**Національний технічний університет України**

**«Київський політехнічний інститут імені Ігоря Сікорського»**

**Факультет інформатики та обчислювальної техніки**

**Кафедра обчислювальної техніки**

**Програмування**

**Лабораторна робота №5**

**«Словники та функції користувача»**

Виконав:

студент групи ІО-63

Братун А.Ю.

Залікова книжка № 6305

Перевірив Новотарський М. А.

Київ 2016 р.

**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.

**Загальне завдання**:

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

**Теоретичні відомості**

**Модулі й пакети**

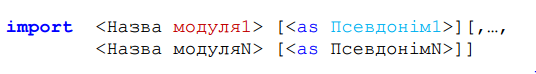
Модулем у мові Python називають будь-який файл з програмним кодом.

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 С. Усі програми, які ми запускали раніше, були розташовані в модулі з назвою "\_\_main\_\_".

Одержати ім'я модуля дозволяє визначений атрибут \_\_name\_\_.

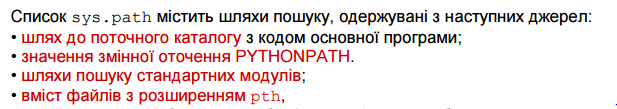
Імпортувати модуль дозволяє інструкція **import**.



Доступ до атрибутів модуля здійснюється за допомогою точкової нотації Наприклад, звернутися до

константи pi, розташованої усередині модуля math, можна так:





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.

pip – використовується для установки і управління програмними пакетами, написаними на Python. Для версії Python 3.5, pip постачається разом з інтерпретатором Python.

**Об'єктно-орієнтоване програмування**

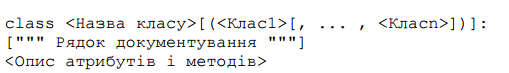
Об'єктно-орієнтоване програмування (ООП) – це спосіб організації програми, що дозволяє використовувати той самий код багаторазово. На відміну від функцій і модулів, ООП дозволяє не тільки розділити програму на фрагменти, але й описати предмети реального світу у вигляді зручних сутностей – об'єктів, а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

Клас описують за допомогою ключового слова class за наступною схемою:



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



При доступі до методів класу використовують такий формат: 

**Інкапсуляція** - це механізм, який об'єднує дані і методи, що маніпулюють цими даними, і захищає і те і інше від зовнішнього втручання або неправильного використання. Коли методи і дані об'єднуються у такий спосіб, створюється об'єкт.

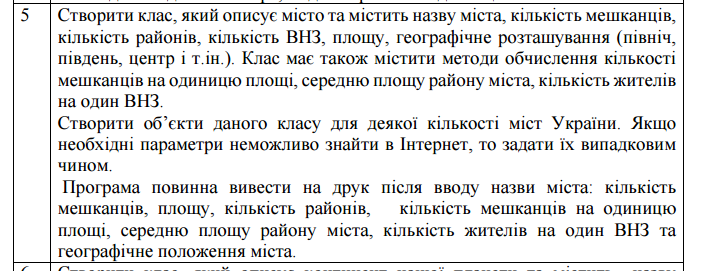
***Наслідування***– процес, завдяки якому один об’єкт може придбати властивості іншого, тобто наслідувати властивість іншого обєкту і додавати риси характерні тільки для нього самого.

***Поліморфізм***– властивість, яка дозволяє одне і те саме ім’я використовувати для вирішення декількох технічно різних задач, тобто основною метою поліморфізму є використання одного імені для задання загальних класу дій. На практиці це значить спроможність об’єктів вибирати внутрішній метод або процедуру, виходячи із типу даних, прийнятих в повідомленні.

**Алгоритм розв`язування завдання**

* Для того щоб виконати завдання потрібно спочатку створити клас
* В середині класу створити методи та забезпечити вивід даних
* В основній програмі створити екземпляри
* ПІсля вводу міста, перевірити чи в програмі є інформація про це місто
* Якщо інформація присутня – вивести дані
* Якщо інформація відсутня – сповістити про це користувача

**Завдання**



**фрагменту тексту програми, який написаний індивідуально.**

**Program.py**

**from** Town **import** Town  
  
kharkiv = Town("Харків", 1499700, 9, 38, 350, [46, 32, 17, 43])  
odessa = Town("Одеса", 1008489, 5, 20, 236, [23, 66, 12, 40])  
lviv = Town("Львів", 723605, 6, 26, 192, [32, 163, 71, 13])  
kiev = Town("Київ", 2908088, 10, 107, 847, [52, 33, 31, 29])  
lughansk = Town("Луганськ", 422373, 4, 5, 257, [65, 23, 67, 14])  
dnipro = Town("Дніпро", 18439, 6, 145, 379, [61,82,16,13])  
cherkassy = Town("Черкаси", 283130, 7, 187, 78, [33,47,54,39])  
zvenigorodka = Town("Звенигородка", 18035, 4, 1, 21, [23,43,44,39])  
semenivka = Town("Семенівка", 10248, 3, 1, 20, [11,25,17,44])  
nejin = Town("Ніжин", 93423, 3, 7, 43, [13,32,76,26])  
krasnopillya = Town("Краснопілля", 8259, 3, 2, 11, [26,35,55,74])  
priluki = Town("Прилуки", 35435, 7, 24, 40, [16,33,49,67])  
  
