

# Домашнее задание № 1

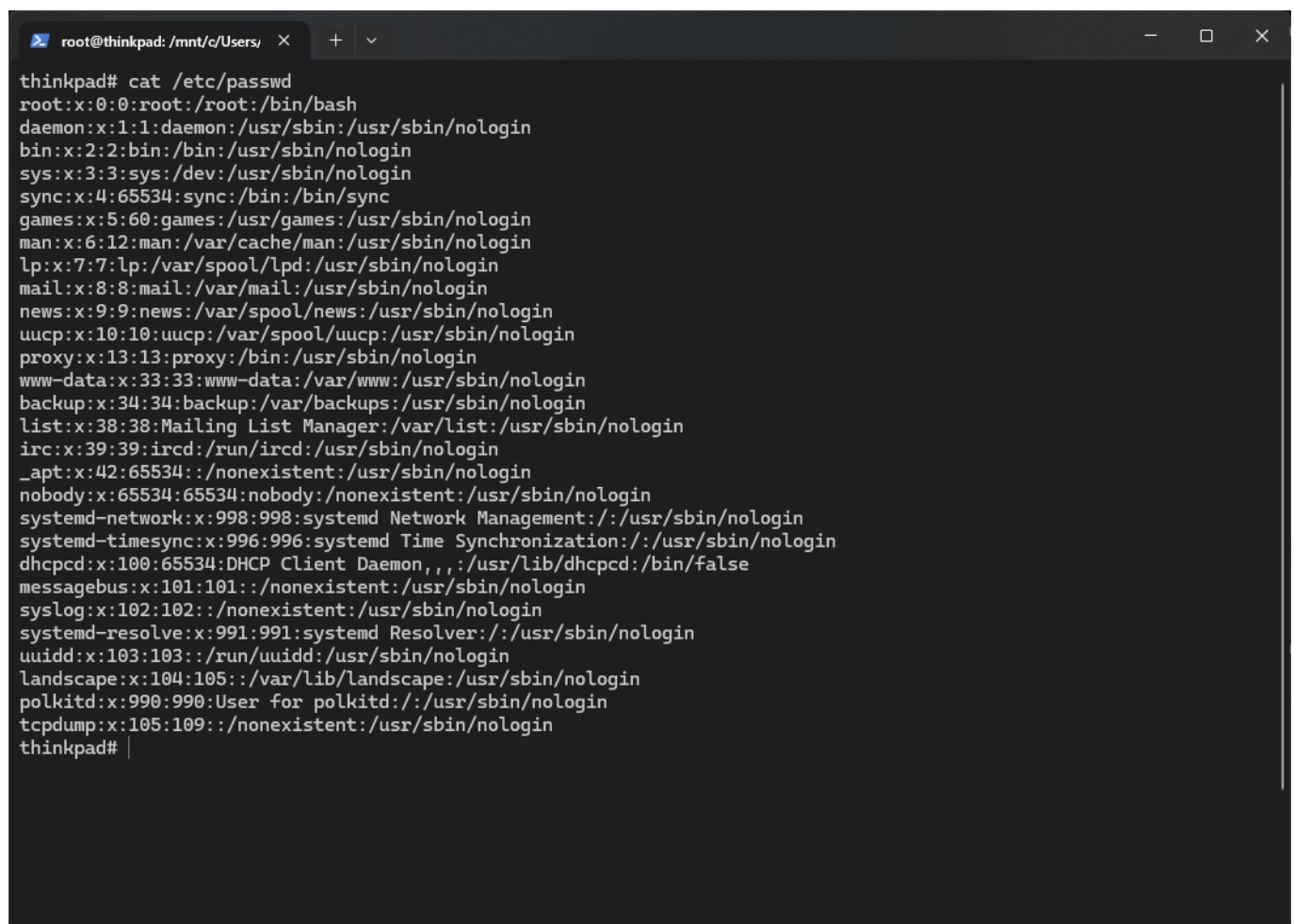
---

## Задание № 1

1. В каком файле хранится основная информация о пользователях? С помощью какой команды он может быть просмотрен?

Основная информация о пользователях на большинстве систем Linux хранится в файле `/etc/passwd`.

Для просмотра содержимого этого файла можно использовать команду:



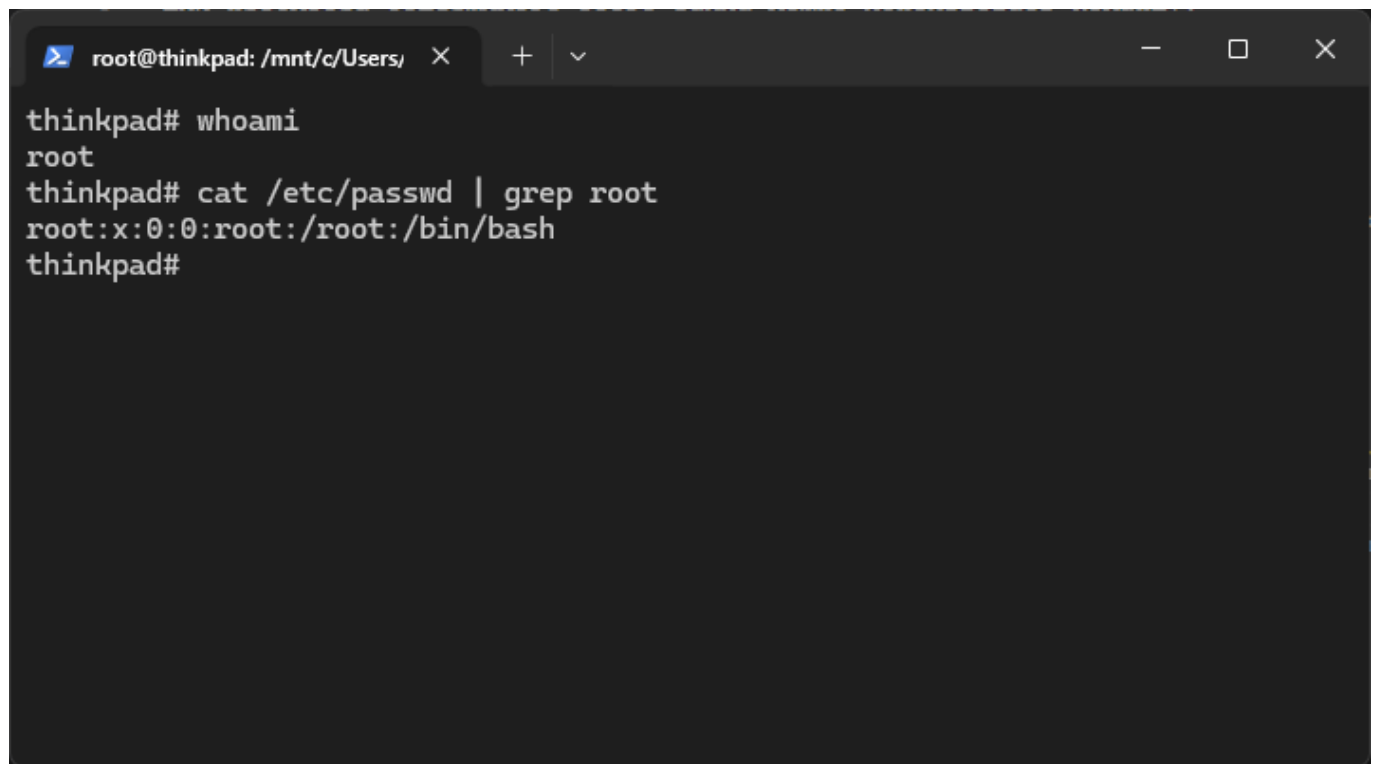
```
root@thinkpad: /mnt/c/Users/ x + v
thinkpad# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nologin
systemd-timesync:x:996:996:systemd Time Synchronization:/:/usr/sbin/nologin
dhcpcd:x:100:65534:DHCP Client Daemon,,,:/usr/lib/dhcpcd:/bin/false
messagebus:x:101:101::/nonexistent:/usr/sbin/nologin
syslog:x:102:102::/nonexistent:/usr/sbin/nologin
systemd-resolve:x:991:991:systemd Resolver:/:/usr/sbin/nologin
uuid:x:103:103::/run/uuid:/usr/sbin/nologin
landscape:x:104:105::/var/lib/landscape:/usr/sbin/nologin
polkitd:x:990:990:User for polkitd:/:/usr/sbin/nologin
tcpdump:x:105:109::/nonexistent:/usr/sbin/nologin
thinkpad#
```

