ASSIGNMENT – 1

Q.1: Introduction to the operating system with key functions.

-> An operating system (OS) is a software program that acts as an intermediary between the computer hardware and the user or application software. It provides a platform for running and managing computer programs, coordinating system resources, and facilitating communication between software and hardware components.

The key functions of the operating system are:

i) **Process Management** : It manages and schedules processes (or tasks) running on the computer system, allocating resources such as CPU time and memory to ensure efficient execution.

ii) **Memory Management**: The OS controls and organizes the computer's memory resources, allocating memory to different processes and managing virtual memory when the physical memory is limited.

iii) **File System Management**: It provides a hierarchical structure for organizing and storing files on storage devices, allowing users and programs to create, access, and manage files and directories.

iv) **Device Management**: The OS controls input and output devices, such as keyboards, mice, printers, and network interfaces, facilitating communication between these devices and software applications.

v) **User Interface**: It provides a means for users to interact with the computer system, presenting information and enabling input through interfaces such as command-line interfaces (CLI) or graphical user interfaces (GUI).

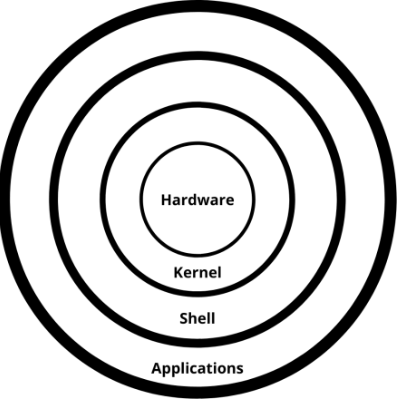
vi) **Security and Protection**: The OS enforces security measures to protect the system and user data, including user authentication, access control, and encryption.

vi) **Networking**: In modern operating systems, networking capabilities are integrated, allowing the system to connect and communicate with other computers and devices over networks. Different types of operating systems exist, including general-purpose operating systems like Windows, macOS, and Linux, as well as specialized operating systems for mobile devices, embedded systems, and servers.

Q.2. Introduction to the Unix/Linux (Architecture).

A. Unix has a graphical user interface similar to the Windows operating system that makes it easy for navigation and a good supportive environment. The internal design view of this operating system can be known from its architecture. The architecture is explained below:

The architecture of this operating system is four layered. It consists of **Hardware**, **Kernel**, System Call interface(**Shell**) and **Application** libraries/tools, utilities, etc…



1. **Hardware**

Hardware is the most simple and least powerful layer in the Unix Architecture. Hardware is the components that are humanly visible. Whatever hardware is connected to a Unix operating system-based machine, comes in the hardware layer.

1. **Kernel**

For this operating system, Kernel is the central core that interacts directly with the hardware of the system. The main functions of Kernal are-

i) Computer hardware such as memory, disc, printers, etc.. are controlled by the kernel.

ii) The kernel schedules the processes, control and executes various user-defined tasks.

iii) Manages the data storage and control the computer accesses by several users.

iv) The kernel is composed of several sub-components such as configurations including boot code, device drivers to control hardware, header files.

1. **Shell**

It is the interface between the user and the kernel. Users can interact with the shell using shell commands. Shell has two main responsibilities which include interpreting the commands given by the users and execute them using the kernel, providing programming ability to the users to write shell commands for a shell script to perform specific tasks.

1. **Applications**

It is the outermost layer that executes the given external applications. UNIX distributions typically come with several useful applications programs as standard. For Example: emacs editor, StarOffice, xv image viewer, g++ compiler etc.

Q.3. Concept of Shell.

-> A shell is a special user program that provides an interface for the user to use operating system services. Shell accepts human-readable commands from users and converts them into something which the kernel can understand. It is a command language interpreter that executes commands read from input devices such as keyboards or from files. The shell gets started when the user logs in or starts the terminal.

Q.4. Types of Shell.

-> Z shell, Bourne shell, Korn shell, Bash, C shell, Tcsh, CSH, Ksh shell,

Restricted shell, Scheme shell, SH, Dash shell, Fish, Friendly interactive shell,

PowerShell, Unix shells, Almquist Shell, Exotic shells, Ion shell, Shell script

Q.5. Command structure.

->

Q.6. Introduction of basic linux commands (sudo, ls, pwd, mkdir, rmdir, rm, cd, cp, wc, mv, cmp, passwd, who, uname).

