
super([<Клас>, <Показчик self>])
```

За допомогою функції `super()` інструкцію

```
Class1.__init__(self) #Викликаємо конструктор базового класу
```

можна записати так:

```
super().__init__() #Викликаємо конструктор базового класу  
або так:
```

```
super(Class2, self).__init__() # Викликаємо конструктор базового класу
```

При використанні функції `super()` **не потрібно явно** передавати вказівник `self` у викликуваний метод. Крім того, у першому параметрі функції `super()` вказують похідний клас (підклас), а не базовий. Пошук ідентифікатора буде проводитися у всіх базових класах. Результатом пошуку стане перший знайдений ідентифікатор у ланцюжку спадкування.

### Множинне спадкування

У визначенні класу в круглих дужках можна вказати **кілька** базових класів через кому.

### Поліморфізм

Парадигма об'єктно-орієнтованого програмування крім спадкування включає ще одну важливу особливість — поліморфізм. Слово «поліморфізм» можна перевести як «багато форм».

В ОО (об'єктно-орієнтованому) програмуванні цим терміном позначають можливість використання того самого імені операції або методу до об'єктів різних класів, при цьому дії, виконувані з об'єктами, можуть суттєво різнитися. Тому можна сказати, що в одного слова багато форм.

### Перевизначення методів

Використання поліморфізму при спадкуванні класів дозволяє перевизначати методи суперкласів їх підкласами. Наприклад, може виникнути ситуація, коли всі підкласи реалізують визначений метод із суперкласу, і лише один підклас повинен мати його іншу реалізацію. У такому випадку метод перевизначають в підкласі.

### Домішки та їх використання

Множинне спадкування, підтримуване Python, дозволяє реалізувати спосіб розширення функціональності класів за допомогою так званих *домішок*.

(mixins). Домішка – це клас, що включає атрибути й методи, які необхідно додати до інших класів. Оголошуються вони так само, як і звичайні класи.



## Для доступу до атрибутів і методів можна використовувати наступні функції:

Функція `getattr()` – повертає значення атрибута по його назві, заданій у вигляді рядка.

За допомогою цієї функції можна сформувати ім'я атрибута динамічно під час виконання програми. Формат функції:

```
getattr(<Об'єкт>, <Атрибут>[, <Значення за замовчуванням>])
```

Якщо зазначений атрибут не знайдений, виконується виключення `AttributeError`. Щоб уникнути виводу повідомлення про помилку, можна в третьому параметрі вказати значення, яке буде повертатися, якщо атрибут не існує.

Функція `setattr()` – задає значення атрибута. Назва атрибута вказується у вигляді рядка.

Формат функції:

```
setattr(<Об'єкт>, <Атрибут>, <Значення>)
```

Другим параметром методу `setattr()` можна передати ім'я неіснуючого атрибута – у цьому випадку атрибут із зазначеним іменем буде створений.

Функція `delattr (<Об'єкт>, <Атрибут>)` – видаляє зазначений атрибут. Назва атрибута вказується у вигляді рядка.

Функція `hasattr (<Об'єкт>, <Атрибут>)` – перевіряє наявність зазначеного атрибута. Якщо атрибут існує, функція повертає значення `True`.

## Спеціальні методи

Класи підтримують наступні спеціальні методи:

Метод `__call__()` – дозволяє обробити виклик екземпляра класу як виклик функції.

Формат методу:

```
__call__(self[, <Параметр1>[, ... , <Параметрn>]])
```

Метод `__getattr__(self, <Атрибут>)` – викликається при доступі до неіснуючого атрибута класу.



Метод `__getattr__(self, <Атрибут>)` – викликається при доступі до будь-якого атрибута класу. Необхідно враховувати, що використання точкової нотації (для доступу до атрибута класу) всередині цього методу призведе до зациклення. Щоб уникнути зациклення, слід викликати метод `__getattr__()` об'єкта `object`. Всередині методу потрібно повернути значення атрибута або виконати виключення `AttributeError`.

Метод `__setattr__(self, <Атрибут>, <Значення>)` – викликається при спробі присвоювання значення атрибуту екземпляра класу. Якщо всередині методу необхідно присвоїти значення атрибуту, то слід використовувати словник `__dict__`, інакше при точковій нотації метод `__setattr__()` буде викликаний повторно, що призведе до зациклення.

