

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

ІКТА
Кафедра ЗІ



Звіт
до лабораторної роботи №3

з курсу: «Програмування скриптовими мовами» на тему: «Робота з рядками»

Виконав:

студент групи КБ-305

Семенчук А.А.

Прийняв:

к.т.н., доцент

Совин Я. Р.

Львів 2024

Мета роботи – ознайомитись з вбудованими рядковими типами Python та операторами і функціями для роботи з ними.

Завдання

2. ЗАВДАННЯ

2.1. Домашня підготовка до роботи 1. Вивчити теоретичний матеріал.

2.2. Виконати в лабораторії

1. Написати програму, яка для заданого тексту використовуючи арифметичні і рядкові операції та операції форматування тексту виводить відформатований текст. Форматування здійснити трьома способами: з допомогою оператора %, методу format() і форматуваних рядкових літералів. Текст для форматування: «Сьогодні в "а" буде проходити позачергове засідання комітету з проблем "b", організоване "с". Було подано "d" заявок на загальну суму "е" тис. гривень. Середня вартість проекту склала "g" гривень.» Наприклад: "а" – 12.30, "b" - «Інформатизація освіти», "с" – МОНУ, "d" – 213, "е" – 123000, "g" = e/d – з точністю до 2-х знаків після коми. Параметри вводяться з клавіатури в діалоговому режимі.

2. Написати програму валідації введеного паролю. Програма не повинна використовувати умовні оператори (if та інші), цикли, регулярні вирази, списки, множини, словники та інші структури даних, функції, класи чи сторонні бібліотеки окрім colorama. Користувач повинен ввести пароль, програма має перевірити наявність у ньому різних типів символів (хоча б одного кожного заданого типу) згідно варіанту у табл. 1 і вивести інформацію про результати перевірки у форматі як показано на рис. 1. Номер варіанту відповідає номеру в списку групи.

3. Написати програму генерації паролю. Програма не повинна використовувати умовні оператори (if та інші), цикли, регулярні вирази, функції, класи чи сторонні бібліотеки окрім random. Користувач повинен ввести кількість різних типів символів у паролі згідно варіанту у табл. 2 і вивести згенерований пароль у форматі як показано на рис. 2.

4. Задані імена файлів у каталозі у вигляді рядка: dir = ("_file1.doc\n" "file2.pdf\n" "file222_.docx\n" "cmd.exe\n" "sys.dll\n" "File7_5.txt\n" "foto1.jpg\n" "song1.mp3\n" "!!!song2.mp3\n" "video.avi\n" "file9.txt\n" "file_3_document.docx\n" "my_document!!!.ppt\n" "main.c\n" "lab3.py\n" "lookup.xml\n" "pic1.png\n" "pic2.bmp\n") Написати програму яка б:

1. Підраховувала і виводила кількість файлів у каталозі.

2. Підраховувала і виводила кількість файлів з заданими розширеннями згідно стовпця А табл. 3.

3. Перейменовувала розширення у всіх файлів згідно стовпця В табл. 3.

4. Приводила імена всіх файлів до заданого регістру згідно стовпця С табл. 3.

5. Видаляла всі файли з заданим розширенням згідно стовпця D табл. 3. Оформлення виводу інформації має виглядати таким чином: Початковий вміст каталогу _file1.doc file2.pdf ... pic1.png pic2.bmp У каталозі є 18 файлів Файлів з розширенням docx: 2 Файлів з розширенням c: 1 23 Каталог після заміни розширення .doc на .docx _file1.docx file2.pdf file222_.docx ... pic1.png pic2.bmp Каталог після приведення імен файлів до верхнього регістру _FILE1.DOCX FILE2.PDF ... PIC1.PNG PIC2.BMP Каталог після видалення файлу PIC1.PNG _FILE1.DOCX FILE2.PDF ... LOOKUP.XML PIC2.BMP

Варіанти завдань

Варіант	Довжина, символів	Набір символів
27.	10	low_char + num_char + spc_char + upp_char

Варіанти завдань

Варіант	Довжина паролю, символів	Набір символів
27.	8	low_char(4) + num_char(2) + spc_char(1) + upp_char(1)

