“Київський фаховий коледж зв’язку”

Циклова комісія комп’ютерної та програмної інженерії

**ЗВІТ ПО ВИКОНАННЮ**

**ЛАБОРАТОРНОЇ РОБОТИ №3**

з дисципліни: «Операційні системи»

**Тема: “Знайомство з базовими командами CLI-режиму в Linux”**

Виконали студенти

групи РПЗ-23а

Команда 1: Туровський В.В.,

Нестропа М.Р. та Мойсеєнко М.О.

Перевірила викладач

Сушанова В.С.

Київ 2025

Робота студентів групи РПЗ-23а Команда 1: Туровський В., Нестропа М., Мойсеєнко М.

**Тема: “Знайомство з базовими командами CLI-режиму в Linux”**

**Мета роботи:**

1. Знайомство з базовими командами CLI-режиму в Linux.
2. Знайомство з базовими текстовими командами в термінальному режимі роботи в різних ОС.

**Матеріальне забезпечення занять:**

1. ЕОМ типу IBM PC.

2. ОС сімейства Windows та віртуальна машина Virtual Box (Oracle).

3. ОС GNU/Linux (будь-який дистрибутив).

4. Сайт мережевої академії Cisco netacad.com та його онлайн курси по Linux

**Короткі теоретичні відомості:**

**Command Line Skills**

The Shell

Once a user has entered a command the terminal then accepts what the user has typed and passes to a shell.

The CLI provides more precise control, greater speed and the ability to automate tasks more easily through scripting.

The shell is the command line interpreter that translates commands entered by a user into actions to be performed by the operating system.

The Linux environment allows the use of many different shells.

The most commonly used shell for Linux distributions is called the Bash shell.

The Bash shell also has many popular features, a few of which are listed below:

* Command line history
* Inline editing
* Scripting: The ability to place commands in a file and then interpret (effectively use Bash to execute the contents of) the file, resulting in all of the commands being executed.
* Aliases: The ability to create short nicknames for longer commands.
* Variables: Used to store information for the Bash shell and for the user.

When a terminal application is run, and a shell appears, displaying an important part of the interface — the prompt.

Typically the prompt contains information about the user and the system. Below is a common prompt structure:

sysadmin@localhost:~$

The prompt shown contains the following information:

* Username (sysadmin)
* System name (localhost)
* Current Directory (~)

The ~ symbol is used as a shorthand for the user's home directory. Typically the home directory for the user is under the /home directory and named after the user account name; for example, /home/sysadmin.

Commands

A command is a software program that when executed on the CLI, performs an action on the computer.

To execute a command, the first step is to type the name of the command.

If you type ls and hit Enter. The result should resemble the example below:

sysadmin@localhost:~$ ls

Desktop Documents Downloads Music Pictures Public Templates Videos

Some commands require additional input to run correctly. This additional input comes in two forms: options and arguments. Options are used to modify the core behavior of a command while arguments are used to provide additional information (such as a filename or a username). Each option and argument is normally separated by a space, although options can often be combined.

The typical format for a command is as follows:

command [options] [arguments]

Keep in mind that Linux is case-sensitive. Commands, options, arguments, variables, and file names must be entered exactly as shown.

Arguments

An argument can be used to specify something for the command to act upon.

If the ls command is given the name of a directory as an argument, it lists the contents of that directory.

sysadmin@localhost:~$ ls /etc/ppp

ip-down.d ip-up.d

Some commands (such as ls) accept multiple arguments:

sysadmin@localhost:~$ ls /etc/ppp /etc/ssh

/etc/ppp:

ip-down.d ip-up.d

/etc/ssh:

moduli ssh\_host\_dsa\_key.pub ssh\_host\_rsa\_key sshd\_configssh\_config

ssh\_host\_ecdsa\_key ssh\_host\_rsa\_key.pub ssh\_host\_dsa\_key ssh\_host\_ecdsa\_key.pub ssh\_import\_id

Options

Options can be used with commands to expand or modify the way a command behaves.