Метод `__delattr__(self, <Атрибут>)` – викликають при видаленні атрибута за допомогою інструкції `del <Екземпляр класу>.<Атрибут>;`

Метод `__len__(self)` – викликають при використанні функції `len()`, а також для перевірки об'єкта на логічне значення при відсутності методу `__bool__()`. Метод повинен повертати додатне ціле число.

Метод `__bool__(self)` – викликають при використанні функції `bool()`.

Метод `__int__(self)` – викликають при перетворенні об'єкта в ціле число за допомогою функції `int()`.

Метод `__float__(self)` – викликають при перетворенні об'єкта в дійсне число за допомогою функції `float()`.

Метод `__complex__(self)` – викликають при перетворенні об'єкта в комплексне число за допомогою функції `complex()`.

Метод `__round__(self, n)` – викликають при використанні функції `round()`.

Метод `__index__(self)` – викликають при використанні функцій `bin()`, `hex()` і `oct()`.

**Методи `__repr__(self)` і `__str__(self)`** – слугують для перетворення об'єкта в рядок.

Метод `__repr__()` викликають при виводі в інтерактивній оболонці, а також при використанні функції `repr()`.

Метод `__str__()` викликають при виводі за допомогою функції `print()`, а також при використанні функції `str()`. Якщо метод `__str__()` відсутній, то буде викликаний метод `__repr__()`.

Як значення методи `__repr__()` і `str()` повинні повертати рядок.

Метод `__hash__(self)` – цей метод слід перевизначити, якщо екземпляр класу заплановано використовувати як ключ словника або всередині множини.

## Підсумок по ООП для Python

Найважливіші положення концепції ООП (об'єктно-орієнтованого програмування) для Python:

1. **Створення екземплярів класів.** На цьому етапі важливим моментом є заповнення атрибутів екземпляра класу під час його створення.
2. **Створення методів,** які інкапсують логіку поведінки в методах класу.
3. **Перевантаження операторів.** Задає індивідуальну поведінку екземпляра класу при виконанні вбудованих операцій, наприклад, при виводі на друк.
4. **Спадкування.** Дозволяє використовувати методи й атрибути класів-предків (суперкласів), а також модифікувати їх.
5. **Модифікація конструкторів.** Дозволяє змінити або додати логіку при створенні суперкласу.

## Індивідуальні завдання

Відповідно до **номера у списку** вибрати індивідуальне завдання. Написати програму. Забезпечити ввід даних з клавіатури комп'ютера та друк результатів. При виводі використовувати форматування.

	Завдання
1	Створити клас, який описує роботу кафе і містить асортимент страв, їх базову ціну, знижки за днями тижня, знижки для постійних відвідувачів та метод обчислення вартості замовлення. Створити об'єкт даного класу, що дозволяє визначити вартість обіду для довільного клієнта з урахуванням знижок. Програма повинна виводити рахунок зі списком

	<b>Завдання</b>
	замовлених страв, їх ціною та знижкою з урахуванням особи клієнта та дня тижня.
2	<p>Створити клас, який описує проведення змагань і містить прізвища учасників, стать, кількість набраних балів, наявність допінгу, вагову категорію та містить метод визначення двох таблиць учасників (за статтю), сформованих у порядку спадання результатів. Окремо створити таблицю дискваліфікованих. Визначення результатів відбувається з урахуванням вагової категорії.</p> <p>Створити об'єкт даного класу, що дозволяє визначити розстановку учасників у таблиці змагань та таблиці дискваліфікованих за умови генерування кількості балів випадковим чином. Максимальна кількість балів – 100. Наявність допінгу у учасника також поставити випадковим чином. Програма повинна виводити таблицю всіх учасників, таблиці переможців з урахуванням статі та таблицю дискваліфікованих.</p>
3	<p>Створити клас, який описує місяць року і містить назву місяця, список днів даного місяця, довжина якого формується в залежності від номеру місяця.</p> <p>Кожен день може бути сонячним, хмарним або дощовим. Кожен день також характеризується середньодобовою температурою. Клас має містити методи визначення середньої кількості сонячних хмарних та дощових днів, методи визначення мінімальної, максимальної та середньої температури.</p> <p>Створити об'єкти даного класу, для кожного місяця року, задати кількості сонячних, хмарних, дощових днів та середньодобову температуру кожного дня випадковим чином з урахуванням пори року та номеру місяця. Програма повинна виводити за запитом середні кількості сонячних, хмарних, дощових днів та середню температуру для кожного місяця за запитом.</p>
