

Разбор практического задания

Задание 1

Напишите универсальный скрипт для распаковки заархивированных файлов. Скрипт должен обрабатывать архивы в форматах `.gz`, `.bz2`, `lzma`, `zip`.

Проверим, что программа `unzip` установлена. Если `dpkg -l` не выйдет с 0, расскажем об этом пользователю.

Далее проверим, что нам передано столько аргументов, сколько мы ожидаем, в противном случае выведем сообщение об ошибке пользователю.

И проверим, существует ли файл, который мы собираемся разархивировать.

Далее используем оператор выбора и просто будем сравнивать имя переданного в качестве аргумента файла с шаблоном.

Проверим наш скрипт на паре «тестовых» архивов с неоригинальными названиями.

Задание 2

Программа `rm` не даёт права на ошибку и не поддерживает механизм на случай, если вы ошиблись или передумали удалять файл.

Напишите скрипт под названием `delete`, который будет вместо удаления файла сжимать его (можно выбрать алгоритм `gzip` или `lzma`) и перемещать в папку `/home/username/TRASH`. Каждый раз при запуске скрипт должен просматривать папку `TRASH` и удалять из неё файлы старше 48 часов.

Напишем скрипт для решения задачи.

1. Проинициализируем несколько переменных. Переменная, в которой будет храниться путь до нашей корзины.
2. Создадим и для них переменные с говорящими названиями.
3. Проверим, что скрипту передан один аргумент, если это не так — выведем сообщение об ошибке.
4. Нужно убедиться, что директория корзины существует. Используем команду `mkdir`: если директории нет — она создастся, а если директория и так существует, не будет выведено никакой ошибки, никаких действий тоже не будет произведено.
5. Чтобы файлы из корзины удалялись через два дня, при помощи команды `find` будем искать старые файлы и удалять их, если такие найдутся.

И ещё две проверки: существует ли файл, который мы хотим удалить?

Не существует ли уже архива с таким же именем? Если существует, давайте не будем его затирать и просто завершим скрипт с ошибкой и сообщением пользователю.

И в самой последней строке — основная логика. Мы сжимаем при помощи `gzip` файл, сохраняем архив в корзину и, если это прошло успешно, удаляем исходный файл. В

противном случае рассказываем об ошибке пользователю.

Задание 3

Напишите скрипт, который будет принимать в качестве аргумента имя скрипта и удалять из него все комментарии.

1. Определим две переменные. Исходное имя файла и файл со скриптом без комментариев.
2. Проверим количество переданных аргументов скрипту.
3. Воспользуемся `sed`. Первую строку мы выведем «как есть», потому что первая строка скрипта — это всегда шебанг, а из вывода остального текста уберём все строки, которые начинаются с решётки.
4. Вернём пользователю имя файла, в котором сохранён отредактированный скрипт.

Проверим скрипт на другом скрипте `mvdir.sh` с комментариями.