



ОНЛАЙН-ОБРАЗОВАНИЕ

Онлайн-образование



Меня хорошо видно && слышно?

Ставьте  , если все хорошо
Напишите в чат, если есть проблемы

Проверить, идет ли запись!



Работа с ОС Linux с помощью Python



Сулагина Виктория

Python QA Engineer

Ozon Технологии

Контакты: victoria.s3517@gmail.com

Преподаватель



Сулагина Виктория

- **Ozon Технологии** - Python QA Engineer
- Eram, Positive Technologies - Software Engineer in Test
- Занимаюсь автоматизацией тестирования на Python более 3-х лет, опыт работы с Python более 5 лет
- Пишу тесты на API (Rest, GraphQL) и UI
- Провожу обучение коллег практикам тестирования и автоматизации

Правила вебинара



Активно участвуем



Задаем вопрос в чат или голосом



Off-topic обсуждаем в Slack #канал группы или #general



Вопросы вижу в чате, могу ответить не сразу

Маршрут вебинара

Процессы



Subprocess, os, sys



Практика



Рефлексия

Цели вебинара | После занятия вы сможете

1

Работать с процессами os из Python кода

2

Работать с файлами и директориями внутри фреймворка и на сервере

Смысл | Зачем вам это уметь

1 Подготовка, поддержка и отладка окружения для автотестов

2 Подготовка вспомогательных скриптов для тестирования систем и мониторинга приложений

The background of the slide features an aerial view of a dense city skyline, likely New York City, with numerous skyscrapers. The image is overlaid with a semi-transparent blue layer. A network diagram, consisting of interconnected nodes and lines, is visible across the blue area, particularly concentrated around the central text. The overall color palette is dominated by various shades of blue and teal.

Процессы и потоки

Процессы

Процесс - запущенная программа, приложение или команда.

Некоторые приложения могут иметь несколько процессов (объединены в группы, группа имеет свой идентификатор, обычно - это идентификатор первого процесса в группе)

Процесс имеет идентификатор - PID (PPID - идентификатор родительского процесса)

Посмотреть полный список процессов: **ps -ef**

Процессы

Какие способы межпроцессного взаимодействия вы знаете?

Способы межпроцессного взаимодействия

- PIPE
- Сокеты
- Коды возврата
- Сигналы
- Переменные окружения

Сигналы

- SIGKILL
- SIGTERM
- SIGSTOP

Посмотреть полный список сигналов: `kill -L`

Коды ответа

- 0 -- Все ОК
- 1 -- Разные ошибки
- 126 -- Невозможно выполнить
- 127 -- Не найдено
- 128 -- Неверный аргумент
- 130 -- Завершение по control-c
- 255+ -- За пределами диапазона

Процессы

- `fork` -- создает новый процесс из переданного.
- `exec` -- запускает процесс.
- `sleep` -- усыпляет.
- `kill` -- посылает сигнал (`SIGTERM`, `SIGKILL`, etc)
- `ps` -- список процессов

Потоки

Поток - это процесс, который использует некоторые ресурсы совместно с другими процессами. Поток имеет определенную структуру и для ядра представляется обычным процессом.

Какие потоки ввода и вывода вам известны и как их можно перенаправить?

Потоки

- stdout
- stderr
- stdin

Потоки

- > -- перенаправить вывод
- < -- перенаправить ввод
- 2> -- перенаправить ошибки
- 2>&1 -- вывод и ошибки
- >> \ << -- перенаправление с перезаписью

dev/null

The background of the slide is a high-angle, blue-tinted aerial photograph of a dense urban skyline, likely New York City. Overlaid on this image is a semi-transparent network of white lines connecting various points, creating a digital or technological aesthetic. The word "Subprocess" is centered in a large, white, sans-serif font.

Subprocess

subprocess

subprocess дает разработчику возможность запускать процессы из Python кода, получать соединение с потоками стандартного ввода, стандартного вывода, и получение кодов возврата от этих процессов.

subprocess

subprocess.call (*args, *, stdin=None, stdout=None, stderr=None, shell=False, cwd=None, timeout=None*)

- 1) Выполняет команду переданную в args
- 2) Блокирует выполнение скрипта и ждет выполнения команды
- 3) Возвращает только код возврата

subprocess

subprocess.check_call (*args, *, stdin=None, stdout=None, stderr=None, shell=False, cwd=None, timeout=None*)