Виконання лабораторної роботи

1.

```
# Введення параметрів
a = input("Введіть час засідання (наприклад, 12:30): ")
b = input("Введіть назву проблеми: ")
c = input("Введіть організацію: ")
d = int(input("Введіть кількість заявок: "))
e = float(input("Введіть загальну суму (в тисячах гривень): "))
g = round(e / d, 2)

# Форматування з оператором %
print("\nОператор %:")
print("Сьогодні в \"%s\" буде проходити позачергове засідання комітету з проблем \"%s\", організоване \"%s\". "
      "Було подано \"%d\" заявок на загальну суму \"%.2f\" тис. гривень. Середня вартість проекту склала \"%.2f\" гривень." % (a, b, c, d, e, g))

# Форматування з методом format()
print("\nМетод format():")
print("Сьогодні в \"{a}\" буде проходити позачергове засідання комітету з проблем \"{b}\", організоване \"{c}\". "
      "Було подано \"{d}\" заявок на загальну суму \"{e:.2f}\" тис. гривень. Середня вартість проекту склала \"{g:.2f}\" гривень.".format(a, b, c, d, e, g))

# Форматування з використанням f-строк
print("\nF-строки:")
print(f"Сьогодні в \"{a}\" буде проходити позачергове засідання комітету з проблем \"{b}\", організоване \"{c}\". "
      f"Було подано \"{d}\" заявок на загальну суму \"{e:.2f}\" тис. гривень. Середня вартість проекту склала \"{g:.2f}\" гривень.")
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
@Velo123 → /workspaces/python-labs (main) $ mkdir lab3
@Velo123 → /workspaces/python-labs (main) $ cd lab3
@Velo123 → /workspaces/python-labs/lab3 (main) $ touch format_text.py
@Velo123 → /workspaces/python-labs/lab3 (main) $ code format_text.py
@Velo123 → /workspaces/python-labs/lab3 (main) $ python format_text.py
Введіть час засідання (наприклад, 12:30): 12:30
Введіть назву проблеми: тест
Введіть організацію: тестова
Введіть кількість заявок: 16
Введіть загальну суму (в тисячах гривень): 16000

Оператор %:
Сьогодні в "12:30" буде проходити позачергове засідання комітету з проблем "тест", організоване "тестова". Було подано "16" заявок на загальну суму "16000.00" тис. гривень. Середня вартість проекту склала "1000.00" гривень.

Метод format():
Сьогодні в "12:30" буде проходити позачергове засідання комітету з проблем "тест", організоване "тестова". Було подано "16" заявок на загальну суму "16000.00" тис. гривень. Середня вартість проекту склала "1000.00" гривень.

F-строки:
Сьогодні в "12:30" буде проходити позачергове засідання комітету з проблем "тест", організоване "тестова". Було подано "16" заявок на загальну суму "16000.00" тис. гривень. Середня вартість проекту склала "1000.00" гривень.
@Velo123 → /workspaces/python-labs/lab3 (main) $
```

