“Київський фаховий коледж зв’язку”

Циклова комісія Комп’ютерної інженерії

**ЗВІТ ПО ВИКОНАННЮ**

**ЛАБОРАТОРНОЇ РОБОТИ №5**

з дисципліни: «Операційні системи»

**Тема: «Знайомство з командами навігації по файловій системі та керування файлами та каталогами»**

Виконавли студенти

групи РПЗ-03

Команда: Губенко Є.О..,

Заїка С.В. та Кресан Р.А.

Перевірив викладач

Сушанова В.С.

Київ 2022

**Мета роботи:**

1. Отримання практичних навиків роботи з командною оболонкою Bash.
2. Знайомство з базовими командами навігації по файловій системі.
3. Знайомство з базовими командами для керування файлами та каталогами.

**Матеріальне забезпечення занять**

1. ЕОМ типу IBM PC.

2. ОС сімейства Windows (Windows 7).

3. Віртуальна машина – Virtual Box (Oracle).

4. Операційна система GNU/Linux – CentOS.

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

**Завдання для попередньої підготовки**

***Готував матеріал студент Губенко Є.О.***

1. Прочитавши матеріал з коротких теоретичних відомостей дайте відповіді на наступні питання:
   1. Порівняйте файлові структури Windows-подібної та Linux-подібної системи.

*Файлові структури Windows-подібної та Linux-подібної систем відрізняються в кількох аспектах:*

1. ***Ієрархія файлів:***

*Windows використовує ієрархію файлів з одним деревом каталогів. Найбільш відомі каталоги включають "Program Files", "Windows", "Users" та "Documents and Settings".*

*У Linux, ієрархія файлів має багато рівнів та починається з кореневого каталогу "/". Типові каталоги включають "/bin", "/etc", "/home", "/usr" та "/var".*

1. ***Розширення файлів:***

*У Windows, файлові розширення додаваються до імені файлу після крапки, яка розділяє ім'я та розширення (наприклад, "file.txt").*

*У Linux, розширення файлів не є обов'язковим, і ім'я файлу може складатися лише з імені без крапки (наприклад, "file").*

1. ***Файлові системи:***

*У Windows, NTFS та FAT32 є найпоширенішими файловими системами.*

*У Linux, EXT4 та Btrfs є найпопулярнішими файловими системами, а також є підтримка для інших, таких як XFS, ZFS, та NTFS.*

1. ***Керування дозволами на файли:***

*У Windows, дозволи на файли та папки керуються за допомогою ACL (Access Control Lists).*

*У Linux, дозволи на файли та папки керуються за допомогою рівня доступу до файлів, які встановлюються для власника, групи та всіх користувачів.*

1. ***Командний рядок:***

*У Windows, командний рядок є відомим як командний рядок Windows (CMD), та використовує команди, які можуть бути виконані в CMD.*

*У Linux, командний рядок є відомим як оболонка (shell), і найпоширенішою оболонкою є bash, яка використовує спеціальні команди, які можна виконати з командного рядка.*

1. ***Символьні посилання:***

*У Windows, символьні посилання (symbolic links) відомі як ярлики (shortcuts), які є посиланнями на файл або папку та можуть бути розташовані в будь-якому місці.*

*У Linux, символьні посилання є файлами, що містять посилання на інший файл або папку, і можуть бути створені в будь-якому місці.*

1. ***Розділення диску:***

*У Windows, диск можна розділити на декілька розділів, які відображаються як окремі диски з різними літерами приводу.*

*У Linux, диск можна розділити на декілька розділів, які відображаються як окремі розділи в системі каталогів.*

1. ***Налаштування системи:***

*У Windows, налаштування системи зазвичай здійснюється через панель управління або за допомогою інших графічних інтерфейсів.*

*У Linux, налаштування системи зазвичай здійснюється через конфігураційні файли, що знаходяться в /etc, або через командний рядок.*

1. ***Розмір системи:***

*У Windows, система зазвичай займає більше місця на диску через велику кількість встановлених додатків та драйверів.*

*У Linux, система зазвичай займає менше місця на диску, оскільки вона має менше встановлених додатків та драйверів за замовчуванням.*

