Лабораторная работа No 11

Программирование в командномпроцессоре ОС UNIX. Командные файлы

Кеан Путхеаро НПИбд-01-20

Содержание

# Цель работы

Изучил основы программирования в оболочке ОС UNIX/Linux. Научился писать небольшие командные файлы.

# Задание

1. Написать скрипт, который при запуске будет делать резервную копию самого се-бя (то есть файла, в котором содержится его исходный код) в другую директориюbackupв вашем домашнем каталоге. При этом файл должен архивироваться од-ним из архиваторов на выборzip,bzip2илиtar. Способ использования командархивации необходимо узнать, изучив справку.
2. Написать пример командного файла, обрабатывающего любое произвольноечисло аргументов командной строки, в том числепревышающеедесять. Напри-мер, скрипт может последовательно распечатывать значения всех переданныхаргументов.
3. Написать командный файл — аналог командыls(без использования самой этойкоманды и командыdir). Требуется, чтобы он выдавал информацию о нужномкаталоге и выводил информацию о возможностях доступа к файлам этого ката-лога.
4. Написать командный файл, который получает в качестве аргумента команднойстроки формат файла (.txt,.doc,.jpg,.pdfи т.д.) и вычисляет количествотаких файлов в указанной директории. Путь к директории также передаётся ввиде аргумента командной строки.

# Выполнение лабораторной работы

1. Написал скрипт, который при запуске будет делать резервную копию самого се-бя (то есть файла, в котором содержится его исходный код) в другую директориюbackupв вашем домашнем каталоге. При этом файл должен архивироваться од-ним из архиваторов на выборzip,bzip2илиtar. Способ использования командархивации необходимо узнать, изучив справку.

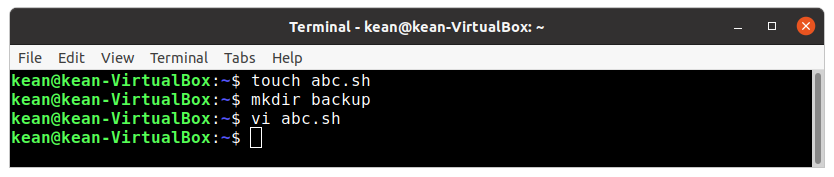


Figure 1: создание файла и каталога

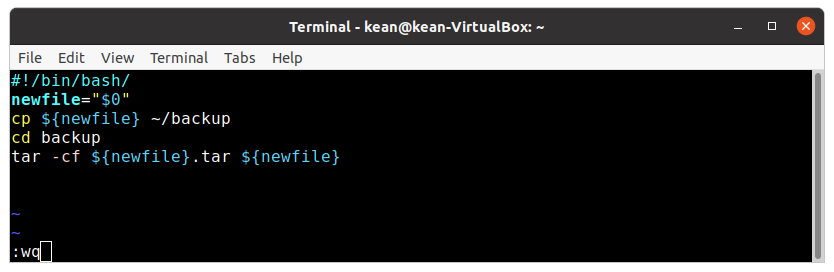


Figure 2: создание файла и каталога

Figure 3: запуск командного файла

Figure 3: запуск командного файла

Посмотрим на результат его работы - перейдем в каталог резервного копирования, в котором должен был быть создан необходимый архив. Мы видим, что он действительно содержит копию исходного файла и был создан новый архив (рисунок 4).

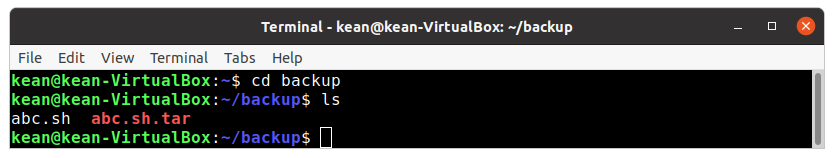


Figure 4: проверка работы командного файла

1. Написал пример командного файла, обрабатывающего любое произвольное число аргументов командной строки, в том числе превышающее десять. Например, скрипт может последовательно распечатывать значения всех переданных аргументов.

