Cd:

* A command that allows the change of the current working directory. This allows the user to have a more organised code for the current project they are working on. To use it the user must simply input the code “cd [option] [directory]”,The items in square brackets are optional. When used without specifying any directory name, cd returns the user to the previous current directory. This provides a convenient means of toggling between two directories.

Mkidr:

* This code allows the user to create a directory. This allows the user to segregate the code within the project, within different directories that cotain differen information about files. To use this code the user must simply input “mkdir [option] directory\_name(s)”, with this you can create a multiple amount of directories.

Cp:

* This command allows you to copy files and directories. This allows the user to create an independent copy of the original file, which can allow the, to access the smae data without tampering the original. To use this code the user must simply input “cp [options] name new\_name”, though there is a when using this since if two files share the same name there is a chance that it can overwrite the previous file.

Pwd:

* This command prints out the whole current path of the whole current directory you are working on. Allows the user to acces the command line interface, which allows you to get a full review of the file that they were working on. To use this coode the user simply inputs “ pwd [option]”, which confirms current directory has actually been changed to what the user intended after using cd.

Mv:

* This command allows the user to rename or move files and directories. This mostly helps the user move diferent values to different directories.To use this coode the user simply inputs “ [options] argument(s)”, The arguments are names of files and directories. If two file names are provided as arguments, mv renames the first as the second. If a list of arguments is provided and the final argument in the sequence is the name of an existing directory, mv moves all of the other items into that directory. If the final argument is not an existing directory and more than two arguments are provided, an error message is returned.

Rm:

* This command allows the user to delete files and directories. This allows the user to remove unwanted files or directories, that causes bugs within the file/directories. To use this code the user must simply input the command “rm [options] [-r directories] filenames”, the items in square brackets are optional. When used just with the names of one or more files, rm deletes all those files without requiring confirmation by the user.

History:

* History command is used to view the previously executed command. This allows the user to simply see if there are any previous errors within the file itself. To use this command the user simply must input “history”, command within the code.

Home directory and ~:

* This is the personal repository of the project, and serves as the main hub of the files and directories. This acts as the root directory of the code and allows the user to have a centralized hub for every directory he or she has. To access this root the user must simply inpuy “cd home” or “cd~”.

File Paths:

* File Paths are the main address of objects within a code be it object or directories. This allows the user to simply organize the objects within the program. To access these paths the user can use the code pwd or echoPATH.

Using the tab key to complete file path:

* This allows the user to be suggested what coomands they can use or finish commands that they are working on. This allows the user to cut time and allows them tol quicly finish their code. To access this function you can simply hit tab as you finish a line of code.

Up and down arrow key for history:

* This allows the user to look through the history of the file/directory. Allows the user to have an easier time looking back at the history of the file/directort. This can be accessed in two ways history-search-forward (), which searches forward, or history-search-backward (), which searches backward, or