1. ***Доступність додатків:***

*У Windows, багато програм та ігор розробляються спеціально для цієї операційної системи, що може зробити їх більш доступними для користувачів.*

*У Linux, деякі програми можуть бути обмежені щодо доступності, але відкритий код дозволяє розробникам створювати та розповсюджувати свої програмне забезпечення безкоштовно* *, що може зробити їх більш доступними для користувачів.*

1. ***Системні вимоги:***

*У Windows, системні вимоги можуть бути високими, залежно від версії операційної системи та додатків, які потрібно встановити.*

*У Linux, системні вимоги зазвичай менші, оскільки операційна система може працювати на менш потужних комп'ютерах.*

1. ***Консоль:***

*У Windows, консоль (command prompt) доступна користувачам, але вона не має такої ж потужності, як консоль в Linux.*

*У Linux, консоль є потужним інструментом для взаємодії з операційною системою та виконання різних завдань.*

1. ***Доступ до файлів:***

*У Windows, доступ до файлів може бути обмежений системою безпеки, яка використовує ACL (Access Control Lists).*

*У Linux, доступ до файлів може бути контрольований за допомогою прав доступу (permissions), що встановлюються для власника, групи та інших користувачів.*

1. ***Антивірусна захист:***

*У Windows, антивірусний захист є важливим аспектом безпеки, оскільки ця операційна система є більш піддається вірусам та шкідливому ПО.*

*У Linux, антивірусний захист може бути менш важливим, оскільки ця операційна система має більшу безпеку за замовчуванням.*

1. ***Підтримка драйверів:***

*У Windows, підтримка драйверів зазвичай забезпечується виробниками апаратного забезпечення, які розробляють драйвери для своїх пристроїв.*

*У Linux, драйвери зазвичай знаходяться в ядрі операційної системи та оновлюються разом з ним.*

*Загалом, Windows та Linux мають різні до роботи з операційною системою та різні переваги та недоліки. Windows зазвичай має більш простий та інтуїтивний інтерфейс користувача, а також більш широкий вибір програм та додатків, але вона може бути менш стабільною та безпечною. Linux зазвичай має більшу стабільність та безпеку, а також більшу контрольованість та налаштування, але вона може бути менш доступною для новачків та мати обмежений вибір програм та додатків.*

* 1. Розкрийте поняття FHS. Як даний стандарт використовується в контексті файлових систем?

*FHS - це скорочення від Filesystem Hierarchy Standard, що перекладається як Стандарт ієрархії файлової системи. Це стандарт, який описує, яким чином файли та каталоги повинні бути організовані в файловій системі в Unix-подібних операційних системах.*

*FHS є важливим стандартом в Unix-подібних операційних системах, оскільки він забезпечує стандартну ієрархію файлової системи, яка дозволяє розробникам програм і адміністраторам систем більш просто і ефективно працювати з файлами та каталогами.*

*Згідно з FHS, файлова система Unix-подібної ОС повинна мати таку ієрархію:*

* */: кореневий каталог*
* */bin: бінарні файли системи*
* */sbin: системні бінарні файли*
* */boot: файли, необхідні для завантаження системи*
* */dev: файли пристроїв*
* */etc: конфігураційні файли системи*
* */home: домашні каталоги користувачів*
* */lib: бібліотеки для програм*
* */media: монтовані пристрої*
* */mnt: тимчасово змонтовані файлові системи*
* */opt: додаткові програми*
* */proc: інформація про процеси*
* */root: домашній каталог користувача root*
* */run: файли runtime*
* */sbin: системні бінарні файли*
* */srv: дані для служб*
* */sys: інформація про пристрої та ядро*
* */tmp: тимчасові файли*
* */usr: додаткові програми та файли*
* */var: змінні файли системи*

*Дотримання стандарту FHS дозволяє розробникам програм та адміністраторам систем легше знаходити та управляти файлами та каталогами в системі. Крім того, дотримання стандарту забезпечує більшу переносимість програм між різними Unix-подібними операційними системами.*

* 1. Перерахуйте основні команди для роботи з файлами та каталогами в Linux: створення, переміщення, копіювання, видалення.

