

# **ОСНОВИ СИСТЕМ ШТУЧНОГО ІНТЕЛЕКТУ, НЕЙРОННИХ МЕРЕЖ ТА ГЛИБОКОГО НАВЧАННЯ**

## **Модуль 6. ВИСОКОРІВНЕВА МОВА ПРОГРАМУВАННЯ PYTHON**

### **Лекція 6.3. Файлові об'єкти. Визначення загальної структури програми**

# ФАЙЛ

**Файл** – іменована область постійної пам'яті в комп'ютері, якою управляє операційна система.

**Файл** – вбудований тип об'єкту, що, забезпечує можливість доступу до цих областей пам'яті.

**Об'єкт типу «файл»** - має тільки методи та атрибути !

# РОБОТА З ФАЙЛАМИ В JAVA

InputStream	OutputStream	Reader	Writer
FileInputStream	FileOutputStream	FileReader	FileWriter
BufferedInputStream	BufferedOutputStream	BufferedReader	BufferedWriter
ByteArrayInputStream	ByteArrayOutputStream	CharArrayReader	CharArrayWriter
FilterInputStream	FilterOutputStream	FilterReader	FilterWriter
DataInputStream	DataOutputStream		
ObjectInputStream	ObjectOutputStream		

# Файлові методи (операції)

Функція . Метод	Опис
<b>Open ()</b>	Створює об'єкт типу файл – «відкриває» файл
<b>f.close()</b>	Закриття відкритого файлу
<b>f.Readable()</b> <b>f.Read(), f.readline(), f.readlines()</b>	Перевірка можливості читання Читання
<b>f.Seekable(), f.Seek()</b>	Перевірка можливості зміни позиції
<b>f.Tell()</b>	Вертає поточну позицію
<b>f.Writable(), f.write(), f.writelines()</b>	Перевірка можливості запису Запис до файлу
<b>f.next()</b>	Вертає наступний рядок

[https://www.tutorialspoint.com/python/file\\_methods.htm](https://www.tutorialspoint.com/python/file_methods.htm)

[https://www.w3schools.com/python/python\\_ref\\_file.asp](https://www.w3schools.com/python/python_ref_file.asp)

# Файлові атрибути

Атрибут	Опис
<b>f.closed</b>	Індикатор поточного статусу файлу
<b>f.encoding</b>	Спосіб кодування/декодування файлу
<b>f.mode</b>	Режим відкриття файлу
<b>f.name</b>	Ім'я відкритого файлу
<b>f.newlines</b>	Універсальний режим завершення рядку

[https://www.tutorialspoint.com/python/file\\_methods.htm](https://www.tutorialspoint.com/python/file_methods.htm)

[https://www.w3schools.com/python/python\\_ref\\_file.asp](https://www.w3schools.com/python/python_ref_file.asp)

# Файлові методи (операції)

Вбудована функція	Опис
Open ()	Створює об'єкт типу файл – «відкриває» файл

*Open(file, mode, buffering, encoding, errors, newline, closed, opener)*

- *file* – path to file.
- *mode* – режим ('r' - читання, 'w' - запис, 'a' - додавання в кінець, 'b' – байтовий режим, 't' - текст, '+' – (r + w)). *За замовчуванням 'rt' – читання тексту.*
- *buffering* – управління буфером;
- *encoding* – режим кодування/декодування;
- *errors* – управління реакцією на помилки;
- *newline* – управління режимом «новий рядок»;
- *closed* – управління дескриптором файлу;
- *opener* – опис обробника помилки.

# Файлові методи (операції)

Вбудована функція	Опис
Open ()	Створює об'єкт типу файл – «відкриває» файл

Важливо: 'r', ... текст – рядки типу **str** – виконується автоматичне кодування / декодування.

'b', – послідовність байтів – деяких змін.

*my\_file = open ('my\_text\_file.txt', 'w') – відкриває файл для запису (тексту!)*

*my\_file = open ('my\_text\_file.txt') – відкриває файл читання (тексту!)*

*my\_file = open ('my\_text\_file.bin', 'wb') – відкриває файл для запису (послідовності байт)*

*my\_file = open ('my\_text\_file.bin', 'rb') – відкриває файл читання (послідовності байт)*

# Файлові методи (операції)

Метод	Опис
<b>f.readable ()</b> <b>f.read ()</b>	Зчитування файлу

**f.readable()** – повертає **True**, коли файл відкрито для читання.

**f.read(*size*)** - зчитує деяку кількість даних і повертає їх у вигляді рядка (у текстовому режимі) або об'єкта байтів (у двійковому режимі); *size* – максимальна кількість даних, що зчитуються (необов'язковий числовий аргумент).