- 1) Выполняет команду переданную в args
- 2) Блокирует выполнение скрипта и ждет выполнения команды
- 3) Если команда вернула не 0, бросит CalledProcessError

subprocess

subprocess.check_output (*args, *, stdin=None, stderr=None, shell=False, cwd=None, encoding=None, errors=None, universal_newlines=None, timeout=None, text=None*)

- 1) Выполняет команду переданную в args
- 2) Блокирует выполнение скрипта и ждет выполнения команды
- 3) Если команда вернула не 0, бросит CalledProcessError
- 4) Если команда вернула 0 вернет stdout

subprocess

subprocess.Popen -- класс для гибкого выполнения команд. Не блокирует выполнение скрипта.

subprocess

Popen.poll -- проверяет статус процесса запущенного Popen.

Popen.wait -- устанавливает таймаут для запущенного процесса, блокирует выполнение скрипта, если процесс не уложился, бросает TimeoutExpired.

Popen.communicate -- передает данные в stdin процесса, возвращает stdout, stderr.

Popen.send_signal -- отправляет сигнал процессу

Popen.terminate -- отправляет SIGTERM процессу

Popen.kill -- отправляет SIGKILL процессу

The image features a high-angle, blue-tinted aerial photograph of a dense urban skyline, likely New York City, with numerous skyscrapers and buildings. A semi-transparent blue band with a white network pattern of dots and lines runs horizontally across the middle of the image. The word "Sys" is centered within this band in a white, sans-serif font.

Sys

sys -- предоставляет доступ к некоторым переменным окружения используемых интерпретатором.

sys.argv -- список аргументов командной строки, переданных скрипту (первый имя скрипта).

sys.executable -- строка, полный путь до интерпретатора Python.

sys.path -- список путей по которым питон ищет модули. Изменяемая копия PYTHONPATH.

sys.platform -- информация об ос

sys.exit -- завершит выполнение с переданным кодом. Бросит SystemExit (finally блоки будут выполнены)

Управление потоками ввода/вывода:

```
sys.stdin  
sys.stdout  
sys.stderr
```


The image features a high-angle, aerial view of a dense urban skyline, likely New York City, with numerous skyscrapers and buildings. The entire image is overlaid with a semi-transparent blue filter. A prominent network pattern, consisting of white dots connected by thin white lines, is superimposed over the cityscape. The pattern is most visible in the central horizontal band, which also contains the text 'OS'.

OS

os -- модуль представляет множество функций для работы с ос, причем их поведение не зависит от ос

- **os.name** -- семейство ОС (posix, nt, java)
- **os.environ** -- словарь с переменными окружения
- **os.environb** -- байтовая версия os.environ
- **os.getcwd** -- текущая директория (откуда запущен скрипт)
- **os.getlogin** -- текущий пользователь
- **os.path** -- модуль для безопасной работы с путями
- **os.chmod** -- установить права на файл
- **os.listdir** -- список файлов в папке
- **os.mkdir** -- создать папку

Слайд с тезисами

1

Работа с процессами через Python - не всегда самое рациональное решение

2

Нужно учитывать универсальность написанных скриптов для запуска на разных системах

Одна мысль на слайде

без картинок

Использование цифр для опроса

0-9



Домашнее задание

Написать Python скрипт, который выводит:

1. Список всех процессов
2. Информацию о конкретном процессе
3. Список в файлов в директории (указать директорию)
4. Текущую директорию и список в файлов в ней
5. Версию ядра
6. Версию операционной системы

Условия:

Пользоваться только модулями subprocess, os, sys и bash внутри subprocess



Пройдите, пожалуйста, опрос в Чате с преподавателем после приёмки вашего ДЗ

Рефлексия




С какими основными мыслями и инсайтами уходите с вебинара?



Достигли ли вы цели вебинара?

Список материалов для изучения

- процессы в Linux:
http://www.linuxcenter.ru/lib/articles/system/linux_processes.phtml
- subprocess: <https://docs.python.org/3/library/subprocess.html>
- os: <https://docs.python.org/3/library/os.html>
- sys: <https://docs.python.org/3/library/sys.html>

The background of the entire image is an aerial photograph of a city with many skyscrapers, tinted in shades of blue. A semi-transparent horizontal band with a teal-to-blue gradient and a white network pattern of dots and lines runs across the middle. The text is centered within this band.

Заполните, пожалуйста,
опрос о занятии по ссылке в чате



Спасибо за внимание!
Приходите на следующие вебинары