*Основні команди для роботи з файлами та каталогами в Linux включають наступні:*

* *Створення файлів та каталогів:*

*touch file.txt: створити порожній файл з назвою file.txt*

*mkdir directory: створити каталог з назвою directory*

* *Переміщення файлів та каталогів:*

*mv source\_file.txt destination/: перемістити файл source\_file.txt до каталогу destination/*

*mv source\_directory/ destination/: перемістити каталог source\_directory до каталогу destination/*

*mv old\_file.txt new\_file.txt: перейменувати файл з old\_file.txt на new\_file.txt*

* *Копіювання файлів та каталогів:*

*cp source\_file.txt destination/: скопіювати файл source\_file.txt до каталогу destination/*

*cp source\_directory/ destination/: скопіювати каталог source\_directory до каталогу destination/*

* *Видалення файлів та каталогів:*

*rm file.txt: видалити файл file.txt*

*rm -r directory: видалити каталог directory та всі файли та каталоги в ньому*

*rmdir directory: видалити порожній каталог directory*

*Важливо пам'ятати, що команди rm та mv можуть бути небезпечними, оскільки вони видаляють або переміщують файли без підтвердження. Рекомендується завжди перевіряти перед виконанням цих команд, щоб упевнитися, що вони виконуються правильно.*

**Хід роботи**

1. Початкова робота в CLI-режимі в Linux ОС сімейства Linux:
   1. Запустіть віртуальну машину VirtualBox, оберіть CentOS та запустіть її. Виконайте вхід в систему під користувачем: CentOS, пароль для входу: reverse (якщо виконуєте ЛР у 401 ауд.) та запустіть термінал.
   2. Запустіть віртуальну машину Ubuntu\_PC (якщо виконуєте завдання ЛР через академію netacad)
   3. Запустіть свою операційну систему сімейства Linux (якщо працюєте на власному ПК та її встановили) та запустіть термінал.
2. Опрацюйте всі приклади команд, що представлені у лабораторних роботах курсу NDG Linux Essentials - Lab 7: Navigating the Filesystem та Lab 8: Managing Files and Directories. Створіть таблицю для опису цих команд\*\*\*?