**f.readline()** - зчитує з файлу один рядок; символ нового рядка (**\n**) залишається в кінці рядка і опускається лише в останньому рядку файлу.

**f.readlines()** - зчитує всі рядки до списку.



# Файлові методи (операції)

Метод	Опис
<b>f.writable ()</b> <b>f.write ()</b>	Зчитування файлу

**f.writable()** – вертає **True**, коли файл відкрито для запису.

**f.write(string)** – записує вміст рядка до файлу і повертає кількість записаних символів.

**f.writelines(string)** – запис наступного елементу списку (рядок) до файлу (файл відкрито з *'wa'*)

!!! Інші типи об'єктів потрібно перетворити - або в рядок (в текстовому режимі), або в байт-об'єкт (у двійковому режимі) - перед їх записуванням.

# Файлові методи (операції)

Метод	Опис
<b>f.tell ()</b> <b>f.seek ()</b>	Вертає поточну позицію в файлі Змінює поточну позицію в файлі

**f.tell ()** – вертає ціле число, що дає поточне положення об'єкта файлу у файлі, представлене як кількість байтів від початку файлу (у двійковому режимі) і номер поточного символу в текстовому режимі.

**f.seek (*offset*, *whence*)** – нова позиція обчислюється від додавання *offset* до *whence* (опорної точки).

*whence* →

= 0 вимірює від початку файлу (за замовчуванням),

= 1 поточне положення файлу,

= 2 кінець файлу.

# СТРУКТУРА PYTHON ПРОГРАМИ

*Ієрархія програми →*

Програма складається з **пакетів та (або) модулів** .

Пакет – Логічно завершена сукупність **модулів** .

Модуль – функціонально завершений фрагмент програми (оформлений як єдиний файл) - складається з **інструкцій** .

Інструкції складаються з **виразів** .

Вирази створюють та обробляють **об'єкти**

# СТРУКТУРА PYTHON ПРОГРАМИ



# ІНСТРУКЦІЇ PYTHON

Інструкція	Дія	Приклад
<b>import</b>	Доступ до модуля	Import math
<b>from</b>	Доступ до атрибутів модуля	From sys import stdin
<b>class</b>	Створення об'єкту	Class sub(super): def ...
<b>try/except/finally</b>	Обробка виключень	Try: action () except : print()
<b>raise</b>	Створення виключення	
<b>assert</b>	Перевірки для налаштування	
<b>with/as</b>	Менеджер контексту	With open('file') as myfile: proc(myfile)
<b>del</b>	Видалення посилань	