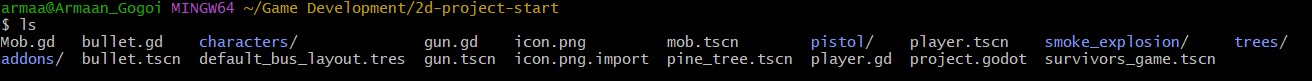
->

sudo

The sudo(**S**uper **U**ser **DO**) command provides an efficient way to grant users the special rights to execute commands at the system (root) level.With sudo, we’ll be able to execute administrative tasks without switching users.

ls

The ls command without any options lists files and directories in a plain format without displaying much information like file types, permissions, modified date, etc.



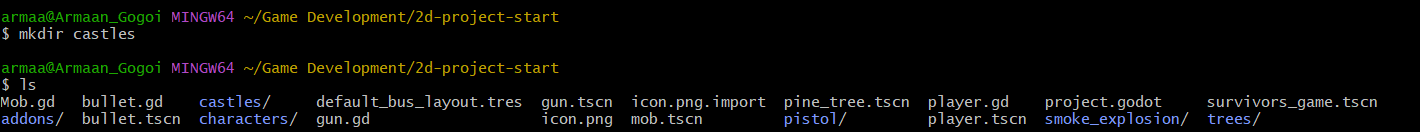
pwd

The pwd command (print working directory) writes the full pathname of the current working directory to the standard output.



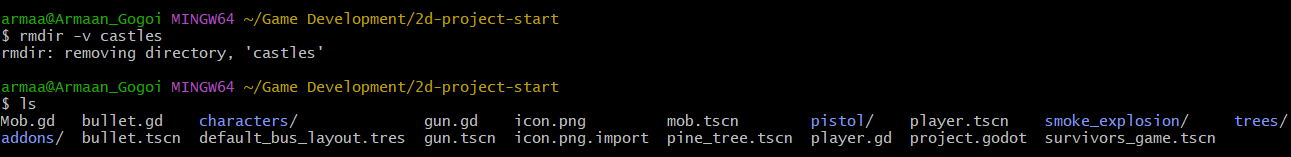
mkdir

The mkdir command in Linux/Unix is a command-line utility that allows users to create new directories. mkdir stands for "make directory." With mkdir , we can also set permissions, create multiple directories at once, and much more.



rmdir

The rmdir command is designed for removing empty directories. Unlike the rm command, which can delete both files and directories, rmdir focuses solely on directories.



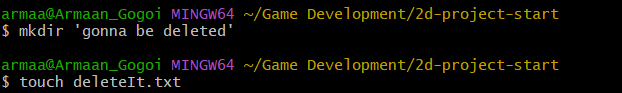
rm

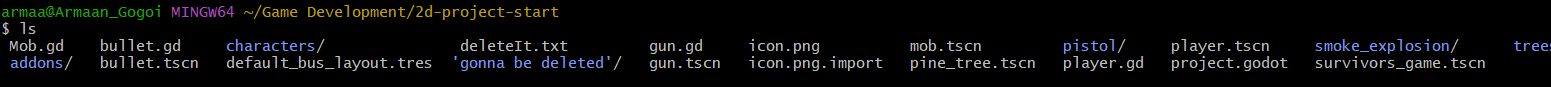
The rm (remove) command is used to delete files and directories.

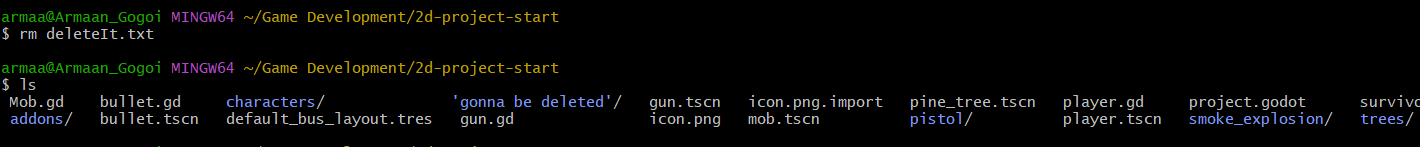
Adding the -i flag like many of the other commands, will give us a prompt on whether we want to actually remove the files or directories.

