

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І
ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ
Факультет інформаційних технологій**

Кафедра комп'ютерних систем, мереж і кібербезпеки

Кросплатформне програмування (Python)

Лабораторна робота 1: *Створення віртуального оточення Python*

Виконав
студент групи КІ-210126
Яловенко Володимир Вікторович

Київ 2024

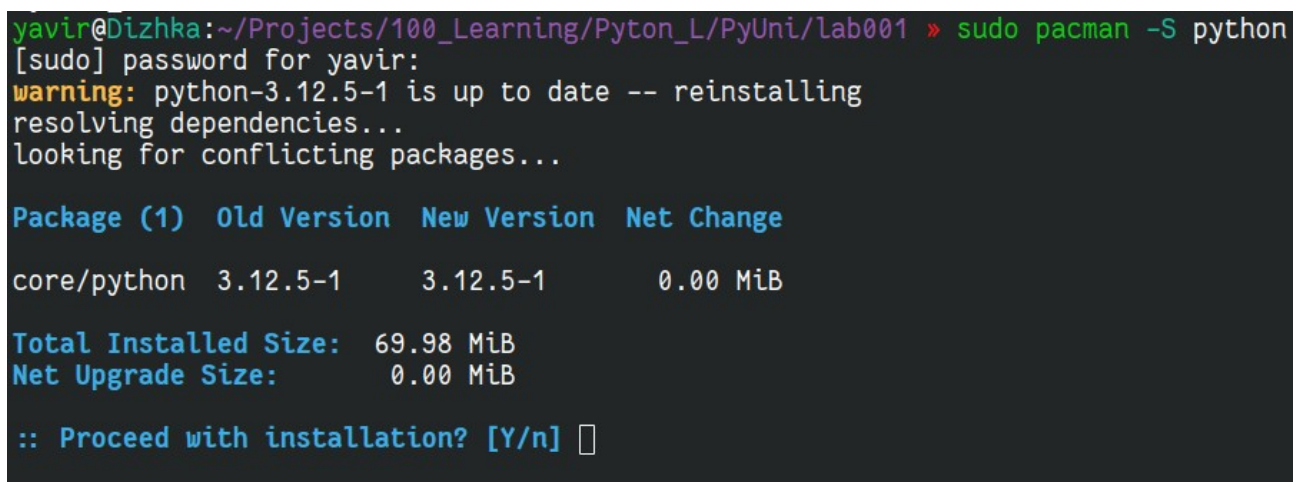
Завдання:

1. Встановити інтерпретатор Python (<https://www.python.org/>)
2. Встановити середовище PyCharm (або Visual Studio Code)
3. Створити віртуальне оточення (ім'я оточення - прізвище студента). В цьому оточенні створити наступні програми на мові Python(в різних файлах)
 - Програма для виводу всіх простих чисел, які розташовані між числами a та b , які задає користувач (не обов'язково $a < b$) .
 - Створити рекурсивну функцію `fib(k)`, яка повертає список n чисел Фібоначчі. Написати програму яка виводить k чисел Фібоначчі, використовуючи функцію `fib(k)`.

1. Встановлення інтерпритатора Python

На своїй основній машині, я використовую операційну систему Arch Linux. На ній, встановлення пакету можна досягнути однією командою.

```
$: sudo pacman -S python
```



```
yavir@Dizhka:~/Projects/100_Learning/Python_L/PyUni/lab001 » sudo pacman -S python
[sudo] password for yavir:
warning: python-3.12.5-1 is up to date -- reinstalling
resolving dependencies...
looking for conflicting packages...

Package (1)  Old Version  New Version  Net Change
core/python  3.12.5-1     3.12.5-1     0.00 MiB

Total Installed Size: 69.98 MiB
Net Upgrade Size:    0.00 MiB

:: Proceed with installation? [Y/n] ☐
```

Рисунок 1: Встановлення Python

Перевіримо версію встановленого інтерпритатора



```
yavir@Dizhka:~/Projects/100_Learning/Python_L/PyUni/lab001 » python --version
Python 3.12.4
```

Рисунок 2: Версія Python

2. Встановити середовище PyCharm (або Visual Studio Code)

Встановлення Visual Studio Code є дуже подібним до встановлення інших пакетів. Єдине що, Visual Studio Code нема у офіційних репозиторіях, тож завантажемо його через AUR (Arch user repository) helper **yay**.

```
yavir@dizhka:~/Projects/100_Learning/Python_L/PyUni/Lab001 » yay -S code
Sync Explicit (1): code-1.92.1-1
[sudo] password for yavir:
warning: code-1.92.1-1 is up to date -- reinstalling
resolving dependencies...
looking for conflicting packages...

Package (1)  Old Version  New Version  Net Change
extra/code   1.92.1-1     1.92.1-1     0.00 MiB

Total Installed Size: 83.73 MiB
Net Upgrade Size:      0.00 MiB
```