# ІНСТРУКЦІЇ PYTHON

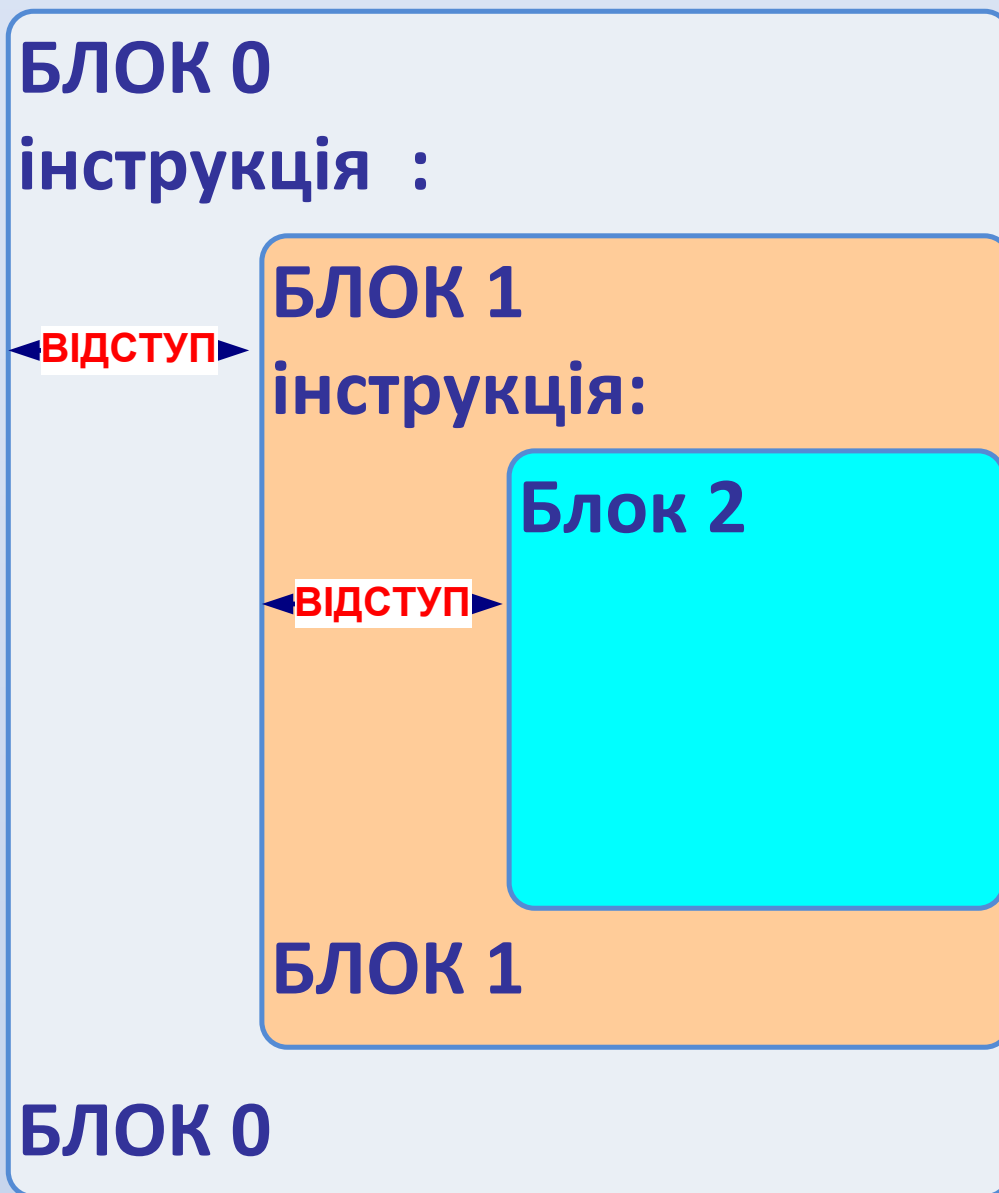
Інструкція	Дія	Приклад
<b>Присвоювання</b>	Створення посилань	A = "gglskfsj"
<b>Виклики</b>	Виклик функції	F = open()
<b>if / elif / else</b>	Вибір	If 'z' in text: print(text)
<b>For / else</b>	Перебір послідовності	for z in mlist: print(z)
<b>while / else</b>	Цикл загального призначення	while x>y: print('Qu')
<b>pass</b>	Пуста інструкція	while True: pass

# ІНСТРУКЦІЇ PYTHON

Інструкція	Дія	Приклад
<b>break</b>	Вихід з циклу	While True: if <i>cond</i> : break
<b>continue</b>	Перехід на початок циклу	While True: if <i>cond</i> : continue
<b>def</b>	Створення функцій та методів	Def foo (a): print (a)
<b>return</b>	Повернення результату	Def foo (a): return (a)
<b>yield</b>	Функції-генератори	Def gen (n): for i in n: yield (i)
<b>global</b>	Простір імен	
<b>nonlocal</b>	Простір імен (3.0)	

# СТРУКТУРА PYTHON ПРОГРАМИ

**Вкладеність блоків  
регулюється  
відступами**





## Рекомендована ЛІТЕРАТУРА

- **Програмування числових методів мовою Python:** підруч. / А. В. Анісімов, А. Ю. Дорошенко, С. Д. Погорілий, Я. Ю. Дорогий ; за ред. А. В. Анісімова. – К. : Видавничо-поліграфічний центр "Київський університет", 2014. – 640 с.
- **Програмування числових методів мовою Python:** навч. посіб. / А. Ю. Дорошенко, С. Д. Погорілий, Я. Ю. Дорогий, Є. В. Глушко ; за ред. А. В. Анісімова. – К. : Видавничо-поліграфічний центр "Київський університет", 2013. – 463 с.
- **Основи програмування Python:** Підручник для студ. спеціальності 122 «Компютерні науки» / А.В.Яковенко; КПІ.- Київ: КПІ, 2018 . – 195 с.
- **Лутц М.** Изучаем Python, 4-е издание. - СПб.: Символ-Плюс. 2011.- 1280 с.: ил.

# Контрольні запитання

- Наведіть визначення файлу в мові Python. Надайте перелік основних методів об'єкту файл.
- Надайте визначення функції *open()*, вкажіть призначення її параметрів та наведіть приклади використання.
- Надайте перелік основних атрибутів об'єкту файл та їх призначення.
- Надайте перелік методів **запису** до файлу, вкажіть їх призначення та наведіть приклади використання.
- Надайте перелік методів **читання** з файлу, вкажіть їх призначення та наведіть приклади використання.
- Надайте перелік методів **позиціювання** в файлі, вкажіть їх призначення та наведіть приклади використання.
- Наведіть перелік компонентів з яких складається програма на мові Python , вкажіть їх визначення.

**The END**

**Модуль 6. Лекція 6.3.**