**ЛАБОРАТОРНА РОБОТА № 6**

**РОБОТА З ФАЙЛАМИ**

***Мета***: ознайомитися з засобами роботи з файлами в мові Python, діями над ними.

**Хід роботи:**

**Завдання 1:** Створіть новий файл numbers.txt у текстовому редакторі і запишіть у нього 10 чисел, кожне з нового рядка. Напишіть програму, яка зчитує ці числа з файла і обчислює їх суму, виводить цю суму на екран і, водночас, записує цю суму у інший файл з назвою sum\_numbers.txt.

Лістинг програми

num = 0  
my\_file = open('numbers.txt', 'r')  
rez\_file = open("sum\_numbers.txt", "w")  
for line in my\_file:  
 num += int(line)  
rez\_file.write(str(num))  
rez\_file.close()  
my\_file.close()  
print(num)

Результат виконання програми:



Рис. 1. Результат виконання програми

Результат виконання програми:

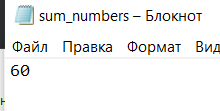


Рис. 2. Результат виконання програми

**Завдання 2:** Реалізуйте програму, яка зчитує довільну кількість цілих чисел, що вводяться з командного рядка, і записує у текстовий файл інформацію, щодо парності або непарності чисел.

Лістинг програми

tmp = ''  
rez\_file = open("rez\_task2.txt", "w")  
num = 1  
while num != 0:  
 num = int(input("Введіть число: "))  
 if num == 0:  
 print("", file=rez\_file, sep='', end='\n')  
 elif num % 2 == 0:  
 print("Парне число {0}".format(num), file=rez\_file, sep='', end='\n')  
 else:  
 print("Не парне число {0}".format(num), file=rez\_file, sep='', end='\n')  
rez\_file.close()

Результат виконання програми:

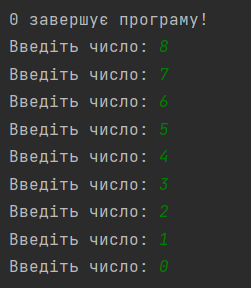


Рис. 3. Результат виконання програми

Результат виконання програми:

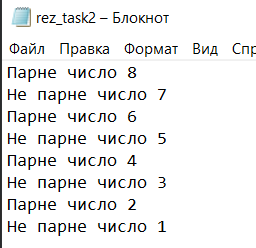


Рис. 4. Результат виконання програми

**Завдання 3:** Створіть новий файл у текстовому редакторі і напишіть кілька рядків тексту у ньому про можливості Python. Кожен рядок повинен починатися з фрази: «Python можна використати для ...» . Збережіть файл з ім’ям learning\_python.txt. Напишіть програму, яка зчитує файл і виводить текст з перебором рядків файла і зі збереженням рядків у списку з подальшим сортуванням списку за довжиною рядків в ньому від найбільшого до найменшого.

Лістинг програми

info\_sort = list()  
with open('learning\_python.txt') as f:  
 for line in f:  
 print(line.rstrip())  
 info\_sort.append(line.rstrip())  
info\_sort.sort()  
info\_sort.reverse()  
print(info\_sort)

Результат виконання програми:

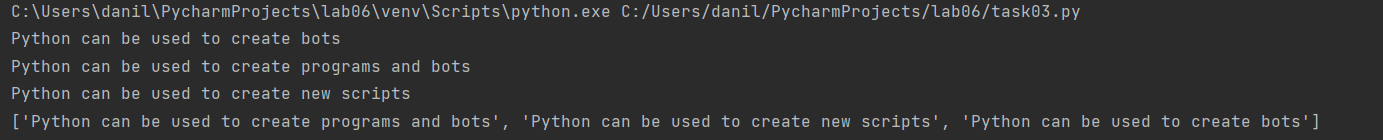


Рис. 5. Результат виконання програми

Результат виконання програми:

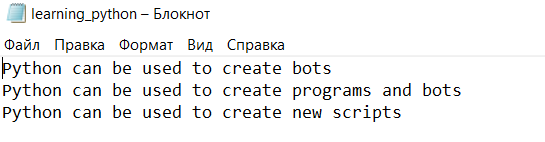


Рис. 6. Результат виконання програми

**Завдання 4:** Прочитайте кожен рядок зі створеного у попередньому завданні файла learning\_python.txt і замініть слово Python назвою іншої мови, наприклад C при виведенні на екран. Отриманий файл має бути створений в новому каталозі, що розміщується в поточному. Відкрийте файл пострічково і дайте можливість користувачеві визначити які змінені фрази є актуальними, наприклад для мови С, а які ні. Всі хибні твердження запишіть в інший файл, а істинні – в поточний.

Лістинг програми

import os  
  
tmp = ''  
tmp1 = ''  
info\_user = ''  
rez\_task4 = ''  
with open('learning\_python.txt') as file\_in:  
 text = file\_in.read()  
text = text.replace("Python", "C")  
os.mkdir("rez\_t4")  
with open('rez\_t4/file.txt', 'w') as f:  
 f.write(text)  
with open('rez\_t4/file.txt', 'r+') as f:  
 while True:  
 line = f.readline()  
 if not line:  
 break  
 print(line)  
 print("Інформація (Актуальна) чи (Не актуальна)? ( Відповідь записати як показано у дужках )")  
 info\_user = str(input("Відповідь: "))  
 if info\_user == "Актуальна":  
 tmp += line + "\n"  
 elif info\_user == "Не актуальна":  
 tmp1 += line + "\n"  
with open('rez\_t4/file.txt', 'w') as f:  
 f.write(tmp)  
with open('rez\_t4/rezult.txt', 'w') as b:  
 b.write(tmp1)

Результат виконання програми:

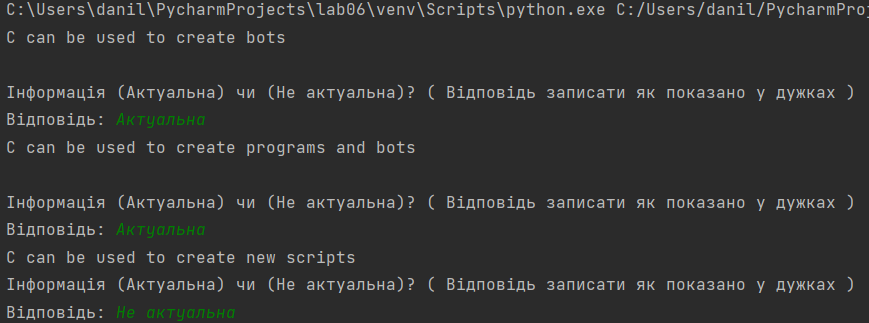


Рис. 7. Результат виконання програми

Результат виконання програми:

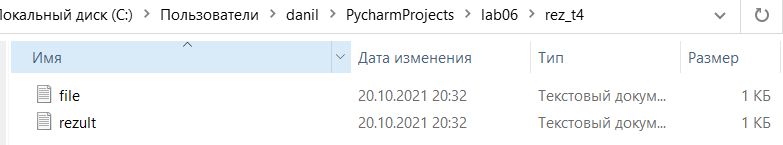
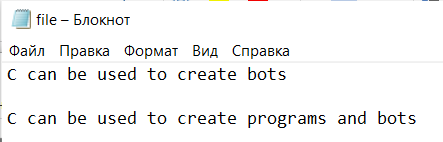


Рис. 8. Результат виконання програми

Результат виконання програми:



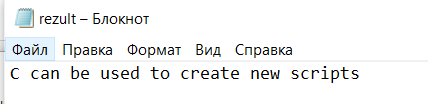


Рис. 9. Результат виконання програми

**Завдання 5:** Створіть порожній файл guest\_book.txt у текстовому редакторі. Напишіть програму, яка запитує у користувачів імена. При введенні кожного імені виведіть на екран рядок з вітанням для користувача і запишіть рядок вітання у файл з ім’ям guest\_book.txt. Простежте за тим, щоб кожне повідомлення розміщувалося в окремому рядку файла з зазначенням часу внесення цього повідомлення. Передбачте зазначення в файлі часу його створення і вказання в ньому часу останніх внесених змін.