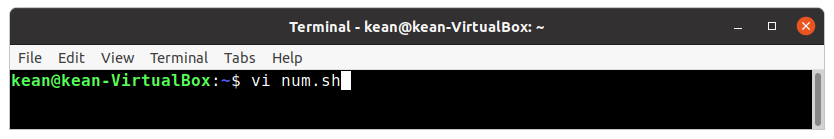


Figure 5: создание файла num.sh

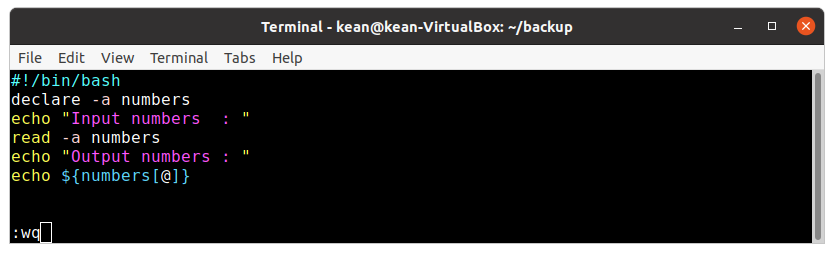


Figure 6: набор текста файла

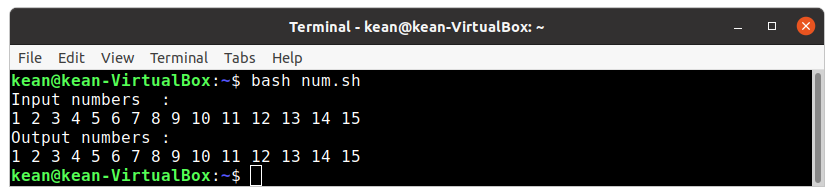


Figure 7: езультат работы командного файла

1. Написать командный файл — аналог команды ls (без использования самой этой команды и команды dir). Требуется, чтобы он выдавал информацию о нужном каталоге и выводил информацию о возможностях доступа к файлам этого каталога

