

### 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.

Оформите выполнение в текстовый документ, где распишите выполнение каждого шага и прикрепите скриншоты.

#### Выполнение

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

в терминале: создаем файл скрипта: mcedit size.py

в *mc*: импортируем *os* модуль: `import os`

если запускаемый модуль *main*, тогда запускаем функцию *main*: `if __name__ == "__main__":`  
`main()`

определяем функцию: `def main():`

получаем текущую директорию: `pwd = os.getcwd()`

получаем список всех файлов и директорий: `items = os.listdir(pwd)`

делаем цикл: `for item in items:`

вывод: `print (item)`

добавляем размер при помощи функции `def get_size(path):`

`size = 0`

добавляем условие если: `if os.path.isfile(path):`

`size = os.path.getsize(path)`

иначе: `else:`

`for dirpath, dirname, filenames in os.walk(path):`

то возвращаем размер: `return size`

далее цикл: `for filename in filenames:`

соединяем пути: `fp = os.path.join(dirpath, filename)`

добавляем условие (если это файл, то суммируем размер): `if os.path.isfile(fp):`

`size += os.path.getsize(fp)`

добавим полный путь и размер к циклу `for item in items`

`full_path = os.path.join(pwd, item)`

```
size = get_size(full_path)
```

сделаем вывод: `print("{} {}".format(size, item))`

для удобства чтения добавим функцию: `def human_readable_size(size):`

и цикл для определения единиц измерения `for unit in ['B', 'KB', 'MB', 'GB', 'TB']:`

добавляем условие если: `if size < 1024:`

то прервать цикл: `break`

если больше или равен, то делим на 1024: `size /= 1024`

возвращаем размер и ед измерения: `return "{}.1f{}".format(size, unit)`

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

создадим переменную: `size_list = []`

и добавим вместо вывода: `size_list.append((size, item))`

отсортируем по уменьшению размера с помощью функции: `size_list.sort(key=lambda x: x[0], reverse=True)`

цикл: `for size, item in size_list:`

вывод списка директорий и файлов: `print("{} {}".format(human_readable_size(size), item))`

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

```
mary@mary-Lenovo-Ideapad-300-15IBR: ~  
$ mcedit size.py  
$ python3 size.py  
4.6GB Загрузки  
1.1GB snap  
31.8MB .cache  
4.3MB Изображения  
3.2MB VirtualBox VMs  
2.2MB .config  
877.3KB .local  
3.7KB .bashrc  
2.0KB Документы  
1.7KB .bash_history  
1.2KB .gnupg  
1009.0B size.py  
807.0B .profile  
641.0B test4.sh  
275.0B size3.sh  
271.0B size4.sh  
241.0B test2.sh  
220.0B .bash_logout  
215.0B size5.sh  
199.0B show_all.sh  
183.0B size.sh  
15.0B test.sh  
0.0B file.txt  
0.0B Рабочий стол  
0.0B .ssh  
0.0B Музыка  
0.0B .sudo_as_admin_successful  
0.0B Общедоступные  
0.0B Видео  
0.0B Шаблоны  
mary@mary-Lenovo-Ideapad-300-15IBR: ~
```

**Шаг 4:** Прикрепить файл скрипта.

Обзор Терминал Вc, 10 ноября 00:59 ru

mary@mary-Lenovo-Ideapad-300-151BR: ~

```
def get_size(path):
    size = 0
    if os.path.isfile(path):
        size = os.path.getsize(path)
    else:
        for dirpath, dirname, filenames in os.walk(path):
            for filename in filenames:
                fp = os.path.join(dirpath, filename)
                if os.path.isfile(fp):
                    size += os.path.getsize(fp)
    return size

def human_readable_size(size):
    for unit in ['B', 'KB', 'MB', 'GB', 'TB']:
        if size < 1024:
            break
        size /= 1024
    return "{:.1f}{}".format(size, unit)

def main():
    pwd = os.getcwd()
    items = os.listdir(pwd)
    size_list = []
    for item in items:
        full_path = os.path.join(pwd, item)
        size = get_size(full_path)
        size_list.append((size, item))

    size_list.sort(key=lambda x: x[0], reverse=True)
    for size, item in size_list:
        print("{} {}".format(human_readable_size(size), item))

if __name__ == "__main__":
    main()
```

1.Помощь 2.Сохранить 3.Блок 4.Замена 5.Копия 6.Переместить 7.Поиск 8.Удалить 9.Переименовать 10.Выход