Лістинг програми

import os  
import time  
from datetime import datetime  
  
book = open('guest\_book.txt', 'at')  
(mode, ino, dev, nlink, uid, gid, size, atime, mtime, ctime) = os.stat("guest\_book.txt")  
time\_mod = time.ctime(mtime)  
created = time.ctime(os.path.getctime("guest\_book.txt"))  
print("last modified: {0}".format(time\_mod), file=book, sep='', end='\n')  
print("created: {0}".format(created), file=book, sep='', end='\n')  
while True:  
 name\_user = input('Enter a name (q - quit): ')  
 if name\_user == 'q':  
 book.close()  
 break  
 current\_datetime = datetime.now()  
 print('Hello, {0:s}. {1}'.format(name\_user, current\_datetime), file=book)  
 print('Hello, {0:s}. {1}'.format(name\_user, current\_datetime))

Результат виконання програми:

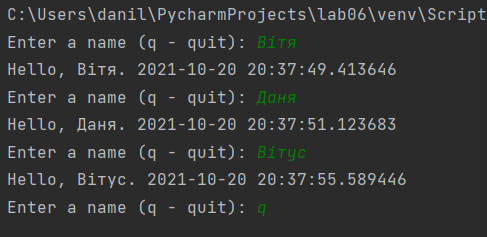


Рис. 10. Результат виконання програми

Результат виконання програми:

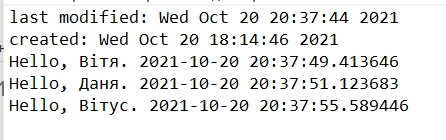


Рис. 11. Результат виконання програми

**Завдання 6:** Збережіть в тектовому файлі публікацію про Python на 3000 слів англійською мовою. Напишіть програму, що аналізуватиме частоту з якою в тексті зустрічастимуться окремі літери чи слова незалежно від їх регістру. Результат робот програми має виводитись в консоль і зберігатись в окремому файлі з зазначенням часу його стоврення, часу виконнання окремих змін, результатів пошкуку і часу, що знадовся на виконнання цього пошуку.

Лістинг програми

import os  
import time  
  
tmp = str(input("Введіть те, що будете шукати у тексті на англ.мові: "))  
tmp1 = tmp.lower()  
startTime = time.time()  
with open('task06\_info.txt', 'rt') as br:  
 full\_info = br.read()  
 count = full\_info.lower().count(tmp1)  
 print(count)  
totalTime = str(round(time.time() - startTime, 10))  
rez = open('rez\_task6.txt', 'w')  
(mode, ino, dev, nlink, uid, gid, size, atime, mtime, ctime) = os.stat("rez\_task6.txt")  
time\_mod = time.ctime(mtime)  
created = time.ctime(os.path.getctime("rez\_task6.txt"))  
print("last modified: {0}".format(time\_mod), file=rez, sep='', end='\n')  
print("created: {0}".format(created), file=rez, sep='', end='\n')  
print("time of searching: {0}".format(totalTime), file=rez, sep='', end='\n')  
rez.write(str(count))  
print(totalTime)

Результат виконання програми:

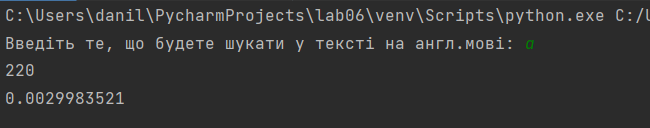


Рис. 12. Результат виконання програми

Результат виконання програми:

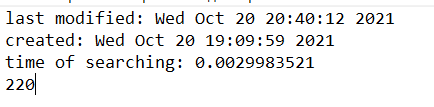


Рис. 13. Результат виконання програми

***Висновки:*** ознайомилися з засобами роботи з файлами в мові Python, діями над ними.