4	<p>Створити клас, який описує книгу та містить назву книги, автора, кількість сторінок, кількість ілюстрацій та містить метод визначення ціни книги в залежності від ціни сторінки тексту, ціни ілюстрації та видавництва.</p> <p>Створити об'єкти даного класу для певного набору книг, виданих видавництвами України. Ціну сторінки та ціну ілюстрації задати випадковим чином.</p> <p>Програма повинна виводити ціну книги, за назвою. Вивести середню ціну книги для введеного автора та для окремого видавництва.</p>
5	<p>Створити клас, який описує місто та містить назву міста, кількість мешканців, кількість районів, кількість ВНЗ (вищих навчальних закладів), площу, географічне розташування (північ, південь, центр і т. ін.). Клас має також містити методи обчислення кількості мешканців на одиницю площі, середню площу району міста, кількість жителів на один ВНЗ.</p>

	<b>Завдання</b>
	<p>Створити об'єкти даного класу для деякої кількості міст України. Якщо необхідні параметри неможливо знайти в Інтернет, то задати їх випадковим чином.</p> <p>Програма повинна вивести на друк після вводу назви міста: кількість мешканців, площу, кількість районів, кількість мешканців на одиницю площі, середню площу району міста, кількість жителів на один ВНЗ та географічне положення міста.</p>
6	<p>Створити клас, який описує континент нашої планети та містить назву континенту, площу, географічне положення, кількість населення та кількість країн, що на ньому розміщені. Методи даного класу повинні визначати кількість населення на одиницю площі; приналежність континенту до північної, південної, східної, західної півкулі або кількох цих півкуль одночасно.</p> <p>Створити об'єкти даного класу для всіх континентів та ввести дані, користуючись даними з Вікіпедії.</p> <p>Програма повинна вивести на друк після вводу назви континенту: кількість мешканців, площу, кількість країн. Також програма повинна виводити список континентів, які розташовані у півкулі за запитом, а також таблицю континентів, відсортованих за густиною населення.</p>
7	<p>Створити клас, який описує країну. Він повинен містити назву країни і назву континенту, на якому вона розміщена, кількість населення, список міст-мільйонників, та кількість населення в кожному з таких міст. Методи класу обчислюють відсоток міського населення в країні та густину населення.</p> <p>Створити об'єкти даного класу для кількох країн та ввести дані, користуючись даними з Вікіпедії.</p> <p>Програма повинна обчислювати та виводити за запитом відсоток міського населення та густину населення в країні. Програма повинна також виводити за запитом найбільше та найменше місто серед тих, що були введені в програму, друкуючи його назву та чисельність населення.</p>
8	<p>Створити клас, який описує планету Сонячної системи. Цей клас повинен містити назву планети, її порядковий номер, рахуючи від Сонця, відстань від Сонця, об'єм та масу, орбітальну швидкість. Методи класу мають визначати густину планети, відношення маси Землі до маси даної планети, відношення орбітальної швидкості Землі до о.ш. планети.</p> <p>Створити об'єкти даного класу для планет Сонячної системи та ввести дані, користуючись даними з Вікіпедії.</p> <p>Програма повинна обчислювати та виводити за запитом густину планети, відношення маси та о.ш. Землі до маси та о.ш. планети, введеної за запитом. Програма повинна також виводити за запитом список планет у порядку зростання їх густини з зазначенням номеру планети та її орбітальної швидкості.</p>

	Завдання
9	<p>Створити клас, який описує факультет КПІ. Клас повинен містити назву, скорочену назву, прізвище декана та номер телефону деканату, список спеціальностей, список кафедр. Методи класу повинні визначати, чи належить кафедра до даного факультету, чи належить спеціальність до даного факультету. Створити об'єкти даного класу, користуючись інформацією сайту НТУУ «КПІ».</p> <p>При введенні скороченої назви факультету програма повинна виводити його повну назву, прізвище декана, номер телефону деканату.</p> <p>При вводі назви спеціальності програма повинна виводити список факультетів, які готують студентів за даною спеціальністю. При вводі назви кафедри програма має виводити відповідний факультет та номер телефону деканату.</p>