For example, using the -l option of the ls command results in a long listing, providing additional information about the files that are listed:

sysadmin@localhost:~$ ls -l

total 0

drwxr-xr-x 1 sysadmin sysadmin 0 Jan 29 2015 Desktop

drwxr-xr-x 1 sysadmin sysadmin 0 Jan 29 2015 Documents

Output Omitted...

Often the character is chosen to be mnemonic for its purpose, like choosing the letter l for long or r for reverse.

Options can be used in conjunction with other options, for example:

sysadmin@localhost:~$ ls -l -r

sysadmin@localhost:~$ ls -lr

sysadmin@localhost:~$ ls -rl

The order of the combined options isn't important. The output of all of these examples would be the same.

Options are often single letters; however, sometimes they are words or phrases as well.

Typically, older commands use single letters while newer commands use complete words for options.

Single-letter options are preceded by a single dash - character, like the -h option.

Full-word options are preceded by two dash -- characters like the full-word form of the -h option, the --human-readable option.

Variables

A variable is a feature that allows the user or the shell to store data. This data can be used to provide critical system information or to change the behavior of how the Bash shell (or other commands) work. Variables are given names and stored temporarily in memory. There are two types of variables used in the Bash shell: local and environment.

Command Types

The type command can be used to determine information about command type:

type command

There are several different sources of commands within the shell of your CLI:

* Internal commands аlso called built-in commands, these commands are built into the shell itself;
* External commands are stored in files that are searched by the shell;
* Aliases can be used to map longer commands to shorter key sequences;
* Functions can also be built using existing commands to create new commands or override commands built-in to the shell or commands stored in files.

Quoting

Quotation marks are used throughout Linux administration and most computer programming languages to let the system know that the information contained within the quotation marks should either be ignored or treated in a way that is very different than it would normally be treated. There are three types of quotes that have special significance to the Bash shell: double quotes ", single quotes ', and back quotes `. Each set of quotes alerts the shell not to treat the text within the quotes in the normal way.

Control statements

Control statements allow you to use multiple commands at once or run additional commands.

Control statements include:

* Semicolon (;)
* Double ampersand (&&)
* Double pipe (||)

**Завдання для попередньої підготовки:**

**Мойсеєнко Микита - зробив словник, виконав завдання 4 та 5.**

1. \*Прочитайте короткі теоретичні відомості до лабораторної роботи та зробіть невеликий словник базових англійських термінів з питань призначення команд та їх параметрів.

1. **Command** – A program that performs a specific action in the terminal.

**2. Shell** – A program that interprets commands and passes them to the operating system.

**3. Prompt** – A text line displaying user and system information, waiting for a command input.

**4. Argument** – Additional information provided to a command (e.g., a filename or directory name).

**5. Option (Parameter)** – A modifier that changes a command’s behavior (e.g., -l in ls -l).

**6. Internal command** – A command built into the shell (e.g., cd).

**7. External command** – A command stored in a separate file and executed by the shell.

**8. Alias** – A short name for a longer command, created by the user.

**9. Variable** – A named storage location used by the shell or user.

**10. Environment variable** – A global variable accessible across all shell processes.

**11. Quoting** – The use of quotation marks to control how the shell interprets characters.

**12. History** – A list of previously executed commands, accessible via the history command.

**13. Control statements** – Symbols and constructs that manage the execution of multiple commands (;, &&, ||).

**14. Script** – A file containing commands that are executed automatically by the shell.

**15. Path** – The address of a file or directory in the file system.

1. Вивчіть матеріали онлайн-курсу академії Cisco “NDG Linux Essentials”:

* Chapter 5 - Command Line Skills
* Chapter 6 - Getting Help

1. Пройдіть тестування у курсі NDG Linux Essentials за такими темами:

* Chapter 05 Exam
* Chapter 06 Exam

1. \*Дайте визначення наступним поняттям:

**Командний інтерпретатор** - A program that reads and executes commands entered by the user, translating them into actions for the operating system.