Также можно воспользоваться командой `less` для удобного просмотра содержимого:

```
less /etc/passwd
```

2. Подробно описать информацию о текущем пользователе, которая хранится в файле, указанном в ответе на предыдущий вопрос.

В данном случае (WSL), мой пользователь - root с UID 0, он принадлежит группе root (GID 0), его домашний каталог - `/root`, а командная оболочка - Bash. Сам пароль хранится в `/etc/shadow`.

A terminal window with a dark background. The title bar shows 'root@thinkpad: /mnt/c/Users/' and window control buttons. The terminal text is as follows:

```
thinkpad# whoami
root
thinkpad# cat /etc/passwd | grep root
root:x:0:0:root:/root:/bin/bash
thinkpad#
```

3. Где и в каком виде хранятся пароли пользователей? С помощью какой команды они могут быть просмотрены?

Пароли пользователей в системах Linux хранятся в зашифрованном виде в файле `/etc/shadow`.  
Содержимое файла `/etc/shadow` можно просмотреть с помощью команды:

```
root@thinkpad: /mnt/c/Users/ X + v - □ X
thinkpad# cat /etc/passwd | grep root
root:x:0:0:root:/root:/bin/bash
thinkpad# cat /etc/shadow
root:*:19993:0:99999:7:::
daemon:*:19993:0:99999:7:::
bin:*:19993:0:99999:7:::
sys:*:19993:0:99999:7:::
sync:*:19993:0:99999:7:::
games:*:19993:0:99999:7:::
man:*:19993:0:99999:7:::
lp:*:19993:0:99999:7:::
mail:*:19993:0:99999:7:::
news:*:19993:0:99999:7:::
uucp:*:19993:0:99999:7:::
proxy:*:19993:0:99999:7:::
www-data:*:19993:0:99999:7:::
backup:*:19993:0:99999:7:::
list:*:19993:0:99999:7:::
irc:*:19993:0:99999:7:::
_apt:*:19993:0:99999:7:::
nobody:*:19993:0:99999:7:::
systemd-network:!:19993:!:!:!:
systemd-timesync:!:19993:!:!:!:
dhcpcd:!:19993:!:!:!:
messagebus:!:19993:!:!:!:
syslog:!:19993:!:!:!:
systemd-resolve:!:19993:!:!:!:
uuidd:!:19993:!:!:!:
landscape:!:19993:!:!:!:
polkitd:!:19993:!:!:!:
tcpdump:!:20131:!:!:!:
thinkpad#
```

\* В данном случае вместо хэша отображен символ "\*", который говорит о том, что вход по паролю запрещен, но пользователь может аутентифицироваться другими методами (например, через SSH-ключ)

\*\* Отображение паролей в чистом виде запрещено в современных системах из соображений безопасности

4. В каком файле хранится информация о группах? С помощью какой команды он может быть просмотрен?

Информация о группах пользователей хранится в файле `/etc/group`.

Для просмотра содержимого этого файла можно использовать команду:

```
root@thinkpad: /mnt/c/Users/ X + - □ X
thinkpad# cat /etc/group
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:syslog
tty:x:5:
disk:x:6:
lp:x:7:
mail:x:8:
news:x:9:
uucp:x:10:
man:x:12:
proxy:x:13:
kmem:x:15:
dialout:x:20:
fax:x:21:
voice:x:22:
cdrom:x:24:
floppy:x:25:
tape:x:26:
sudo:x:27:
audio:x:29:
dip:x:30:
www-data:x:33:
backup:x:34:
operator:x:37:
list:x:38:
irc:x:39:
src:x:40:
shadow:x:42:
utmp:x:43:
```

Также можно воспользоваться командой `less` для удобного просмотра:

```
less /etc/group
```

5. Информация из каких файлов позволяет узнать полный перечень групп, в которых состоит пользователь? На примере текущего пользователя и реальных системных файлов провести данный анализ

Для получения полного перечня групп, в которых состоит пользователь, используются следующие файлы:

1. `/etc/group` — основной файл, содержащий информацию о группах и их членах
2. `/etc/gshadow` — содержит зашифрованные пароли для групповых учетных записей и их дополнительные параметры
3. `etc/passwd` — этот файл хранит информацию о пользователях, но он также может содержать информацию о группах, к которым относится пользователь, через их идентификатор группы

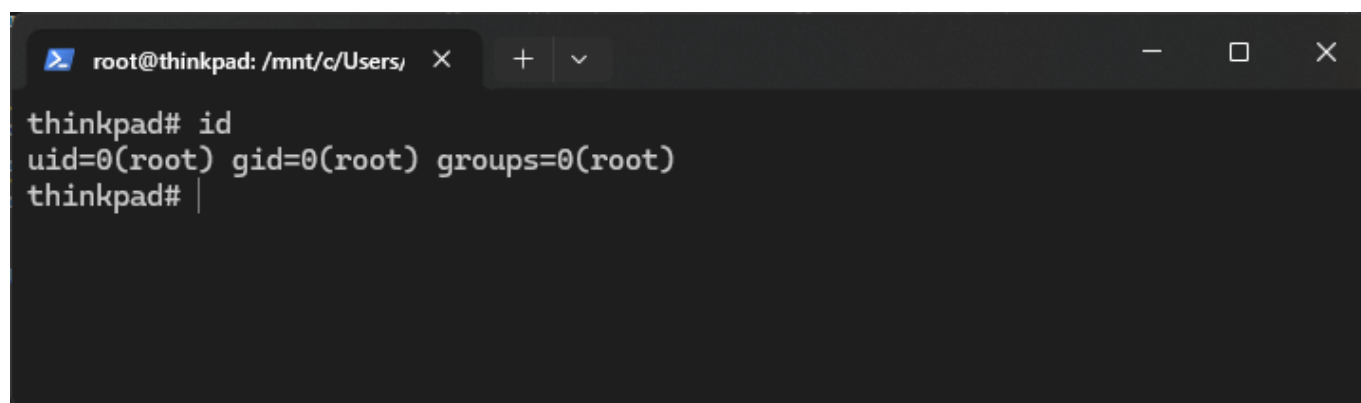
(GID).

