

### **Практическое задание №13. Настройка функций и модулей. Обработка исключений и работа с файлами.**

**Цель:** научиться использовать современные методы контроля производительности информационно-коммуникационной системы и локализовывать отказ и инициировать корректирующие действия

**Задание:** Необходимо создать и редактировать, в соответствии с заданием, скрипты на языке программирования Python.

#### **Инструкция:**

**Шаг 1:** Написать python скрипт анализа размера каждой директории и файла в текущей директории с выводом результатов в терминал, используя функции и циклы;

**Шаг 2:** Добавить сортировку по уменьшению размера;

**Шаг 3:** Прикрепить скриншот вывода работы скрипта.

**Шаг 4:** Прикрепить файл скрипта.

#### **ВЫПОЛНЕНИЕ ЗАДАНИЯ**

**Шаг 1:** Написать python скрипт анализа размера каждой директории и файла в текущей директории с выводом результатов в терминал, используя функции и циклы;

- Устанавливаем python3 при помощи apt:

```
sudo apt install python3
```

- Создаем функцию получения размера при помощи вызова подпрограммы du:

```
command = ['du', '--bytes', '--summarize', entry_name]
du_output = subprocess.check_output(command)
size = int(du_output.split()[0].decode('utf-8'))
```

- Получаем список файлов в текущей директории при помощи os.listdir и создаем новый список файлов, который содержит имена файлов вместе с их размерами (см. скрин ниже. get\_current\_directory\_entries\_with\_size функция)

```
1  #!/usr/bin/env python3
2  import os
3  import subprocess
4
5  # Функция возвращает размер записи файловой системы по имени.
6  def get_entry_size(entry_name):
7      try:
8          command = ['du', '--bytes', '--summarize', entry_name]
9          du_output = subprocess.check_output(command)
10         size = int(du_output.split()[0].decode('utf-8'))
11     except:
12         return 0
13
14     return size
15
16 # Функция возвращает список записей файловой системы вместе с их размерами.
17 def get_current_directory_entries_with_size():
18     entries = []
19
20     entries_names = os.listdir()
21     for entry_name in entries_names:
22         entries.append({
23             'name': entry_name,
24             'size': get_entry_size(entry_name),
25         })
26
27     return entries
28
29 # Главная функция выводит список записей файловой системы.
30 def main():
31     entries = get_current_directory_entries_with_size()
32
33     print("Размер (байты) | Имя файла")
34     for entry in entries:
35         print("%14s | %s" % (entry['size'], entry['name']))
36
37 if __name__ == "__main__":
38     main()
```