***Готував матеріал студент Губенко Є.О***

|  |  |
| --- | --- |
| Назва команди | Її призначення та функціональність |
| pwd | Визначає місце знаходження користувача у файловій системі, показує поточну робочу директорію (print working directory) |
| cd Documents | Команда cd здійснює перехід до каталогу, який у неї вказаний як  аргумент. В даному випадку це каталог Documents |
| cd / | cd - це команда в терміналі, яка призначена для зміни поточної робочої директорії. Коли ви виконуєте команду cd, ви переходите в іншу директорію, вказану як аргумент команди. Наприклад, якщо ви виконаєте команду cd /home/user/Documents, ви перейдете в директорію /home/user/Documents. |
| cd /home | cd /home - це команда в терміналі, яка призначена для переходу до директорії /home. Якщо ви виконаєте цю команду, ви перейдете в директорію /home, де зазвичай зберігаються домашні директорії користувачів. Виконання команди cd /home еквівалентне виконанню команди cd без аргументів, оскільки /home є домашньою директорією на більшості систем Linux. |
| cd ~ | cd ~ - це команда в терміналі, яка призначена для переходу до домашньої директорії поточного користувача. Знак ~ у команді cd позначає домашню директорію. Ця команда допомагає вам швидко перейти до вашої основної директорії, щоб ви могли продовжувати роботу з файлами та документами. |
| echo ~ ~sysadmin ~root ~mail ~nobody | ~sysadmin, ~root, ~mail, і ~nobody - це спеціальні скорочення, які позначають домашні директорії для користувачів sysadmin, root, mail та nobody відповідно. Використовуючи ці скорочення в команді echo, ви можете вивести шлях до домашньої директорії для цих користувачів. Наприклад, якщо ви виконаєте команду echo ~sysadmin, то на екрані з'явиться шлях до домашньої директорії користувача sysadmin. |
| cd ~root | cd ~root - це команда в терміналі, яка призначена для переходу до домашньої директорії користувача root. Знак ~ позначає домашню директорію поточного користувача, тому ~root вказує на домашню директорію користувача root |
| cd /usr/bin | cd /usr/bin - це команда в терміналі, яка призначена для переходу до директорії /usr/bin. Шлях /usr/bin є одним з стандартних шляхів у більшості дистрибутивів Linux і містить виконувані файли системи. |
| cd /usr | cd /usr - це команда в терміналі, яка призначена для переходу до директорії /usr. Директорія /usr містить файли, які використовуються для забезпечення функцій, що не є необхідними для роботи системи на початку її запуску, але є необхідними для забезпечення додаткових функцій після її запуску. |
| cd /usr/share/doc | cd /usr/share/doc - це команда в терміналі, яка призначена для переходу до директорії /usr/share/doc. Директорія /usr/share/doc містить документацію, яку можна використовувати з програмами, що встановлені на системі. |
| cd bash | cd bash - це команда в терміналі, яка призначена для переходу до піддиректорії з назвою "bash" в поточній робочій директорії.  Проте в більшості систем, директорії з назвою "bash" не містяться в поточній робочій директорії користувача. Найчастіше, "bash" є назвою інтерпретатора командного рядка, що використовується в більшості дистрибутивів Linux. |
| cd .. | cd .. - це команда в терміналі, яка призначена для переходу до батьківської директорії відносно поточної робочої директорії. |
| cd ../dict | Команда cd ../dict призначена для переходу до директорії з назвою "dict" в батьківській директорії відносно поточної робочої директорії. |
| ls | Команда ls в терміналі призначена для відображення списку файлів і папок, які знаходяться в поточній робочій директорії.  При виклику команди ls без будь-яких аргументів, вона відображає список файлів та папок в поточній директорії у вигляді простого тексту. При цьому, файли та папки відображаються у вигляді їх назв та деякої інформації про них, такої як дата та час останньої зміни, розмір файлу та права доступу. |
| -a | Опція -a команди ls призначена для відображення всіх файлів і папок, включаючи ті, що починаються з крапки (приховані файли).  За замовчуванням, команда ls не відображає приховані файли. Однак, якщо викликати команду ls з опцією -a, то всі файли та папки, які знаходяться в поточній робочій директорії, будуть відображені, включаючи файли з назвами, що починаються з крапки. |
| -l /etc/hosts | Опція -l команди ls призначена для відображення детальної інформації про файли та папки, включаючи права доступу, власника, групу, розмір файлу та дату та час останньої зміни.  Якщо викликати команду ls з опцією -l та вказати аргумент /etc/hosts, то буде відображено детальна інформація про файл /etc/hosts. Цей файл містить список IP-адрес та відповідних їм доменних імен. |