Скриншоты представлены выше

6. Привести перечень команд (желательно, не менее 2 шт.), которые позволяют проверить информацию о текущем пользователе, которая была получена в результате поиска ответа на предыдущий вопрос.

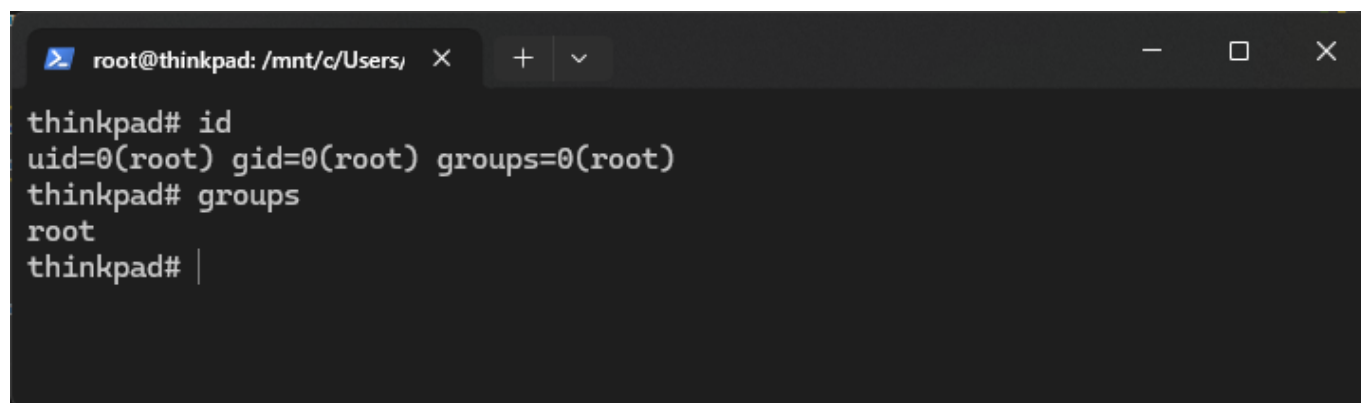
Для проверки информации о текущем пользователе, включая группы, к которым он принадлежит, можно использовать следующие команды:

1. Команда `id` - Эта команда выводит идентификатор пользователя, группы и группы, в которых он состоит.



```
root@thinkpad: /mnt/c/Users/ X + v - □ X
thinkpad# id
uid=0(root) gid=0(root) groups=0(root)
thinkpad#
```

2. Команда `groups`- Эта команда отображает группы, к которым принадлежит текущий пользователь.



```
root@thinkpad: /mnt/c/Users/ X + v - □ X
thinkpad# id
uid=0(root) gid=0(root) groups=0(root)
thinkpad# groups
root
thinkpad#
```

## Задание № 2

1. Что такое переменная оболочки (локальная переменная)? Какими свойствами она обладает? Привести пример команды, создающей переменную оболочки, и пример команды, выводящей её значение

Переменная оболочки (локальная переменная) — это переменная, которая доступна только в пределах текущей сессии или конкретной оболочки (сессии терминала). Такие переменные сохраняются только временно и исчезают после завершения сессии.

**Свойства переменной оболочки:**

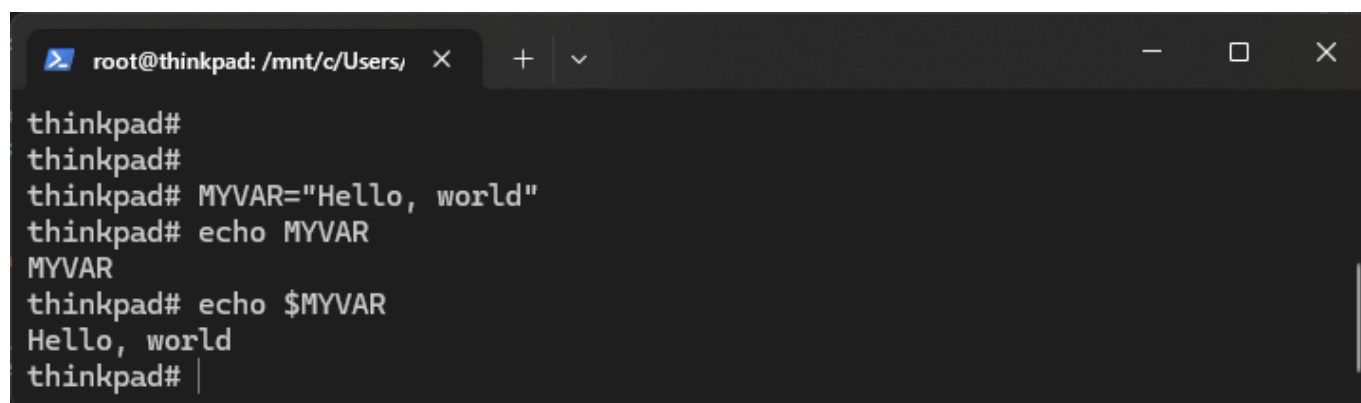
- Только временное использование — значение переменной сохраняется только в текущей сессии оболочки.
- Изменяемость — переменные можно изменять и удалять вручную или через команды.
- Область действия — доступ ограничен текущей сессией или определенным процессом.

Пример команды, создающей переменную оболочки:

```
MYVAR="Hello, world!"
```

Пример команды, выводящей значение переменной:

```
echo $MYVAR
```



```
root@thinkpad: /mnt/c/Users/ X + v
thinkpad#
thinkpad#
thinkpad# MYVAR="Hello, world"
thinkpad# echo MYVAR
MYVAR
thinkpad# echo $MYVAR
Hello, world
thinkpad# |
```

## 2. Что такое переменная окружения (глобальная переменная)?

Переменная окружения (глобальная переменная) — это переменная, которая доступна для всех процессов, запущенных в системе, включая оболочку и дочерние процессы. Такие переменные обычно используются для хранения информации, которая должна быть доступна системным утилитам и другим программам.

### Свойства переменной окружения:

- Глобальное использование — доступна всем процессам и приложениям в системе.
- Изменяемость — значения можно изменять с помощью команд.
- Взаимодействие с другими программами — позволяет программам обмениваться данными между собой.