2.

```
from colorama import Fore, Style

# Введення паролю
password = input("Введіть пароль (мінімум 8 символів): ")

# Змінні для повідомлень
length_message = ("Довжина не менше 8 символів - FAIL!", "Довжина не менше 8 символів - OK!")
upp_message = ("Великі літери - FAIL!", "Великі літери - OK!")
low_message = ("Маленькі літери - FAIL!", "Маленькі літери - OK!")
num_message = ("Цифри - FAIL!", "Цифри - OK!")
src_message = ("Спеціальні символи - FAIL!", "Спеціальні символи - OK!")
valid_message = ("Пароль не валідний!", "Пароль валідний!")

# Перевірки
length_check = len(password) >= 8

# Перевірка на наявність великих літер
upp_check = sum([
    "A" in password, "B" in password, "C" in password, "D" in password, "E" in password,
    "F" in password, "G" in password, "H" in password, "I" in password, "J" in password,
    "K" in password, "L" in password, "M" in password, "N" in password, "O" in password,
    "P" in password, "Q" in password, "R" in password, "S" in password, "T" in password,
    "U" in password, "V" in password, "W" in password, "X" in password, "Y" in password,
    "Z" in password
]) >= 1

# Перевірка на наявність маленьких літер
low_check = sum([
    "a" in password, "b" in password, "c" in password, "d" in password, "e" in password,
    "f" in password, "g" in password, "h" in password, "i" in password, "j" in password,
    "k" in password, "l" in password, "m" in password, "n" in password, "o" in password,
    "p" in password, "q" in password, "r" in password, "s" in password, "t" in password,
    "u" in password, "v" in password, "w" in password, "x" in password, "y" in password,
    "z" in password
]) >= 1
```

```
    "f" in password, "g" in password, "h" in password, "i" in password, "j" in
password,
    "k" in password, "l" in password, "m" in password, "n" in password, "o" in
password,
    "p" in password, "q" in password, "r" in password, "s" in password, "t" in
password,
    "u" in password, "v" in password, "w" in password, "x" in password, "y" in
password,
    "z" in password
]) >= 1

# Перевірка на наявність цифр
num_check = sum([
    "0" in password, "1" in password, "2" in password, "3" in password, "4" in
password,
    "5" in password, "6" in password, "7" in password, "8" in password, "9" in
password
]) >= 1

# Перевірка на наявність спеціальних символів
spc_check = sum([
    "!" in password, "@" in password, "#" in password, "$" in password, "%" in
password,
    "^" in password, "&" in password, "*" in password, "_" in password, "-" in
password
]) >= 1

# Вивід результатів перевірки
print(Fore.GREEN + length_message[length_check] + Style.RESET_ALL)
print(Fore.GREEN + upp_message[upp_check] + Style.RESET_ALL)
print(Fore.GREEN + low_message[low_check] + Style.RESET_ALL)
print(Fore.GREEN + num_message[num_check] + Style.RESET_ALL)
print(Fore.GREEN + spc_message[spc_check] + Style.RESET_ALL)

# Вивід загальної валідності пароля
valid = length_check * upp_check * low_check * num_check * spc_check
print(Fore.GREEN + valid_message[valid] + Style.RESET_ALL)
```

```

@Velovo123 →/workspaces/python-labs/lab3 (main) $ python validate_password.py
Введіть пароль (мінімум 8 символів): 01234
Довжина не менше 8 символів - FAIL!
Великі літери - FAIL!
Маленькі літери - FAIL!
Цифри - OK!
Спеціальні символи - FAIL!
Пароль не валідний!
@Velovo123 →/workspaces/python-labs/lab3 (main) $ python validate_password.py
Введіть пароль (мінімум 8 символів): dfghAA45&*
Довжина не менше 8 символів - OK!
Великі літери - OK!
Маленькі літери - OK!
Цифри - OK!
Спеціальні символи - OK!
Пароль валідний!
@Velovo123 →/workspaces/python-labs/lab3 (main) $ python validate_password.py
Введіть пароль (мінімум 8 символів): мій_пароль!!!
Довжина не менше 8 символів - OK!
Великі літери - FAIL!
Маленькі літери - FAIL!
Цифри - FAIL!
Спеціальні символи - OK!
Пароль не валідний!
@Velovo123 →/workspaces/python-labs/lab3 (main) $ 

```

3.

```

import random

# Інформація про студента
print("Семенчук Анатолій Анатолійович, КБ-305, 2024. Варіант 27")

# Введення кількості символів
upp_count = int(input("Введіть кількість великих латинських літер в паролі: "))
low_count = int(input("Введіть кількість малих латинських літер в паролі: "))
num_count = int(input("Введіть кількість цифр в паролі: "))
spc_count = int(input("Введіть кількість спеціальних символів !@#$%^&*_- в паролі: "))

# Символи для паролю
low_char = "abcdefghijklmnopqrstuvwxyz"
upp_char = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
num_char = "0123456789"
spc_char = "!@#$%^&*_- "

# Формування паролю
password = (
    random.sample(low_char, low_count) +
    random.sample(num_char, num_count) +
    random.sample(spc_char, spc_count) +
    random.sample(upp_char, upp_count)
)

# Перемішування символів і виведення паролю
random.shuffle(password)
print("Password:", "".join(password))