Figure 8: создание файла ls.sh

Figure 8: создание файла ls.sh

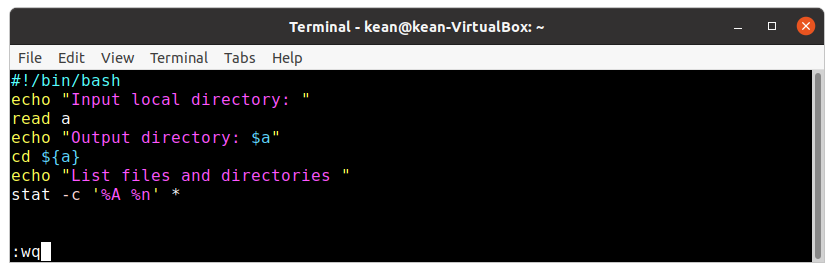


Figure 9: набор текста файла

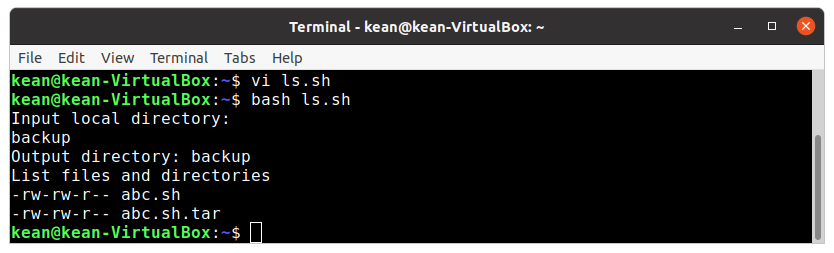


Figure 10: результат работы командного файла

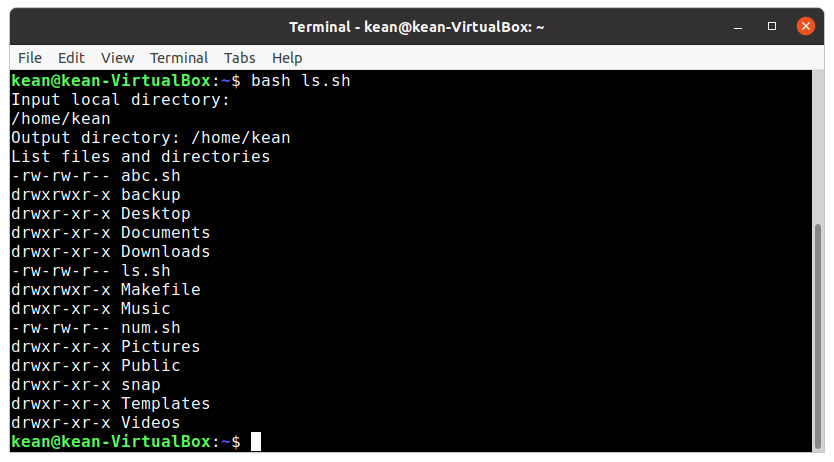


Figure 11: результат работы командного файла

1. Написать командный файл, который получает в качестве аргумента командной строки формат файла (.txt, .doc, .jpg, .pdf и т.д.) и вычисляет количество таких файлов в указанной директории. Путь к директории также передаётся в виде аргумента командной строки.

Figure 12: создание файла find.sh

Figure 12: создание файла find.sh

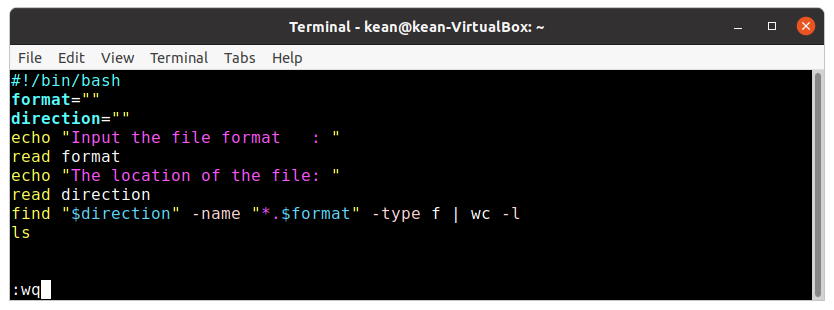


Figure 13: результат работы

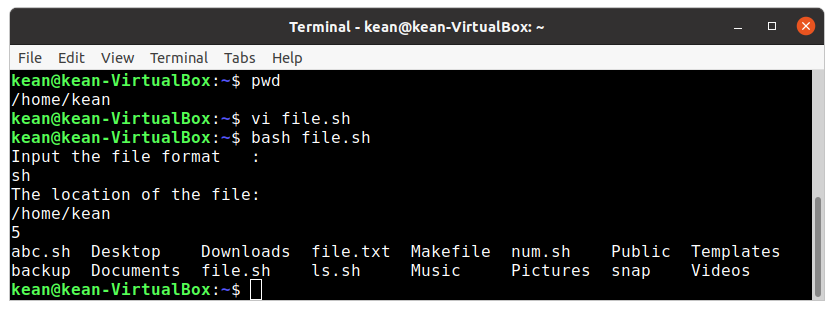


Figure 14: результат работы

# Вывод

Изучил основы программирования в оболочке ОС UNIX/Linux. Научился писать небольшие командные файлы.

# Библиография

1. (Лабораторная работа №11) https://esystem.rudn.ru/pluginfile.php/1142377/mod\_resource/content/2/008-lab\_shell\_prog\_1.pdf
2. (stackexchange) https://vi.stackexchange.com/questions/10209/execute-current-buffer-as-bash-script-from-vim