  
list\_cities = [lughansk, kiev, lviv, kharkiv, odessa,dnipro,cherkassy,zvenigorodka,semenivka,nejin,krasnopillya,priluki]  
list\_names = ["Київ","Луганськ" ,"Харків","Львів", "Оеса","Дніпро", "Черкаси", "Звенигородка",'Семенівка','Ніжин','Краснопілля','Прилуки']  
  
print("Вітаю в програмі опису міста!")  
  
**while True**:  
 print("Список доступних міст:"),  
 **for** i **in** list\_names:print(i, end=',')  
 print('\n')  
  
 value = str(input("Введіть назву міста: ")).lower()  
 value = value[0].upper() + value[1:]  
 **if** value **in** list\_names:  
 **for** i **in** list\_cities:  
 **if** i.name == value:  
 obj = i  
 **break  
 else**:  
 print('Інформація про це місто відсутня')  
 **continue** obj.print\_me()  
 print("Якщо бажаєте продовжити, нажміть щось, якщо хочете закінчить введіть q")  
 a = input()  
 **if** a == "стоп":  
 print("З вами було приємно мати справу")  
 **break**

**Town.py**

**class Town**:  
  
 name = ""  
  
 **def** \_\_init\_\_(self, name, people, regions, universities, areas, location):  
 self.name = name  
 self.people = people  
 self.regions = regions  
 self.universities = universities  
 self.areas = areas  
 self.location = location  
  
 **def thickness**(self):  
 result = round((self.people / self.areas), 2)  
 **return** result  
  
 **def Square\_area**(self):  
 result = round((self.areas / self.regions), 2)  
 **return** result  
  
 **def population\_for\_universities**(self):  
 result = round((self.people / self.universities), 2)  
 **return** result  
  
 **def print\_location**(self):  
 location\_dirs = ['Північ', 'Південь', 'Схід', 'Захід']  
 print("Географічне розташування".center(32))  
 **for** i **in** range(len(location\_dirs)):  
 print(location\_dirs[i], ' - ', self.location[i], "градусів")  
  
 **def print\_me**(self):  
 print("\*"\*35)  
 print("Інформація про місто".center(25, ' '))  
 print(self.name.center(25, ' '))  
 print("Кількість мешканців:", self.people, "осіб")  
 print("Площа:", self.areas, 'км^2')  
 print("Кількість районів:", self.regions)  
 print("Густота населення:", self.thickness(), "осіб/км^2")  
 print("Середня площа району міста:", self.Square\_area(), 'км^2')  
 print("Кількість жителів на один ВНЗ:", self.population\_for\_universities())  
 self.print\_location()  
 print("\*"\*35)

**Контрольный приклад**

**Вітаю в програмі опису міста!**

**Список доступних міст:**

**Київ,Луганськ,Харків,Львів,Оеса,Дніпро,Черкаси,Звенигородка,Семенівка,Ніжин,Краснопілля,Прилуки,**

**Введіть назву міста: київ**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Інформація про місто**

**Київ**

**Кількість мешканців: 2908088 осіб**

**Площа: 847 км^2**

**Кількість районів: 10**

**Густота населення: 3433.4 осіб/км^2**

**Середня площа району міста: 84.7 км^2**

**Кількість жителів на один ВНЗ: 27178.39**

**Географічне розташування**

**Північ - 52 градусів**

**Південь - 33 градусів**

**Схід - 31 градусів**

**Захід - 29 градусів**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Якщо бажаєте продовжити, нажміть щось, якщо хочете закінчить введіть 'стоп'**

**Список доступних міст:**

**Київ,Луганськ,Харків,Львів,Оеса,Дніпро,Черкаси,Звенигородка,Семенівка,Ніжин,Краснопілля,Прилуки,**

**Введіть назву міста: звенигородка**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Інформація про місто**

**Звенигородка**

**Кількість мешканців: 18035 осіб**

**Площа: 21 км^2**

**Кількість районів: 4**

**Густота населення: 858.81 осіб/км^2**

**Середня площа району міста: 5.25 км^2**

**Кількість жителів на один ВНЗ: 18035.0**

**Географічне розташування**

**Північ - 23 градусів**

**Південь - 43 градусів**

**Схід - 44 градусів**

**Захід - 39 градусів**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Якщо бажаєте продовжити, нажміть щось, якщо хочете закінчить введіть 'стоп'**

**стоп**

**З вами було приємно мати справу**

**Аналіз результатів та висновки.**

Під час виконання роботи проблем не виникло