Рисунок 3: Встановлення Visual Studio Code

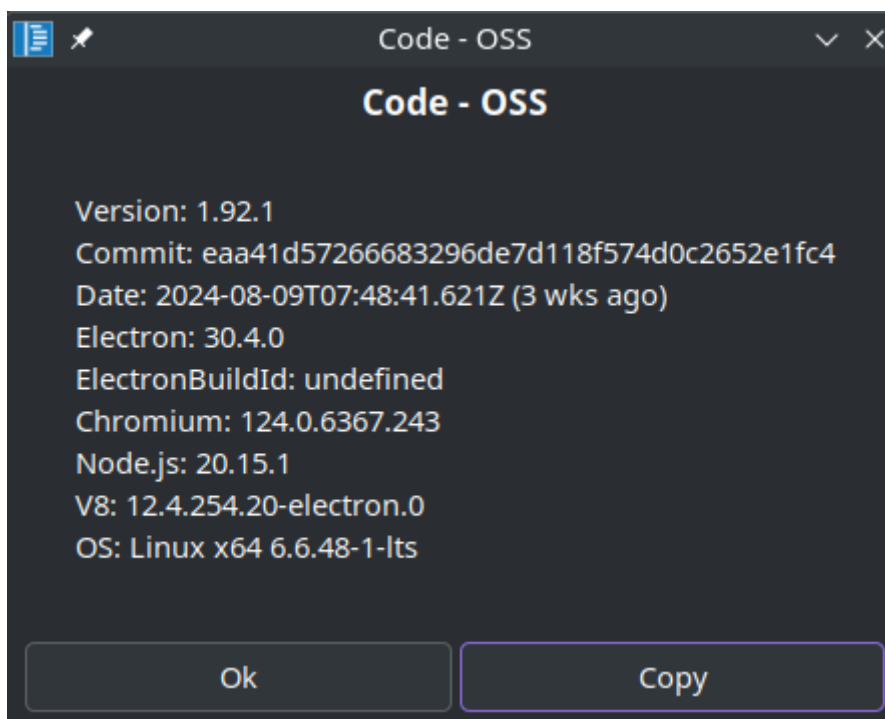


Рисунок 4: Вікно About Visual Studio Code

3. Створити віртуальне оточення

Для написання програм, будемо використовувати віртуальне оточення Python

```
yavir@Dizhka:~/Projects/100_Learning/Python_L/PyUni/lab001 » py -m venv Yalovenko
yavir@Dizhka:~/Projects/100_Learning/Python_L/PyUni/lab001 » source Yalovenko/bin/activate
(Yalovenko) yavir@Dizhka:~/Projects/100_Learning/Python_L/PyUni/lab001 » ls Yalovenko
bin include lib lib64 pyenv.cfg
(Yalovenko) yavir@Dizhka:~/Projects/100_Learning/Python_L/PyUni/lab001 »
```

Рисунок 5: Створення віртуального середовища

У цьому віртуальному оточенні створюємо файли, які містять код програм, функції яких визначені у завданні.

Код програми для виводу всіх простих чисел:

```
def is_prime(num):
    if num > 1:
        for i in range(2, (num//2) + 1):
            if (num % i) == 0:
                return False
        else:
            return True
    else:
        return False

def prime_range(num1, num2):
    primeList = []
    for i in range(num1, num2):
        if is_prime(i) is True:
            primeList.append(i)
    return primeList

primeList = []
print("Enter numbers to define a prime search range: ")
num1 = int(input("Number a: "))
num2 = int(input("Number b: "))

primeList = prime_range(num1, num2)

for i in primeList:
    print(str(i) + " ", end="")
print("")
```

```
(Yalovenko) yavir@Dizhka:~/Projects/100_Learning/Python_L/PyUni/lab001/Yalovenko » python prime_range.py
Enter numbers to define a prime search range:
Number a: 13
Number b: 98
13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97
```

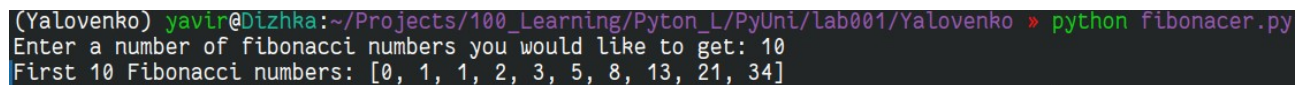
Рисунок 6: Результат роботи програми 1

Код рекурсивної функції fib(k)

```
def fib(k):  
    if k == 1:  
        return [0]  
    elif k == 2:  
        return [0, 1]  
    else:  
        fib_list = fib(k - 1)  
        fib_list.append(fib_list[-1] + fib_list[-2])  
        return fib_list
```

```
k = int(input("Enter a number of fibonacci numbers you would like  
to get: "))  
fib_sequence = fib(k)  
print(f"First {k} Fibonacci numbers: {fib_sequence}")
```

Результат роботи програми:



```
(Yalovenko) javir@Dizhka:~/Projects/100_Learning/Python_L/PyUni/lab001/Yalovenko » python fibonacer.py  
Enter a number of fibonacci numbers you would like to get: 10  
First 10 Fibonacci numbers: [0, 1, 1, 2, 3, 5, 8, 13, 21, 34]
```

Рисунок 7: Результат роботи програми 2