**Оболонка** - A command-line interface (CLI) that allows users to interact with the operating system by entering commands. It acts as a bridge between the user and the system.

**Команда** - An instruction given by the user in the terminal to perform a specific task, such as listing files, navigating directories, or executing programs.

1. \*\*Дайте відповіді на наступні питання:

* **Яку базову інформацію надає рядок запрошення prompt?**

The terminal prompt displays information about the current system and user state. It usually includes the username, hostname, current directory, shell type, and special symbols ($ for a regular user, # for root).

* **Для чого команді потрібні параметри та аргументи?**

Parameters and arguments modify a command's behavior. Arguments typically specify files or directories the command should work with, while parameters (options) configure the command (e.g., -l in ls -l for detailed file listing).

* **Яке призначення команд ls, які параметри та аргументи вона може мати? Наведіть 3 приклади.**

The ls command lists files and directories. Common options:

- -l – displays detailed information about files.

- -a – shows all files, including hidden ones (. and ..).

- -h – presents file sizes in a human-readable format (KB, MB).

Examples:

- ls -l – lists files in detailed format.

- ls -a – shows all files, including hidden ones.

- ls -lh /home/user – lists files in the specified folder with human-readable sizes.

* **Яким чином можна використати історію команд, які переваги це надає?**

Command history allows users to view and reuse previously entered commands, saving time. Key commands:

- history – displays the list of past commands.

- !n – executes the command at position n in history.

- !! – repeats the last executed command.

* **Яке призначення команди echo?**

The echo command prints text or variable values to the terminal. Examples:

- echo "Hello, world!" – prints "Hello, world!".

- echo $HOME – displays the user's home directory path.

* **Охарактеризуйте поняття змінної в оболонці Bash, які типи змінних вона підтримує?**

A Bash variable is a named storage location. Types of variables:

- Local variables (defined within the current shell session).

- Global variables (exported to other processes via export).

- Special variables (e.g., $? – exit status of the last command).

* **Яке призначення команд env, export та unset?**

- env – displays environment variables.

- export VAR=value – makes a variable available to child processes.

- unset VAR – removes a variable.

* **Які команди для отримання довідки по командам в терміналі ви знаєте?**

- man <command> – opens the manual for the command.

- --help – provides a short help message (ls --help).

- info <command> – gives detailed documentation (similar to man).

1. Підготувати в електронному вигляді початковий варіант звіту:

* Титульний аркуш, тема та мета роботи
* Словник термінів
* Відповіді на п.4 та п.5 з завдань для попередньої підготовки

**Хід роботи:**

* 1. Опрацюйте всі приклади команд, що представлені у лабораторній роботі курсу ***NDG Linux Essentials - Lab 5: Command Line Skills*** та ***Lab 6: Getting Help.*** Створіть таблицю для опису цих команд

**Виконав Нестропа Максим, та дав відповіді на контрольні запитання**

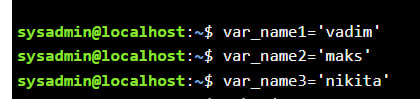
| Назва команди | Її призначення та функціональність |
| --- | --- |
| ls | Виводить інформації про каталоги та файли. За замовчуванням без аргументів відображає інформацію для поточного каталогу |
| ls -l | Використанні параметру **-l** в команді **ls** дозволяє відобразити інформацію про файли, розташовані в поточному робочому каталозі, у довгому форматі, який надає більш розширену додаткову інформацію |
| ls -l /tmp | Використання аргументу **/tmp** в поєднанні з параметром **-l** в команді **ls** дозволяєвідобразити детальну інформацію про файли в каталозі /tmp. |
| history | Виводить історію команд |
| variable1 | Створення локальної назви та оголошення її |
| echo | Вивід |
| man | Виведення базової бібліотеки |
| env | Список змінних середовища |
| export | Перетворення локольної зміної на зміну середовища |
| which | Шукає розташування команди |
| whereis | Для пошуку команд або монуалів |
| locate | Для пошуку будь якого файлу або каталогу |
| alias | Для скорочення назви або коду |
| type | Відображає місцезнаходження |
| uname | Довідка |
| for | Цикл |
| pwd | Відображення поточного місцезнаходження |