10	<p>Створити клас, який описує область України. Клас повинен містити назву області, її площу, кількість населення та список головних міст. Методи класу повинні визначати, чи належить місто до даної області та густину населення.</p> <p>Створіть об'єкти даного класу, які описують області України, та введіть потрібну інформацію, користуючись даними Вікіпедії.</p> <p>При введенні назви області програма повинна виводити площу, кількість населення, густину населення та список головних міст. При вводі назви міста програма повинна виводити назву області, у якій воно розміщено. За запитом програма повинна виводити список областей, відсортованих за густиною населення з зазначенням назви області та густини населення.</p>
11	<p>Створити клас, який описує магазин одягу на прикладі інтернет-магазину <a href="http://www.lila-shop.ua/">http://www.lila-shop.ua/</a>. Клас містить назву магазину, список товарів, список постійних клієнтів, накопичувальну скидку для постійних клієнтів, сезонну знижку та знижку, яка враховує час приходу в магазин. Клас містить метод обчислення накопичувальної знижки для конкретного постійного клієнта, метод обчислення сезонної знижки та метод обчислення в залежності від години відвідування магазину.</p> <p>Створити об'єкт даного класу, який описує роботу деякого магазину, ввівши асортимент товару, постійних покупців та відсотки для трьох видів знижок.</p> <p>Програма повинна виводити розрахунок вартості конкретного вибраного товару для покупця, враховуючи всі види знижок. Програма також повинна виводити список постійних покупців зі значеннями накопичувальної знижки та список товарів з їх цінами.</p>
12	<p>Створити клас, який описує інтернет-магазин на прикладі <a href="http://rozetka.com.ua">http://rozetka.com.ua</a>. Клас повинен містити багаторівневу структуру зберігання інформації про товари, яка включає каталог товарів, кожен пункт якого містить категорії. Кожна категорія містить словник товарів, ключом якого є код товару, а кортеж значень включає назву, ціну та інші дані. Клас повинен мати методи, які дають можливість сортування</p>

	Завдання
	<p>товарів за спаданням ціни та за зростанням ціни, пошуку товару за фрагментом назви, за діапазоном ціни.</p> <p>Створити об'єкт даного класу, використовуючи дані з магазину «rozetka».</p> <p>Програма повинна мати простий інтерфейс пошуку товару, який базується на використанні методів класу. Програма також має формувати корзину покупок з виводом списку вибраних товарів та загальної суми.</p>
13	<p>Створити клас, який описує англо-український словник. Клас повинен містити список можливих варіантів перекладу. Методи класу повинні забезпечити простий пошук англійських слів за введеним підрядком.</p> <p>Створити об'єкт даного класу, який дозволяє, користуючись методами та атрибутами класу, одержати український переклад.</p> <p>Програма повинна виводити український переклад введеного англійського слова з можливими варіантами. За приклад взяти словник <a href="http://e2u.org.ua/">http://e2u.org.ua/</a>.</p>
14	<p>Створити клас, який описує тлумачний словник української мови. Клас повинен містити словник з ключів-українських слів та значень у вигляді кортежу синонімів. Методи класу мають реалізувати пошук слова за підрядком.</p> <p>Створити об'єкт даного класу, який дозволяє, користуючись методами та атрибутами класу, одержати синоніми українського слова.</p> <p>Програма повинна виводити синоніми введеного українського слова з можливими варіантами. За приклад взяти словник <a href="http://sum.in.ua/">http://sum.in.ua/</a></p>
15	<p>Створити клас, який описує найбільші міста нашої планети. Клас повинен містити словник, в якості ключів якого є континенти, а в якості значень є словники, ключами яких є міста-мільйонники даного континенту, а значеннями – кортежі з площею, кількістю населення та географічними координатами. Методи класу повинні знаходити найближче розташоване місто до введеного, а також найближче місто за площею та найближче за густиною населення.</p> <p>Створити об'єкт даного класу, який дозволяє з використанням методів класу знаходити найближче розташоване міста та подібні за площею і густиною населення міста. Для вводу інформації про міста користуватися даними з Вікіпедії.</p> <p>Програма повинна виводити за запитом найближче розташоване місто, а при повторних запитах міста, які відсортовані за відстанню від даного міста. Також програма повинна виводити найближче за площею та густиною населення місто і всі міста для введеного континенту.</p>
16	<p>Введіть клас держав нашої планети. Клас повинен містити словник з ключами-державами та кортежами-значеннями, що включають назву столиці, її площу та кількість населення. Методи класу повинні визначати список держав, столиці яких містять кількість населення в заданому діапазоні. Визначати список держав, які мають столиці з</p>