```

```
@VeIovo123 →/workspaces/python-labs/lab3 (main) $ python generate_password.py
Семенчук Анатолій Анатолійович, КБ-305, 2024. Варіант 27
Введіть кількість великих латинських літер в паролі: 1
Введіть кількість малих латинських літер в паролі: 4
Введіть кількість цифр в паролі: 2
Введіть кількість спеціальних символів !@#$%^&*_- в паролі: 1
Password: wd!40ts1
```

4.

```
# Початковий вміст каталогу
dir_content = (
    "_file1.doc\n"
    "file2.pdf\n"
    "file222_.docx\n"
    "cmd.exe\n"
    "sys.dll\n"
    "File7_5.txt\n"
    "foto1.jpg\n"
    "song1.mp3\n"
    "!!!song2.mp3\n"
    "video.avi\n"
    "file9.txt\n"
    "file_3_document.docx\n"
    "my_document!!!.ppt\n"
    "main.c\n"
    "lab3.py\n"
    "lookup.xml\n"
    "pic1.png\n"
    "pic2.bmp\n"
)

# Перетворення на список для обробки
files = dir_content.strip().split("\n")

# Підрахунок загальної кількості файлів
total_files = len(files)
print("У каталозі є", total_files, "файлів")

# Підрахунок файлів з розширеннями mp3 і png
mp3_files = sum(file.endswith(".mp3") for file in files)
png_files = sum(file.endswith(".png") for file in files)
print("Файлів з розширенням mp3:", mp3_files)
print("Файлів з розширенням png:", png_files)

# Заміна розширення .avi на .xml
files = [file.replace(".avi", ".xml") if file.endswith(".avi") else file for file
in files]
print("\nКаталог після заміни розширення .avi на .xml:")
print("\n".join(files))

# Приведення імен файлів до "Т" - перша велика літера
```

```

files = [file.title() for file in files]
print("\nКаталог після приведення імен файлів до формату 'T':")
print("\n".join(files))

# Видалення файлів з розширенням xml
files = [file for file in files if not file.endswith(".xml")]
print("\nКаталог після видалення файлів з розширенням xml:")
print("\n".join(files))

```

```

@VeloVo123 →/workspaces/python-labs/lab3 (main) $ python file_operations.py
У каталозі є 18 файлів
Файлів з розширенням mp3: 2
Файлів з розширенням png: 1

```

```

Каталог після заміни розширення .avi на .xml:
_file1.doc
file2.pdf
file222_.docx
cmd.exe
sys.dll
File7_5.txt
foto1.jpg
song1.mp3
!!!song2.mp3
video.xml
file9.txt
file_3_document.docx
my_document!!!.ppt
main.c
lab3.py
lookup.xml
pic1.png
pic2.bmp

```

```

Каталог після приведення імен файлів до формату 'T':
_File1.Doc
File2.Pdf
File222_.Docx
Cmd.Exe
Sys.Dll
File7_5.Txt
Foto1.Jpg
Song1.Mp3
!!!Song2.Mp3
Video.Xml
File9.Txt
File_3_Document.Docx
My_Document!!!.Ppt
Main.C
Lab3.Py
Lookup.Xml
Pic1.Png
Pic2.Bmp

```

```

Каталог після видалення файлів з розширенням xml:
_File1.Doc
File2.Pdf
File222_.Docx
Cmd.Exe
Sys.Dll
File7_5.Txt
Foto1.Jpg
Song1.Mp3
!!!Song2.Mp3
Video.Xml
File9.Txt
File_3_Document.Docx
My_Document!!!.Ppt
Main.C
Lab3.Py
Lookup.Xml
Pic1.Png
Pic2.Bmp

```

Висновок: У ході виконання роботи ми ознайомилися з вбудованими рядковими типами Python, а також із різними операторами і функціями для їх обробки. Ми навчилися форматовувати рядки за

допомогою оператора `%`, методу `format()` та f-строк, що дозволяє легко вставляти значення змінних у текст і налаштовувати його відображення. Також була створена програма для перевірки валідності пароля, яка використовує перевірку вмісту рядка за заданими критеріями без використання умовних операторів та циклів. Це дало змогу розширити наші навички роботи з текстовими даними та познайомитися з можливостями Python для ефективної обробки рядкових типів.