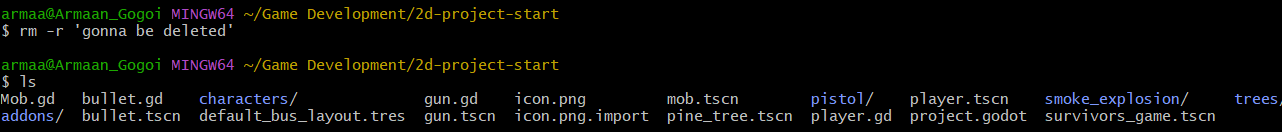
We can’t just rm a directory by default, we’ll need to add the -r flag (recursive) to remove all the files and any subdirectories it may have

$ rm -r directory



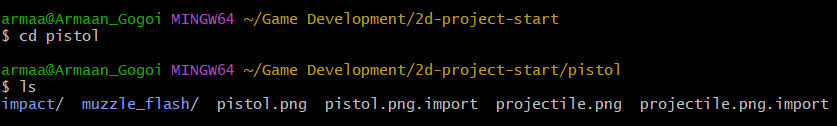






cd

The 'cd' command allows users to change their current working directory within the file system.



cp

The cp command is used to copy directories and files. The command contains three primary operation modes, represented by the argument types shown to the program to copy a file to other files, multiple files to any directory, or to copy the whole directories to other directories.

To copy a file into the same directory syntax will be,

cp **<existing** file name**>** **<new** file name**>**

To copy a file to a different directory

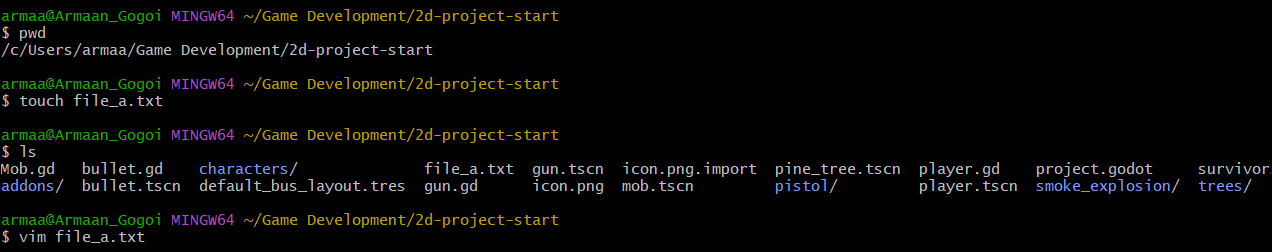
cp <existing file name> <destination directory>

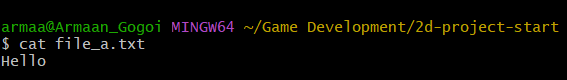
Option **'r'** with the copy command can be used to copy a directory including all its content from a source directory to the destination directory.

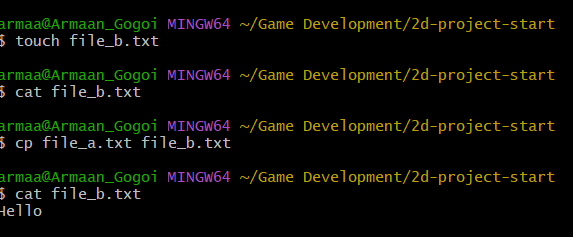
cp -r **<sourceDirectory>** **<destinationDirectory>**

Multiple files or directories can be copied to a destination directory at once. In this case, target must be a directory. To copy multiple fileswe can use **wildcards** (cp \*.extension) having same pattern.

cp \*.**<extension>** **<destinationDirectory>**







wc

The **wc** command counts the number of lines, words, and bytes in the files specified by the *File* parameter. The command writes the number of newline characters, words, and bytes to the standard output and keeps a total count for all named files.

To display the line, word, and byte counts of a file, the syntax is

wc <filename>



mv

The **mv** command is a UNIX utility for renaming and relocating files and directories in a filesystem.

* **Rename a file**. Type the source filename and the destination filename:

mv [options] [source-file] [destination-file]

* **Rename (or move) a directory**. Type the source directory and the destination directory. If a destination directory exists, **mv** moves the source directory to the specified destination.