| -R /etc/udev | Опція -R команди ls призначена для відображення директорій та піддиректорій у рекурсивному порядку. Це означає, що будуть відображені всі файли та папки, що знаходяться в вказаній директорії та її піддиректоріях.  Якщо викликати команду ls з опцією -R та вказати аргумент /etc/udev, то буде відображено всі файли та папки в директорії /etc/udev та її піддиректоріях у рекурсивному порядку. |
| -d /etc/s\* | Опція -d команди ls призначена для відображення тільки директорії, що вказані в командному рядку. Вона не дозволяє відображати детальну інформацію про файли та папки, а лише назви директорій.  Якщо викликати команду ls з опцією -d та вказати аргумент /etc/s\*, то будуть відображені назви всіх директорій, що починаються з літери "s" в директорії /etc. |
| –d /etc/[abcd]\* | Опція -d команди ls призначена для відображення тільки директорії, що вказані в командному рядку. Вона не дозволяє відображати детальну інформацію про файли та папки, а лише назви директорій.  Якщо викликати команду ls з опцією -d та вказати аргумент /etc/[abcd]\*, то будуть відображені назви всіх директорій, що починаються з літер a, b, c або d в директорії /etc. |
| echo \* | Команда echo виводить на екран текстові дані, які їй передані у вигляді аргументів. Коли до команди echo додається символ джокера \*, вона виводить на екран список всіх файлів та папок в поточній директорії (за винятком тих, які починаються з крапки ".", тобто прихованих файлів). |
| echo D\*  echo P\* | Команда echo виводить на екран текст, який їй передається у вигляді аргументів. Якщо до команди echo додається символ джокера \*, то вона виводить на екран список всіх файлів та папок, імена яких починаються на задану послідовність символів.  У випадку echo D\*, команда виведе на екран імена всіх файлів та папок, імена яких починаються з літери "D".  У випадку echo P\*, команда виведе на екран імена всіх файлів та папок, імена яких починаються з літери "P". |
| echo \*s | echo \*s, команда виведе на екран імена всіх файлів та папок, імена яких містять літеру "s". |
| echo D\*n\*s | echo D\*n\*s, команда виведе на екран імена всіх файлів та папок, імена яких починаються з літери "D", містять якусь кількість символів між першою та останньою літерою "n" та закінчуються на літеру "s". |
| echo ?????? | echo ??????, команда виведе на екран імена всіх файлів та папок, імена яких складаються з рівно 6 символів |
| echo D???????? | echo D????????, команда виведе на екран імена всіх файлів та папок, які починаються з літери D і мають довжину 9 символів |
| echo ?????\*s | ?????\*s, команда виведе на екран імена всіх файлів та папок, які мають довжину не менше 6 символів, починаються з будь-яких 5 символів, а закінчуються на s. |
| echo [DP]\*  echo [!DP]\* | [DP]\*, команда виведе на екран імена всіх файлів та папок, які починаються з літери "D" або "P".  [!DP]\*, команда виведе на екран імена всіх файлів та папок, які не починаються з літер "D" або "P".  Знак ! вказує на інвертування шаблону |
| echo [D-P]\*  echo [!D-P]\* | [D-P]\*, команда виведе на екран імена всіх файлів та папок, які починаються з літер в діапазоні між "D" та "P" включно.  [!D-P]\*, команда виведе на екран імена всіх файлів та папок, які не починаються з літер в діапазоні між "D" та "P" включно. |
| rm hosts  cp –v /etc/hosts hosts | rm hosts - ця команда видаляє файл з іменем "hosts" у поточній директорії.  cp -v /etc/hosts hosts - ця команда копіює файл "/etc/hosts" у поточну директорію з новим іменем "hosts". Параметр "-v" забезпечує вивід на екран інформації про процес копіювання. |
| -l hosts  –p hosts /home/sysadmin | ls -l hosts - ця команда виводить детальну інформацію про файл з іменем "hosts" у поточній директорії, таку як права доступу, власник, розмір тощо.  cp -p hosts /home/sysadmin - ця команда копіює файл з іменем "hosts" у поточну директорію до директорії "/home/sysadmin" зі збереженням всіх атрибутів файлу, таких як дата створення, дата модифікації тощо. Параметр "-p" забезпечує збереження всіх атрибутів файлу під час копіювання. |
| mkdir Myetc  cp –R /etc/udev Myetc  ls –l Myetc  ls –lR Myetc | mkdir Myetc - ця команда створює нову директорію з іменем "Myetc" у поточній директорії.  cp -R /etc/udev Myetc - ця команда копіює директорію "/etc/udev" разом з її вмістом до директорії "Myetc". Параметр "-R" дозволяє скопіювати директорію разом з її піддиректоріями та файлами.  ls -l Myetc - ця команда виводить детальну інформацію про директорію з іменем "Myetc" у поточній директорії, таку як права доступу, власник, розмір тощо.  ls -lR Myetc - ця команда виводить детальну інформацію про всі файли та директорії, що містяться в директорії "Myetc" у поточній директорії та всіх її піддиректоріях. Параметр "-R" дозволяє вивести інформацію про всі піддиректорії та файли рекурсивно. |
| rm -r Myetc | команда rm -r є потенційно небезпечною, оскільки вона рекурсивно видаляє файли і каталоги без підтвердження |