**Шаг 2:** Добавить сортировку по уменьшению размера;

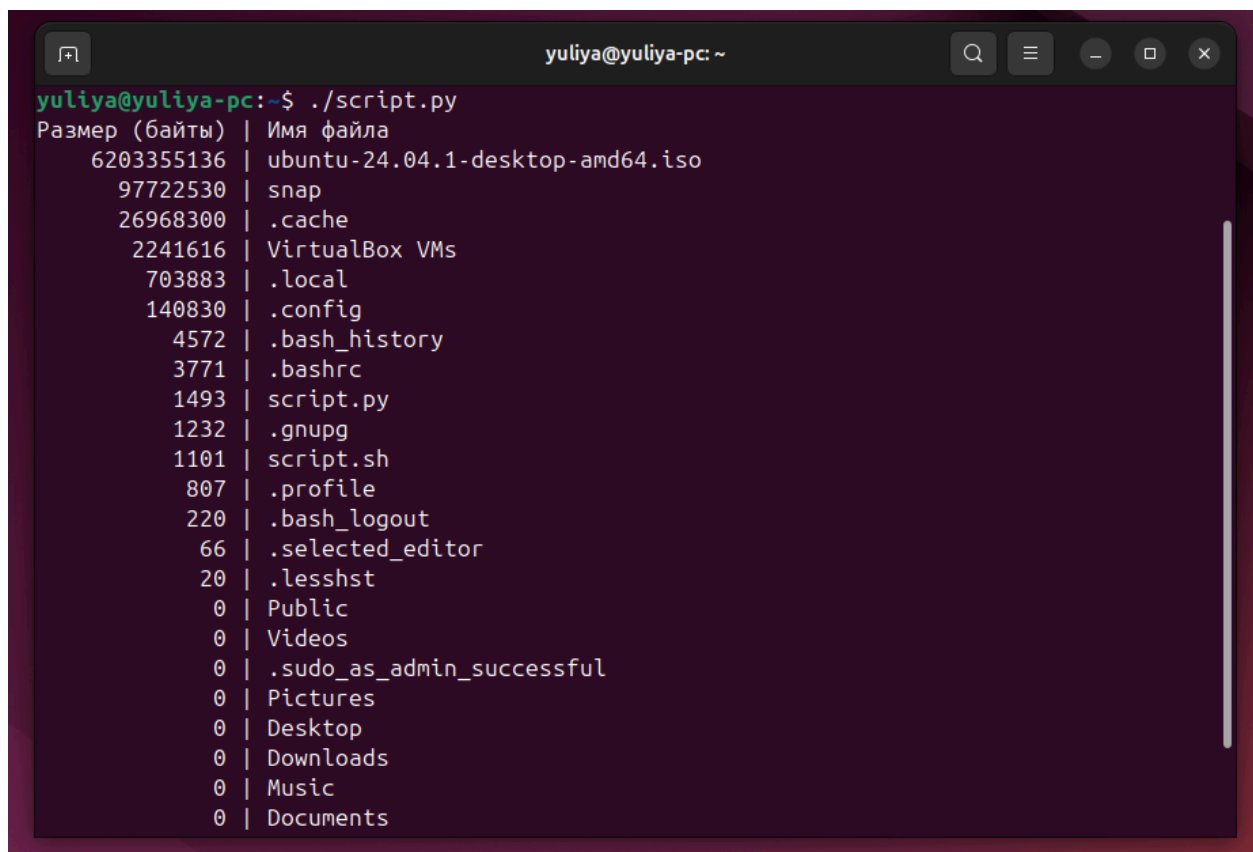
- Добавляем функцию сортировки по уменьшению размера файла, используя встроенный в список метод `sort` и отдельную функцию для получения ключа (размера), по которому нужно сортировать.

```

28
29 # Функция принимает список записей и сортирует их по уменьшению размера.
30 def sort_downsizing(entries):
31     def key_size(entry):
32         return entry['size']
33     entries.sort(key = key_size, reverse = True)
34
35 # Главная функция выводит список записей файловой системы.
36 def main():
37     entries = get_current_directory_entries_with_size()
38     sort_downsizing(entries)
39
40     print("Размер (байты) | Имя файла")
41     for entry in entries:
42         print("%14s | %s" % (entry['size'], entry['name']))
43
44 if __name__ == "__main__":
45     main()
46

```

**Шаг 3:** Прикрепить скриншот вывода работы скрипта.



```

yuliya@yuliya-pc: ~
yuliya@yuliya-pc:~$ ./script.py
Размер (байты) | Имя файла
6203355136 | ubuntu-24.04.1-desktop-amd64.iso
97722530 | snap
26968300 | .cache
2241616 | VirtualBox VMs
703883 | .local
140830 | .config
4572 | .bash_history
3771 | .bashrc
1493 | script.py
1232 | .gnupg
1101 | script.sh
807 | .profile
220 | .bash_logout
66 | .selected_editor
20 | .lessht
0 | Public
0 | Videos
0 | .sudo_as_admin_successful
0 | Pictures
0 | Desktop
0 | Downloads
0 | Music
0 | Documents

```

**Шаг 4:** Прикрепить файл скрипта.

Смотрите соседний файл.