Пример команды, создающей переменную окружения:

```
export MYENV="Значение переменной окружения"
```

Пример команды, выводящей значение переменной окружения:

```
echo $MYENV
```

```
root@thinkpad: /mnt/c/Users/  X + v - □ X
thinkpad# echo MYVAR
MYVAR
thinkpad# echo $MYVAR
Hello, world
thinkpad# export MYENV="Значение переменной окружения"
thinkpad# echo $MYENV
Значение переменной окружения
thinkpad# |
```

3. На примере двух переменных, созданных в процессе подготовки ответов на предыдущие вопросы, показать на практике основное различие между локальной и глобальной переменными

Различие между глобальной и локальной переменными можно продемонстрировать, создав дочерний bash-процесс и попробовав вывести в нем значения данных переменных. Глобальная переменная будет так же доступна в дочернем процессе, в то время как локальная переменная - нет, так как она доступна только в рамках текущей сессии терминала.

```
root@thinkpad: /mnt/c/Users/  X + v - □ X
thinkpad#
thinkpad#
thinkpad# MYVAR="Hello, world"
thinkpad# echo MYVAR
MYVAR
thinkpad# echo $MYVAR
Hello, world
thinkpad# export MYENV="Значение переменной окружения"
thinkpad# echo $MYENV
Значение переменной окружения
thinkpad# bash
root@thinkpad:/mnt/c/Users/lmz# echo $MYENV
Значение переменной окружения
root@thinkpad:/mnt/c/Users/lmz# echo $MYVAR

root@thinkpad:/mnt/c/Users/lmz# |
```

4. Каким образом возможно задать значение переменной, которое будет устанавливаться автоматически при входе в систему для текущего пользователя?

Чтобы задать значение переменной, которое будет автоматически устанавливаться при входе в систему для текущего пользователя, необходимо добавить её в файл конфигурации оболочки пользователя.

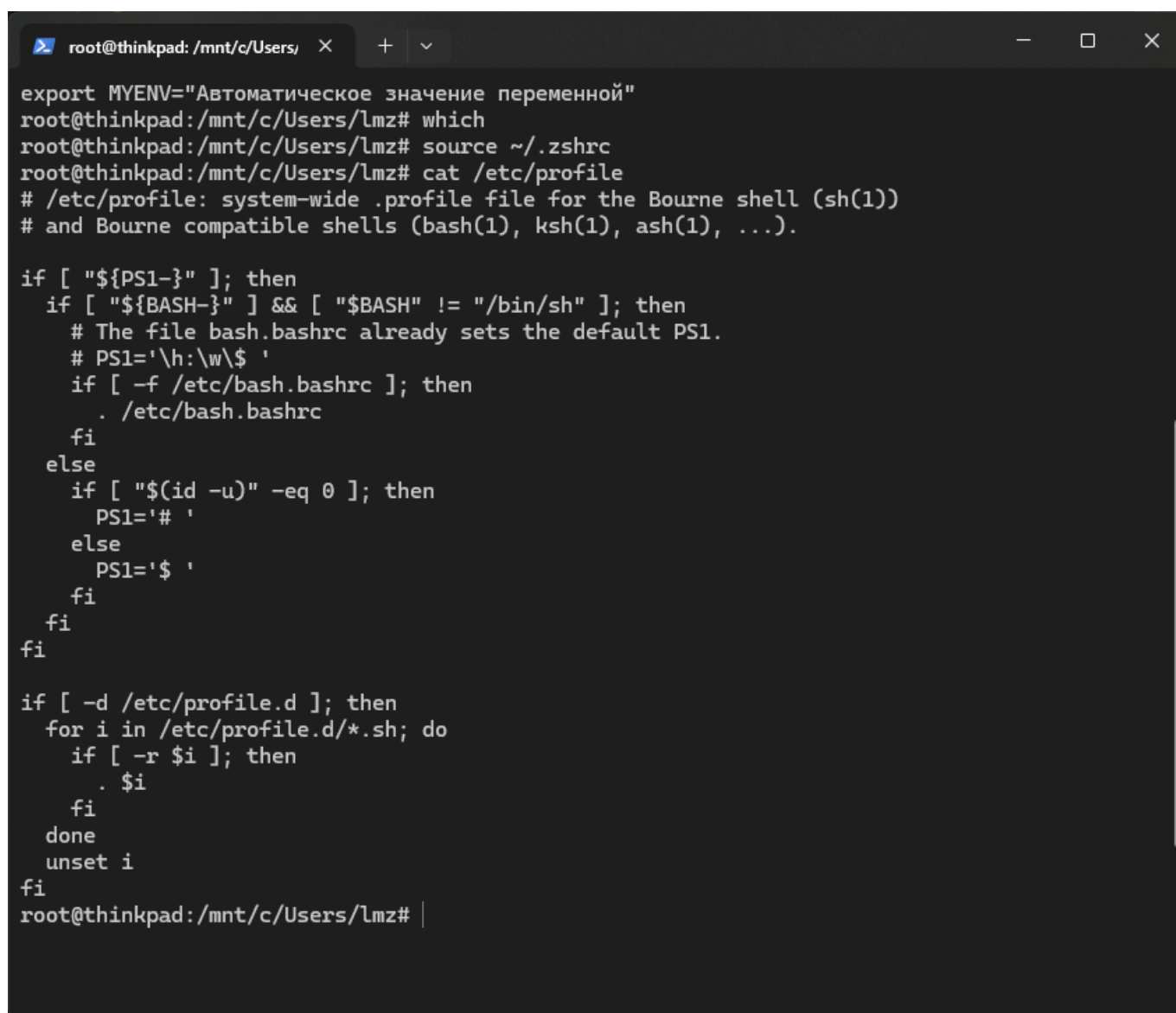
Файлы конфигурации оболочки:

`~/.bashrc` (или `~/.zshrc`) - выполняется для интерактивных оболочек при входе в терминал (чаще всего используется).

`~/.bash_profile` или (`~/.profile`, `~/.zsh_profile`): Выполняется только при входе в систему (например, через SSH или консоль).

Чтобы добавить переменную в конфигурационный файл Bash одной строкой, можно использовать команду `echo` с перенаправлением в файл. Например:

```
echo 'export MYENV="Автоматическое значение переменной"' >> ~/.zshrc
```



```
root@thinkpad: /mnt/c/Users/lmz # export MYENV="Автоматическое значение переменной"
root@thinkpad: /mnt/c/Users/lmz # which
root@thinkpad: /mnt/c/Users/lmz # source ~/.zshrc
root@thinkpad: /mnt/c/Users/lmz # cat /etc/profile
# /etc/profile: system-wide .profile file for the Bourne shell (sh(1))
# and Bourne compatible shells (bash(1), ksh(1), ash(1), ...).

if [ "${PS1-}" ]; then
  if [ "${BASH-}" ] && [ "$BASH" != "/bin/sh" ]; then
    # The file bash.bashrc already sets the default PS1.
    # PS1='\h:\w\$ '
    if [ -f /etc/bash.bashrc ]; then
      . /etc/bash.bashrc
    fi
  else
    if [ "$(id -u)" -eq 0 ]; then
      PS1='# '
    else
      PS1='$ '
    fi
  fi
fi

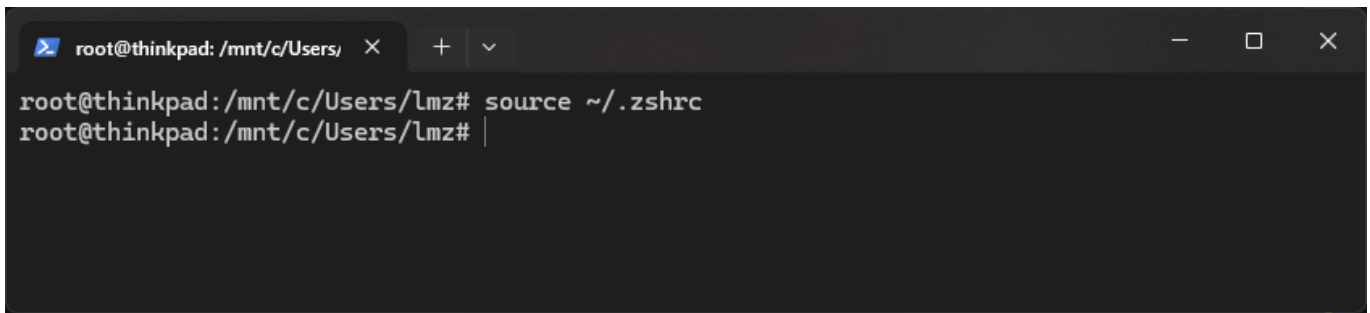
if [ -d /etc/profile.d ]; then
  for i in /etc/profile.d/*.sh; do
    if [ -r $i ]; then
      . $i
    fi
  done
  unset i
fi
root@thinkpad: /mnt/c/Users/lmz #
```