\*\*\*Скріншоти виконання команд в терміналі можна не представляти, достатньо коротко описати команди в таблиці.

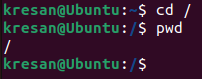
***Готував матеріал студент Кресан Руслан***

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

* Визначте ваш поточний робочий каталог;



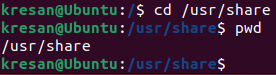
* Перейдіть до кореневого каталогу та визначте Ваш поточний робочий каталог (дві команди);



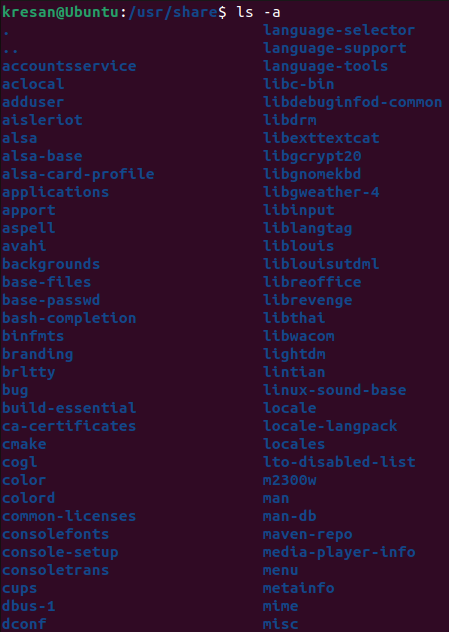
* Перегляньте вміст поточного каталогу у довгому форматі (скористайтесь відповідним ключем команди ls);



* Перейдіть до каталогу /usr/share та визначте Ваш поточний робочий каталог (дві команди)



* Перегляньте вміст поточного каталогу включаючи і приховані файли (hidden files) (скористайтесь відповідним ключем команди ls);



* Перейдіть до каталогу /etc;



* Перегляньте вміст даного каталогу, але щоб виводило тільки назви файлів, що починаються з літери вашого імені;



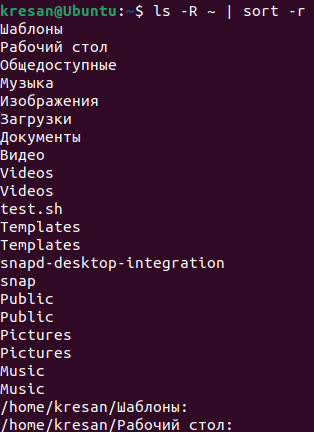
* Перегляньте вміст даного каталогу, але щоб виводило тільки файли, назви яких складаються з 6 літер;



* Перегляньте вміст даного каталогу, але щоб виводило тільки файли, назви яких закінчуються на літери ваших імен, наприклад якщо ваші імена Ivan, Anna, Maks, то вибірку робиму, щоб назви файлів закінчувались на літери [i,a,m];

|  |  |
| --- | --- |
|  | *r – Руслан*  *s – Сергій*  *e – Єгор* |

* Перейдіть до домашнього каталогу поточного користувача та перегляньте його вміст у рекурсивному (зворотному до алфавітного) форматі (виконати цю дію через конвеєр команд);



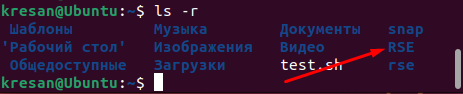
* В поточній директорії створити директорію з назвою вашої групи;

*Я вирішив що назва нашої групи буде складатися з перших літер наших імен.*

*Назва групи “RSE”*



* Переглянути оновлений вміст домашнього каталогу поточного користувача. Скористайтесь ключем -r команди ls, яку інформацію ви отримаєте?



* Перейдіть у створену вами директорію з назвою Вашої групи та створіть у ній порожній файл lab5



* Створити в даній директорії 3 директорії з прізвищами студентів вашої команди surname1, surname2, surname3\* (команда mkdir мульти аргумента, тому всі три каталоги можна створити однією командою);



* Перейдіть у перший підкаталог surname1 та створіть порожній файл з ім’ям першого студента name1;





* За допомогою команди echo «Hello, my name is Name1» > name1 внесіть у цей файл дані про студента (символ > дозволяє вивід команди echo перенаправити одразу у файл name1;