	<b>Завдання</b>
	<p>заданим діапазоном площі, визначати відношення площі столиці до площі держави, а також визначати назву держави, до якої належить столиця.</p> <p>Створити об'єкт класу, який використовує всі методи класу, для введення інформації скористатися даними Вікіпедії.</p> <p>Програма повинна виводити список держав, які містять населення з введеного діапазону у порядку спадання. Також при виводі держав, що мають столиці в заданому діапазоні площ, використати сортування за спаданням. В обох згаданих випадках виводити також назви столиць, кількість їх населення та площі. При вводі назви столиці, програма повинна виводити державу, до якої вона належить.</p>
17	<p>Створити клас, який описує роботу салону з продажу автомобілів. Салон повинен містити словник автомобілів, структурований за виробниками та марками. Кожен автомобіль повинен характеризуватися ціною, роком випуску та пробігом. Методи класу повинні визначати наявність в салоні марок автомобілів заданого виробника, визначати автомобілі, які потрапляють в заданий діапазон цін, визначати автомобілі з пробігом, який знаходиться в заданому діапазоні.</p> <p>Створити об'єкт класу, який використовує всі методи класу, для введення інформації скористатися даними <a href="http://privat-auto.info/">http://privat-auto.info/</a>.</p> <p>Програма повинна виводити на друк список автомобілів та їх параметри за введеним виробником, виводити автомобілі, які знаходяться у заданому діапазоні цін або пробігу. Також програма повинна виводити на друк список автомобілів та їх параметри заданої марки.</p>
18	<p>Створити клас, який описує бібліотеку. Клас повинен містити назву бібліотеки, словник з ключами-жанрами книг (пригоди, фантастика, історичний роман і т.д.) Як значення задати кортеж, який містить прізвище автора, його ім'я, назву книги, кількість сторінок та видавництво. Методи класу повинні визначати список книг з введеного жанру, визначати жанри, у яких існують книги даного автора, визначати видавництво і автора при введенні назви книги, знаходити всі книги, у назвах яких зустрічається підрядок вводу.</p> <p>Створити об'єкт класу, який описує конкретну бібліотеку та реалізує задані методи класу. В якості джерела інформації для введення даних скористайтесь <a href="http://library.kpi.ua/">http://library.kpi.ua/</a></p> <p>Програма повинна за запитом виводити книги, що написав відповідний автор, книги за жанром, за діапазоном сторінок, за видавництвом. При написанні програми для пошуку даних застосувати пошук за підрядком.</p>
19	<p>Створити клас дисциплін, які ви вивчаєте. Клас має містити словник з ключами-назвами предметів. В якості значень задати кортежі, які</p>



	Завдання
	<p>містять як елементи прізвище викладача, кількість годин на тиждень та наявність екзамену, диференційного заліку або простого заліку.</p> <p>Методи класу мають визначати прізвище викладача за назвою предмету та назву предмету за прізвищем викладача, якщо викладач читає кілька предметів, то вони повинні бути перераховані при виводі.</p> <p>Створити об'єкт даного класу, користуючись своїм розкладом.</p> <p>Програма повинна виводити після введення предмету прізвище викладача і навпаки. При виводі повідомляти про наявність екзамену або заліку. Також за запитом необхідно виводити список предметів, які мають однакову кількість годин на тиждень.</p>
20	<p>Створіть клас, який описує квартиру. Він повинен містити номер квартири, кількість квадратних метрів, кількість спожитої за місяць холодної води, кількість спожитої гарячої води, величину спожитої електроенергії, суму, яка сплачена за місяць за холодну воду, сплачену суму за споживання води, суму за електроенергію. Методи класу повинні обраховувати величину переоплати чи заборгованості по кожному платежу з урахуванням тарифів, які прийняті в Україні. Ці тарифи можна дізнатися: <a href="http://kyivenergo.ua/">http://kyivenergo.ua/</a> та <a href="https://www.giok.kiev.ua/law/tariff/">https://www.giok.kiev.ua/law/tariff/</a></p> <p>Створити об'єкти даного класу для квартир багатоквартирного будинку. Задати випадковим чином суми оплати та кількості спожитих послуг.</p> <p>Програма повинна виводити кількості спожитих послуг, суми необхідної оплати за них, реальну оплату за кожну послугу та величину переоплати або заборгованості. При введенні назви послуги та визначенні значення параметра «переплата» або «заборгованість» програма повинна виводити номери квартир та відповідно їх переоплати або заборгованості. Програма також повинна виводити загальну переоплату або заборгованість по будинку.</p>