### Применение изменений:

Чтобы изменения сразу вступили в силу, необходимо выполнить команду:

```
source ~/.zshrc
```



A terminal window with a dark background. The title bar shows 'root@thinkpad: /mnt/c/Users/...' and window control buttons. The terminal text shows the prompt 'root@thinkpad: /mnt/c/Users/lmz#' followed by the command 'source ~/.zshrc' and a new prompt line.

```
root@thinkpad: /mnt/c/Users/lmz# source ~/.zshrc
root@thinkpad: /mnt/c/Users/lmz#
```

Теперь переменная MYENV будет автоматически устанавливаться при входе в систему.

5. Каким образом возможно задать значение переменной, которое будет устанавливаться автоматически при входе в систему для всех пользователей?

Чтобы задать значение переменной, которая будет устанавливаться автоматически при входе в систему для всех пользователей, её нужно прописать в системных конфигурационных файлах, которые используются для настройки окружения.

#### **Системные файлы для задания переменных:**

1. **/etc/profile** - Этот файл выполняется при входе в систему всеми пользователями.

```

root@thinkpad: /mnt/c/Users/ \ X + v
export MYENV="Автоматическое значение переменной"
root@thinkpad:/mnt/c/Users/lmz# which
root@thinkpad:/mnt/c/Users/lmz# source ~/.zshrc
root@thinkpad:/mnt/c/Users/lmz# cat /etc/profile
# /etc/profile: system-wide .profile file for the Bourne shell (sh(1))
# and Bourne compatible shells (bash(1), ksh(1), ash(1), ...).

if [ "${PS1-}" ]; then
  if [ "${BASH-}" ] && [ "$BASH" != "/bin/sh" ]; then
    # The file bash.bashrc already sets the default PS1.
    # PS1='\h:\w\$ '
    if [ -f /etc/bash.bashrc ]; then
      . /etc/bash.bashrc
    fi
  else
    if [ "$(id -u)" -eq 0 ]; then
      PS1='# '
    else
      PS1='$ '
    fi
  fi
fi

if [ -d /etc/profile.d ]; then
  for i in /etc/profile.d/*.sh; do
    if [ -r $i ]; then
      . $i
    fi
  done
  unset i
fi
root@thinkpad:/mnt/c/Users/lmz#

```

2. **Файлы в директории /etc/profile.d/** - Можно создать отдельный скрипт в этой директории для установки переменной.

```

root@thinkpad: /mnt/c/Users/ \ X + v
Z97-byobu.sh      Z99-cloudinit-warnings.sh  bash_completion.sh  gawk.sh
root@thinkpad:/mnt/c/Users/lmz# ls /etc/profile.d --la
ls: unrecognized option '--la'
Try 'ls --help' for more information.
root@thinkpad:/mnt/c/Users/lmz# ls /etc/profile.d -la
total 44
drwxr-xr-x  2 root root 4096 Dec 22 00:39 .
drwxr-xr-x 93 root root 4096 Feb 12 22:37 ..
-rw-r--r--  1 root root  96 Apr 22  2024 01-locale-fix.sh
-rw-r--r--  1 root root 1557 Feb 10  2024 Z97-byobu.sh
-rwxr-xr-x  1 root root 3396 Sep  5 22:30 Z99-cloud-locale-test.sh
-rwxr-xr-x  1 root root  841 Sep  5 22:30 Z99-cloudinit-warnings.sh
-rw-r--r--  1 root root  835 Aug 21 02:39 apps-bin-path.sh
-rw-r--r--  1 root root  726 Sep 18  2023 bash_completion.sh
-rw-r--r--  1 root root 1107 Mar 31  2024 gawk.csh
-rw-r--r--  1 root root  757 Mar 31  2024 gawk.sh
-rwxr-xr-x  1 root root  898 Nov 28  2023 update-motd.sh
root@thinkpad:/mnt/c/Users/lmz#

```

3. **/etc/environment** - Используется для задания переменных окружения, но в отличие от других файлов, он не поддерживает сложные команды (только формат **VARIABLE=value**).

```
root@thinkpad: /mnt/c/Users/ \ + v
Try 'ls --help' for more information.
root@thinkpad:/mnt/c/Users/lmz# ls /etc/profile.d -la
total 44
drwxr-xr-x  2 root root 4096 Dec 22 00:39 .
drwxr-xr-x 93 root root 4096 Feb 12 22:37 ..
-rw-r--r--  1 root root  96 Apr 22 2024 01-locale-fix.sh
-rw-r--r--  1 root root 1557 Feb 10 2024 Z97-byobu.sh
-rwxr-xr-x  1 root root 3396 Sep  5 22:30 Z99-cloud-locale-test.sh
-rwxr-xr-x  1 root root  841 Sep  5 22:30 Z99-cloudinit-warnings.sh
-rw-r--r--  1 root root  835 Aug 21 02:39 apps-bin-path.sh
-rw-r--r--  1 root root  726 Sep 18 2023 bash_completion.sh
-rw-r--r--  1 root root 1107 Mar 31 2024 gawk.csh
-rw-r--r--  1 root root  757 Mar 31 2024 gawk.sh
-rwxr-xr-x  1 root root  898 Nov 28 2023 update-motd.sh
root@thinkpad:/mnt/c/Users/lmz# cat /etc/environment
PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/g
ames:/snap/bin"
root@thinkpad:/mnt/c/Users/lmz#
```

6. Что такое переменная среды окружения PATH? Вывести значение переменной среды окружения PATH

