Контрольные вопросы

1. Как в Python осуществляется чтение / запись текстовых файлов?

Чтение / запись файлов в Python осуществляется с помощью ф-ции открытия файла open() и функции ~~открытия~~ непосредственно чтения и записи данных

2. Как в Python можно получить информацию о содержимом директории?

Все необходимые функции для работы с файловой системой в Python содержатся в модулях: os, shutil, pathlib, glob.

Пример кода для получения списка только папок в текущей директории:

```
Folders = [entry For entry in os.listdiF(os.getcwd()]
          iF os.path.isdir(entry)]
print (Folders)
```

3. Как в Python можно скопировать, удалить и переименовать файл?

Библиотека под названием shutil включает в себя много полезных функций для копирования

```
import shutil
shutil.copyFile ("D:\\test.txt", "D:\\test2.txt")
```

Удалению файла

```
import os
os.remove ("D:\\test.txt")
```

Переименовать

```
import shutil
shutil.move("D:\\test.txt", "D:\\test1.txt")
```

4. Каковы базовые синтаксические элементы регулярных выражений?

'.' - любой символ

'^' - начало строки

'$' - конец строки

'*' - 0 или более повторений

'+' - 1 или более повторений

'?' - 0 или 1 повторений

'*?', '+?', '??' - ограничить жадность

'*+', '++', '?+' притягательные квантификаторы

'{m}' - m повторений

'{m, n}' - как можно больше повторений

'{m, n}?' - как можно меньше повторений

'{m,n}+' - притягательная версия квантиф

'\' - экранирование специальных символов

'[]' - символьный класс

'|' - или

'(...)' - группа с захватом

Расширения регулярных выражений

'(?aiLmsux)' - установка

'(?aiLmsuy-imsy:..)' - установка и удаление флагов

'(?>...)' - атомарная группа

'(?:...)' - группа без захвата

'(?P<name>...)' - именованная группа

'(?P=name)' - обратная ссылка на именованную группу.

'(?#...)' - комментарий

'(?=...)' - опережающая позитивная проверка

'(?<!...)' - негативная ретроспективная проверка

'(?(id/name)yes-pattern|no-pattern)' - yes pattern

Специальные последовательности.

'\number' - соответствие группы с тем же номером

'\A' только с начала строки

'\b' пустая строка (начало или конец)

'\B' пустая строка (не начало)

'\d' любая десятичная цифра

'\D' - не цифра

'\Z' - только конец

5. Какие возможности предоставляет модуль subprocess?

Модуль subprocess позволяет создавать

При этом он может подключаться к стандартным потокам ввода/вывода и получать код возврата.

Можно выполнить любые команды Linux из скрипта.