21	<p>Створіть клас, який описує Вікіпедію. Клас повинен містити словник з ключами-розділами Вікіпедії (Україна, Культура і мистецтво, Технології та інженерія, Прикладні та природничі науки і т. ін.). Відповідними значеннями є словники з ключами, що відображають підкатегорії, та значеннями у вигляді кортежів. При створенні структури даних користуватися інформацією за адресою <a href="https://uk.wikipedia.org/">https://uk.wikipedia.org/</a>. Методи класу повинні визначати статті, які містять у назві підрядок з запиту, визначати всі підкатегорії введеної категорії.</p> <p>Створити об'єкт даного класу, який наслідує методи класу. Для заповнення даними скористатися інформацією з Вікіпедії.</p> <p>Програма повинна виводити список статей за введеним підрядком, виводити всі підкатегорії за введеною категорією.</p>
22	<p>Створіть клас, який описує структуру кабінету міністрів. Клас повинен містити повні та скорочені назви міністерств, словник, ключами якого є скорочені назви міністерств. В якості значення використайте кортеж,</p>

	Завдання
	<p>який складається з контактного телефону, електронної адреси міністерства та прізвища міністра. Методи класу повинні визначати повну назву міністерства за його скороченою назвою, визначати контактний телефон та ін. за скороченою назвою.</p> <p>Створити об'єкт класу, який описує Кабінет Міністрів України. Для заповнення даними скористатися інформацією <a href="http://www.kmu.gov.ua/">http://www.kmu.gov.ua/</a> Програма повинна виводити назву міністерства після введення прізвища міністра, телефон і email після введення назви міністерства, виводити повну назву після введення скороченої назви, виводити всі міністерства, які у повній назві мають підрядок запиту.</p>
23	<p>Створіть клас, який описує українську мову. Клас повинен містити багаторівневий словник з ключами – назвами частин мови. Кожна ключ-частина мови повинен мати значення у вигляді словника другого рівня і т. д. відповідно до класифікації, яка подана у Вікіпедії: <a href="https://uk.wikipedia.org/wiki/Частини_мови">https://uk.wikipedia.org/wiki/Частини_мови</a>. На найнижчому рівні словника повинні знаходитися кортежі зі словами. Методи класу повинні визначати всі характеристики слова як частини мови при його введенні, визначати приклади слів, які відповідають введеній характеристиці частини мови, методи вводу та модифікації слів.</p> <p>Створити об'єкт класу та ввести дані, використовуючи інформацію з Вікіпедії.</p> <p>Програма повинна виводити всі характеристики слова як частини мови при його введенні, виводити приклади слів, які відповідають введеній характеристиці частини мови, вводити та редагувати слова.</p>
24	<p>Створіть клас, який описує структуру міської державної адміністрації. Клас повинен містити назви Управлінь міської державної адміністрації та їх скорочені назви. Створіть словник, у якому як ключ використовується скорочена назва Управління. Як значення задати прізвище керівника та контактний телефон.</p> <p>Методи класу повинні визначати повну назву управління після введення скороченої назви, визначати управління при пошуку за підрядком, визначати керівника та контактні дані за запитом, додавати нові та редагувати введені управління</p> <p>Створити об'єкт класу, який описує Київську міську державну адміністрацію (КМДА) та ввести дані, використовуючи інформацію з <a href="https://kievcity.gov.ua/">https://kievcity.gov.ua/</a>.</p> <p>Програма повинна виводити на друк повну назву управління після введення скороченої назви, виводити управління при пошуку за підрядком, виводити керівника та контактні дані за запитом, додавати нові та редагувати введені управління</p>
25	<p>Створити клас, який описує структуру торгової мережі супермаркетів. Клас повинен містити назву мережі, інформацію про акції та товари, що підпадають під акційні знижки, перелік товарів з поділом на тематичні</p>

	Завдання
	<p>групи та ціною і кодами. Методи класу повинні визначати знижку відповідно до акції або акцій, в яких бере участь той чи інший товар. Створити об'єкт даного класу, який описує мережу сільпо. Для вводу інформації скористатися <a href="http://silpo.ua/ua/">http://silpo.ua/ua/</a>.</p> <p>При вводі товару програма повинна виводити на друк список товарів, які містять підрядок назви товару з їх кодами. При вводі коду товару програма повинна вивести ціну товару, розрахунок поточної знижки на товар та вартість товару з урахуванням знижок.</p>