Переменная среды окружения **PATH** содержит в себе строку со списком директорий (разделенных символом "📁", в которых по-умолчанию будет осуществляться поиск исполняемого файла команды.

```
root@thinkpad: /mnt/c/Users/ \ + v
-rw-r--r--  1 root root 1107 Mar 31 2024 gawk.csh
-rw-r--r--  1 root root  757 Mar 31 2024 gawk.sh
-rwxr-xr-x  1 root root  898 Nov 28 2023 update-motd.sh
root@thinkpad:/mnt/c/Users/lmz# cat /etc/environment
PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/g
ames:/snap/bin"
root@thinkpad:/mnt/c/Users/lmz# echo PATH
PATH
root@thinkpad:/mnt/c/Users/lmz# echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/
usr/lib/wsl/lib:/mnt/c/Python312/Scripts:/mnt/c/Python312:/mnt/c/Windows/System32:/mnt/c
/Windows:/mnt/c/Windows/System32/wbem:/mnt/c/Windows/System32/WindowsPowerShell/v1.0:/mnt
/c/Windows/System32/OpenSSH:/mnt/c/Program Files/Vagrant/bin:/mnt/c/Program Files/dotnet/
:/mnt/c/Program Files (x86)/GnuPG/bin:/mnt/c/Program Files/nodejs:/mnt/c/ProgramData/choc
olatey/bin:/mnt/c/Program Files/Git/cmd:/mnt/c/Users/lmz/AppData/Local/Programs/Python/Lau
ncher:/mnt/c/Users/lmz/AppData/Local/Microsoft/WindowsApps:/mnt/c/Users/Алексей/AppData/L
ocal/Programs/Microsoft VS Code/bin:/mnt/c/Users/lmz/AppData/Roaming/npm:/snap/bin
root@thinkpad:/mnt/c/Users/lmz#
```

7. Указать команду, которая позволяет добавить к значению переменной среды окружения PATH абсолютный путь домашнего каталога текущего пользователя

```
export PATH=$PATH:$HOME
```

```
root@thinkpad: /mnt/c/Users/ \ X + v
root@thinkpad: /mnt/c/Users/lmz# export PATH=$PATH:$HOME
root@thinkpad: /mnt/c/Users/lmz# PATH
PATH: command not found
root@thinkpad: /mnt/c/Users/lmz# echo PATH
PATH
root@thinkpad: /mnt/c/Users/lmz# echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/usr/lib/wsl/lib:/mnt/c/Python312/Scripts:/mnt/c/Python312:/mnt/c/Windows/System32:/mnt/c/Windows:/mnt/c/Windows/System32/wbem:/mnt/c/Windows/System32/WindowsPowerShell/v1.0:/mnt/c/Windows/System32/OpenSSH:/mnt/c/Program Files/Vagrant/bin:/mnt/c/Program Files/dotnet:/mnt/c/Program Files (x86)/GnuPG/bin:/mnt/c/Program Files/nodejs:/mnt/c/ProgramData/chocolatey/bin:/mnt/c/Program Files/Git/cmd:/mnt/c/Users/lmz/AppData/Local/Programs/Python/Launcher/:/mnt/c/Users/lmz/AppData/Local/Microsoft/WindowsApps:/mnt/c/Users/Алексей/AppData/Local/Programs/Microsoft VS Code/bin:/mnt/c/Users/lmz/AppData/Roaming/npm:/snap/bin:/root
```

## Задание № 3

1. Чем отличаются команды, встроенные в командный интерпретатор, и внешние? С

**Встроенные команды** выполняются внутри самого командного интерпретатора (оболочки, например Bash). Они не требуют запуска отдельного процесса, что делает их выполнение быстрее.

**Внешние команды** - это отдельные исполняемые файлы, расположенные в директориях, указанных в переменной окружения **PATH** (например, **/bin**, **/usr/bin**). Выполнение внешней команды запускает новый процесс.

---

С помощью каких команд возможно определить, является команда внешней или внутренней?

1. Команда **type** - сообщает, встроена ли команда в оболочку или является внешней.
2. Команда **which**: Показывает путь к исполняемому файлу внешней команды. Если команда встроенная, **which** не даст результата
3. Команда **command -v**

```

root@thinkpad: /mnt/c/Users/ X + v
root@thinkpad:/mnt/c/Users/lmz# type dir
dir is /usr/bin/dir
root@thinkpad:/mnt/c/Users/lmz# which dir
/usr/bin/dir
root@thinkpad:/mnt/c/Users/lmz# command -v dir
/usr/bin/dir
root@thinkpad:/mnt/c/Users/lmz# type cd
cd is a shell builtin
root@thinkpad:/mnt/c/Users/lmz# which cd
root@thinkpad:/mnt/c/Users/lmz# command -v dir
/usr/bin/dir
root@thinkpad:/mnt/c/Users/lmz# command -v cd
cd
root@thinkpad:/mnt/c/Users/lmz# |

```

2. Чем отличаются команды `man` и `help`, предоставляющие справочную информацию? Привести примеры их использования с учётом особенностей получения справочной информации для встроенных и внешних команд