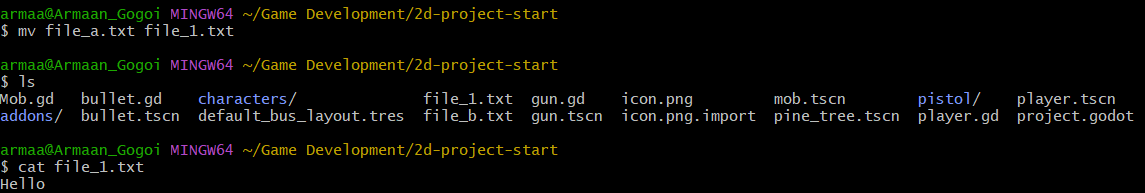
mv [options] [source-directory] [destination-directory]

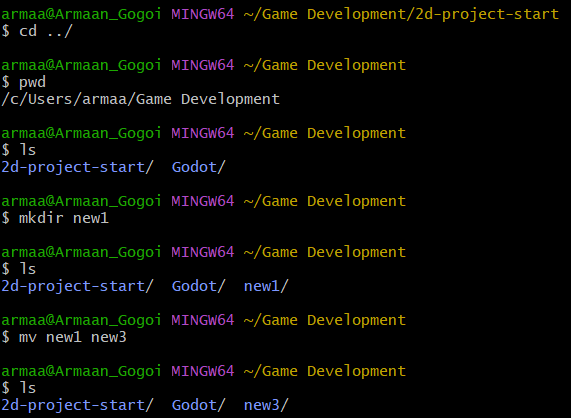
* **Move a file, preserving the filename**. Type the filename followed by the destination directory.

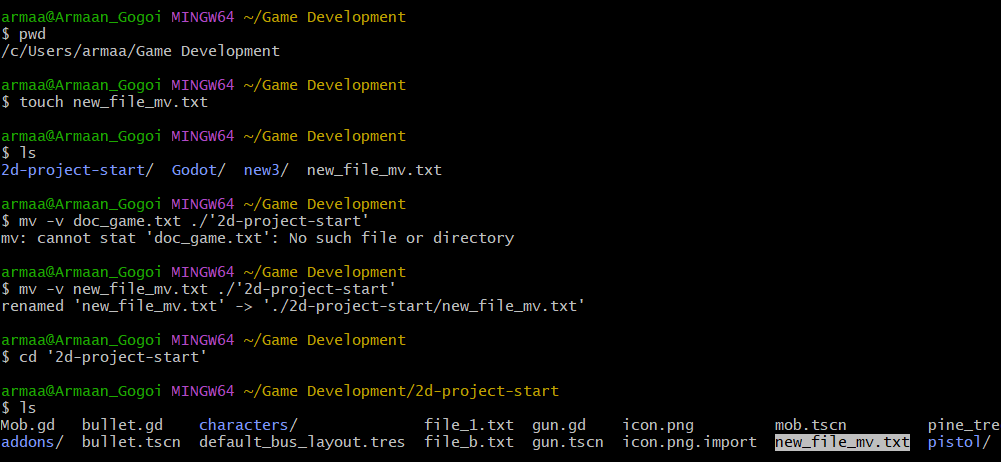
mv [options] [file] [destination-directory]

* **Move a file and change its name**. Type the filename followed by the destination directory and a new filename.

mv [options] [file1] [directory/file2]



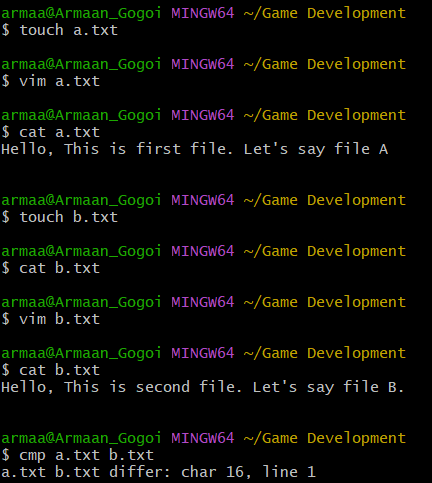






cmp

compares the contents of two files and reports the first character that differs.



passwd

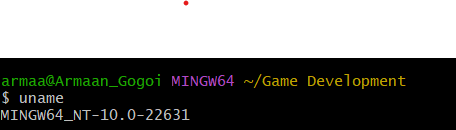
The passwd command in Linux is used to change a user's password. It can be used by both regular users and administrators, depending on the desired operation.

who

The who command in Linux is used to display information about the users currently logged into the system. It shows details such as the username, terminal, date and time of login, and the IP address or hostname from which the user is logged in.

uname

The uname command in Linux is used to display system information. It provides details about the system's hardware, operating system, and kernel. By default, uname outputs the kernel name, but it can also provide additional information with various options.



Q.7. How to install, update, upgrade and remove any package in linux (apt-get).

->

Q.8. >, >> option for directing the output of a command.

->