* Перегляньте вміст файлу name1 за допомогою команди cat name1 (має містити щойно введену Вами інформацію)



* Зробіть копію першого файлу name1 та перейменуйте її у файл з другим ім’ям студенту Вашої команди name2;



* Перегляньте вміст каталогу, обидва файли мають з’явитися;



* Перегляньте вміст другого файлу cat name2 (він має поки що містити повну копію вмісту файлу name1)



* Замініть зміст файлу name2, щоб він містив відповідне ім’я другого студента за допомогою команди echo «Hello, my name is Name2» > name2



* Перегляньте вміст другого файлу cat name2 (він вже має містити оновлену інформацію)



* Перемістіть файл name2 у директорію surname2;



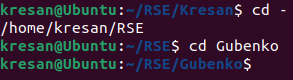
* Зробіть копію першого файлу name1 та перейменуйте її у файл з третім ім’ям студенту Вашої команди name3;



* Перемістіть файл name3 у директорію surname3;



* Перейдіть до директорії surname3;



* Перегляньте вміст третього файлу командою cat name3 (він має містити дані про другого студента)



*Він містить дані першого студента, а не другого, тому що ми копіювали файл першого студента у попередньому пункті (Зробіть копію першого файлу name1 та перейменуйте її у файл з третім ім’ям студенту Вашої команди name3;)*

* Замініть зміст файлу name3, щоб він містив відповідне ім’я третього студента за допомогою команди echo «Hello, my name is Name3» > name3



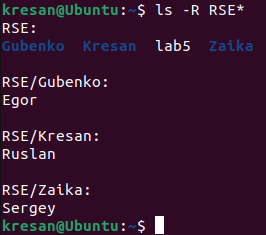
* Перегляньте вміст файлу за допомогою cat name3 (він вже має містити оновлену інформацію)



* Поверніться до домашнього каталогу користувача;



* Перегляньте вміст даного каталогу, але щоб виводило тільки Ваш підкаталог з назвою групи та весь його вміст (підкаталоги surname1, surname2, surname3 та файли name1, name2, name3) до того ж файли та катлоги були відкоремлені кольорами (скористайтесь відповідним ключем -R команди ls та не забудьте використати спеціальний glob-шаблон [імя каталогу]\*)



***Готував матеріал студент Заїка С.В.***

1. Опишіть дії, які виконують команди для переміщення по системі каталогів:

* команда cd /

*Команда cd / переміщує користувача в кореневий каталог системи. Тобто, якщо на даний момент користувач знаходиться в будь-якому іншому каталозі, ця команда переміщує його в каталог "/" (кореневий каталог).*

* команда cd /home

*Команда cd /home переміщує користувача в підкаталог "/home". Цей каталог часто містить домашні каталоги користувачів, тому ця команда часто використовується для швидкого доступу до домашніх каталогів.*

* команда cd ~

*Команда cd ~ переміщує користувача в його домашній каталог. Ця команда виконує ту саму дію, що й команда cd без аргументів.*

* команда cd (без аргумента)

*Команда cd (без аргумента) також переміщує користувача в його домашній каталог. Ця команда виконує ту саму дію, що й команда cd ~.*

* команда cd ..

*Команда cd .. переміщує користувача на один каталог назад від поточного. Наприклад, якщо користувач знаходиться в каталозі "/home/user/Documents", команда "cd .." перемістить його в каталог "/home/user".*

* команда cd ../..

*Команда cd ../.. переміщує користувача на два каталоги назад від поточного. Наприклад, якщо користувач знаходиться в каталозі "/home/user/Documents", команда "cd ../.." перемістить його в каталог "/home".*

* команда cd –

*Команда cd - переміщує користувача в останній каталог, в якому він перебував раніше. Наприклад, якщо користувач знаходиться в каталозі "/home/user/Documents", а перед цим він був в каталозі "/home/user", команда "cd -" перемістить його назад у каталог "/home/user".*

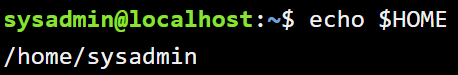
***Готував матеріал студент Заїка С.В.***

