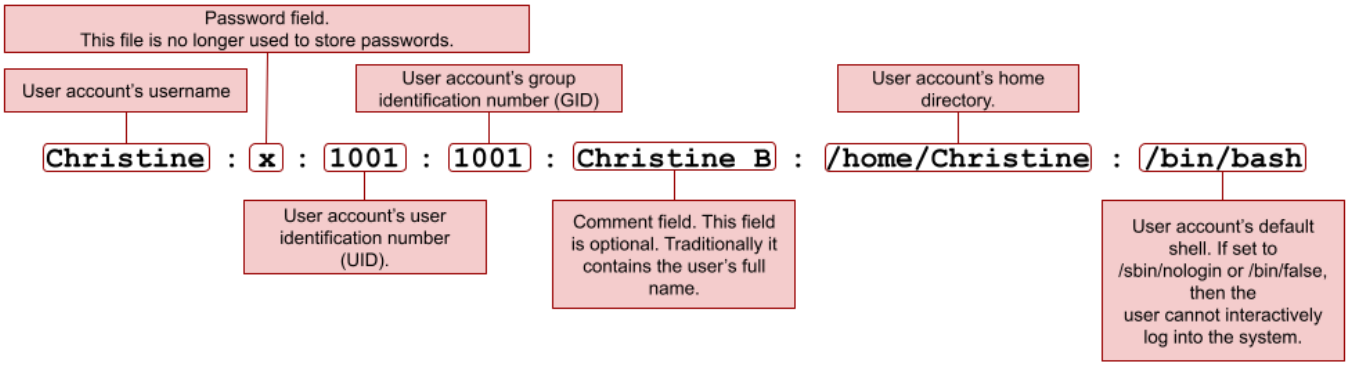
# Lab 5 | Handling Text Files

The `/etc/passwd` file stores user's account information. Each account data occupies a single line in a file and when a new account is created a new entry with the new user's information is added. The `/etc/passwd` records contain several fields, 7 in total. These fields are described in the image below. In this lab, we are going to use the passwd file in combination with the commands for handling text files. The goal is to demonstrate how to use these commands successfully for administering a system.




## Pre Work:

- Create a directory called `Lab5` and change your current working directory to `Lab5`. You will complete the entire lab from this directory.
- Before you start working on this lab, prepare your submission file and make an initial commit.
- For examples of commands go to: Linux Commands
- For the presentation go here

## Question 1

Cat, head and tail commands are used for displaying the content of a file.

1. Display the content of the `/etc/passwd` file.
2. Display the content of the `/etc/passwd` file in reverse order.
3. Display the content of the `/etc/passwd` file with line numbers and the $ to indicate the end of every line.
4. Display the first 5 lines of a the `/etc/passwd` file.
5. Display the last 5 lines of the `/etc/passwd` file.

> Take a screenshot of **YOUR TERMINAL ONLY** showing all the commands that you used to complete this question

## Question 2

The cut command is very useful when working with files that are already formatted using a field separator. The cut command can show specific information about each line of text in a given file.

1. Display the first field of the `/etc/passwd` file.
2. Display the last 5 users in the `/etc/passwd` file.

3. Display a list of all the users and their designated login shell separated by an `=` sign.
4. The sort command is another amazing tool in any linux user's tool box. Sort allows you to display data in a given order. Cut the first and 3rd field of the `/etc/passwd` field and sort the output.
5. Repeat the previous command but this time only show the last 5 entries.

> Take a screenshot of **YOUR TERMINAL ONLY** showing all the commands that you used to complete this question

## Question 3

The wc command is used to count the number of lines, characters and words in a file.

1. How many lines does the `/etc/passwd` file have?
2. How many words does the `/etc/passwd` file have?

Grep is the holy grail of command line tools. It allows us to search for specific strings inside a file. Here are some examples of the usage of grep: https://robertalberto.com/linuxcommands/commands/grep.html

3. How many users can login with the `/bin/bash` shell?
4. How many users have the `/sbin/nologin` shell assigned?
5. Display your user's information in `/etc/passwd` file

> Take a screenshot of **YOUR TERMINAL ONLY** showing all the commands that you used to complete this question

## Question 4

The ip command is used to manage network interfaces. To display the current NICs configuration, type: `ip ad` which is short for `ip address`. We are going to use the commands we learned to parse the output of the ip command.

 ip address cmd

1. Run the `ip ad` command and display all the lines that match the string `inet`. How many lines did you get?
2. Run the `ip ad` command and display all the lines that match the string `inet6`. Display the output in reverse order.
3. Run the `ip ad` command and display all the lines that match the string `inet` or `inet6` sort the output and save it to a file.
4. Run the `ip ad` command and display only the 3rd line that matches the string `inet`.
5. Run the `ip ad` command and display all the ipv4 addresses sorted.

> Take a screenshot of **YOUR TERMINAL ONLY** showing all the commands that you used to complete this question

## Question 5

1. Run the following command and save the output to a markdown file: `echo "# Information about my pc"`. You can use any naming convention you want for the file as long as it is a markdown file.

2. Run the following command and append the output to the markdown file you created earlier: `echo "## CPU Information"`
3. The `lscpu` command displays a lot of information about the CPU the computer has. Use the `lscpu`, `grep`, and the pipe (|) to extract, and append to the file you created earlier, the following information from the output of the `lscpu` command:
    - Architecture
    - Threads
    - Cores
    - Model name
    - CPU Frequency
    - Virtualiation technology supported
4. Run the following command and append the output to the markdown file you created earlier: `echo "## RAM Information"`
5. The command `lshw -c memory` displays information about the RAM installed in your system. Extract and append to the file the following information:
    - Memory size:
6. Display the content of the file you created earlier showing all the data that has been appended so far.

> Take a screenshot of **YOUR TERMINAL ONLY** showing all the commands that you used to complete this question