**Команда `help`** предоставляет справку только для встроенных команд оболочки. Формат справки обычно менее подробный.

```

root@thinkpad: /mnt/c/Users/ X + v
root@thinkpad:/mnt/c/Users/lmz# man curl
root@thinkpad:/mnt/c/Users/lmz# help curl
bash: help: no help topics match 'curl'. Try 'help help' or 'man -k curl' or 'info curl'.
root@thinkpad:/mnt/c/Users/lmz# help cd
cd: cd [-L|[-P [-e]] [-@]] [dir]
    Change the shell working directory.

    Change the current directory to DIR. The default DIR is the value of the
    HOME shell variable. If DIR is "-", it is converted to $OLDPWD.

    The variable CDPATH defines the search path for the directory containing
    DIR. Alternative directory names in CDPATH are separated by a colon (:).
    A null directory name is the same as the current directory. If DIR begins
    with a slash (/), then CDPATH is not used.

    If the directory is not found, and the shell option 'cdable_vars' is set,
    the word is assumed to be a variable name. If that variable has a value,
    its value is used for DIR.

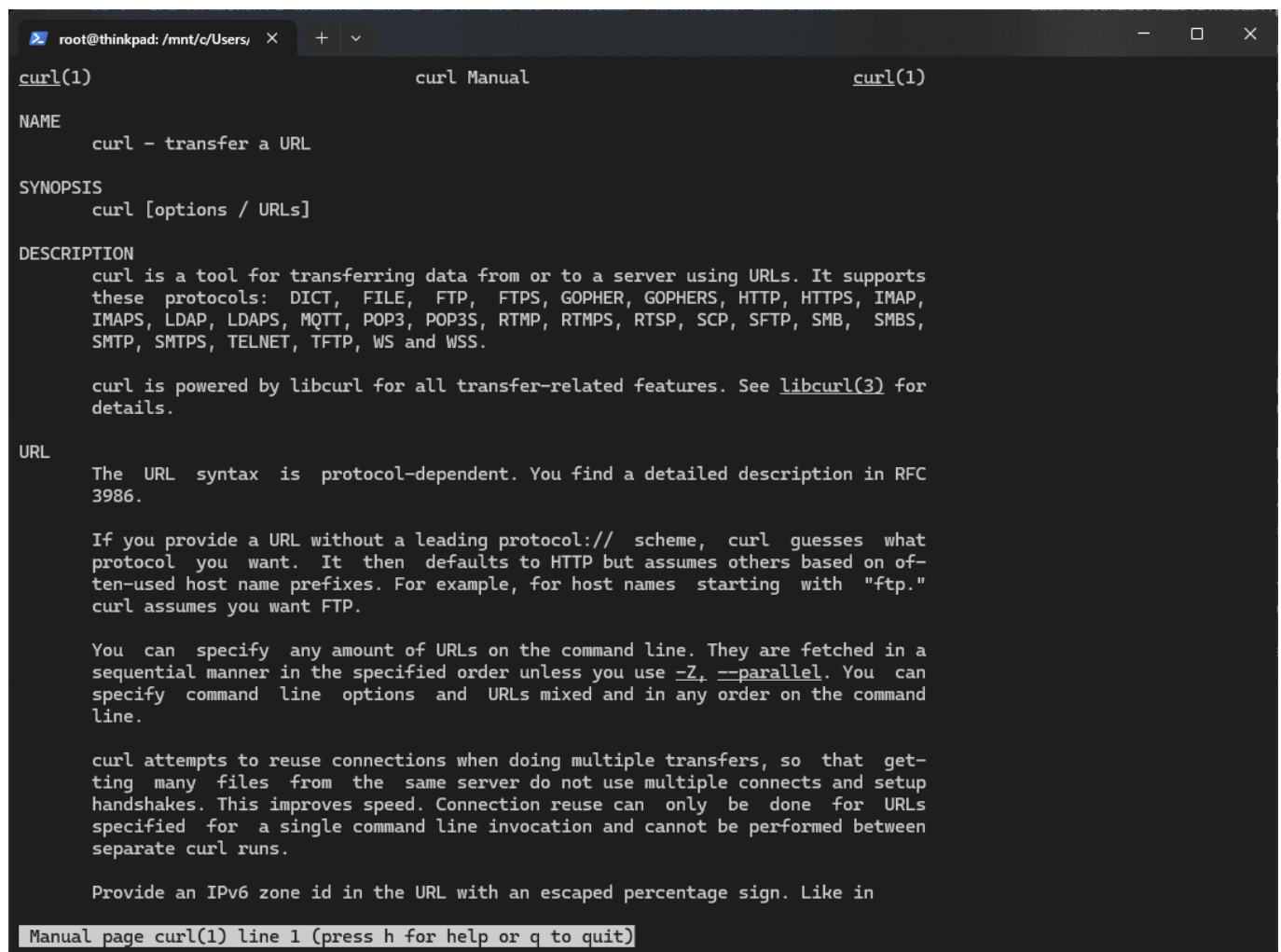
Options:
  -L      force symbolic links to be followed: resolve symbolic
          links in DIR after processing instances of '..'
  -P      use the physical directory structure without following
          symbolic links: resolve symbolic links in DIR before
          processing instances of '..'
  -e      if the -P option is supplied, and the current working
          directory cannot be determined successfully, exit with
          a non-zero status
  -@      on systems that support it, present a file with extended
          attributes as a directory containing the file attributes

The default is to follow symbolic links, as if '-L' were specified.
'..' is processed by removing the immediately previous pathname component
back to a slash or the beginning of DIR.

Exit Status:
Returns 0 if the directory is changed, and if $PWD is set successfully when
-P is used; non-zero otherwise.
root@thinkpad:/mnt/c/Users/lmz# man cd
No manual entry for cd

```

**Команда `man`**, напротив - предоставляет подробные справочные страницы (man-pages) для внешних команд, конфигурационных файлов, системных вызовов, библиотек и т.д. Формат справки более полный и структурированный.



```
root@thinkpad: /mnt/c/Users/  curl(1)  curl Manual  curl(1)

NAME
    curl - transfer a URL

SYNOPSIS
    curl [options / URLs]

DESCRIPTION
    curl is a tool for transferring data from or to a server using URLs. It supports
    these protocols: DICT, FILE, FTP, FTPS, GOPHER, GOPHERS, HTTP, HTTPS, IMAP,
    IMAPS, LDAP, LDAPS, MQTT, POP3, POP3S, RTMP, RTMPS, RTSP, SCP, SFTP, SMB, SMBS,
    SMTP, SMTPS, TELNET, TFTP, WS and WSS.

    curl is powered by libcurl for all transfer-related features. See libcurl\(3\) for
    details.

URL
    The URL syntax is protocol-dependent. You find a detailed description in RFC
    3986.

    If you provide a URL without a leading protocol:// scheme, curl guesses what
    protocol you want. It then defaults to HTTP but assumes others based on of-
    ten-used host name prefixes. For example, for host names starting with "ftp."
    curl assumes you want FTP.

    You can specify any amount of URLs on the command line. They are fetched in a
    sequential manner in the specified order unless you use -Z, --parallel. You can
    specify command line options and URLs mixed and in any order on the command
    line.

    curl attempts to reuse connections when doing multiple transfers, so that get-
    ting many files from the same server do not use multiple connects and setup
    handshakes. This improves speed. Connection reuse can only be done for URLs
    specified for a single command line invocation and cannot be performed between
    separate curl runs.

    Provide an IPv6 zone id in the URL with an escaped percent sign. Like in

Manual page curl(1) line 1 (press h for help or q to quit)
```

3. С помощью какой команды можно делать псевдонимы для существующих команд?  
Привести примеры использования данной команды

Для создания псевдонима используется команда `alias`.

```
alias ll='ls -alF'
```



```

root@thinkpad: /mnt/c/Users/ \ x + v
root@thinkpad:/mnt/c/Users/lmz# alias ll='ls -aF'
root@thinkpad:/mnt/c/Users/lmz# ll
total 4884
drwxrwxrwx 1 root root 4096 Feb 12 22:21 ./
dr-xr-xr-x 1 root root 4096 Feb 10 22:30 ../
drwxrwxrwx 1 root root 4096 Feb 12 22:15 .gitconfig*
-rwxrwxrwx 1 root root 185 Feb 12 22:21 .vimrc /
drwxrwxrwx 1 root root 4096 Oct 26 16:26 .vscode /
-rwxrwxrwx 1 root root 356 Oct 26 16:19 .packettracer*
drwxrwxrwx 1 root root 4096 Feb 11 21:39 .ssh /
drwxrwxrwx 1 root root 4096 Nov 21 01:12 .vscode /
drwxrwxrwx 1 root root 4096 Feb 10 22:54 .vscode /
drwxrwxrwx 1 root root 4096 Oct 24 15:46 .vscode /
drwxrwxrwx 1 root root 4096 Feb 10 22:30 .vscode /
lrwxrwxrwx 1 root root 32 Feb 10 22:30 'Application Data' -> /mnt/c/Users/lmz/AppData/Roaming /
lrwxrwxrwx 1 root root 4096 Oct 27 23:22 Cisco Packet Tracer 8.2.2 /
drwxrwxrwx 1 root root 4096 Feb 10 22:33 Contacts /
lrwxrwxrwx 1 root root 60 Feb 10 22:30 Cookies -> /mnt/c/Users/lmz/AppData/Local/Microsoft/Windows/NetCookies /
drwxrwxrwx 1 root root 4096 Feb 12 22:54 Desktop /
drwxrwxrwx 1 root root 4096 Feb 11 22:16 Documents /
drwxrwxrwx 1 root root 4096 Feb 12 22:53 Downloads /
drwxrwxrwx 1 root root 4096 Feb 10 22:33 Favorites /
drwxrwxrwx 1 root root 4096 Feb 10 22:33 .links /
lrwxrwxrwx 1 root root 30 Feb 10 22:30 'Local Settings' -> /mnt/c/Users/lmz/AppData/Local /
drwxrwxrwx 1 root root 4096 Nov 10 21:35 .vscode /
drwxrwxrwx 1 root root 4096 Feb 10 22:33 .vscode /
-rwxrwxrwx 1 root root 2621440 Feb 11 22:12 NTUSER.DAT*
-rwxrwxrwx 1 root root 65536 Feb 10 22:30 NTUSER.DAT{ebbd224e-e7dc-11ef-ac73-a2b5c8c6a69c}.TM.blf*
-rwxrwxrwx 1 root root 524288 Feb 10 22:30 NTUSER.DAT{ebbd224e-e7dc-11ef-ac73-a2b5c8c6a69c}.TMContainer000000000000000000000001.regtrans-ms*
-rwxrwxrwx 1 root root 524288 Feb 10 22:30 NTUSER.DAT{ebbd224e-e7dc-11ef-ac73-a2b5c8c6a69c}.TMContainer000000000000000000000002.regtrans-ms*
lrwxrwxrwx 1 root root 68 Feb 10 22:30 NetHood -> /mnt/c/Users/lmz/AppData/Roaming/Microsoft/Windows/Network Places /
drwxrwxrwx 1 root root 4096 Oct 24 15:28 OneDrive /
drwxrwxrwx 1 root root 4096 Feb 10 22:33 Pictures /
lrwxrwxrwx 1 root root 68 Feb 10 22:30 PrintHood -> /mnt/c/Users/lmz/AppData/Roaming/Microsoft/Windows/Printer Connections /
lrwxrwxrwx 1 root root 57 Feb 10 22:30 Recent -> /mnt/c/Users/lmz/AppData/Roaming/Microsoft/Windows/Recent /
drwxrwxrwx 1 root root 4096 Feb 10 22:33 Saved Games /