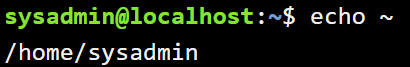
**Відповіді на контрольні запитання**

1. Як можна переглянути шлях до домашньої директорії користувача за допомогою команди echo? Існує 2 способи, наведіть обидва приклади у терміналі (відповідь є у матеріалах академії cisco на сайті netacad.com)

*Шлях до домашньої директорії користувача можна отримати за допомогою команди echo та спеціальної змінної середовища $HOME. Ось два способи отримання шляху до домашньої директорії користувача у терміналі:*

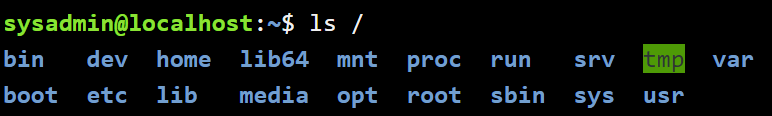
*Використання спеціальної змінної середовища $HOME за допомогою команди echo:*

*Використання символу тильда ~ в поєднанні з командою echo:*



1. Чи можна переглянути вміст кореневого каталогу, перебуваючи у домашньому каталозі користувача без переходу у кореневий каталог? Продемонструйте це в командному рядку.

*Так, можна переглянути вміст кореневого каталогу, не переходячи до нього, за допомогою абсолютного шляху до цього каталогу. Абсолютний шлях до кореневого каталогу в більшості Linux систем зазвичай є /. Отже, щоб переглянути вміст кореневого каталогу, не змінюючи поточного каталогу, можна використовувати наступну команду:*

*Ця команда виведе список файлів та папок, що знаходяться у кореневому каталозі. При цьому, користувач буде перебувати у своєму домашньому каталозі.*

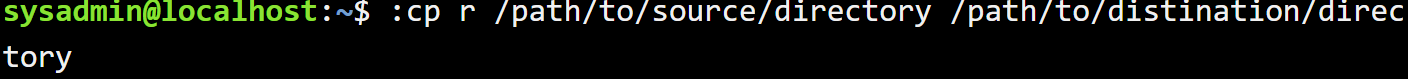
1. Яким чином в терміналі можна додати інформацію в порожній файл?

*Використання команди echo та перенаправлення виводу у файл. Наприклад, команда нижче додасть рядок "Hello World" до порожнього файлу myfile.txt:*



1. Як скопіювати та видалити існуючий каталог? Чи буде відмінність в командах, якщо каталог буде не порожній при цьому.

*Копіювання порожнього каталогу можна виконати за допомогою команди cp з опцією -r:*

*Видалення порожнього каталогу можна виконати за допомогою команди rm з опцією -r* *:*

*Якщо каталог містить файли та/або підкаталоги, то для їх копіювання та видалення треба використовувати опцію -r разом із командами cp та rm, як у вищезгаданих прикладах. Важливо бути обережним і переконатися, що ви використовуєте правильну команду та правильний шлях до каталогу, щоб уникнути втрати даних.*

1. У якому з наведених нижче прикладів відбувається переміщення файлу? його перейменування? одночасно обидві дії?

* mv /work/tech/comp.png. /Desktop
* mv /work/tech/comp.png. /work/tech/my\_car.png
* mv /work/tech/comp.png. /Desktop/computer.png

*mv /work/tech/comp.png. /Desktop: у цьому прикладі файл comp.png переміщується в директорію /Desktop.*

*mv /work/tech/comp.png. /work/tech/my\_car.png: у цьому прикладі файл comp.png перейменовується на my\_car.png в тій самій директорії /work/tech.*

*mv /work/tech/comp.png. /Desktop/computer.png: у цьому прикладі файл comp.png переміщується в директорію /Desktop та одночасно перейменовується на computer.png.*

*Отже, переміщення файлу відбувається в першому та третьому прикладах, перейменування - у другому та третьому, а одночасно обидві дії - тільки у третьому прикладі.*

**Висновки**

В ході виконання лабораторної роботи ми отримали практичні навички роботи з командною оболонкою Bash, також познайомились з базовими командами навігації по файловій системі та з базовими командами для керування файлами та каталогами.