Bash: Автодополнение — Файл ~/.bashrc. Функции для автодополнения.

Category: .bashrc,bash & sh,GNU/Linux,автодополнение
2025-02-15

Выбор места для размещения функции автодополнения зависит от предпочтений и того, хотите ли вы, чтобы функция была доступна только вам или другим пользователям системы. Простые функции могут быть размещены в ~/.bashrc, в то время как более организованные сценарии могут быть помещены в специальный каталог для автодополнения.

Файл ~/.bashrc.

Если вы создаете функцию автодополнения, которая будет использоваться только вами, самым простым решением является добавление функции непосредственно в файл ~/.bashrc.

```
$ mcedit ~/.bashrc
```

Например:

```
# Функция автодополнения
_mycommand_completion() {
# Замените на нужные аргументы
local options="option1 option2 option3"
COMPREPLY=($(compgen -W "$options" -- "${COMP_WORDS[COMP_CWORD]}"))
}
complete -F _mycommand_completion mycommand
```

После добавления этой функции не забудьте выполнить source ~/.bashrc или перезапустить оболочку, чтобы изменения вступили в силу.

```
$ source ~/.bashrc
```

Открыть разбор функции.

Разберем функцию автодополнения и компоненты, которые в ней используются, по частям.

1. Определение функции.

```
_mycommand_completion() {
```

Эта строка определяет новую функцию автодополнения с именем _mycommand_completion. Имя функции обычно начинается с символа подчеркивания, чтобы обозначить, что это функция автодополнения, связанная с конкретной командой.

2. Локальные переменные.

```
local options="option1 option2 option3"
```

Здесь определяется локальная переменная options, которая содержит строку с тремя вариантами **option1**, **option2** и **option3**. Эти варианты будут предложены пользователю для автодополнения, когда он начнет вводить команду **mycommand**.

3. Генерация вариантов автодополнения.

```
COMPREPLY=($(compgen -W "$options" -- "${COMP_WORDS[COMP_CWORD]}"))
```

- compgen — это встроенная команда **bash**, используемая для генерации списка возможных вариантов.
- -W "$options" — указывает, что compgen должен использовать значения в переменной options в качестве возможных вариантов автодополнения.
- -- "${COMP_WORDS[COMP_CWORD]}" — передает текущее слово, которое пользователь пытается дополнить.
- COMP_WORDS — это массив, содержащий все слова командной строки, а COMP_CWORD — это индекс текущего слова, на котором пользователь работает. Так, например, если пользователь вводит **mycommand op**, COMP_WORDS будет равен (**mycommand**, **op**), а COMP_CWORD будет равен **1** (**второе слово**).

Таким образом, COMPREPLY заполняется списком доступных вариантов автодополнения, которые соответствуют текущему вводу пользователя.

4. Связывание функции с командой.

```
complete -F _mycommand_completion mycommand
```

Эта строка связывает функцию автодополнения _mycommand_completion с командой **mycommand**.

Это означает, что при вводе команды **mycommand** и нажатии клавиши Tab оболочка вызовет функцию _mycommand_completion, чтобы получить

доступные варианты для автозавершения.

Итог:

Когда пользователь вводит команду **mycommand** и нажимает клавишу Tab для автодополнения:

1. Вызывается функция _mycommand_completion.
2. Она генерирует список возможных параметров (опций) на основе заранее определенной переменной options.
3. Оболочка предлагает пользователю варианты **option1**, **option2**, **option3**, которые он может выбрать для завершения команды.

Таким образом, функция автодополнения позволяет быстрее вводить команды и параметры.

Файловый каталог для автодополнения.

Если вы хотите лучше организовать свои функции автодополнения, вы можете создать отдельный файл для каждой функции и разместить их в специальном каталоге, например, ~/.bash_completion.d. Затем добавьте в файл ~/.bashrc команду для загрузки всех этих скриптов.

1. Создайте каталог:

```
$ mkdir -p ~/.bash_completion.d
```

2. Создайте файл для автодополнения, например **mycommand_completion.sh**, и добавьте в него функцию.

3. В ~/.bashrc добавьте код для источника всех файлов в этом каталоге:

```
# Автозагрузка функций из ~/.bash_completion.d
if [ -d ~/.bash_completion.d ]; then
for file in ~/.bash_completion.d/*; do
[ -f "$file" ] && . "$file"
done
fi
```

Система автодополнения bash-completion.

Для более сложных случаев вы можете использовать уже существующие системы автодополнения **bash**, некоторые из которых предоставляются отдельными пакетами или проектами. Например, вы можете использовать bash-completion, который содержит множество клиенто-ориентированных функций автодополнения.

Установка и настройка bash-completion может быть выполнена через менеджер пакетов системы (например, **apt**, **dnf**, **brew** и так далее), после чего необходимо будет убедиться, что они включены в файле ~/.bashrc.

```
$ sudo dnf -y install bash-completion
```

Пример использования bash-completion.

После установки bash-completion просто добавьте следующее в ~/.bashrc:

```
if [ -f /etc/bash_completion ]; then
. /etc/bash_completion
fi
```

Затем вы сможете создать функции автодополнения в соответствии с документацией для bash-completion.

Автодополнение пользовательских путей.

Вы также можете настроить функции для конкретных команд, которые будут предлагать определенные файлы или пути, основываясь на вводе.

Это может быть полезно, если у вас есть специфические команды или сценарии, которые требуют дополнительных аргументов.

Настройка пользовательского автодополнения.

Предположим, у вас есть скрипт my_script.sh, и вам нужно, чтобы автодополнение предлагало определенные файлы в определенного каталога.

Вот как можно настроить функцию пользовательского автодополнения:

Пример:

```
_my_script_completions() {
local cur="${COMP_WORDS[COMP_CWORD]}"
# Замените на нужный путь
local dir="/path/to/my/directory"
COMPREPLY=($(compgen -f "${dir}/${cur}"))
}
complete -F _my_script_completions my_script.sh
```

Теперь, когда вы будете набирать **my_script.sh** и потом пробел, а затем начальную часть имени файла, **bash** предложит вам варианты из указанного каталога (**/path/to/my/directory**).

Настройка автоматического завершение для пользовательских команд.

Если вы создаете свои команды или скрипты, вы можете написать функции для автодополнения.

Например, для скрипта **my_script_2.sh** вы можете создать специальные функции, которые будут предоставлять возможные аргументы (**arg1**, **arg2**, **arg3**).

Пример:

```
_my_script_completions_arg() {
local cur=${COMP_WORDS[COMP_CWORD]}
# Замените на нужные аргументы
local options="arg1 arg2 arg3"
COMPREPLY=( $(compgen -W "${options}" -- ${cur}) )
}
complete -F _my_script_completions_arg my_script_arg.sh
```

Теперь, когда вы будете набирать **my_script_2.sh** и потом пробел, а затем нажмете клавишу Tab, bash предложит вам варианты из заранее заготовленного списка возможных аргументов (**arg1**, **arg2**, **arg3**).

Оригиналы источников информации.

1. Нет.