```

4. С помощью какого вызова команды, которая используется в предыдущем вопросе, можно определить псевдоним `d` для некой существующей команды, позволяющий выводить текущую дату и время в формате Unix time (количество секунд, прошедших с 1 января 1970 года 00:00:00 UTC)?

```
alias d='date +%s'
```

```

root@thinkpad: /mnt/c/Users/ \ x + v
root@thinkpad:/mnt/c/Users/lmz# alias d='date +%s'
root@thinkpad:/mnt/c/Users/lmz# d
1739391297
root@thinkpad:/mnt/c/Users/lmz#

```

5. Как сделать так, чтобы псевдоним, определение которого было необходимо для ответа на предыдущий вопрос, автоматически определялся при каждом входе текущего

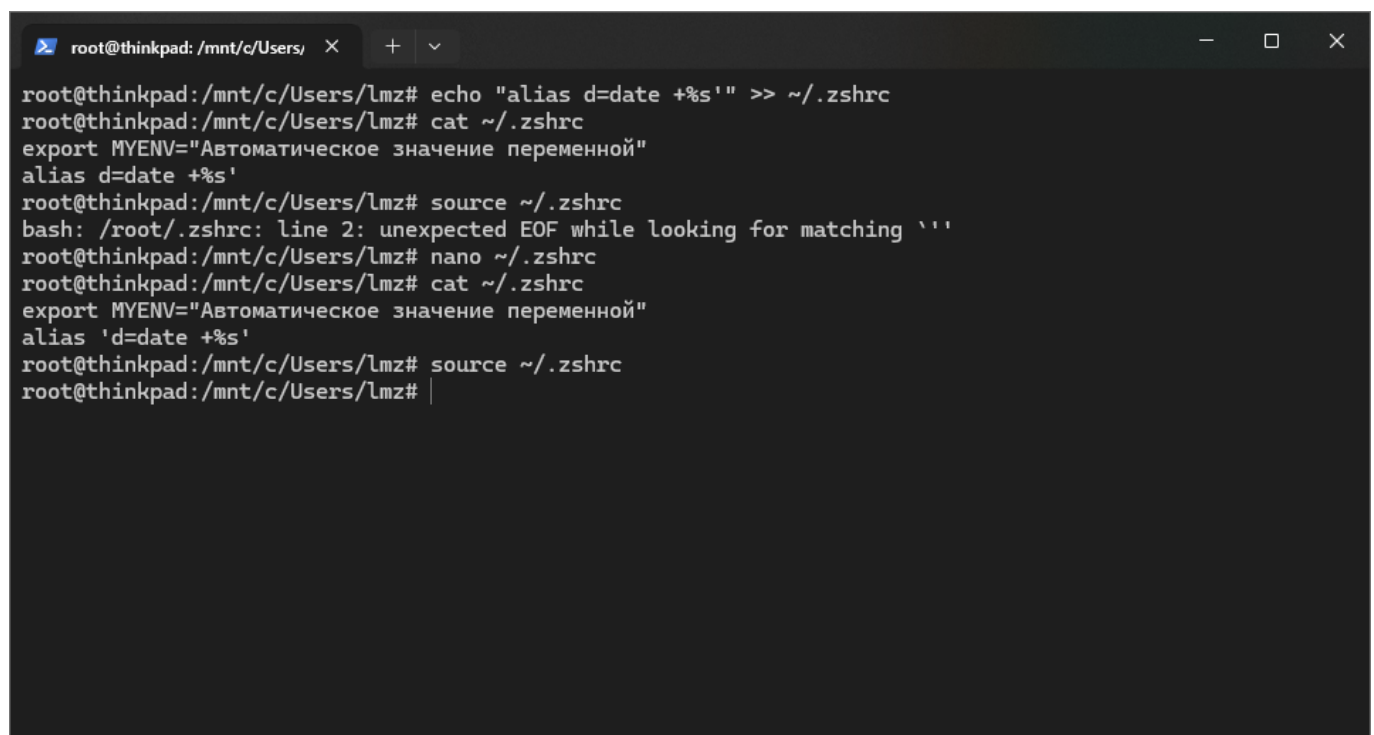
пользователя в систему?

Достаточно добавить строчку установки псевдонима в файл конфигурации оболочки командной строки пользователя (`.bashrc`, `.zshrc`)

```
alias d='date +%s'
```

Для того, чтобы изменения вступили в силу, без перезагрузки текущей сессии пользователя необходимо выполнить команду

```
source ~/.zshrc
```



```
root@thinkpad: /mnt/c/Users/lmz# echo "alias d=date +%s" >> ~/.zshrc
root@thinkpad: /mnt/c/Users/lmz# cat ~/.zshrc
export MYENV="Автоматическое значение переменной"
alias d=date +%s'
root@thinkpad: /mnt/c/Users/lmz# source ~/.zshrc
bash: /root/.zshrc: line 2: unexpected EOF while looking for matching `''
root@thinkpad: /mnt/c/Users/lmz# nano ~/.zshrc
root@thinkpad: /mnt/c/Users/lmz# cat ~/.zshrc
export MYENV="Автоматическое значение переменной"
alias 'd=date +%s'
root@thinkpad: /mnt/c/Users/lmz# source ~/.zshrc
root@thinkpad: /mnt/c/Users/lmz#
```