26	<p>Створити клас, який описує структуру торгової мережі, яка має філіали, що розташовані в містах України. Клас повинен містити назву мережі, словник, ключами якого є міста України, у яких існують філіали мережі. Як значення запишіть кортеж, який містить місцеву адресу, телефон та іншу інформацію за бажанням.</p> <p>Методи класу мають визначати всі міста з філіалами при введенні підрядка назви міста, визначати відомі дані про філіал при вводі міста, вводити інформацію про нові філіали та редагувати введені філіали. Створити об'єкт класу, який описує мережу магазинів «WINTIME» та ввести дані, використовуючи інформацію з <a href="http://www.winetime.ua/store/">http://www.winetime.ua/store/</a>.</p> <p>Програма повинна виводити на друк всі міста з філіалами магазину, які починаються з введеної букви; виводити адресу, телефон та іншу доступну інформацію філіалу магазину, який розміщений у місті, що вводиться у запиті, вводити нові філіали та редагувати параметри введених філіалів.</p>
27	<p>Створити клас, який описує континент. Клас повинен містити назву континенту, словник з ключами – назвами країн, які розміщені на даному континенті, і значеннями-кортежами. Елементами таких кортежів повинні бути назви столиць відповідних країн, їх площа та кількість населення. Методи класу повинні визначати приналежність країни до континенту, визначати параметри країни при вводі її назви. Створити об'єкти даного класу для кількох континентів, користуючись даними Вікіпедії.</p> <p>Програма повинна виводити список країн при введенні назви континенту і виводити континент, кількість населення і площу при введенні назви країни.</p>
28	<p>Створити клас, який описує район міста. Клас повинен містити назву району, інформацію про район: площа, населення, місцезнаходження та історичні місцевості, які він включає. Методи класу мають визначати інформацію про район за його назвою, визначати, на якому березі район знаходиться, визначати перелік історичних назв місцевостей. Створити об'єкти даного класу, які описують райони міста Києва. Для вводу інформації скористатися <a href="https://uk.wikipedia.org/wiki/Адміністративно-територіальний_устрій_Києва">https://uk.wikipedia.org/wiki/Адміністративно-територіальний_устрій_Києва</a>.</p>

	Завдання
	Програма повинна виводити на друк інформацію про район при вводиті його назви. Також необхідно за запитом виводити райони, які знаходяться на лівому березі, або на правому березі. При вводиті історичної місцевості вивести назву району, який її включає.
29	<p>Створити клас, який описує країну-члена Євросоюзу. Клас має містити назву країни, інформацію про площу країни, кількість населення, рік вступу та місцезнаходження. Методи класу повинні визначати зону місцезнаходження країни, визначати інформацію про країну за запитом.</p> <p>Створити об'єкти даного класу, які описують країни Євросоюзу.</p> <p>Для вводу інформації скористатись даними з Вікіпедії:  <a href="https://uk.wikipedia.org/wiki/Європейський_Союз">https://uk.wikipedia.org/wiki/Європейський Союз</a>.</p> <p>Програма повинна виводити всі країни, що знаходяться в одній зоні місцезнаходження, наприклад: Східна Європа, Північна Європа і т.ін.</p> <p>Також програма повинна виводити дані про країну при введенні її назви, виводити сусідні країни при введенні назви країни.</p>

### Зміст звіту:

#### 1. Титульний лист повинен мати такий вигляд:

Національний технічний університет України  
«Київський політехнічний інститут ім. І. Сікорського»  
Факультет інформатики та обчислювальної техніки  
Кафедра обчислювальної техніки

Програмування  
Лабораторна робота №6

«Основи об'єктно-орієнтованого програмування. Модулі та пакети»

Виконав:  
студент групи ІО-\_\_

\_\_\_\_\_  
Номер у списку групи:  
Перевірів Новотарський М.А.

Київ 2016р.

2. Мета лабораторної роботи та загальне завдання
3. Короткі теоретичні відомості, які відображають інформацію про модулі, пакети та елементи ООП, **що були використані при написанні лабораторної роботи.**
4. Завдання та алгоритм його розв'язування (у довільній формі).
5. Роздруківка того фрагменту тексту програми, який написаний індивідуально.
6. Роздруківка результатів виконання програми з контрольним прикладом
7. Аналіз результатів та висновки.