**Туровський Вадим - виконав завдання 2**

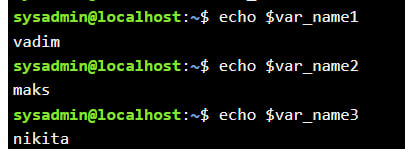
* 1. Робота в в терміналі (закріплення практичних навичок) **обов'язково представити свої скріншоти:**

2.1. Робота зі змінними (Variables) та псевдонімами (Aliases) в терміналі:

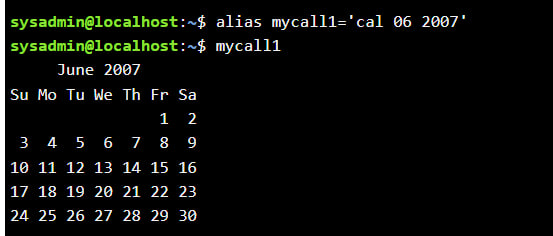
* Створіть змінні, що будуть містити Ваші імена та прізвища $var\_name1, $var\_name2, $var\_name3

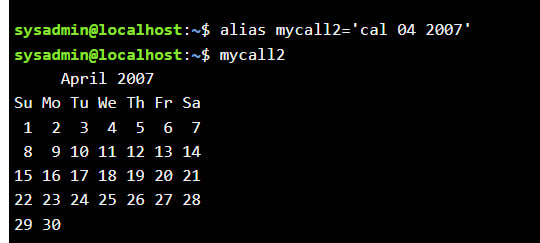


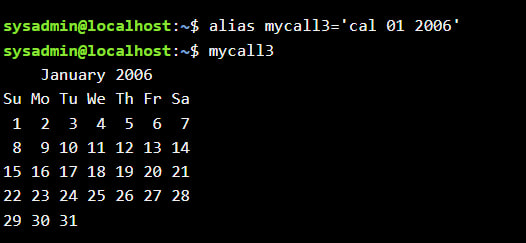
* За допомогою команди echo виведіть імена студентів вашої команди



* Створіть псевдоніми mycal1, mycal2, mycal3 для команди cal для автоматичного виведення календарю вашого року народження

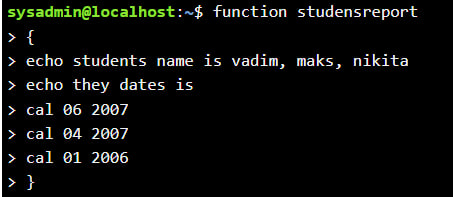


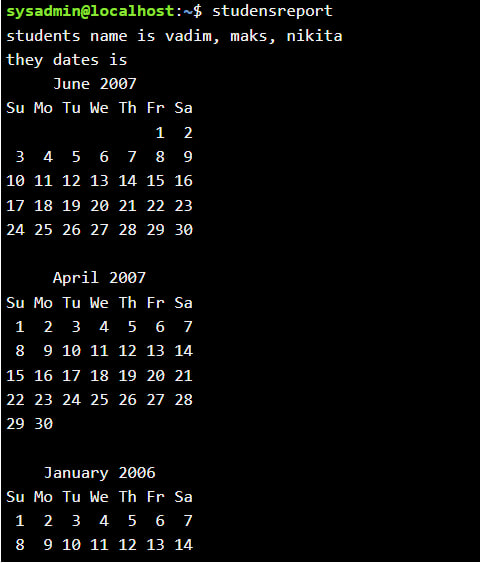




2.2. \*Робота з функціями (Functions) в терміналі:

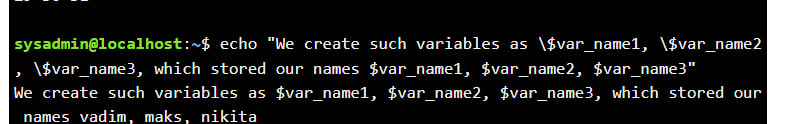
* Створіть функцію students\_report, що порядково буде виводити спочатку імена студентів Вашої команди, а потім роки їх народження



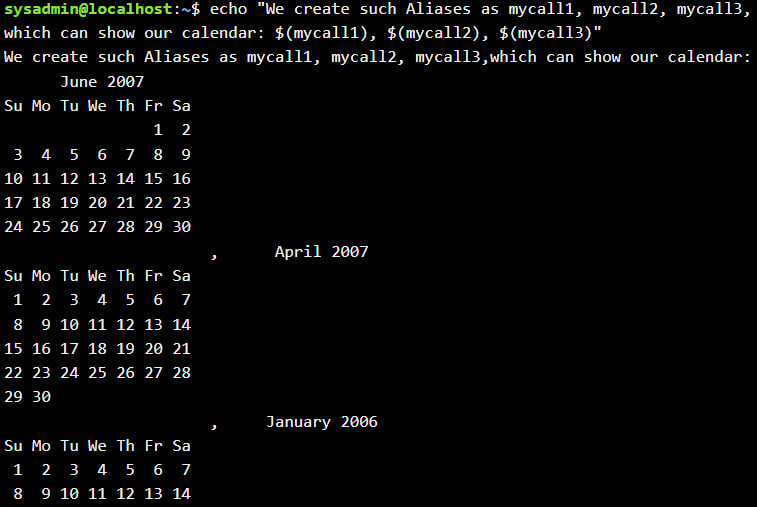


2.3. \*Робота з лапками (Quoting) в терміналі. Виведіть в командному рядку наступні речення:

* “We create such variables as $var\_name1, $var\_name2, $var\_name3, which stored our names Name1, Name2, Name3” (у реченні спочатку виводимо назви змінних, а потім їх вміст)

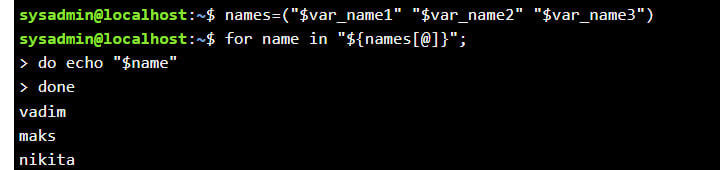


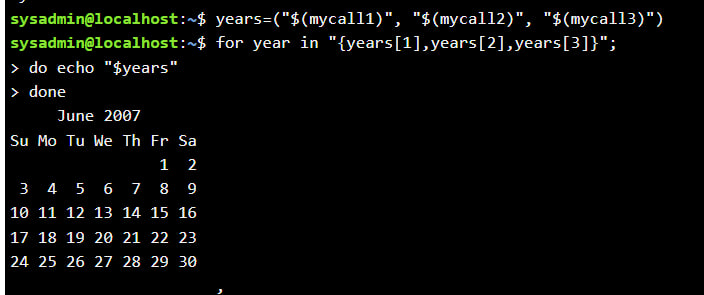
* “We create such Aliases as mycal1, mycal2, mycal3, which can show our calendars: Calendar1, Calendar2, Calendar3” (у реченні спочатку виводимо назву команди-псевдонімів, потім вивід цих команд).



2.4. \*\*Робота з інструкціями керування (Control Statements) в терміналі:.

