Міністерство освіти і науки України

Національний технічний університет «ХПІ»

Навчально-науковий інститут комп’ютерних наук та інформаційних

технологій

Кафедра комп’ютерної інженерії та програмування

**ЗВІТ**

з практичного заняття №3

з дисципліни «Проектування серверних застосунків»

Виконала: студентка групи КН-919д

Федюкіна Поліна Олегівна

Перевірив: старший викладач

Лисиця Дмитро Олександрович

Харків - 2022

**Тема**: Підключення модулів. ABC-класи. Списки

**Мета**: Розгляд особливостей використання модулів, ABC-класів та списків.

**Хід роботи**

Таблиця 1.1 – Індивідуальне завдання для варіанту 5 (25)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Варіант | Операції | Рядок | | Функції |
| 5(25) | S.insert(i,x)  S[i:j] = t  S.reverse() | Зміст | Розділювач | len(), max() |
| слова | # |
|

1. Розробимо програму мовою Python, яка виконує наступні функції:
2. Створює список a\_list, елементами якого є об’єкти типів: int, float, bool, str та list;
3. Для кожного елемента списку визначає тип елементу та ABC-клас;
4. Для списка a\_list виконує операції, номера яких задані у колонці «Операції» табл. №2;
5. Створює рядок st, який містить слова або числа, розділенні символами, вказаними у колонці «Рядок/Розділювач» табл. №2;
6. Перетворює рядок st в список b\_list, елементами якого є рядки або числа;
7. Виконує над елементами списку b\_list функції, вказані у колонці «Функції»;
8. Виводить результат виконання функцій на екран.

Пишемо код наведений нижче:

import collections

a\_list = [13, 5.78, 'James', ['a', 'b', 'c'], False]

for i in a\_list:

print(f'Тип: {type(i)} Значение: {i}')

if isinstance(i, collections.abc.Container):

print("Объект пренадлежит к ABC-классу Container")

elif isinstance(i, collections.abc.Hashable):

print("Объект пренадлежит к ABC-классу Hashable")

elif isinstance(i, collections.abc.Iterable):

print("Объект пренадлежит к ABC-классу Iterable")

elif isinstance(i, collections.abc.Iterator):

print("Объект пренадлежит к ABC-классу Iterator")

elif isinstance(i, collections.abc.Reversible):

print("Объект пренадлежит к ABC-классу Reversible")

elif isinstance(i, collections.abc.Generator):

print("Объект пренадлежит к ABC-классу Generator")

elif isinstance(i, collections.abc.Sized):

print("Объект пренадлежит к ABC-классу Sized")

elif isinstance(i, collections.abc.Callable):

print("Объект пренадлежит к ABC-классу Callable")

elif isinstance(i, collections.abc.Collection):

print("Объект пренадлежит к ABC-классу Collection")

elif isinstance(i, collections.abc.Sequence):

print("Объект пренадлежит к ABC-классу Sequence")

elif isinstance(i, collections.abc.MutableSequence):

print("Объект пренадлежит к ABC-классу MutableSequence")

elif isinstance(i, collections.abc.Set):

print("Объект пренадлежит к ABC-классу Set")

elif isinstance(i, collections.abc.MutableSet):

print("Объект пренадлежит к ABC-классу MutableSet")

elif isinstance(i, collections.abc.Mapping):

print("Объект пренадлежит к ABC-классу Mapping")

elif isinstance(i, collections.abc.MutableMapping):

print("Объект пренадлежит к ABC-классу MutableMapping")

elif isinstance(i, collections.abc.MappingView):

print("Объект пренадлежит к ABC-классу MappingView")

elif isinstance(i, collections.abc.ItemsView):

print("Объект пренадлежит к ABC-классу ItemsView")

elif isinstance(i, collections.abc.Awaitable):

print("Объект пренадлежит к ABC-классу Awaitable")

elif isinstance(i, collections.abc.Awaitable):

print("Объект пренадлежит к ABC-классу Awaitable")

elif isinstance(i, collections.abc.Coroutine):

print("Объект пренадлежит к ABC-классу Coroutine")

elif isinstance(i, collections.abc.AsyncIterable):

print("Объект пренадлежит к ABC-классу AsyncIterable")

elif isinstance(i, collections.abc.AsyncIterator):

print("Объект пренадлежит к ABC-классу AsyncIterator")

elif isinstance(i, collections.abc.AsyncGenerator):

print("Объект пренадлежит к ABC-классу AsyncGenerator")

print()

a\_list.insert(1, 43)

a\_list.reverse()

a\_list = a\_list[2:5]

print()

for i in range(len(a\_list)):

print(f'Type: {type(a\_list[i])} Value: {a\_list[i]}')

print()

st = 'Hello#my#name#is#Lina'

b\_list = st.split('#')

for i in range(len(b\_list)):

print(f'Элемент {i} - {b\_list[i]} len() - {len(b\_list[i])}, max() - {max(b\_list[i])}')

Результат виконання коду наведений нижче на рис. 1.1

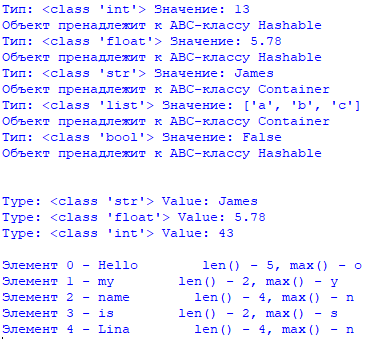


Рисунок 1.1 – Результат виконання програми

**Висновки**: під час виконання цієї практичної роботи розглянула та навчилася користуватися модулями, ABC-класами, створювати списки за допомогою мови Python.