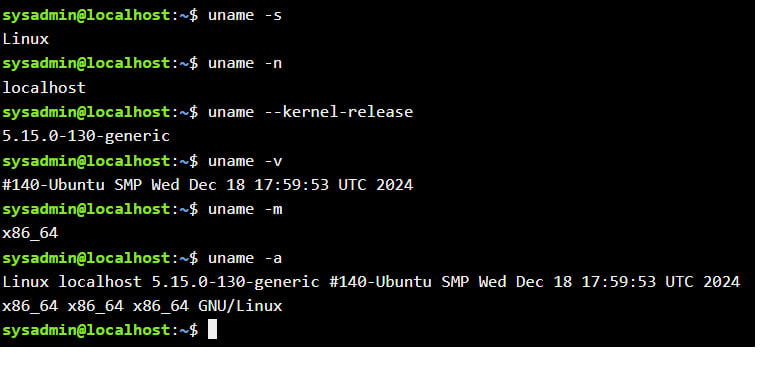
* Чи можна завдання 2.1 та 2.2 ходу роботи виконати через інструкції керування без написання окремої функції, як це буде виглядати?





2.5. Робота з командами довідки (Man Pages) в терміналі:

* На прикладі команди uname продемонструйте як отримати довідку. На основі отриманої додаткової інформації наведіть 5 різних варіантів виводу результату інформації по даній команді з використанням 5 різних параметрів (Options)



**Контрольні запитання:**

1. **Які типи команд існують в оболонці Bash?**

Bash supports built-in commands (e.g., cd, echo, export), external commands (e.g., ls, grep, awk), shell scripts, and aliases.

1. **Що таке змінні оточення? Які вони бувають. Як їх можна переглянути в терміналі?**

Environment variables store configuration settings and system information. There are system-wide variables (e.g., $PATH, $HOME) and user-defined variables. They can be viewed using commands like env, printenv, or echo $VARIABLE\_NAME.

1. **\*Опишіть змінну $PS1. Як в терміналі переглянути її вміст?**

You can change the command prompt using export PS1="NewPrompt> ". To make it permanent, add this line to .bashrc or .bash\_profile.

1. **\*Як можна змінити значення змінної $PS1? Що при цьому відбудеться в рядку запрошенні в bash (рядок запрошення перед початком кожної команди). Як змінити значення цієї змінної не на поточний сеанс, а за замовчуванням?**

$PS1 defines the Bash prompt appearance. Changing it updates the prompt for new commands. To modify it for the current session:

export PS1="\u@\h:\w$ "

To make it permanent, add the same line to ~/.bashrc.

1. **\*Для чого використовують лапки в оболонці Bash? .**

Symbolic (ln -s target link) and hard links (ln target link) allow file and directory references. They are used for shortcuts, shared libraries, and backup management.

1. **\*\*Для чого використовують інструкції керування, які їх види Ви знаєте?**

- Control structures manage execution flow in scripts. Common ones include:

- Conditional statements (if, case)

- Loops (for, while, until)

- Logical operators (&&, ||)

- Functions

1. **\*\*В чому різниця якщо в кінці рядку запрошення bash стоїть символ $ чи #? Наприклад на екрані ми бачимо наступні записи**





[user@localhost ~]$ # Regular user

[root@localhost ~]# # Root user

1. **\*\*Яке призначення команд whereis та locate? Яка між ними відмінність?**

whereis finds the binary, source, and manual files for a command (whereis ls).

- locate searches for files in the database index (locate filename). It is faster than find but may be outdated.

**Оформлення звіту:**

1. Титульний аркуш
2. Тема та мета роботи
3. Завдання попередньої підготовки
4. Основні позиції ходу роботи
5. Відповіді на контрольні запитання
6. Висновки за результатами роботи **(обов’язково!!!)**

**Система оцінювання лабораторної роботи:**

Виконано завдання базового рівня складності - **3 бали**

Виконано завдання базового та середнього рівня складності - **4 бали**

Виконано завдання всіх рівнів складності (в тому числі й підвищеного) - **5 балів**

Завдання середнього рівня складності позначені в завданнях (\*)

Завдання підвищеного рівня складності позначені в завданнях (\*\*)

**Примітка**: за виконання робіт в командах та оформлення звітів з використанням системи контролю версій (git) та англійської мови може бути нараховано **додатковий 1 бал**.